

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA E
INFORMÁTICA INDUSTRIAL**

CECILIA HAYDEE VALLEJOS DE SCHATZ

**SISTEMA INTELIGENTE PARA MONITORAMENTO E PREDIÇÃO
DO ESTADO CLÍNICO DE PACIENTES BASEADO EM LÓGICA
FUZZY E REDES NEURAS**

TESE

CURITIBA

2014

CECILIA HAYDEE VALLEJOS DE SCHATZ

**SISTEMA INTELIGENTE PARA MONITORAMENTO E PREDIÇÃO
DO ESTADO CLÍNICO DE PACIENTES BASEADO EM LÓGICA
FUZZY E REDES NEURAIIS**

Tese de doutorado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de “Doutor em Ciências”.
Área de Concentração: Engenharia Biomédica.

Orientador: Prof. Dr. Fábio Kurt Schneider

Co-orientador: Prof. Dr. Paulo José Abatti

CURITIBA

2014

Dados Internacionais de Catalogação na Publicação

V182s Vallejós de Schatz, Cecília Haydée
2014 Sistema inteligente para monitoramento e predição do estado
clínico de pacientes baseado em lógica *fuzzy* e redes neurais /
Cecília Haydée Vallejós de Schatz.-- 2014.
141 f.: il.; 30 cm

Texto em português, com resumo em inglês
Tese (Doutorado) - Universidade Tecnológica Federal do
Paraná. Programa de Pós-Graduação em Engenharia Elétrica e
Informática Industrial, Curitiba, 2014
Bibliografia: f. 133-141

1. Redes neurais (Computação). 2. Lógica difusa. 3. Redes de
sensores sem fio. 4. Sistemas inteligentes de controle. 5.
Sistemas de transmissão de dados. 6. Monitoramento de pacientes.
7. Software - Desenvolvimento. 8. Simulação (Computadores). 9.
Engenharia elétrica - Teses. I. Schneider, Fábio Kurt, orient.
II. Abatti, Paulo José, coorient. III. Universidade Tecnológica
Federal do Paraná - Programa de Pós-Graduação em Engenharia
Elétrica e Informática Industrial. IV. Título.

CDD 22 -- 621.3

Título da Tese Nº. 95

**“Sistema Inteligente para Monitoramento e Predição
do Estado Clínico de Pacientes Baseado em Lógica
Fuzzy e Redes Neurais”**

por

Cecilia Haydée Vallejos de Schatz

Orientador: Prof. Dr. Fábio Kurt Schneider

Esta tese foi apresentada como requisito parcial à obtenção do grau de DOUTOR EM CIÊNCIAS – Área de Concentração: Engenharia Biomédica, pelo Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial – CPGEI – da Universidade Tecnológica Federal do Paraná – UTFPR, às 14h do dia 18 de fevereiro de 2014. O trabalho foi aprovado pela Banca Examinadora, composta pelos doutores:

Prof. Dr. Fábio Kurt Schneider
(Presidente – UTFPR)

Prof.^a Dr.^a Márcia Ito
(IBM Research Brazil)

Prof. Dr. Júlio Cesar Nievola
(PUCPR)

Prof. Dr. Hugo Vieira Neto
(UTFPR)

Prof. Dr. Joaquim de Mira Júnior
(UTFPR-PG)

Visto da Coordenação:

Prof. Ricardo Lüders, Dr.
(Coordenador do CPGEI)

“ A Folha de Aprovação assinada encontra-se no Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial (CPGEI)”.

AGRADECIMENTOS

Aos meus orientadores Prof. Dr. Fábio K. Schneider e Prof. Dr. Paulo J. Abatti, pelo incentivo e orientação durante a realização do meu trabalho.

Agradeço também à Prof^ª. Dr^ª. Valéria Arruda e à Profa. Dra. Myriam Delgado, pelas aulas que deram início a este trabalho e pela orientação nesta pesquisa.

Agradeço ao Prof. Dr. Hipólito Carraro Junior, médico intensivista do Hospital de Clínicas de Curitiba; ao Dr. Luciano Machado, médico intensivista do Hospital São Vicente de Curitiba; à enfermeira Melissa de Costa, chefe de enfermagem da UTI do Hospital São Vicente de Curitiba e à enfermeira Karin Godarth, chefe do centro cirúrgico do Hospital do Idoso Zilda Arns. Todos contribuíram enormemente na base deste trabalho.

A todo o pessoal da direção do Hospital do Idoso Zilda Arns, em especial à assessora de operações Leila Dufflot, e a meus colegas da área de Engenharia Clínica pela compreensão e suporte aos meus estudos.

Agradeço novamente ao meu orientador Prof. Dr. Fábio K. Schneider e à minha colega no Hospital do Idoso Zilda Arns, Dr^ª. Oraide Woehl pelo auxílio na correção deste trabalho.

Agradeço aos professores e pesquisadores da banca examinadora pela atenção e contribuições oferecidas.

Ao meu marido Michael pelo carinho, dedicação e compreensão na longa jornada percorrida.

Aos meus filhos e a toda minha família e amigos próximos pelo grande amor demonstrado no apoio, paciência e compreensão dedicados durante todo o período de estudos.

À CAPES pelo apoio financeiro.

RESUMO

VALLEJOS DE SCHATZ, CECILIA HAYDÉE. Sistema inteligente para monitoramento e predição do estado clínico de pacientes baseado em lógica *fuzzy* e redes neurais. 2014. 141 f. Tese de Doutorado. Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial da Universidade Tecnológica Federal do Paraná. Curitiba, 2014.

O conforto e a liberdade de movimentos de pacientes com doenças crônicas e que têm que ser continuamente monitorados é um tema que tem incentivado o desenvolvimento de novas tecnologias como as redes de sensores corporais sem fios (WBAN) e novas áreas de pesquisa como a telemedicina. Além disso, a incorporação de *software* inteligente que permite simular o raciocínio dos especialistas, auxiliá-los na tomada de decisões e detectar com antecedência condições anormais ou tendência ao desenvolvimento de determinadas doenças, abre um campo ainda maior de pesquisas, como o campo da Inteligência Artificial na Medicina (AIM). O monitoramento de pacientes por meio de equipamentos sem fios, em conjunto com a tecnologia AIM, permite desenvolver soluções práticas para monitorar pacientes sem descuidar de seu conforto. Nesta tese foram pesquisadas técnicas inteligentes para o desenvolvimento de uma aplicação que permita monitorar cinco sinais vitais de pacientes sem que eles precisem usar leitos hospitalares. Em uma primeira etapa, os procedimentos médicos tipicamente usados pelos especialistas para avaliar um paciente foram estudados e transformados em regras para o modelo *fuzzy*. O modelo *fuzzy* proposto permite analisar o estado clínico presente do paciente e criar as saídas desejadas (*targets*) que permitam treinar as redes neurais artificiais. Posteriormente foi desenvolvido um modelo neural que, analisando os dados atuais e saídas anteriores do paciente, permite prever o seu estado clínico futuro próximo. A fim de achar a metodologia mais exata, cinco redes neurais artificiais foram analisadas e comparadas umas às outras. As redes Elman MISO, Elman MIMO, e NNARX – totalmente conectadas e podadas – foram testadas. O modelo *fuzzy* teve um excelente resultado concordando com as respostas dadas pelos especialistas em 99,76% dos casos. Depois de analisar as redes propostas no conjunto de validação, os resultados revelaram que unicamente a rede NNARX podada pode oferecer a mais alta acurácia de 99,82%, enquanto os outros modelos degradam o seu desempenho em até 35%. As técnicas de parada antecipada para o treinamento junto com a obtenção de valores médios de MSE, FPE e coeficientes de correlação conseguiram obter as melhores topologias de cada tipo de rede, fazendo quase desnecessária a sua poda. As redes NNARX e P-NNARX conseguiram resultados bem melhores que as redes restantes, mas a acurácia na rede P-NNARX observou um aumento de 1,27% em relação à rede NNARX. Como conclusão, pode-se dizer que, para este caso particular, as redes NNARX capturam a essência do sistema dinâmico não linear muito melhor do que as redes Elman. Finalmente, a rede P-NNARX foi a escolhida para a implementação do sistema inteligente proposto nesta tese. A sua acurácia foi de 99,25% para uma predição no tempo $(t + d)$, onde $d = 1$ segundo, utilizando os dados de 30 novos pacientes. Foram feitas mais provas com períodos de predição maiores e o sistema demonstrou uma ligeira diminuição na acurácia, chegando a 94,58% para $d = 60$ segundos, mas ainda ficando na faixa dos 90%. Os resultados demonstram o alto nível de generalização do sistema e o excelente desempenho na predição dos três estados clínicos do paciente (estável, semiestável e instável). Pretende-se que este sistema inteligente possa ser usado como ferramenta para a medicina preventiva em pacientes crônicos.

Palavras-chave: Redes Neurais Artificiais. *Fuzzy* Logic. Redes Neurais Recorrentes. Redes Neurais Autorregressivas com entradas exógenas. Redes Elman.

ABSTRACT

VALLEJOS DE SCHATZ, CECILIA HAYDÉE. Intelligent system for monitoring and forecasting the clinical state of patients based on *fuzzy* logic and artificial neural networks. 2014. 141 p. Doctoral thesis. Graduate School of Electrical Engineering and Applied Computer Science of the Federal University of Technology - Paraná. Curitiba, 2014.

The comfort and freedom of movements of patients that have to be continually monitored is a theme that has motivated the development of new technologies such as networks of wireless body sensors (WBAN) and new research areas such as telemedicine. In addition, the incorporation of intelligent software to simulate the reasoning of experts, assist them in decision making and in early detection of abnormal conditions or tendencies to develop certain diseases, opens an even larger field of research, such as the field of Artificial Intelligence in Medicine (AIM beings its acronym in English). Patient monitoring through wireless equipment and AIM technology allows to develop practical solutions to control patients in environments outside of clinics or hospitals. In this thesis, intelligent tools were used for the development of an application that allows monitoring of five vital signs of patients without them being present in a hospital bed. In a first step, typical medical procedures used by specialists for evaluating a patient were studied and transformed into rules for the *fuzzy* model. The proposed *fuzzy* model allows the analysis of the current state of the patient to create the desired outputs (targets) that are used to train the artificial neural networks. Then, a neural model was developed which, by analysing current and historic patient data, forecasts patients' clinical status in the near future. In order to find the most exact methodology, five artificial neural networks were analyzed and compared with each other using thousands of real patient data sets. Elman MISO, Elman MIMO and NNARX – fully connected and pruned – were tested. The *fuzzy* model answered in a excellent form, agreeing in 99.76% to the answers given by the experts. After analyzing the proposed networks in the validation dataset, it was discovered that the pruned NNARX can offer the highest overall accuracy of 99.82%, whereas the others show a decrease of up to 35%. Through techniques such as early stopping for the training with the search of the mean of MSE, FPE and correlation coefficients it was possible to achieve the best topologies of every network type, making their pruning almost unnecessary. The fully connected NNARX and the P-NNARX achieved much better results than other networks, but an increase of 1.27% was observed in the overall accuracy of the pruned network with respect to the NNARX. It can be said that for this particular case, NNARX networks capture the essence of the non-linear dynamic system much better than Elman. Finally, the P-NNARX model was chosen for the implementation of the proposed smart system. Its overall accuracy was of 99.25%, for the prediction time ($t + d$), with $d = 1$ second, by using unseen data of 30 new patients. More tests made with longer prediction periods demonstrate a slight decrease in the overall accuracy reaching up to 94.58% for $d = 60$ seconds. Nevertheless, it still remained over 90%. Results demonstrate the high generalization level of the system and its excellent performance in predicting the three possible patient conditions (stable, semi-stable, unstable). The next step is to turn this intelligent system into an usefull tool for preventive medicine for chronic patients.

Keywords: Artificial Neural Network. *Fuzzy* Logic. Recurrent Neural Network. Neural Network Auto-Regressive model with eXogenous inputs. Elman Networks.

LISTA DE FIGURAS

Figura 1 – Arquitetura de Rede de Sensores Médicos sem Fios	17
Figura 2 – Representação das pressões sistólicas e diastólicas ao longo de dois ciclos cardíacos.	23
Figura 3 – Curva da pressão arterial	24
Figura 4 – Função triangular de pertinência	29
Figura 5 – Função trapezoidal de pertinência.....	29
Figura 6 – Lógica convencional (a) vs. lógica <i>fuzzy</i> (b).....	30
Figura 7 – Sistema de inferência <i>fuzzy</i>	33
Figura 8 – Exemplo de um problema de classificação.	36
Figura 9 - Exemplos de RNA <i>feedforward</i>	38
Figura 10 – Topologia de uma rede <i>feedforward</i> com viés e indicação dos pesos	39
Figura 11 – Topologias das redes <i>feedforward</i> e das redes recorrentes	40
Figura 12 - Redes Recorrentes Simples (Elman/Jordan) e completamente conectada (Hopfield)	41
Figura 13 - Gradiente descendente em um critério quadrático unidimensional com diferentes taxas de aprendizado.....	46
Figura 14 - Espaço de mapeamento para redes com nenhuma, uma ou duas camadas escondidas.....	49
Figura 15 – Parada antecipada.....	52
Figura 16 – Passos do enfoque da modelagem.....	55
Figura 17 – Rede Neural com vetor de regressão NNARX	58
Figura 18 – Dados obtidos da tela de Physionet (MIMIC – Reg.: 441n).....	72
Figura 19 - Divisão dos conjuntos de dados dos dez pacientes.....	73
Figura 20 – Fluxograma com as distintas etapas da criação do modelo <i>fuzzy</i>	76
Figura 21 – Representação dos pares de cada forma de função de pertinência.....	78
Figura 22 – Funções de pertinência para a Frequência Cardíaca	79
Figura 23 – Funções de pertinência para a Pressão Sistólica Não Invasiva	79
Figura 24 – Funções de pertinência para a Pressão Diastólica Não Invasiva.....	80
Figura 25 – Funções de pertinência para a Pressão Média Não Invasiva.....	81
Figura 26 – Funções de pertinência para a Saturação de Oxigênio.....	81
Figura 27 – Funções de pertinência para a saída de Condição do Paciente	82
Figura 28 – Rede Elman MISO	86
Figura 29 - Desempenho dos distintos algoritmos de treinamento - Rede Elman MISO	89
Figura 30 – Resultados da poda da rede E-MISO	91
Figura 31 – Topologia da rede PE-MISO.....	91
Figura 32 – Rede E-MIMO com inclusão do Modelo <i>Fuzzy</i> para predição da saída final	92
Figura 33 - Desempenho dos distintos algoritmos de treinamento - Rede Elman MIMO	96
Figura 34 – Estrutura da rede NNARX	98
Figura 35 - Análise da quantidade de neurônios ocultos para a rede NNARX.....	102
Figura 36 – Topologia da rede neural NNARX(6,3,1) totalmente conectada.....	103
Figura 37 – Resultados da poda da rede NNARX.....	104
Figura 38 – Topologia da rede PNNARX	105
Figura 39 – Coincidências e divergências entre resultados do modelo <i>fuzzy</i> e respostas esperadas pelos especialistas	107
Figura 40 – Saída do modelo E-MISO - Dados de treinamento e de validação.....	111
Figura 41 – Saída da rede PE-MISO - Dados de treinamento e validação.....	114

Figura 42 – Validação da rede neural NNARX.....	116
Figura 43 – Funções de autocorrelação e de correlação cruzada – Rede NNARX.....	117
Figura 44 – Validação da rede neural P-NNARX.....	120
Figura 45 – Funções de autocorrelação e de correlação cruzada – Rede P-NNARX.....	121
Figura 46 – Curvas das diferentes predições – Sistema FNNARX.....	126
Figura 47 - Comparações da saída real vs. a saída predita dos pacientes Pac01, Pac02, Pac03, Pac04 e Pac07.....	127
Figura 48 - Comparações da saída real vs. a saída predita dos pacientes Pac10, Pac17, Pac20, Pac24 e Pac30.....	128

LISTA DE TABELAS

Tabela 1 – Monitoramento hemodinâmico – Principais variáveis hemodinâmicas	22
Tabela 2 - Exemplo de uma matriz de confusão de 2 classes.....	61
Tabela 3 – Guia para classificar a exatidão dos métodos avaliados	64
Tabela 4 – Registros de dados fisiológicos consultados no site Physionet	71
Tabela 5 - Exemplo de uma matriz de confusão de 3 classes.....	74
Tabela 6 – Valores dos sinais vitais não invasivos para um adulto saudável.....	77
Tabela 7 – Relação: Valores defuzzificados vs. Variáveis linguísticas de saída	84
Tabela 8 – Especificação das entradas e saídas do modelo Elman MISO	86
Tabela 9 – Diferentes combinações de redes Elman MISO	87
Tabela 10 – Análise das cinco melhores arquiteturas Elman MISO	90
Tabela 11 – Especificação das entradas e saídas do modelo Elman MIMO	92
Tabela 12 – Diferentes combinações de redes Elman MIMO.....	93
Tabela 14 – Resultados obtidos na busca dos regressores.....	99
Tabela 15 – Resultados obtidos na busca do melhor modelo NNARX	101
Tabela 17 – Comparação de resultados do modelo <i>fuzzy</i> com a opinião dos especialistas	108
Tabela 18 – Arquiteturas ótimas para cada modelo de rede – Dados de validação	108
Tabela 19 – Matriz de confusão da rede E-MISO	109
Tabela 20 – Matriz de confusão da rede PE-MISO.....	112
Tabela 21 – Análise das redes E-MISO e PE-MISO.....	113
Tabela 25 – Matriz de confusão do modelo NNARX	115
Tabela 26 – Matriz de confusão do modelo P-NNARX.....	118
Tabela 28 – Indicadores de desempenho das distintas arquiteturas de redes analisadas.....	122
Tabela 29 – Análise dos indicadores de desempenho do sistema FNNARX.....	124
Tabela 30 – Tendência dos indicadores com diferentes tempos de predição	126

LISTA DE SIGLAS

Ac	Acurácia
AIC	<i>Akaike's Information Criterion</i> (Critério de Informação de Akaike)
AIM	<i>Artificial Intelligence in Medicine</i> (Inteligência Artificial na Medicina)
AR	Autorregressivo (Modelo)
BFGS	<i>BFGS quasi-Newton Backpropagation</i> – Algoritmo de Treinamento
EEG	Eletroencefalograma
Es	Especificidade
FC	Frequência cardíaca
FN	Falsos Negativos
FP	Falsos Positivos
FPE	<i>Final Prediction Error</i> (Erro de Predição Final)
FVN	Fração de Verdadeiros Negativos
FVP	Fração de Verdadeiros Positivos
GDB	<i>Gradient Descent Backpropagation</i> – Algoritmo de Treinamento <i>Gradient Descent with Momentum and Adaptive Learning</i>
GDMALB	<i>Backpropagation</i> – Algoritmo de Treinamento
IA	Inteligência Artificial
LMB	<i>Levenberg-Marquardt Backpropagation</i> – Algoritmo de Treinamento
MA	Média Móvel (Modelo)
MDL	<i>Rissanen's Minimum Description Length</i>
MLP	<i>Multilayer Perceptron</i> – Percéptrons de multicamadas
MSE	<i>Mean Square Error</i> (Erro Quadrático Médio)
OBS	<i>Optimal Brain Surgeon</i>
PAD	Pressão Arterial Diastólica
PAM	Pressão Arterial Média
PAS	Pressão Arterial Sistólica
PDA	<i>Personal Digital Assistant</i>
PD	Pressão Diastólica
PM	Pressão Média
PS	Pressão Sistólica

QoS	<i>Quality of Service</i> (Qualidade de Serviço)
RB	<i>Resilient Backpropagation</i> – Algoritmo de Treinamento
RMSE	<i>Root-Mean Square Error</i> (Erro da Raiz Quadrada Média)
RNA	Rede Neural Artificial
SCGB	<i>Scaled Conjugate Gradient Backpropagation</i> – Algoritmo de Treinamento
Se	Sensibilidade
Se _{Estável}	Sensibilidade para o estado Estável
Se _{Semiestável}	Sensibilidade para o estado Semiestável
Se _{Instável}	Sensibilidade para o estado Instável
SO	Saturação de oxigênio
Sp	<i>Specifity</i> (Especificidade)
SpO ₂	Oximetria de pulso
TP	Temperatura
VP	Verdadeiros Positivos
VN	Verdadeiros Negativos
WMSN	<i>Wireless Medical Sensor Network</i> (Rede sem fios de sensores médicos)

LISTA DE ACRÔNIMOS

ANFIS	<i>Artificial Neural Fuzzy Inference System</i> (Sistema de Inferência Artificial Neuro-Fuzzy)
ARIMA	Autorregressivo de Média Móvel (Modelo)
ARMA	Autorregressivo Integrado de Média Móvel (Modelo)
ECG	Eletrocardiograma
MIMIC	<i>Multi-parameter Intelligent Monitoring for Intensive Care</i> (Monitoramento multiparamétrico Inteligente para Cuidados Intensivos)
MIMO	<i>Multiple Inputs Multiple Outputs</i> (Múltiplas Entradas Múltiplas Saídas)
MISO	<i>Multiple Inputs Single Output</i> (Múltiplas Entradas Saída Única)
NARMA	Rede Neural Autorregressiva de Média Variável
NARMAX	Rede Neural Autorregressiva de Média Variável com Entradas Exógenas
NNARX	<i>Neural Network Autorregressive with eXogenous inputs</i> (Rede Neural Autorregressiva com Entradas Exógenas)
UTI	Unidades de terapia intensiva
WBAN	<i>Wireless Body Area Network</i> (Rede sem fios de Área Corporal)
WLAN	<i>Wireless Local Area Network</i> (Rede sem fios de Área Local)

LISTA DE SÍMBOLOS

$\mu_A(x)$	Função de pertinência do conjunto <i>fuzzy</i> A
J_d	Erro de treinamento
t_d	Saída desejada
z_d	Saída real
$J(w)$	Soma dos erros de todas as unidades de saída da rede
Δw_{ij}	Ajuste de pesos do neurônio ij
η	Taxa de aprendizado
η_{opt}	Taxa de aprendizado ótima
Z_N	Conjunto de treinamento
Z_T	Conjunto de validação
$u(t)$	Vetor de entradas (sinais de entrada)
$y(t)$	Vetor de saídas (sinais de saída)
θ	Vetor de pesos
\hat{y}	Saída desejada
$\hat{y}(t)$	Vetor de saídas preditas
$\varphi(t)$	Vetor de entradas à rede neural
V_N	Erro de treinamento
V_T	Erro de validação
$\hat{r}_{U_\varepsilon}(\tau)$	Validação cruzada entre as entradas e o erro de predição
$\hat{r}_{\varepsilon\varepsilon}(\tau)$	Autocorrelação da predição
ε	Erro de predição
z	Ordem do modelo

SUMÁRIO

1 INTRODUÇÃO.....	15
1.1 MOTIVAÇÃO.....	16
1.2 DESCRIÇÃO DO SISTEMA.....	16
1.3 OBJETIVOS.....	17
1.4 CONTRIBUIÇÕES.....	19
1.5 ESTRUTURA DO TRABALHO.....	19
2 FUNDAMENTAÇÃO TEÓRICA.....	21
2.1 SINAIS VITAIS E MONITORAMENTO HEMODINÂMICO.....	21
2.1.1 Pressão Sanguínea.....	21
2.1.2 Frequência Cardíaca.....	25
2.1.3 Saturação de Oxigênio.....	25
2.2 INTELIGÊNCIA ARTIFICIAL NA MEDICINA.....	26
2.3 LÓGICA FUZZY.....	27
2.3.1 Representação <i>Fuzzy</i> do Conhecimento.....	28
2.3.2 Variáveis Linguísticas.....	30
2.3.3 Regras <i>Fuzzy</i>	31
2.3.4 Sistema de Inferência <i>Fuzzy</i>	32
2.3.5 Metodologias Utilizadas para o Projeto do Modelo <i>Fuzzy</i>	34
2.3.6 Aplicações <i>Fuzzy</i> na Área Biomédica.....	34
2.4 REDES NEURAIS ARTIFICIAIS (RNA).....	35
2.4.1 Redes de Alimentação Direta.....	37
2.4.2 Redes Neurais Recorrentes.....	39
2.4.3 Redes Recorrentes Simples.....	40
2.4.4 Redes Recorrentes Totalmente Conectadas.....	42
2.4.5 Aplicações de Redes Neurais Artificiais Recorrentes na Área Biomédica.....	43
2.4.6 Algoritmo <i>Backpropagation</i>	43
2.4.7 Estrutura Ideal da Rede – Número de Camadas e Neurônios Escondidos.....	47
2.4.8 Critérios de Parada.....	50
2.4.9 Técnicas para Diminuir o Sobreajuste.....	51
2.4.10 Poda da Rede.....	52
2.5 SÉRIES TEMPORAIS.....	53
2.5.1 Testes.....	54
2.5.2 Seleção da Estrutura do Modelo Neural.....	55
2.5.3 Estimação do Modelo.....	56
2.5.4 Validação do Modelo.....	57
2.5.5 Estrutura do Modelo NNARX.....	58
2.5.6 Seleção da Ordem do Modelo.....	59
2.5.7 Critérios de Parada para Redes NNARX.....	59
2.5.8 Aplicações em Séries Temporais.....	60
2.6 VALIDAÇÃO DOS RESULTADOS.....	60
2.7 REDES DE SENSORES BIOMÉDICOS.....	64
2.7.1 Características Físicas dos Sensores.....	65
2.7.2 Privacidade na Transferência de Dados Médicos.....	65
2.7.3 Tolerância a Falhas.....	66
2.7.4 Qualidade de Serviço (QoS).....	66
2.7.5 Segurança e Confidencialidade dos Dados.....	66
2.7.6 Aplicações de Sensores Médicos.....	67

2.7.7 Sistemas WBAN no Mercado.....	67
3 PROPOSTA DE SISTEMA INTELIGENTE PARA MONITORAMENTO DOS SINAIS VITAIS DE PACIENTES EM TEMPO REAL	69
3 PROPOSTA DE SISTEMA INTELIGENTE PARA MONITORAMENTO DOS SINAIS VITAIS DE PACIENTES EM TEMPO REAL	69
3.1 AQUISIÇÃO DE DADOS	70
3.2 PRÉ-PROCESSAMENTO DE DADOS	72
3.3 ADAPTAÇÃO DA MATRIZ DE CONFUSÃO.....	74
3.4. DEFINIÇÃO DO MODELO <i>FUZZY</i> PARA RECONHECIMENTO DO ESTADO DO PACIENTE.....	75
3.4.1 Definição das entradas do modelo.....	75
3.4.2 Aquisição do conhecimento dos especialistas	76
3.4.3 Definição dos conjuntos <i>fuzzy</i>	77
3.4.4 Conjunto <i>Fuzzy</i> de Entrada - Frequência Cardíaca	78
3.4.5 Conjunto <i>Fuzzy</i> de Entrada - Pressão Sistólica Não Invasiva	79
3.4.6 Conjunto <i>Fuzzy</i> de Entrada - Pressão Diastólica Não Invasiva.....	80
3.4.7 Conjunto <i>Fuzzy</i> de Entrada - Pressão Média Não Invasiva.....	80
3.4.8 Conjunto <i>Fuzzy</i> de Entrada - Saturação de Oxigênio.....	81
3.4.9 Conjunto <i>Fuzzy</i> de Saída – Condição do Paciente	82
3.4.10 Definição das regras <i>fuzzy</i>	82
3.4.11 Inferência	84
3.4.12 Defuzzificação	84
3.5 ANÁLISE DAS DISTINTAS REDES NEURAIS ARTIFICIAIS	84
3.6 ARQUITETURA REDE RECORRENTE NEURAL ELMAN MISO – E-MISO.....	85
3.6.1 Função de Treinamento e Número de Neurônios na Camada Escondida	86
3.7 PODA DA REDE E-MISO – PE-MISO	90
3.8 ARQUITETURA DA REDE RECORRENTE NEURAL ELMAN MIMO.....	92
3.8.1 Função de Treinamento e Número de Neurônios na Camada Escondida	93
3.9 ARQUITETURA REDE NEURAL NNARX.....	97
3.9.1 Seleção da Estrutura do Modelo Neural	98
3.9.2 Seleção da Ordem do Modelo	99
3.9.3 Camada Escondida e Funções de Ativação	101
3.9.4 Estimação do Modelo	103
3.10 ARQUITETURA REDE NEURAL NNARX PODADA - PNNARX	104
4 DISCUSSÕES	106
4.1 VALIDAÇÃO DO MODELO <i>FUZZY</i>	106
4.2 ESCOLHA DA REDE E A SUA TOPOLOGIA	108
4.3 RESULTADOS DO MODELO ELMAN MISO(5,5,1) – E-MISO.....	109
4.4 RESULTADOS DO MODELO ELMAN MISO(5,5,1) PODADO – PE-MISO.....	112
4.5 RESULTADOS DO MODELO NNARX(6,3,1) - NNARX.....	115
RESULTADOS DO MODELO NNARX(6,3,1) PODADO – P-NNARX.....	118
4.6 ANÁLISE DOS RESULTADOS	122
4.7 RESULTADOS DA IMPLEMENTAÇÃO DO SISTEMA INTELIGENTE PREDITIVO <i>FUZZY</i> -NNARX (FNNARX)	123
5 CONCLUSÕES E PESQUISAS FUTURAS	129
5.1 TRABALHOS FUTUROS	132
REFERÊNCIAS	133

1. INTRODUÇÃO

Muitas doenças não transmissíveis, cuja prevalência é alta e aumenta em faixas etárias maiores, poderiam ser evitadas, ou pelo menos prevenidas, através de um monitoramento constante do paciente. Estudos epidemiológicos brasileiros registram prevalências de hipertensão de 40% a 50% entre adultos com mais de 40 anos de idade (MINISTÉRIO DA SAÚDE, 2010). Mesmo sendo assintomática, a hipertensão arterial é responsável por complicações cardiovasculares, encefálicas, coronárias, renais e vasculares periféricas (MINISTÉRIO DA SAÚDE, 2010). Já na população idosa brasileira a prevalência da hipertensão arterial encontra-se entre 60% e 80% (GIACOMIN et al., 2003; KEARNEY, 2005). No ano de 2010, as doenças do aparelho circulatório, terceira causa de internação hospitalar, representaram 29% das mortes no Brasil (PIEGAS et al., 2013). Muitos destes óbitos não ocorreriam se existissem ações adequadas de prevenção e tratamento precoce.

O monitoramento de sinais vitais de pacientes é uma tarefa essencial em Unidades de Terapia Intensiva (UTI), assim como em outros setores dentro de um hospital. Equipamentos médicos como oxímetros de pulso, monitores não invasivos de pressão arterial ou dispositivos eletrocardiográficos estão ligados atualmente aos monitores de cabeceira, que analisam, interpretam e apresentam a condição atual do paciente, utilizando valores numéricos e alarmes.

Para oferecer maior comodidade e liberdade de movimentos aos pacientes, em muitos países desenvolvidos já é frequente a utilização de Redes de Área Corporal sem Fios (*Wireless Body Area Networks*, abreviada como WBANs). Estas redes são compostas por sensores pequenos, finos e leves distribuídos em torno do corpo humano com o propósito de monitorar as funções vitais do paciente e do meio ambiente em que ele se encontra. Neste caso o monitoramento deve ser não invasivo e confiável em relação a sua exatidão e segurança na transmissão de dados. Para estes pacientes, as tecnologias de telemonitoramento têm significado grandes melhorias na qualidade dos serviços de saúde com uma perceptível redução de custos (E-HEALTH INNOVATION, 2012).

A longevidade da população, a necessidade de desospitalização, o incremento das patologias crônicas, os crescentes custos da internação hospitalar e os constantes problemas de infecção hospitalar, entre outros fatores, fizeram o mercado de *home care* crescer (E-

HEALTH INNOVATION, 2012). Além disso, os altos custos dos cuidados de saúde, especialmente para o tratamento de doenças crônicas, deram especial relevância à medicina preventiva, em que a principal preocupação é o reconhecimento do risco de doença e de intervenção aos primeiros sinais.

A utilização de pacotes de *software* médicos “inteligentes” associados aos sistemas WBAN permitem a análise e a interpretação dos sinais vitais dos pacientes de forma contínua e sem a necessidade do auxílio de um médico.

O uso de técnicas de sistemas inteligentes no telemonitoramento de pacientes e para o auxílio no diagnóstico médico tornou-se uma importante área de pesquisa na busca da transformação de dados em informações clínicas valiosas. Tais sistemas não somente permitem conhecer o estado do paciente em tempo real sem ter que ser supervisionado de forma constante por um especialista, mas também têm o potencial de detectar precocemente condições anormais no paciente.

1.1. MOTIVAÇÃO

A necessidade de monitorar pacientes fora de leitos hospitalares deu origem ao projeto aqui apresentado. Inicialmente, com o potencial de monitorar cinco sinais vitais (i.e., frequência cardíaca, pressão sanguínea sistólica, diastólica e média e saturação de oxigênio), um conjunto de programas baseados em técnicas inteligentes avalia o estado do paciente, ativando alarmes caso o mesmo esteja instável. Como os valores dos sinais vitais podem variar de acordo com fatores como a idade, saúde do paciente ou a mobilidade, este trabalho tem em vista o caso de pacientes com doenças crônicas, mas que não precisam de hospitalização, e sim de um monitoramento periódico.

1.2. DESCRIÇÃO DO SISTEMA

O *software* foi desenvolvido para pacientes adultos entre 50 e 65 anos de idade e o telemonitoramento foi realizado durante o dia, com o paciente acordado e em atividade normal (sem esforço).

Todos os parâmetros medidos e as conclusões obtidas pelo sistema inteligente devem ser enviados ao servidor, onde especialistas poderão fazer um acompanhamento contínuo do paciente e atuar quando algum dos parâmetros apresentar valores fora do normal. Toda esta informação pode ser armazenada num banco de dados para seu posterior uso como base de dados biomédicos para pesquisas, ou para desenvolvimento de novas ferramentas de mineração de dados, também conhecida como *data mining*. A figura 1 apresenta o desenho da

rede proposta, onde é possível identificar três sub-redes. O primeiro nível está formado pelas WBANs (redes corporais), uma para cada paciente. Já o segundo nível contém o conjunto de WBANs e possíveis sensores ambientais. Esta rede é chamada de Rede de Sensores Médicos sem Fios (*Wireless Medical Sensor Network*, abreviada como WMSN). E a terceira e última rede é a Rede de Área Local (*Wireless Local Area Network*, ou simplesmente WLAN) que conecta a WMSN à Internet.

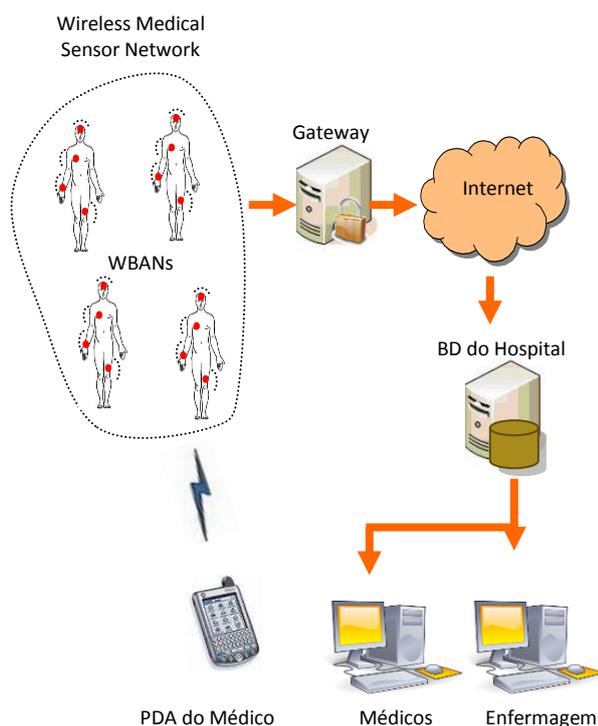


Figura 1 – Arquitetura de Rede de Sensores Médicos sem Fios

1.3. OBJETIVOS

O objetivo geral deste trabalho é desenvolver um sistema inteligente, seguro na transmissão dos dados, confiável na sua exatidão e não invasivo, que através da análise de cinco sinais vitais, permita a detecção e previsão do estado do paciente não hospitalizado, em tempo real, oferecendo mobilidade e conforto ao paciente, em conformidade com a inferência do estado do paciente dada por especialistas clínicos.

Para alcançar o objetivo acima explicitado, foram definidos os seguintes objetivos específicos.

1. Modelar o raciocínio dos especialistas.

O processo de raciocínio médico foi analisado com a ajuda de especialistas de Unidades de Terapia Intensiva (UTI). A ideia principal foi compreender como os médicos inferiam o estado do paciente em função de determinados parâmetros fisiológicos.

2. Modelar o sistema especialista baseado em lógica *fuzzy*.

A lógica *fuzzy* foi utilizada para a construção de um algoritmo que, com base em cinco parâmetros fisiológicos não invasivos, permitisse determinar o estado do paciente no momento atual. Ela foi escolhida pela facilidade na elaboração de regras *fuzzy* partindo do conhecimento dos especialistas, além de sua particularidade de lógica multivalente muito aproveitada em sistemas dinâmicos, não determinísticos e não lineares, como são os sinais fisiológicos humanos. Para desenvolver o algoritmo *fuzzy* foi utilizado Matlab 6.1.0.450 (Release 12.1, The Mathworks, Inc., 3 Apple Hill Drive, Natick, MA 01760-2098, United).

3. Avaliar sistemas especialistas baseados em redes neurais.

O módulo preditivo permitiria indicar ao pessoal médico quais seriam os próximos estados clínicos do paciente em função do estado atual e alguns estados e saídas anteriores. A fim de achar o modelo neural mais apropriado, foram estudadas e comparadas redes neurais recorrentes e séries temporais não lineares para a construção do módulo preditivo. Os *targets* utilizados para a construção da rede neural foram gerados com o módulo *fuzzy*, simulando o raciocínio médico. Para o estudo, comparação e desenvolvimento do módulo preditivo foi utilizado Matlab 6.1.

4. Propor um sistema inteligente utilizando lógica *fuzzy* e redes neurais para monitoramento e predição de estados clínicos de pacientes.

O módulo *fuzzy*, que permite determinar a condição do paciente no momento atual ou tempo (t), mais o módulo preditivo, que prediz o estado do paciente no tempo ($t + d$), onde $d = 1, 2, \dots$ segundos, foram programados e combinados conformando assim um sistema capaz de indicar o estado clínico atual do paciente e de prever o futuro próximo. É importante destacar que neste trabalho o tempo d corresponde a um segundo, simplesmente para conseguir trabalhar com uma amostragem suficientemente grande, mas para implementações futuras esse tempo poderia ser estendido a mais segundos e inclusive minutos. Para o desenvolvimento do sistema inteligente foi utilizado o Matlab 6.1.

1.4. CONTRIBUIÇÕES

Existem muitos trabalhos dedicados a estudar o estado atual do paciente em função de diferentes parâmetros, porém não se tem trabalhado na predição do estado de um paciente em função de sinais vitais não invasivos, utilizando séries temporais. Os trabalhos realizados na área de predição, utilizando redes neurais recorrentes e séries temporais, têm como finalidade a predição do clima, de finanças e de processos industriais. No entanto, há poucos sistemas de predição com parâmetros fisiológicos validados clinicamente.

Neste trabalho foram analisadas as vantagens da lógica *fuzzy* para tratar com conceitos difusos ou multivalentes. As vantagens e desvantagens das redes recorrentes Elman e de séries temporais foram cuidadosamente avaliadas com o objetivo de identificar o melhor modelo para a predição de estados futuros do paciente. Com o fim de minimizar o tamanho da rede neural sem comprometer o desempenho dela, foram aplicadas técnicas para a seleção da ordem do modelo e para a poda da rede.

Parte dos resultados obtidos durante a pesquisa para este trabalho foram publicados em Schatz et al. (2011 e 2012).

Este trabalho apresenta as seguintes contribuições:

- Formalização e apresentação de um novo sistema inteligente para a classificação e predição do estado do paciente, em função de cinco sinais vitais.
- Análise, formalização e simulação do raciocínio médico, utilizando cinco sinais vitais para obter o estado clínico do paciente, expresso por meio de regras SE *antecedente* ENTÃO *consequente*.
- Análise e comparação de resultados de cinco modelos de redes diferentes (MISO Elman, MISO Elman com poda, MIMO Elman, NNARX e NNARX com poda).
- A utilização de séries temporais para a predição da condição de pacientes, baseadas em parâmetros fisiológicos, buscando a otimização de resultados de forma automatizada.
- Formalização do método de avaliação de matrizes de confusão ternárias (com 3 estados).

1.5. ESTRUTURA DO TRABALHO

Em geral, a tese está organizada da seguinte maneira:

O capítulo 2 apresenta a fundamentação teórica sobre a inteligência artificial na medicina, tratando de redes neurais, redes neurais recorrentes, séries temporais e lógica *fuzzy*. Também

dá uma breve descrição dos sensores médicos que poderiam ser utilizados para a obtenção dos sinais vitais, características técnicas e disponibilidade de redes WBAN no mercado.

O capítulo 3 desenvolve o modelo *fuzzy* para analisar o estado atual dos pacientes e fornecer os valores alvos (*targets*) a serem utilizados nas redes neurais. Também compara e analisa cinco modelos de redes neurais, demonstrando o seu desempenho através de diferentes indicadores.

O capítulo 4 traz quadros comparativos e análises das matrizes de confusão das cinco redes estudadas, identificando finalmente aquela com melhor desempenho. Também são apresentados os resultados do sistema inteligente testados com os dados de 30 pacientes novos, demonstrando o real comportamento do mesmo.

No capítulo 5 são apresentadas as conclusões e as pesquisas futuras. Como conclusão a autora apresenta a melhor configuração para o desenvolvimento do sistema inteligente de predição do estado do paciente.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados conceitos importantes sobre sinais vitais e monitoramento hemodinâmico, inteligência artificial na medicina, lógica *fuzzy*, redes neurais artificiais e suas técnicas de avaliação, redes neurais recorrentes, séries temporais, sensores, redes WBAN, redes *wireless*, segurança e confidencialidade dos dados.

2.1. SINAIS VITAIS E MONITORAMENTO HEMODINÂMICO

Os sinais vitais são indicadores básicos da condição física geral da pessoa, e indicam que a pessoa está viva (BLACK et al., 2009). Os sinais a medir são usualmente temperatura, frequência respiratória, batimentos cardíacos (pulso) e pressão sanguínea. Todos estes valores têm uma faixa normal, e é o pessoal médico o responsável por interpretar estes dados e identificar anomalias. As faixas de valores destes parâmetros variam em função da idade e dos efeitos dos medicamentos (ENCICLOPEDIA OF SURGERY, 2012).

A UTI dos hospitais é o ambiente onde os pacientes instáveis são monitorados continuamente e podem até precisar de suporte artificial avançado de vida. Embora atualmente os monitores médicos possam avaliar um número impressionante de variáveis fisiológicas (ver tabela 1), a maioria das UTIs monitora e exibe somente a pressão sanguínea, frequência cardíaca e saturação de oxigênio através de oximetria de pulso (SpO₂) (PINSKY, 2007). Na sequência, estas três variáveis fisiológicas serão analisadas.

2.1.1. Pressão Sanguínea

É a pressão exercida pelo volume de circulação do próprio sangue nas paredes das artérias e veias, depois de bombeado pelo músculo cardíaco. A pressão arterial é regulada pelos mecanismos de homeostase¹ do organismo pelo volume de sangue, o lúmen das artérias e arteríolas, e a força de contração cardíaca (MOSBY'S MEDICAL DICTIONARY, 2009).

¹ Processo pelo qual um organismo ou sistema mantém os seus próprios parâmetros constantes, independentemente das condições do ambiente externo por meio de mecanismos fisiológicos (THE AMERICAN HERITAGE SCIENCE DICTIONARY, 2005).

Tabela 1 – Monitoramento hemodinâmico – Principais variáveis hemodinâmicas

Monitoramento não invasivo

ECG

FC, disritmias, variabilidade FC

Oximetria de Pulso

SpO₂, FC

Pressão arterial

Esfigmomanometria

PS sistólica e diastólica, FC, pulso paradoxal

Pressão venosa central

Distensão venosa jugular, refluxo hepatojugular, onda cãnon (dissociação A-V), regurgitação tricúspide

Monitoramento invasivo

Cateterismo arterial

PS sistólica e diastólica, PAM, FC e pressão sanguínea

Análise de gases sanguíneos arteriais

pH, PaO₂, SaO₂, PCO₂, hemoglobina

Análise das ondas da pressão arterial

Volume de ejeção, saída cardíaca, PPV e SVV

Cateterismo venoso central

Pressão venosa central, ondas de pressão venosa

(ondas "v"), variações de respiração

Análise de gases sanguíneos centrais

pH, PaO₂, SaO₂, PCO₂, hemoglobina

Índices de termo diluição

Volume de ejeção, saída cardíaca, volume de sangue intratorácico, volume final diastólico global e DO₂

Cateterismo Pulmonar

PS sistólica e diastólica, PAM, ondas de pressão

(ondas "v") e Ppao

Análise de gases sanguíneos venosos mistos

pH, PvO₂, SvO₂, PvCO₂, hemoglobina

Saída cardíaca de termo diluição

Volume de ejeção, saída cardíaca, fração de ejeção na ventricular direita e volume final diastólico

Monitoramento ecocardiográfico de doppler esofágico

Volume de ejeção, saída cardíaca e SVV

FC = frequência cardíaca; SpO₂ = saturação oxigênio; PS = pressão sanguínea; PAM = pressão arterial média; PaO₂= pressão parcial de oxigênio em sangue arterial; SaO₂ = saturação oxigênio arterial; PCO₂= pressão parcial de dióxido de carbono; DO₂ = fornecimento de oxigênio; Ppao = pressão de oclusão arterial pulmonar; PvO₂ = pressão parcial de oxigênio venoso misto; SvO₂ = saturação oxigênio venoso; PvCO₂ = pressão de dióxido de carbono misto; SVV = variação do volume de ejeção; PPV = variação do volume de pulso

Fonte: PINSKY, 2007

Em condições normais, o coração tem um ritmo entre 60 a 80 batimentos por minuto. Em cada um desses momentos, o coração bombeia uma certa quantidade de sangue diretamente para a artéria aorta, sendo depois encaminhado para as inúmeras ramificações que se espalham por todas partes do organismo.

A pressão sanguínea sobe e desce em função das exigências diárias do coração e dos músculos, tais como exercício, sono e stress. Ela é registrada com dois números, a pressão arterial sistólica e a pressão arterial diastólica.

A pressão arterial sistólica (PAS), também denominado de pressão arterial máxima, corresponde ao valor medido no momento em que o ventrículo esquerdo bombeia uma certa quantidade de sangue para a aorta.

A pressão arterial diastólica (PAD) , também conhecida como a pressão arterial mínima, corresponde ao momento em que o ventrículo esquerdo volta a encher-se para retomar todo o processo da circulação.

A figura 2 mostra a pressão na aorta, com valores variando entre 80 mmHg e 120 mmHg, e no ventrículo esquerdo, variando entre 0 mmHg e 120 mmHg, ao longo de dois ciclos cardíacos, assim como a pressão arterial sistólica e diastólica.

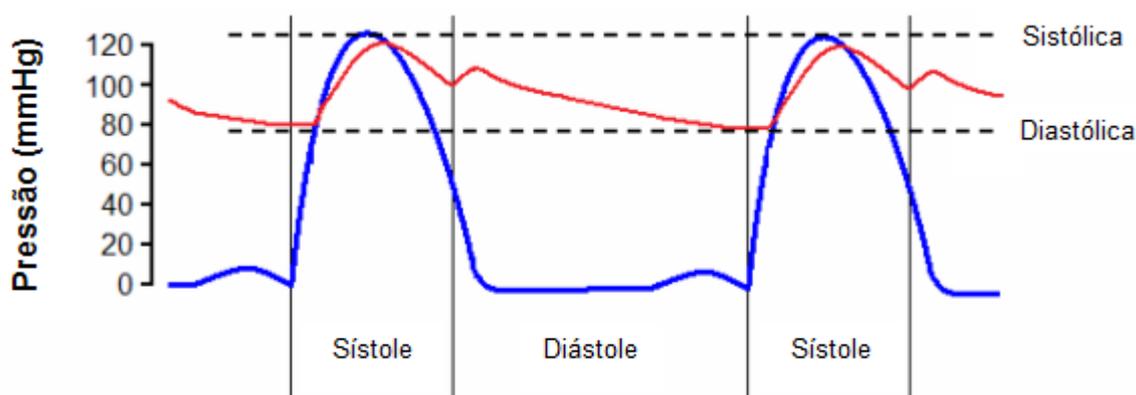


Figura 2 – Representação das pressões sistólicas e diastólicas ao longo de dois ciclos cardíacos.
 Fonte: Adaptado de Wikimedia Commons (Arquivo: Cardiac Cycle Left Ventricle.PNG)

A pressão arterial média (PAM) é o valor médio da pressão no interior da artéria durante um ciclo completo de um batimento cardíaco. Este valor determina a intensidade média com que o sangue vai fluir pelos vasos sistêmicos, ou seja, a pressão de perfusão dos órgãos corporais.

A PAM não é a média das PAS e PAD, porque para frequências cardíacas normais investe-se uma fração maior do ciclo cardíaco na diástole do que na sístole. E assim que a PAM está determinada em torno de 67% pela PAD e em torno de 33% pela PAS, ou seja, a PAM está mais próxima da PAD do que da PAS² (GUYTON; HALL, 2011).

² No entanto, para frequências cardíacas muito elevadas, a diástole compreende uma fração menor do ciclo cardíaco e a PAM aproxima-se mais da média das PAS e PAD (GUYTON; HALL, 2011).

Em condições de normotermia³, o fluxo de sangue para os órgãos principais tem uma PAM de 55 - 60 mmHg (POLO et al., 2000). Se a PAM desce deste valor por um tempo considerável, o órgão alvo não receberá suficiente irrigação sanguínea e pode se tornar isquêmico⁴. Em repouso, ela pode ser calculada com a seguinte equação:

$$PAM \cong PAD + \frac{1}{3}(PAS - PAD) \quad (1)$$

A figura 3 representa graficamente o resultado da PAM, utilizando os valores da PAS e da PAD. Na aorta e grandes artérias de um jovem adulto saudável, a pressão arterial é de aproximadamente 120 mmHg durante a sístole e 80 mmHg durante a diástole. Algumas pessoas saudáveis têm a pressão arterial bem abaixo da média para sua idade, apesar de terem um coração e vasos sanguíneos completamente saudáveis. Este é o caso, por exemplo, dos atletas.

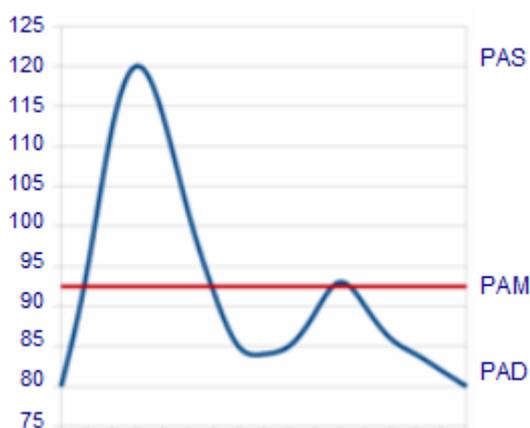


Figura 3 – Curva da pressão arterial
A pressão arterial média PAM (linha reta) difere da média aritmética da pressão sistólica (PAS) e da pressão diastólica (PAD).
Fonte: Autoria própria

O valor da pressão arterial sistólica representa um importante fator de risco para doenças cardiovasculares em pessoas com mais de 50 anos de idade. Na maioria das pessoas, a pressão arterial sistólica aumenta regularmente com a idade, devido ao aumento da rigidez das artérias, à progressiva acumulação de placa nelas, e ao aumento da incidência de doenças cardíacas e vasculares (AMERICAN HEART ASSOCIATION, 2013).

³ Temperatura central corporal medida com valores entre 35 °C e 37,5 °C.

⁴ A isquemia é a falta de suprimento sanguíneo para um tecido orgânico, resultando em falta de glicose e de oxigênio nas células, o que se denomina hipoxia (NLM, 2013).

Mas a pressão arterial diastólica também é um marcador útil. Uma pressão arterial diastólica baixa pode ser um indicador precoce do aumento do débito/gasto cardíaco sistólico e/ou rigidez arterial em adultos jovens e um marcador tardio da rigidez ventricular e arterial em idosos. Por outro lado, uma pressão arterial diastólica elevada em qualquer idade é um marcador de aumento da resistência vascular periférica. No caso de estar gravemente elevada sugere uma crise de hipertensão, especialmente em idosos (FRANKLIN, 2013).

2.1.2. Frequência Cardíaca

Representa a taxa das contrações do coração registradas em batimentos por minuto (bpm). A taxa de contração cardíaca descrita como normal é de 60-100 bpm; a rápida é de 100 bpm ou mais, e a lenta é de 60 bpm ou menos. Esta quantidade de batimentos pode variar dependendo da idade e condição física da pessoa (AMERICAN HEART ASSOCIATION, 2013).

Quando a frequência cardíaca em adultos é de 60 bpm, ou menos, é denominada bradicardia. Esta pode causar baixa pressão arterial, devido à diminuição do esforço do músculo cardíaco, levando menos oxigênio às células. No entanto, adultos fisicamente ativos têm muitas vezes uma frequência cardíaca de repouso inferior a 60 bpm sem causar problemas. Durante o sono, o batimento cardíaco também pode cair abaixo de 60 bpm. Além disso, as pessoas idosas são mais propensas a ter problemas de bradicardia (AMERICAN HEART ASSOCIATION, 2013).

A frequência cardíaca rápida, especialmente acima dos 100 bpm, é denominada taquicardia. Esta pode acontecer em resposta a febre, exercício intenso, excitação nervosa, insuficiência renal, por infecção de algum órgão, por deficiente funcionamento da glândula tireoide, tabagismo, álcool ou drogas. Por períodos curtos pode ser causada por estresse, mas se aparece todos os dias ou várias vezes ao dia pode ser um sintoma de alguma outra doença. Durante a taquicardia aumenta a quantidade de oxigênio fornecido para as células do corpo, aumentando a taxa na qual o sangue circula através dos vasos (AMERICAN HEART ASSOCIATION, 2013).

2.1.3. Saturação de Oxigênio

É a quantidade de oxigênio ligado à hemoglobina no sangue, expressa como uma porcentagem da capacidade de ligação máxima no momento da medição (DORLAND'S MEDICAL DICTIONARY FOR HEALTH CONSUMERS, 2007).

A cor do sangue varia dependendo da quantidade de oxigênio que transporta. Quando a molécula de hemoglobina libera oxigênio perde a sua cor rosa, adquirindo um tom mais azulado. Os oxímetros de pulso determinam a saturação de oxigênio no sangue calculando espectrofotometricamente o “grau” de azul do sangue arterial e expressando isso em termos de saturação. A leitura, obtida através da oximetria de pulso, usa um sensor de luz que contém duas fontes de luz (vermelho e infravermelho) que são absorvidas pela hemoglobina e transmitidas através dos tecidos a um fotodetector. A quantidade de luz transmitida através do tecido é então convertida para um valor digital que representa a porcentagem de saturação da hemoglobina com oxigênio (SCHUTZ, 2001).

A saturação de oxigênio obtida por oximetria de pulso (SpO₂) está em valores normais quando ela se acha entre 97% e 99% em indivíduos saudáveis. Mas um valor de saturação de oxigênio de 95% é clinicamente aceitável num paciente com um nível normal de hemoglobina. Se o valor da saturação de oxigênio do paciente está entre 90% e 95%, ele precisa receber tratamento imediato e monitoração da resposta do mesmo. Valores abaixo de 90% indicam que o paciente está com hipoxia grave e precisa ser hospitalizado.

Os valores de saturação de oxigênio obtidos da oximetria de pulso (SpO₂) são uma parte de uma avaliação completa do estado de oxigenação do paciente e não são um substituto de medição da pressão parcial de oxigênio ou de ventilação (SCHUTZ, 2001).

2.2. INTELIGÊNCIA ARTIFICIAL NA MEDICINA

Shapiro (1992) define a inteligência artificial (IA) da seguinte forma:

“É um campo da ciência e da engenharia preocupado com a compreensão computacional do que é comumente chamado de comportamento inteligente, e com a criação de artefatos que exibem tal comportamento.”.

O matemático britânico Alan Turing (1950) foi um dos fundadores da ciência da computação moderna e da IA. Ele considerava que se pode falar de inteligência artificial quando um ser humano não é capaz de distinguir se o interlocutor com o que está falando é uma outra pessoa ou uma máquina.

A medicina moderna enfrenta o desafio de adquirir, analisar e aplicar um grande volume de conhecimento, necessário para resolver complexos problemas clínicos. Como a IA é capaz de analisar dados complexos e grandes quantidades de informação, suas técnicas podem ser aplicadas em quase todos os campos da medicina. Seu potencial para explorar relações significativas dentro de um conjunto de dados pode ser utilizado no diagnóstico, no processo de tomada de decisão terapêutica, em tratamento e em prognóstico (PATEL et. al, 2009).

Atualmente, a Inteligência Artificial na Medicina (AIM sua sigla em inglês) é um componente essencial da informática biomédica e uma das metodologias que pode ajudar a resolver problemas na saúde. Em (PATEL et. al, 2009) Shortliffe diz que a IA está amplamente representada no campo da informática biomédica, em áreas como a representação de conhecimento, desenvolvimento de ontologias, modelagem de domínios de terminologia e semântica, suporte à decisão, raciocínio em condições de incerteza, processamento de imagem e muitos outros.

Os progressos da AIM ficam demonstrados com o surgimento de programas de IA que auxiliam o médico clínico na formulação de um diagnóstico, na tomada de decisões ou na predição de resultados. Eles são desenvolvidos para ajudar os especialistas nas suas funções diárias, assistindo-lhes com tarefas que dependem da manipulação de dados e conhecimento. Estes sistemas incluem redes neurais artificiais (RNA), sistemas especialistas nebulosos ou difusos, computação evolucionária e sistemas inteligentes híbridos, tentando compreender e construir entidades inteligentes.

2.3. LÓGICA *FUZZY*

O termo *fuzzy* em língua inglesa pode ser traduzido para o português como indistinto, incerto, nebuloso ou difuso.

A lógica *fuzzy* é um ramo da Inteligência Artificial que ajuda os computadores a simularem o raciocínio a partir de dados incertos ou incompletos.

Segundo Russell e Norvig (2004, p. 512) a lógica *fuzzy* é um método para raciocínio com expressões lógicas que descrevem a pertinência a conjuntos *fuzzy*. Ela trata de imitar o raciocínio humano que pode ser induzido a partir de afirmações vagas e usando o senso comum.

Na lógica clássica, convencional, as coisas são verdadeiras ou falsas, pertencem a um conjunto ou a outro, mas não a ambos ao mesmo tempo. O número um para a lógica convencional, sempre pertencerá ao conjunto dos números ímpares e não ao dos números pares. Isto acontece em sistemas bivalentes, onde um objeto não pode pertencer ao mesmo tempo ao conjunto e ao complemento do conjunto, ou a nenhum dos dois (KOSKO, ISAKA, 1993).

Os elementos que pertencem aos conjuntos *fuzzy*, ou também denominados multivalentes podem pertencer parcialmente a um ou mais conjuntos *fuzzy*. A diferença entre a lógica clássica e a difusa é muito sutil quanto às fronteiras. Nos conjuntos clássicos os limites entre

conjuntos são precisos, mas para a lógica *fuzzy* estes limites são vagos, imprecisos e criam contradições parciais. A expressão “bastante quente” pode pertencer em 80% ao conjunto “quente” e em 20% ao conjunto “frio”. É por isso que a lógica *fuzzy* trabalha muito bem com conjuntos de conceitos subjetivos ou com limites imprecisos como “frio”, “calor”, “alto”, “baixo”, “demasiado” ou “apenas”, entre outros exemplos.

2.3.1. Representação *Fuzzy* do Conhecimento

Os conjuntos *fuzzy* foram introduzidos por Zadeh (1965, p.338) como uma forma de representar e manipular dados que não são precisos, mas sim difusos. Um conjunto *fuzzy* (ou conjunto difuso) para Zadeh “é uma classe de objeto com um grau contínuo de pertinência entre zero e um”. Um conjunto *fuzzy* A em um universo X é definido por uma função de pertinência $\mu_A(x): X \rightarrow [0,1]$, e representado por um conjunto de pares ordenados

$$A = \{\mu_A(x)/x\} \quad x \in X \quad (1)$$

onde $\mu_A(x)$ indica o quanto x é compatível com o conjunto A . Nestes conjuntos um determinado elemento pode pertencer a mais de um conjunto *fuzzy*, com diferentes graus de pertinência.

Um conjunto *fuzzy* pode ser representado como um triângulo (figura 4) com um pico (centro), uma largura esquerda $(m-a) > 0$, uma largura direita $(b-m) > 0$ e a sua função de pertinência está dada pela seguinte equação:

$$\mu_A(x) = \begin{cases} 0 & \text{se } x \leq a \\ \frac{x-a}{m-a} & \text{se } a < x \leq m \\ \frac{b-x}{b-m} & \text{se } m < x \leq b \\ 0 & \text{se } x > b \end{cases} \quad (2)$$

Também pode ser representado como uma função trapezoidal (figura 5) com a seguinte função de pertinência.

$$\mu_A(x) = \begin{cases} 0 & \text{se } x \leq a \\ \frac{x-a}{m-a} & \text{se } a < x \leq m \\ 1 & \text{se } m < x \leq n \\ \frac{b-x}{b-n} & \text{se } n < x \leq b \\ 0 & \text{se } x > b \end{cases} \quad (3)$$

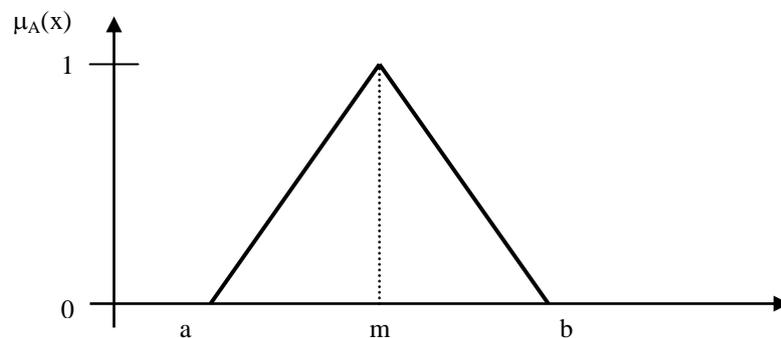


Figura 4 – Função triangular de pertinência

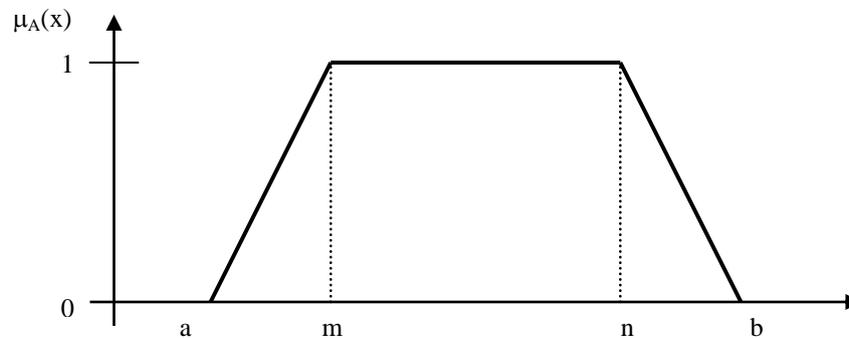


Figura 5 – Função trapezoidal de pertinência

As funções de pertinência normalmente são representadas a partir de funções comuns como a triangular e trapezoidal, devido a sua simplicidade matemática, mas existem muitas mais funções possíveis como a gaussiana, a sigmoide, a gamma, a pi, a cauchy, entre outras.

A diferença entre a lógica clássica e a lógica *fuzzy* é algo que Aristóteles chamou de “a lei do meio excluído” (BART, SATORU, 1993). Na teoria de conjuntos clássica, um objeto pertence ou não pertence a um conjunto, em outras palavras, não existe meio termo. Na lógica *fuzzy* um objeto pode pertencer 100% a um conjunto ou pode ter certo grau de pertinência a cada um deles.

A figura 6 mostra um caso onde a temperatura corporal é avaliada segundo a lógica clássica e a lógica *fuzzy*. Segundo a lógica clássica, quando uma pessoa tem 38° C de temperatura corporal, considera-se que tem temperatura alta, mas para o modelo apresentado na figura 6 a lógica *fuzzy* considera que 38° C corresponde a 20% de temperatura normal e 80% de febre. Isto é análogo à expressão “febre leve”.

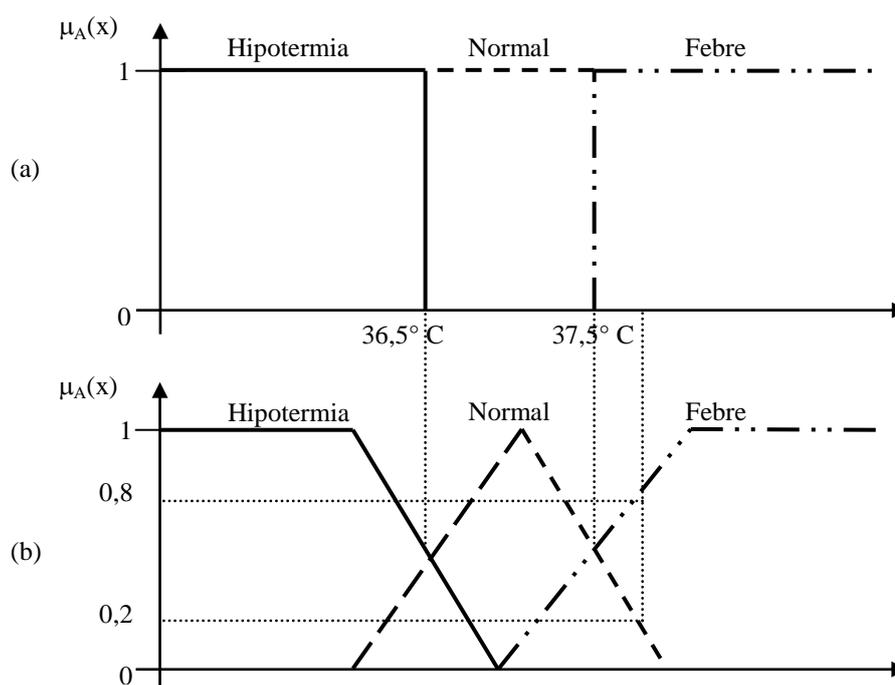


Figura 6 – Lógica convencional (a) vs. lógica *fuzzy* (b)

Enquanto a lógica convencional atribui a cada valor uma única categoria, a lógica difusa permite uma gradual transição onde um valor pode pertencer a duas categorias com diferentes graus de pertinência.

Fonte: Adaptado de Oberli et al. (1999, p.30)

2.3.2. Variáveis Linguísticas

Uma variável linguística é uma variável cujos valores são os nomes dos conjuntos *fuzzy* e, é utilizada para representar de modo impreciso um conceito ou uma variável de um dado problema (REZENDE, 2005). Ela admite valores unicamente linguísticos como “frio”, “muito alto”, “levemente agitado”, e não valores numéricos.

Os valores de uma variável linguística podem ser sentenças construídas a partir de termos primários (alto, médio e baixo, por exemplo), conectivos lógicos (e, ou, não), modificadores (muito, pouco, levemente, mais ou menos, por exemplo) e delimitadores (como parênteses).

Para Zadeh, uma variável linguística é caracterizada por uma quintupla $\langle N, T(N), X, G, M \rangle$, onde:

N: nome da variável;

T(N): conjunto de termos de linguísticos de N, ou seja, o conjunto de nomes que a variável N pode assumir, também chamados termos primários;

X: universo de discurso (domínio);

G: gramática para a geração dos termos de N, conectivos lógicos, modificadores e delimitadores;

M: regra semântica, para associar a cada valor gerado por G um conjunto *fuzzy* no universo X, representando o seu significado.

No caso da variável temperatura corporal da figura 6, ter-se-ia:

N: temperatura corporal;

T(N): {hipotermia, normal, febre};

X: [28, 45];

G: temperatura “não muito baixa” ou “levemente alta”, por exemplo.

M: associa aos valores de G (“temperatura não muito baixa” e “temperatura levemente alta”) um conjunto *fuzzy*.

2.3.3. Regras *Fuzzy*

Enquanto as variáveis em matemática costumam ter resultados numéricos, na lógica *fuzzy* as variáveis trabalham com resultados linguísticos não numéricos para gerar as regras *fuzzy*. Estas regras formam parte do conhecimento do sistema especialista e descrevem como tomar uma decisão (escolher uma saída) classificando as entradas. O conhecimento está representado por regras ‘se-então’ com m entradas e n saídas, conforme descrito abaixo.

Se (entrada₁ pertence à função_de_pertinência₁)

e/ou (...)

e/ou (entrada_m pertence à função_de_pertinência_m)

então (saída₁ pertence à função_de_pertinência₁)

então (...)

então (saída_n pertence à função_de_pertinência_n)

Consideremos o seguinte caso:

Tentando determinar se uma pessoa está acima do peso ou não, variáveis de entrada com peso (baixo, normal e alto) e altura (baixa, normal e alta) podem ser analisadas para obter uma saída (abaixo do peso, normal ou sobrepeso).

Se (peso é baixo)

e (altura é alta)

então (pessoa está abaixo do peso)

O processo de fuzzificação determinará o que significa “baixo” na variável de entrada peso e “alta” na variável de entrada altura.

Para a construção das regras é importante levar em conta o seguinte :

- o número de regras cresce exponencialmente com o número de variáveis de entrada;
- quanto mais regras a construir, mais difícil é saber se elas são adequadas;
- se o número de variáveis de entrada for muito grande, será difícil compreender a relação causal entre as entradas e as saídas e portanto será difícil a construção das regras.

2.3.4. Sistema de Inferência *Fuzzy*

O sistema de inferência *fuzzy* é mostrado na figura 7, onde estão identificadas as funções de cada bloco.

Neste sistema de inferência *fuzzy*, os valores precisos de entrada consideram-se não *fuzzy*, e são resultantes de medições ou observações.

O primeiro passo é efetuar um mapeamento destes dados para os conjuntos *fuzzy* relevantes, o que é realizado no estágio de fuzzificação. Este processo consiste em converter uma variável real, também chamado crisp, em um grau de pertinência, que quantifica o grau de pertinência para sua correspondente variável linguística. Neste estágio ocorre também a ativação das regras relevantes para uma dada situação. As regras são fornecidas pelos especialistas em forma de sentenças linguísticas.

O resultado da inferência é um conjunto *fuzzy*, ou seja, um conjunto de variáveis linguísticas com seus respectivos graus de pertinência. Mas, para que esta informação possa ser utilizada, é preciso obter um valor concreto, *crisp* e representativo daquele conjunto.

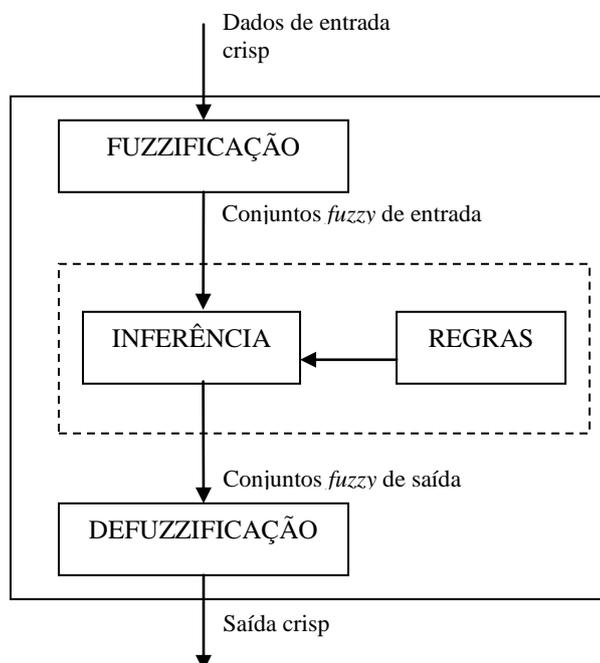


Figura 7 – Sistema de inferência *fuzzy*
Fonte: Autoria própria

A defuzzificação é o processo inverso à fuzzificação. Uma vez obtido o conjunto *fuzzy* de saída, com a ajuda das regras que foram fornecidas pelos especialistas, é efetuada uma interpretação dessa informação, para obter um valor representativo como saída.

O processo de defuzzificação é muito importante, pois determina o sucesso da saída. A defuzzificação é o processo em que as funções de pertinência são testadas para achar o grau de pertinência; logo os graus de pertinência são usados nas equações *fuzzy* e uma região de saída (curva) é definida. E a partir daí a saída é deduzida (KARTALOPOULOS, 1996, p.129).

Muitas técnicas têm sido desenvolvidas para produzir uma saída concreta e única partindo dessa curva de saída. As três mais comuns são:

- **centroide**, é o método de defuzzificação mais popular e consiste em achar o centro da área sobre a curva.
- **maximizador**, pela qual a saída máxima é a selecionada,
- **média ponderada**, a média do valor máximo do conjunto de saída.

2.3.5. Metodologias Utilizadas para o Projeto do Modelo *Fuzzy*

O projeto de um modelo *fuzzy* não é uma tarefa fácil, especialmente em problemas complexos do mundo real. Normalmente os sistemas *fuzzy* podem ser gerados utilizando duas metodologias. A primeira é com o auxílio de especialistas na área, os quais ajudam na formulação das regras e das distintas funções de pertinência para cada entrada. Isto permite a introdução no sistema de subjetividades do especialista. A segunda metodologia utilizada para modelar sistemas *fuzzy* consiste em repetitivos ajustes do sistema para achar os componentes *fuzzy*, ou seja, as funções de pertinência das entradas e das saídas, e a formulação das regras, a partir de amostras representativas e com a ajuda de algoritmos genéticos. Este é o caso dos sistemas *fuzzy* adaptativos (WONG; HAMOUDA, 2000, p.251).

As pesquisas mais recentes em sistemas *fuzzy* apontam para a otimização do número de termos *fuzzy* em problemas complexos (REZENDE, 2005). Se tivermos em conta que um simples modelo *fuzzy* de duas entradas pode ter mais de dez mil combinações possíveis de regras *fuzzy* (WONG; HAMOUDA, 2000, p.252), fica claro que a aplicação do bom senso e o conhecimento dos especialistas faria restringir o escopo, mas as regras *fuzzy* selecionadas não seriam necessariamente as mais adequadas.

Mas, segundo os autores acima mencionados, com a geração automática de regras perde-se uma das características mais importantes da lógica *fuzzy* que é a capacidade de descrever o sistema em termos linguísticos. A utilização de termos linguísticos permite projetar o sistema com um raciocínio mais humanizado, especialmente com o uso das regras *fuzzy* “se... então”.

2.3.6. Aplicações *Fuzzy* na Área Biomédica

Na área biomédica há muitas aplicações desenvolvidas utilizando ferramentas *fuzzy*. Na subárea de estudo de sinais vitais, por exemplo, Liu e colaboradores (2008, p.81) apresentaram um algoritmo baseado em lógica *fuzzy* que permite qualificar a confiabilidade dos valores de frequência cardíaca e frequência respiratória em tempo real, ou seja, enquanto eles são medidos como uma cadeia contínua de dados biomédicos. Este algoritmo complementa um monitor de sinais vitais desenvolvido pelo Instituto de Pesquisa de Medicina Ambiental do Exército dos Estados Unidos e permite monitorar os sinais do soldado enquanto ele está no campo de batalha.

Oberli e colaboradores (1999, p.29) apresentaram também um sistema especialista baseado em lógica *fuzzy* que avalia os alarmes emitidos pelo monitor de sinais vitais analisando sete sinais vitais de pacientes com cirurgia cardíaca.

Uribe e colaboradores (2011) também propõem um sistema de suporte à decisão baseado em métodos *fuzzy* para o monitoramento de pacientes com risco cardiovascular. Nesse trabalho eles utilizam seis variáveis fisiológicas não invasivas para determinar em três níveis se o paciente está em condições normais ou se há necessidade de emitir um alarme. No caso de alarme, o sistema indica uma entre seis possíveis doenças cardiovasculares.

No trabalho de Leite e colegas (2011) os autores apresentam um modelo *fuzzy* capaz de monitorar e classificar a condição dos sinais vitais de um paciente na UTI, enviando alarmes.

As técnicas *fuzzy* também podem ser usadas para gerar regras precisas obtidas de redes neurais treinadas para a criação de programas que ajudem no diagnóstico de câncer de mama, como foi demonstrado por Setiono (1999).

Alguns problemas complexos requerem a combinação de várias técnicas inteligentes a fim de achar a melhor solução. A combinação de lógica *fuzzy* com redes neurais, chamada de neuro *fuzzy*, é muito usada e normalmente auxilia na detecção dos conjuntos e regras *fuzzy* usando redes neurais. Para a subárea de detecção de câncer as ferramentas *fuzzy* e neuro-*fuzzy* são também muito utilizadas. Obi e colegas (2011) trabalharam num sistema inteligente de auto aprendizado a ser aplicado num sistema especialista neuro-*fuzzy* para o diagnóstico da leucemia.

Faisal e colaboradores (2012) desenvolveram um sistema de inferência neuro-*fuzzy* para o diagnóstico do risco em pacientes com dengue com a ajuda de nove parâmetros não invasivos.

Portmann e colegas (2010) criaram o Prometheus, uma estrutura *software/hardware* para suporte ao usuário/paciente. O sistema usa a lógica *fuzzy* e está baseado numa variedade de dados, tais como localização e condições internas e externas, para construir um serviço de apoio ao usuário na sua casa.

2.4. REDES NEURAIAS ARTIFICIAIS (RNA)

Uma das grandes vantagens proporcionadas pelas RNA para a identificação de sistemas é que não é necessário ter conhecimento do processo para conseguir obter as relações funcionais entre as entradas e as saídas do sistema.

Embora exista uma grande quantidade de arquiteturas de RNA e algoritmos de aprendizagem, para aplicações na área de controle tem sido usadas com maior frequência as RNAs arquitetura multicamada para frente (*feedforward*) e o algoritmo de aprendizado chamado de retropropagação do erro (*backpropagation*).

As RNA são capazes de resolver problemas de aproximação, classificação, predição, categorização e otimização e são aplicadas para casos de reconhecimento de caracteres e de

voz, predição de séries temporais, modelagem de processos, controle, entre outros (REZENDE, 2005).

A predição visa tentar antecipar os estados seguintes de um determinado sistema, baseado nos seus estados anteriores. Dada uma sequência de vetores de características correspondentes aos instantes de tempo $(t - n), \dots, (t - 1), (t)$, a rede tenta prever o valor do vetor no instante seguinte, isto é, no instante de tempo $(t + 1)$. Para isso considera que a saída seguinte do sistema é uma função das entradas e saídas anteriores, então o problema pode ser descrito segundo a seguinte equação:

$$y(t+1) = f(y(t), y(t-1), \dots, y(t-m), x(t), x(t-1), \dots, x(t-p)) \quad (4)$$

onde x são as entradas externas à rede, y são as saídas anteriores da rede, m e p são os atrasos utilizados nos regressores das saídas e das entradas. Exemplos típicos são a predição financeira, a predição de tempo e a modelagem de processos industriais.

O objetivo para os problemas de classificação é atribuir um padrão de entrada a uma classe entre um conjunto de classes conhecidas. Em outras palavras, a saída da rede é uma estimativa da probabilidade de pertinência da entrada a uma classe particular. Exemplos típicos de problemas de classificação são o reconhecimento de voz, de caracteres e de imagens.

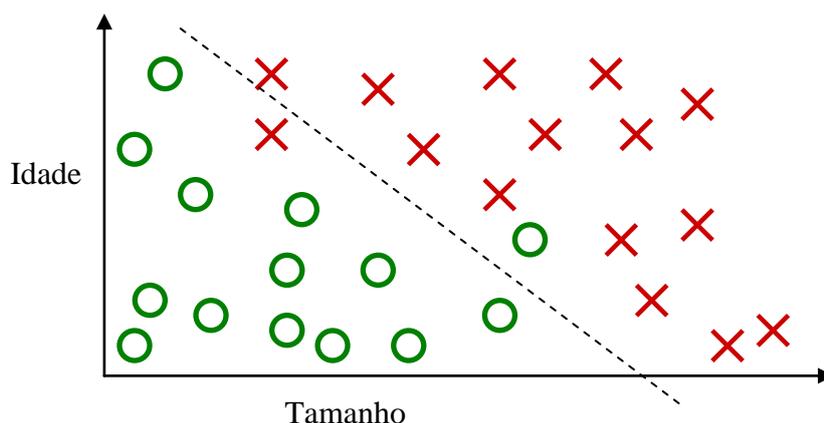


Figura 8 – Exemplo de um problema de classificação.

A figura separa duas classes no espaço bidimensional, classificando a malignidade dos tumores em função da idade do paciente e do tamanho do tumor.

Fonte: Autoria própria

A figura 8 mostra um exemplo de separação linear de duas classes ‘x’ e ‘o’, onde se trata de classificar a malignidade de um tumor em função da idade do paciente e do tamanho do

tumor. A reta de separação representa a solução da rede para o problema de separação de classes. A cada novo par de entradas (*tamanho, idade*) apresentado à rede, ela deve responder se o tumor é benigno ou maligno dependendo se ele estiver à esquerda ou à direita da reta de separação. Para que as superfícies de separação entre as classes possam ser perfeitamente definidas, é necessário apresentar à rede em treinamento um conjunto de exemplos de cada classe suficientemente representativo.

É importante esclarecer que as RNA são capazes de resolver problemas muito mais complexos, como são os problemas não-lineares de reconhecimento de padrões ou predição no tempo.

Existem duas categorias principais de estruturas de redes neurais: redes acíclicas ou redes de alimentação direta, mais conhecidas como redes *feedforward*, e redes cíclicas ou redes recorrentes. Uma rede de alimentação direta representa uma função de sua entrada atual, não tendo nenhum estado interno além dos pesos propriamente ditos. Em contrapartida, uma rede recorrente utiliza suas saídas para alimentar novamente suas próprias entradas, comportamento característico de um sistema dinâmico. Por sua vez, as redes recorrentes podem admitir memória de curto prazo, o que as torna mais semelhantes ao comportamento do cérebro, assim como também mais complicadas, o que não acontece com as redes de alimentação direta (RUSSELL; NORVIG, 2004).

2.4.1. Redes de Alimentação Direta

As redes neurais artificiais são modelos matemáticos para análise de dados inspirados nas estruturas neurais do cérebro. Elas conseguem resolver problemas complexos, não lineares, com aplicações em várias áreas.

Estas redes são inerentemente acíclicas (também chamadas de redes sem memória) devido ao fato de que o sinal é propagado somente da entrada para a saída da rede.

As redes com múltiplas camadas de neurônios escondidos (ver figura 9), também chamadas perceptrons multicamadas (*Multilayer Perceptrons* em inglês ou MLP), são computacionalmente mais poderosas do que as redes sem camadas escondidas ou perceptrons simples. As MLP podem tratar dados que não são linearmente separáveis, e o seu desempenho como classificador (acurácia) depende do número de neurônios nas camadas escondidas. As MLP se distinguem dos percéptrons pela presença de uma ou mais camadas escondidas. As entradas de um neurônio são as saídas dos neurônios da camada anterior, portanto não há ligação entre neurônios de uma mesma camada.

O perceptron e o MLP representados na figura 9 são, respectivamente, exemplos de modelos de rede sem camada escondida e rede de múltiplas camadas escondidas, no caso da figura 9 de uma camada escondida. O MLP é, provavelmente, o protótipo de rede neural mais utilizado. O principal motivo é a sua capacidade para modelar relações funcionais simples e complexas.

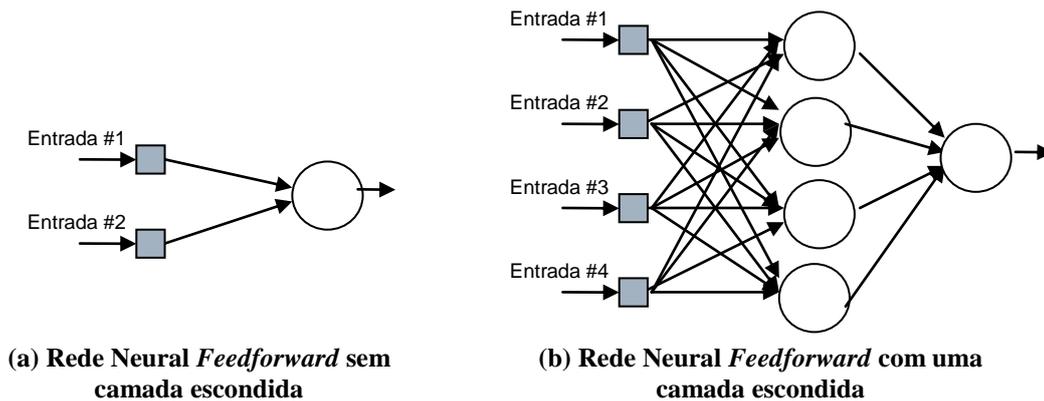


Figura 9 - Exemplos de RNA *feedforward*

A figura 10 mostra uma rede neural com uma camada de unidades de entrada, com E unidades de entrada; uma camada escondida, com H unidades escondidas; e uma camada de unidades de saída, com K unidades de saída, neste caso $K=1$. Estas unidades estão conectadas umas com outras através de pesos $w_{ji}^{(c)}$, onde c representa a camada de origem, i representa a unidade desta camada e j representa a unidade da camada de destino. Cada uma das unidades, exceto as de entrada, tem também uma conexão particular com uma unidade de viés (*bias*, em inglês). As unidades são chamadas de neurônios, pois sua estrutura é semelhante às conexões dos neurônios biológicos.

Os números de neurônios de entrada e saída de uma rede neural são conhecidos previamente, devido à dimensionalidade das entradas e as categorias que se quer obter na saída, porém as quantidades de camadas escondidas e número de neurônios de cada uma delas são pontos a definir muito cuidadosamente.

As redes de alimentação direta MLP envolvem unidades escondidas em uma ou mais camadas escondidas. A vantagem de adicionar camadas escondidas é que isto aumenta o espaço de hipóteses que a rede pode representar. Com uma única camada escondida suficientemente grande, é possível representar qualquer função contínua; com duas camadas,

até mesmo funções descontínuas podem ser representadas (RUSSELL; NORVIG, 2004, p. 720).

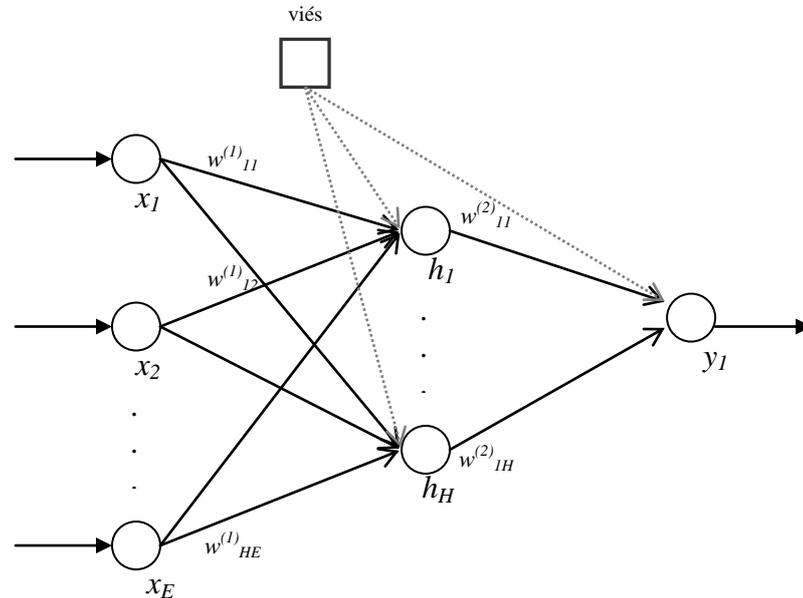


Figura 10 – Topologia de uma rede *feedforward* com viés e indicação dos pesos

2.4.2. Redes Neurais Recorrentes

As topologias de redes neurais artificiais normalmente correspondem a gráficos acíclicos, mas as redes recorrentes são redes neurais artificiais que usam as saídas da rede no tempo (t) como entrada da mesma rede no tempo ($t + 1$) (MITCHELL, 1997, p.119). Desta forma estas redes são denominadas cíclicas. As redes recorrentes normalmente são escolhidas devido ao seu alto desempenho em problemas de predição de séries temporais.

Uma rede recorrente trabalha da seguinte forma: imagine uma rede que tenha que prever o estado do tempo para o dia seguinte $y(t + 1)$, baseada nos dados do tempo no dia de hoje $x(t)$. Se utilizarmos para realizar a tarefa uma rede neural tipo *feedforward* teríamos uma rede com uma topologia semelhante à mostrada na figura 11(a). Uma grande desvantagem desta rede para um problema deste tipo é que o valor a prever $y(t + 1)$, depende só da entrada $x(t)$ e não pode capturar outros valores anteriores de $x(t)$ que também poderiam influir no resultado. Ou seja, uma rede *feedforward* não possui memória para armazenar valores anteriores. Este inconveniente pode ser resolvido utilizando redes recorrentes como a mostrada na figura 11(b). Nela pode-se notar um novo neurônio b na camada escondida e uma nova unidade de entrada $c(t)$. O valor de $c(t)$ pode ser definido como o valor da unidade b no tempo ($t - 1$).

As redes recorrentes têm como característica a realimentação da rede com os estados anteriores, evoluindo para um estado estacionário. A característica destas redes faz que com que elas sejam muito úteis para seu uso em sistemas dinâmicos.

As redes recorrentes podem ser simples ou totalmente conectadas, dependendo das conexões entre os neurônios. Existem exemplos conhecidos de redes neurais parcialmente recorrentes, como a rede de Jordan (1986) ou a rede de Elman (1990) e a rede de Hopfield (1982) que é uma rede totalmente recorrente.

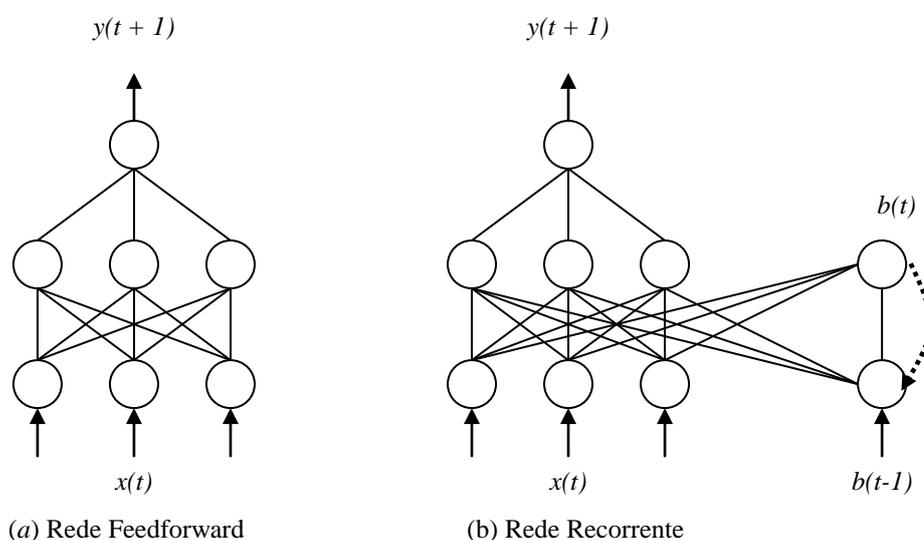


Figura 11 – Topologias das redes *feedforward* e das redes recorrentes
Fonte: Adaptado de (MITCHELL, 1997, p. 120)

2.4.3. Redes Recorrentes Simples

As redes de Elman (ELMAN, 1991), assim nominadas devido ao seu criador, o professor da Universidade da Califórnia Jeffrey Elman, cuja topologia é mostrada na figura 12(a), são conhecidas como redes recorrentes simples e são um aperfeiçoamento das redes *feedforward* devido à inclusão de realimentação entre as camadas imediata e adjacente. Esta realimentação oferece à rede uma memória de eventos imediatos anteriores, afetando as atualizações dos pesos em cada uma das camadas da rede.

A rede de Elman é uma rede recorrente onde a realimentação se dá na saída de cada neurônio da camada escondida para todos os neurônios da mesma camada. Uma camada, chamada camada de contexto, também escondida, composta de unidades de contexto, simula a

memória da rede armazenando os valores das saídas prévias dos neurônios da camada escondida. Os valores das unidades de contexto são a retroalimentação completa para os neurônios da camada escondida e servem como entradas adicionais à rede. Na rede de Elman os valores da camada de saída não retroalimentam a rede.

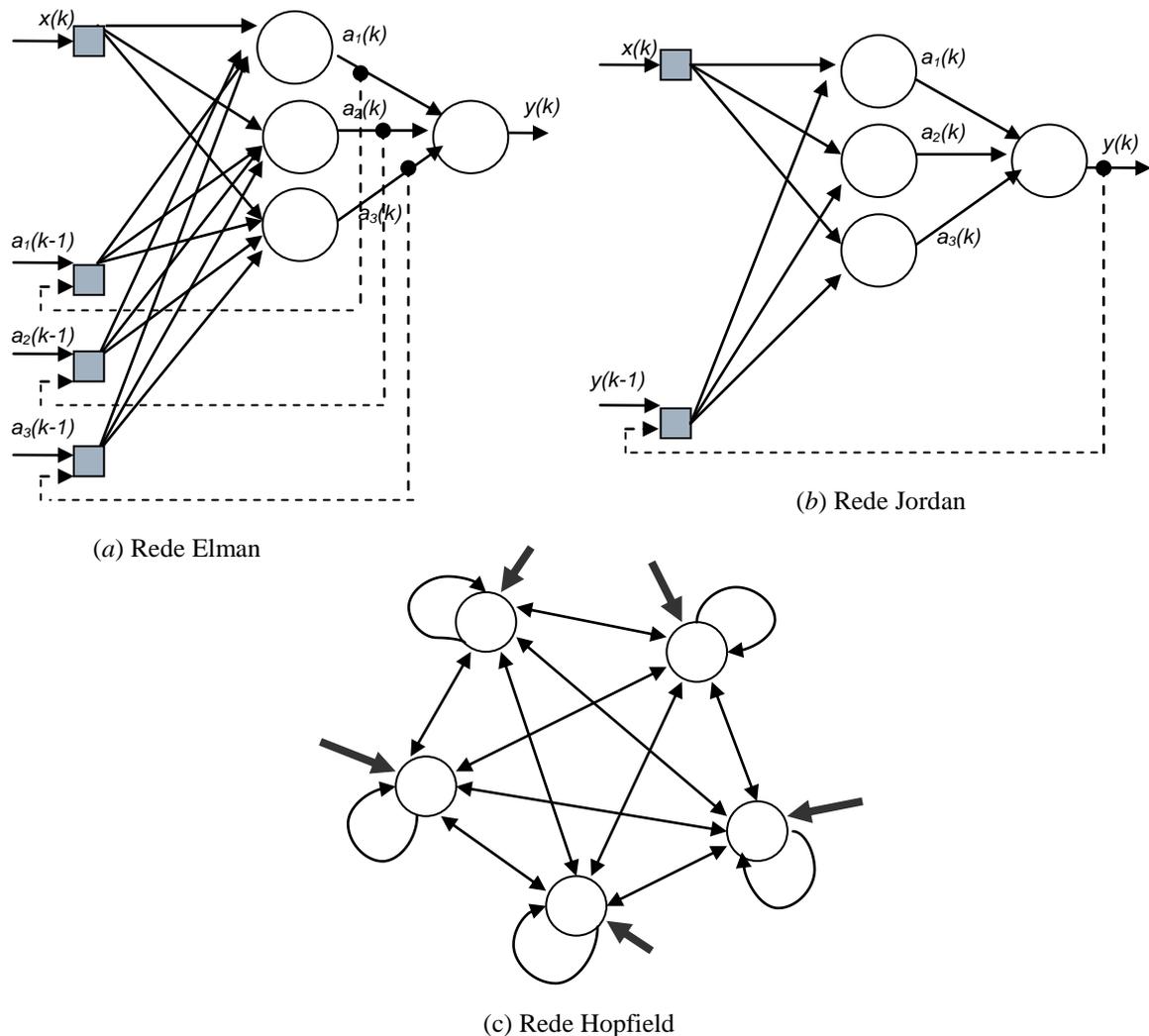


Figura 12 - Redes Recorrentes Simples (Elman/Jordan) e completamente conectada (Hopfield)
Fonte: Autoria própria

O funcionamento da rede Elman é o seguinte: No instante t (inicial) o sinal é propagado pela rede e as unidades de contexto, inicializadas com a saída da camada escondida com o valor zero, não influenciarão na saída da rede, ou seja, na primeira iteração a rede se comportará como uma rede *feedforward*. Ainda na primeira iteração, os neurônios escondidos ativarão os neurônios da camada de contexto e esses armazenarão a saída desta iteração que será utilizada no próximo ciclo. O algoritmo *backpropagation* é então aplicado para a correção dos pesos sinápticos, com exceção das sinapses recorrentes que são fixas em 1. No

instante $(t + 1)$ o processo é repetido. A diferença é que a partir de agora os neurônios escondidos serão ativados pelas unidades de entrada e pelas unidades de contexto que possuem o valor de saída dos neurônios escondidos no instante (t) .

Para que uma rede Elman tenha melhor possibilidades de aprender um problema, ela requer um número relativamente maior de neurônios na camada escondida. Isto acontece porque Elman não é tão bom achando os pesos mais adequados para os neurônios escondidos (MOHD, 2004).

A rede de Jordan, assim denominada devido ao seu criador, o professor da Universidade de Berkeley Michael I. Jordan (JORDAN, 2012), é semelhante à rede de Elman. Mas neste caso, as unidades de contexto são alimentadas a partir da camada de saída em vez da camada escondida. A topologia da rede Jordan pode ser observada na figura 12(b).

A principal vantagem das redes de Elman em relação às redes Jordan é que o número de unidades de contexto não é directamente determinada pela dimensão de saída (como em redes Jordan), mas sim pelo número de neurônios ocultos. Nas redes de Elman é possível aumentar ou diminuir a complexidade do problema aumentando ou diminuindo a quantidade de neurônios ocultos e assim obter uma maior quantidade de neurônios na camada de contexto. Mas nas redes de Jordan, se precisar aumentar o número de neurônios na camada de contexto é necessário aumentar a quantidade de neurônios na saída e isto significa mudar a definição do problema.

2.4.4. Redes Recorrentes Totalmente Conectadas

A rede Hopfield (HOPFIELD, 1982), cujo nome vem do seu criador, o professor da Universidade de Princeton John Hopfield, é uma rede recorrente totalmente conectada. Assim, o sinal de entrada é propagado para frente e para trás produzindo-se uma dinâmica. Em algum ponto, a evolução vai parar num estado estacionário, mas pode ser também que não pare nunca. A desvantagem desta rede é que requer tempos de processamento muito altos, em comparação com as redes recorrentes simples. A sua topologia é mostrada na figura 12(c).

Para redes totalmente conectadas, as únicas escolhas possíveis são o número de camadas escondidas e seus tamanhos. Neste caso se escolhe empiricamente a arquitetura de rede que oferece a mais alta exatidão de predição sobre os conjuntos de validação (RUSSELL; NORVIG, 2004).

2.4.5. Aplicações de Redes Neurais Artificiais Recorrentes na Área Biomédica

Nesta seção alguns trabalhos desenvolvidos com a utilização de redes neurais recorrentes são apresentados.

Para prever valores de glicose no sangue existem vários trabalhos semelhantes apresentados. Sandham e colegas (1998) usaram uma rede Elman com 95 neurônios na camada escondida para prever futuros valores de glicose no sangue, usando como parâmetros de entrada a insulina, a dieta, exercícios, valores de glicose no sangue anteriores e outros fatores. Mougiakakou e colaboradores (2006) apresentaram uma rede recorrente em tempo real para prever valores de glicose no sangue em crianças com diabetes tipo 1.

As redes neurais recorrentes também são utilizadas no estudo dos sinais de EEG (eletroencefalograma), e assim podem prever futuras crises de epilepsia. Gayatri e colegas (2010) propõem uma rede Elman usando uma única entrada, o valor de ApEn (parâmetro estatístico que mede a previsibilidade dos valores de amplitude atuais com base nos seus valores de amplitude anteriores), para a implementação de um sistema inteligente para a detecção da epilepsia.

No trabalho de Lalitha (2007) fica demonstrada que a rede neural Elman tem um desempenho melhor que a rede MLP para a detecção automática de níveis de anestesia. Logo após estudar três parâmetros caóticos não lineares, como a dimensão de correlação, e os expoentes Lyapunov e Hurst, os autores chegaram à conclusão que a rede Elman com o expoente Lyapunov apresenta uma acurácia de 99% na detecção dos níveis de anestesia.

2.4.6. Algoritmo *Backpropagation*

Um dos mais populares métodos para treinar redes MLP é o algoritmo *backpropagation* que está baseado no gradiente descendente do erro. Ele é poderoso, útil, fácil de entender e muito simples mesmo para modelar redes complexas.

O algoritmo aprende os pesos da rede neural multicamada, partindo de um conjunto fixo de unidades e interconexões. A sua função é tentar minimizar o erro quadrático entre os valores de saída da rede e os valores alvos (*targets*, em inglês) fornecidos.

Este algoritmo está baseado numa regra de aprendizagem que corrige o erro durante o treinamento (HAYKIN, 1994). O processo de aprendizagem está constituído de duas fases: uma fase de propagação do sinal (*feedforward*) e uma retropropagação do erro (*backpropagation*).

A aprendizagem começa com uma rede não treinada, à qual é apresentada uma entrada de treinamento e os valores desejados para a saída são determinados. A função erro é uma função

escalar dos pesos que é minimizada quando as saídas da rede coincidem com as saídas desejadas. Os pesos são ajustados para reduzir esta medida de erro.

O erro de treinamento J é a soma das diferenças entre a saída desejada t_j e a saída real z_j para o total de exemplos de treinamento:

$$J(\mathbf{w}) \equiv \frac{1}{2} \sum_{j=1}^M (t_j - z_j)^2 = \frac{1}{2} (\mathbf{t} - \mathbf{z})^2 \quad (5)$$

onde M é o total de exemplos de treinamento, \mathbf{t} representa o vetor das saídas desejadas (*targets*), \mathbf{z} é o vetor das saídas reais (ambos os vetores de comprimento M) e \mathbf{w} representa todos os pesos da rede (DUDA et al., 2001).

No caso de ter uma rede com múltiplas saídas, a equação (6) define J como a soma dos erros de todas as unidades de saída da rede:

$$J(\mathbf{w}) \equiv \frac{1}{2} \sum_{j=1}^M \sum_{i=1}^N (t_{ij} - z_{ij})^2 \quad (6)$$

onde N é o conjunto de unidades de saídas na rede, M é o total de exemplos de treinamento, t_{ij} e z_{ij} são os valores alvos e de saída associados com a unidade de saída i e com o exemplo de treinamento j (MITCHELL, 1997).

Como descrito acima, a regra de aprendizagem *backpropagation* está baseada no gradiente descendente, em que os pesos são inicializados com valores aleatórios, e são trocados para gradualmente reduzir o erro. Este ajuste de pesos pode ser descrito pela seguinte equação:

$$\Delta w_{ij} = -\eta \frac{\partial J}{\partial w_{ij}} \quad (7)$$

onde η é a taxa de aprendizado e $\frac{\partial J}{\partial w_{ij}}$ é a derivada parcial do erro J em relação aos pesos w_{ij}

(REZENDE, 2005).

O *backpropagation*, também chamado de Regra Delta generalizada, é uma extensão do método do gradiente para redes de multicamadas. Na primeira fase do treinamento o sinal de entrada é propagado para frente (fase *feedforward*) das entradas até a saída da rede. Aqui o valor da saída desejada t_i para a entrada atual x_i é conhecido, então o erro para a camada de

saída pode ser calculado e seus pesos ajustados. Mas como não existem valores de saída desejados para as camadas intermediárias da rede, o ajuste dos pesos das camadas intermediárias é feito através da propagação para trás do erro da camada de saída, caracterizando o treinamento como *backpropagation* (REZENDE, 2005).

A atualização do vetor de pesos na iteração i é realizado seguindo a equação (8) (DUDA et al., 2001):

$$\mathbf{w}(i+1) = \mathbf{w}(i) + \Delta \mathbf{w}(i) \quad (8)$$

Então, o conceito de retropropagação, ou propagação para trás, consiste na utilização do erro da camada de saída para estimar o erro dos neurônios das camadas escondidas.

Devido às não linearidades na superfície de erro, a função $J(\mathbf{w})$ pode apresentar irregularidades, causando o surgimento de mínimos locais e regiões planas de gradiente nulo e assim dificultar o treinamento (REZENDE, 2005). O problema de ter muitos mínimos locais na superfície do erro é que isso dificulta a descoberta do mínimo global e diminui o desempenho de classificação da rede treinada.

Apesar da falta de uma convergência garantida ao mínimo global, o *backpropagation* é um método de aproximação de funções altamente efetivo. Em muitas aplicações o problema dos mínimos locais não é tão grave. Quando o gradiente descendente cai num mínimo local com respeito a um dos pesos, isso não significa que necessariamente será um mínimo local também para os outros pesos. Na verdade, quantos mais pesos a rede tiver, mais dimensões terá que possam fornecer “rotas de escape” para que o gradiente descendente caia longe do mínimo local em relação a este único peso (MITCHELL, 1997).

O tamanho da taxa de aprendizado η determina a velocidade com que a rede atinge o mínimo global na função $J(\mathbf{w})$. Se alguns pesos convergem mais cedo que outros (aprendizado não uniforme) a rede não pode apresentar desempenho igualmente bom ao longo de todas as entradas. A figura 13 mostra o efeito das diferentes taxas de aprendizado na convergência em uma dimensão única. A taxa de aprendizado ótima é aquela que conduz ao mínimo erro local num único passo de aprendizado e pode ser calculada da seguinte forma (DUDA et al., 2001):

$$\eta_{opt} = \left(\frac{\partial^2 J}{\partial w^2} \right)^{-1} \quad (9)$$

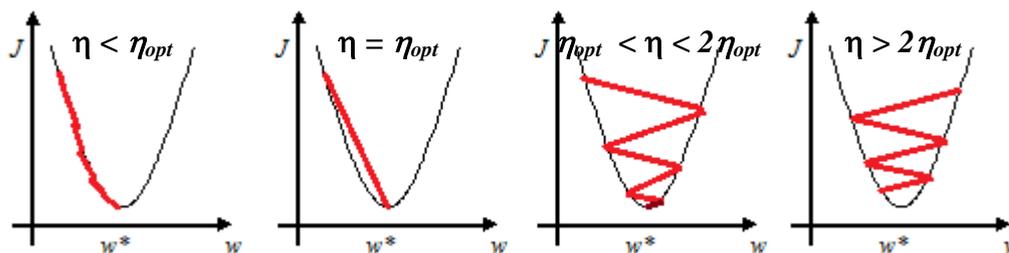


Figura 13 - Gradiente descendente em um critério quadrático unidimensional com diferentes taxas de aprendizado. Se $\eta < \eta_{opt}$ a convergência está assegurada; se $\eta = \eta_{opt}$ só é necessário um único passo de aprendizado para achar o mínimo erro; se $\eta_{opt} < \eta < 2\eta_{opt}$ o resultado irá oscilar mas, mesmo assim, irá convergir; e se $\eta > 2\eta_{opt}$ o resultado diverge.

Fonte: DUDA et al., 2001, cap. 6.8.9.

Desta forma a máxima taxa de aprendizado que irá convergir é $2\eta_{opt}$. Uma taxa de aprendizado η na faixa de $\eta_{opt} < \eta < 2\eta_{opt}$ vai resultar numa convergência mais lenta. Por conseguinte, para um aprendizado rápido e uniforme temos que calcular a segunda derivada da função critério em relação a cada peso e definir a melhor taxa de aprendizado para cada peso separadamente (DUDA et al., 2001).

Para um determinado conjunto de dados, o algoritmo de aprendizado deve ser responsável pela adaptação dos parâmetros da rede, para que em um número finito de iterações do algoritmo haja convergência para uma solução. O critério de convergência varia de acordo com o algoritmo e com o paradigma de aprendizado, mas pode envolver, por exemplo, a minimização de uma função-objetivo, a variação do erro de saída ou a variação das magnitudes dos vetores de peso da rede (REZENDE, 2005).

O objetivo do aprendizado é achar o vetor de pesos \mathbf{w} que faz a rede convergir durante o treinamento. A diferença entre os algoritmos de treinamento é justamente a forma de obter o vetor de pesos. Eles podem ser classificados como algoritmos de aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço (REZENDE, 2005).

No aprendizado supervisionado existe um supervisor externo que monitora a resposta \hat{y}_i da rede para cada vetor de entrada x_i . O conjunto de entrada está formado por pares de entrada e saída (x_i, y_i) , onde x_i representa o vetor de entrada e y_i representa o vetor de saída desejado. A função do algoritmo é ajustar os pesos de forma tal que a saída \hat{y}_i da rede para o vetor de entrada x_i se aproxime o suficiente à saída desejada y_i dentro de limites de tolerância estabelecidos (REZENDE, 2005).

O problema do aprendizado não supervisionado envolve a aprendizagem de padrões na entrada, quando não são fornecidos valores de saída específicos (RUSSELL; NORVIG, 2004). Neste aprendizado não existe um supervisor externo que controle as saídas. O conjunto

de treinamento está formado unicamente pelo vetor de entrada x_i . O ajuste dos pesos para obter vetor de pesos \mathbf{w} é realizado apenas com base nos valores dos vetores de entrada (REZENDE, 2005). Este tipo de aprendizado, baseado na regularidade e redundância das entradas, permite a descoberta das características estatisticamente relevantes nos dados de entrada, como a descoberta de agrupamentos ou classes (por ex. identificação de classes de usuários de cartões de crédito com base em seus perfis de compra).

O aprendizado por reforço é um paradigma intermediário entre o aprendizado supervisionado e o não supervisionado. Nele existe um crítico externo que supervisiona as saídas e retorna um sinal de reforço ou penalidade associado à última ação da rede. Caso a última ação tenha acarretado uma degradação no desempenho, ela será penalizada, tendo menor chance em situações futuras. Caso a ação tenha como resultado uma melhora no desempenho, ela será reforçada, ocorrendo com maior chance em situações futuras (REZENDE, 2005).

2.4.7. Estrutura Ideal da Rede – Número de Camadas e Neurônios Escondidos

O desempenho de uma RNA depende amplamente da sua arquitetura e da seleção de parâmetros (pesos). Em uma rede com número de parâmetros muito grande a mesma estará sujeita à sobreajuste (*overfitting*, em inglês), podendo memorizar todos os dados de treinamento, formando uma extensa tabela de busca. Isto não garante que esta rede irá realizar boas generalizações precisamente com os novos dados de entrada ao modelo que não foram utilizados no treinamento (RUSSELL; NORVIG, 2004).

Não existem orientações específicas para a concepção de uma arquitetura perfeita. A seleção de uma topologia adequada para um determinado problema baseia-se principalmente no método de tentativa e erro.

No entanto, nesta seção são apresentadas algumas propostas em relação ao número de camadas escondidas e ao número de neurônios em cada camada escondida.

As camadas escondidas não são necessárias em todos os casos. Mesmo que a função que se deseja aprender seja um pouco não linear, a generalização pode ser melhor com um modelo linear simples do que com um complicado modelo não linear, principalmente se não houver dados suficientes ou no caso de ter muito ruído para estimar as não linearidades com exatidão (SARLE, 1996).

Segundo DUDA et al. (2001) e SARLE (1996) podem-se especificar as seguintes regras:

- Redes sem camada escondida podem aproximar funções lineares.

- Redes com uma camada escondida podem aproximar qualquer função que contém um mapeamento contínuo de um espaço finito para um outro.
- Redes neurais com duas camadas escondidas podem representar funções com limites arbitrários de exatidão arbitrária.

A figura 14 apresenta o espaço de mapeamento para redes com nenhuma, uma ou duas camadas escondidas.

No caso de redes com só uma unidade de entrada, não se obtém nenhuma vantagem em utilizar mais de uma camada escondida (SARLE, 1996).

Para funções muito complicadas, com muitos vales ou colinas, é útil dispor de duas camadas escondidas e várias unidades na segunda camada escondida. Cada unidade na segunda camada escondida permite à rede ajustar cada colina ou vale separadamente (SARLE, 1996).

Está demonstrado empiricamente que as redes com várias camadas escondidas são mais propensas a ficar presas em mínimos locais indesejáveis. Por esse motivo, na ausência de uma razão específica para estabelecer múltiplas camadas escondidas é mais recomendável proceder usando apenas uma única camada escondida (DUDA et al., 2001).

Enquanto as quantidades de unidades de entrada e saída de uma rede neural são ditadas pela dimensionalidade dos vetores de entrada e de saída da rede, não existem regras para determinar o número correto de neurônios escondidos.

Usando muito poucos neurônios na camada escondida pode resultar em falta de ajuste (em inglês, *underfitting*), ou seja, o estimador não é flexível o suficiente para capturar as tendências subjacentes nos dados observados. Nestes casos, tanto o erro de treinamento como o de validação são muito grandes (HEATON, 2008, DUDA et al., 2001).

A utilização de muitos neurônios na camada escondida pode resultar em sobreajuste (*overfitting*). Nestes casos, a rede neural tem tanta capacidade de processamento de informações que a quantidade limitada de informação contida no conjunto de treinamento não é suficiente para treinar todos os neurônios nas camadas escondidas. Isto também implica um aumento no tempo de treinamento da rede (HEATON, 2008).

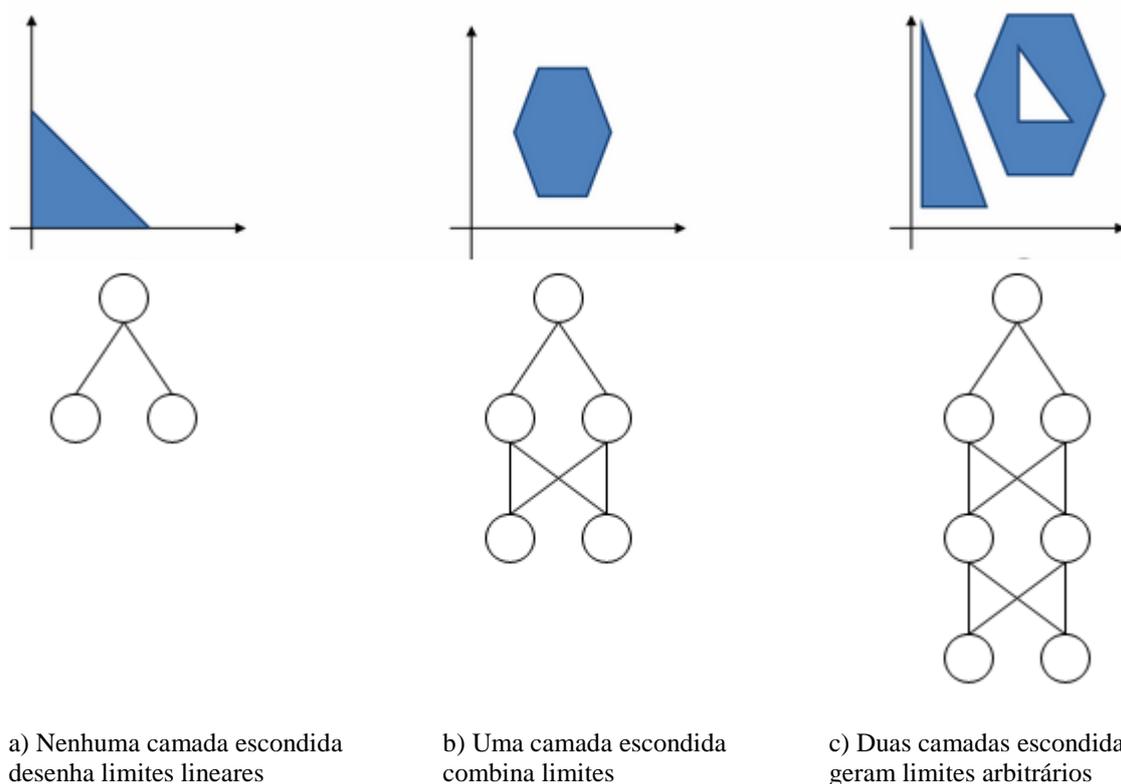


Figura 14 - Espaço de mapeamento para redes com nenhuma, uma ou duas camadas escondidas.

Existem algumas regras de ouro estabelecidas por alguns pesquisadores que auxiliam na pesquisa do número ótimo de neurônios escondidos.

Duda e colegas (2001) sugerem que o número total de pesos na rede seja aproximadamente $n/10$, onde n é o total de amostras de treinamento.

Heaton (2013) sugere que o número de neurônios escondidos tem que estar entre o número das unidades de entrada e o número das unidades de saída, mas não deve ser maior do que o dobro do número das unidades de entrada. Mais especificamente, o número de neurônios escondidos tem que ser $2/3$ da soma das unidades de entrada e das unidades de saída.

Lawrence (1994) propõe que a quantidade de neurônios escondidos seja calculada da seguinte forma: $j = m \cdot e$, onde m é a quantidade de dados no conjunto de treinamento e e é a tolerância de erro.

Baum e Haussler (1988) propõem que o número de neurônios na camada escondida tem que ser calculado usando a fórmula $j = \frac{m \cdot e}{n + z}$, onde j é o número de neurônios na camada escondida, m é a quantidade de dados no conjunto de treinamento, e é a tolerância de erro, n é o número de entradas e z é o número de saídas.

Outro método consiste em começar com um grande número de neurônios escondidos e depois com a técnica da poda remover conexões ou unidades desnecessárias.

Finalmente, a escolha da arquitetura da rede neural resultará de várias provas utilizando a técnica de tentativa e erro.

2.4.8. Critérios de Parada

Quando o treinamento da rede finaliza em função de ter encontrado um mínimo local, mas o erro de treinamento continua alto, isto usualmente significa que as características mais relevantes do problema ainda não foram aprendidas pela rede. Nesse caso, o melhor é alterar os pesos da rede e treiná-la novamente. No entanto, a convergência a um mínimo não global é aceitável no caso em que o erro seja relativamente baixo. Ou seja, não é absolutamente necessário que a rede convirja a um mínimo global para ter um desempenho aceitável (DUDA et al., 2001).

As arquiteturas de redes neurais padrões, como as redes multicamadas, são propensas ao sobreajuste. Enquanto a rede parece ficar cada vez melhor, ou seja o erro sobre o conjunto de treinamentos diminui, em algum momento durante o treinamento ela começa a piorar novamente, isto é, o erro em exemplos nunca vistos aumenta. A expectativa é que durante o treinamento o erro de generalização evolua segundo o mostrado na figura 15. O erro de generalização é estimado pelo erro de validação, ou seja, o erro médio no conjunto de validação, um conjunto fixo de exemplos que não estão no conjunto de treinamento (PRECHELT, 2012).

Existem vários métodos para a determinação do momento em que o treinamento de uma rede neural deve ser encerrado. Uma boa determinação destes critérios é fundamental para um bom treinamento e conseqüentemente uma boa generalização. Os critérios de parada mais utilizados são:

- **Número de ciclos:** Define o número de ciclos de treinamento, ou seja, o número de vezes em que o conjunto de treinamento é apresentado à rede. Um número excessivo de ciclos pode levar a rede à perda do poder de generalização⁵ (sobreajuste, do inglês *overfitting*). Por outro lado, com um pequeno número de ciclos a rede pode não chegar ao seu melhor desempenho (falta de ajuste, do inglês *underfitting*).

⁵ Uma boa generalização implica em um bom desempenho da rede com dados não vistos anteriormente.

- **Erro:** Consiste em encerrar o treinamento após o erro médio quadrático ficar abaixo de um valor α pré-definido. É importante lembrar que um erro médio quadrático muito pequeno não implica necessariamente em uma boa generalização. Neste caso o *backpropagation* é susceptível a sobreajustar os exemplos de treinamento ao custo da redução da acurácia sobre os outros exemplos não vistos (MITCHELL, 1997).
- **Combinação dos métodos anteriores:** Outro método de parada frequentemente utilizado é uma combinação dos dois métodos anteriores. Assim, o treinamento da rede é finalizado quando qualquer um dos critérios acima é atendido.
- **Parada antecipada:** Nesta técnica, conhecida em inglês como *early stopping* (PRECHELT, 2012), o treinamento é interrompido a cada x ciclos e é realizada uma estimativa do erro da rede sobre o conjunto de dados de validação. A partir do momento em que o erro medido no conjunto de validação apresentar crescimento, o treinamento é encerrado. O que se deseja com esta técnica é descobrir o momento exato em que a rede começa a perder generalização.

Os passos para realizar esta técnica são:

- 1) Dividir os dados em conjuntos de treinamento e de validação;
- 2) Treinar a rede usando só o conjunto de treinamento e avaliar o erro do conjunto de validação a cada k iterações (épocas);
- 3) Interromper o treinamento assim que o erro no conjunto de validação for maior do que era k iterações atrás;
- 4) Utilizar o conjunto de pesos anteriores como o resultado de treinamento.

2.4.9. Técnicas para Diminuir o Sobreajuste

Existem duas técnicas básicas para combater o problema de sobreajuste para o aprendizado com *backpropagation* (PRECHELT, 2012): reduzir a dimensão do espaço de parâmetros (dimensão da rede) ou reduzir o tamanho efetivo da dimensão de cada parâmetro (valores dos pesos); ou seja, por seleção de modelo ou por regularização⁶. Técnicas para produzir um espaço de parâmetros de dimensão reduzida são os *métodos construtivos*⁷ e os *métodos de poda*, descritos na subseção 2.4.10.

⁶ A regularização envolve adicionar uma penalidade para a função de erro. Esta penalidade pode ser o decaimento do peso, incentivando os pesos a permanecerem pequenos e assim evitar o sobreajuste.

⁷ Os métodos construtivos começam com uma arquitetura mínima de rede e adicionam neurônios até que uma solução satisfatória seja encontrada, caminho oposto ao utilizado pelos métodos de poda.

Técnicas para reduzir a dimensão de cada parâmetro são os procedimentos *decaimento de peso* e de *validação cruzada*.

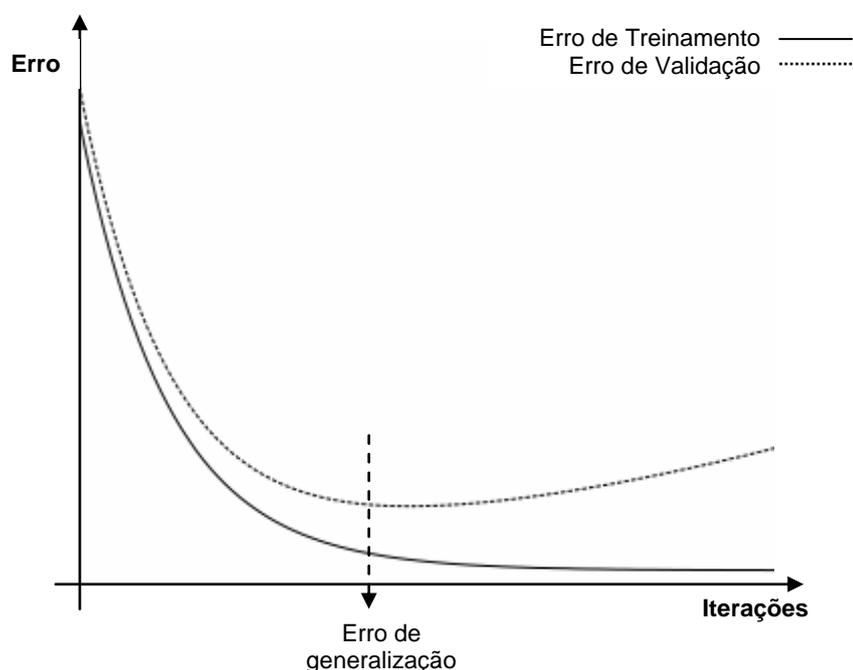


Figura 15 – Parada antecipada
O treinamento é interrompido quando o erro, para os dados de validação, começa a aumentar.
Fonte: PRECHELT, 2012, p. 56

O método conhecido como “decaimento do peso” (*weight decay*, em inglês) baseia-se na diminuição de cada peso em um pequeno fator durante cada iteração. Isto consiste em adicionar um termo de penalidade à definição de J mantendo os valores dos pesos pequenos e assim evitar superfícies de erro complexas (MITCHELL, 1997).

Outro método usado para diminuir o sobreajuste consiste em utilizar um conjunto de validação para monitorar o erro (validação cruzada), enquanto o conjunto de treinamento é usado para conduzir o gradiente descendente.

2.4.10. Poda da Rede

Uma forma de remover os pesos supérfluos é podando a rede. O princípio da poda consiste em estabelecer uma rede com um número relativamente grande de neurônios na camada intermediária, e durante a etapa de treinamento cortar as conexões (ou pesos) dos neurônios que possuem pouca influência, ou seja, desativar os pesos um de cada vez, até achar a arquitetura ótima (ROUSS, 2008). Neurônios que tiverem todas as conexões cortadas

serão eliminados, e portanto, ao final dos cortes, sobrarão somente os neurônios realmente necessários à modelagem, ou seja, além de remover conexões, também é possível remover unidades que não estejam contribuindo muito para o resultado (RUSSELL; NORVIG, 2004).

Após a rede ser treinada pela primeira vez, o algoritmo da poda identifica uma seleção ótima de conexões que podem ser descartadas. Então a rede é treinada novamente, e se seu desempenho não tiver diminuído, o processo é repetido.

Existem basicamente dois métodos para podar redes neurais: *Optimal Brain Damage* (OBD) e *Optimal Brain Surgeon* (OBS). Estes tentam apagar pesos mantendo o erro de treinamento tão baixo quanto possível. Ambos os métodos usam informação da derivada segunda do erro, ou seja, estes algoritmos estão à procura de mudanças no erro. Se a mudança no erro é pequena, o peso pode ser removido (DUDA et al., 2001).

No método OBD as conexões são cortadas durante a etapa de treinamento e a rede neural não é treinada após os cortes. No método OBS, as conexões são cortadas e, após o corte de uma conexão a rede é retreinada, permitindo que um número maior de cortes seja efetuado (CERQUEIRA et al., 2001). Além disso, para facilitar os tempos computacionais, o algoritmo OBD considera que a matriz Hessiana, que contém a segunda derivada dos erros em relação aos pesos é diagonal, enquanto o algoritmo OBS não faz nenhuma suposição a respeito e considera todos os termos da matriz. É por isso que o algoritmo OBS é considerado com melhor desempenho que o OBD (HASSIBI; STORK, 1992; DUDA et al., 2001).

Como conclusão, pode-se afirmar que a melhor arquitetura não é necessariamente aquela totalmente conectada, mas sim aquela que menor erro de generalização apresenta.

2.5. SÉRIES TEMPORAIS

A predição de séries temporais é uma disciplina que pode ser usada para apoiar a tomada de decisões em domínios como física, biologia, finanças, planejamento, gestão, manutenção e controle de processos industriais.

Uma série temporal é a realização de um processo ou um fenômeno variando no tempo. A análise das séries de tempo é um processo que permite inferir as características mais importantes do fenômeno, por meio da observação. Identificando essas variáveis físicas que influenciam o fenômeno é possível escrever uma fórmula paramétrica que simule o comportamento do fenômeno formando assim um modelo conceitual (PRUDENCIO et al., 2001). A predição dos valores futuros da série temporal permite decidir sobre uma estratégia de controle ou aperfeiçoar a atividade, produção, seleção, etc.

Mas quando não existe suficiente informação, ou o fenômeno é muito complexo, o uso de modelos caixa preta pode ser uma boa alternativa (SJOBERG et al., 1994). Estes modelam a série temporal ajustando parâmetros usando como entradas o valor atual e alguns anteriores. Exemplos destes modelos clássicos são os modelos AR (Autorregressivo) paramétrico linear, o MA (Média Móvel), o ARMA (Autorregressivos de Média Móvel) que é usado para tratar séries temporais estacionárias e o ARIMA (Modelos Autorregressivos Integrados de Média Móvel), de Box e Jenkins (1976), que baseia a sua predição na hipótese da existência de uma relação linear entre as entradas do modelo e a saída dele.

Mas existem muitos problemas em que estes modelos não podem ser aplicados devido à existência de relações não lineares no modelo a desenvolver, e isto limita a exatidão da estimação.

Uma ferramenta que permite trabalhar com modelos não lineares são as redes neurais artificiais. Elas são muito úteis para problemas cuja solução requer conhecimento difícil de especificar, mas para os quais existe uma abundância de exemplos. Elas apresentam melhor desempenho em comparação aos modelos lineares, mas a desvantagem que têm é que são susceptíveis a sobreajuste (*overfitting*) e aos mínimos locais (PRUDENCIO et al., 2001).

Segundo Nørgaard e colegas (2000), o procedimento que deve ser executado para identificar um sistema dinâmico consiste em alguns passos básicos, conforme mostrado na figura 16. Cada um dos passos do diagrama, serão explicados nas subseções 2.5.1 até 2.5.4.

2.5.1. Testes

Nesta fase experimental a coleta de dados é produzida. Os dados obtidos tomam a seguinte forma:

$$Z_N = \{[u(t), y(t)] / t = 1, \dots, N\}; \quad (10)$$

onde Z_N é o conjunto de treinamento; $\{u(t)\}$ é o conjunto de entradas (sinal); $\{y(t)\}$ representa o sinal de saída e 't' representa o instante da amostra. No caso em que o sistema apresente mais de um sinal de entrada e/ou saída $u(t)$ e $y(t)$ são representados por vetores.

O objetivo do treinamento é achar os pesos distintos (θ) que vão aproximar as saídas preditas ($\hat{y}(t)$) das saídas reais ($y(t)$). Desta forma a rede produzirá saídas mais perto da realidade.

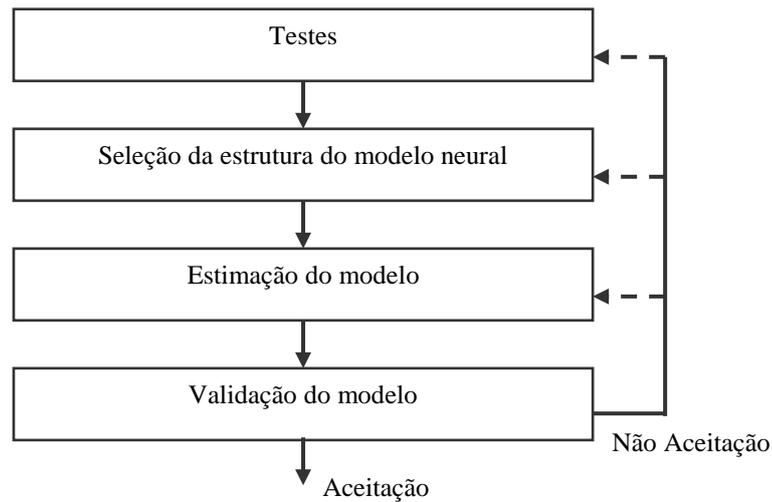


Figura 16 – Passos do enfoque da modelagem

2.5.2. Seleção da Estrutura do Modelo Neural

Uma vez obtidos os conjuntos de dados, o passo seguinte é selecionar a estrutura do modelo neural. Neste passo é preciso determinar os regressores do modelo e a arquitetura da rede. A determinação dos regressores consiste em descobrir quantos valores anteriores da série e quantas previsões passadas serão necessárias na previsão atual. Este tema está mais detalhado na subseção 2.5.6.

Dentro da família de modelos não lineares autorregressivos existem vários casos diferentes (NØRGAARD et al., 2000) (GAO; ER, 2005). Os modelos mais frequentemente usados são o NNAR (Rede Neural Autorregressiva), NNARX (Rede Neural Autorregressiva com entradas exógenas), NNARMA (Rede Neural Autorregressiva com média variável) e NNARMAX (Rede Neural Autorregressiva com média variável e entradas exógenas), os quais estão representados nas equações (11-14):

Modelo NNAR(n_y)

$$\varphi(t) = [y(t-1), \dots, y(t- n_y)]^T \quad (11)$$

Modelo NNARX(n_y, n_u)

$$\varphi(t) = [y(t-1), \dots, y(t- n_y), \dots, u(t-1), \dots, u(t- n_u)]^T \quad (12)$$

Modelo NNARMA(n_y, n_e)

$$\varphi(t) = [y(t-1), \dots, y(t-n_y), \dots, e(t-1), \dots, e(t-n_e)]^T \quad (13)$$

Modelo NNARMAX(n_y, n_u, n_e)

$$\varphi(t) = [y(t-1), \dots, y(t-n_y), \dots, u(t-1), \dots, u(t-n_u), \dots, e(t-1), \dots, e(t-n_e)]^T \quad (14)$$

onde $\varphi(t)$ é o vetor das entradas à rede neural, y são as saídas anteriores, e é o erro de predição ou ruído, e u são as entradas externas. As variáveis y , e e u representam as entradas ao sistema; n_y , n_u , n_e são os máximos atrasos (*lags*, em inglês) de cada entrada e o erro de predição está dado por $e(t) = y(t) - \hat{y}(t)$.

2.5.3. Estimação do Modelo

Uma vez que foi identificado o modelo para a análise da série temporal, a estimação do modelo permite minimizar o erro de predição achando os valores dos parâmetros do modelo (pesos). No modelo de Box-Jenkins, esta tarefa consiste na aplicação de uma técnica de regressão linear, mas nas redes neurais, consiste no treinamento do processo para achar os pesos apropriados (PRUDENCIO, 2001). O resultado do treinamento é um conjunto de pesos que constituem o modelo que dará a melhor predição. O método erro de predição tem como objetivo minimizar o seguinte critério:

$$V_N = \frac{1}{2N} \sum_{t=1}^N \left\{ [Y(t) - \hat{Y}(t | \theta)]^T [Y(t) - \hat{Y}(t | \theta)] \right\} \quad (15)$$

onde V_N é o erro de treinamento; N é o número de amostras no conjunto de treinamento; $Y(t) = [y^1(t), \dots, y^n(t)]$ é o vetor das saídas reais; $\hat{Y}(t) = [\hat{y}^1(t|\theta), \dots, \hat{y}^n(t|\theta)]$ é o vetor das saídas preditas; $\theta = [W_2 \ W_1]$ é o vetor dos pesos a ser definidos; W_1 é um vetor que contém os pesos entre as entradas e a camada escondida e W_2 contém os pesos entre a camada escondida e as saídas.

Para a escolha de um bom algoritmo de treinamento tem-se que levar em conta que um longo tempo de treinamento pode levar a um sobreajuste (*overfitting*, em inglês). Também é desejável que o algoritmo evite os mínimos locais (PRUDENCIO, 2001).

Uma vez achados os pesos no treinamento é possível realizar uma poda (*pruning*, em inglês) à rede. O método da poda está descrito com detalhes na subseção 2.4.10.

2.5.4. Validação do Modelo

Os métodos mais comuns para validar um modelo são a análise dos resíduos, por meio de uma correlação cruzada entre as entradas e o erro de predição $r_{U\varepsilon}(k)$, e a autocorrelação $r_{\varepsilon\varepsilon}(k)$ da predição.

A correlação cruzada compara uma sequência de dados com outra. É uma medida estatística do alinhamento e proximidade de dois conjuntos diferentes de informação e frequentemente é utilizada para achar um sinal de curta duração inserido num sinal mais longo.

A autocorrelação é a correlação cruzada entre uma série de tempo dada e uma versão defasada da mesma série ao longo de intervalos de tempos. Em outras palavras, é a correlação cruzada de um sinal com ele próprio. Permite encontrar padrões de repetição num sinal periódico com ruído. É frequentemente utilizada no processamento de sinais no domínio do tempo.

Dadas as medições Y_1, Y_2, \dots, Y_n , e as entradas U_1, U_2, \dots, U_n a função de correlação cruzada entre elas com atraso k é definida como:

$$r_{U\varepsilon}(k) = \frac{\sum_{t=1}^{N-k} (Y_{t-k} - \bar{Y})}{\sqrt{\sum_{t=1}^N (U(t) - \bar{U})^2 \sum_{t=1}^N (Y_t - \bar{Y})^2}} \quad (16)$$

Dadas as medições Y_1, Y_2, \dots, Y_n , a função de autocorrelação com atraso k é definida como:

$$r_{\varepsilon\varepsilon}(k) = \frac{\sum_{t=1}^{N-k} (Y_t - \bar{Y})(Y_{t-k} - \bar{Y})}{\sum_{t=1}^N (Y_t - \bar{Y})^2} \quad (17)$$

Se toda a informação relativa às variáveis mais importantes do sistema foi introduzida, o erro de predição $\varepsilon = Y_t - \bar{Y}$ é independente do conjunto particular de dados usados para a validação (ROUSS; CHARON, 2008).

2.5.5. Estrutura do Modelo NNARX

O modelo autorregressivo não linear com entradas exógenas (NNARX) como modelo para trabalhar com sistemas não lineares e séries temporais é apresentado aqui. A sua fórmula, descrita na equação 12, prediz uma série temporal \hat{y} onde as entradas passadas $u_1(t-1), \dots, u_1(t-e_1), \dots, u_n(t-1), \dots, u_n(t-e_n)$ e as saídas anteriores $y_1(t-1), \dots, y_1(t-s_1)$ são as entradas do modelo. Os valores e_1, \dots, e_n e s_1 são respectivamente os regressores de tempo das entradas e saída.

$$\hat{y}_i(t) = f[u_1(t-1), \dots, u_1(t-e_1), \dots, u_n(t-1), \dots, u_n(t-e_n), \dots, y_1(t-1), \dots, y_1(t-s_1)] \quad (18)$$

O modelo NNARX pode ser implementado de muitas maneiras, mas a mais simples é utilizando uma rede neural *feedforward* com uma memória incorporada, tal como mostrado na figura 17.

Estabelecer os regressores de entrada e saída (a ordem do modelo) é um dos pontos fundamentais ao modelar uma rede NNARX.

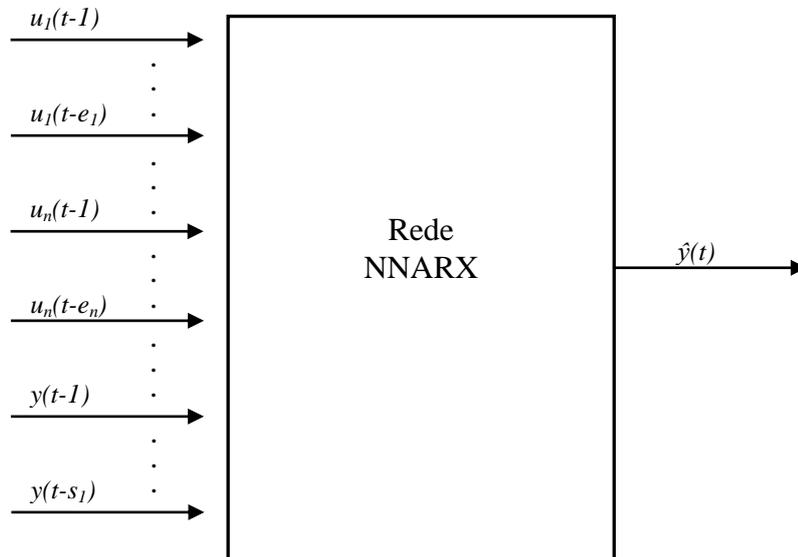


Figura 17 – Rede Neural com vetor de regressão NNARX

2.5.6. Seleção da Ordem do Modelo

A ordem do modelo pode ser definida como a complexidade do mesmo, pois determina a quantidade de entradas do modelo e o tempo de atraso (tempo em termos de número de amostras antes que a saída responda à entrada), e depende da estrutura do modelo.

A seleção da ordem do modelo pode ser feita usando diferentes critérios. Este ponto é muito importante, pois um modelo com uma ordem muito alta pode se ajustar demais aos dados (*overfitting*), e um modelo com uma ordem muito baixa conduz à insensibilidade ao ruído (AIBINU et al., 2008). Alguns dos critérios mais populares para selecionar a ordem do modelo são o Critério de Informação de Akaike (AIC), o Erro da Predição Final (FPE), o Coeficiente de Regressão de Lipschitz e o Comprimento da descrição mínima de Rissanen (MDL).

2.5.7. Critérios de Parada para Redes NNARX

Os critérios de parada da função são o erro de teste e o erro FPE (*Final Prediction Error*).

- Erro de teste: O erro de teste V_T é a estimativa mais confiável e, portanto, é razoável escolher a rede com menor erro de teste ao final (NØRGAARD et al., 2000). Este erro está baseado nos dados de validação $Z_T(U_2, Y_2)$ e por isso é completamente diferente do erro produzido com os dados de treinamento $Z_N(U_1, Y_1)$, o V_N (ver equação 15). Um valor de V_T perto de V_N significa que o modelo obtido após o treinamento é preciso. O erro de teste é composto por dois tipos de erros (ROUSS; CHARON; 2007):
 - O erro de viés que aparece quando não é possível achar o modelo ótimo. Isto acontece quando o modelo não tem neurônios suficientes no sistema e é chamado de *undertraining*.
 - O erro de variância que é devido a um excesso de neurônios na rede. Isto incrementa o número de mínimos locais e a variância dos pesos estimados e é chamado de *overtraining*.
- FPE (*Final Prediction Error*): A busca da ordem do modelo é feita minimizando a média do erro da variância em cada passo da predição. A fórmula do FPE é:

$$FPE = V \left(\frac{1 + d/N}{1 - d/N} \right) \quad (19)$$

onde V é função $loss^8$, d é o número de parâmetros do modelo e N é o número de amostras.

2.5.8. Aplicações em Séries Temporais

Todos os trabalhos achados pela autora de aplicações feitas com séries temporais aplicam-se para áreas de planificação, gestão, produção, manutenção e controle de processos industriais.

Dua e colegas (2011) tem trabalhado com séries temporais na área biomédica para classificar eventos de epilepsia e desenvolver métodos de mineração de padrões temporais.

Rouss e Charon (2008) desenvolveram um modelo mecânico não linear para um sistema de célula de combustível PEM utilizando uma abordagem neural combinando um perceptron multicamada com NNARMAX. O modelo pode prever o comportamento mecânico e as melhores posições dos acelerômetros.

San Roque e Czernichow (2009) apresentaram um trabalho para a predição da curva da demanda total de energia elétrica da França, utilizando um banco de dados dos anos 1990 a 1992.

Diaconescu (2008) apresentou um trabalho para a predição de séries temporais caóticas usando redes neurais NNARX, chegando à conclusão que a rede NNARX tem o potencial para capturar a dinâmica dos sistemas não lineares, com diferentes *delays*, mas tem problemas como a limitação no tempo de aprendizado e a instabilidade própria de todo sistema dinâmico.

2.6. VALIDAÇÃO DOS RESULTADOS

Existem várias opções para avaliar o desempenho das redes neurais. Nos casos de aprendizado supervisionado, é muito útil a construção de uma matriz de confusão que permite a visualização do desempenho de um classificador. Uma matriz de confusão ordena todos os casos do modelo em categorias, determinando se o valor previsto corresponde ao valor real.

Nela, cada linha da matriz representa os valores previstos do modelo (o que o algoritmo classificou), enquanto as colunas representam os valores reais. O número de acertos de cada classe se localiza na diagonal principal da matriz, enquanto os demais elementos da matriz representam erros de classificação. A matriz de confusão de um classificador ideal possui todos os elementos fora da diagonal principal iguais a zero, uma vez que ele não comete erros.

⁸ A função $loss$ mede a concordância entre o modelo e os dados, geralmente o erro quadrático médio de previsão a um passo à frente para um determinado conjunto de parâmetros (ARAHAL et al., 2006).

Na tabela 2 se apresenta uma matriz de confusão binária (2 classes), onde os diagnósticos corretos e incorretos são armazenados, tais como verdadeiros positivos (VP), falsos positivos (FP), verdadeiros negativos (VN) e falsos negativos (FN).

Os conceitos VP, FP, VN e FN são definidos da seguinte maneira.

- VN: define os casos em que o paciente estava saudável e que foram corretamente classificados.
- FP: denota os casos em que o paciente foi classificado erroneamente como doente, mas estava saudável.
- FN: denota os casos em que o paciente foi classificado erroneamente como saudável, mas estava doente.
- VP: define os casos em que o paciente foi corretamente classificado como doente.

Tabela 2 - Exemplo de uma matriz de confusão de 2 classes

		Realidade		
		Saudável	Doente	
Classificador	Saudável	(VN)	(FN)	Total Saudáveis Classificados = VN + FN
	Doente	(FP)	(VP)	Total Doentes Classificados = FP + VP
		Total Saudáveis Reais = VN + FP	Total Doentes Reais = FN + VP	Total de amostras = VN + FN + FP + VP

Partindo da matriz de confusão podem-se definir as equações de sensibilidade, especificidade e exatidão ou acurácia listadas abaixo. Quanto mais próximo de 1, melhor é o desempenho indicado por estas equações.

$$\text{Sensibilidade} = \frac{VP}{VP + FN} \quad (20)$$

$$\text{Especificidade} = \frac{VN}{VN + FP} \quad (21)$$

$$Acurácia = \frac{VP + VN}{VN + FN + VP + FP} \quad (22)$$

A sensibilidade (Se) é a probabilidade de classificar corretamente um paciente, com o valor definido como positivo. Indica a capacidade do estimador para classificar como positivos aqueles casos que realmente são positivos. Ou seja, a sensibilidade caracteriza a capacidade do estimador para detectar a doença em pacientes doentes (proporção de pacientes doentes corretamente identificados). O valor dessa classificação concorda com o estado real do paciente. Também é chamada de fracção de verdadeiros positivos (FVP).

A especificidade (Sp) indica a capacidade do estimador para classificar como negativos aqueles casos que realmente são negativos. É a capacidade do estimador para detectar a ausência da doença em pacientes saudáveis (proporção de pacientes saudáveis corretamente identificados). Também é chamada de fracção de verdadeiros negativos (FVN).

Já a acurácia avalia a efetividade geral do algoritmo

Para uma análise mais detalhada da rede utilizaremos também as seguintes três fórmulas de sensibilidade por classe $Se_{estável}$, $Se_{semiestável}$ e $Se_{instável}$, uma para cada classe (condição do paciente).

$$Sensibilidade_{estável} = \frac{\text{número de estados estáveis corretamente detectados}}{\text{número real de estados estáveis}} \quad (23)$$

$$Sensibilidade_{semiestável} = \frac{\text{número de estados semiestáveis corretamente detectados}}{\text{número real de estados semiestáveis}} \quad (24)$$

$$Sensibilidade_{instável} = \frac{\text{número de estados instáveis corretamente detectados}}{\text{número real de estados instáveis}} \quad (25)$$

Como medida de desempenho das redes analisadas foi utilizado o Erro Médio Quadrático (*MSE*, em inglês), cuja fórmula está representada na equação 26, onde Y_i é o vetor da saída real, \hat{Y}_i é o vetor da saída simulada e N é o número de amostras.

$$MSE = \frac{1}{N} \sum_{i=1}^N (Y_i - \hat{Y}_i)^2 \quad (26)$$

Outro indicador utilizado para medir o desempenho das redes foi o coeficiente de correlação. Esta é uma medida que analisa a relação entre duas variáveis. O coeficiente de correlação assume valores que variam entre +1 e -1. Os valores podem ser analisados da seguinte maneira:

- 0 indica que não existe uma relação;
- +1 indica uma relação linear perfeita positiva: quando uma variável aumenta, a outra variável também aumenta em uma regra linear exata;
- -1 indica uma relação linear perfeita negativa: quando uma variável aumenta, a outra variável decresce em uma regra linear exata;
- Valores entre 0 e 0,3 (ou 0 e -0,3) indicam uma relação linear fraca positiva (ou negativa) através de uma regra linear instável;
- Valores entre 0,3 e 0,7 (ou -0,3 e -0,7) indicam uma relação linear positiva (ou negativa) moderada;
- Valores entre 0,7 e 1 (ou -0,7 e -1) indicam uma relação linear positiva (ou negativa) forte através de uma regra linear firme.

A equação que caracteriza este indicador é dada por:

$$r = \frac{\sum_{i=1}^N (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^N (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^N (Y_i - \bar{Y})^2}} \quad (27)$$

onde X são as entradas, \bar{X} é a média aritmética das entradas, Y são as saídas e \bar{Y} é a média aritmética das saídas.

Para classificar a acurácia dos métodos avaliados, foi definida um critério, baseado na tradicional pontuação acadêmica, e apresentada na tabela 3.

Tabela 3 – Guia para classificar a exatidão dos métodos avaliados

Grau de exatidão	Diagnóstico
90 % – 100 %	Excelente
80 % – 90 %	Muito bom
70 % – 80 %	Bom
60 % – 70 %	Pobre
< 60 %	Muito pobre

Fonte: Autoria própria

2.7. REDES DE SENSORES BIOMÉDICOS

Em razão de que o sistema desenvolvido neste trabalho de tese tem o potencial para utilizar dados procedentes de uma rede de sensores biomédicos WBAN, é importante apresentar uma breve sinopse do que são os sensores biomédicos, as suas características físicas e suas aplicações.

Os sensores biomédicos são habitualmente usados em medicina e no campo biológico, a fim de medir uma grande variedade de sinais vitais. Quando o paciente está instável, ou após uma cirurgia complicada, normalmente estes sinais são medidos utilizando um monitor médico numa UTI. Para isso, o paciente tem que estar prostrado numa cama e preso a fios que o conectam ao monitor. O monitor apresenta os dados do paciente numa tela em formato de curvas e números, emitindo alarmes quando os valores são anormais.

Para oferecer maior comodidade e liberdade de movimentos aos pacientes, em muitos países já é frequente a utilização de Redes de Área Corporal sem Fios. Estas redes estão compostas por sensores pequenos, finos e leves distribuídos em torno do corpo humano com o propósito de monitorar as funções vitais do corpo e do meio ambiente. Neste caso o monitoramento tem que ser não invasivo e confiável.

Os sensores podem ser aplicados sobre o corpo como pequenos eletrodos inteligentes, integrados à roupa do paciente ou implantados sob a pele. Cada eletrodo ou nó sensor é capaz de processar e transmitir informações de uma estação base para diagnóstico e prescrição. A WBAN fornece vigilância da saúde em longo prazo de pacientes em estados fisiológicos naturais sem restringir suas atividades normais (ULLAH et al., 2010). Estas tecnologias possuem um enorme potencial no domínio do monitoramento ambulatorial e emergências (JOVANOVA et al., 2005). Uma das aplicações mais recentes da tecnologia sem fio

ambulatorial foi o caso dos mineiros chilenos presos a 2300 metros abaixo da superfície da terra por 69 dias. Os médicos foram capazes de controlar a sua frequência cardíaca, temperatura corporal, frequência respiratória, saturação de oxigênio e níveis sanguíneos de atividade de pressão à distância usando um transmissor Bluetooth (HEGUY, 2010).

Estudos mostram que os pacientes muitas vezes são admitidos em unidades de terapia intensiva por causa de necessidade de monitoramento, ao invés de ficar só com cuidados de enfermagem especializados. Sensores sem fio não só permitem o monitoramento de mais pacientes com menos pessoal em ambientes de saúde, mas também permitem aos médicos rapidamente adicionar ou remover os sensores quando as condições mudam. Os pacientes podem ser monitorados sob um ambiente menos estressante porque não estão presos aos equipamentos através de cabos. É importante ressaltar que este tipo de sensor permite aos médicos monitorar os sinais vitais essenciais dos pacientes, enquanto os médicos estão fora das áreas hospitalares especializadas (GENERAL ELECTRIC COMPANY, 2012). Além disso, pacientes com doenças crônicas podem ser monitorados continuamente em seus ambientes de origem (BASHSHUR, 2009). Isso melhora a qualidade do atendimento e da vida, permitindo uma redução significativa nos custos de tratamento e monitoramento.

2.7.1. Características Físicas dos Sensores

Os sensores biomédicos sem cabos devem satisfazer os requisitos de peso mínimo, formato mínimo, baixo consumo de energia para um monitoramento contínuo e prolongado. Além disso têm que ser de fácil integração a uma WBAN, com protocolos padrões, de fácil personalização e calibração (JOVANOVA et al., 2005).

Outra característica muito importante a avaliar no momento de decidir a compra de sensores biomédicos é o tempo de vida das baterias. O baixo consumo de energia é um fator muito importante na aquisição de quantidades grandes e contínuas de dados por longos períodos de tempo usando sensores compactos com quantidades limitadas de recursos energéticos sem precisar substituir as baterias frequentemente.

2.7.2. Privacidade na Transferência de Dados Médicos

A transferência de dados entre o servidor pessoal de um usuário e o servidor médico ou hospitalar exige que toda a informação sensível relacionada à saúde do paciente seja criptografada (WARREN, 2005). As mensagens podem ser criptografadas usando um *software* específico ou técnicas de *hardware*. Além disso, antes de uma possível inclusão dos

dados num banco de dados de pesquisa, todos os registros devem ser despojados de qualquer informação que se refira a um usuário específico.

2.7.3. Tolerância a Falhas

Alguns nós sensores podem deixar de funcionar corretamente devido à falta de energia, danos físicos ou ambientais, ou interferência. A falha de nós sensores não deve afetar a tarefa global da rede de sensores. A tolerância a falhas é a capacidade de sustentar as funcionalidades da rede de sensores sem qualquer interrupção, devido a falhas de sensores de nós individuais. No caso de um nó sensor parar de funcionar, um nó de back-up na vizinhança imediata pode assumir o papel do nó que falhou ou não está funcionando adequadamente, de modo que as medições críticas não sejam perdidas (BHARGAVA et al., 2003), (WARREN, 2005), (ZIGBEE, 2012).

2.7.4. Qualidade de Serviço (QoS)

Enquanto a QoS das redes celulares pode ser aceitável para comunicação de voz, pode não ser suficiente para as transmissões de dados médicos. Ainda hoje, especialmente nos países em desenvolvimento, existem problemas com as comunicações de telefonia celular. Esses problemas incluem conexões perdidas sem aviso prévio, falhas para estabelecer conexões, ou mesmo a degradação aparentemente leve do serviço. Em aplicações médicas, uma QoS baixa pode ter consequências graves, especialmente quando as redes sem fio são usadas para a transmissão sem fio de alarmes críticos de dispositivos médicos, ondas de dados fisiológicos contínuos, controle em tempo real de dispositivos médicos terapêuticos ou no caso de telemetria médica em tempo real.

A taxa de transferência de dados pode ser prejudicada, às vezes drasticamente, com um aumento no número de transmissores similares, num determinado local. Em muitos casos, é essencial que os dados médicos sejam transmitidos e recebidos sem erro (DRAFT GUIDANCE FOR INDUSTRY AND FDA STAFF, 2007).

2.7.5. Segurança e Confidencialidade dos Dados

Ao lidar com informações médicas dos pacientes em uma WBAN, é necessário prestar especial atenção na segurança da transmissão, detecção de falhas e no baixo consumo de energia. Maior segurança ainda é necessária para as redes sem fio na área da saúde, especialmente por causa de regulamentos rigorosos de privacidade do paciente e da

disponibilidade necessária de informações de saúde protegidas para entidades prescritas. A fim de obter resultados confiáveis, a WBAN deve detectar falhas de transmissão iniciais, bem como enviar sinais de alarme e avisos em tempo real. Detecção de falhas é, portanto, indispensável porque a transmissão de medições incorretas ou a perda de mensagens do sinal de alarme pode obviamente ter consequências drásticas.

2.7.6. Aplicações de Sensores Médicos

Podem-se encontrar aplicações de sensores médicos em diversas áreas como saúde, militar, monitoramento ambiental, casa e segurança. Algumas das aplicações na saúde incluem monitoramento de pacientes, administração de medicamentos, telemonitoramento de dados fisiológicos humanos e localização de médicos e pacientes dentro de um hospital (AKYILDIZ et al., 2002). Nesta área, os pesquisadores da Universidade de Southampton estão desenvolvendo sensores médicos inteligentes com capacidades de diagnóstico que podem ser usados pelos pacientes para controlar os seus sintomas, alertando ao médico de clínica geral se uma intervenção médica é necessária. Eles estão atualmente trabalhando em algoritmos avançados de ultrabaixa potência para o processamento de sinais e na criação de sensores médicos inteligentes com capacidade de tomada de decisão. Tais sensores precisam ser vestíveis (*wearable* em inglês, conhecidos também como e-têxteis), de baixa potência, capazes de funcionar em ambientes ruidosos, 24 horas por dia utilizando algoritmos de baixo consumo de energia, de modo que esta tecnologia permita ao médico do paciente ser alertado a qualquer momento, se a assistência médica for necessária (CHIRON, 2013).

2.7.7. Sistemas WBAN no Mercado

Existem muitos sistemas WBAN atualmente no mercado. Corventis (CORVENTIS, 2013) e CardioNet (CARDIONET, 2013) são dois exemplos que permitem o monitoramento dos sinais vitais e anormalidades cardíacas. Quando uma arritmia é detectada, eles adquirem um sinal de ECG e o transmitem (via zLink para Corventis) para o centro de monitoramento de saúde. Os pacientes que assinam o serviço CardioNet recebem monitoramentos de ECG em tempo real cujas medições são registradas continuamente, e automaticamente transmitidas através de um modem celular integrado sem fios num assistente digital pessoal (PDA). Os sinais de ECG são rastreados 24 horas por dia por técnicos da estação central, com

encaminhamento imediato ao médico para avaliação das mudanças de frequência ou ritmo ou em casos de sintomas cardíacos relevantes (ROTHMAN et al., 2007).

Em um projeto relacionado, o Centro Interuniversitário de Microeletrônica (IMEC) na Holanda desenvolveu um sistema de oito canais sem fio para um auricular de monitoramento ambulatorial de EEG (IMEC, 2013a), que transmite dados em tempo real para um receptor localizado até 10 metros de distância do sistema. Eles desenvolveram também um sistema de monitoramento de EEG de dois canais, sem bateria, e que é alimentado usando o calor do corpo e a luz ambiente, o que poderia ser utilizado para monitorar as ondas cerebrais após um ferimento na cabeça (IMEC, 2013b). Evidentemente, um sistema livre de bateria só é possível quando todos os componentes de tal sistema operam a níveis de potência extremamente baixa. Outras aplicações em tempo real dos equipamentos sem fio foram desenvolvidas para idosos, que além de detectar os sinais vitais, também podem detectar quedas (HALO, 2013), ou podem localizar pessoas com Alzheimer, autismo, demência, síndrome de Down ou outras condições cognitivas semelhantes, usando uma pulseira baseada em frequência de rádio (LOCATOR, 2012) ou sistemas de rastreamento GPS (ALERT, 2013). Há também outros produtos sem fio fáceis de usar que ajudam a monitorar a qualidade do sono (ZEO, 2013) (BLAKE et al., 2009), ronco, apneia e mesmo a intensidade do bruxismo (IME, 2013).

Pacientes com doenças crônicas podem ser continuamente monitorados nas suas casas, transferindo não apenas as informações médicas do paciente, mas também informação ambiental em tempo real. É possível conseguir interoperabilidade entre WBANs diferentes que monitoram um determinado paciente ou outros sensores ambientais circundantes (SCHATZ et al., 2012). No entanto, este tipo de interoperabilidade só é possível se os protocolos de comunicação empregados forem do tipo padrão, tais como Wi-Fi, Bluetooth ou ZigBee. Um exemplo é o trabalho de Gupta et al. (2010), em que eles desenvolveram uma plataforma de troca de dados para armazenar, compartilhar, pesquisar, visualizar e analisar dados de dispositivos heterogêneos, facilitando a interação entre eles.

Ao final de 2013 foi desenvolvido no Instituto de Tecnologia de Karlsruhe, na Alemanha, um cinturão de sensores capaz de monitorar a atividade cardíaca de um paciente durante seis meses, em forma não invasiva, tornando-se um recorde mundial de monitoramento cardíaco de pacientes. O cinturão é composto de quatro sensores “secos”, que não exigem pastas condutoras, e desta forma a irritação da pele pode ser evitada (KARLSRUHER INSTITUT FÜR TECHNOLOGIE, 2013).

3. PROPOSTA DE SISTEMA INTELIGENTE PARA MONITORAMENTO DOS SINAIS VITAIS DE PACIENTES EM TEMPO REAL

A finalidade deste trabalho é desenvolver um sistema que utilize técnicas inteligentes, que permita o monitoramento dos sinais vitais do paciente em tempo real a cada período determinado (nas arquiteturas apresentadas neste trabalho seria a cada segundo) e que permita estabelecer o estado do paciente no momento atual (t) assim como também o seu estado futuro ($t + d$), onde $d = 1$ neste trabalho.

Os estados nos quais um paciente pode-se achar são: estável, semiestável ou instável. Considera-se que um paciente está estável quando todos os seus sinais vitais estão dentro da faixa considerada normal. Um paciente acha-se semiestável quando um de seus sinais vitais tem um valor na faixa considerada alta ou baixa. Finalmente, o paciente estará instável quando dois ou mais sinais vitais estão fora da faixa considerada normal.

As medições realizadas a cada segundo foram pensadas para poder trabalhar com muitas mais amostras do que se fossem tomadas a cada cinco minutos. Os dados de um paciente com 2.000 amostras consistiriam em 2.000 amostras tomadas a cada segundo, mas também poderiam ser 33 amostras tomadas a cada minuto ou 6 amostras tomadas a cada cinco minutos. Evidentemente, tomar a medição a cada segundo permite treinar a rede com uma amostragem bem grande. A fim de evitar o *overfitting* e minimizar a probabilidade de aprendizado por periodicidade, os dados serão divididos em conjuntos e sua sequência será mudada, segundo o mostrado na seção 3.2.

Em princípio os alarmes são emitidos a cada momento em que o sistema detecta que o paciente está semiestável ou instável. Mas este ponto é facilmente configurável. Poder-se-ia pensar num sistema que controle o paciente a cada 3 minutos e que após duas ocasiões seguidas, nas quais o paciente estava semiestável, ou uma em que o paciente estava instável o alarme seja ativado. Este tipo de parametrização é facilmente modificável uma vez que o núcleo inteligente do *software* seja programado, e os resultados obtidos não seriam afetados.

O sistema proposto consiste principalmente em dois módulos: um módulo *fuzzy* para estabelecer o estado do paciente no momento atual (ou seja, no momento da medição dos sinais vitais) e um módulo composto por um modelo de rede neural que permita fazer a predição do estado do paciente no momento seguinte. Ambos módulos não se excluem

mutuamente. De fato, o módulo de rede neural artificial precisa do módulo *fuzzy* para gerar os *targets* com os quais a rede será treinada.

A decisão de utilizar a lógica *fuzzy* foi devido à facilidade na incorporação dos conhecimentos e experiência dos especialistas sem a necessidade de utilizar modelos matemáticos avançados. Uma desvantagem poderia ser que o ajuste dos resultados obtidos com o módulo *fuzzy*, de acordo com os especificados pelos especialistas, requer muito esforço, baseando o trabalho continuamente no método de tentativa e erro.

O módulo preditivo foi realizado utilizando uma rede neural recorrente, devido a que se dispõe de uma memória curta que permite estabelecer uma previsibilidade no resultado. O treinamento da rede não foi um problema por contar com um amplo conjunto de exemplos de sinais vitais obtidos do site Physionet (PHYSIONET, 2012), cujos *targets* (ou resultados esperados) são obtidos no módulo *fuzzy*.

Os diferentes modelos que aqui se apresentam estão baseados na aquisição de sinais vitais de pacientes em tempo real; no uso de técnicas inteligentes, tais como lógica *fuzzy* e redes neurais artificiais; no pré-processamento dos dados obtidos para gerar um resultado (estado do paciente); e na emissão de alarmes ante condições anormais nos parâmetros fisiológicos.

3.1. AQUISIÇÃO DE DADOS

A fim de simular a aquisição dos dados fisiológicos dos pacientes, foram utilizados os sinais fisiológicos já armazenados no site Physionet. Este site oferece livre acesso à comunidade de pesquisadores biomédicos a bancos de dados de sinais fisiológicos, neurais e cardiopulmonares de pacientes saudáveis e não saudáveis (pacientes com problemas cardíacos, de epilepsia, de apneia, de movimento e de idade) (PHYSIONET, 2012). Foram consultados três bancos de dados: o MIMIC⁹ II Waveform DB versão 2, MIMIC II Waveform DB versão 3 e MIMIC Database Numerics. O detalhe dos registros consultados de cada banco de dados é mostrado na tabela 4.

Uma vez que dos 28 registros consultados só dois apresentaram os dados de temperatura corporal, este parâmetro foi desconsiderado, baseando assim a construção do sistema inteligente nos outros cinco parâmetros restantes: frequência cardíaca (FC), pressão sistólica não invasiva (PS), pressão diastólica não invasiva (PD), pressão média não invasiva (PM) e saturação de oxigênio (SO).

O critério de seleção dos pacientes a serem utilizados levou em consideração a completude de dados fisiológicos e as diferentes variantes de estado do paciente, para os

⁹ Abreviatura de Multi-parameter Intelligent Monitoring for Intensive Care.

efeitos de ter uma maior variabilidade de amostras possíveis. Também foi importante analisar cada um dos vetores de entrada e evitar aquelas sequências onde os valores foram exatamente iguais, a fim de evitar problemas de divisão por zero ao obter os coeficientes de correlação linear. Para realizar o treinamento das redes foram tomados 2.000 dados de dez pacientes, e para realizar a validação das redes foram tomados 1.000 dados de outros dez pacientes.

Tabela 4 – Registros de dados fisiológicos consultados no site Physionet

Pac.	Banco de Dados	Registro	FC	PS	PD	PM	SO	T	Dados para Treinamento	Dados para Validação
P01	MIMICI3	3600039n	x	x	x	x	x		14.001-16.000	---
P02	MIMICI2	a45535n	x	x	x	x	x		11.001-13.000	---
P03	MIMICI2	a45580n	x	x	x	x	x	x	14.001-16.000	---
P04	MIMICI2	a45620n	x	x	x	x	x		18.001-20.000	---
P05	MIMICI2	a46072n	x	x	x	x	x		15.001-17.000	---
P06	MIMICI2	a46155n	x	x	x	x	x		39.501-41.500	---
P07	MIMICI2	a46297n	x	x	x	x	x	x	4.001-6.000	---
P08	MIMICI2	a45569	x	x	x	x	x		1-2.000	---
P09	MIMIC	240n	x	x	x	x	x		1-2.000	---
P10	MIMIC	441n	x	x	x	x	x		10.001-12.000	---
P11	MIMICI2	a46293n	x	x	x	x	x		---	3.000-3.999
P12	MIMICI2	a46384n	x	x	x	x	x		---	14.200-15.199
P13	MIMIC	214n	x	x	x	x	x		---	1-1.000
P14	MIMIC	215n	x	x	x	x	x		---	23.000-23.999
P15	MIMIC	440n	x	x	x	x	x		---	33.000-33.999
P16	MIMICI3	3601980n	x	x	x	x	x		---	23.200-24.199
P17	MIMICI2	a46169n	x	x	x	x	x		---	1.500-2.499
P18	MIMICI3	3900769n	x	x	x	x	x		---	6.000-6.999
P19	MIMICI3	3900790n	x	x	x	x	x		---	29.600-30.599
P20	MIMICI3	3900881n	x	x	x	x	x		---	15.000-15.999
P21	MIMIC	463n	x	x	x	x	x		---	---
P22	MIMICI2	a45625n	x	x	x	x	x		---	---
P23	MIMICI2	a46066n	x	x	x	x	x		---	---
P24	MIMICI2	a46215n	x	x	x	x	x		---	---
P25	MIMICI2	a46359n	x	x	x	x	x		---	---
P26	MIMICI2	a45534n	x	x	x	x	x		---	---
P27	MIMICI2	a45470n	x	x	x	x	x		---	---
P28	MIMICI2	n10150n	x	x	x	x	x		---	---

MIMICI2: MIMIC II Waveform DB, versão 2; MIMICI3: MIMIC II Waveform DB, versão 3; MIMIC: MIMIC Database Numerics;
 FC: Frequência cardíaca; PS: Pressão sistólica não invasiva; PD: Pressão diastólica não invasiva; PM: Pressão média não invasiva; SO: Saturação de oxigênio; T: Temperatura

3.2. PRÉ-PROCESSAMENTO DE DADOS

Uma vez que os dados obtidos tinham algumas seções com valores em zero, causadas provavelmente por desprendimento dos sensores, eles foram restaurados utilizando o valor medido anteriormente. Mas a escolha das sequências de dados para treinamento e validação foi feita considerando aquelas sem valores nulos em grandes extensões.

Para projetar e treinar o modelo *fuzzy* e os diferentes modelos de redes neurais artificiais, foram utilizadas dez matrizes de 5 x 2.000 (5 parâmetros e 2.000 amostras). Essas matrizes foram construídas a partir dos dados dos dez pacientes utilizados para treinamento, ou seja os registros dos pacientes P01 até P10. Na figura 18 é possível visualizar os dados do paciente P10 (Banco de Dados MIMIC, registro 441n). Dos sete sinais oferecidos só foram utilizados cinco deles que se encontram sinalizados pelas colunas tracejadas.

sample interval (1.024 sec)	HR (bpm)	NBPmean (mmHg)	NBPsys (mmHg)	NBPdias (mmHg)	PULSE (bpm)	RESP (bpm)	SpO2 (%)
0	48.000	57.000	101.000	37.000	48.000	10.000	97.000
1	48.000	57.000	101.000	37.000	48.000	10.000	97.000
2	48.000	57.000	101.000	37.000	48.000	10.000	97.000
3	48.000	57.000	101.000	37.000	48.000	10.000	97.000
4	48.000	57.000	101.000	37.000	48.000	10.000	97.000
5	48.000	57.000	101.000	37.000	48.000	10.000	97.000
6	48.000	57.000	101.000	37.000	48.000	10.000	97.000
7	48.000	57.000	101.000	37.000	48.000	10.000	97.000
8	48.000	57.000	101.000	37.000	48.000	10.000	97.000
9	48.000	57.000	101.000	37.000	48.000	10.000	97.000
10	48.000	57.000	101.000	37.000	48.000	10.000	97.000
11	48.000	57.000	101.000	37.000	48.000	10.000	97.000
12	48.000	57.000	101.000	37.000	48.000	10.000	97.000
13	48.000	57.000	101.000	37.000	48.000	10.000	97.000
14	48.000	57.000	101.000	37.000	48.000	10.000	97.000
15	48.000	57.000	101.000	37.000	48.000	10.000	97.000
16	48.000	57.000	101.000	37.000	48.000	10.000	97.000
17	48.000	57.000	101.000	37.000	48.000	10.000	97.000
18	48.000	57.000	101.000	37.000	48.000	10.000	97.000
19	49.000	57.000	101.000	37.000	48.000	10.000	97.000
20	49.000	57.000	101.000	37.000	48.000	10.000	97.000
21	48.000	57.000	101.000	37.000	48.000	10.000	97.000
22	48.000	57.000	101.000	37.000	48.000	0.000	97.000

Figura 18 – Dados obtidos da tela de Physionet (MIMIC – Reg.: 441n)

Fonte: Adaptado do site Physionet

Os conjuntos de dados dos dez pacientes utilizados para o treinamento foram divididos em três grupos de quantidades variáveis em cada caso, como pode ser observado na matriz desenhada na figura 19a. Primeiramente foram tomados os primeiros blocos de cada paciente, ou seja: os primeiros 300 do paciente 1, os primeiros 400 do paciente 2, os primeiros 1.250 do paciente 3 e assim sucessivamente até tomar os primeiros 600 dados do paciente 10. Em

seguida foram utilizados os dados 301 até 500 do paciente 1, os dados 401 até 800 do paciente 2 e assim até utilizar os dados 601 até 1.700 do paciente 10. Após utilizar o segundo bloco do paciente 10 foram utilizados os terceiros blocos de cada paciente. Desta forma o total de dados de pacientes forma uma matriz de 20.000 dados.

Paciente 1	1	300	301	500	501				2.000	
Paciente 2	1		400	401	800	801			2.000	
Paciente 3	1				1.250	1.251		1.800	1.801	2.000
Paciente 4	1				930	931		1.500	1.501	2.000
Paciente 5	1				850	851		1.600	1.601	2.000
Paciente 6	1				970	971		1.890	1.891	2.000
Paciente 7	1			750	751			1.500	1.501	2.000
Paciente 8	1				1.000	1.001		1.550	1.551	2.000
Paciente 9	1				980	981		1.900	1.901	2.000
Paciente 10	1		600	601				1.700	1.701	2.000

a) Dados de treinamento

Paciente 1	1				500	501		800	801	1.000
Paciente 2	1	300	301					700	701	1.000
Paciente 3	1	250	251					800	801	1.000
Paciente 4	1	330	331		550	551				1.000
Paciente 5	1			450	451	600	601			1.000
Paciente 6	1			470	471			890	891	1.000
Paciente 7	1					650	651	750	751	1.000
Paciente 8	1	100	101		550	551				1.000
Paciente 9	1			480	481			900	901	1.000
Paciente 10	1				600	601	700	701		1.000

b) – Dados de validação

Figura 19 - Divisão dos conjuntos de dados dos dez pacientes

Esta aleatoriedade nas amostras dos dez pacientes foi feita deliberadamente, para não provocar periodicidade nos dados. Desta forma, a rede está forçada a aprender o comportamento dos sinais do paciente com poucas amostras anteriores.

O mesmo critério foi utilizado para realizar o conjunto de dados para a validação, mas para este caso os blocos tomaram outros tamanhos, a fim de minimizar a probabilidade de aprendizado por periodicidade. Como para a validação foram tomados 1.000 dados de cada paciente, o total de dados para validar as redes é de 10.000. A figura 19b apresenta o critério utilizado para realizar a partição dos conjuntos.

3.3. ADAPTAÇÃO DA MATRIZ DE CONFUSÃO

Numa classificação binária a matriz de confusão é construída baseando-se em diagnósticos corretos e incorretos, tais como verdadeiros positivos (VP), falsos positivos (FP), verdadeiros negativos (VN) e falsos negativos (FN).

Neste caso, trata-se de uma classificação trinária, o que quer dizer que os resultados das redes testadas consistiriam em três classes diferentes: paciente estável, semiestável ou instável. Em função disto os critérios utilizados para criar a matriz de confusão são um tanto diferentes daqueles de uma classificação binária, dando como resultado uma matriz com as características mostradas na tabela 5.

Tabela 5 - Exemplo de uma matriz de confusão de 3 classes

		Realidade				
		Estável	Semiestável	Instável		
Classificador	Estável	a (VN)	b (FN)	c (FN)	Total Estáveis Classificados=	$a+b+c$
	Semiestável	d (FP)	e (VP)	f (FN)	Total Semiestáveis Classificados=	$d+e+f$
	Instável	g (FP)	h (FP)	i (VP)	Total Instáveis Classificados=	$g+h+i$
		Total Estáveis Reais = $a+d+g$	Total Semiestáveis Reais = $b+e+h$	Total Instáveis Reais = $c+f+i$	Total de amostras	
			/ / /		/ / /	

Fonte: Autoria própria

Os valores encontrados ao longo da diagonal principal da matriz de confusão são as classificações corretas e os que estão nos triângulos adjacentes à diagonal principal representam os erros (a confusão entre as classes).

A característica trinária da saída, define os conceitos VP, FP, VN e FN da seguinte maneira.

- VP: define os instantes de tempo onde o paciente estava não estável (semiestável e instável) e que foram corretamente classificados. Na matriz de confusão definida na tabela 5 seriam os valores e e i .
- FP: denota os instantes de tempo onde o estado do paciente foi classificado mais gravemente do que realmente era. Na matriz de confusão seriam os valores d , g e h .
- FN: denota os instantes de tempo onde o estado do paciente foi classificado menos gravemente do que realmente era. Na matriz de confusão seriam os valores b , c , e f .
- VN: define os instantes de tempo onde o paciente estava estável e que foram corretamente classificados. Na matriz de confusão seria o valor a .

Uma vez obtidos os valores de VN, VP, FP e FN podem ser utilizadas as fórmulas (20-25) de sensibilidade, especificidade, acurácia geral e sensibilidade por classes.

3.4. DEFINIÇÃO DO MODELO *FUZZY* PARA RECONHECIMENTO DO ESTADO DO PACIENTE

Os sistemas *fuzzy* não trabalham com equações e modelos, em vez disso usam regras *fuzzy* para inferências que simulam o raciocínio humano, e são criadas a partir do conhecimento de especialistas humanos. Uma vez que o conhecimento médico é inerentemente incompleto e impreciso, as ferramentas *fuzzy* são muito úteis, porque trabalham muito bem com conceitos ou conjuntos imprecisos, tais como: “algo de temperatura alta”, “pressão muito alta”, “leve melhora” ou “quase baixa frequência respiratória”.

É precisamente por esta característica que surgiu a ideia de se utilizar um modelo *fuzzy* para o reconhecimento do estado do paciente em tempo real. Utilizando cinco entradas, e com o auxílio dos especialistas consultados, formularam-se as regras do modelo *fuzzy* que contém o conhecimento dos especialistas humanos. No fluxograma da figura 20 é possível observar as distintas etapas para a criação do modelo *fuzzy*.

3.4.1. Definição das entradas do modelo

Este ponto consiste no estudo dos sinais vitais que servirão como entrada ao modelo *fuzzy* e a definição da faixa de valores normais, baixos, muito baixos, altos e muito altos de cada um deles em função dos dados obtidos da literatura médica.

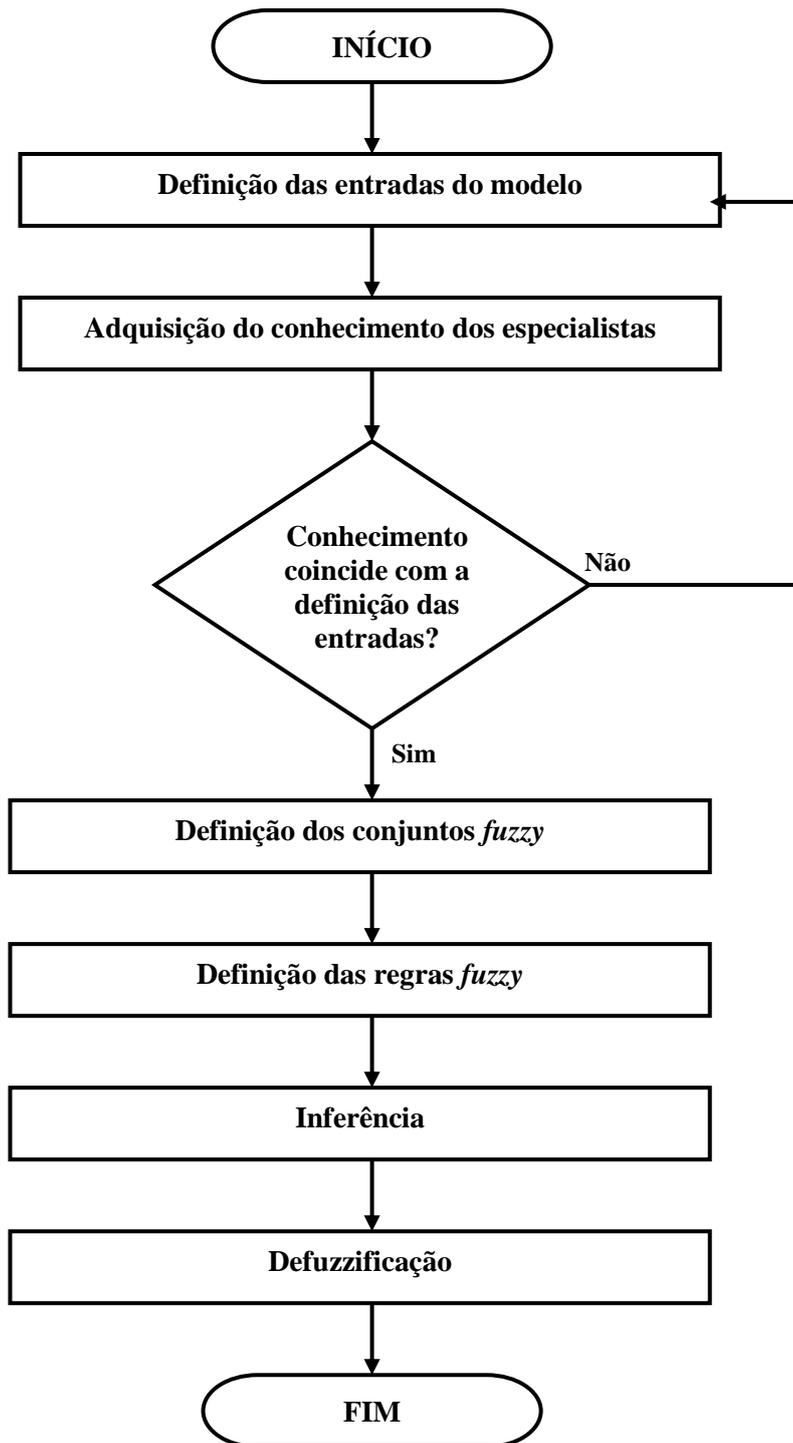


Figura 20 – Fluxograma com as distintas etapas da criação do modelo *fuzzy*
 Fonte: Adaptado de Leite et al. (2011, p. 1-17)

3.4.2. Aquisição do conhecimento dos especialistas

Foram consultados quatro especialistas sobre a importância de cada um dos sinais vitais a utilizar e suas correspondentes faixas de valores. Como os valores dos sinais vitais podem

variar de acordo com fatores como a idade, saúde do paciente ou a mobilidade, foi solicitado aos especialistas que os valores de cada sinal vital se correspondam com pacientes com doenças crônicas, mas que não precisam de hospitalização, e sim de um monitoramento periódico. Os valores foram calculados para adultos entre 50 – 65 anos de idade e o telemonitoramento será realizado com o paciente em estado de atividade normal diurna (não repouso)¹⁰. Em função desta informação foi necessário um reajuste na definição das entradas do modelo. A tabela 6 foi elaborada em forma conjunta com os especialistas consultados.

Tabela 6 – Valores dos sinais vitais não invasivos para um adulto saudável

	Muito Baixo	Baixo	Normal	Alto	Muito Alto
Frequência cardíaca (bpm)	< 40	40 – 60	60 – 100	100 – 130	> 130
Pressão Arterial Sistólica (mmHg)	< 90	90 - 100	100 – 130	130 – 140	> 140
Pressão Arterial Diastólica (mmHg)	< 60	60 - 70	70 – 85	85 – 90	> 90
Pressão Arterial Média (mmHg)	< 60	60 - 70	70 – 110	110 – 130	> 130
Saturação Oxigênio SpO2 (%)	< 90	90 – 94	94 – 100	---	---

Fontes: Dr. Hipólito Carraro Jr., médico intensivista do Hospital de Clínicas de Curitiba; Dr. Luciano Machado, médico intensivista do Hospital São Vicente de Curitiba; enfermeira Melissa de Costa, Chefe de enfermagem da UTI do Hospital São Vicente de Curitiba; enfermeira Karin Godarth, Chefe do Centro Cirúrgico do Hospital do Idoso Zilda Arns.

3.4.3. Definição dos conjuntos *fuzzy*

Logo após ter obtido o conhecimento dos especialistas, foram geradas as funções de pertinência para cada conjunto de entrada utilizando o MATLAB®, conforme os valores da tabela 6. A seguir, foram definidas as funções de pertinência para Frequência Cardíaca (FC), Pressão Sistólica Não Invasiva (PS), Pressão Diastólica Não Invasiva (PD), Pressão Média Não Invasiva (PM) e Saturação de Oxigênio (SO), respectivamente.

Cada um dos sinais vitais tem um domínio ou universo, que representa o conjunto de valores que o sinal pode assumir. Por sua vez, cada sinal pode ter várias funções de pertinência, que se correspondem às diferentes faixas de valores que o sinal pode ter. Estas funções de pertinência são representadas de várias formas, sendo as usadas neste trabalho as

¹⁰ Neste trabalho foram consideradas estas circunstâncias, mas num trabalho futuro poder-se-iam utilizar outros valores, outros pacientes ou inclusive, por exemplo, permitir ao médico alterar os valores predefinidos de acordo com as características do paciente ou com medicação ministrada.

formas triangular e trapezoidal. Cada função de pertinência triangular está representada por três pares de números da forma: $\{(a,0); (b,1); (c,0)\}$ indicando os limites do triângulo. Por sua vez, cada função de pertinência trapezoidal está representada por quatro pares de números da forma: $\{(a,0); (b,1); (c,1); (d,0)\}$ indicando também os limites do trapézio. Para exemplificar este ponto, ambas as figuras estão representadas na figura 21.

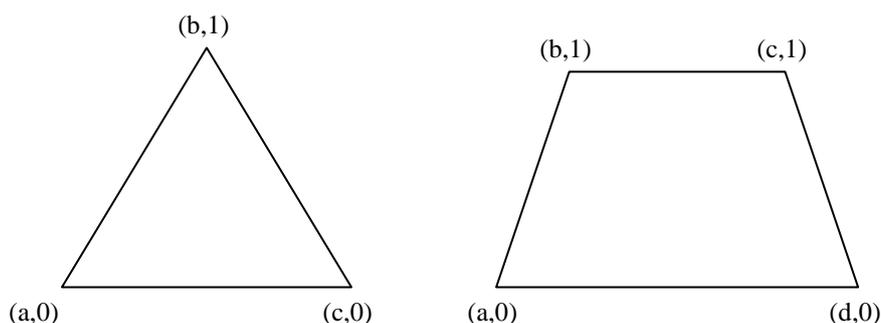


Figura 21 – Representação dos pares de cada forma de função de pertinência

Os valores destes pares foram obtidos por meio de tentativa e erro, buscando achar aqueles pares cujos resultados do modelo *fuzzy* estejam de acordo aos dados pelos especialistas, mostrados na tabela 6.

3.4.4. Conjunto *Fuzzy* de Entrada - Frequência Cardíaca

O domínio da variável FC é de $[0\ 200]$, o que quer dizer que o modelo *fuzzy* interpreta a FC entre valores de 0 a 200 bpm. Neste domínio, a FC pode se dividir em cinco faixas (ou funções de pertinência), representadas pelas seguintes variáveis linguísticas: Muito Baixa, Baixa, Normal, Alta e Muito Alta. As funções de pertinência para FC foram representadas com trapézios cujos parâmetros estão representados com quatro pares de números. A seguir, o detalhe de cada função de pertinência:

- i. FC Muito Baixa = $\{(0,1); (0,1); (35,1); (50,0)\}$;
- ii. FC Baixa = $\{(38,0); (45,1); (55,1); (62,0)\}$;
- iii. FC Normal = $\{(55,0); (65,1); (95,1); (105,0)\}$;
- iv. FC Alta = $\{(98,0); (108,1); (125,1); (135,0)\}$;
- v. FC Muito Alta = $\{(125,0); (135,1); (200,1); (200,1)\}$;

A figura 22 apresenta as cinco funções de pertinência para a variável FC.

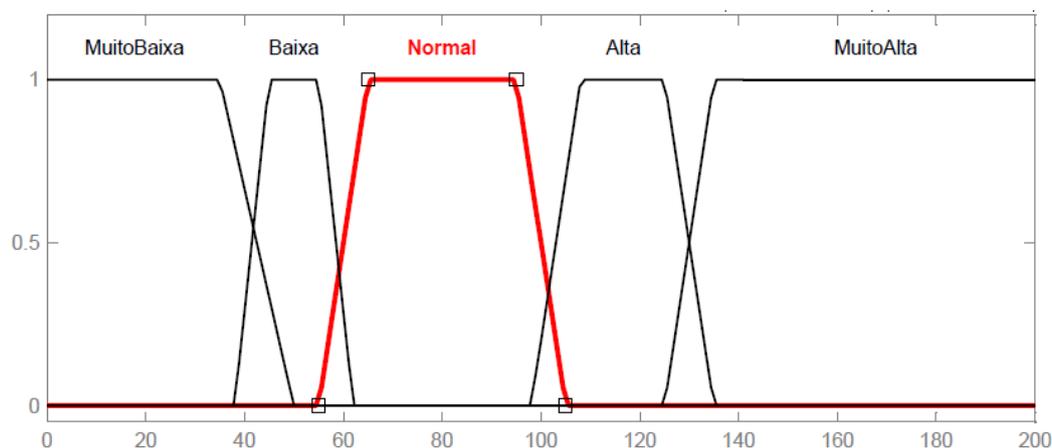


Figura 22 – Funções de pertinência para a Frequência Cardíaca

3.4.5. Conjunto *Fuzzy* de Entrada - Pressão Sistólica Não Invasiva

O domínio da variável PS é [0 200], o que indica que o modelo *fuzzy* interpreta a PS entre 0 e 200 mmHg. Neste domínio, a PS está representada por cinco funções de pertinência com as seguintes variáveis linguísticas: Muito Baixa, Baixa, Normal, Alta e Muito Alta. As funções de pertinência para PS, que foram representadas com trapézios e triângulos, têm os seguintes parâmetros:

- i. PS Muito Baixa = $\{(0,1); (0,1); (85,1); (95,0)\}$;
- ii. PS Baixa = $\{(85,0); (95,1); (105,0)\}$;
- iii. PS Normal = $\{(95,0); (105,1); (125,1); (135,0)\}$;
- iv. PS Alta = $\{(128,0); (135,1); (142,0)\}$;
- v. PS Muito Alta = $\{(135,0); (148,1); (200,1); (200,1)\}$;

A figura 23 apresenta as cinco funções de pertinência para a variável PS.

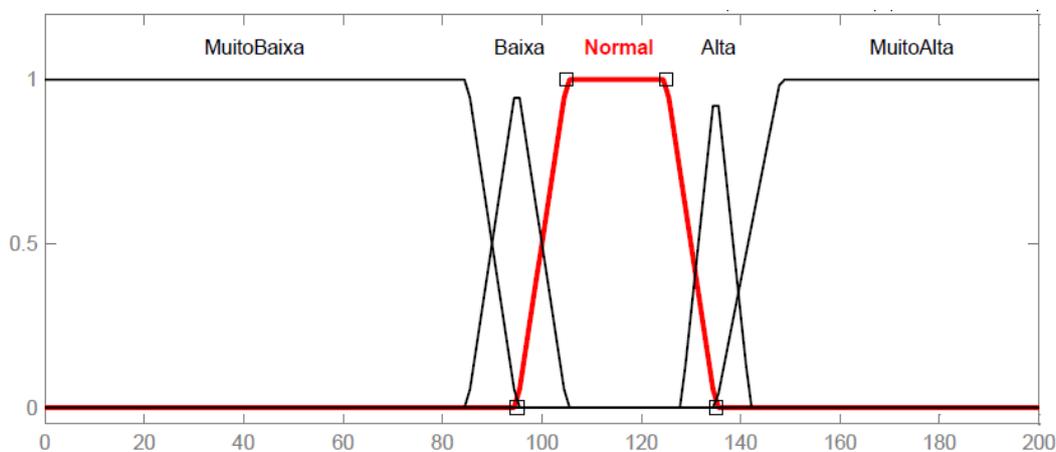


Figura 23 – Funções de pertinência para a Pressão Sistólica Não Invasiva

3.4.6. Conjunto *Fuzzy* de Entrada - Pressão Diastólica Não Invasiva

O domínio da variável PD é [0 150], ou seja, o modelo *fuzzy* interpreta a PD entre 0 a 150 mmHg. Neste domínio a PD está representada por cinco funções de pertinência com as seguintes variáveis linguísticas: Muito Baixa, Baixa, Normal, Alta e Muito Alta. As funções de pertinência para PD, representadas com trapézios e triângulos, têm os seguintes parâmetros:

- i. PD Muito Baixa = $\{(0,1); (0,1); (55,1); (62,0)\}$;
- ii. PD Baixa = $\{(0,55); (65,1); (75,0)\}$;
- iii. PD Normal = $\{(67,0); (73,1); (82,1); (90,0)\}$;
- iv. PD Alta = $\{(82,0); (87,1); (93,0)\}$;
- v. PD Muito Alta = $\{(86,0); (94,1); (150,1); (150,1)\}$;

A figura 24 apresenta as cinco funções de pertinência para a variável PD.

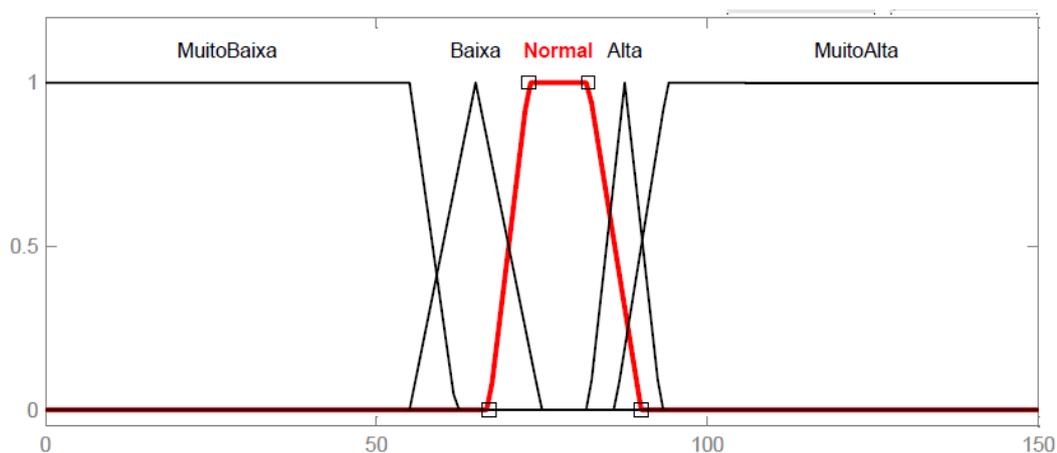


Figura 24 – Funções de pertinência para a Pressão Diastólica Não Invasiva

3.4.7. Conjunto *Fuzzy* de Entrada - Pressão Média Não Invasiva

O domínio da variável PM é [0 170], o que significa que o modelo *fuzzy* interpreta a PM entre 0 e 170 mmHg. Neste domínio, a PM está constituída por cinco funções de pertinência, as quais têm as seguintes variáveis linguísticas: Muito Baixa, Baixa, Normal, Alta e Muito Alta. As funções de pertinência para PM, também representadas com trapézios e triângulos, têm os seguintes parâmetros:

- i. PM Muito Baixa = $\{(0,1); (0,1); (55,1); (65,0)\}$;
- ii. PM Baixa = $\{(58,0); (65,1); (72,0)\}$;
- iii. PM Normal = $\{(65,0); (75,1); (105,1); (115,0)\}$;
- iv. PM Alta = $\{(105,0); (115,1); (125,1); (135,0)\}$;

- v. PM Muito Alta = $\{(125,0); (135,1); (170,1); (170,1)\}$;

A figura 25 apresenta as cinco funções de pertinência para a variável PM.

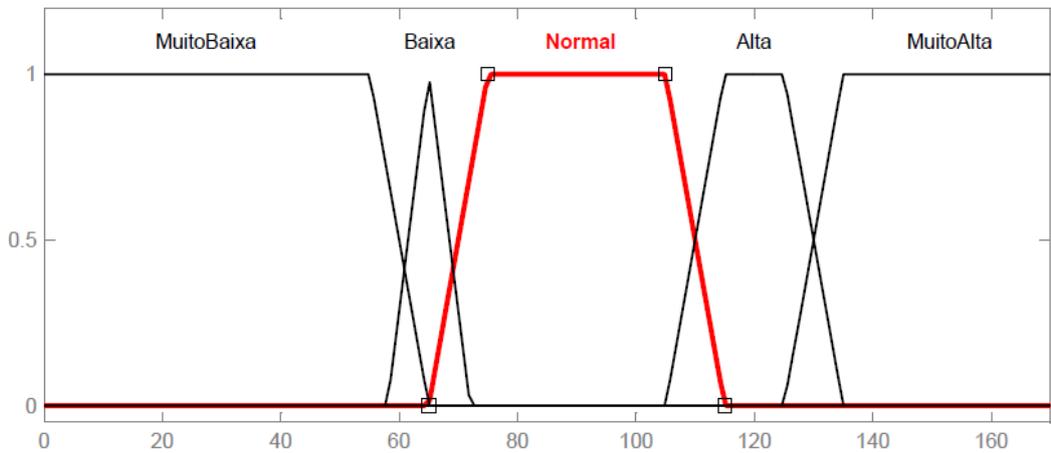


Figura 25 – Funções de pertinência para a Pressão Média Não Invasiva

3.4.8. Conjunto *Fuzzy* de Entrada - Saturação de Oxigênio

O domínio da variável SO é $[0\ 100]$, o que indica que o modelo *fuzzy* interpreta a SO entre 0 e 100%. Neste domínio a SO está representada por três funções de pertinência com as variáveis linguísticas: Muito Baixa, Baixa e Normal. Como é um valor medido em porcentagem, não existem valores superiores a 100, por conseguinte não foram consideradas as variáveis linguísticas Alta e Muito Alta. As funções pertinência para SO, representadas por dois trapézios e um triângulo, têm os seguintes parâmetros:

- i. SO Muito Baixa = $\{(0,1); (0,1); (86,1); (92,0)\}$;
- ii. SO Baixa = $\{(88,0); (92,1); (95,0)\}$;
- iii. SO Normal = $\{(92,0); (96,1); (100,1); (100,1)\}$;

A figura 26 apresenta as três funções de pertinência para a variável PS.

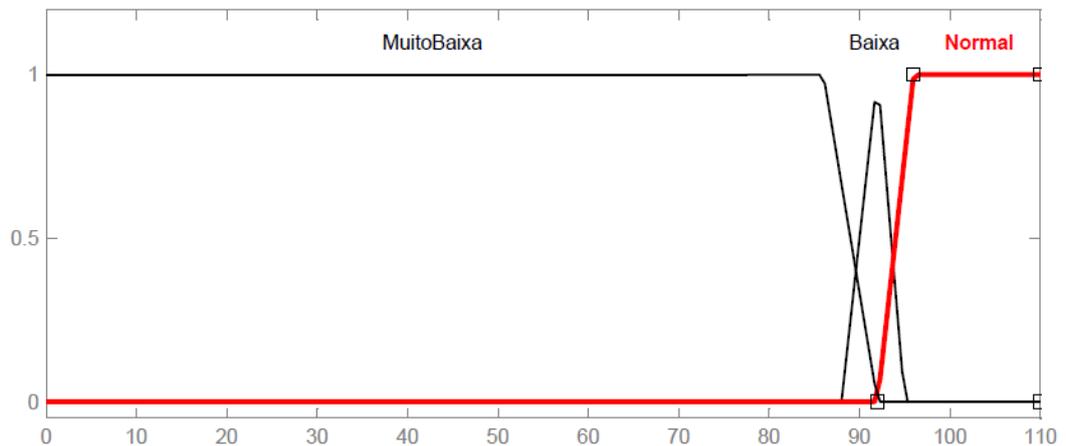


Figura 26 – Funções de pertinência para a Saturação de Oxigênio

3.4.9. Conjunto *Fuzzy* de Saída – Condição do Paciente

A saída representa o estado ou condição do paciente em um determinado momento e depende dos valores dos sinais vitais utilizados como entrada ao modelo *fuzzy*. O domínio da saída é [0 7], ou seja, os valores que a saída pode tomar estão entre 0 e 7. Logo, para poder interpretar estes valores, se definiram três variáveis linguísticas, as quais são os três estados possíveis, em que um paciente encontra-se num determinado momento. Eles são: Estável (sem alarme), Semiestável (alarme médio) e Instável (alarme forte).

Os parâmetros de cada uma das funções de pertinência são:

- i. CONDIÇÃO Estável = $\{(0,1); (0,1); (0.5,1); (1.5,0)\}$;
- ii. CONDIÇÃO Semiestável = $\{(0.5,0); (1.5,1); (2.5,0)\}$;
- iii. CONDIÇÃO Instável = $\{(1.5,0); (2.5,1); (7,1); (7,1)\}$;

A figura 27 apresenta as três funções de pertinência para a variável Condição do Paciente.

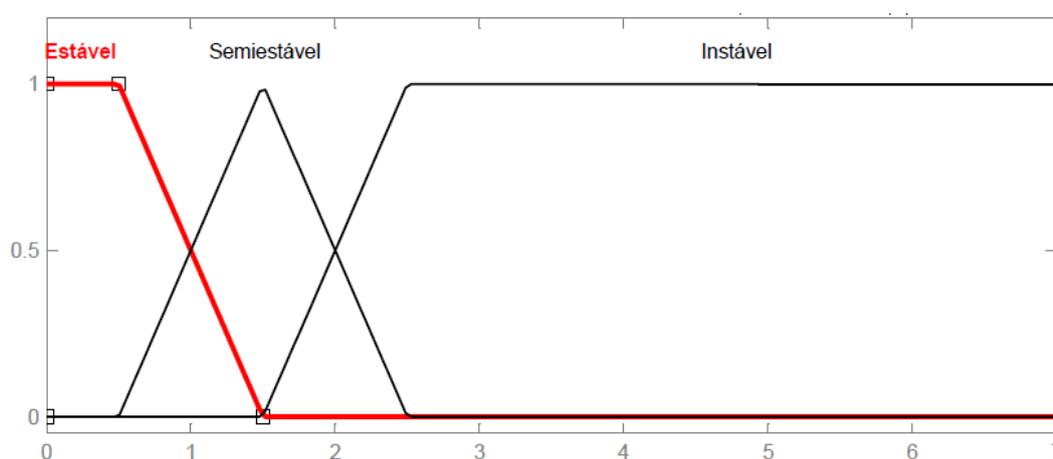


Figura 27 – Funções de pertinência para a saída de Condição do Paciente

Logo a saída tem que ser defuzzificada para obter uma saída crisp ou representativa. A defuzzificação é vista na subseção 3.4.12.

3.4.10. Definição das regras *fuzzy*

As regras foram criadas com a seguinte estrutura: SE <premissas> ENTÃO <conclusão>. Cada regra representa um possível estado do paciente, ou seja uma particular combinação dos conjuntos *fuzzy*. O propósito das regras é conseguir que o modelo *fuzzy* chegue a uma conclusão (estado do paciente). Lembrando a relação entre o número de regras e a velocidade de cálculo do modelo *fuzzy*, é importante minimizar o número de regras sem

perder a confiabilidade no resultado. O critério adotado para a geração das regras foi o seguinte:

- a. Num momento determinado, se o paciente apresenta um sinal com valor Muito Baixo ou Muito Alto é considerado o estado INSTÁVEL (alarme forte).
- b. Num momento determinado, se o paciente apresenta dois sinais com valor Baixo, ou dois sinais com valor Alto, ou um sinal com valor Baixo e outro com valor Alto é considerado o estado INSTÁVEL (alarme forte).
- c. Num momento determinado, se o paciente apresenta um sinal com valor Baixo, ou um com valor Alto e todos os outros com valor Normal é considerado o estado SEMIESTÁVEL (alarme médio).
- d. Num momento determinado, se o paciente apresenta todos os sinais com valor Normal é considerado o estado ESTÁVEL (sem alarme).

A construção das regras foi baseada nas normativas especificadas anteriormente chegando a contar com 51 regras. Alguns exemplos das regras geradas são mostrados a seguir:

Considerando que

MB=Muito Baixo, B=Baixo, N=Normal, A=Alto, MA=Muito Alto, E=Estável,

SE=Semiestável e I=Instável.

Regra 1: **If** (FC=N and PS=N and PD=N and PM=N and SO=N) **Then** (Status=E)

Regra 2: **If** (FC=B and PS=N and PD=N and PM=N and SO=N) **Then** (Status = SE)

Regra 3: **If** (FC=N and PS=A and PD=N and PM=N and SO=N) **Then** (Status =SE)

Regra 4: **If** (SO=MB) **Then** (Status=I)

Regra 5: **If** (FC=B and SO=B) **Then** (Status=I)

Usando estas regras, cada valor de cada conjunto de entrada é analisado para obter um respectivo grau de pertinência no conjunto de saída.

3.4.11. Inferência

Neste modelo *fuzzy* foi utilizada a abordagem de Mamdani (MAMDANI, 1974), na qual tanto o antecedente como o consequente das regras são variáveis linguísticas. Ao usar este método, por cada valor de entrada são obtidos distintos graus de pertinência para cada um deles. O sistema de inferência determina o grau de compatibilidade destes graus de pertinência com as regras da base de regras. O modelo adiciona todas as saídas de cada regra e as combina num único conjunto *fuzzy* para obter a saída geral do sistema.

3.4.12. Defuzzificação

Esta etapa é usada para transformar o conjunto *fuzzy* de saída obtido no processo de inferência, ou seja o conjunto de variáveis linguísticas com seus respectivos graus de pertinência, num número único e simples. Neste passo, a saída inferida é defuzzificada, usando o procedimento Média do Máximo (Mean of Maximum, MoM), para obter um valor exato. Neste, a saída precisa é obtida tomando-se a média do valor máximo do conjunto de saída.

A correspondência entre os valores de defuzzificação e as variáveis linguísticas da saída está representada na tabela 7.

Tabela 7 – Relação: Valores defuzzificados vs. Variáveis linguísticas de saída

Saída do modelo <i>fuzzy</i>	Estado do Paciente	Ação a executar
< 1	Estável	Nenhuma
1 – 2	Semiestável	Alerta médio ao especialista
> 2	Instável	Alerta forte ao especialista

3.5. ANÁLISE DAS DISTINTAS REDES NEURAI ARTIFICIAIS

Com a finalidade de oferecer ao paciente e ao seu cuidador uma alternativa para predizer o estado do paciente num futuro próximo ($t + d$), onde $d = 1, 2, \dots$ segundos, foram analisadas cinco arquiteturas de redes neurais artificiais. Três delas são diferentes abordagens da rede recorrente neural Elman (MISO e MIMO com e sem poda) e os outros modelos correspondem à rede neural NNARX, utilizada na identificação de sistemas dinâmicos não lineares. O critério de seleção depende da eficiência das estruturas. Portanto, o desempenho das distintas redes foi testado e comparado.

Para o treinamento e validação das redes neurais estudadas foram utilizados os dados obtidos do site Physiobank (ver seção 3.1). Foram utilizados dez registros de pacientes de diferentes bancos de dados. Para o treinamento das redes se utilizaram 2.000 amostras dos dez pacientes, ou seja a rede foi treinada com 20.000 amostras, e para as validações foram utilizados 1.000 amostras de outros dez pacientes, somando um total de 10.000 amostras.

As arquiteturas de redes neurais analisadas trabalham com as mesmas cinco entradas (FC, PS, PD, PM e SO).

O critério de parada utilizado para treinar as redes foi o de parada antecipada (ver subseção 2.4.8) o qual foi realizado da seguinte forma: o conjunto de treinamento que tem 20.000 amostras foi dividido em 10 partes, cada uma de 2.000 amostras. Do mesmo modo, o conjunto de validação de 10.000 amostras também foi dividido em 10 partes de 1.000 amostras cada uma. O treinamento foi interrompido a cada 200 ciclos e uma estimativa do erro foi realizada no conjunto de validação. Em cada corrida foram utilizados dados distintos de treinamento e validação. Este processo foi repetido 10 vezes para cada combinação (*algoritmo de treinamento, quantidade de neurônios na camada escondida*) e logo foi realizada uma média aritmética dos erros e dos tempos de corrida.

Todas as arquiteturas de redes propostas terão a continuação do nome o trio (*ni, no, ns*), onde *ni* indica a quantidade de entradas à rede, *no* representa a quantidade de neurônios ocultos na camada oculta e *ns* indica a quantidade de saídas da rede.

3.6. ARQUITETURA REDE RECORRENTE NEURAL ELMAN MISO – E-MISO

Nesta primeira arquitetura, a rede neural foi construída com as cinco entradas que correspondem aos sinais vitais dos pacientes, e uma saída que corresponde ao estado do paciente no tempo ($t + d$), com $d = 1$ segundo. Para obter os *targets* que permitam treinar a rede, os vetores de entrada foram pré-processados utilizando o modelo *fuzzy* proposto na seção 3.4. Deste modo a rede é treinada com os dados fisiológicos dos dez pacientes e os *targets* gerados com o modelo *fuzzy* já desenvolvido. Na tabela 8 é possível observar as entradas e a saída do modelo Elman MISO.

A rede Elman MISO, cujo diagrama esquemático está apresentado na figura 28, utiliza a função de transferência tangente hiperbólica sigmoide (em MATLAB conhecida como *tansig*) para as unidades de entrada, tem uma camada escondida e na unidade de saída a função de transferência utilizada é a de transferência linear (também conhecida como *purelin*).

Tabela 8 – Especificação das entradas e saídas do modelo Elman MISO

Entradas	Saídas
$u_1(t)$ = frequência cardíaca	$y(t+1)$ = estado do paciente
$u_2(t)$ = pressão arterial sistólica	
$u_3(t)$ = pressão arterial diastólica	
$u_4(t)$ = pressão arterial média	
$u_5(t)$ = saturação de oxigênio	

Fonte: Autoria própria

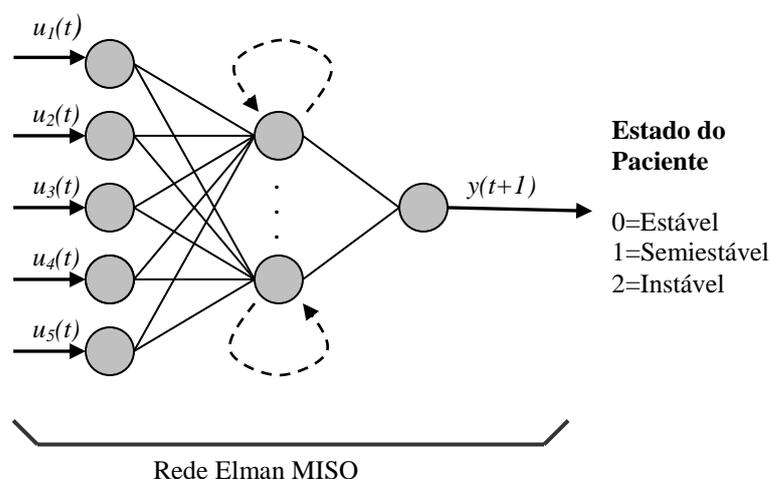


Figura 28 – Rede Elman MISO

3.6.1. Função de Treinamento e Número de Neurônios na Camada Escondida

Para decidir a quantidade de neurônios na camada escondida e o algoritmo de treinamento a utilizar foram feitos vários testes. O critério para avaliar a melhor arquitetura foi diminuir o Erro Quadrático Médio (MSE) e utilizar o menor número de neurônios possíveis na camada escondida. A tabela 9 resume os resultados de treinamento da rede neural Elman MISO variando a quantidade de neurônios ocultos e comparando o tempo decorrido e o MSE obtido, tudo isto com seis algoritmos de treinamento. Os algoritmos de treinamento utilizados foram *Resilient Backpropagation (RB)*, *Levenberg-Marquardt Backpropagation (LMB)*, *BFGS quasi-Newton Backpropagation (BFGS)*, *Gradient Descent with Momentum and Adaptive*

*Learning Backpropagation (GDMALB), Gradient Descent Backpropagation (GDB) e Scaled Conjugate Gradient Backpropagation (SCGB)*¹¹.

Tabela 9 – Diferentes combinações de redes Elman MISO (continua)

Algoritmo	Quantidade de Neurônios	MSE médio	Tempo médio (segundos)
RB	1	0,441	04
RB	2	0,441	04
RB	3	0,441	05
RB	4	0,441	04
RB	5	0,441	04
RB	6	0,441	04
RB	7	0,447	04
RB	8	0,441	05
RB	9	0,441	06
RB	10	0,416	06
RB	11	0,441	07
RB	12	0,488	06
RB	13	0,417	07
RB	14	0,441	07
RB	15	0,441	07
LMB	1	0,356	03
LMB	2	0,419	04
LMB	3	0,443	08
LMB	4	0,444	09
LMB	5	0,700	12
LMB	6	0,395	09
LMB	7	0,814	14
LMB	8	0,476	14
LMB	9	0,633	40
LMB	10	0,455	28
LMB	11	0,397	24
LMB	12	0,582	36
LMB	13	0,613	40
LMB	14	0,818	51
LMB	15	0,351	79
BFGS	1	0,455	01
BFGS	2	0,407	02
BFGS	3	0,522	03
BFGS	4	0,276	02
BFGS	5	0,334	06
BFGS	6	0,382	04
BFGS	7	0,307	05
BFGS	8	0,338	04
BFGS	9	0,535	06
BFGS	10	0,394	06
BFGS	11	0,446	08

RB = Resilient Backpropagation; LMB = Levenberg-Marquardt Backpropagation; BFGS = BFGS quasi-Newton Backpropagation; GDMALB = Gradient Descendt with Momentum and Adaptative Learning Backpropagation; GDB = Gradient Descendt Backpropagation; SCGB = Scaled Conjugate Gradient Backpropagation.

¹¹ Cada um destes algoritmos pode ser consultado no site Mathworks (MATHWORKS PRODUCT DOCUMENTATION, 2014).

Tabela 9 – Diferentes combinações de redes Elman MISO (conclusão)

Algoritmo	Quantidade de Neurônios	MSE médio	Tempo médio (segundos)
BFGS	12	0,404	09
BFGS	13	0,388	09
BFGS	14	0,250	08
BFGS	15	0,342	11
GDMALB	1	0,361	17
GDMALB	2	0,413	17
GDMALB	3	0,384	19
GDMALB	4	0,420	17
GDMALB	5	0,310	17
GDMALB	6	0,368	20
GDMALB	7	0,301	17
GDMALB	8	0,274	17
GDMALB	9	0,278	21
GDMALB	10	0,226	17
GDMALB	11	0,208	18
GDMALB	12	0,221	18
GDMALB	13	0,115	20
GDMALB	14	0,179	22
GDMALB	15	0,183	21
GDB	1	0,442	22
GDB	2	0,442	23
GDB	3	0,443	24
GDB	4	0,443	25
GDB	5	0,443	25
GDB	6	0,442	23
GDB	7	0,443	23
GDB	8	0,443	24
GDB	9	0,442	22
GDB	10	0,443	22
GDB	11	0,442	24
GDB	12	0,442	26
GDB	13	0,445	25
GDB	14	0,443	26
GDB	15	0,444	26
SCGB	1	0,413	6
SCGB	2	0,462	10
SCGB	3	0,340	10
SCGB	4	0,225	14
SCGB	5	0,108	16
SCGB	6	0,184	16
SCGB	7	0,231	16
SCGB	8	0,255	18
SCGB	9	0,167	17
SCGB	10	0,102	17
SCGB	11	0,161	21
SCGB	12	0,182	19
SCGB	13	0,095	18
SCGB	14	0,187	25
SCGB	15	0,122	19

RB = Resilient Backpropagation; LMB = Levenberg-Marquardt Backpropagation; BFGS = BFGS quasi-Newton Backpropagation; GDMALB = Gradient Descend with Momentum and Adaptive Learning Backpropagation; GDB = Gradient Descend Backpropagation; SCGB = Scaled Conjugate Gradient Backpropagation.

Na figura 29 um gráfico comparativo dos valores na tabela 9 é apresentado. Como é possível observar, a rede Elman MISO apresentou melhores rendimentos (menores MSEs) quando for treinada com os algoritmos SCGB e GDMALB. As melhores combinações, aquelas que apresentam os menores valores de MSE, estão assinaladas na tabela 9 em cor cinza.

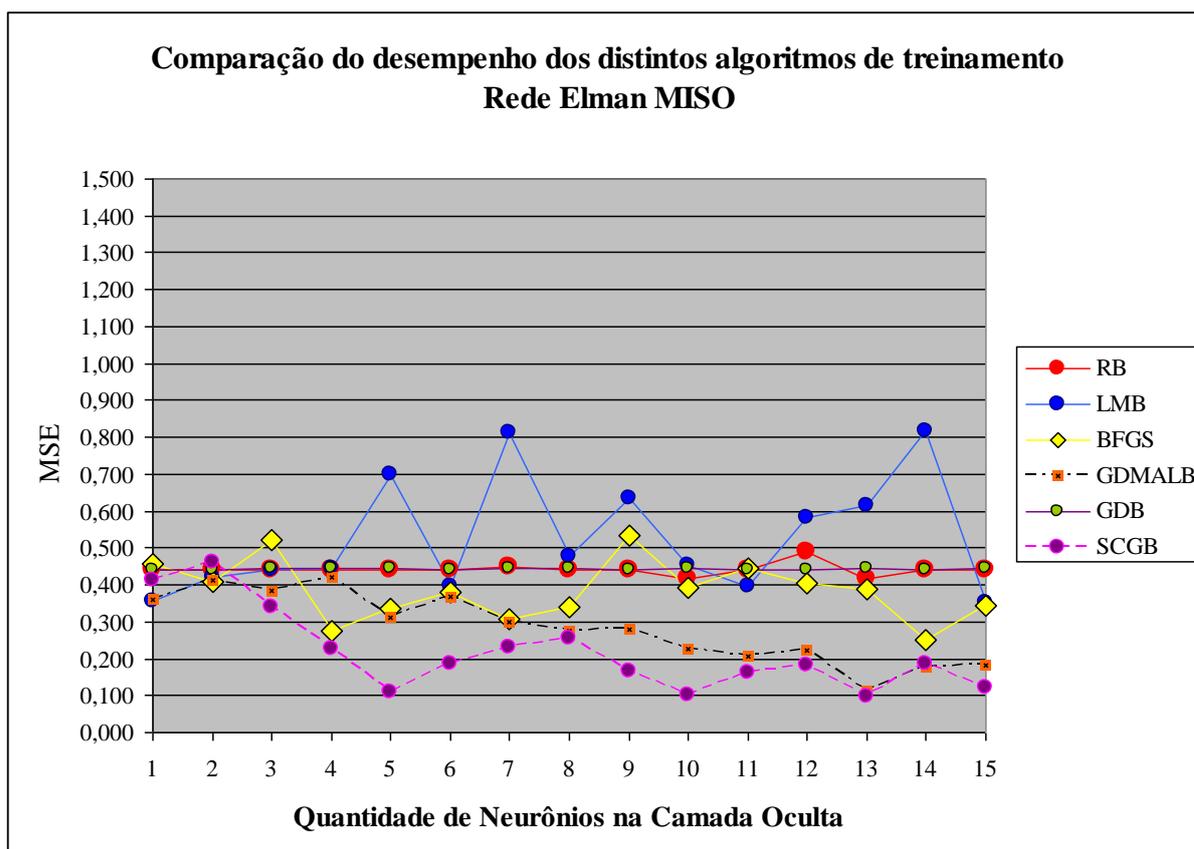


Figura 29 - Desempenho dos distintos algoritmos de treinamento - Rede Elman MISO

Com o objetivo de escolher a melhor arquitetura, aquela que representa melhor o modelo Elman MISO, foram testadas as cinco melhores combinações (menores resultados de MSE). Estas combinações, quatro com o algoritmo SCGB com 5, 10, 13 e 15 neurônios escondidos respectivamente e uma com o algoritmo GDMALB de 13 neurônios escondidos, foram treinadas com a função de treinamento *tansig* nas unidades de entrada e a função de treinamento *purelin* para a unidade de saída. Todas as arquiteturas foram analisadas em forma particular com vários indicadores de desempenho utilizando o conjunto de validação seguindo as equações (20-22, 26,27). Os resultados podem ser observados na tabela 10.

Tabela 10 – Análise das cinco melhores arquiteturas Elman MISO

Algoritmo de treinamento	SCGB	SCGB	SCGB	SCGB	GDMALB
Configuração da rede	(5,5,1)	(5,10,1)	(5,13,1)	(5,15,1)	(5,13,1)
MSE médio	0,108	0,102	0,095	0,122	0,115
MSE particular	0,583	0,928	0,105	0,211	0,531
Coefficiente Correlação	0,503	0,562	0,901	0,844	0,744
Sensibilidade (%)	71,33	58,45	99,27	99,69	99,99
Especificidade (%)	0,79	30,03	0,00	0,00	0,00
Acurácia Geral (%)	64,16	54,21	89,55	93,71	83,24

GDMALB = Gradient Descendt with Momentum and Adaptative Learning Backpropagation; SCGB = Scaled Conjugate Gradient Backpropagation.

Fonte: Autoria própria

As últimas três arquiteturas cujo valor de especificidade é 0,00% serão descartadas, uma vez que elas não conseguem classificar corretamente os estados nos quais o paciente está estável (ausência de doença, ou verdadeiros negativos na matriz de confusão). No que diz respeito às outras duas arquiteturas a topologia SCGB(5,5,1) tem melhor MSE, então será escolhida para modelar a arquitetura Elman MISO, de agora em diante chamada E-MISO.

Os resultados obtidos da rede E-MISO serão analisados no capítulo 4.

Com vistas a melhorar o desempenho da rede E-MISO, o próximo passo será a aplicação da técnica da poda para remover conexões e/ou unidades desnecessárias.

3.7. PODA DA REDE E-MISO – PE-MISO

Nesta seção será realizada a poda da rede E-MISO que possui atualmente 36 pesos ativos e 5 neurônios escondidos. Para realizá-la foi utilizada a técnica *Optimal Brain Surgeon* (OBS). Esta técnica foi escolhida por apresentar melhor desempenho que a *Optimal Brain Damage* (OBD), segundo o visto na subseção 2.4.10.

A figura 30 tem que ser lida da direita para a esquerda. No eixo x estão representados os 36 pesos. A rotina vai eliminando um por um os pesos e testando os erros de treinamento, de teste (erro de generalização) e o FPE (Erro de Predição Final) de cada uma das redes intermédias. Na figura 30 consegue-se observar que o erro de teste não muda em nenhuma das topologias testadas, sendo todas elas equivalentes. Como não faz diferença qual delas utilizar, será escolhida aquela topologia com a menor quantidade de conexões possíveis, que considere a conexão de todas as unidades de entrada e de saída e que não deixe nenhum neurônio da camada escondida conectado a alguma unidade de saída sem conexão com alguma unidade de entrada. Neste caso tal topologia consegue-se com 21 pesos.

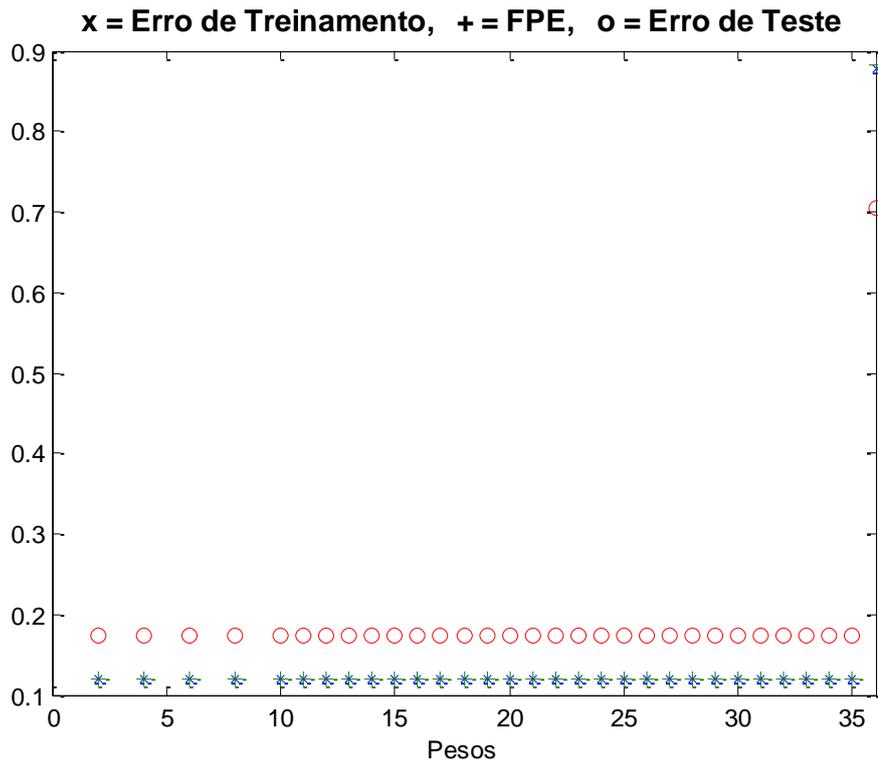


Figura 30 – Resultados da poda da rede E-MISO

A rede Elman MISO(5,5,1) podada, detalhada na figura 31, resulta ter a mesma estrutura original, com as 5 unidades de entrada e os 5 neurônios na camada escondida, mas a diferença está na redução do número de conexões. As sinapses contínuas representam os pesos com valor positivo e as tracejadas tem valor negativo.

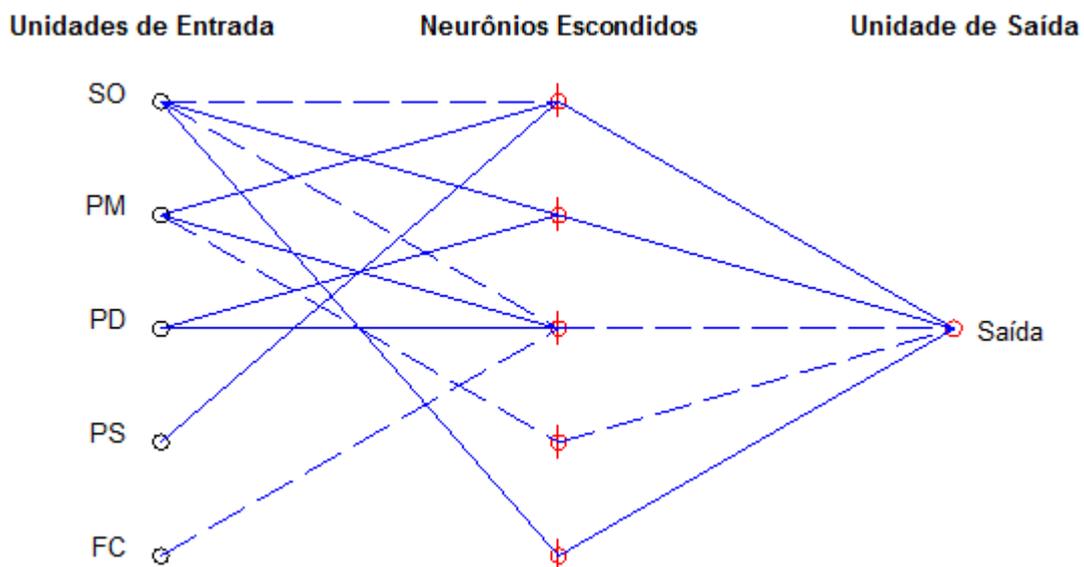


Figura 31 – Topologia da rede PE-MISO

Uma vez achada a arquitetura ótima de rede, ela tem que ser treinada e validada novamente. Neste caso o algoritmo de treinamento a utilizar será o SCGB que foi o escolhido para treinar a rede E-MISO. O critério de parada utilizado para o treinamento é o de parada antecipada, explicado na subseção 2.4.8., e também utilizado no treinamento da rede Elman.

3.8. ARQUITETURA DA REDE RECORRENTE NEURAL ELMAN MIMO

Esta rede neural foi construída com as cinco entradas, correspondentes aos sinais vitais dos pacientes, e cinco saídas que seriam os valores futuros, tempo $(t + d)$, com $d = 1$, dessas entradas. Na tabela 11 é possível observar as entradas e saídas do modelo Elman MIMO.

Tabela 11 – Especificação das entradas e saídas do modelo Elman MIMO

Entradas	Saídas
$u_1(t)$ = frequência cardíaca	$y_1(t+1)$ = frequência cardíaca
$u_2(t)$ = pressão arterial sistólica	$y_2(t+1)$ = pressão arterial sistólica
$u_3(t)$ = pressão arterial diastólica	$y_3(t+1)$ = pressão arterial diastólica
$u_4(t)$ = pressão arterial média	$y_4(t+1)$ = pressão arterial média
$u_5(t)$ = saturação de oxigênio	$y_5(t+1)$ = saturação de oxigênio

A rede Elman MIMO, cujo diagrama esquemático está apresentado na figura 32, utiliza a função de transferência tangente hiperbólica sigmoide para as unidades de entrada, tem uma camada escondida e nas unidades de saída a função de transferência utilizada é a de transferência linear.

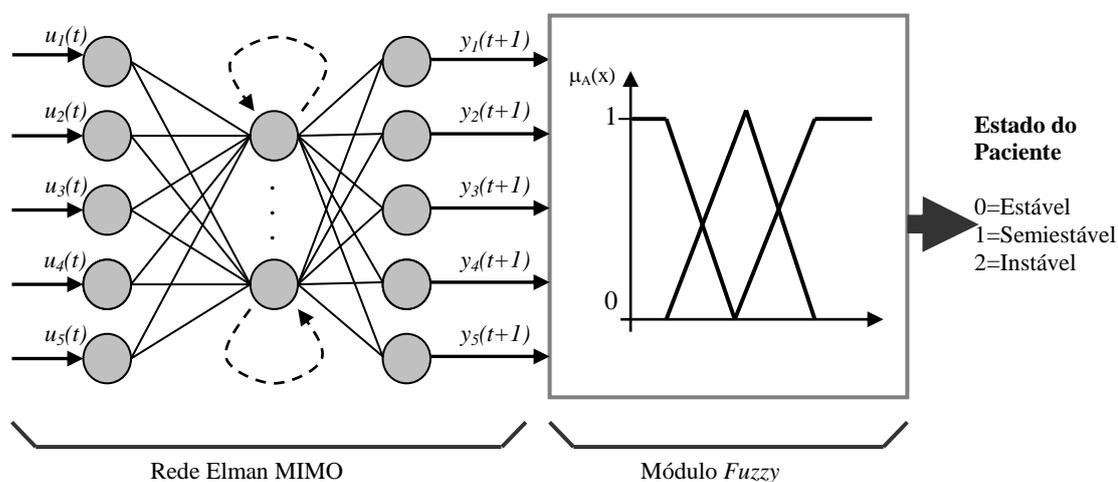


Figura 32 – Rede E-MIMO com inclusão do Modelo Fuzzy para predição da saída final

Como a rede neural Elman tem cinco saídas, mas se precisa unicamente de uma saída (estado estável, semiestável o instável), foi necessário adicionar o modelo *fuzzy* à saída da rede Elman, modelo proposto na seção 3.4. Então, neste caso, a arquitetura ficaria composta pelos módulos “rede Elman MIMO” e “módulo *Fuzzy*”.

3.8.1. Função de Treinamento e Número de Neurônios na Camada Escondida

Neste caso, tal como na arquitetura anterior (rede E-MISO) para definir a quantidade de neurônios na camada escondida e o algoritmo de treinamento a utilizar foram feitos vários testes. O critério de parada utilizado foi o de parada antecipada (ver subseção 2.4.8).

Aqui também o critério para avaliar a melhor combinação foi diminuir o Erro Quadrático Médio (MSE) utilizando o menor número de neurônios possíveis na camada escondida. A tabela 12 resume os resultados de treinamento da rede neural Elman MIMO, utilizando os mesmos algoritmos de treinamento que a rede E-MISO.

Tabela 12 – Diferentes combinações de redes Elman MIMO (continua)

Algoritmo	Quantidade Neurônios	MSE médio	Tempo médio (mm:ss)
RB	1	233,762	00:03
RB	2	234,690	00:03
RB	3	234,690	00:03
RB	4	234,690	00:03
RB	5	234,690	00:03
RB	6	234,690	00:03
RB	7	234,690	00:03
RB	8	234,690	00:03
RB	9	234,678	00:04
RB	10	234,680	00:04
RB	11	234,688	00:04
RB	12	234,670	00:04
RB	13	234,681	00:04
RB	14	234,689	00:04
RB	15	234,690	00:04
RB	16	234,669	00:04
RB	17	234,681	00:05
RB	18	234,657	00:05
GDMALB	1	234,638	00:11
GDMALB	2	233,743	00:12
GDMALB	3	227,830	00:12
GDMALB	4	228,654	00:13
GDMALB	5	237,161	00:14

RB = Resilient Backpropagation; LMB = Levenberg-Marquardt Backpropagation; BFGS = BFGS quasi-Newton Backpropagation; GDMALB = Gradient Descendt with Momentum and Adaptative Learning Backpropagation; GDB = Gradient Descendt Backpropagation; SCGB = Scaled Conjugate Gradient Backpropagation.

Tabela 12 - Diferentes combinações de redes Elman MIMO (continua)

Algoritmo	Quantidade Neurônios	MSE médio	Tempo médio (mm:ss)
GDMALB	6	230,083	00:15
GDMALB	7	241,109	00:16
GDMALB	8	241,162	00:18
GDMALB	9	244,260	00:19
GDMALB	10	224,774	00:27
GDMALB	11	235,951	00:30
GDMALB	12	235,054	00:30
GDMALB	13	234,077	00:30
GDMALB	14	234,266	00:33
GDMALB	15	236,736	00:34
GDMALB	16	233,518	00:33
GDMALB	17	228,705	00:37
GDMALB	18	237,371	00:35
GDB	1	65,443	00:20
GDB	2	118,543	00:20
GDB	3	173,856	00:20
GDB	4	206,851	00:21
GDB	5	222,804	00:21
GDB	6	229,490	00:21
GDB	7	232,208	00:22
GDB	8	234,182	00:24
GDB	9	234,261	00:24
GDB	10	235,221	00:24
GDB	11	235,388	00:26
GDB	12	234,930	00:26
GDB	13	235,307	00:26
GDB	14	235,826	00:28
GDB	15	235,075	00:29
GDB	16	234,801	00:31
GDB	17	235,398	00:33
GDB	18	235,805	00:33
SCGB	1	265,735	00:10
SCGB	2	291,551	00:15
SCGB	3	313,143	00:15
SCGB	4	278,112	00:16
SCGB	5	285,683	00:14
SCGB	6	257,842	00:19
SCGB	7	244,537	00:21
SCGB	8	259,897	00:28
SCGB	9	244,010	00:27
SCGB	10	269,409	00:29
SCGB	11	232,091	00:30
SCGB	12	204,352	00:32
SCGB	13	215,580	00:37
SCGB	14	216,237	00:33
SCGB	15	215,967	00:37
SCGB	16	177,712	00:44
SCGB	17	154,447	00:50
SCGB	18	157,018	00:48

RB = Resilient Backpropagation; LMB = Levenberg-Marquardt Backpropagation; BFGS = BFGS quasi-Newton Backpropagation; GDMALB = Gradient Descendt with Momentum and Adaptative Learning Backpropagation; GDB = Gradient Descendt Backpropagation; SCGB = Scaled Conjugate Gradient Backpropagation.

Tabela 12 - Diferentes combinações de redes Elman MIMO (conclusão)

Algoritmo	Quantidade Neurônios	MSE médio	Tempo médio (mm:ss)
LMB	1	256,897	00:22
LMB	2	405,831	00:25
LMB	3	378,748	00:23
LMB	4	192,962	01:26
LMB	5	573,923	01:48
LMB	6	638,292	01:54
LMB	7	839,387	02:57
LMB	8	298,692	04:56
LMB	9	168,59	05:25
LMB	10	302,177	07:26
LMB	11	458,091	08:06
LMB	12	766,003	11:05
LMB	13	345,726	17:27
LMB	14	197,399	23:07
LMB	15	411,465	23:40
LMB	16	530,162	45:46
LMB	17	346,828	38:12
LMB	18	256,830	36:08
BFGS	1	234,690	00:01
BFGS	2	235,542	00:02
BFGS	3	235,177	00:02
BFGS	4	235,785	00:02
BFGS	5	235,785	00:02
BFGS	6	235,785	00:02
BFGS	7	225,930	00:05
BFGS	8	235,907	00:02
BFGS	9	225,921	00:07
BFGS	10	235,907	00:03
BFGS	11	219,835	00:09
BFGS	12	227,305	00:16
BFGS	13	242,020	00:07
BFGS	14	225,776	00:08
BFGS	15	225,992	00:14
BFGS	16	204,821	00:15
BFGS	17	235,907	00:05
BFGS	18	235,907	00:06

RB = Resilient Backpropagation; LMB = Levenberg-Marquardt Backpropagation; BFGS = BFGS quasi-Newton Backpropagation; GDMALB = Gradient Descendt with Momentum and Adaptative Learning Backpropagation; GDB = Gradient Descendt Backpropagation; SCGB = Scaled Conjugate Gradient Backpropagation.

A tabela 12 apresenta os valores obtidos combinando os distintos algoritmos de treinamento testados com diferentes quantidades de neurônios na camada escondida. Dos algoritmos estudados, quatro deles apresentaram valores de MSE muito semelhantes, em torno de 240. O algoritmo GDB mostrou uma tendência crescente com os primeiros cinco neurônios até se estabilizar num valor próximo a 235. O algoritmo LMB mostrou-se bem instável, ficando num valor por abaixo de 200 em apenas três ocasiões. Já o algoritmo SCGB apresentou uma tendência decrescente, e em apenas três configurações conseguiu ficar abaixo de 200. Na figura 33 um gráfico comparativo destes valores é apresentado.

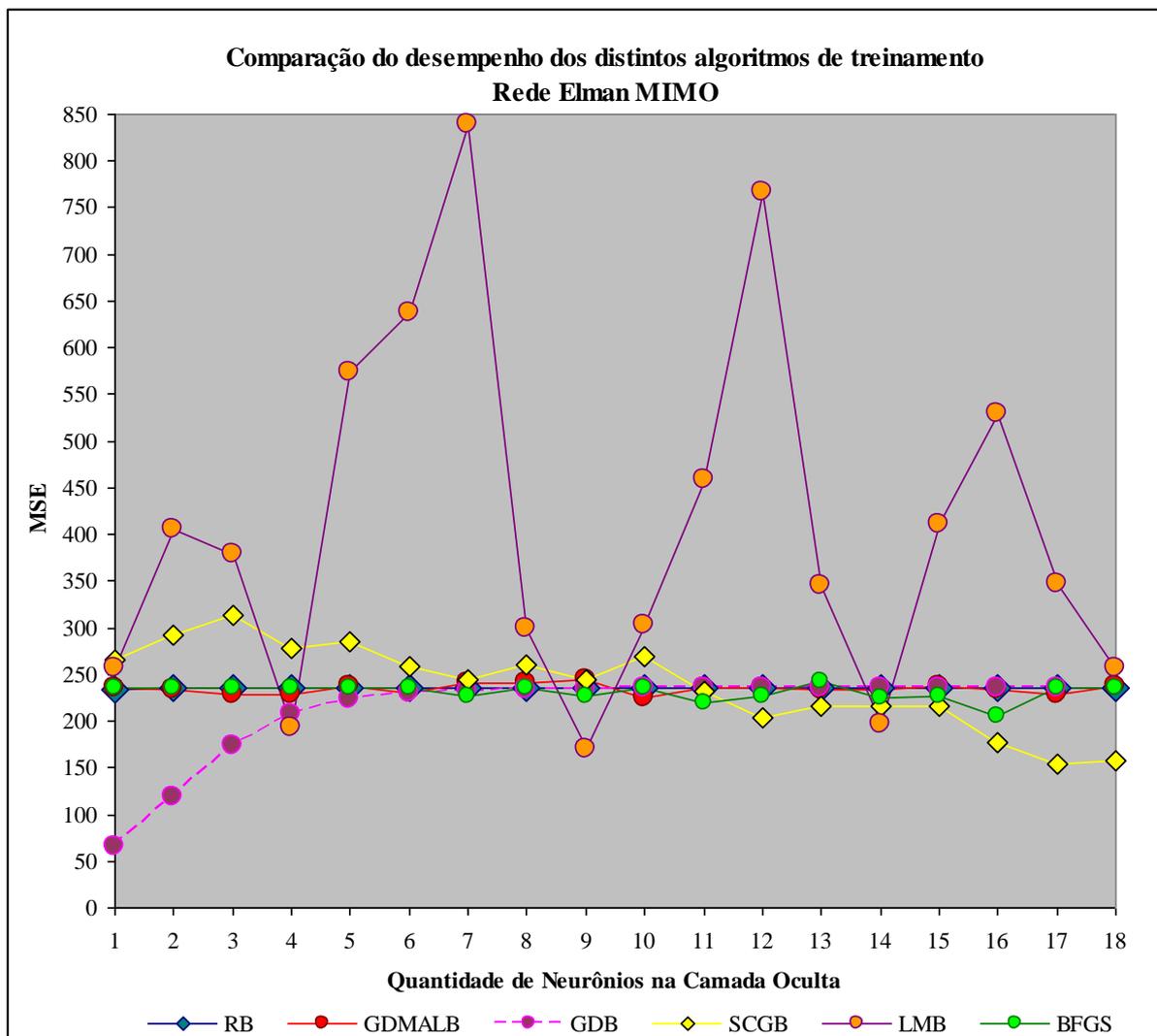


Figura 33 - Desempenho dos distintos algoritmos de treinamento - Rede Elman MIMO

Aquelas cinco combinações, cujos valores de MSE são iguais ou inferiores a 170, estão assinaladas na tabela 12 em cor cinza. Elas foram treinadas com as mesmas funções de treinamento que a rede E-MISO e analisadas com as mesmas equações (20-22,26,27). Os resultados podem ser observados na tabela 13.

Todas as arquiteturas analisadas apresentam a especificidade com um valor de 0,00%, o que indica que nenhuma delas consegue classificar corretamente os estados nos quais o paciente está estável (ausência de doença, ou verdadeiros negativos na matriz). Similarmente, todas elas têm uma sensibilidade de 100%, indicando que conseguem classificar corretamente os estados nos quais o paciente não está estável.

Todas as arquiteturas têm praticamente o mesmo MSE particular e a mesma acurácia geral e em todas elas também o coeficiente de correlação está com um valor perto de zero, indicando uma nula relação linear entre a saída simulada da rede e os *targets*.

Tabela 13 – Análise das cinco melhores arquiteturas Elman MIMO

Algoritmo de treinamento	GDB	GDB	SCGB	SCGB	LMB
Configuração da rede	(5,1,5)	(5,2,5)	(5,17,5)	(5,18,5)	(5,9,5)
MSE médio	65,443	118,543	154,447	157,018	168,59
MSE particular	0,427	0,427	0,427	0,400	0,427
Coefficiente Correlação	-0,0002	0,0002	-0,038	-0,003	0,000
Sensibilidade (%)	100,00	100,00	100,00	100,00	100,00
Especificidade (%)	0,00	0,00	0,00	0,00	0,00
Acurácia Geral (%)	74,99	74,99	74,99	74,99	74,99

LMB = Levenberg-Marquardt Backpropagation; GDB = Gradient Descendt Backpropagation; SCGB = Scaled Conjugate Gradient Backpropagation.

Fonte: Autoria própria

Nenhuma destas arquiteturas é viável por ter o indicador de especificidade em zero e o coeficiente de correlação perto de zero. Estas redes só conseguem prognosticar os estados instáveis dos pacientes, independentemente dos valores dos sinais vitais atuais, e por este motivo não será feita a poda da rede. Finalmente, a topología Elman MIMO é desaconselhada para criar o sistema inteligente de predição do estado clínico do paciente.

3.9. ARQUITETURA REDE NEURAL NNARX

Para obter o modelo foram utilizados os cinco vetores de dados (sinais vitais do paciente) e a saída desejada da rede (os *targets*) foi obtida via pré-processamento dos vetores de entrada utilizando o modelo *fuzzy* proposto na seção 3.4.

Para esta rede foi usada a estratégia tipo NNARX, mostrada na figura 34, e aplicada a ferramenta desenvolvida por Nørgaard et al. (2000). A expressão que corresponde a este modelo está dada pelas equações (28)-(30).

$$\hat{y}(t + 1) = f(\varphi(t)) \quad (28)$$

$$\varphi(t) = [FC(t), \dots, FC(t - e_1), PS(t), \dots, PS(t - e_2), PD(t), \dots, PD(t - e_3), PM(t), \dots, PM(t - e_4), SO(t), \dots, SO(t - e_5), y(t), \dots, y(t - s_1)] \quad (29)$$

$$\varphi(k) = [\varphi_1, \varphi_2, \varphi_3, \varphi_4, \varphi_5, \varphi_6] \quad (30)$$

onde:

$FC(t) \in \mathfrak{R}$	valor presente de frequência cardíaca;
$PS(t) \in \mathfrak{R}$	valor presente de pressão sistólica não invasiva;
$PD(t) \in \mathfrak{R}$	valor presente de pressão diastólica não invasiva;
$PM(t) \in \mathfrak{R}$	valor presente de pressão média não invasiva;
$SO(t) \in \mathfrak{R}$	valor presente de saturação de oxigênio;
e_1, \dots, e_5, s_1	regressores das entradas e da saída do modelo;
$\varphi(t) \in \mathfrak{R}^Z$	vetor de entradas à rede neural no instante t ;
$\hat{y}(t+1) \in \mathfrak{R}$	vetor de predições de saídas a um período de amostragem;
$f : \mathfrak{R}^Z \rightarrow \mathfrak{R}$	mapeamento que representa a função de aproximação da rede neural.
Z	Ordem do modelo – Soma dos regressores nos vetores de entrada

Os dados de treinamento e de validação foram os mesmos já utilizados para as redes Elman, seguindo a metodologia já explicada na seção 3.2.

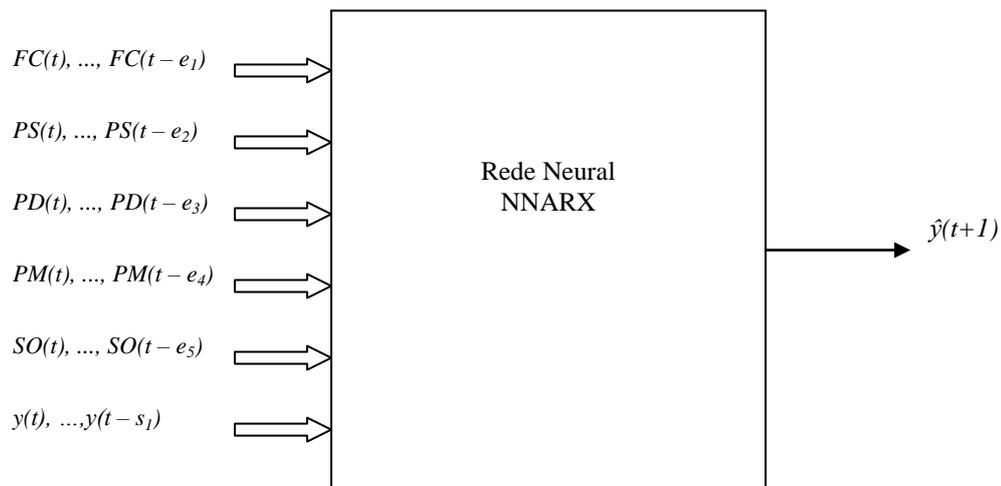


Figura 34 – Estrutura da rede NNARX

3.9.1. Seleção da Estrutura do Modelo Neural

Da família de modelos não lineares autorregressivos já explicados na subseção 2.5.2, foi escolhida a estrutura de rede NNARX devido a sua adaptabilidade com o sistema, já que é o único modelo dentre os avaliados que não tem realimentação do erro. Como os sinais fisiológicos não são necessariamente sinais periódicos não é útil nem desejável a realimentação com os erros das medições anteriores. Em modelos como os utilizados aqui, que têm vários parâmetros de entrada, muitos deles mutuamente independentes, uma

realimentação contendo o erro anterior somaria mais erro ao sistema. Neste caso, é preferível utilizar as saídas reais anteriores do paciente, fornecidas pelo módulo *fuzzy*, em vez de as saídas anteriores previstas pela rede NNARX, com o erro (ruído) não desejado.

O modelo NNARX aqui avaliado recebe cinco parâmetros fisiológicos mais a saída desejada como valores de entrada (os regressores de cada um deles são estudados na seção seguinte). Também é analisada a quantidade ótima de neurônios para a camada escondida da rede.

3.9.2. Seleção da Ordem do Modelo

Antes de treinar o modelo, foi escolhida a ordem do modelo baseando a pesquisa em dois critérios distintos: Cumprimento da descrição mínima de Rissanen (MDL, *Rissanen's Minimum Description Length*) e Critério de informação de Akaike (AIC, *Akaike Information Criterion*). Estes critérios permitem estabelecer o melhor grau do modelo e seus atrasos para cada estrutura de modelo.

Para obter a ordem do modelo foram utilizadas as funções *struc*, *arxstruc* e *selstruc* disponíveis na documentação de Mathworks (MATHWORKS PRODUCT DOCUMENTATION, 2014). A seguir estão detalhados cada um dos comandos:

Struc cria uma matriz de todas as combinações possíveis de ordens do modelo para uma faixa específica de valores *na*, *nb* e *nk*, onde:

- na*: quantidade de saídas anteriores usadas para determinar a predição;
- nb*: quantidade de entradas anteriores usadas para determinar a predição;
- nk*: retardo de tempo.

Arxstruc toma a saída de *struc* e calcula sistematicamente um modelo ARX para cada ordem de modelo. Ao comparar a saída do modelo com a saída medida retorna uma função *loss* para cada modelo, que consiste na soma normalizada dos erros de predição ao quadrado.

Selstruc compara os valores da função *loss* para os diferentes casos e seleciona o melhor.

Os resultados obtidos, disponíveis na tabela 14, estão apresentados em forma de terna (*na*, *nb*, *nk*).

Foram analisados sete casos, com duas até oito entradas e saídas anteriores e os resultados são apresentados na tabela 14.

Vetor de		Configurações						
Entrada	Critério	(1:2,1:2,1)	(1:3,1:3,1)	(1:4,1:4,1)	(1:5,1:5,1)	(1:6,1:6,1)	(1:7,1:7,1)	(1:8,1:8,1)
FC	MDL	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1
	AIC	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	8,1,1
PS	MDL	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1
	AIC	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	8,1,1
PD	MDL	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1
	AIC	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	8,8,1
PM	MDL	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1
	AIC	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	8,1,1
SO	MDL	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1	1,1,1
	AIC	1,1,1	1,3,1	1,4,1	1,5,1	1,5,1	1,5,1	8,5,1

MDL: Comprimento da descrição mínima de Rissanen; AIC: Critério de informação de Akaike.

Cada vetor de entrada foi avaliado utilizando os dois critérios e considerando sete configurações, $(1:na,1:nb,nk)$, onde na são as saídas anteriores, nb as entradas anteriores e nk é o tempo de atraso, que foi tomado com o valor um (1).

Para quase todos os casos testados, os resultados indicam que a melhor ordem do modelo seria, para cada sinal, uma entrada anterior e uma saída anterior.

Cada uma das medições obtidas produziu como resultado uma percentagem da variância da saída que não é explicada pelo modelo. Em todos os casos esse valor está entre 0,00696% e 0,00698. Como a diferença entre estes valores é bastante pequena, para escolher a melhor ordem do modelo optou-se por aquele que apresenta a menor quantidade de entradas e saídas anteriores. Neste caso o critério MDL (Rissanen's Minimum Description Length) foi o escolhido, em que para todos os vetores de entrada as ternas mínimas sugeridas foram (1,1,1).

O critério MDL que apresentou as menores quantidades de entradas e saídas anteriores para cada configuração estudada, está dado pela seguinte equação:

$$V_{MDL} = V \left(1 + \log(N) \frac{d}{N} \right) \quad (31)$$

onde V é a função *loss*, d é o número de parâmetros do modelo e N é o número de pontos de dados usados para a estimação. Aqui V_{MDL} recebe a estrutura que minimiza o critério MDL.

Os valores obtidos e apresentados na tabela 14 dão como resultado os regressores da equação (29), onde:

$\varphi_1 = e_1 = 1$, $\varphi_2 = e_2 = 1$, $\varphi_3 = e_3 = 1$, $\varphi_4 = e_4 = 1$, $\varphi_5 = e_5 = 1$ e $\varphi_6 = e_6 = 1$. Isto significa que z (ordem do modelo) é igual a 6 ($z = 1+1+1+1+1+1$).

3.9.3. Camada Escondida e Funções de Ativação

Uma vez escolhidos os valores dos regressores, foram estudados dezoito casos de configurações NNARX, variando nelas o número de neurônios na camada escondida, a fim de achar a melhor configuração. Os resultados analisados são o erro médio quadrático, o coeficiente de correlação e o erro de predição final. A tabela 15 apresenta os resultados obtidos.

Tabela 15 – Resultados obtidos na busca do melhor modelo NNARX

Quantidade de neurônios na camada escondida	MSE médio	FPE médio	Coefficiente de correlação médio	Tempo (segundos)
1	0,000	0,014	0,965	2
2	0,005	0,026	0,942	3
3	0,007	0,033	0,941	4
4	0,041	0,032	0,941	5
5	0,226	0,106	0,820	9
6	1,003	0,049	0,914	13
7	1,202	0,042	0,920	17
8	0,488	0,125	0,849	21
9	1,005	0,056	0,903	21
10	1,135	0,050	0,908	24
11	1,374	0,119	0,791	32
12	0,743	0,075	0,865	35
13	0,347	0,096	0,806	43
14	0,700	0,076	0,841	40
15	0,402	0,079	0,852	43
16	0,489	0,090	0,835	37
17	0,556	0,102	0,764	49
18	0,295	0,141	0,820	51

MSE: Erro Quadrático Médio; FPE: Erro de Predição Final.

O critério de parada utilizado foi o de parada antecipada (ver subseção 2.4.8), seguindo a metodologia explicada na seção 3.2. As funções de ativação não linear para os neurônios escondidos são as do tipo tangente hiperbólica $f(x) = \tanh(x)$ e para o neurônio de saída foi selecionada a função de ativação linear $f(x) = x$.

Na figura 35, um gráfico comparativo apresenta os valores de MSE, de FPE e de coeficientes de correlação (CC) para cada modelo de rede NNARX. Para achar as melhores combinações há que se escolher os valores com o menor MSE e FPE e com o maior coeficiente de correlação.

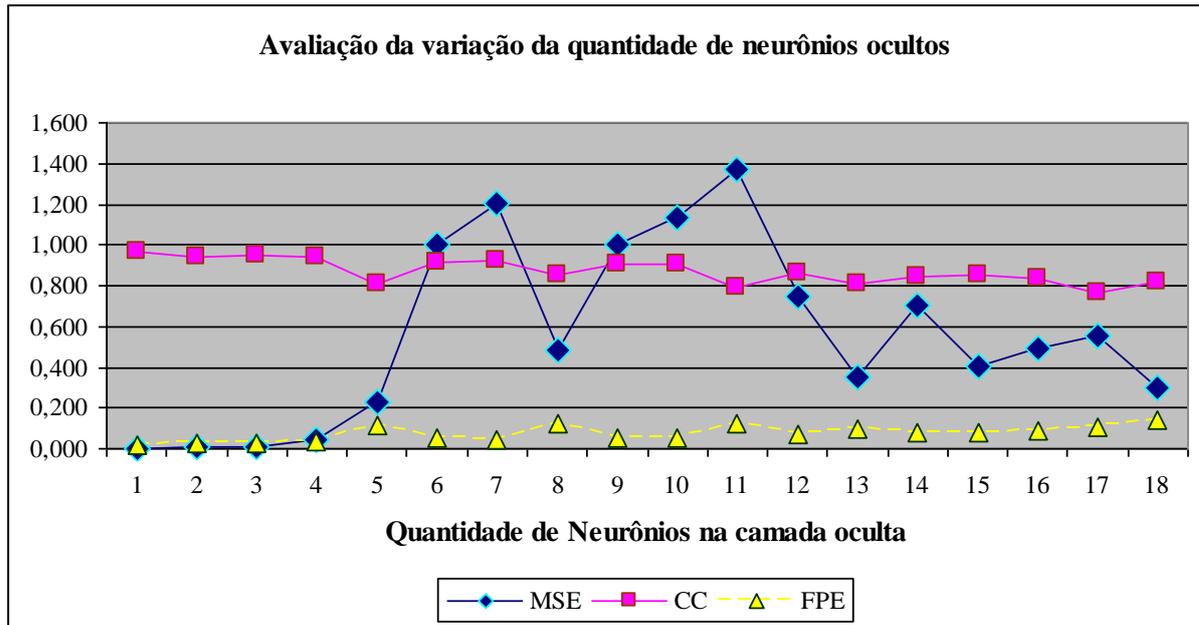


Figura 35 - Análise da quantidade de neurônios ocultos para a rede NNARX
Fonte: Autoria própria.

Seguindo a metodologia utilizada nas redes anteriores, se escolheram os seis melhores casos para avaliar seus valores de sensibilidade, especificidade e acurácia geral. Os modelos NNARX com 1, 2, 3, 4, 5 e 18 neurônios ocultos serão avaliados particularmente utilizando-se o conjunto de validação seguindo as equações (20-22,26,27). Os resultados podem ser observados na tabela 16.

Tabela 16 – Análise das seis melhores arquiteturas NNARX – Dados de Validação

Neurônios ocultos	1	2	3	4	5	18
MSE particular	0,047	0,038	0,115	0,037	0,054	0,075
CC particular	0,965	0,970	0,945	0,975	0,951	0,917
FPE particular	0,014	0,015	0,041	0,012	0,024	0,040
Sensibilidade (%)	97,51	96,37	98,57	96,43	94,58	96,32
Especificidade (%)	78,05	96,95	98,61	96,95	96,95	92,01
Acurácia Geral (%)	95,45	96,42	98,57	96,47	94,78	95,95

MSE = Mean Square Error (Erro Quadrático Médio); CC = Coeficiente de Correlação; FPE = Final Prediction Error (Erro de Predição Final).

Fonte: Autoria própria

Conforme a tabela 16, a topologia NNARX com três neurônios ocultos apresenta os melhores resultados. O coeficiente de correlação perto de um indica uma correlação muito boa entre a saída predita da rede e a saída real (os *targets*). Por outro lado, os indicadores de sensibilidade, especificidade e acurácia geral acima de 98%, indicam que esta é a melhor topologia NNARX e por isso é a escolhida para desenvolver a rede NNARX.

3.9.4. Estimação do Modelo

A rede NNARX foi treinada utilizando o algoritmo de Levenberg-Marquardt, aplicando a ferramenta desenvolvida por Nørgaard et al. (2000) com um máximo de 200 iterações em cada passo de treinamento.

A rede contém 6 entradas, 3 neurônios na camada escondida e uma saída (ver figura 36). Neste ponto os pesos W_1 se W_2 são calculados:

- W_1 , contém os pesos das 6 entradas mais 3 de viés dos 3 neurônios da camada escondida, o que dá um total de 21 pesos.
- W_2 , contém os pesos dos 3 neurônios ocultos mais 1 de viés da única saída, o que dá um total de 4 pesos.

Portanto, $\theta = [W_2 \ W_1]$ contém 25 pesos.

A figura 36 apresenta a topologia da rede neural NNARX(6,3,1). Nela os pesos positivos são representados em linhas sólidas, enquanto uma linha pontilhada representa um peso negativo. A linha vertical através dos neurônios representa o viés.

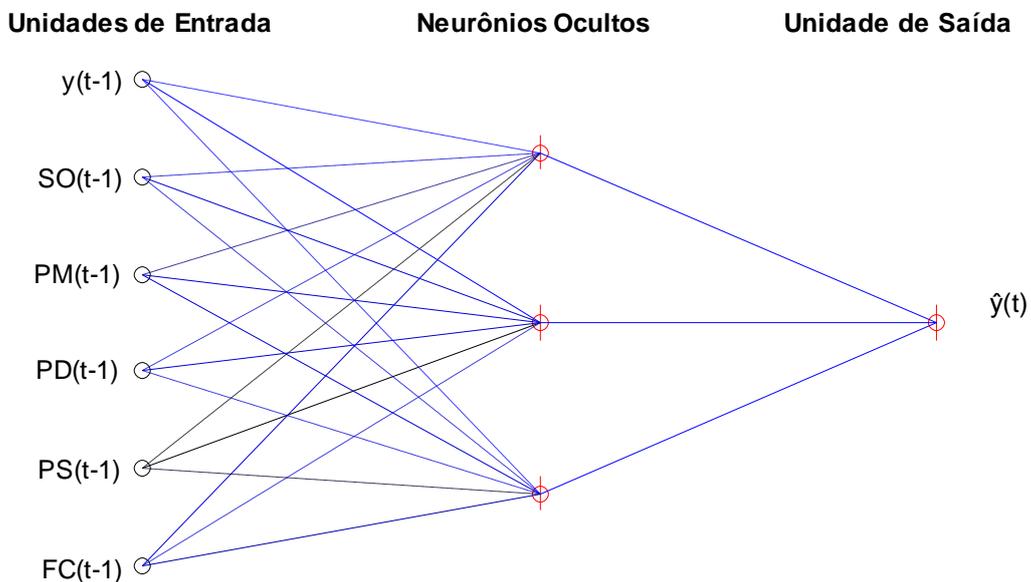


Figura 36 – Topologia da rede neural NNARX(6,3,1) totalmente conectada
Fonte: Autoria própria.

3.10. ARQUITETURA REDE NEURAL NNARX PODADA - PNNARX

Para realizar a poda da rede NNARX foi utilizada a função *nnprune*¹² que, aplicando a estratégia de *Optimal Brain Surgeon* (OBS), determina a estrutura ótima da rede neural de um sistema dinâmico.

A figura 37, produto da poda da rede, tem que ser lida da direita para a esquerda. Na escala do eixo *x* estão representados os 25 pesos. A rotina vai eliminando um por um os pesos e testando os erros de treinamento, de teste e o FPE de cada uma das redes intermédias. Na figura é possível observar que o menor erro de teste pode ser encontrado com uma topologia com 23 pesos, ponto que pode ser comprovado também com os comandos *min* e *index* do Matlab.

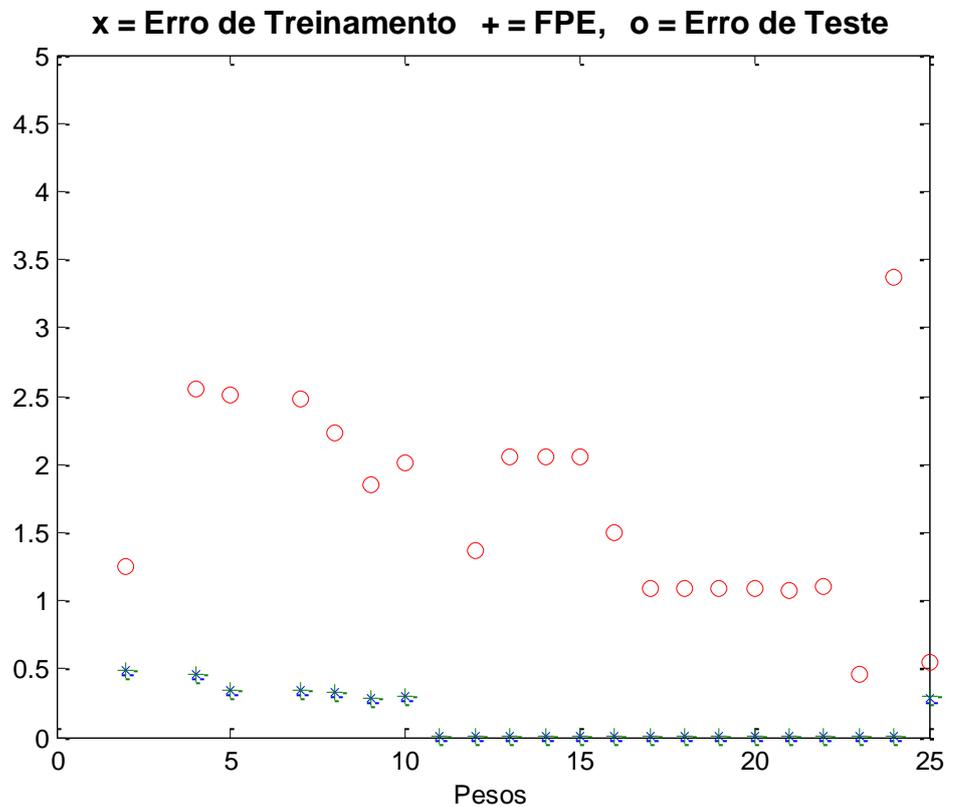


Figura 37 – Resultados da poda da rede NNARX

A rede otimizada, indicada na figura 38, resulta ter a mesma estrutura que a original com 6 entradas ativadas e 3 neurônios na camada escondida, mas com dois pesos a menos. Neste caso, a poda não consistiu na eliminação de unidades, mas sim na mudança e eliminação de pesos. Esta nova topologia corresponde à estrutura ótima do sistema modelado NNARX.

¹² Escrita por Magnus Nørgaard da Technical University of Denmark.

Normalmente, as redes podadas oferecem vantagens computacionais se comparadas às redes totalmente conectadas, oferecendo resultados iguais ou melhores com uma mínima quantidade de conexões, e conseqüentemente menor tempo de resposta da rede.

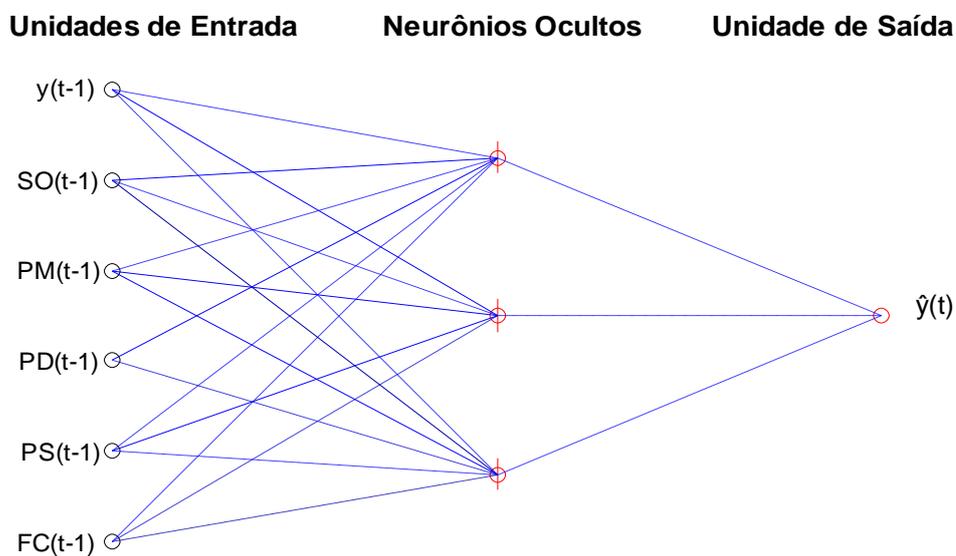


Figura 38 – Topologia da rede PNNARX

Uma vez que a arquitetura de rede PNNARX ótima é achada, ela tem que ser novamente treinada e validada. Para o treinamento o critério de parada utilizado é o de parada antecipada, explicado na subsecção 2.4.8.

DISCUSSÕES

Neste capítulo são analisados o modelo *fuzzy* e as diferentes arquiteturas consideradas no capítulo 3. Elas são avaliadas utilizando os distintos indicadores obtidos na fase de validação das redes assim como aqueles resultantes da análise da matriz de confusão gerada em cada caso.

4.1. VALIDAÇÃO DO MODELO *FUZZY*

A validação do modelo *fuzzy* foi realizada utilizando três conjuntos de dados:

- dez registros de dados de pacientes utilizados para o treinamento, cada um com 2.000 amostras;
- dez registros de dados de pacientes utilizados para a validação, cada um com 1.000 amostras;
- trinta registros de dados de pacientes utilizados na pós-validação ou teste, cada um com 1.500 amostras. Este conjunto será utilizado para avaliar a implementação do sistema inteligente preditivo FNNARX na seção 4.7.

Os cinquenta registros de dados utilizados correspondem a cinquenta pacientes diferentes, conformando um total de 75.000 amostras. Desse lote foram selecionadas aquelas amostras cujos dados fisiológicos não se repetem, obtendo 4.513 medições totalmente diferentes. Analisando os dados e comparando os resultados gerados pelo modelo *fuzzy* com os resultados dados pelos especialistas (segundo a tabela 6) foi obtida uma sensibilidade de 99,75%, uma especificidade de 100,00% e uma acurácia de 99,76%. Um 0,24% (11 medições) corresponde aos casos em que as respostas geradas pelo modelo *fuzzy* não foram as mesmas que as obtidas seguindo o raciocínio dos especialistas, utilizando os valores estabelecidos na tabela 6. Esta discrepância ocorre somente nos casos em que o modelo *fuzzy* gera o estado semiestável quando os especialistas diagnosticariam um estado instável. Esses casos são estudados na tabela 17.

A figura 39 apresenta um gráfico circular mostrando as coincidências e divergências entre o modelo *fuzzy* e as respostas geradas seguindo o raciocínio dos especialistas.

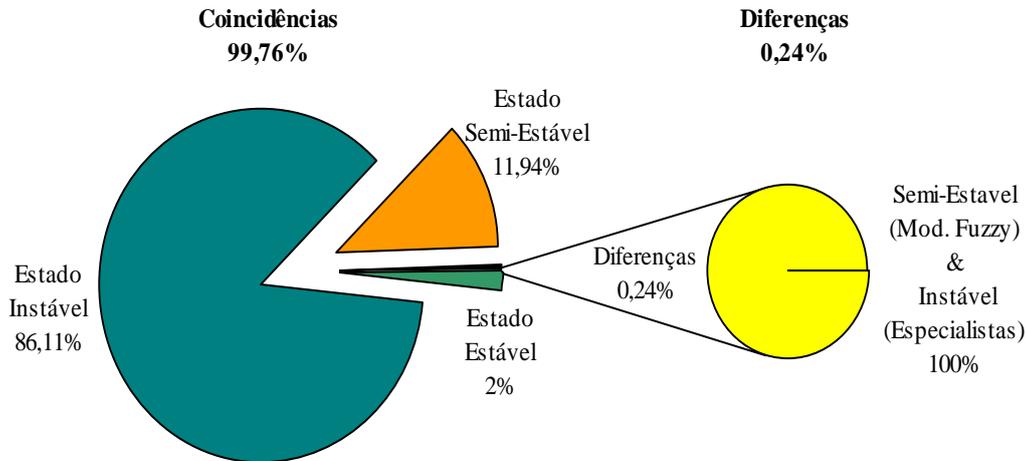


Figura 39 – Coincidências e divergências entre resultados do modelo *fuzzy* e respostas esperadas pelos especialistas

Como é possível perceber na tabela 17, os casos de divergência na emissão dos alarmes são resultantes da característica vaga ou imprecisa do modelo *fuzzy*. Analisando os casos, é possível ver que nos 11 casos a frequência cardíaca (FC) está num valor “quase normal” (um valor normal de FC é de 60-100 bpm, segundo a tabela 6 estabelecida pelos especialistas), mas o valor da pressão diastólica não invasiva (PD) está baixo. É por isso que o modelo *fuzzy* proposto só atende ao valor da PD, que está evidentemente baixo, e considera a FC dentro da faixa normal. Desta forma o sistema emite um alarme médio indicando que o paciente está semiestável, e segundo a tabela 6 desenvolvida pelos especialistas o paciente estaria instável.

As ferramentas *fuzzy* têm demonstrado ser muito valiosas para a área médica, devido à sua capacidade de trabalho com informações imprecisas e incompletas. Esta pesquisa, em particular, demonstrou haver certas diferenças com os resultados previstos pelos especialistas, em especial nas regiões de fronteira entre as diferentes funções de pertinência. Isto não representa um problema, e sim consiste em um exemplo da principal diferença entre a lógica clássica e a lógica *fuzzy*. Por exemplo, num sistema de lógica clássica (bivalente) o alarme se ativa quando a FC é superior a 100 bpm. Se a FC do paciente flutua constantemente entre 99 e 100 o alarme oscilaria o tempo todo nos estados ligado e desligado. No entanto, com a lógica *fuzzy*, o sistema reconhece o valor de 100 de FC como parte da fronteira entre as funções de pertinência “normal” e “alta”, e considera que o valor 100 bpm de FC tem maior grau de pertinência para a função “normal” que para a função “alta”.

Tabela 17 – Comparação de resultados do modelo *fuzzy* com a opinião dos especialistas

Nro.	Modelo						Especialista
	FC (bpm)	PSNI (mmHg)	PDNI (mmHg)	PMNI (mmHg)	SpO2 (%)	<i>Fuzzy</i>	
1	101	108	68	74	99	Semiestável	Instável
2	101	108	68	74	100	Semiestável	Instável
3	101	102	66	75	100	Semiestável	Instável
4	100	102	66	75	100	Semiestável	Instável
5	100	102	66	75	99	Semiestável	Instável
6	101	102	66	75	99	Semiestável	Instável
7	101	103	65	71	100	Semiestável	Instável
8	100	103	65	71	100	Semiestável	Instável
9	100	103	65	71	99	Semiestável	Instável
10	101	103	65	71	99	Semiestável	Instável
11	100	103	65	71	98	Semiestável	Instável

Os dados utilizados para desenvolver o modelo *fuzzy* proposto correspondem a pacientes reais anônimos e foram obtidos do site Physionet.org. Todas as fases de estudo e planejamento do modelo *fuzzy* (definição dos conjuntos *fuzzy* e das regras e a validação dos resultados) foram acompanhadas e supervisionadas por especialistas na área médica.

4.2. ESCOLHA DA REDE E A SUA TOPOLOGIA

Os resultados mostram que a eficiência do algoritmo de treinamento e o número de neurônios na camada escondida são os fatores mais importantes a serem analisados para a escolha da arquitetura da rede.

Tabela 18 – Arquiteturas ótimas para cada modelo de rede – Dados de validação

Rede	Entradas	Neurônios na camada escondida	Saídas	Algoritmo de treinamento	MSE	Coefficiente Correlação
E-MISO	5	5	1	<i>Scaled Conjugate Gradient Backpropagation</i>	0,583	0,503
PE-MISO	5	5	1	<i>Scaled Conjugate Gradient Backpropagation</i>	0,519	0,360
NNARX	6	3	1	<i>Levenberg-Marquardt</i>	0,115	0,945
PNNARX	6	3	1	<i>Levenberg-Marquardt</i>	0,002	0,994

Fonte: Autoria própria

Após sucessivas provas das diferentes arquiteturas analisadas, testando algoritmos de treinamento e variando a quantidade de neurônios na camada escondida, procurando a combinação com menor MSE e maior coeficiente de correlação, foram obtidas várias topologias, cada uma delas representa a melhor topologia para cada tipo de rede. Estes resultados são apresentados na tabela 18. Na sequência, cada uma das topologias testadas serão discutidas. A topologia Elman MIMO não será considerada, devido a que em todos os casos testados a sua especificidade e coeficiente de correlação foram de 0%.

4.3. RESULTADOS DO MODELO ELMAN MISO(5,5,1) – E-MISO

A tabela 19 apresenta a matriz de confusão dos resultados obtidos pela rede E-MISO com os dados de validação. Nela foram registrados os resultados das simulações dos dez pacientes analisados.

Segundo os resultados acima apresentados é possível observar uma taxa de acertos da rede muito pobre com uma exatidão de 37,00% para os três estados possíveis. Dos erros, 41,44% foram pessimistas (saída real melhor que a prevista) e 21,56% foram otimistas (saída real pior que a prevista). Os momentos estáveis dos pacientes tiveram 1,61% das predições corretas; os momentos semiestáveis foram corretamente preditos em 32,68%; e o caso dos momentos instáveis as predições corretas chegaram a 76,71%. Ou seja, segundo a classificação estabelecida na tabela 3, a rede tem um comportamento muito pobre para predizer os estados estáveis e semiestáveis do paciente, mas um comportamento bom para os momentos em que o paciente está num estado instável. Esta topologia é desaconselhada para a construção do sistema de monitoramento de pacientes por ter uma porcentagem de acertos muito pobre.

Tabela 19 – Matriz de confusão da rede E-MISO

		Realidade			Total Classificados
		Estável	Semiestável	Instável	
Classificador	Estável	8	828	79	915
	Semiestável	490	654	1.668	2.812
	Instável	0	519	5.754	6.273
	Total Reais	498	2.001	7.501	10.000
Predições Corretas (%)		1,61	32,68	76,71	37,00
Erros otimistas (%)		0,00	41,38	23,29	21,56
Erros pessimistas (%)		98,39	25,94	0,00	41,44

As comparações entre as saídas simuladas pela rede E-MISO e os *targets* fornecidos, tanto dos dados de treinamento como os de validação são mostradas na figura 40. Nela pode-se observar alguns desvios e problemas na classificação dos três estados. O eixo x indica o número de amostras usadas no modelo e o eixo y representa as diferentes condições nas quais um paciente pode estar: estável, semiestável ou instável.

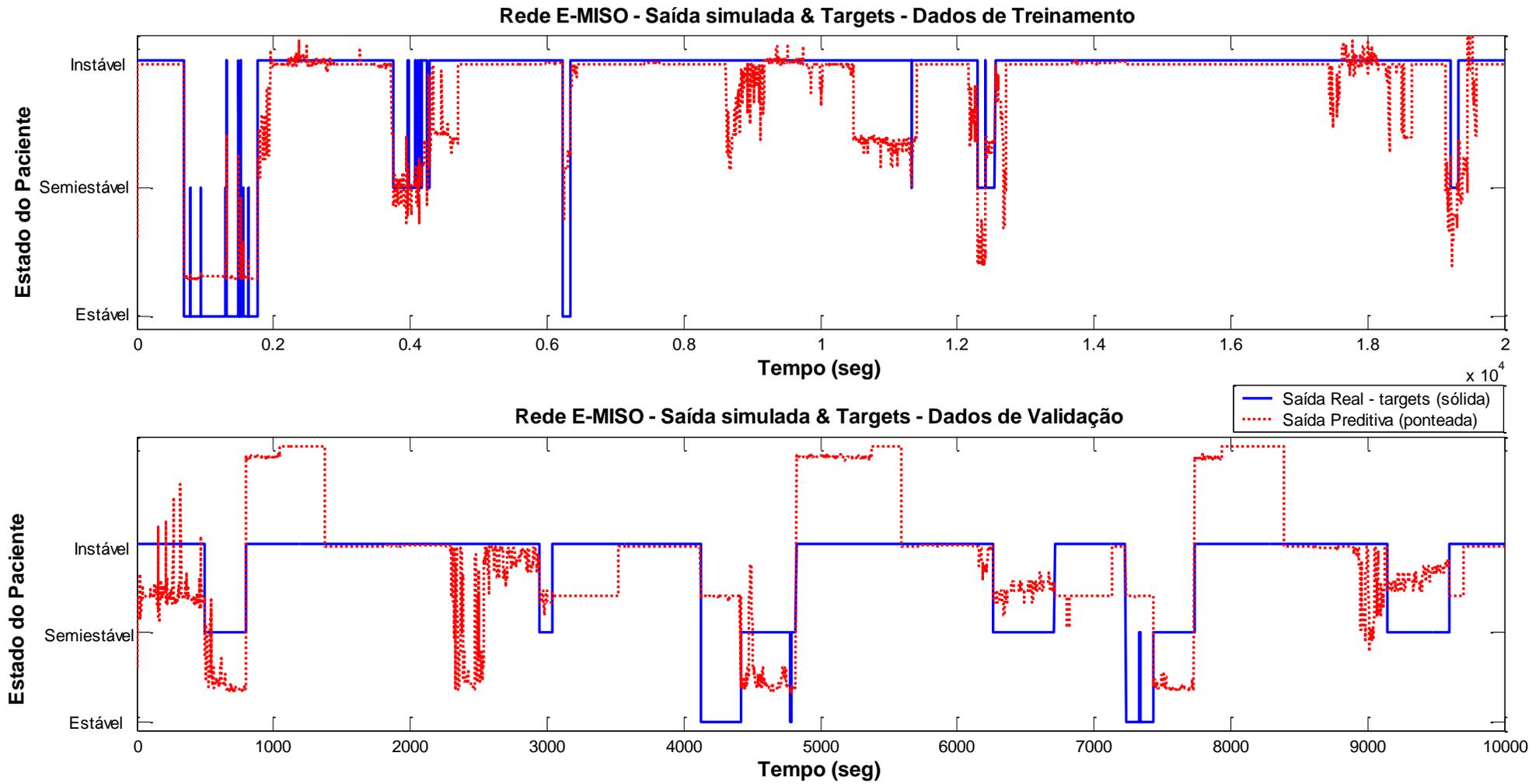


Figura 40 – Saída do modelo E-MISO - Dados de treinamento e de validação
 Comparação entre a saída real (*targets* obtidos via modelo *fuzzy*) e a saída simulada.

4.4. RESULTADOS DO MODELO ELMAN MISO(5,5,1) PODADO – PE-MISO

A tabela 20 apresenta a matriz de confusão dos resultados obtidos pela rede PE-MISO, utilizando os dados de validação. Nela foram registrados os resultados das simulações dos dez pacientes analisados.

Tabela 20 – Matriz de confusão da rede PE-MISO

		Realidade			Total Classificados
		Estável	Semiestável	Instável	
Classificador	Estável	0	0	1	1
	Semiestável	489	3	912	1.404
	Instável	9	1.998	6.588	8.595
	Total Reais	498	2.001	7.501	10.000
Predições Corretas (%)		0,00	0,15	87,83	29,33
Erros otimistas (%)		0,00	0,00	12,17	4,06
Erros pessimistas (%)		100,00	99,85	0,00	66,61

Fonte: Autoria própria

Esta topologia de rede não apresenta bons resultados, o que é devido, em principio, ao fato de que a percentagem de predições corretas foi de apenas 29,33%. Portanto, os erros de predição são elevados, sendo 66,61% os erros pessimistas (saída real melhor que a prevista) e 4,06% os otimistas (saída real pior que a prevista). Os momentos estáveis dos pacientes tiveram 0,00% das predições corretas; os momentos semiestáveis foram corretamente preditos em 0,15%; e o caso dos momentos instáveis as predições corretas chegaram a 87,83%. Ou seja, segundo a classificação estabelecida na tabela 3, a rede tem um comportamento muito bom para predizer os estados instáveis do paciente, mas um comportamento ruim para os momentos em que o paciente está num estado estável ou semiestável. Como a rede tem uma percentagem de acertos muito ruim, esta topologia é desaconselhada para a construção do sistema de monitoramento de pacientes.

A rede PE-MISO é submetida às mesmas equações (20-22,26,27) já utilizadas para a rede sem poda para medir o seu desempenho. Os resultados são apresentados na tabela 21.

A interpretação dos resultados obtidos pelas redes E-MISO e a rede PE-MISO foi feita da seguinte maneira: para analisar o grau de relação linear existente entre a saída simulada da rede e os *targets* foi calculado o coeficiente de correlação linear com os dados do conjunto de validação. O coeficiente de correlação medido em E-MISO e em PE-MISO foi de 0,503 e de

0,360 indicando uma moderada relação linear entre os *targets* e a saída simulada em ambas as redes.

Tabela 21 – Análise das redes E-MISO e PE-MISO

Topologia da rede	E-MISO	PE-MISO
MSE	0,583	0,519
Coefficiente Correlação	0,503	0,360
Sensibilidade (%)	71,33	87,83
Especificidade (%)	0,79	0,00
Acurácia Geral (%)	64,16	65,91

Fonte: Autoria própria

A sensibilidade em E-MISO, de 71,33%, é considerada boa, indicando a capacidade do estimador de classificar corretamente os estados semiestáveis e instáveis do paciente (presença de doença ou verdadeiros positivos na matriz de confusão). Mas em PE-MISO, superior a 80%, é considerada muito boa segundo a tabela 3.

A especificidade da rede E-MISO é de 0,79% e na rede PE-MISO de 0,00%. Em ambas as redes o estimador está impossibilitado para classificar corretamente os estados estáveis do paciente. Isto já indica que as duas redes são modelos não viáveis para representar a construção do sistema de monitoramento de pacientes.

Finalmente, o desempenho geral das redes, representado pela acurácia geral, é também pobre, já que em E-MISO é de 64,16% e em PE-MISO chega a 65,91%.

Todas as qualificações foram feitas considerando o estabelecido na tabela 3 – Guia para classificar a exatidão dos métodos avaliados, disponível na *seção 2.6.*

As comparações entre as saídas simuladas pela rede PE-MISO e os *targets* fornecidos, tanto dos dados de treinamento como os de validação são mostradas na figura 41. Nela, a curva gerada com os dados de validação apresenta problemas para classificar corretamente os momentos estáveis e semiestáveis do paciente.

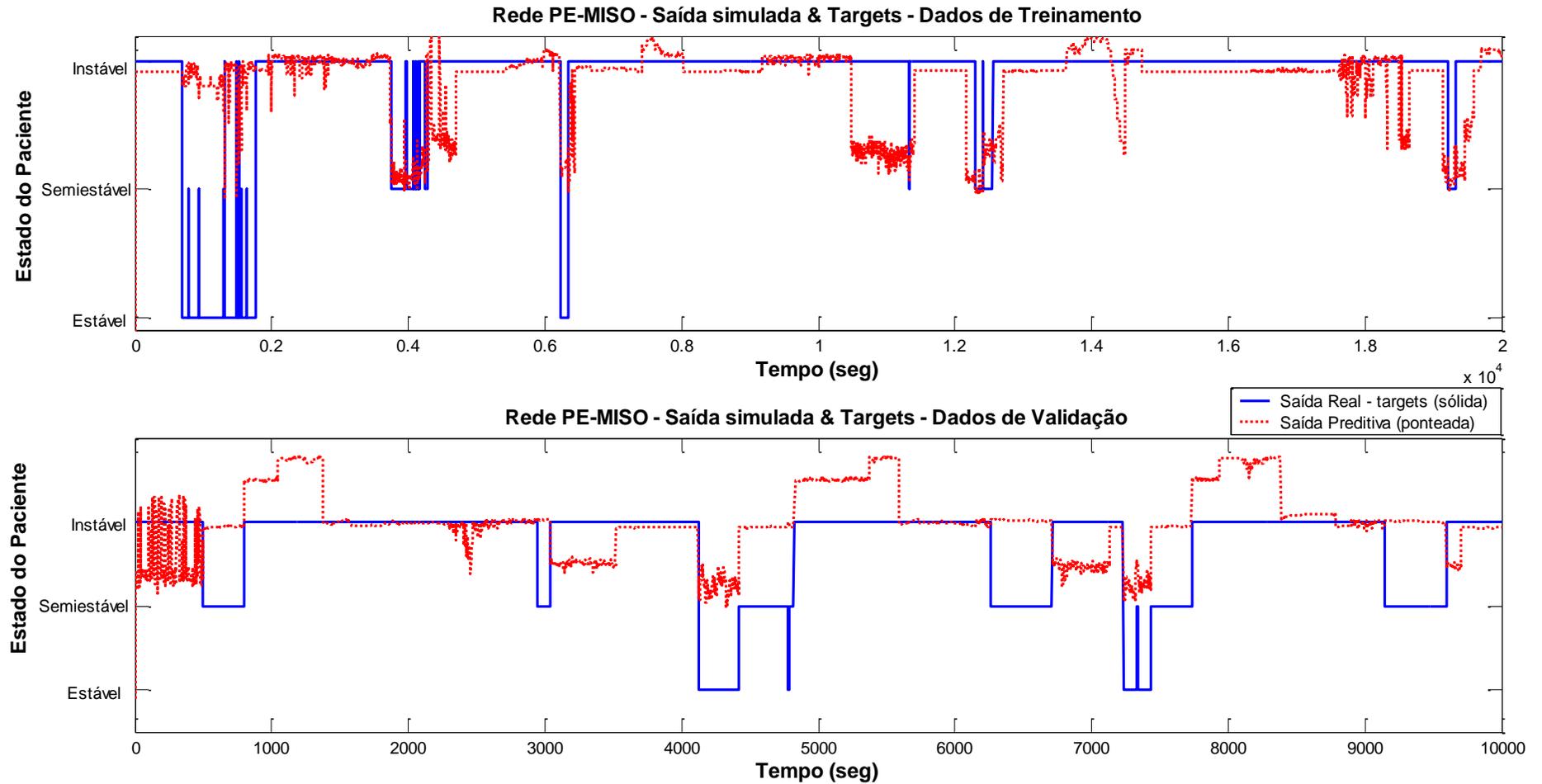


Figura 41 – Saída da rede PE-MISO - Dados de treinamento e validação. Comparação entre a saída real (*targets* obtidos via modelo *fuzzy*) e a saída simulada.

4.5. RESULTADOS DO MODELO NNARX(6,3,1) - NNARX

A tabela 25 apresenta a matriz de confusão dos resultados obtidos pela rede NNARX(6,3,1).

Tabela 25 – Matriz de confusão do modelo NNARX

		Realidade			Total Classificados
		Estável	Semiestável	Instável	
Classificado r	Estável	495	130	0	625
	Semiestável	1	1.867	6	1.874
	Instável	2	4	7.494	7.500
	Total Reais	498	2.001	7.500	9.999
Predições Corretas (%)		99,40	93,30	99,92	97,54
Erros otimistas (%)		0,00	6,50	0,08	2,19
Erros pessimistas (%)		0,60	0,20	0,00	0,27

Fonte: Autoria própria

Os resultados acima detalhados demonstram que a rede NNARX tem uma grande taxa de acerto com uma exatidão de 97,54% para os três estados. Dos erros, 0,27% correspondem a resultados errados pessimistas (saída real melhor que a prevista) e 2,19% a resultados errados otimistas (saída real pior que a prevista). No caso dos pacientes estáveis, 99,40% das predições foram corretas; os pacientes semiestáveis tiveram um 93,30% de predições corretas; e para o caso dos pacientes instáveis as predições corretas chegaram a 99,92%. Ou seja, segundo a classificação estabelecida na tabela 3, a rede tem um excelente comportamento para os três estados estabelecidos. Como o comportamento geral da rede também é excelente (97,54%) esta topologia pode ser considerada para a construção do sistema de monitoramento de pacientes.

A figura 42 apresenta as curvas obtidas com os dados de treinamento e validação. Nelas pode-se observar que a saída simulada com os dados de treinamento é muito boa e similar aos *targets* fornecidos pelo modelo *fuzzy* (simulação do raciocínio dos especialistas). Com os dados de validação a saída simulada mantém um desempenho muito bom, ficando bastante perto dos *targets*.

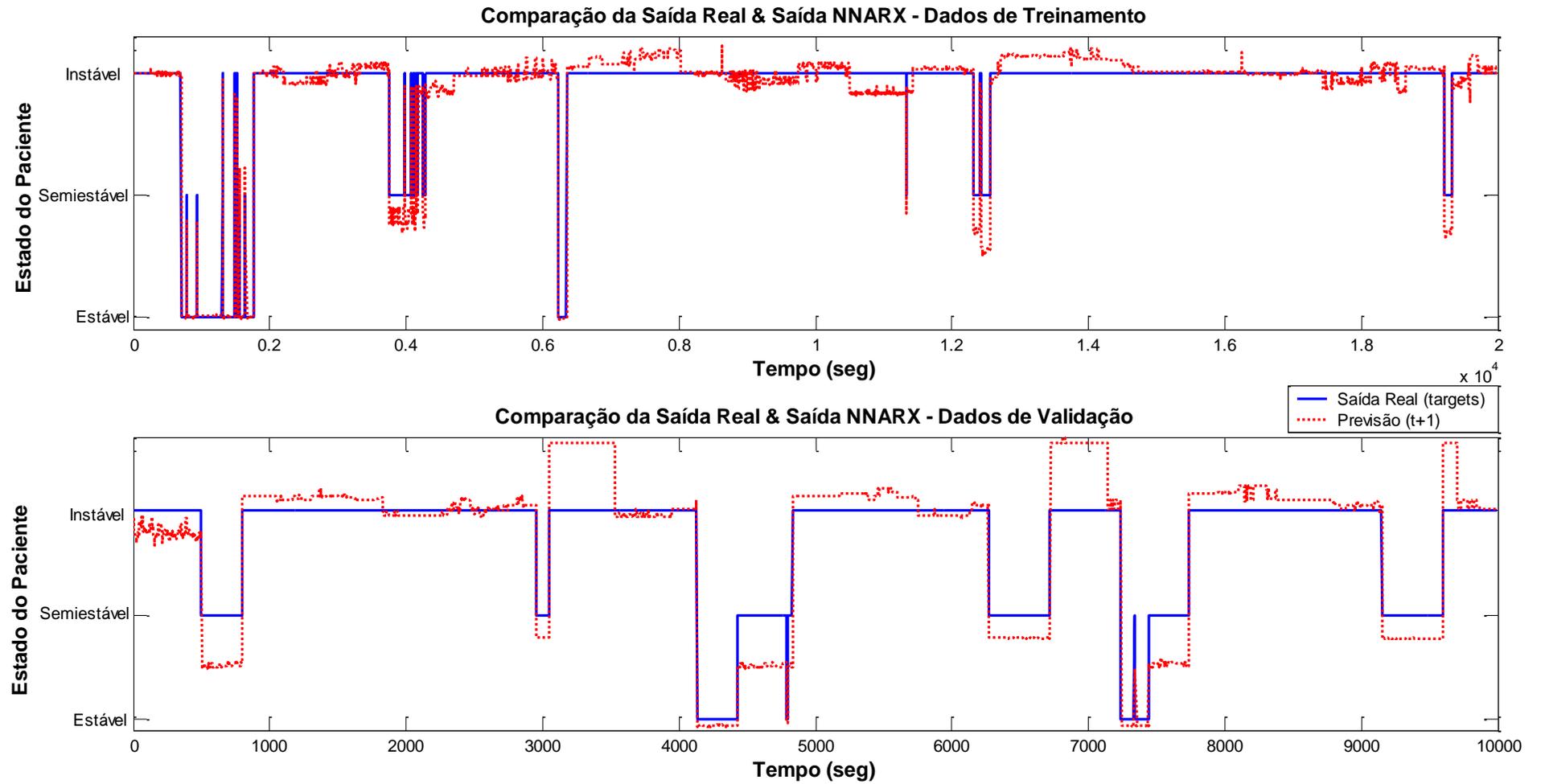


Figura 42 – Validação da rede neural NNARX
Dados de treinamento e de validação

A função *nvalid*¹³ permite validar uma rede neural de um sistema dinâmico e produz uma série de gráficos de interesse. Analisando a função de autocorrelação da figura 43 é possível observar a autocorrelação do erro de predição, em que a curva não se encontra dentro de sua faixa de confiança (sinalizada com linhas tracejadas).

Também os valores de correlação cruzada das variáveis FC, PS, PD, PM e SO estão fora da faixa de confiança para diversos atrasos (*lag*, em inglês).

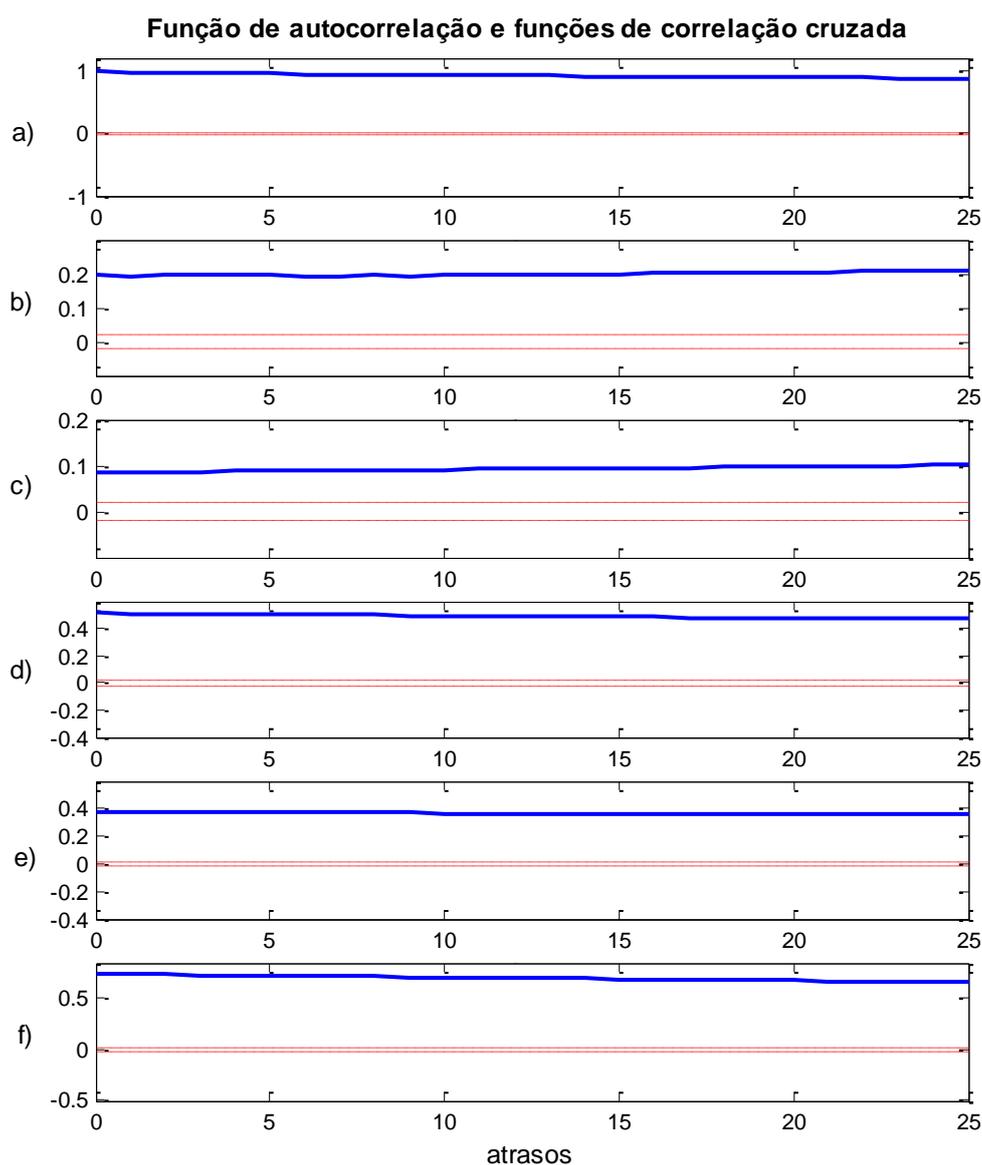


Figura 43 – Funções de autocorrelação e de correlação cruzada – Rede NNARX

a) Função de autocorrelação do erro de predição; b - f) Funções de correlação cruzada entre cada entrada e o erro de predição: b) da frequência cardíaca; c) da pressão sistólica; d) da pressão diastólica; e) da pressão média; f) da saturação de oxigênio. Dados de Validação

¹³ Escrita por Magnus Nørgaard da Technical University of Denmark (MATHWORKS PRODUCT DOCUMENTATION, 2014).

Lembrando que a correlação cruzada tem por objetivo encontrar um sinal de curta duração inserido num sinal mais longo e que a autocorrelação busca padrões de repetição num sinal periódico com ruído, podemos inferir que o comportamento observado na figura 43 comprova a não linearidade do sistema.

RESULTADOS DO MODELO NNARX(6,3,1) PODADO – P-NNARX

A tabela 26 apresenta a matriz de confusão dos resultados obtidos pela rede P-NNARX otimizada via poda.

Tabela 26 – Matriz de confusão do modelo P-NNARX

		Realidade			Total Classificados
		Estável	Semiestável	Instável	
Classificado r	Estável	494	4	0	498
	Semiestável	2	1.993	6	2.001
	Instável	2	4	7.494	7.500
	Total Reais	498	2.001	7.500	9.999
	Predições Corretas (%)	99,20	99,60	99,92	99,57
	Erros otimistas (%)	0,00	0,20	0,08	0,09
	Erros pessimistas (%)	0,80	0,20	0,00	0,34

Os resultados acima detalhados demonstram que a rede P-NNARX tem uma excelente taxa de acerto com uma exatidão de 99,57% para os três estados. Dos erros, 0,34% correspondem a resultados errados pessimistas (saída real melhor que a prevista) e 0,09% de resultados errados otimistas (saída real pior que a prevista). Os momentos estáveis analisados tiveram uma acurácia de 99,20%, os momentos analisados como semiestáveis tiveram 99,60% de acerto e finalmente os momentos onde o paciente encontrava-se instável foram preditos com 99,92% de acerto. Ou seja, segundo a classificação estabelecida na tabela 3, a rede tem um excelente comportamento para predizer qualquer dos três estados possíveis. Uma vez que o comportamento geral da rede também é excelente (99,57%), esta topologia pode ser considerada para a construção do sistema de monitoramento de pacientes.

A rede NNARX e P-NNARX são submetidas às equações (20-22,26,27) para medir o seu desempenho. Os resultados são apresentados na tabela 27.

Tabela 27 – Análise das redes NNARX e P-NNARX

Topologia da rede	NNARX	P-NNARX
MSE	0,115	0,002
Coefficiente Correlação	0,945	0,994

FPE	0,041	0,003
Sensibilidade (%)	98,57	99,89
Especificidade (%)	98,61	98,41
Acurácia Geral (%)	98,57	99,82

Segundo os valores obtidos na tabela 27, a rede P-NNARX apresenta melhorias no desempenho em relação à rede NNARX. Na rede P-NNARX os valores de MSE e FPE são mais baixos e o de coeficiente de correlação um pouco mais alto que na rede NNARX. Já os valores de sensibilidade e acurácia geral subiram levemente. Então, como conclusão, pode se dizer que a rede P-NNARX apresenta melhores resultados que a rede NNARX.

Todas as qualificações foram feitas considerando o estabelecido na tabela 3, disponível na seção 2.6.

As validações feitas no conjunto de treinamento e validação, visíveis na figura 44, mostram que as predições estão também bem perto dos *targets*, oferecendo uma resposta melhor que a obtida com a rede NNARX com 25 pesos (figura 42).

Analisando a função de autocorrelação da figura 45 é possível observar uma função de autocorrelação do erro de predição não muito boa, em que a curva também não se encontra dentro de sua faixa de confiança (sinalizada com linhas tracejadas), como ocorreu na rede NNARX.

Também os valores de correlação cruzada das variáveis FC, PS, PM e SO estão fora da faixa de confiança para diversos atrasos (lag). Somente os valores da variável PD permanecem dentro da faixa de confiança para alguns poucos atrasos. Todo este comportamento comprova, também neste caso, a não linearidade do sistema.

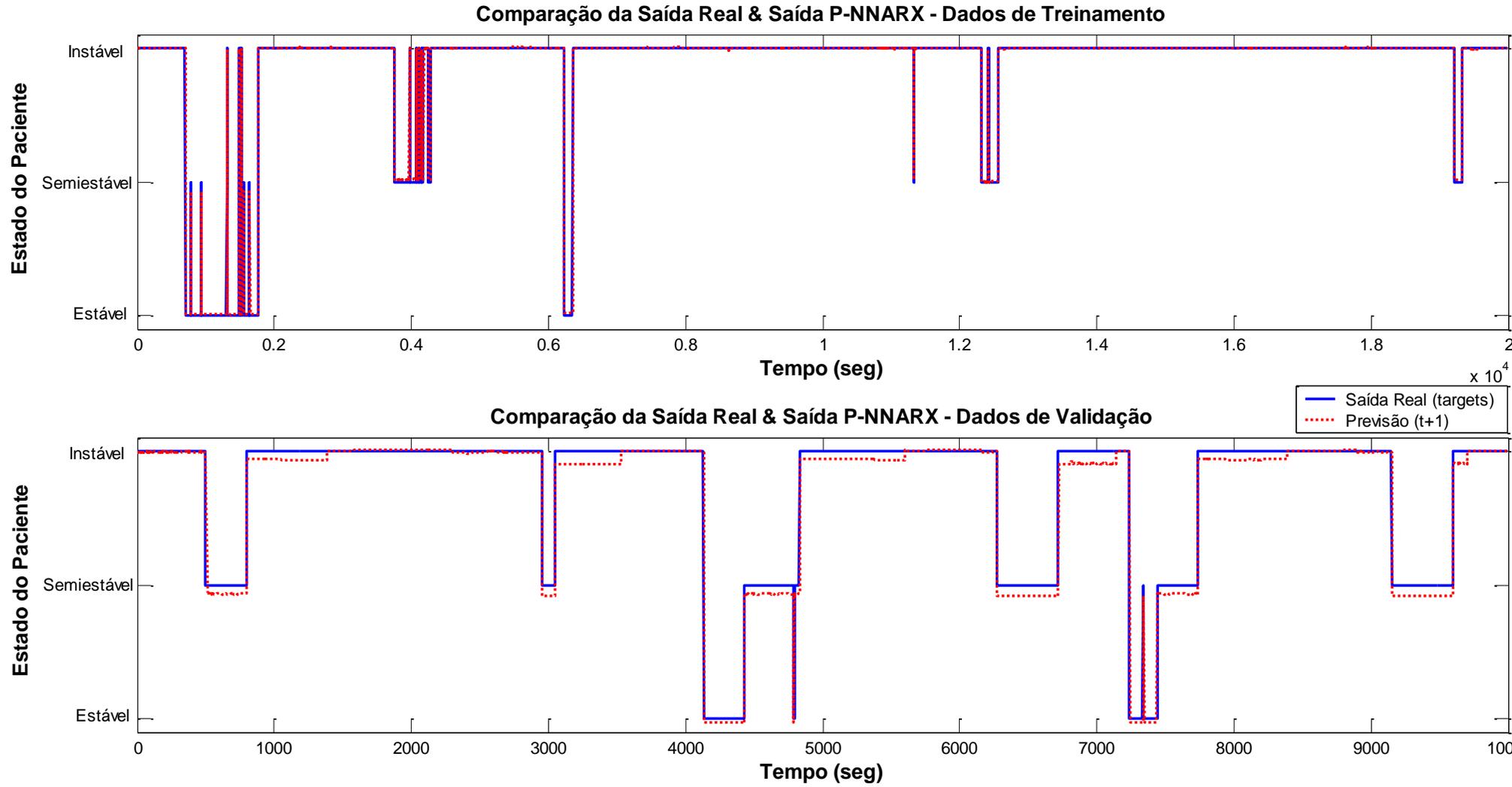


Figura 44 – Validação da rede neural P-NNARX
 Dados de treinamento e de validação

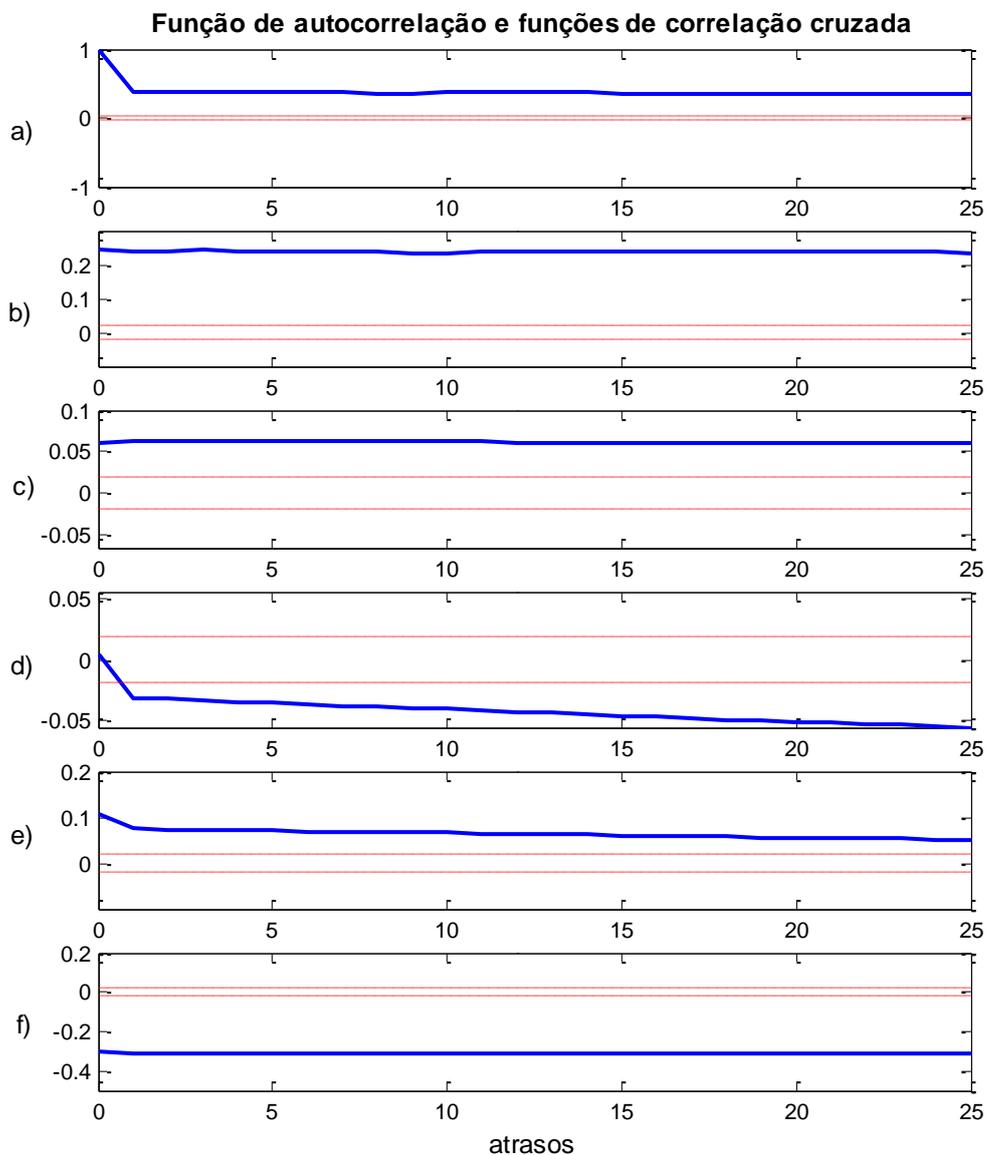


Figura 45 – Funções de autocorrelação e de correlação cruzada – Rede P-NNARX

a) Função de autocorrelação do erro de predição P-NNARX; b) – f) Funções de correlação cruzada entre cada entrada e o erro de predição: b) da frequência cardíaca; c) da pressão sistólica; d) da pressão diastólica; e) da pressão média; f) da saturação de oxigênio.

Fonte: Autoria própria

4.6. ANÁLISE DOS RESULTADOS

Seguindo as equações (20-26) obtemos os valores de sensibilidade (Se), especificidade (Es), acurácia geral (Ac), sensibilidade para os estados estáveis ($Se_{(E)}$), sensibilidade para os estados semiestáveis ($Se_{(SE)}$) e sensibilidade para os estados instáveis ($Se_{(I)}$) das arquiteturas de redes neurais estudadas. Um resumo de todos esses indicadores é apresentado na tabela 28.

Tabela 28 – Indicadores de desempenho das distintas arquiteturas de redes analisadas

	Rede Neural	MSE	Se (%)	Es (%)	Ac (%)	$Se_{(E)}$ (%)	$Se_{(SE)}$ (%)	$Se_{(I)}$ (%)
Dados de Treinamento (10 pacientes)	E-MISO	0,123	87,74	88,02	87,76	89,37	80,85	87,99
	PE-MISO	0,318	88,31	0,00	83,12	0,00	92,13	87,80
	NNARX	0,074	99,79	92,44	99,37	95,86	93,18	99,88
	P-NNARX	0,003	99,88	98,04	99,78	99,20	97,65	99,18
Dados de Validação (10 pacientes)	E-MISO	0,583	71,33	0,79	64,16	1,61	32,68	76,71
	PE-MISO	0,519	87,83	0,00	65,91	0,00	0,15	87,83
	NNARX	0,115	98,57	98,61	98,57	99,40	93,30	99,92
	P-NNARX	0,002	99,89	98,41	99,82	99,20	99,60	99,92

Na tabela anterior pode-se notar que os resultados da rede E-MISO totalmente conectada para os dados de treinamento são muito bons, mas para os dados de validação esta rede tem problemas para prever corretamente os estados estáveis e semiestáveis dos pacientes, já que sua $Se_{(E)}$ tem um valor de só 1,61% e sua $Se_{(SE)}$ de 37,63%. Este resultado provavelmente foi causado por *overfitting* da rede.

A rede PE-MISO não pode ser considerada, uma vez que não pode prever o estado estável nem semiestável com os dados de validação. Isto pode ser visto na Es , na $Se_{(E)}$ e na $Se_{(SE)}$ que têm valor de 0,00%. Cabe lembrar que a especificidade é a capacidade da rede para prever a ausência da doença em pacientes saudáveis.

Já a rede NNARX apresenta excelentes resultados em todos os seus indicadores, com valores superiores a 98% de acurácia geral nos conjuntos de treinamento e de validação. Os valores de $Se_{(E)}$, $Se_{(SE)}$ e $Se_{(I)}$ acima de 93% no conjunto de validação indicam a facilidade desta topologia de rede NNARX para prognosticar qualquer estado clínico.

A rede P-NNARX oferece ainda um desempenho melhor que a rede NNARX. A acurácia da rede superior a 99%, junto com os valores de $Se_{(E)}$, $Se_{(SE)}$ e $Se_{(I)}$ acima de 99% no conjunto de validação, indicam que esta rede consegue prognosticar qualquer estado clínico. Também pode-se constatar a diminuição considerável do MSE de 0,115 que tem a rede NNARX até 0,002 que apresenta a rede P-NNARX.

A poda das redes foi realizada com o fim de obter o número ótimo de neurônios, regressores e conexões, simplificando a sua estrutura e tentando melhorar o seu desempenho. No entanto, a poda aplicada às redes não conseguiu a diminuição de quantidade de neurônios na camada oculta, só a diminuição nos números de conexões. No caso da rede PE-MISO, só conseguiu melhorar a acurácia em 2,73%, mas teve uma importante redução na $Se_{(SE)}$ chegando a quase 0%. Já a rede P-NNARX apresentou melhorias na acurácia subindo 1,27%, e na $Se_{(SE)}$ com 6,75%. Provavelmente esta diferença mínima nos resultados obtidos com a poda das redes seja o resultado da ênfase aplicada na técnica de parada para o treinamento somada aos reiterados treinamentos a fim de conseguir um MSE médio. A aplicação desta técnica conseguiu uma topologia de rede (quantidade de neurônios ocultos) praticamente ótima, fazendo quase desnecessária a técnica de poda aplicada.

Considerando que para escolher a melhor arquitetura de rede entre várias, o mais acertado é escolher aquela que oferece a mais alta acurácia sobre os conjuntos de validação, podemos concluir que a grande diferença obtida com o modelo P-NNARX em relação aos outros modelos analisados faz com que ele seja o escolhido para desenvolver o sistema inteligente preditivo para avaliar pacientes.

4.7. RESULTADOS DA IMPLEMENTAÇÃO DO SISTEMA INTELIGENTE PREDITIVO FUZZY-NNARX (FNNARX)

Uma vez escolhida a rede P-NNARX como rede neural para a predição do estado clínico do paciente, foi desenvolvido o sistema inteligente utilizando MATLAB versão 6.0.

Este passo envolve a estruturação, codificação, testes e implementação do sistema inteligente para monitoramento de pacientes composto do modelo *fuzzy* para o monitoramento no tempo atual (t) do paciente mais o módulo baseado na topologia P-NNARX para o monitoramento no tempo seguinte ($t+d$), onde $d = 1, 2, \dots$ e é medido em segundos.

Para validar a resposta do sistema em geral, foram utilizados os dados do conjunto de teste composto por 30 novos pacientes, obtidos do site Physionet (PHYSIONET, 2013) com 1.500 amostras cada um. Os dados dos pacientes foram pré-processados com o módulo *fuzzy*

elaborado na seção 3.4 para assim conseguir os *targets* que logo foram utilizados para alimentar a rede NNARX.

Os resultados obtidos do sistema preditivo do estado clínico de pacientes FNNARX são mostrados na tabela 29.

Tabela 29 – Análise dos indicadores de desempenho do sistema FNNARX

Paciente	Banco de Dados	Registro	Sequência de dados	MSE	Se	Es	Ac	Se _{Estável}	Se _{SemiEstável}	Se _{Instável}
Pac01	MIM3	3097068n	15.000-16.499	0,007	99,67	100,00	99,27	100,00	93,10	99,65
Pac02	MIM3	3097167n	2.750-4.249	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac03	MIM3	3097277n	33.000-34.499	0,052	97,33	100,00	94,80	100,00	94,81	94,79
Pac04	MIM3	3099857n	4.200-5.699	0,016	99,66	100,00	98,40	100,00	73,97	99,65
Pac05	MIM3	3097678n	6.500-7.999	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac06	MIM3	3097709n	3.500-4.999	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac07	MIM3	3099857n	11.700-13.199	0,010	99,46	100,00	99,00	100,00	97,90	99,31
Pac08	MIM3	3097780n	4.500-5.999	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac09	MIM3	3097864n	300-1.799	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac10	MIM3	3097922n	1.000-2.499	0,006	99,67	100,00	99,40	100,00	97,99	99,62
Pac11	MIM3	3098142n	4.000-5.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac12	MIM3	3098226n	4.000-5.499	0,001	99,93	100,00	99,93	100,00	100,00	99,93
Pac13	MIM3	3098357n	54.600-56.099	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac14	MIM3	3098373n	35.600-37.099	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac15	MIM3	3098501n	46.000-47.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac16	MIM3	3098536n	20.100-21.599	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac17	MIM3	3098578n	41.000-42.499	0,098	94,88	100,00	90,19	100,00	86,45	92,34
Pac18	MIM3	3098589n	14.000-15.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac19	MIM3	3098757n	46.000-47.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac20	MIM3	3098875n	33.500-34.999	0,019	98,99	100,00	98,07	100,00	98,96	90,07
Pac21	MIM3	3098891n	18.000-19.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac22	MIM3	45470n	13.000-14.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac23	MIM3	45534n	1-1.500	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac24	MIM3	3099857n	10.000-11.499	0,021	99,03	94,98	98,33	98,01	98,68	97,63
Pac25	MIM3	46215n	12.000-13.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac26	MIM3	46066n	26.500-27.999	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac27	MIM3	46359n	22.000-23.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac28	MIM3	10150n	28.000-29.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac29	MIM2	10143n	11.000-12.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
Pac30	MIM2	10160n	1.000-2.499	0,000	100,00	100,00	100,00	100,00	100,00	100,00
MÉDIA				0,008	99,62	99,83	99,25	99,93	98,06	99,10

MIM2: MIMIC II Waveform DB, versão 2; MIM3: MIMIC II Waveform DB, versão 3; MSE: Erro Quadrático Médio; Se: Sensibilidade; Es: Especificidade; Ac: Acurácia Geral; SeE: Sensibilidade Estável; SeSE: Sensibilidade Semi estável; SeI: Sensibilidade Instável

Várias leituras podem ser realizadas em função dos resultados obtidos:

- Em primeiro lugar, o MSE obtido dos dados de validação com 10 pacientes da rede P-NNARX foi de 0,002 e o obtido do sistema inteligente proposto, com 30 pacientes novos, foi de 0,008.

- Os indicadores de sensibilidade, especificidade e acurácia geral, superiores a 99%, indicam uma excelente resposta do sistema. Os três estados possíveis (estável, semiestável e instável) podem ser preditos com uma acurácia geral de 99,25%.
- Do mesmo modo, os indicadores de sensibilidade para os três estados possíveis ($Se_{\text{estável}}$, $Se_{\text{semiestável}}$, $Se_{\text{instável}}$), superiores a 98%, também indicam a excelente classificação realizada pelo sistema preditivo inteligente para cada um dos estados possíveis. Neles, pode-se observar que o sistema tem uma maior facilidade para predizer os estados estável e instável, com valores superiores a 99%, enquanto que a eficácia para predizer os estados semiestáveis chega a 98,06%. Mas ainda assim é considerado um excelente resultado.
- A acurácia geral com um valor de 99,25% está bem perto da acurácia obtida no conjunto de validação da rede P-NNARX, a qual foi de 99,82%. Esta diferença mínima pode ser explicada pelo fato que o sistema proposto foi testado com 30 pacientes nunca vistos. Este valor alto de acurácia geral do sistema inteligente proposto demonstra o alto grau de generalização do sistema.
- Aqueles casos em que os indicadores Se , Es , Ac , $Se_{\text{estável}}$, $Se_{\text{semiestável}}$ e $Se_{\text{instável}}$, chegam a 100,00% correspondem a pacientes que não apresentam variações de estado ao longo da amostragem, por exemplo os pacientes Pac02 e Pac30. Em contrapartida, os casos em que a acurácia tem um valor diferente de 100,00%, indicam que o paciente apresenta mais de um estado clínico ao longo da amostragem, ou seja alguma variação no seu estado. Por exemplo, os pacientes Pac01, Pac03, Pac04, Pac07, Pac10, Pac17, Pac20 e Pac24. Os gráficos dos estados clínicos destes pacientes são apresentados nas figuras 47 e 48.
- O paciente Pac17, que apresenta a menor acurácia com um valor de 90,19% apresenta uma curva com muitas variações de estado continuamente e sem periodicidade (ver figura 48).
- O sistema proposto apresenta um comportamento excelente com aqueles pacientes que não apresentam muitas alterações de estado em breves períodos de tempo, ou seja, a curva não tem a tendência a apresentar picos breves e repetidos de forma não periódica.
- Como conclusão, pode-se afirmar que o sistema proposto para predição do estado clínico de pacientes apresenta uma excelente resposta geral, sendo ela ótima nos

casos em que o paciente não apresenta muitas variações de estado clínico durante a amostragem.

Com o fim de analisar a tendência dos resultados com um tempo de predição maior, foi realizado um estudo comparativo das seguintes predições: $(t + 10)$, $(t + 30)$ e $(t + 60)$. Estes resultados são apresentados na tabela 30.

Tabela 30 – Tendência dos indicadores com diferentes tempos de predição							
Predições	MSE	Se (%)	Es (%)	Ac (%)	Se _{Estável} (%)	Se _{SemiEstável} (%)	Se _{Instável} (%)
Tempo $(t + 1)$	0,008	99,62	99,83	99,25	99,93	98,06	99,10
Tempo $(t + 10)$	0,035	97,94	99,31	96,49	99,73	92,23	95,22
Tempo $(t + 30)$	0,049	97,05	99,00	95,10	99,58	88,79	92,92
Tempo $(t + 60)$	0,054	96,48	98,33	94,58	99,17	92,33	93,32

Na tabela 30 e na figura 46 podem-se observar que na medida em que o tempo de predição aumenta o erro MSE também aumenta gradualmente. Da mesma forma, os restantes indicadores diminuem também gradualmente, mas ainda conservam um resultado excelente. A acurácia para as predições no tempo $(t + 1)$ chega a 99,25% e para as predições no tempo $(t + 60)$ diminuem em quase 5% chegando até 94,58%, mas continuando na faixa dos 90%.

Pode-se concluir que a rede apresenta excelentes resultados para predições de um passo à frente, assim como também de até 60 passos à frente, mas é de se esperar uma ligeira diminuição na sua acurácia geral.

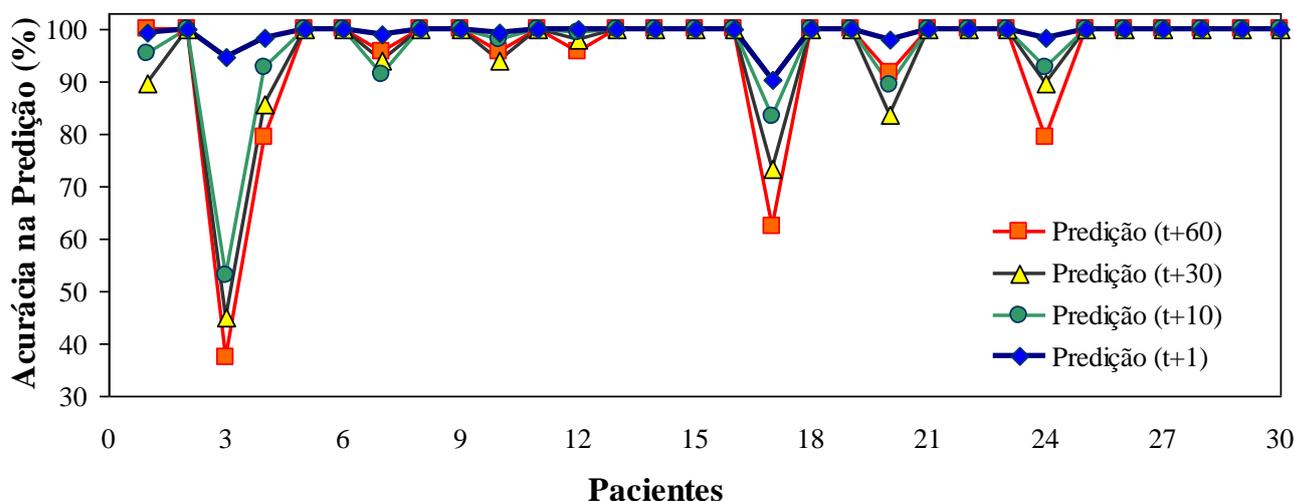


Figura 46 – Curvas das diferentes predições – Sistema FNNARX

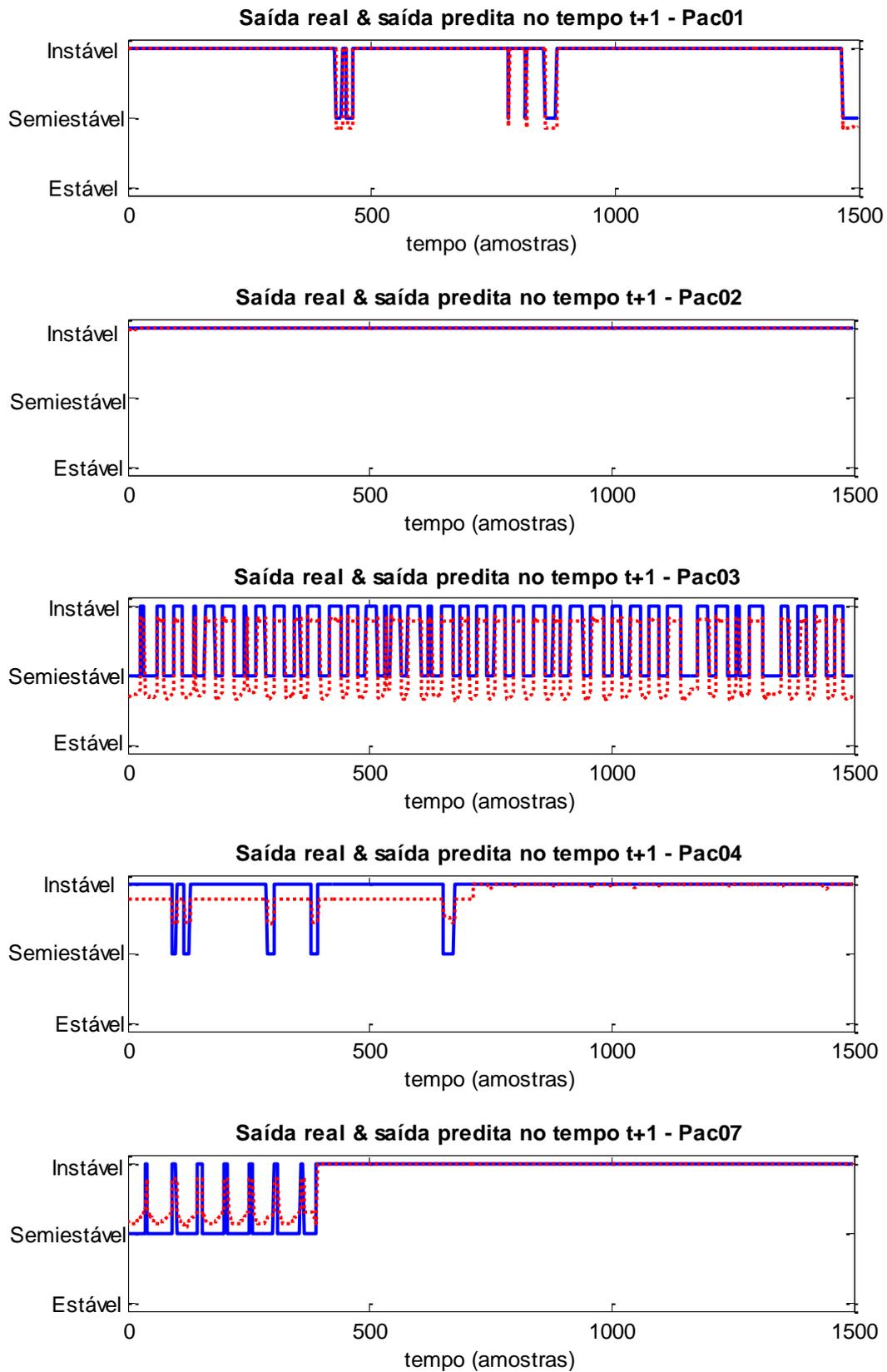


Figura 47 - Comparações da saída real vs. a saída predita dos pacientes Pac01, Pac02, Pac03, Pac04 e Pac07
Fonte: Autoria própria

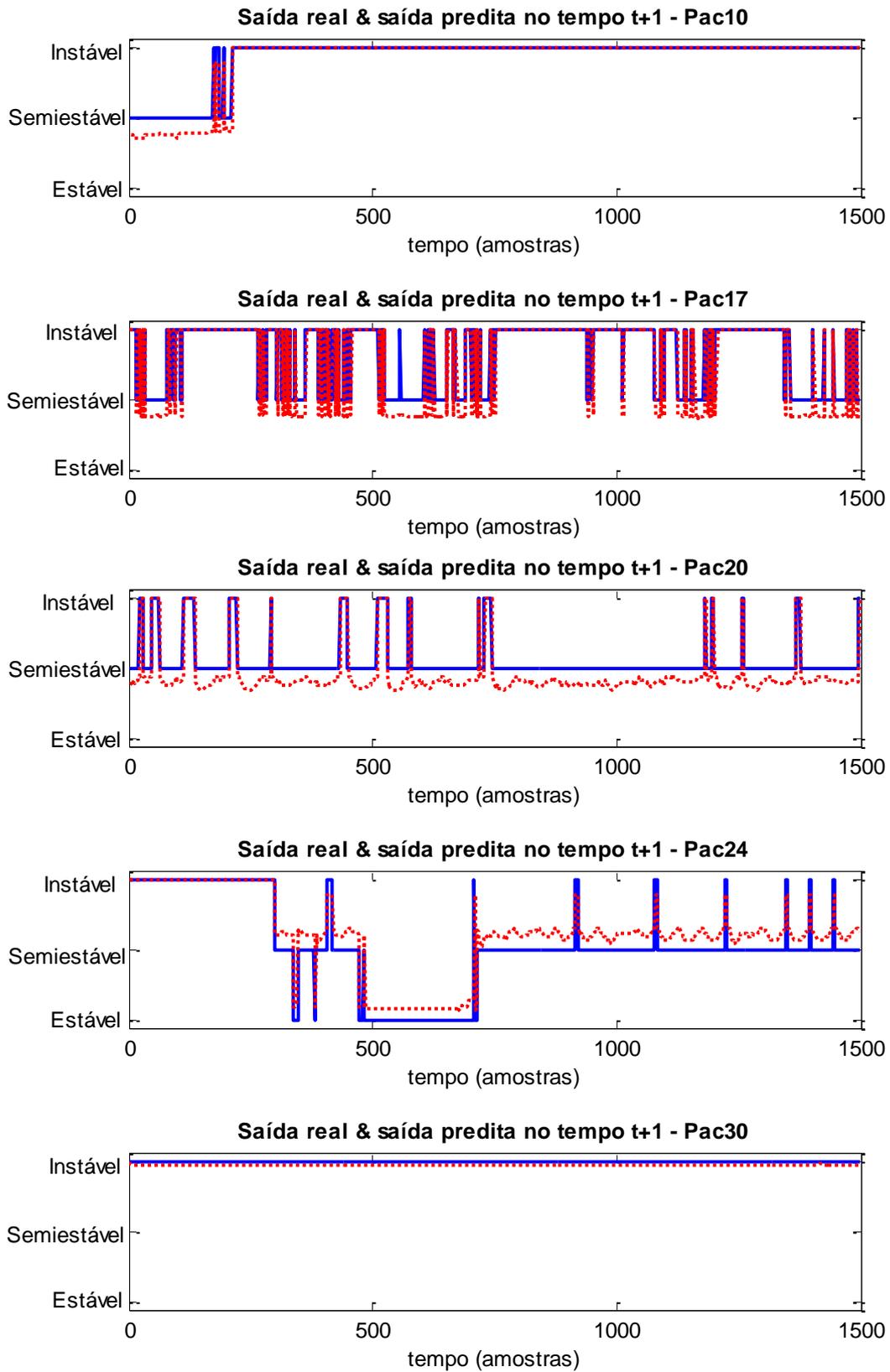


Figura 48 - Comparações da saída real vs. a saída predita dos pacientes Pac10, Pac17, Pac20, Pac24 e Pac30
 Fonte: Autoria própria

CONCLUSÕES E PESQUISAS FUTURAS

Esta tese começou com o intento de responder a seguinte pergunta:

Como podemos construir um sistema inteligente, seguro e confiável capaz de interpretar os sinais vitais de um paciente em um momento atual e predizer o possível estado dele em um futuro próximo, sendo requisitos fundamentais a sua mobilidade e conforto?

Os benefícios de uma rede de sensores *wireless* para monitoramento de pacientes ficaram amplamente demonstrados em virtude de sua comodidade e liberdade de movimentos para os pacientes. Se a rede consiste em pequenos e leves sensores, com baterias de alta durabilidade, não invasivos, confiáveis e seguros, ela pode oferecer ao paciente conforto e segurança no monitoramento, permitindo-lhe continuar com o desenvolvimento normal de suas atividades diárias. A informação obtida dos sensores não é somente valiosa para o conhecimento do estado do paciente no tempo atual, mas também como fonte de dados para o desenvolvimento de ferramentas baseadas em técnicas de *datamining* para a detecção de doenças futuras.

O sistema inteligente desenvolvido intenta auxiliar o médico no monitoramento constante do paciente, emitindo alarmes em casos quando os parâmetros estão fora da faixa de valores estabelecidos como normais. Ele conta com as vantagens da lógica *fuzzy* e das redes neurais. As vantagens obtidas da lógica *fuzzy* referem-se à facilidade na elaboração de regras *fuzzy* partindo do conhecimento dos especialistas e a sua particularidade de lógica multivalente que permite que um elemento possa pertencer a um, dois ou mais conjuntos simultaneamente. Na lógica *fuzzy*, as fronteiras entre estes conjuntos é difusa, vaga, característica muito aproveitada em sistemas não determinísticos, não lineares e dinâmicos como são, neste caso, os sinais fisiológicos vitais da área médica. O modelo *fuzzy* desenvolvido neste trabalho consegue interpretar os sinais do paciente no momento atual. Para a predição do estado no paciente no tempo seguinte, foi necessária a utilização de redes neurais recorrentes.

O benefício obtido das redes neurais artificiais é a capacidade de identificação de sistemas sem precisar possuir o conhecimento necessário do processo para conseguir gerar as relações funcionais entre as entradas e saídas do sistema. Elas são muito úteis para resolver problemas não lineares cuja solução requer conhecimento difícil de especificar, mas para os quais existe uma abundância de exemplos. As RNAs aprendem de exemplos fornecidos e, através de um algoritmo de aprendizado, conseguem adaptar os parâmetros da rede para convergir numa solução. A utilização das redes neurais recorrentes neste trabalho permitiu a predição de

estados futuros, através da realimentação do modelo com saídas anteriores. Ou seja, concedeu ao sistema uma memória curta que permite estabelecer uma previsibilidade na condição do paciente.

A fim de resolver problemas complexos, com frequência a solução ideal consiste em uma inteligente combinação de várias técnicas, cada uma das quais deve ser aplicada a esse aspecto particular do problema a que melhor se encaixa. Neste trabalho, o estado do paciente no tempo atual é definido através de um módulo *fuzzy* por contar com regras proporcionadas pelos especialistas. O módulo preditivo foi realizado com uma rede neural recorrente, por contar com um amplo conjunto de exemplos, cujos *targets* já foram obtidos pelo módulo *fuzzy*.

A fim de trabalhar com dados reais foram utilizados os sinais fisiológicos reais de pacientes armazenados em bancos de dados do site de livre acesso PhysioNet. Para a seleção destes pacientes, escolhidos num total de 58, foi levada em consideração a completude dos dados de cada um dos parâmetros, evitando extensos intervalos com zeros, e dando preferência à variedade nas condições dos pacientes. Para obter o conjunto de treinamento foram tomados 2.000 registros de 10 pacientes, para os dados de validação 1.000 de outros 10 pacientes e para os dados de teste 1.500 de outros 30 pacientes. Cada conjunto de dados dos pacientes foi dividido em três seções e intercalados entre eles, com o propósito de conseguir maior variabilidade na série de dados e evitar qualquer periodicidade involuntária.

O treinamento das redes foi feito unicamente com o conjunto de treinamento, avaliando o erro no conjunto de validação a cada 200 iterações (épocas). O treinamento foi interrompido quando o erro do conjunto de validação foi maior do que era 200 iterações atrás. Então, nesses casos, o conjunto de pesos anteriores foi utilizado como o resultado do treinamento. Esta operação foi realizada 10 vezes obtendo-se uma média dos valores de MSE, FPE e os coeficientes de correlação.

Neste trabalho, um modelo *fuzzy*, duas redes neurais Elman, a rede NNARX e as técnicas de poda das redes foram estudadas, testadas e comparadas, a fim de avaliar com que exatidão conseguem monitorar e prever o comportamento não linear dos sinais vitais do paciente.

As respostas do modelo *fuzzy* foram excelentes, com 99,76% de coincidências em relação às respostas dadas pelos especialistas. No entanto, ficou claro que as saídas do modelo *fuzzy* são mais próximas ao raciocínio humano, devido à tolerância nas fronteiras de valores, do que aquelas obtidas dos especialistas expressas numa tabela com as faixas normais e anormais dos diferentes parâmetros testados. Isto acontece por causa da natureza difusa do modelo *fuzzy* nas fronteiras das funções de pertinência de cada entrada.

Experiências e estudos comparativos com conjunto de dados não lineares de pacientes reais demonstram um desempenho superior das redes NNARX e P-NNARX sobre as redes Elman. Na rede NNARX foi utilizado o critério MDL para selecionar a ordem das variáveis de entrada ao aproximador funcional, ou seja para escolher a ordem do modelo.

Tanto a rede E-MISO e a rede NNARX foram podadas, a fim de melhorar o seu desempenho com a diminuição de conexões e/ou unidades escondidas. As podas das redes não conseguiram uma diminuição na quantidade de neurônios ocultos, mas sim de conexões. No caso da poda da rede E-MISO, a melhoria no desempenho foi muito leve, mas mostrou deficiências na classificação dos estados estáveis e semiestáveis. A rede NNARX ofereceu excelentes indicadores com os dados de validação, ver tabela 28, mas a sua poda melhorou ainda esses indicadores, chegando a uma acurácia de 99,82%.

A aplicação da técnica de parada no treinamento junto com os reiterados treinamentos, a fim de se conseguir valores de MSE, de FPE e de coeficientes de correlação médios, mostraram sua efetividade em achar uma topologia de rede bem perto da ótima, fazendo quase desnecessária a técnica de poda.

Finalmente, segundo os resultados observados na tabela 28, pode-se concluir que a rede P-NNARX é o modelo que melhor se ajusta ao sistema. Para este caso particular, as redes neurais recorrentes NNARX capturam a essência do sistema dinâmico não linear muito melhor do que as redes recorrentes Elman. Por este motivo foi o modelo escolhido para desenvolver o módulo preditivo do sistema inteligente FNNARX.

A acurácia obtida com o modelo FNNARX com o tempo de predição $(t + 1)$ foi de 99,25%. Este resultado demonstra o comportamento real do sistema, uma vez que foi testada com os dados de trinta novos pacientes (pacientes nunca vistos) e ilustra o alto nível de generalização do sistema.

Outros testes com tempos de predição de $(t + 10)$, $(t + 30)$ e $(t + 60)$ indicam uma ligeira diminuição na acurácia, chegando até 94,58% com $(t + 60)$. Este comportamento indica que quanto maior for o período a ser previsto, menor é a acurácia obtida, conforme já era esperado.

Espera-se que com o desenvolvimento de trabalhos futuros, o sistema inteligente possa ser utilizado por especialistas clínicos em hospitais, hospitais-dia ou casas de repouso para idosos para poder realizar trabalhos de telemonitoramento em pacientes não hospitalizados, convertendo-se numa ferramenta complementar para os médicos, permitindo uma melhor qualidade de vida aos pacientes e diminuindo os custos hospitalares.

Considera-se que o contínuo avanço na tecnologia de sensores WBAN, de tamanhos reduzidos, leves e com larga durabilidade nas baterias, tornou o uso contínuo destes sensores cada vez mais confortável, permitindo um telemonitoramento invisível e cômodo.

5.1. TRABALHOS FUTUROS

Além do trabalho aqui apresentado, há muitas áreas para exploração neste domínio e diversas tarefas complementares que permitirão enriquecer o sistema proposto. A autora desta tese está particularmente interessada em:

- **Integração do sistema inteligente a uma WBAN:** Criação da rede de sensores biomédicos de área corporal, obtenção das leituras de dados e transmissão sem fios dos sinais vitais dos pacientes ao sistema desenvolvido nesta tese.
- **Diferenciação de pacientes:** Consideração das características variáveis dos pacientes, como gênero, idade, altura, peso, medicação, mobilidade, hora do dia.
- **Personalização e adaptação do sistema às necessidades de cada paciente:** Para atingir este ponto seria necessário integrar um módulo de personalização das faixas de pertinência do módulo *fuzzy*, configurável pelo especialista médico, e o módulo de treinamento da rede neural.
- **Integração a módulos inteligentes:** Integração do sistema a outros módulos inteligentes, para um tratamento personalizado ou emissão de diagnóstico.
- **Sistema multipacientes:** Criação de uma rede centralizada com vários pacientes, que sejam simultaneamente controlados em tempo real.
- **Controle de integridade do fluxo de entrada:** Verificação da integridade dos dados de entrada, como atrasos, conjunto de dados com zeros, valores de sinais vitais fora da faixa dos valores esperados.
- **Alertas SMS:** Implementação de alertas para os especialistas, que serão enviados quando os sinais vitais dos pacientes estejam instáveis.
- **Quadro de Informação do Paciente:** Geração de um sistema para acessar as informações do paciente, aplicável, por exemplo, em hospitais para idosos, serviços de atendimento domiciliar ou asilos.

REFERÊNCIAS

AIBINU, A.; SALAMI, M.; SHAFIE, A.; NAJEEB, A. Comparing Autoregressive Moving Average (ARMA) coefficients determination using Artificial Neural Networks with other techniques. **World Academy of Science, Engineering and Technology**, v. 42, p. 173-178, 2008.

AKAIKE, H. Information theory as an extension of the Maximum Likelihood principle, **In Second International Symposium on Information Theory**, v. 1, p. 267-281, 1973.

AKYILDIZ, I.; SU, W.; SANKARASUBRAMANIAM, Y.; CAZIRCI, E. Wireless sensor networks: a survey, **Computer Networks**, v. 38, no. 4, p. 393-422, 2002.

ALERT: Alert-it Care Alarm Technology. Disponível em: <<http://www.alert-it.co.uk>>. Acesso em: 28 abr. 2013.

AMERICAN HEART ASSOCIATION. <<http://www.heart.org>>. Acesso em: 27 nov. 2013.

ARAHAL, Manuel; SORIA, Manuel; DIAZ, Francisco. **Técnicas de predicción con aplicaciones em Ingeniería**. Sevilla: Universidad de Sevilla, 2006.

BASHSHUR, R.; SHANNON, G.; KRUPINSKI, E.; GRIGSBY, J.; KVEDAR, J.; WEINSTEIN, R.; SANDERS, J.; RHEUBAN, K.; NESBITT, T.; ALVERSON, D.; MERRELL, R.; LINKOUS, J.; STEWART FERGUSON, A.; WATERS, R.; STACHURA, M.; ELLIS, D.; ANTONIOTTI, N.; JOHNSTON, B.; DOARN, C.; YELLOWLEES, P.; NORMANDIN, S.; TRACY, J. National Telemedicine Initiatives: Essential to Healthcare Reform, **Telemedicine and e-Health**, v. 15, n. 6, p. 600-610, 2009.

BAUM, E. B.; HAUSSLER, D. What net size gives valid generalization? **Neural Computation**, n. 1, p. 151-160, 1989.

BHARGAVA, A.; ZOLTOWSKI, M. Sensors and Wireless Communication for Medical Care, Database and Expert Systems Applications, 2003. **Proceedings of the 14th International Workshop on Database and Expert Systems Applications**, p. 956-960, 2003.

BLACK, D., BROOKES, I., GROVES, R., HOMES, A., HUCKER, H., MCKEOWN, C. Editores, 2009. **Collins English dictionary New Zealand edition (10th ed.)**. Glasgow: Harper Collins.

BLAKE, S.; PITTMAN, S.; MACDONALD, M.; SUN, K.; LANZI, B.; CLARK, D.; HUESER, L.; FABREGAS, S.; SHAMBROOM, J.; WHITE, D. Assessment of a wireless dry sensor to detect sleep in healthy volunteers and subjects with sleep disorders. **Sleep**, v. 32, p. A370, 2009.

BOX, George; JENKINS, Gwilym; REINSEL, Gregory. **Time Series Analysis: Forecasting and Control**. 3 ed. NJ: Prentice Hall, 1994.

CARDIONET. <<http://www.cardionet.com>>. Acesso em: 28 abr. 2013.

CERQUEIRA, E. O.; ANDRADE, J. C.; POPPI, R.; MELLO, C. Redes Neurais e suas aplicações em calibração multivariada. **Química Nova**, v. 24, n. 6, p. 864-873, 2001.

CHIRON – Cyclic and person-centric Health management. Disponível em: <<http://www.chiron-project.eu>>. Acesso em: 28 abr. 2013.

CORVENTIS PiiX. <<http://www.corventis.com>>. Acesso em: 28 abr. 2013.

DIACONESCU, E. The use of NARX Neural Networks to predict Chaotic Time Series. **WSEAS Transactions on Computer Research**, v. 3, p. 182-191, 2008.

DORLAND'S MEDICAL DICTIONARY FOR HEALTH CONSUMERS. Saunders Ed. Elsevier, 2007.

DRAFT GUIDANCE FOR INDUSTRY AND FDA STAFF, U.S. Department of Health and Human Services, Food and Drug Administration, Center for Devices and Radiological Health, 2007.

DUA, S.; SAINI, S.; SINGH, H. Temporal Pattern Mining for Multivariate Time Series Classification. **Journal of Medical Imaging and Health Informatics**, v. 1, p. 164-169, 2011.

DUDA, Richard O.; HART, Peter E.; STORK, David G. **Pattern Classification**, 2. ed. New York: J Wiley, 2001.

E-HEALTH INNOVATION. São Paulo: Publimed Editora, Edição 1, Ano 1. 2012

ENCICLOPEDIA OF SURGERY. Disponível em: <<http://www.surgeryencyclopedia.com/St-Wr/Vital-Signs.html#b>>. Acesso em: 17 mai. 2012.

ELMAN, J. L. Distributed Representations, Simple Recurrent Networks and Grammatical Structure. **Machine Learning**, v. 7, p. 195-225, 1991.

FAISAL, T.; TAIB, M. N.; IBRAHIM, F. Adaptive Neuro-*Fuzzy* Inference System for diagnosis risk in dengue patients. **Expert Systems with Applications**, v. 39, p. 4483-4495, 2012.

FRANKLIN, S. S. Importancia de la presión arterial diastólica en relación con la edad. **Hipertensión, Elsevier España**, v. 24, n. 4, p. 172-175, 2007.

GARCEZ DA LUZ, Paulo Denis. **Sistema de Monitoramento em Tempo Real de Pacientes em Ambiente Hospitalar utilizando Transporte de Informações Biológicas através de Redes Ad-Hoc**. 2008. 168. Dissertação - Universidade Tecnológica Federal de Paraná, Curitiba, Brazil, 2008.

GAO, Y.; ER, M. NARMAX time series model prediction: feed-forward and recurrent *fuzzy* neural network approaches. **Elsevier: Fuzzy Sets and Systems**, v. 150, p. 331-350, 2005.

GAYATRI, M.; KUMAR, A.; JANGHU, M.; KAUR, M.; PRASAD, T. V. Implementation of epileptic EEG using recurrent neural network. **IJCSNS International Journal of Computer Science and Network Security**, v. 10, no. 3, p. 290-296, 2010.

GENERAL ELECTRIC COMPANY, **Vital signs to go wireless with GE's body sensors**. Disponível em: <<http://www.gereports.com/vital-signs-to-go-wireless-with-ge-body-sensors>>. Acesso em: 6 jun. 2012.

GIACOMIN, K.; UCHOA, E.; FIRMO, J. O. A.; LIMA-COSTA, M. F. **Projeto Bambuí: um estudo de base populacional da prevalência e dos fatores associados à necessidade de cuidador entre idosos**. Cadernos de Saúde Pública, Rio de Janeiro, v. 21, p. 80-91, 2005.

GUPTA, V.; POURSOHI, A.; UDUPI, P. Sensor Network: An open data exchange for the web of things. **8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)**, v., p. 753 - 755, 2010.

GUYTON, A. C.; HALL, J. E. **Fisiologia Médica**. Barcelona: Elsevier España, Edição 12. 2011.

HALO: Halo Monitoring. Disponível em: <<http://www.halomonitoring.com>>. Acesso em: 28 abr. 2013.

HASSIBI, B.; STORK, D. G. Second order derivatives for network pruning: Optimal Brain Surgeon. **Advances in Neural Information Processing Systems**, HANSON, S. J; COWAN, J. D.; GILES, C. L. Eds., Morgan Kaufman, v. 5, p. 164-171, 1993.

HAYKIN, Simon. **Neural Networks: A comprehensive foundation**. USA, Michigan: Macmillan, 1994.

HEATON, Jeff. **Introduction to Neural Networks with Java – Second Edition**. USA, Chesterfield: Heaton Research, Inc, 2008.

HEGUY, Silvina. Alistan los últimos detalles para el rescate de los mineros en Chile. **Clarín**, Buenos Aires, 7 out. 2010. Disponível em: <http://www.clarin.com/mundo/america_latina/Alistan-ultimos-detalles-mineros-Chile_0_349165127.html>. Acesso em: 28 abr. 2013.

HOPFIELD, J. J. Neural networks and physical systems with emergent collective computational abilities. **Proceedings of the National Academy of Sciences of the USA**, v. 79, n. 8, p. 2554–2558, 1982.

IME: Innovationszentrum Medizinische Elektronik. Disponível em: <<http://www.ime-muenchen.de/>>. Acesso em: 11 jun. 2012.

IMEC: Interuniversity Microelectronic Centre. Disponível em: <http://www2.imec.be/be_en/press/imec-news/archive-2009/imec-develops-8-channel-wireless-eeg-scanner-for-ambulatory-monitoring.html>. Acesso em: 28 abr. 2013a.

IMEC: Interuniversity Microelectronic Centre. Disponível em: <http://www2.imec.be/be_en/press/imec-news/archive-2008/wireless-eeg-system-self-powered-by-body-heat-and-light.html>. Acesso em: 28 abr. 2013b.

JORDAN. Disponível em: <<http://www.cs.berkeley.edu/~jordan/>>. Acesso em: 31 de mai. de 2012.

JOVANOV, E.; MILENKOVIC, A.; OTTO, C.; DE GROEN, P. A Wireless Body Area Network of intelligent motion sensors for computer assisted physical rehabilitation. **Journal of Neuro Engineering and Rehabilitation**, v. 2, n. 6, p. 16-23, 2005.

KARTALOPOULOS, Stamatios V. **Understanding Neural networks and Fuzzy Logic**. Piscataway, NJ: IEEE Press, 1996.

KARLSRUHER INSTITUT FÜR TECHNOLOGIE. Disponível em: <http://www.kit.edu/besuchen/pi_2013_14247.php>. Acesso em: 28 nov. 2013.

KEARNEY, P. M. Global Burden Of Hypertension: Analysis Of Worldwide Data. London: Lancet, v. 365, n. 9455, p. 217-23, 2005.

KOSKO, B.; ISAKA, S. *Fuzzy Logic*. **Scientific American**, v.269, p. 76-81, 1993, doi:10.1038/scientificamerican0793-76.

LALITHA, V.; ESWARAN, C. Automated detection of anesthetic depth levels using chaotic features with artificial neural networks. **J Med Syst**, v. 31, p. 445-452, 2007.

LAWRENCE, J. **Introduction to Neural Networks: Design, Theory, and Applications**. USA, California: California Scientific Software, 1994.

LEITE, C. R. M.; Sizilio, G. R. A., NETO, A. D. D.; VALENTIM, R. A. M.; GUERREIRO, A. M. G. A *fuzzy* model for processing and monitoring vital signs in ICU patients. **Biomedical Engineering OnLine**, v. 10, p. 1-17, 2011.

LIU, J.; McKENNA, T.M.; GRIBOK, A.; BEIDLEMAN, B. A.; THARION, W.J.; REIFMAN, J. A *fuzzy* logic algorithm to assign confidence levels to heart and respiratory rate time series. **Institute of Physics and Engineering in Medicine**, v. 29, p. 81-94, 2008.

LOCATOR: Locator Systems. Disponível em: <<http://www.locatorsystemscorp.com>>. Acesso em: 07 jun. 2012.

MATHWORKS PRODUCT DOCUMENTATION. Disponível em: <<http://www.mathworks.com/help/matlab/index.html>>. Acesso em: 13 mar 2014.

MAMDANI, E. H. Application of *fuzzy* algorithms for control of simple dynamic plant. In: **Proceedings of IEEE**, v. 121, nro. 12, p. 1585-1588, 1974.

MINISTÉRIO DA SAÚDE: **Cadernos de Planejamento - Sistema de Planejamento do SUS**, vol. 9, Brasília: Editora MS, 2010.

MITCHELL, Tom. **Machine Learning**. Illinois: WCB McGraw-Hill, 1997.

MOHD, Kamaruddin Bin Abd Hamid. **Multiple Faults Detection Using Artificial Neural Network**. 2004. 177. Tese - Faculty of Chemical and Natural Resources Engineering, Universiti Teknologi Malaysia, Malaysia, 2004.

MOSBY'S MEDICAL DICTIONARY. Elsevier, 8th Edition, 2009.

MOUGIAKAKOU, S. G.; PROUNTZOU, A.; ILIOPOULOU, D.; NIKITA, K. S.; VAZEOU, A.; BARTSOCAS, C. S. Neural network based glucose – Insulin metabolism models for children with type 1 diabetes. **In: Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS'06)**, v. p. 3545-3548, 2006.

NLM, U.S. National Library of Medicine. Disponível em: <<http://www.nlm.nih.gov>>. Acesso em: 28 nov 2013.

NØRGAARD, M.; RAVN, O.; POULSEN, N. K.; HANSEN, L. K. Neural network for modelling and control of dynamic systems: a practitioner's handbook, **Springer-Verlag**, London, UK, 2000.

OBERLI, C; URZUA, J. U.; SAEZ, C., GUARINI, M.; CIPRIANO, A.; GARAYAR, B.; LEMA, G.; CANESSA, R.; SACCO, C. IRARRAZAVAL, M. An expert system for monitor alarm integration. **Journal of Clinical Monitoring and Computing**, v.15, p. 29-35, 1999.

OBI, J. C.; IMIANVAN, A. A. Interactive Neuro-Fuzzy Expert System for Diagnosis of Leukemia. **In: Global Journal of Computer Science and Technology**, v. 11, p. 43-50, 2011.

OLIVEIRA REZENDE, Solange. **Sistemas Inteligentes**. São Paulo: Editora Manole, 2005.

PATEL, V. L.; SHORTLIFFE, E. H.; STEFANELLI, M.; SZOLOVITS, P.; BERTHOLD, M. R.; BELLAZZI, R.; ABU-HANNA, A. The coming of age of artificial intelligence in medicine. **In: Artif. Intell. Med.**, v. 46, p. 5-17, 2009.

PHYSIONET. Disponível em: <<http://physionet.org/>>. Acesso em: 18 mai. 2013.

PIEGAS, L. S.; AVEZUM, A.; GUIMARÃES, H. P.; MUNIZ, A. J.; REIS, H. J.; DOS SANTOS, E. S.; KNOBEL, M.; SOUZA, R. Comportamento da síndrome coronária aguda. Resultados de um registro brasileiro. **Arquivos Brasileiros de Cardiologia**, vol.100, no.6; São Paulo: Epub, 2013.

PINSKY, M. R. Hemodynamic Evaluation and Monitoring in the ICU. **Chest – Official publication of the American College of Chest Physicians**, v. 132, nro. 6, p. 2020-20029, doi: 10.1378/chest.07-0073, 2007.

POLO, A.; TERCEDOR, A.; PANIAGUA-SOTO, J.; ACOSTA, F.; CAÑADAS, A. Monitorización neurofisiológica en la cirugía de escoliosis con hipotensión controlada. **Revista Española de Anestesiología y Reanimación**, v. 47, nro. 8, p. 367-370, 2000.

PORTMAN, E.; ANDRUSHEVICH, A.; KISTLER, R.; KLAPPROTH, A. Prometheus – Fuzzy Information Retrieval for Semantic Homes and Environments. **IEEE**, 2010.

PRECHELT, L. Early Stopping – but when?. **Neural Networks: Tricks of the Trade**, Lecture Notes in Computer Science, v. 7700, p. 53-67, 2012.

PRUDÊNCIO, R.; LUDERMIR, T. Design of Neural Networks for Time Series Prediction Using Case-Initialized Genetic Algorithms. **Proceedings of the Eighth International Conference on Neural Information Processing**, v. , p. 990-995, 2001.

RISSANEN, J. Modelling by shortest data description, **Automatica**, v. 14, p. 465-471, 1978.

ROTHMAN, S.; LAUGHLIN, J.; SELTZER, J.; WALIA, J.; BAMAN, R.; SIOUFFI, S.; SANGRIGOLI, R.; KOWEY, P. The Diagnosis of Cardiac Arrhythmias: A Prospective Multi-Center Randomized Study Comparing Mobile Cardiac Outpatient Telemetry Versus Standard Loop Event Monitoring. **Journal of Cardiovascular Electrophysiology**, v. 18, no. 3, p. 241-247, 2007.

ROUSS, V.; CHARON, W. Multi-input and multi-output neural model of the mechanical nonlinear behaviour of a PEM fuel cell system. **Journal of Power Sources**, v. 175, p. 1 -17, 2008.

RUSSELL, S; NORVIG, P. **Inteligência Artificial**. Rio de Janeiro: Elsevier Editora Ltda., 2004.

SAN ROQUE, A.; CZERNICHOW, T. **Predicción de series temporales no lineales con el modelo NARMAX**. Anales de mecánica y electricidad, v. 1, p. 46-57, 2009.

SARLE, W. S. Neural Network FAQ Maintainer: How Many Hidden Units Should I Use?”, **Neural Network FAQ Part 1-7**, <http://www.faqs.org/faqs/ai-faq/neural-nets/part3/section-9.html>, 1996.

SCHATZ, C.; PONTI MEDEIROS, H.; SCHNEIDER, F.; ABATTI, P. **Wireless Medical Sensor Networks: Design requirements and enabling technologies**. *Telemedicine and e-Health*, v. 18, n. 5, p. 394-399, doi:10.1089/tmj.2011.0169, 2012.

SCHATZ, C. V.; MEDEIROS, H.; SCHNEIDER, F.; ABATTI, P. Wireless Protocols for Ad-Hoc Medical Sensor Networks. **Imaging and Signal Processing in Healthcare and Technology**, DOI: 10.2316/P.2011.737-038, 2011

SCHUTZ, S. Oxygen Saturation Monitoring by Pulse Oximetry, In: American Association of Critical-Care Nurses (AACN) Procedure Manual for Critical Care, 4th Ed., p. 77-82, 2001.

SETIONO, R. Generating concise and accurate classification rules for breast cancer diagnosis. **Artificial Intelligence in Medicine**, v. 18, p. 205-219, 2000.

SHAPIRO, S. C. **Artificial Intelligence**. Encyclopedia of Artificial Intelligence, v. 1, 2nd ed. New York: Wiley, 1992.

SJOBERG, J.; HJALMARSSON, H.; LJUNG, L. Neural Networks in System Identification. **10 th IFAC Symp. On System Identification**, v. 2, p. 49-72, 1994.

THE AMERICAN HERITAGE SCIENCE DICTIONARY. Boston; Houghton Mifflin Company, 2005

TORO, J.; AMARILES, E. Hipertensión diastólica e hipertensión sistólica. ¿Cuál es la más importante? Disponível em:
<<http://www.laboratoriosamerica.com.co/medicosonline/congreso2003/descargas/HipertensionDiastolicaySistolica.pdf>>. Acesso em: 15 set. 2012.

ULLAH, S.; HIGGINS, H.; BRAEM, B.; LATRE, B; BLONDIA, C.; MOERMAN, I.; SALEEM, S.; RHAMAN, Z.; KWAK, K. A Comprehensive Survey of Wireless Body Area Networks: on PHY, MAC, And Network Layers Solutions. **Journal of Medical Systems**, v. 36, n. 3, p. 1065-1094, 2012.

URIBE, C.; ISAZA, C.; FLOREZ-ARANGO, J. F. Qualitative-*Fuzzy* Decision Support System for Monitoring Patients with Cardiovascular Risk. **Eighth International Conference on Fuzzy Systems and Knowledge Discovery**, v. , p. 1621-1625, 2011.

WARREN, S.; LEBAK, J. ; YAO, J.; CREEKMORE, J.; MILENKOVIC, A.; JOVANOVIĆ, E. Interoperability and Security in Wireless Body Area Network Infrastructures. **Proceedings of the 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society**, p. 3837-3840, 2005.

WONG, S. V.; HAMOUDA, A. M. S. Optimization of *fuzzy* rules design using genetic algorithm. In: **Advances in Engineering Software**, v. 31, nro. 4, p. 251-262, 2000.

ZADEH, L.A. *Fuzzy Sets*. **Information and Control**, v.8, p.338-353, 1965.

ZEO. Disponível em: <<http://www.myzeo.com>>. Acesso em: 28 abr. 2013.

ZIGBEE, **ZigBee Wireless Sensor Applications For Health, Wellness And Fitness**. Disponível em: <<https://docs.zigbee.org/zigbee-docs/dcn/09-4962.pdf>>. Acesso em: 06 jun. 2012.