



Universidade Federal
do Rio de Janeiro

Escola Politécnica

RECONHECIMENTO DE PADRÕES DE CONDIÇÕES DE FUNCIONAMENTO DE
ROLAMENTOS EM MÁQUINAS ROTATIVAS COM AUXÍLIO DE REDES
NEURAS ARTIFICIAIS.

Fernando da Silva Barra

Projeto de Graduação apresentado ao Curso de Engenharia Mecânica da Escola Politécnica da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Engenheiro.

Orientador: Prof. Fernando Augusto de Noronha Castro Pinto.

Rio de Janeiro
Agosto de 2013



UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
Departamento de Engenharia Mecânica
DEM/POLI/UFRJ



RECONHECIMENTO DE PADRÕES DE CONDIÇÕES DE FUNCIONAMENTO DE
ROLAMENTOS EM MÁQUINAS ROTATIVAS COM AUXÍLIO DE REDES
NEURAIS ARTIFICIAIS.

Fernando da Silva Barra

PROJETO DE GRADUAÇÃO APRESENTADO AO CURSO DE ENGENHARIA
MECÂNICA DA ESCOLA POLITÉCNICA DA UNIVERSIDADE FEDERAL DO
RIO DE JANEIRO, COMO PARTE DOS REQUISITOS NECESSÁRIOS À
OBTENÇÃO DO TÍTULO DE ENGENHEIRO.

Aprovado por:

Prof. Fernando Augusto de Noronha Castro Pinto, Dr. Ing.

Prof. Thiago Gamboa Ritto, D.Sc.

Prof. Vitor Ferreira Romano, Dott.Ric.

RIO DE JANEIRO, RJ - BRASIL

AGOSTO DE 2013

Barra, Fernando da Silva

Reconhecimento de padrões de condições de funcionamento de rolamentos em máquinas rotativas com auxílio de redes neurais artificiais/ Fernando da Silva Barra. – Rio de Janeiro: UFRJ/ Escola Politécnica, 2013.

IX, 54 p.; 29,7 cm.

Orientador: Fernando Augusto de Noronha Castro
Pinto

Projeto de Graduação – UFRJ / POLI / Engenharia Mecânica, 2013.

Referências Bibliográficas: p. 51.

1. Manutenção Preditiva. 2. Medições de Vibrações.
3. Redes Neurais Artificiais. 4. Rolamentos. I. Pinto,
Fernando Augusto de Noronha Castro. II. Universidade
Federal do Rio de Janeiro, UFRJ, Engenharia Mecânica.
III. Título

**Dedico este trabalho a todos
que contribuíram para minha
formação como ser humano e
profissional.**

AGRADECIMENTOS

Agradeço primeiramente a minha família, que forneceu condições essenciais para o desenvolvimento dos meus estudos com tranquilidade e respeito.

Ao orientador Fernando Augusto de Noronha Castro Pinto, pela compreensão e sabedoria compartilhada. E Wallace de Souza Pacheco, pela dedicação, paciência e apoio durante todo o período de desenvolvimento deste trabalho.

Aos amigos que compartilharam deste árduo caminho: Ronald Sudré, Rafael Chaves, Gabriel Mesquita, Márcio Barbosa, Sandro Alves, Maurício Machado, Gustavo Bittencourt e Poti-Ayrá.

RESUMO

Este trabalho apresenta o estudo sobre a aplicação de redes neurais artificiais para reconhecimento de padrões de condição de rolamentos em máquinas rotativas. Em um modelo experimental, localizado no Laboratório de Acústica e Vibrações da UFRJ, foram medidos sinais de vibração em um mancal específico. As condições de funcionamento da máquina foram alteradas trocando o rolamento que funcionava de forma padrão, por um rolamento defeituoso. Os sinais foram medidos utilizando acelerômetros. Com o auxílio do *software* LabVIEW, através do segmento Signal Express, pertencente a National Instruments, foi possível a visualização e exportação dos dados para um formato que pudesse ser tratado pelo *software* MATLAB. Os dados foram divididos em pacotes de amostras menores, as quais foram extraídas características estatísticas que foram utilizadas como parâmetros de entrada da rede neural. A Rede neural artificial escolhida foi uma rede multicamada com retropropagação e alimentação direta. Foram experimentadas diferentes configurações da rede variando a quantidade de camadas ocultas, o número de neurônios nas camadas ocultas e o método de treinamento.

ABSTRACT

This study is about an application of artificial neural networks for pattern recognition of bearings condition in rotating machinery. In an experimental model, located at the Laboratory of Acoustic and Vibrations at UFRJ, were measured vibrations signals in a specific bearing. The operating conditions of the machine were altered by changing a standard operating bearing for a defective bearing. The signals were measured by the use of accelerometers. With the use of software LabVIEW Signal Express (National Instruments), it was possible to view and export data to a format that could be handled by the software MATLAB. The data packets were divided into smaller samples, which statistical characteristics were extracted to be used as input parameters of the neural network. The chosen artificial neural network was a multilayer with backpropagation and feedforward. Several network configurations were tested varying the amount of hidden layers, the number of neurons in hidden layers and the training method.

ÍNDICE

AGRADECIMENTOS.....	v
RESUMO	vi
ABSTRACT	vii
CAPÍTULO 1 - INTRODUÇÃO	1
CAPÍTULO 2 -MANUTENÇÃO PREDITIVA.....	3
CAPÍTULO 3 - ROLAMENTOS	5
3.1 Rolamentos	5
3.2 Características Construtivas	6
3.2.1 Construção dos rolamentos	7
3.2.2 Classificação de Rolamentos	7
CAPÍTULO 4 - SENSORES.....	11
4.1 Erros Associados.....	12
4.1.1 Erros de Inserção	12
4.1.2 Erros de Operação.....	12
4.1.3 Erros Característicos.....	12
4.1.4 Erros Dinâmicos.....	13
4.1.5 Erros de Ambiente.....	13
4.2 Características dos Sensores	13
4.2.1 Sensibilidade	13
4.2.2 Erro de Sensibilidade.....	14
4.2.3 Faixa (<i>Range</i>)	14
4.2.4 Precisão	15
4.2.5 Resolução	15
4.2.6 Acurácia.....	15
4.2.7 Offset.....	15
4.2.8 Linearidade.....	15
4.2.9 Histerese.....	16
4.2.10 Tempo de Resposta.....	16
4.3 Acelerômetro	16
CAPÍTULO 5 - REDES NEURAIAS ARTIFICIAIS	19
5.1 O conceito de Rede Neural Artificial	19

5.2 Modelos de Neurônios.....	20
5.2.1 Modelo de Neurônio Simples.....	20
5.2.2 Função de Transferência.....	21
5.2.3 Modelo de Neurônio com Vetor Input.....	22
5.3 Camadas de neurônios.....	24
5.3.1 Camadas Simples de Neurônios.....	24
5.3.2 Camadas Múltiplas de Neurônios.....	25
5.4 Redes Neurais <i>Feedforward</i> (Alimentação Direta).....	26
CAPÍTULO 6 - REDES NEURAI ARTIFICIAIS NO MODELO EXPERIMENTAL.....	27
6.1 Domínio do Tempo.....	27
6.2 Modelo Experimental.....	27
6.3 Aquisição de dados.....	30
6.4 Tratamento dos dados.....	35
CAPÍTULO 7 - VARIAÇÕES DAS CONFIGURAÇÕES DA REDE NEURAL.....	37
7.1 Aprendizagem.....	37
7.1.1 Rede Neural Artificial Supervisionada.....	37
7.1.2 Retropropagação (<i>Backpropagation</i>).....	38
7.2 RNA Proposta.....	38
7.2.1 Configurações Iniciais.....	40
7.3 Quantidade de Neurônios e Quantidade de camadas ocultas.....	41
7.4 Variações do Método.....	43
7.4.1 BFGS Quasi-Newton.....	44
7.4.2 Resilient Backpropagation.....	45
7.4.3 Scaled Conjugate Gradient.....	47
CAPÍTULO 8 - CONCLUSÃO.....	49
TRABALHOS FUTUROS.....	50
REFERÊNCIAS BIBLIOGRÁFICAS.....	51
APÊDICE A. Algoritmos.....	52
Função tratamento de dados.....	52
Rede Neural Artificial.....	53

CAPÍTULO 1

INTRODUÇÃO

Rolamentos são considerados componentes fundamentais para o funcionamento, em condições ideais, de máquinas rotativas. Este estudo desenvolveu-se com o objetivo de avaliar a possibilidade de compor um sistema de manutenção preditiva, que através de análise de padrões possa monitorar e diagnosticar as condições de funcionamento da máquina, com foco em defeitos relacionados a rolamentos. Segundo LIU, T. I., MENGEL, J. M., (1992), rolamentos de esferas são amplamente utilizados em vários tipos de máquinas e equipamento. A fim de melhorar a produtividade e a qualidade do produto, um sistema de monitorização on-line e contínuo é essencial para verificar o estado dos rolamentos.

Para a aquisição de dados, iremos fazer uso de acelerômetros que irão fornecer sinais de aceleração no domínio do tempo. Estes sinais serão analisados e processados utilizando o segmento Signal Express do LabVIEW, ambos vinculados a plataforma da National Instruments. Posteriormente, estes dados serão exportados para o MATLAB, onde serão calculadas características estatísticas. Espera-se que tais características forneçam os parâmetros de entrada necessários para o desenvolvimento de um sistema de inteligência artificial, que possa reconhecer esses padrões de funcionamento. De acordo com SAMANTA, B. *et al.* (2003), média, rms, variância, assimetria e curtose são indicados para serem extraídos do sinal de vibração para o diagnóstico.

Uma rede neural artificial será configurada com o auxílio do *toolbox* do MATLAB e treinada para que possa reconhecer, a princípio, dois padrões de funcionamento. O padrão de funcionamento de um rolamento em perfeitas condições e o padrão de funcionamento de um rolamento defeituoso.

Em um segundo momento, serão experimentadas diferentes arquiteturas da rede. Com o objetivo de avaliar qual será o comportamento da rede e determinar a arquitetura mais eficiente, serão alteradas as quantidades de camadas ocultas e as quantidades de neurônios nestas camadas ocultas. LIU, T. I., MENGEL, J. M., (1992), demonstra que o sucesso da classificação depende da estrutura da rede neural artificial. Outro fator que

será analisado, são os diferentes métodos de treinamento da rede neural artificial. Os treinamentos serão repetidos utilizando os métodos Levenberg-Marquardt, BFGS Quasi-Newton, *Resilient Backpropagation*, *Scaled Conjugate Gradient*. Os resultados serão plotados em tabelas com as taxas de acerto. Ao final, a expectativa é que seja possível uma avaliação dos efeitos das diferentes arquiteturas e métodos de treinamento na capacidade de reconhecimento de padrão da rede neural artificial.

CAPÍTULO 2

MANUTENÇÃO PREDITIVA

Atualmente, questões sobre o quanto um equipamento pode ser produtivo ou o quanto ele custa são considerações importantes nas tomadas de decisões sobre qual seria a melhor escolha para determinado serviço. Porém, perspectivas envolvendo quanto tempo ele pode operar perfeitamente em condições específicas, sem intervenções, estão se tornando tendências de mercado.

Tal preocupação deve-se ao fator risco, que requer maior atenção quando elevado, tornando a manutenção um importante aliado da produtividade. O tipo de manutenção aplicado pode estar diretamente conectado a questões como: Qual período de tempo necessário para reparos? Corre-se risco de parada inesperada da produção? Qual é o ciclo de vida para o equipamento em condições específicas? É possível programar uma parada para manutenção? É possível prever falhas? Entre outras.

Como explicitado acima, a escolha adequada do tipo de manutenção a ser adotada está diretamente relacionada ao risco e continuidade que a atividade implica, e conseqüentemente, ao custo que o sistema de manutenção escolhido irá acarretar.

Atividades de alto risco e nas quais é fundamental a não parada da produção, tendem a requerer sistemas de manutenção preditiva. Esse tipo de sistema costuma apresentar custos elevados, porém proporciona um maior controle sobre a vida útil do equipamento ou de componentes específicos, o que o torna um fator determinante para a produtividade, reduzindo riscos de paradas inesperadas, possibilitando a programação de paradas para inspeção ou reparos, aumentando a segurança da operação, diminuindo o tempo de não operação do equipamento, proporcionando uma melhor utilização e prolongando a vida útil.

A manutenção preditiva baseia-se, de forma geral, na monitoração e diagnóstico de máquinas ou componentes através do uso de sensores. Os sensores podem medir diversas grandezas: temperatura, vibração, pressão, voltagem, amperagem, entre outras. Eles geram sinais que são traduzidos em dados, os quais devem ser tratados de forma

adequada para que se tornem informações. Tais informações, quando precisas e em tempo hábil, possibilitam tomadas de decisões coerentes pelos respectivos responsáveis. Sendo assim, parte crucial do sistema de produção.

Muitos sistemas de manutenção preditiva e sistemas de diagnósticos de falhas utilizam a condição da máquina para identificar e classificar falhas através da análise de vibrações (ZHANG *et al.*, 1996). A preferência pelo uso de análises de vibrações ao longo dos estudos proporciona, hoje, um vasto histórico de padrões de funcionamento de máquinas e componentes em múltiplas condições de operação, possibilitando o estudo de problemas específicos.

Segundo ARRUDA SANTIAGO, D. F. (2004), a análise de vibrações tem sido amplamente usada no diagnóstico de falhas e monitoração da condição de máquinas rotativas e é feita, em geral, no domínio do tempo ou no domínio da frequência.

A frequência de aquisição de dados varia de sistema mecânico para sistema mecânico. Sendo aconselhável, que para sistemas onde a falha de componentes pode acarretar em danos catastróficos, o intervalo de tempo para aquisição de dados seja menor, conseqüentemente apresentando maior frequência de aquisição.

$$\text{Frequência} = \frac{1}{\text{Período}} = f = \frac{1}{T} \quad (2.1)$$

Possibilitando um controle rígido sobre as condições de funcionamento da máquina. Um exemplo deste tipo de aplicação seria o uso de manutenção preditiva em turbinas a gás. No caso de equipamentos em que falhas não oferecem riscos acentuados, pode-se utilizar frequências de aquisições menores, com intervalos de tempo maiores entre as tomadas de dados. A frequência de aquisição de dados está relacionada também ao custo final, pois maiores frequências significam maior volume de dados adquiridos e conseqüentemente maior será a necessidade de capacidade de processamento para o tratamento desses dados e armazenamento da informação.

A obtenção e tratamento de dados será estudada de forma mais profunda a frente, nos capítulos 6.3 e 6.4

CAPÍTULO 3

ROLAMENTOS

3.1 Rolamentos

Rolamentos são componentes importantes presentes em muitos equipamentos mecânicos. Eles podem ser encontrados em diferentes arranjos que variam de acordo com a necessidade da aplicação, tendo participação efetiva na performance do sistema, contribuindo para o aumento da produtividade e melhora da qualidade do produto.

Seu princípio de funcionamento consiste em diminuir o atrito entre superfícies, utilizando elementos rolantes (esferas ou rolos), para que as superfícies rolem sobre eles com deslizamento mínimo. Os elementos rolantes posicionados entre as superfícies interna e externa, também possuem objetivo de reagir de forma segura às cargas aplicadas e utilizá-las para garantir o máximo de rolamento e mínimo de deslizamento. As cargas as quais geralmente os rolamentos estão sujeitos são radiais ou axiais, ou ainda, uma combinação de ambas.

De forma geral, um rolamento bem dimensionado e condicionado, contribui de maneira significativa para o rendimento do sistema. Dentre os fatores que influenciam o rendimento, podemos destacar: eixos, mancais, lubrificantes e vedação. Tais fatores quando funcionando de maneira inadequada, podem gerar vibrações específicas características.

Para a condição padrão de funcionamento, é necessário que não haja desbalanceamento ou desalinhamento dos eixos e os mancais devem ser especificados de maneira correta para reagir às cargas. O lubrificante tem influência na proteção contra desgastes e corrosão, diminuindo o atrito e assegurando o não aquecimento do arranjo, desta forma, também contribuindo para evitar a perda de potência no rolamento. A vedação deve garantir a não contaminação do lubrificante. Estes, em conjunto devem assegurar um funcionamento padrão do sistema prolongando a vida útil do rolamento.

3.2 Características Construtivas

Algumas características construtivas relativas aos arranjos geralmente encontrados em conjunto com rolamentos, variando de modelo para modelo. Algumas referências podem ser encontradas na ISO 5593:1997: Rolamentos – Vocabulário. Alguns tipos de rolamento serão exemplificados mais adiante no capítulo 3.2.2.

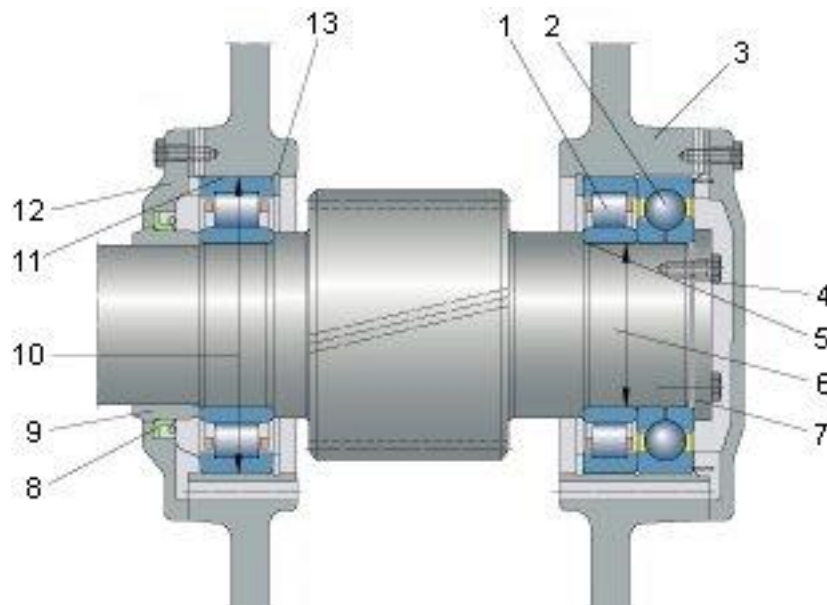


Figura 3.2 - Arranjo de rolamento (SKF Brasil).

- 1 – Rolamento de rolos cilíndricos;
- 2 – Rolamento de esferas de quatro pontos de contato;
- 3 – Mancal;
- 4 – Eixo;
- 5 – Ressalto de encosto do eixo;
- 6 – Diâmetro do eixo;
- 7 – Chapa de fixação;
- 8 – Vedação radial de eixo;
- 9 – Anel espaçador;
- 10 – Diâmetro do fundo do mancal;

11 – Furo interno do mancal;

12 – Tampa do mancal;

13 – Anel de retenção.

3.2.1 Construção dos rolamentos

Um rolamento, em geral, é formado por um anel com pista interno fixado ao eixo e outro anel com pista externo fixado no mancal, corpos rolantes (esferas ou rolos) e um elemento retentor dos corpos rolantes (gaiola).

- Pista: É a superfície de contato que suporta a carga aplicada no rolamento e onde os corpos rolantes giram.
- Corpos rolantes: Podem ser generalizados em dois tipos: rolos e esferas.

A distinção entre eles se deve a geometria, esferas tendem a ter apenas contato pontual entre as pistas dos anéis interno e externo, enquanto os rolos tendem a ter um contato linear ao longo do seu comprimento.

- Gaiolas: A gaiola tem apenas o objetivo de manter os corpos rolantes separados em intervalos regulares entre as pistas internas e externas, garantindo a liberdade de movimento, atuando também para evitar que os elementos rolantes se desprendam do rolamento quando manuseados. Elas não são projetadas para suportar carga diretamente. Os modelos de gaiola variam de acordo com o processo de fabricação, podendo ser usinadas, forjadas ou prensadas.

3.2.2 Classificação de Rolamentos

Existem diferentes formas de classificar os rolamentos em grupos, a principal é devido ao tipo de elemento rolante, dividindo-se em rolamentos de esferas e rolamento de rolos.

- Esferas: Podem ser subdivididos de acordo com configuração de seus anéis: rolamento rígido de esferas, de contato angular e axiais.

- Rolos: Podem ser subdivididos de acordo com a forma: cilíndricos, esféricos, cônicos e agulhas.

Pode-se considerar também, a classificação de acordo com a direção de aplicação das cargas:

- Rolamentos Radiais: apresentam ângulo de contato menor que 45° e suportam cargas radiais.
- Rolamentos Axiais: apresentam ângulo de contato maior que 45° e suportam cargas axiais.

Outra característica que pode ser considerada na classificação é o número de pistas de contato: simples, dupla, múltipla.

Exemplos de Rolamentos Radiais:

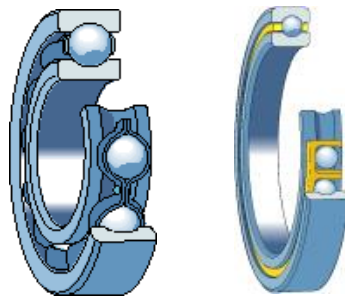


Figura 3.3 - Rolamento Rígido de Esferas (SKF Brasil).



Figura 3.4 - Rolamento Rígido de Esferas com Vedação de Contato (SKF Brasil).

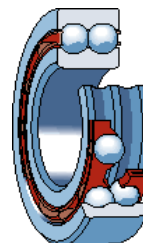


Figura 3.5 - Rolamento Rígido de Esferas de Duas Carreiras (SKF Brasil).

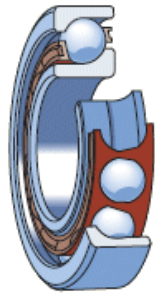


Figura 3.6 – Rolamento de Esferas de Contato Angular (SKF Brasil).



Figura 3.7 – Rolamento de Esferas de Contato com anel interno inteiriço (SKF Brasil).



Figura 3.8 – Rolamento de Agulhas com e sem anel interno (SKF Brasil).

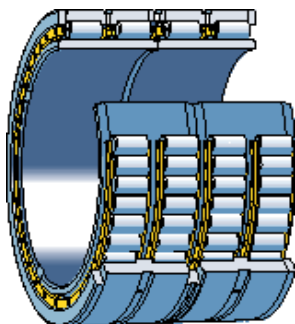


Figura 3.9 – Rolamento de Rolos Cilíndricos com quatro carreiras (SKF Brasil).

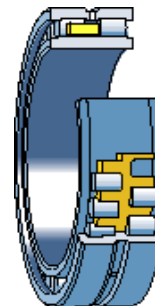


Figura 3.10 – Rolamento de Rolos Cilíndricos com duas carreiras (SKF Brasil).

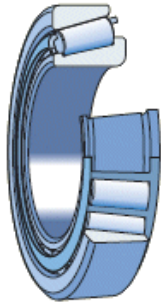


Figura 3.11 – Rolamento de Rolos Cônicos individual (SKF Brasil).

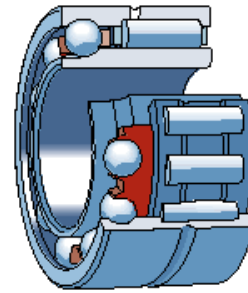


Figura 3.12 – Rolamento de Esferas de contato angular / agulhas (SKF Brasil).

Exemplos de Rolamentos Axiais:

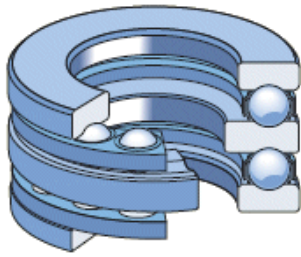


Figura 3.13 – Rolamento Axial de Esfera (SKF Brasil).

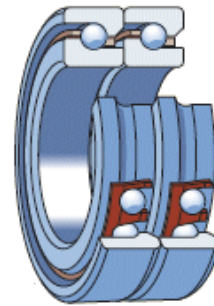


Figura 3.14 – Rolamento Axial de Esferas de contato angular (SKF Brasil).

CAPÍTULO 4

SENSORES

Sensores são equipamentos que permitem medir grandezas físicas, possibilitando a quantização e análise dessas grandezas. Alguns sensores podem converter a energia recebida em outros tipos de energia, gerando sinais diferentes dos recebidos para facilitar a análise. A esse tipo de sensor, denominamos transdutores. Geralmente, os transdutores transformam a energia oriunda de grandezas físicas em sinais de natureza elétrica proporcionais ao parâmetro do sinal original. Estes dispositivos, permitem que sejam realizadas amostragens para que, futuramente, um banco de dados seja formado, possibilitando assim, uma análise satisfatória do que está ocorrendo. Para isso, é importante que o sensor esteja calibrado, bem posicionado e que ele possua um tempo de resposta eficiente para a medição proposta.

Dentre os princípios mais utilizados para formas de medição, encontramos:

- Capacitância
- Magnetismo
- Indução
- Efeito hall
- Piezoeletricidade
- Efeito Piroelétrico
- Resistência Elétrica
- Radiação

Definimos sensor ativo, aquele que necessita adicionar energia ao sistema físico a ser medido, isto é, ele necessita ser alimentado por uma fonte de energia externa. Podemos citar como exemplo o sonar.

Definimos sensor passivo, aquele que provê sua própria energia ou que retira energia do sistema físico ao qual está medindo. Neste caso, é importante ter o cuidado para que os sinais não interfiram no sistema. Podemos citar como exemplo o termopar.

A calibração de sensores, geralmente, consiste em expor o sensor à condições, por exemplo frequências, conhecidas, permitindo desta forma, o alinhamento com os resultados esperados e a determinação da sensibilidade.

4.1 Erros Associados

Para sensores, o conceito de erro associado está relacionado à diferença do real valor do mensurando e o valor medido.

4.1.1 Erros de Inserção

É o erro relacionado com a inserção do sensor no sistema que está sendo medido. Para minimizar erros de inserção é fundamental que o instrumento de medida não interfira no funcionamento do sistema.

4.1.2 Erros de Operação

São erros associados aos procedimentos realizados pelo operador. Dentre eles podemos citar posicionamento do sensor, erro de leitura, entre outros.

4.1.3 Erros Característicos

São erros associados ao sensor, isto é, erros referentes à diferença entre as características da função de transferência do modelo teórico e as características modelo real. Podem incluir problemas de linearidade entre relações, valores de offset, entre outros.

4.1.4 Erros Dinâmicos

Muitos sensores são calibrados em condições estáticas, ou quase estáticas. Devido ao amortecimento, podem apresentar dificuldade em responder a variações rápidas do parâmetro de entrada, podendo resultar em diferenças entre o que está sendo medido e a tensão resultante na saída do sensor. Tal fato, pode ser responsável por erros de tempo de resposta, distorção de amplitude e de fase.

4.1.5 Erros de Ambiente

São erros relacionados, como o nome diz, ao ambiente. Entre os fatores podemos destacar: variação e faixa de temperatura, choques mecânicos, radiação, ação química entre outros.

4.2 Características dos Sensores

Abaixo estão relacionadas algumas características importantes inerentes aos possíveis resultados obtidos a partir de medições utilizando sensores.

4.2.1 Sensibilidade

Por definição, a sensibilidade é a inclinação da curva característica de saída do sensor. Podendo ser descrita como a entrada física mínima capaz de gerar um sinal de saída detectável.

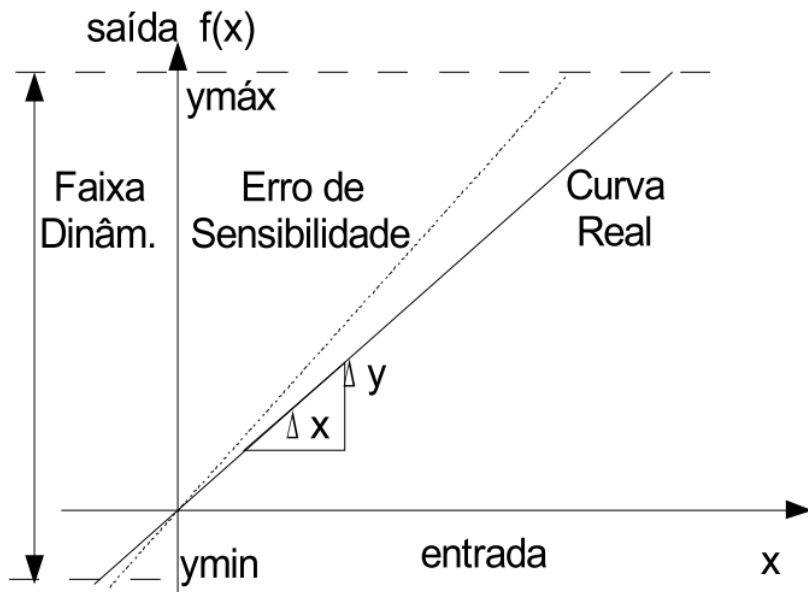


Figura 4.1 – Curva de Sensibilidade.

4.2.2 Erro de Sensibilidade

Existe uma curva de transferência entrada-saída esperada, quando há alterações de inclinação em relação a essa curva ideal, fica caracterizado o erro de sensibilidade.

4.2.3 Faixa (*Range*)

Faixa está relacionada aos valores mínimos e máximos do parâmetro de entrada que um sensor pode medir, não existindo obrigatoriedade desses valores mínimos e máximos serem simétricos em relação a um zero.

Denominamos Faixa Dinâmica, a faixa total de operação do sensor, ou seja, a fatia que compreende entre os valores mínimo e máximo. Podemos representar a faixa dinâmica como:

$$R_{Din} = Y_{máx} - Y_{mín} \quad (4.2)$$

4.2.4 Precisão

Ao conceito de precisão, atribui-se a ideia de realizarmos a medição de um mesmo valor diversas vezes e a saída reproduzida ser sempre a mesma. Isto caracterizaria um sensor ideal. Constituindo assim, uma relação com o conceito de reprodutibilidade de uma medição. Porém, na realidade, os sensores fornecem uma faixa. O que demonstra que os resultados representados estão distribuídos em torno de um valor central, o qual seria o valor correto.

4.2.5 Resolução

Resolução é o menor incremento da variável medida que pode ser detectado no valor de saída do sensor. Geralmente, a resolução pode ser representada como porcentagem da faixa de leitura ou em valores absolutos.

4.2.6 Acurácia

É a máxima diferença que existirá entre o valor verdadeiro a ser medido e o valor de saída do sensor. Geralmente, a acurácia pode ser representada como porcentagem da escala total ou em valores absolutos.

4.2.7 Offset

Quando o configuramos o valor de saída do sensor para zero (nulo) ou algum valor conhecido, porém encontramos um valor de saída diferente do esperado. Denominamos offset a diferença entre o valor de saída medido, real, e o especificado para aquela condição, o qual era o esperado.

4.2.8 Linearidade

Linearidade é caracterizada pelo quanto a curva do sensor se desvia da curva considerada real. Geralmente é apresentada em termos de percentual de não linearidade.

$$\text{Não Linearidade}(\%) = \frac{D_{INMAX}}{IN_{FS}} \cdot 100\% \quad (4.3)$$

D_{INMAX} é o máximo desvio da curva ideal.

IN_{FS} é a entrada máxima, admitida como escala total.

4.2.9 Histerese

É a capacidade do sensor de acompanhar as variações de entrada. Se para um estímulo crescente do sinal de entrada ao passar pelo valor X, o sensor apresenta um determinado valor de saída. E um valor de saída diferente para um estímulo decrescente do sinal de entrada ao passar pelo mesmo valor X, fica caracterizado o erro de histerese.

4.2.10 Tempo de Resposta

Caracteriza o tempo necessário para que após um estímulo na entrada do sensor, a saída altere seu valor para um valor final de acordo com o *range* tolerado em relação ao novo valor correto.

4.3 Acelerômetro

Quando necessitamos de instrumentação adequada para medir vibrações mecânicas, impactos mecânicos, detecção de velocidade, detecção de aceleração, detecção de inclinação, entre outros, o tipo de sensor mais indicado para tal medição é o acelerômetro.

Alguns fatores importantes em medições utilizando acelerômetros giram em torno do seu princípio de funcionamento. Em geral, utilizando uma base que fica em contato com a superfície a qual iremos medir a aceleração. Uma camada com material piezoelétrico conectada a uma massa conhecida. O princípio deste sistema será exemplificado de forma simplificada abaixo:

Temos:

m = massa conhecida

a = aceleração (queremos saber)

F = força

f = força elástica

x = deslocamento conhecido

k = constante conhecida

$F = m \cdot a$

$f = k \cdot x$

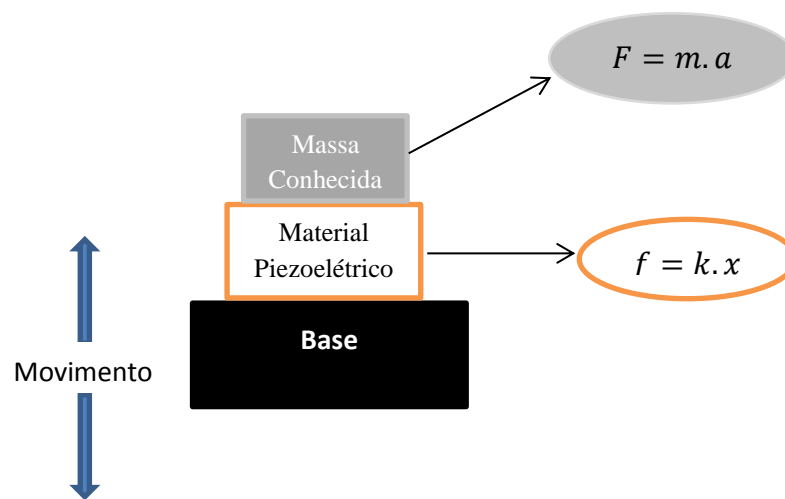


Figura 4.2 – Modelo teórico de um acelerômetro.

Como aceleração é uma grandeza local, precisamos determinar a posição do acelerômetro de acordo com o modo de vibração, de forma a potencializar da melhor forma possível o sinal de entrada.

A direção de posicionamento na instalação é outro fator relacionado ao princípio de funcionamento dos acelerômetros, que influencia as medições realizadas por estes dispositivos.

Existem outros modelos de acelerômetros que utilizam outros princípios para realizar medições. Como por exemplo, o Servo acelerômetro, que faz uso de um sistema semelhante a um pêndulo para medir inclinações de um eixo. É comum encontrar estes

modelos em sistemas de posicionamento de mísseis e em sistemas de perfuração direcional.

CAPÍTULO 5

REDES NEURAIS ARTIFICIAIS

Este capítulo introduz de forma resumida o conceito de Rede Neural Artificial, os processos para o desenvolvimento desse tipo de inteligência artificial e seus componentes. É apresentada a ideia de como a rede pode ser configurada, assim como os conceitos de neurônios e camadas ocultas.

5.1 O conceito de Rede Neural Artificial

Rede Neural Artificial é um tipo de sistema computacional, que teve sua arquitetura baseada no funcionamento de células biológicas, denominadas neurônios. Uma tentativa de reconstituir o processamento, armazenamento e aprendizado realizado por estas células encontradas no cérebro humano, as quais estão conectadas umas as outras formando uma malha conhecida como Rede Neural. Os neurônios são capazes de transmitir estímulos através de interações químicas, chamadas sinapses. Ao longo do tempo, os modelos propostos de redes neurais artificiais buscaram uma aproximação com o modelo natural através de conexões com funções similares às sinapses. Com isto, torna-se necessário o fator experiência. Ou seja, este modelo computacional precisa ser treinado, processando dados, errando e aprendendo até que a avaliação feita sobre seu rendimento seja conclusiva. Sendo por vezes, aceitável para a solução de problemas específicos, e outras vezes, considerados inadequados. Neste último, deve-se repensar o modelo computacional de forma a adequar-se melhor ao caso proposto.

Os processos de desenvolvimento de redes neurais artificiais costumam apresentar o seguinte formato:

1. Obtenção de dados
2. Desenvolver a rede
3. Configurar a rede

4. Introduzir os pesos e biases
5. Treinar a rede
6. Validar a rede, avaliação dos resultados dos treinos
7. Usar a rede

5.2 Modelos de Neurônios.

Neurônios, ou nós, podem ser considerados elementos de processamento interconectados. Os pesos associados à saída de um neurônio estão relacionados à memória que armazena o conhecimento adquirido durante o treinamento.

5.2.1 Modelo de Neurônio Simples

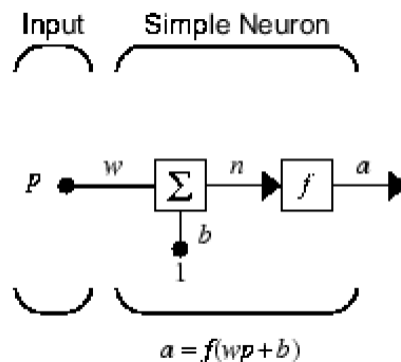


Figura 5.1 - Modelo de Neurônio Simples. (BEALE, M.H. *et al*, 2011)

Acima, temos o exemplo da estrutura encontrada em um neurônio simples. A forma como a informação é tratada neste tipo de neurônio permite distinguir três diferentes operações: função peso (*weight function*), função entrada de rede (*net input function*) e função de transferência (*transfer function*).

A entrada escalar p é multiplicada pelo peso escalar w , resultando no produto wp (função peso). Em seguida, a entrada já ponderada pelo peso, wp , é adicionada ao bias escalar b , que tem o objetivo de aumentar a capacidade de adaptação ao conhecimento fornecido a rede. O bias, ou entrada de polarização, atua de forma a aumentar ou

diminuir a entrada da função de transferência f . Desta forma é gerada a entrada da malha n (função entrada de rede). A entrada da malha então passa por uma função de transferência f , produzindo uma saída a .

O formato como a função peso se apresenta pode variar de configuração para configuração. Podemos encontrar informações mais detalhadas em “*Transfer Functions*” no *Neural Network Toolbox Reference* para MATLAB.

5.2.2 Função de Transferência

Como vimos no capítulo anterior, a função de transferência define a saída de um neurônio. Algumas funções de transferência f bastante utilizadas estão representadas abaixo.

- Função de Transferência Linear: geralmente são utilizadas como funções de aproximação e encontradas na última camada de redes neurais com múltiplas camadas.

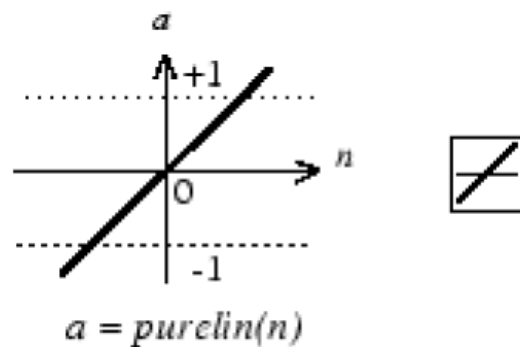


Figura. 5.2 - Função de Transferência Linear (BEALE, M.H. *et al*, 2011).

- Função Log-Sigmoide: Aceita valores de entrada entre mais e menos infinito, e gera saída com valores definidos entre 0 e 1. Por ser diferenciável, geralmente é encontrada em camadas escondidas, ou *hidden layers*, de redes multicamadas. É uma das mais utilizadas em redes neurais.

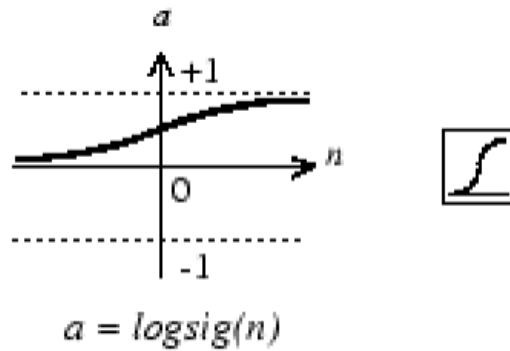


Figura 5.3 - Função de Transferência Log-Sigmoide (BEALE, M.H. *et al*, 2011).

- Função Tan-Sigmoide: É considerada uma alternativa para o uso de funções sigmoides, resultando em saídas com valores entre -1 e 1.

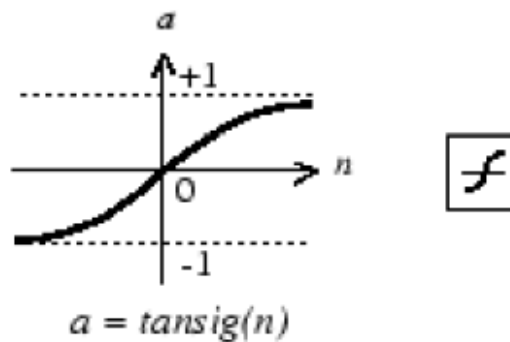


Figura 5.4 - Função de Transferência Tan-Sigmoide (BEALE, M.H. *et al*, 2011).

De forma generalizada, as saídas resultantes de funções sigmoides são indicadas para problemas de reconhecimento de padrão.

5.2.3 Modelo de Neurônio com Vetor Input

Um neurônio pode apresentar múltiplas entradas, na forma de vetor input. Este vetor é formado por elementos individuais.

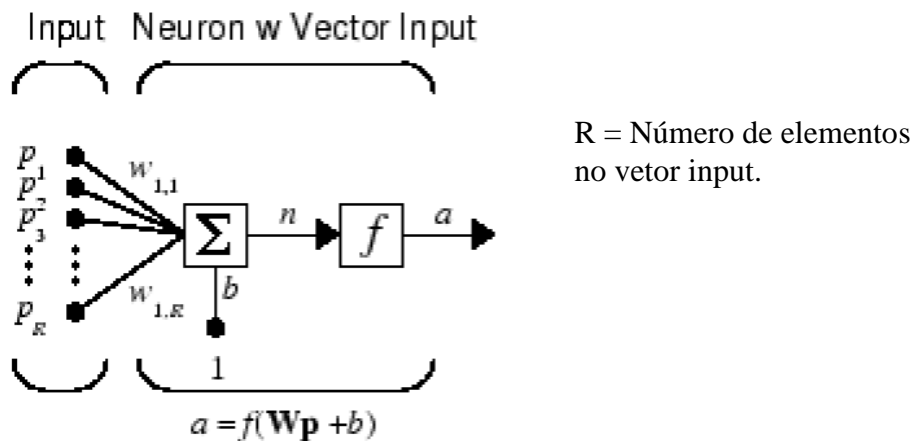


Figura 5.5 – Exemplo de Neurônio com Vetor Input (BEALE, M.H. *et al*, 2011).

O mais usual é utilizar uma notação simplificada. Onde os elementos $p_1, p_2, p_3 \dots p_R$, formam o vetor \mathbf{p} composto por R elementos.

Os pesos $w_{1,1}, w_{1,2}, w_{1,3} \dots w_{1,R}$, formam a matriz com uma coluna \mathbf{W} .

E novamente o produto $\mathbf{W}\mathbf{p}$ será somado ao bias b multiplicada por uma constante 1. Gerando a função net, ou entrada n .

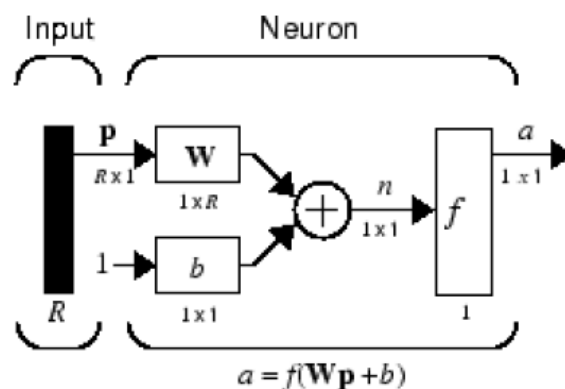


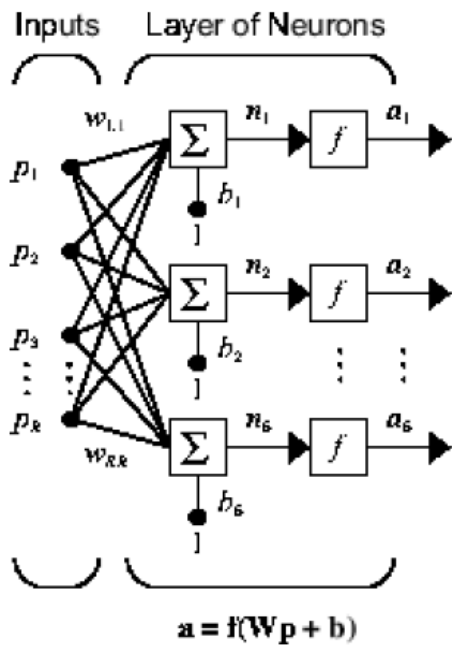
Figura 5.6 – Exemplo de Neurônio com Vetor Input (BEALE, M.H. *et al*, 2011).

Em seguida, n é submetida à aplicação da função de transferência f , originando a saída a .

5.3 Camadas de neurônios

Os neurônios podem ser combinados em camadas, as redes neurais por sua vez, podem ser compostas por várias camadas de neurônios.

5.3.1 Camadas Simples de Neurônios

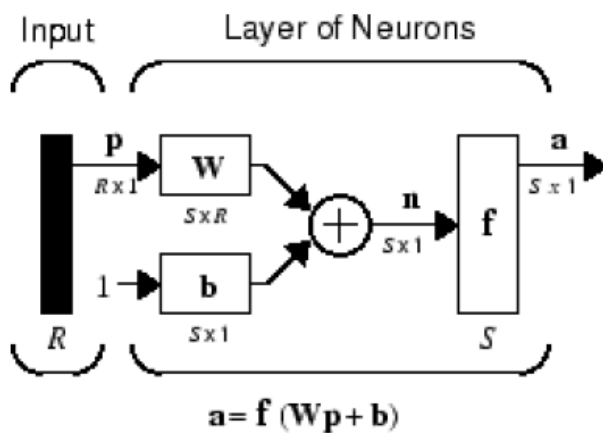


R = Número de elementos no vetor input.

S = Número de neurônios na camada.

Figura 5.7 – Exemplo de camada única de neurônios (BEALE, M.H. *et al*, 2011).

Ou pela notação simplificada,



R = Número de elementos no vetor input.

S = Número de neurônios na camada.

Figura 5.8 – Exemplo de camada única de neurônios (BEALE, M.H. *et al*, 2011).

Neste modelo, \mathbf{p} é o vetor input com extensão R. \mathbf{W} é a matriz SXR e \mathbf{a} e \mathbf{b} são vetores com extensão S. A camada possui uma matriz peso (weight) \mathbf{W} , que multiplica o vetor input \mathbf{p} , o vetor bias \mathbf{b} , a soma e a função de transferência.

5.3.2 Camadas Múltiplas de Neurônios

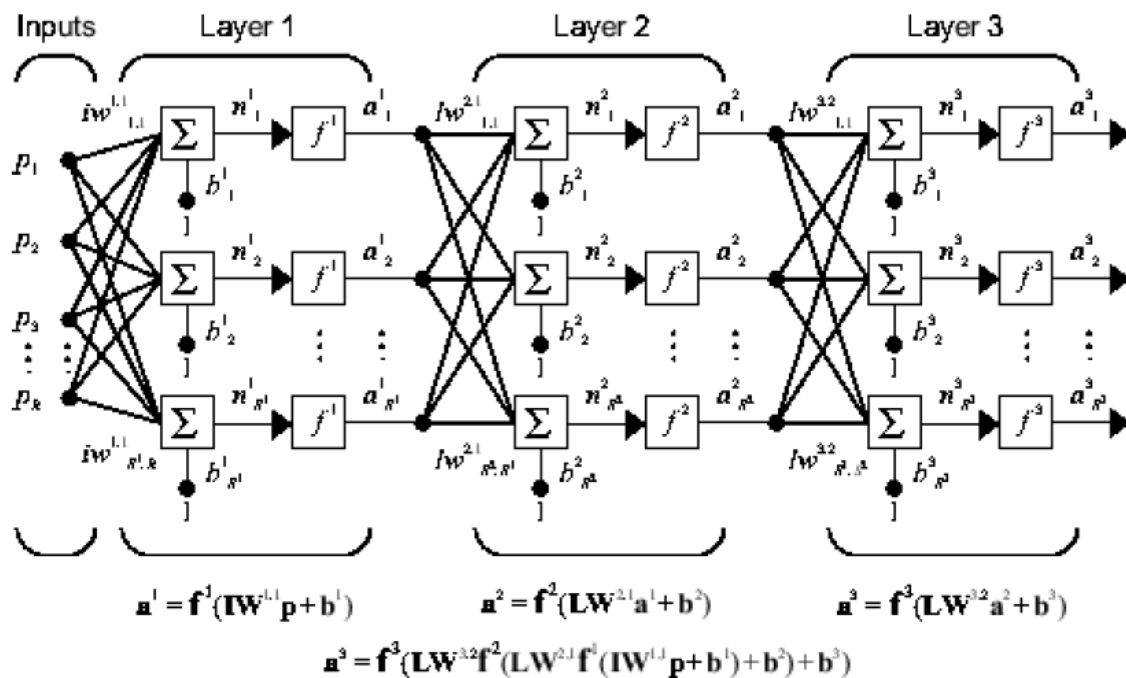


Figura 5.9 – Exemplo de camada múltipla de neurônios (BEALE, M.H. *et al*, 2011).

Acima, observamos o exemplo de uma rede de múltiplas camadas, no caso, três camadas. Poderíamos identificá-las como:

- Camada de entrada: Responsável por receber as entradas (*inputs*), dados ou padrões;
- Camada intermediária: Processa os dados e é responsável pela extração de características que possibilitem a classificação;
- Camada de saída: Responsável pela conclusão da análise e apresentação do resultado.

Podemos observar que é fundamental a interação entre as camadas e que, no caso de redes *feedforward*, neurônios de uma mesma camada não interagem entre eles.

Encontramos referências com nomenclaturas para camadas intermediárias também como *hidden layers*, camadas escondidas ou ocultas.

5.4 Redes Neurais *Feedforward* (Alimentação Direta)

Algumas das características das redes *feedforward*, é a presença de uma camada de entrada, uma camada de saída e uma ou mais camadas ocultas (*hidden layers*) com neurônios apresentando funções de transferência sigmoide. Isto permite que a rede obtenha um desempenho melhor no aprendizado sobre a relação entre os vetores input e output (entrada e saída). A camada de saída, pode ser linear ou não-linear. No trabalho, o foco é o reconhecimento de padrões, por isto, na camada de saída foi introduzida uma função de transferência sigmoide, com objetivo de restringir nossos resultados entre 0 e 1.

CAPÍTULO 6

REDES NEURAIS ARTIFICIAIS NO MODELO

EXPERIMENTAL

Neste capítulo será demonstrado como foi realizada a aquisição de dados de vibração no modelo experimental proposto. O tratamento desses dados, condicionado a escolhas de características de amostras no domínio do tempo. A introdução destes dados como entrada da rede neural artificial e, como saída, a condição do rolamento.

6.1 Domínio do Tempo

Análise de sinais no domínio do tempo pode ser considerada como toda forma de análise aplicada diretamente em sinais medidos no tempo e permite obter subsídios para uma avaliação global da condição funcionamento de uma máquina.

Para criar uma associação entre as condições padrão e a situação em que há evidências de uma possível falha, serão utilizados métodos estatísticos para a análise. Qualquer alteração na distribuição de probabilidade pode representar um indicativo de uma possível falha.

Segundo SAMANTA, B. *et al.* (2003) e PACHECO, W. S. (2012), os quatro primeiros momentos da distribuição de probabilidade, média, variância, desvio padrão, assimetria, curtose, em conjunto com RMS, podem ser usados no diagnóstico.

6.2 Modelo Experimental

O modelo experimental escolhido é formado por um sistema constituído por uma máquina rotativa, elaborada para realizar ensaios controlados para um outro experimento, onde o objetivo era monitorar as alterações no campo de pressão sonora emitido. E, através de análises dos sinais, avaliar as condições da máquina ao longo do tempo. Tal equipamento encontra-se no LAVI/COPPE, e foi idealizado por Wallace de

Souza Pacheco para Tese de Doutorado *Reconhecimento de Padrões de Operação de Máquinas utilizando técnicas de identificação de fontes sonoras* (2012). A máquina foi escolhida para ser usada como modelo experimental, pois permite alterações em suas configurações de forma a simular defeitos conhecidos. Permitindo que sejam realizadas medições em condições padrão de funcionamento e em condições fora dos padrões, ou situações de falha.

O sistema mecânico é composto por um motor elétrico, um eixo principal acoplado ao motor elétrico e um segundo eixo, mais curto, paralelo ao eixo principal. Cada eixo está apoiado sobre dois mancais, ambos semelhantes. O eixo principal possui dois discos acoplados, fixados por parafuso, para que seja possível variar suas condições de funcionamento, além de uma engrenagem que o acopla ao segundo eixo, também fixada por parafusos. O eixo mais curto possui, além da engrenagem que o acopla ao eixo principal, uma segunda engrenagem, ambas também fixadas por parafusos, para permitir diferentes configurações.

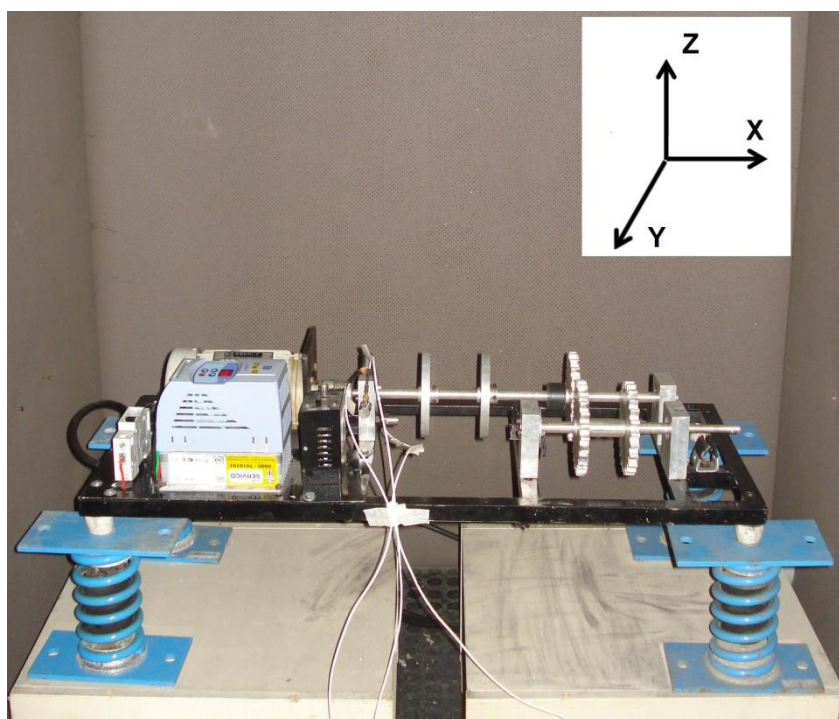


Figura 6.1 – Modelo Experimental.

Existe ainda a possibilidade de adaptar um sistema de frenagem na máquina para criar carga e variar suas condições de funcionamento.

Os eixos são fabricados em aço ABNT 1045 e possuem 13 mm de diâmetro.

O sistema é movido por um motor de indução com as seguintes características:

Tabela 6.1 – Especificação Motor Elétrico

Características	Motor Elétrico
Fabricante	Eberle
Tipo	Trifásico
Rotação	3345 rpm
Potência	$\frac{1}{4}$ CV

Para que fosse possível variar a frequência de rotação do motor, foi utilizado um inversor de frequência do tipo CFW 10, monofásico, 110 Volts, 50-60 Hz, de 1 a 7,1 A, o que possibilitou variações de rotação do motor de até 60 Hz.

Todos os quatro rolamentos são do tipo 6000 2RSC3V2.

Tabela 6.2 – Especificação do Rolamento

Basic Bearing No.	Bore d mm	Outer Diameter D mm	Width W mm	Radius r min mm	Weight kg
6000	10	26	8	0.3	0.019
Basic Load Rating KN		Ball Complement		Max Runout Speed	
Dynamic C	Static C0	No.	Size mm	Grease r/min	Oil r/min
4.55	1.96	7	4.763	20000	28000

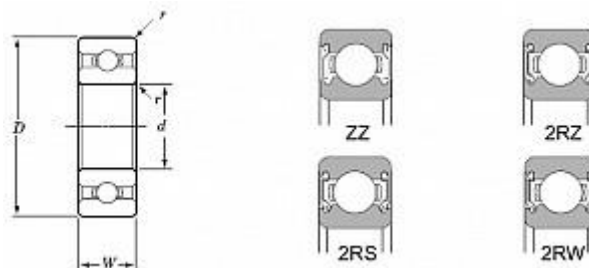


Figura 6.2– Especificação do Rolamento (FK Bearing Group).

Para que fosse possível medir a rotação do motor, um tacômetro foi construído utilizando um sensor óptico e uma peça cilíndrica com uma tira de cor metálica, e instalado no eixo principal.

6.3 Aquisição de dados

No modelo de estudo, os dados de vibração foram obtidos através de acelerômetros modelo PCB 353B03, de propósito geral, sensibilidade de $10\text{mV}/(\text{m}/\text{s}^2)$, frequência de 1 a 7 KHz, 10-32 conector lateral, posicionados no mancal 1. Com o auxílio do *software* LabVIEW oferecido pela plataforma National Instruments, foi possível, utilizando o segmento Signal Express, a obtenção e visualização das amostras coletadas. O Signal Express também permitiu a exportação das informações para um formato que pudesse ser acessado pelo algoritmo em linguagem MATLAB. A figura 6.3 mostra a distribuição nominal dos mancais. As figuras 6.4 e 6.5 mostram o posicionamento dos acelerômetros instalados no mancal 1. E as figuras 6.6 e 6.7, a distribuição da aceleração pelo tempo visualizada através do Signal Express.

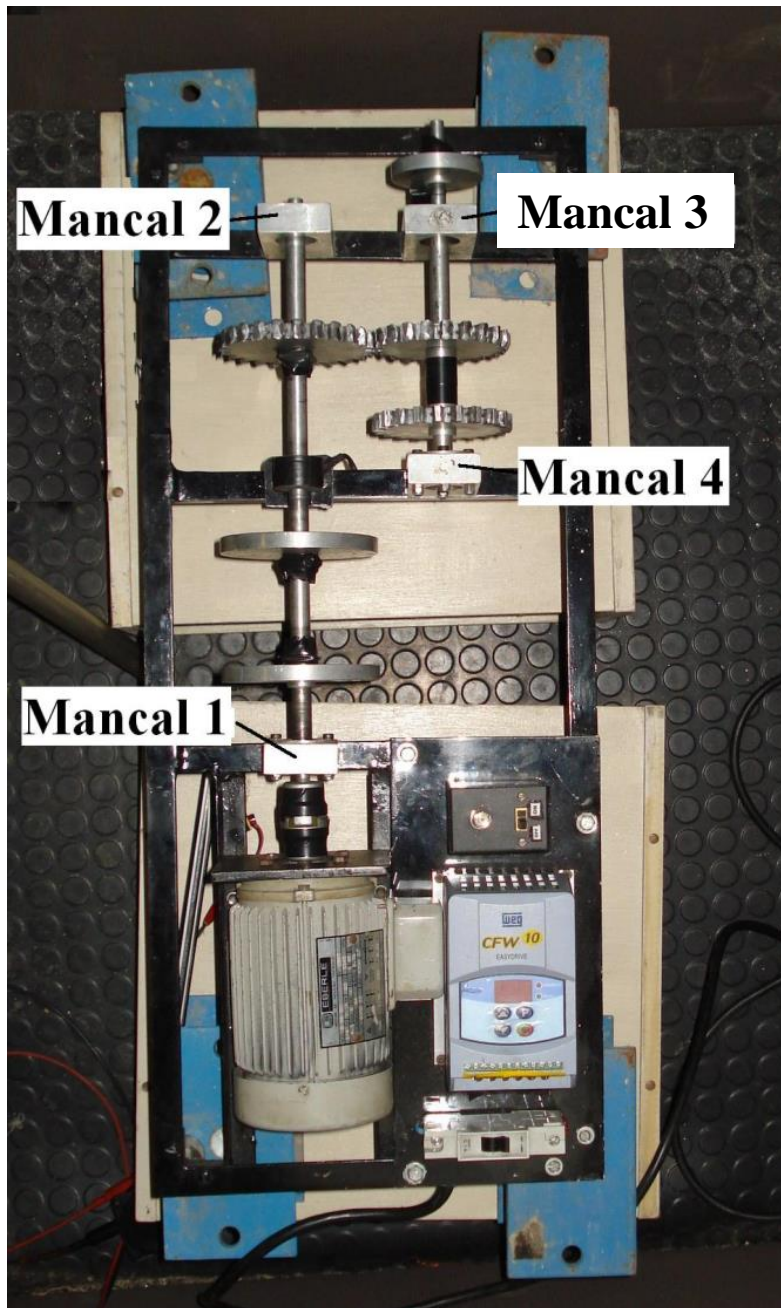


Figura 6.3 – Arranjo dos mancais.

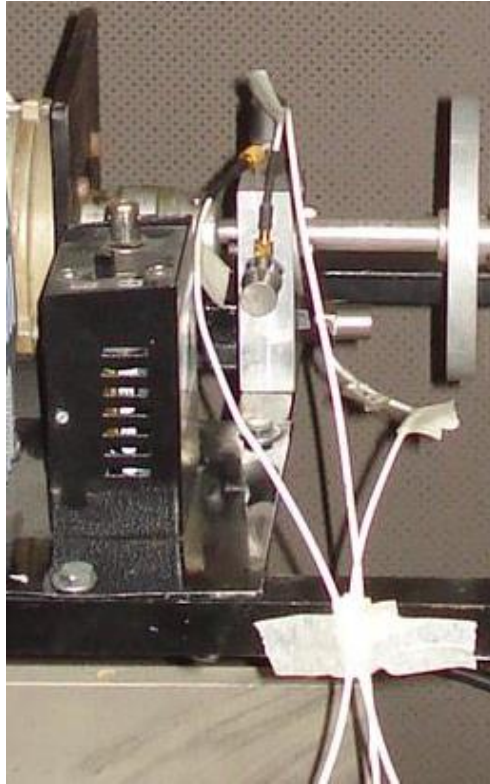


Figura 6.4 – Arranjo dos acelerômetros no modelo.

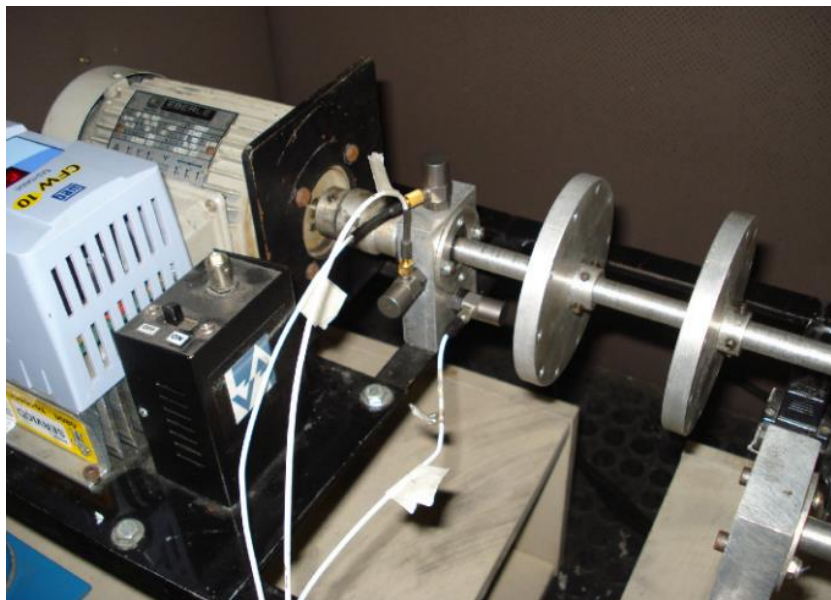


Figura 6.5 – Arranjo dos acelerômetros no modelo.

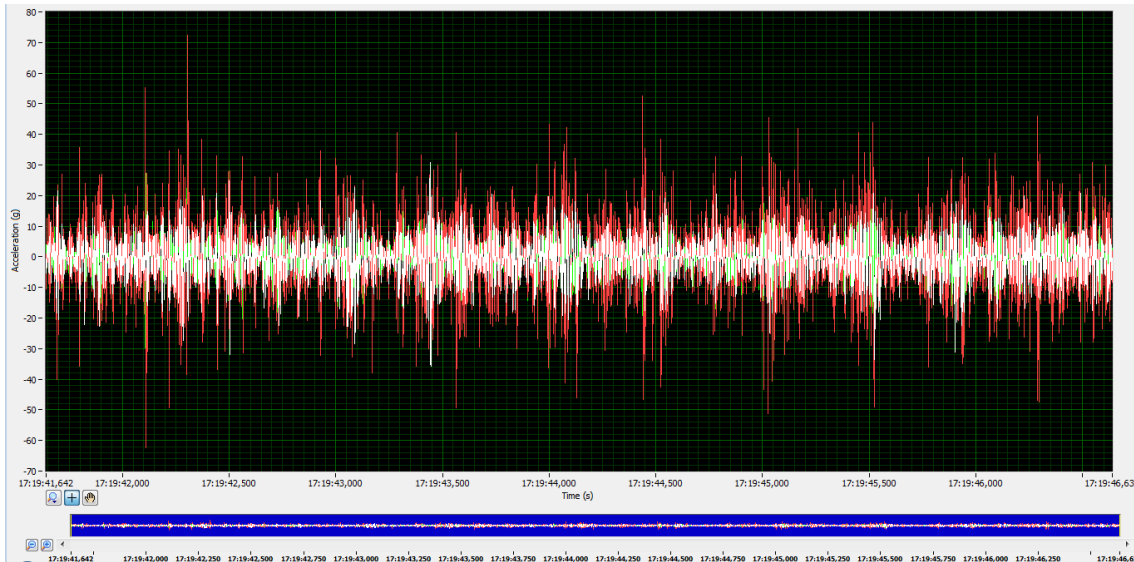


Figura 6.6 – Assinatura observada pelo Signal Express, 32Hz – Rolamento Padrão.

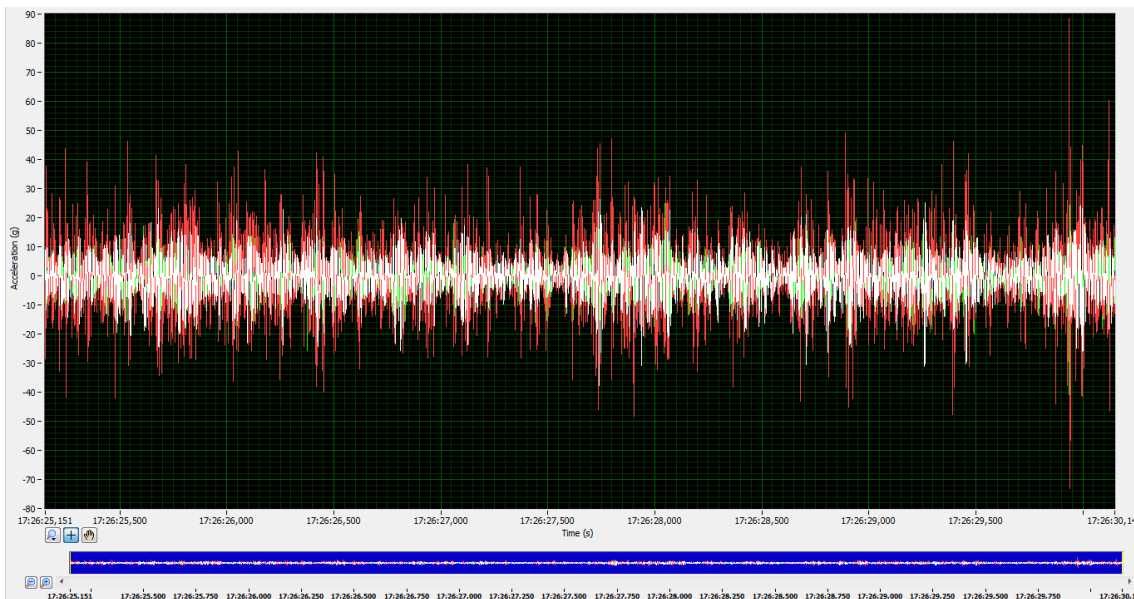


Figura 6.7 – Assinatura observada pelo Signal Express, 32Hz – Rolamento Defeituoso.

Os sinais de aceleração nas direções x, y e z, foram adquiridos no domínio do tempo, duração 5s e frequência de aquisição de 20KHz. Seguindo as duas equações abaixo, chegamos à amostra:

$$freq = \frac{1}{\Delta t} \quad (6.1)$$

$$T = m\Delta t \quad (6.2)$$

Onde:

$freq$ = frequência

Δt = período

T = tempo de medição

m = amostra

$$20000 = \frac{1}{\Delta t}$$

$$m = \frac{T}{\frac{1}{freq}}$$

$$m = 5.20000 = 100000$$

Nesta proposta, as medições foram realizadas com a bancada funcionando sem carga, perante duas condições de rolamentos: funcionando de forma padrão e funcionando de forma fora do padrão ou defeituoso. No segundo caso, o rolamento que funcionava de forma padrão foi substituído por um rolamento com defeito pista interna. Foram experimentadas três velocidades diferentes do motor: 10 Hz, 23 Hz, e 32 Hz. A velocidade escolhida para treinar, validar e aplicar a rede neural artificial foi a de 32 Hz. É importante destacar que podem existir outras fontes de alterações nos sinais que não sejam inerentes aos rolamentos, e sim devido ao manuseio e configuração da bancada.

6.4 Tratamento dos dados

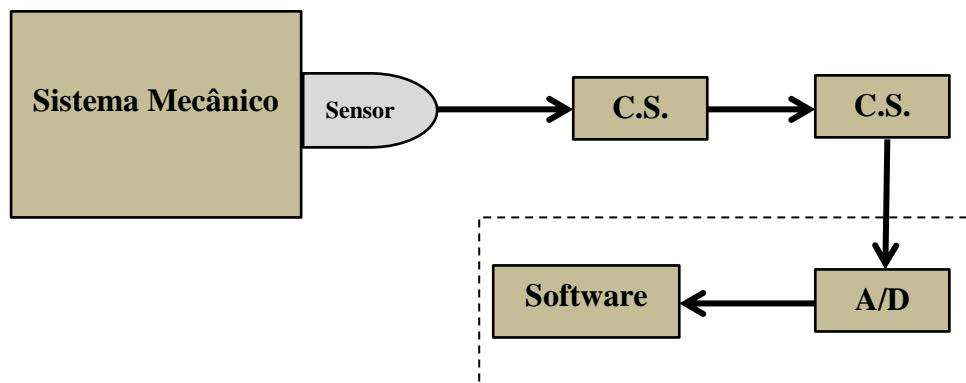


Figura 6.7 – Arranjo usual de tomada de dados.

Onde:

C.S. = Conversor de sinal

A/D = Conversor Analógico/Digital

Com a frequência de aquisição de 20 KHz, foram obtidos cem mil amostras, as quais foram divididas em pacotes de dados menores, compostos por cem amostras. Para uma representatividade de maior confiabilidade, foi utilizada uma sobreposição (overlapping) de cinquenta amostras para cada pacote de dados. Ou seja, o contador parte da amostra $i=1$ até $i=100$, posteriormente para o próximo cálculo, o contador retoma a partir da amostra $i=51$ prosseguindo até $i=150$. E assim sucessivamente.

As características escolhidas para a avaliação foram:

- Média;
- Variância;
- Desvio Padrão;
- Assimetria;
- Curtose;
- RMS (*Root Mean Square*).

Tais características estatísticas foram extraídas dos dados coletados pelos três acelerômetros, nas direções x, y, z, com auxílio de um algoritmo em linguagem de software MATLAB. Existem seis características para cada direção, desta forma o vetor input, quando considerando os três eixos coordenados, irá apresentar dezoito características de entrada de rede. Espera-se que tais propriedades sejam suficientes para que a rede neural artificial seja capaz de distinguir entre rolamentos normais e rolamentos defeituosos.

Baseado em SAMANTA, B. *et al.* (2003) / *Engineering Applications of Artificial Intelligence*, o qual propõem um diagrama sequencial para processo, foi estipulado o seguinte procedimento:

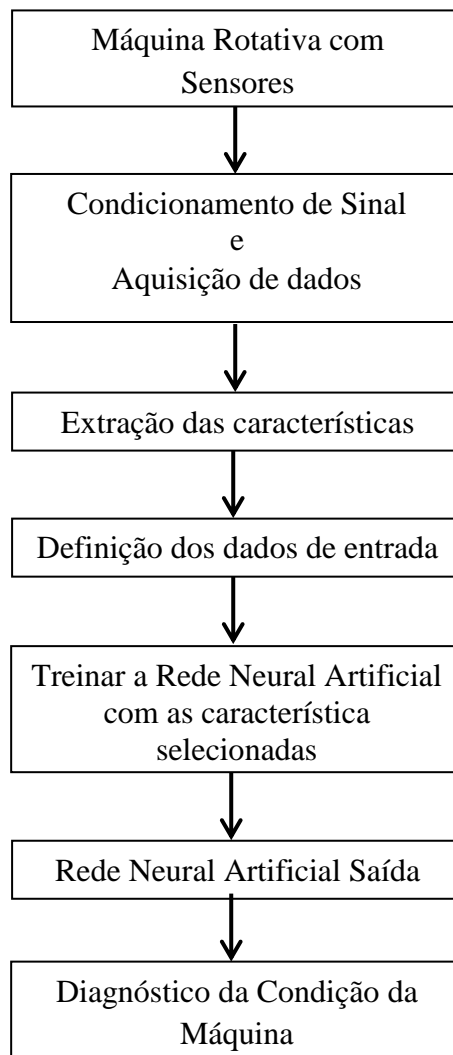


Figura 6.8 – Fluxograma Geral de Rede Neural.

CAPÍTULO 7

VARIAÇÕES DAS CONFIGURAÇÕES DA REDE NEURAL

7.1 Aprendizagem

Podemos considerar aprendizagem, o processo pelo qual ajustes preestabelecidos por algoritmos, chamados algoritmos de aprendizagem, atuam na modificação dos parâmetros da rede. As variações nas configurações de tais parâmetros, generalizando, bias e pesos, são responsáveis por distinguir algoritmos de aprendizagem.

A aprendizagem é um processo pelo qual os parâmetros livres, os pesos e as entradas de polarização de uma rede neural, são adaptados através de um processo de estimulação pelo ambiente no qual a rede está inserida (HAYKIN, 1999). O tipo de aprendizagem é determinado pela maneira pela qual a modificação dos parâmetros ocorre.

No início, tais parâmetros apresentam valores randômicos, que serão ajustados ao longo do processo de aprendizagem.

7.1.1 Rede Neural Artificial Supervisionada

Entre os processos de aprendizagem, o utilizado neste modelo foi o de rede neural supervisionada. Neste tipo, existe uma referência que representa qual é a resposta esperada. A partir disto, quando introduzimos um conjunto de dados de entrada na rede, a saída é comparada com a referência, permitindo a distinção sobre o resultado da saída entre certo ou errado. Caso esteja errado, é possível que a rede ajuste seus parâmetros, possibilitando um treinamento eficiente. Este tipo de rede é amplamente utilizado para reconhecimento de padrões.

7.1.2 Retropropagação (*Backpropagation*)

A técnica de retropropagação, utiliza a diferença entre a saída esperada e a saída resultante para ajustar os pesos. Ou seja, em um primeiro momento, os pesos são mantidos fixos enquanto o sinal de entrada é propagado. Em seguida, através da diferença entre a saída esperada e saída obtida, é gerado um erro. Os pesos são ajustados quando este sinal de erro é propagado no mesmo neurônio, porém no sentido contrário.

Este processo é repetido até que o resultado da saída convirja para saída previamente estabelecida, o que significa que o erro resultante atingiu ou é menor que o erro aceitável. Outra condição possível seria a de que o processo é repetido até que o número limite de iterações seja atingido.

7.2 RNA Proposta

Para o trabalho, foi proposto uma rede neural artificial multicamadas, com alimentação direta (*feedforward*) e retropropagação (*backpropagation*).

Os procedimentos apresentados abaixo são comumente utilizados para o treinamento deste tipo de rede e costumam apresentar boa resposta. Segundo PACHECO, W. S. (2012), o algoritmo padrão de uma Rede Neural Multicamada segue os seguintes passos:

- 1) A rede é iniciada aleatoriamente (pesos são escolhidos de forma aleatória);
- 2) Um conjunto de dados para o treinamento é apresentado à rede e o erro entre o resultado da simulação e os dados reais é calculado;
- 3) O erro resultante é propagado, ajustando os pesos de forma a minimizar o erro global;
- 4) Uma nova simulação ocorre e o erro é calculado e propagado novamente;
- 5) O processo repete-se até atingir um critério de parada.

A figura 7.1, mostra um fluxograma dos passos apresentados acima para o algoritmo genérico da rede.

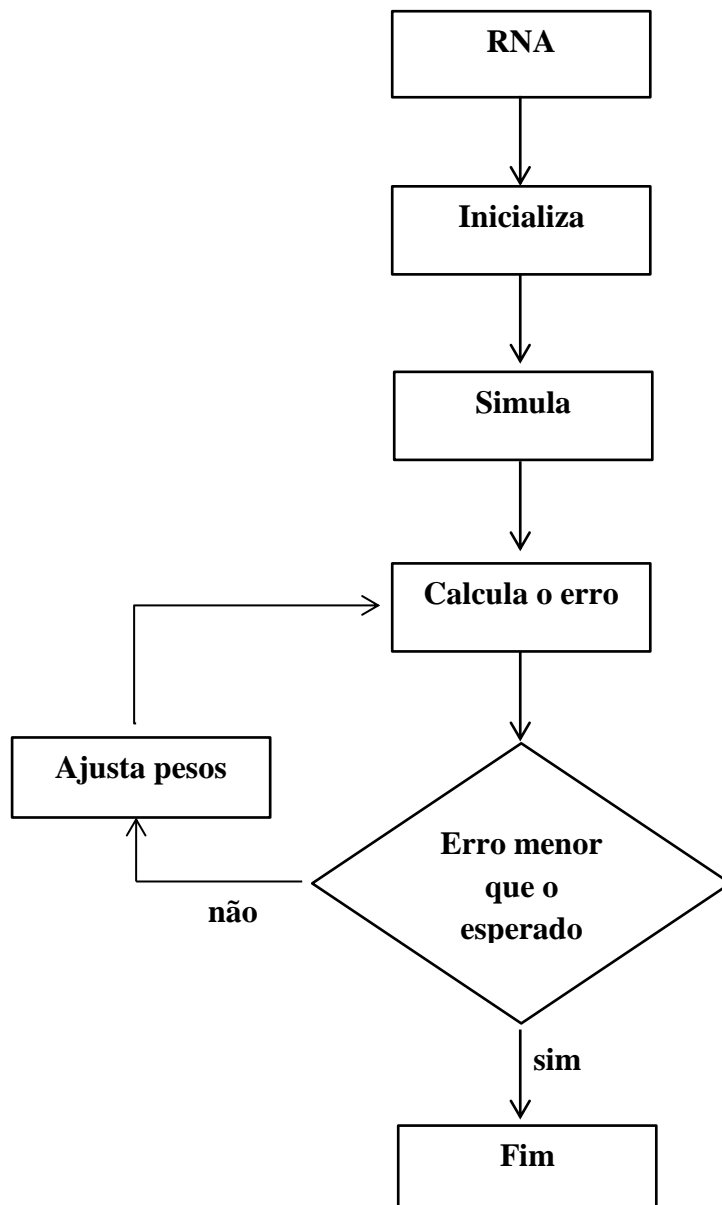


Figura 7.1 – Fluxograma de um algoritmo genérico de uma Rede Neural Multicamada

7.2.1 Configurações Iniciais

A rede neural proposta (multicamada, de alimentação direta, com retropropagação de erro), foi criada e treinada com o auxílio do *toolbox* de rede neurais do programa MATLAB. Os pesos e bias iniciais foram fornecidos pelo MATLAB. Nos capítulos a seguir, serão demonstrados os efeitos da variação de alguns parâmetros em busca da configuração que nos resulte uma resposta satisfatória. Dentre os fatores possíveis, serão considerados:

- Número de neurônios das camadas ocultas;
- Número de camadas ocultas;
- Método de otimização.

A divisão dos dados utilizada foi de 70% para treinamento, 15% para validação e 15% para teste. Tais representações podem ser alteradas buscando uma melhor eficiência.

O *toolbox* do MATLAB oferece quatro funções diferentes para a divisão desses dados. Dentre elas, podemos destacar a opção de dividir randomicamente esses dados ou dividir em blocos permanentes. No último caso, os três blocos (treinamento, validação e teste), terão sempre a mesma fração do conjunto de dados originais. Para que a rede aprendesse com os dados, foi utilizado a função randômica.

Como critério de parada foi utilizado a função de performance MSE (*Mean Square Erro*). A quantidade de 1000 épocas, que representa o número máximo de iterações. Quando a quantidade de iterações atingir 1000, então o treino irá parar. O gradiente de performance, quando sua magnitude for menor que $1,00e^{-05}$. Número de verificações de validação, representa o número de iterações sucessivas em que o performance de validação não consegue diminuir. O número de verificação de validação utilizado foi 6. Quando atingido qualquer um dos critérios acima, o treinamento irá parar.

A porcentagem de acerto representa a porcentagem de classificações efetuadas com sucesso, e pode ser considerada um indicador de desempenho. Usaremos este critério

para determinar a configuração que apresenta o desempenho mais satisfatório na classificação entre rolamentos com defeito e rolamentos normais.

7.3 Quantidade de Neurônios e Quantidade de camadas ocultas

Com o objetivo de avaliar a melhor configuração para rede, foram testados diferentes quantidades de camadas ocultas e quantidades de neurônios nas camadas ocultas.

De forma generalizada, o *Neural Network Toolbox User's Guide* para MATLAB alerta que o padrão inicial do *toolbox* para a quantidade de neurônios nas camadas ocultas é de 10 neurônios. Contudo, aumentar a quantidade de neurônios tende a aumentar a flexibilidade da rede. Para reconhecimento de padrões, uma única camada oculta costuma apresentar resultados satisfatórios. Porém, caso o resultado não seja o esperado, duas ou mais camadas podem atender a necessidade. O número de neurônios da camada oculta está relacionado ao poder da rede, isto significa que aumentando o número de neurônios também aumenta a demanda por capacidade computacional. Além de aumentar a probabilidade da ocorrência de *overfitting*, que pode culminar em resultados inconclusivos.

Neste primeiro momento, foi utilizado o método de Levenberg-Marquardt, por ser considerado o que oferece a resposta satisfatória em menor tempo, e foi variado o número de neurônios na camada oculta.

Tabela 7.1 – Arquitetura utilizando Levenberg-Marquardt com apenas uma camada oculta.

Arquitetura	Taxa de Acerto (%)
18x5x2	95.2226
18x8x2	95.3977
18x9x2	95.4977
18x10x2	95.6478
18x11x2	95.6728
18x12x2	94.8974
18x13x2	95.5478

18x14x2	95.7229
18x15x2	95.8979
18x18x2	95.8479
18x20x2	95.0725
18x21x2	95.6228
18x22x2	96.0730
18x23x2	94.9225
18x24x2	95.3477
18x25x2	94.9475
18x27x2	95.4477
18x28x2	95.0975
18x29x2	95.7729
18x30x2	95.2726

Como podemos observar o neurônio ótimo para a situação foi encontrado na configuração 18x22x2, isto significa que nessa configuração a rede foi capaz de classificar a condição da máquina com eficiência de 96.0730%.

A seguir, foram testadas configurações com duas camadas ocultas.

Tabela 7.2 - Arquitetura utilizando Levenberg-Marquardt com duas camadas ocultas.

Arquitetura	Taxa de Acerto (%)
18x6x1x2	95.3477
18x9x2x2	95.6728
18x8x3x2	95.6978
18x9x4x2	95.4727
18x5x5x2	95.8979
18x8x6x2	95.6978
18x10x7x2	95.3977
18x4x8x2	95.6478
18x9x9x2	95.7229
18x10x10x2	95.5728

Pela tabela 7.2, podemos observar que a configuração que apresentou maior capacidade de classificar a condição da máquina foi a 18x5x5x2, com 95.8979% de eficiência.

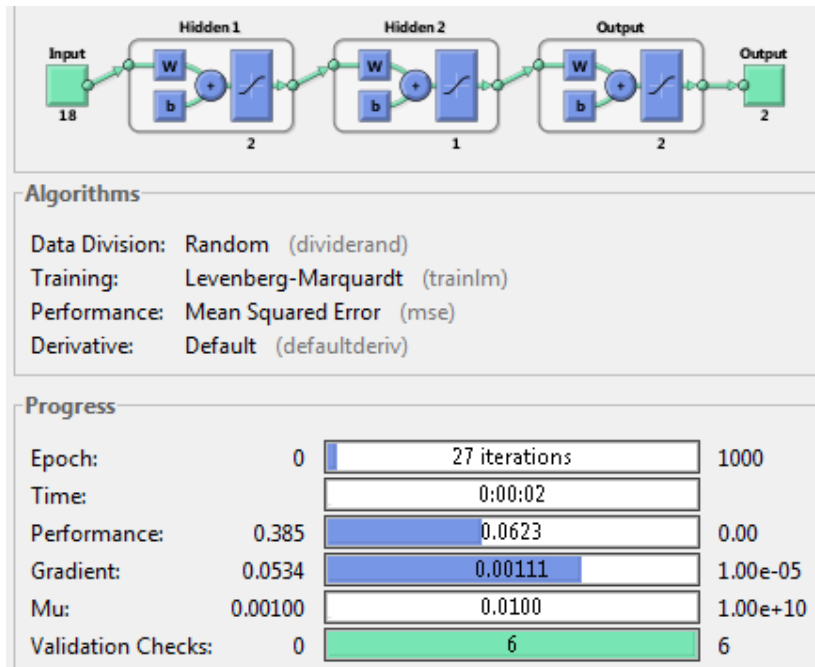


Figura 7.2 – Processo de treinamento

7.4 Variações do Método

Após a avaliação da capacidade de classificação da rede pelo método indicado como mais rápido, Levenberg-Marquardt, neste segmento será apresentado os resultados das arquiteturas quando aplicados os métodos BFGS Quasi-Newton e outros dois métodos indicados para reconhecimento de padrões pelo *Neural Network Toolbox User's Guide*.

7.4.1 BFGS Quasi-Newton

Tabela 7.3 - Arquitetura utilizando BFGS Quasi-Newton com apenas uma camada oculta.

Arquitetura	Taxa de Acerto (%)
18x2x2	92.6463
18x3x2	95.1226
18x4x2	95.2476
18x5x2	94.2221
18x6x2	94.9725
18x7x2	93.3967
18x8x2	94.6473
18x9x2	95.0225
18x10x2	94.3222
18x12x2	94.8974
18x14x2	94.6473
18x19x2	91.2956
18x20x2	93.1216
18x21x2	94.5973
18x22x2	94.1971
18x23x2	94.3222
18x25x2	93.8969
18x26x2	93.4967
18x29x2	95.1226
18x30x2	94.8974

Configuração com maior eficiência na classificação da condição da máquina 18x4x2, com 95.2476%.

Tabela 7.4 - Arquitetura utilizando BFGS Quasi-Newton com duas camadas ocultas.

Arquitetura	Taxa de Acerto (%)
18x5x1x2	95.4477
18x3x2x2	95.2476
18x1x3x2	95.0975
18x4x4x2	95.4477
18x5x5x2	95.3227
18x8x6x2	95.3977
18x6x7x2	95.3977
18x1x8x2	95.2976
18x9x9x2	95.3477
18x6x10x2	95.1476

A tabela 7.4 demonstra que a arquitetura que melhor se configurou para avaliação da condição de funcionamento da máquina entre bom e ruim, foi 18x4x4x2, apresentando 95.4477% de eficiência.

7.4.2 Resilient Backpropagation

Tabela 7.5 - Arquitetura utilizando Resilient Backpropagation com apenas uma camada oculta.

Arquitetura	Taxa de Acerto (%)
18x2x2	90.8454
18x3x2	93.9220
18x4x2	94.3472
18x5x2	90.5953
18x6x2	94.7474
18x7x2	91.3957
18x8x2	93.4717
18x9x2	91.7459

18x10x2	91.5708
18x12x2	90.1951
18x16x2	94.8974
18x19x2	94.6723
18x20x2	91.3957
18x21x2	91.2206
18x22x2	94.4472
18x23x2	90.4702
18x25x2	94.8974
18x26x2	95.0475
18x28x2	95.1726
18x30x2	94.7474

Os resultados acima demonstram que a configuração que mostrou-se mais eficiente na classificação de padrões da condição da máquina foi 18x28x2, com 95.1726% de eficiência.

Tabela 7.6 - Arquitetura utilizando Resilient Backpropagation com duas camadas ocultas.

Arquitetura	Taxa de Acerto (%)
18x5x1x2	94.4222
18x6x2x2	94.7224
18x9x3x2	94.6223
18x8x4x2	95.0475
18x9x5x2	95.1976
18x9x6x2	95.0725
18x7x7x2	95.0725
18x6x8x2	94.8224
18x9x9x2	95.0475
18x8x10x2	94.6723

Pela tabela 7.6, notamos que a arquitetura mais eficiente foi 18x9x5x2 utilizando o método de *Resilient Backpropagation* com duas camadas de neurônios ocultas, apresentando 95.1976% de eficiência na avaliação da condição da máquina.

7.4.3 Scaled Conjugate Gradient

Tabela 7.7 - Arquitetura utilizando Scaled Conjugate Gradient com apenas uma camada oculta.

Arquitetura	Taxa de Acerto (%)
18x2x2	92.2211
18x3x2	94.2471
18x4x2	92.9715
18x5x2	92.1461
18x6x2	91.6458
18x8x2	90.9705
18x9x2	91.6208
18x10x2	94.4222
18x12x2	85.8179
18x15x2	93.7969
18x18x2	94.8474
18x19x2	94.3972
18x20x2	92.8714
18x21x2	94.7974
18x22x2	90.7204
18x23x2	89.6698
18x25x2	95.2476
18x26x2	93.5218
18x28x2	94.7974
18x30x2	94.3222

Pelos resultados obtidos, observamos que a arquitetura de maior eficiência foi a 18x25x2 com 95.2476% de eficiência.

Tabela 7.8 - Arquitetura utilizando Scaled Conjugate Gradient duas camadas ocultas.

Arquitetura	Taxa de Acerto (%)
18x6x1x2	94.8474
18x2x2x2	94.8974
18x3x3x2	94.9225
18x9x4x2	95.1476
18x6x5x2	95.2226
18x1x6x2	94.6223
18x4x7x2	95.0725
18x3x8x2	94.8224
18x8x9x2	94.8974
18x7x10x2	93.5718

A tabela 7.8 demonstra que pelo método *Scaled Conjugate Gradient*, a configuração que apresentou melhor desempenho foi 18x6x5x2, com 95.2226% de eficiência na distinção da condição da máquina entre bom e defeituoso.

CAPÍTULO 8

CONCLUSÃO

Os resultados representam que a rede foi capaz de avaliar a condição do rolamento, em rolamento normal ou rolamento com defeito, em sua melhor arquitetura, com eficiência de cerca de 96%.

A porcentagem determinada para o treinamento influencia diretamente na capacidade de classificação da rede. O método utilizado para o treinamento também afetou os resultados apresentados. Levenberg-Marquardt e BFGS Quasi-Newton, apresentaram resultados com menor *range*, faixa entre a pior e a melhor classificação, quando variamos a arquitetura da rede. Porém, em algumas situações apresentaram resultados inconclusivos, o que pode comprometer a classificação. *Resilient Backpropagation* e *Scaled Conjugate Gradient*, apesar de apresentarem um maior *range*, quando variamos a arquitetura da rede, se mostraram estáveis, não apresentando resultados inconclusivos. Demonstrando assim, que estes dois métodos são mais indicados.

A arquitetura da rede é um fator determinante para a eficiência da classificação. Realizando um comparativo entre as arquiteturas, foi demonstrado que a quantidade de neurônios pouco interfere na capacidade de classificação da rede. Demonstrando que a quantidade de neurônios é um valor empírico.

Quando consideramos a quantidade de camadas ocultas, comparando uma ou duas camadas ocultas para o mesmo método de otimização, os valores dos resultados com melhor classificação foram bem próximos, variando cerca de 0,2 para Levenberg-Marquardt e BFGS Quasi-Newton e 0,025 para *Resilient Backpropagation* e *Scaled Conjugate Gradient*. Considerando todas as tentativas de configurações utilizando duas camadas ocultas para os métodos que não apresentaram inconclusões, *Resilient Backpropagation* e *Scaled Conjugate Gradient*, foi mantida a variação de cerca de 6% entre os valores globais de melhor e pior classificação. Quando utilizado apenas uma camada oculta, a variação foi de cerca de 4 pontos percentuais. Podemos considerar que

para este caso específico de reconhecimento de padrão, apenas uma camada oculta atende de forma eficiente o propósito.

Contudo, apesar da arquitetura ser um fator determinante para o melhor rendimento da rede, não existe na bibliografia metodologia que permita a generalização. Tornando necessária a experimentação, análise e ajuste desses parâmetros caso a caso.

TRABALHOS FUTUROS

Neste primeiro momento o foco da rede neural artificial foi voltado para a distinção entre dois tipos de padrões, o funcionamento de um rolamento normal e o funcionamento de um rolamento defeituoso. Seguem como propostas futuras algumas ponderações.

Utilizar as possibilidades de diferentes configurações da rede e avaliar o treinamento adequado para identificar outros tipos de defeitos relacionados a rolamentos ou abrangendo outros componentes de máquinas a rotativas. Ampliando a possibilidade de saídas para além de duas classificações.

Utilizar o domínio da frequência, para extrair informações de vibrações que permitam o reconhecimento de padrões de funcionamento. As informações no domínio da frequência são comumente utilizadas para diagnósticos de máquinas e diferentes defeitos produzem diferentes frequências características.

REFERÊNCIAS BIBLIOGRÁFICAS

BEALE, M. H., HAGAN, M. T., DEMUTH, H. B., 2011, *Neural Network Toolbox User's Guide*. R2011b. MATLAB.

BONNIARD, M. C., 2011, *Identificação de defeitos em bombas de grande porte através do método de decomposição ortogonal de Karhunen – Loève*. Tese de M. Sc, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

BUDYNAS, R. G., NISBETT, J.K., 2011, *Elementos de Máquinas de Shigley – Projeto de Engenharia Mecânica, 8ed. Porto Alegre, AMGH Editora Ltda.*

FK Bearing Group. Disponível em: <http://www.fk-bearing.com/en/products/emq-ball-bearing/60-series>. Acessado em: 06 abr. 2013.

HAYKIN, S., 1999, *Neural networks a Comprehensive Foundation*, 2ed. New Jersey, Prentice-Hall, Upper Saddle River.

ISO 3945, *Mechanical Vibration of large rotation machines with speed range from 10 to 200 r/s – Measure and evaluation of vibration severity in situ*, International Organization for Standardization, 1985.

LIU, T. I., MENGEL, J. M., 1992, “Intelligent Monitoring of Ball Conditions”, *Mechanical Systems and Signal Processing*, v. 6, n. 5, pp. 419-431.

MONTGOMERY, D. C, RUNGER, G. C., 2002, *Applied Statistics and Probability for Engineers*, 3ed. New York, John Wiley & Sons, Inc.

PACHECO, W. S., 2012, *Reconhecimento de Padrões de Operação de Máquinas utilizando técnicas de identificação de fontes sonoras*. Tese de D.Sc, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

SAMANTA, B., AL-BALUSHI, K.R., AL-ARAIMI, S.A., 2003, “Artificial Neural Networks and Support Vector Machines with Genetic Algorithm for Bearing Fault Detection”, *Engineering Applications of Artificial Intelligence*, v. 16, pp. 657-665.

SAMANTA, B., 2004, “Gear fault detection using artificial neural networks and support vector machines with genetic algorithms”, *Mechanical Systems and Signal Processing*, v. 18, pp 625-644.

SANTIAGO, D. F. A., 2004, *Diagnóstico de Falha em Máquinas Rotativas Utilizando Transformada de Wavelet e Redes Neurais Artificiais*. Tese de D.Sc, UNICAMP/DEM/DPM, Campinas, SP, Brasil.

SKF Brasil. Disponível em: <http://www.skf.com/br/products/bearings-units-housings/index.html>. Acesso em: 06 abr. 2013.

APÊDICE A. Algoritmos

Função tratamento de dados

```
function resultado=trat_dados_f(nome_arq)

nome_dir= 'C:\ENSAIOS_MATLAB\';
%nome_arq= '32hz_bom'; nome do arquivo com os dados no formato txt

dados=load ([nome_dir,nome_arq]);

ns=100;%n° de amostras por pacote
nt1=size(dados);
nt=nt1(:,1);
nc=nt1(:,2);

qnt_pac=((nt/ns)+((nt/ns)-1));%n° de loops
for j=1:nc
for i=1:qnt_pac
    %media(1)=mean(dados(1:100))
    %media(2)=mean(dados(51:150))
    %media(3)=mean(dados(101:200))
    %media(4)=mean(dados(151:250))
    media(i,j)=mean(dados(((i-1)*ns/2)+1:(i*ns)-(i-1)*ns/2,j));
end
end
media=media';

for j=1:nc
for i=1:qnt_pac

    desvio(i,j)=std(dados(((i-1)*ns/2)+1:(i*ns)-(i-1)*ns/2,j));

end
end
desvio=desvio';

for j=1:nc
for i=1:qnt_pac
variancia(i,j)=(var(dados(((i-1)*ns/2)+1:(i*ns)-(i-1)*ns/2,j)));
end
end
variancia=variancia';

for j=1:nc
for i=1:qnt_pac
assimetria(i,j)=(skewness(dados(((i-1)*ns/2)+1:(i*ns)-(i-1)*ns/2,j)));
end
end
assimetria=assimetria';

for j=1:nc
```

```

for i=1:qnt_pac
curtose(i,j)=(kurtosis(dados(((i-1)*ns/2)+1:(i*ns)-(i-1)*ns/2,j)));
end
end
curtose=curtose';

for j=1:nc
for i=1:qnt_pac
rms(i,j)=(sqrt(mean((dados(((i-1)*ns/2)+1:(i*ns)-(i-1)*ns/2,j).^2))));
end
end
rms=rms';

resultado=[media;desvio;variância;assimetria;curtose;rms];

```

Rede Neural Artificial

```

clear all
resultado_bom=trat_dados_f('32hz_bom');
resultado_ruim=trat_dados_f('32hz_ruim');
%%
I=ones(1,1999);
II=zeros(1,1999);
target1=[I;II];
target2=[II;I];
target=[target1 target2];
%%
% Solve a Pattern Recognition Problem with a Neural Network
% Script generated by NPRTOOL
% Created Thu Jan 17 13:54:37 BRST 2013
%
% This script assumes these variables are defined:
%
%   input_rms - input data.
%   target - target data.
inputs = [resultado_bom,resultado_ruim];
targets = target;
% Create a Pattern Recognition Network
neurocult1=1;
for i=1:10; %variação de n° neurônios na camada oculta
neurocult2=1;
for j=1:10; %variação de n° neurônios na camada oculta
hiddenLayerSize = [neurocult1,neurocult2];
net = patternnet(hiddenLayerSize);
% Choose Input and Output Pre/Post-Processing Functions
% For a list of all processing functions type: help nnprocess
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
net.outputs{2}.processFcns = {'removeconstantrows','mapminmax'};
%%
stream=RandStream.getGlobalStream;
reset(stream);%garante que o aleatório terá sempre os mesmos valores.
%%
% Setup Division of Data for Training, Validation, Testing
% For a list of all data division functions type: help nndivide

```

```

net.divideFcn = 'dividerand'; % Divide data randomly
net.divideMode = 'sample'; % Divide up every sample
net.divideParam.trainRatio = 70/100; % Porcentagem para treinamento
net.divideParam.valRatio = 15/100; % Porcentagem para validação
net.divideParam.testRatio = 15/100; % Porcentagem para teste
% For help on training function 'trainlm' type: help trainlm
% For a list of all training functions type: help nntrain
net.trainFcn = 'trainlm'; % Levenberg-Marquardt
% Choose a Performance Function
% For a list of all performance functions type: help nnperformance
net.performFcn = 'mse'; % Mean squared error
% Choose Plot Functions
% For a list of all plot functions type: help nnplot
%net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
% 'plotregression', 'plotfit','plotconfusion'};
% Train the Network
[net,tr] = train(net,inputs,targets);
% Test the Network
outputs = net(inputs);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs)
% Recalculate Training, Validation and Test Performance
trainTargets = targets .* tr.trainMask{1};
valTargets = targets .* tr.valMask{1};
testTargets = targets .* tr.testMask{1};
trainPerformance = perform(net,trainTargets,outputs)
valPerformance = perform(net,valTargets,outputs)
testPerformance = perform(net,testTargets,outputs)
% View the Network
%view(net)
% Plots
% Uncomment these lines to enable various plots.
%figure, plotperform(tr)
%figure, plottrainstate(tr)
%figure, plotconfusion(targets,outputs)
%figure, ploterrhist(errors)
c = confusion(targets,outputs);
acertos(i,j)=100-c*100
neurocult2=neurocult2+1;
end
neurocult1=neurocult1+1;
end
[MaxAc,neurooptimo]=max(acertos)
plot(tr.time(1,1:22),'DisplayName','tr.time(1,1:22)','YDataSource','tr
.time(1,1:22)');figure(gcf)
plot(tr.gradient(1,1:22),'DisplayName','tr.gradient(1,1:22)','YDataSou
rce','tr.gradient(1,1:22)');figure(gcf)

```