UNIVERSIDADE FEDERAL DO PARANÁ

ANDRÉ ANASTÁCIO DE OLIVEIRA

IDENTIFICAÇÃO DE MADEIRAS UTILIZANDO A ESPECTROMETRIA NO INFRAVERMELHO PRÓXIMO E REDES NEURAIS COM A HEURÍSTICA DE LEVEMBERG-MARQUARDT

CURITIBA

2013

UNIVERSIDADE FEDERAL DO PARANÁ

ANDRÉ ANASTÁCIO DE OLIVEIRA

IDENTIFICAÇÃO DE MADEIRAS UTILIZANDO A ESPECTROMETRIA NO INFRAVERMELHO PRÓXIMO E REDES NEURAIS COM A HEURÍSTICA DE LEVEMBERG-MARQUARDT

Dissertação apresentada ao Programa de Pós-Graduação em Métodos Numéricos em Engenharia, área de concentração em Programação Matemática, Departamento de Construção Civil e Setor de Ciências Exatas, Departamento de Matemática da Universidade Federal do Paraná, como parte das exigências para obtenção do título de Mestre em Métodos Numéricos em Engenharia.

Orientador: Profº. Dr. Paulo Henrique Siqueira

Co-orientadora: Profª. Dra. Silvana Nisgoski

CURITIBA

2013

FICHA CATALOGRÁFICA

TERMO DE APROVAÇÃO

AGRADECIMENTOS

Meus pais e irmãos por todo auxílio e compreensão.

Minha noiva, Andréa, pelo carinho, incentivo e paciência.

A minha colega de trabalho, Maria Oliva, pelo incentivo.

Ao Jose Henrique Ferreira por ser um exemplo.

Ao meu grande amigo, professor Guilherme J. Delben.

Aos meus orientadores, Professor Paulo e Professora Silvana por toda ajuda, apoio e confiança.

Aos membros da banca, Professor Luiz Neves e Professora Deise Bertholdi, por suas importantes contribuições.

Um agradecimento especial à Professora Graciela, por acreditar em mim mais do que eu próprio.

RESUMO

A identificação de uma espécie arbórea torna-se complexa quando tem-se à disposição apenas sua madeira, o que exige uma análise mais profunda para sua caracterização. Utilizando-se a espectrometria no infravermelho próximo é possível determinar a quantidade de absorção ou reflexão da radiação eletromagnética relativa a uma determinada faixa de números de onda. Como a quantidade de dados fornecidos pelo espectrofotômetro é grande, há a necessidade de utilizar-se ferramentas matemáticas e computacionais para o tratamento dessa informação. O objetivo deste trabalho é avaliar a aplicação de Redes Neurais Artificiais, com a heurística de Levemberg-Marquardt, como ferramenta auxiliar no reconhecimento de algumas espécies de madeira mediante a análise dos seus espectros ópticos, obtidos por aparelho de espectrometria no infravermelho próximo em modo de absorbância. Foram realizados três experimentos com os espectros das espécies de árvores Canela (*Nectandra* spp.), Canela Sassafrás (*Ocotea odorifera*), Imbuia (*Ocotea porosa*) e Eucalipto (*Eucalyptus* spp.). Os espectros analisados não passaram por nenhum tipo de filtro estatístico. No primeiro experimento, as quatro espécies foram utilizadas para o treinamento e teste da rede, utilizando-se as funções de ativação *tangente hiperbólica - sigmoidal* e *logarítmica - sigmoidal*, respectivamente. Obteve-se um reconhecimento superior a 99% das espécies, considerando-se uma tolerância ao erro de 2%. No segundo experimento as quatros espécies foram analisadas em combinações de três, obtendo-se, para uma tolerância de 2%, 100% de reconhecimento em todos os casos. No terceiro experimento, a faixa de números de onda analisada inicialmente foi dividida em quatro partes, cada uma foi analisada separadamente. Observou-se que a última parte analisada obteve o menor erro médio quadrático. Por fim, conclui-se que a utilização de Redes Neurais Artificiais para identificação de madeira apresentou resultados promissores, visto ser flexível aos ruídos existentes e não exigir que os dados passem por prévio tratamento estatístico antes de serem utilizados.

Palavras-chave: Redes Neurais Artificiais. Heurística de Levemberg-Marquardt. Identificação de madeira. Espectrometria no infravermelho próximo.

ABSTRACT

The identification of tree species becomes complex when it has available only your wood, which requires further analysis for their characterization. Using near infrared spectrometry is possible to determine the amount of absorption or reflection of electromagnetic radiation at certain range of wave numbers. As the amount of data provided by a spectrophotometer is large, there is a need to use mathematical and computational tools for the treatment of such information. The objective of this study is to evaluate the application of Artificial Neural Networks, with Levemberg-Marquardt's heuristic, as an auxiliary tool in the recognition of some wood species by analyzing their optical spectra obtained per device of near infrared spectroscopy in absorbance mode. Three experiments were conducted with the spectra of the species of trees Canela (*Nectandra* spp.), Canela Sassafrás (*Ocotea odorifera*), Imbuia (*Ocotea porosa*) and Eucalipto (*Eucalyptus* spp.). The specters analyzed doesn't passed by any type of statistical filter. In the first experiment, the four species were used for training and testing the network, using the sigmoidal hyperbolic tangent activation functions and sigmoidal logarithmic activation functions, respectively. Obtained a recognize over 99% of the species, considering an error tolerance of 2%. In the second experiment the four species were analyzed in combinations of three, to a tolerance of 2%, was reached 100% recognition in all cases. In the third experiment, the range of wave numbers initially analyzed was divided into four parts, each part was analyzed separately. It was observed that the last part examined had the lowest mean squared error. Finally, it is concluded that the use of neural networks to identify wood yielded promising results, since it is flexible to noise exist and do not require data undergo preliminary statistical treatment before being used.

Key words: Artificial Neural Network. Levenberg-Marquardt heuristics. Wood identification. Near infrared spectroscopy.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

CAPÍTULO I - INTRODUÇÃO

1.1 APRESENTAÇÃO DO PROBLEMA

Segundo Bremananth *et. al.* (2009), a identificação de uma espécie de árvore é mais fácil quando se tem acesso às suas folhas, flores e frutos. No entanto, esse trabalho torna-se complexo quando tem-se à disposição apenas sua madeira, sendo necessária uma análise mais aprofundada para identificação dos seus padrões físicos e anatômicos.

Atualmente existe uma lacuna entre a fiscalização e a identificação da madeira após o corte ou beneficiamento. Para a extração da madeira o madeireiro precisa de uma licença para exploração e do *Documento de Origem Florestal* (DOF) que permite o transporte do material retirado. A dificuldade na identificação das espécies madeireiras tem atrapalhado o sistema de fiscalização, pois muitas vezes o que está contido na carga não corresponde ao que foi autorizado para corte. Instituições como *World Wild Florest Brasil* (WWF-Brasil) e o *Instituto de Pesquisa Tecnológica* (IPT) fornecem cursos de capacitação para identificação botânica de madeira com o objetivo de treinar profissionais dos órgãos de fiscalização e policiamento ambiental na classificação macroscópica das espécies de madeira mais comercializadas no seu estado de atuação. A FIGURA 2 ilustra a fiscalização de madeira feita pela Polícia Rodoviária Federal (OLIVEIRA *et al.*, 2012).

Mesmo técnicos especializados possuem dificuldades em distinguir uma madeira de outra, visto a enorme variedade de espécies encontradas no Brasil. Até o final da década de 90 o IPT contava com uma xiloteca[1] dotada com mais de 18.000 amostras de madeira, espécimes nacionais e internacionais, abrangendo 1000 espécies, 600 gêneros e 100 famílias, além de uma coleção com mais de 13 mil lâminas histológicas (FRANCA, 2012).

As principais partes de um tronco são ilustradas na FIGURA 1.

[1]Xiloteca (em grego, *xilo* significa madeira e *theke*, coleção) guarda amostras de madeiras usadas para identificação de espécies. (FRANCA, 2012)

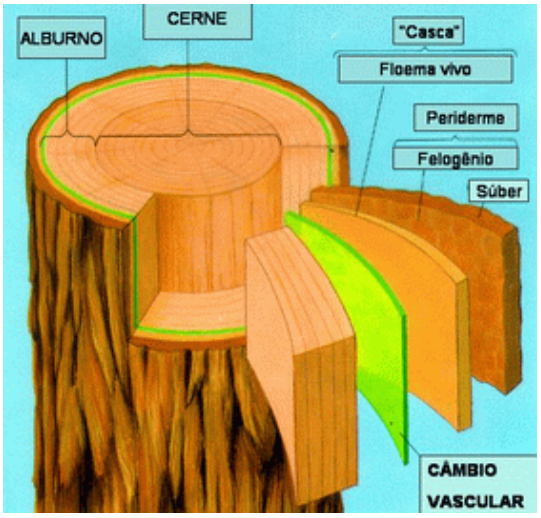


FIGURA 1: CAMADAS DO TRONCO
FONTE: DICAF(2012)

FIGURA 2: EXTRAÇÃO ILEGAL DE MADEIRA NO MATO GROSSO.
FONTE: EXTRAÇÃO (2012)

Em geral a classificação da madeira pode ser feita por meio de processos de identificação macroscópicos, uso de faca de polimento e uma lupa conta-fios, FIGURA 3, ou microscópicos, pela observação de lâmina histológica em microscópio óptico, FIGURA 4 (ZENID e CECCANTINI, 2007; NISGOSKI, 1999).

Também há a possibilidade de realizar essa análise mediante outras ferramentas, tais como as informações obtidas pela espectroscopia no infravermelho (CARNEIRO, 2008; LENGOWSKI, 2012).

FIGURA 3: ESTUDO PARA IDENTIFICAÇÃO BOTÂNICA DE MADEIRA, 1987 - ACERVO HISTÓRICO IPT
FONTE: IPT (2012)

FIGURA 4: IDENTIFICAÇÃO DE MADEIRA POR EXAME MICROSCÓPICO – ACERVO IPT
FONTE: IPT(2012)

Por meio dos instrumentos analíticos modernos pode-se extrair uma grande quantidade de informações (variáveis) de um elevado número de amostras (objetos) e analisá-los em um curto período de tempo. Isso ocasiona a necessidade de lidar com dados multivariados, complexos e de difícil interpretação, recorrendo-se a ferramentas matemáticas e estatísticas para a sua explicação.

De acordo com Pontes (2009), as técnicas de reconhecimento de padrões têm o objetivo de encontrar diferenças e similaridades entre conjuntos de amostras que foram submetidas a algum tipo de estudo, sendo que estes podem ser do tipo instrumental (espectrométricas, cromatográficas, etc.) ou mediante parâmetros de interesse (como densidade, pH, concentração de algumas espécies, etc.). Quando a quantidade de dados registrados é pequena a distinção entre os objetos (amostras) torna-se mais simples, o que, geralmente, não ocorre quando trabalha-se com uma quantidade elevada de variáveis (informações). Isso faz com que ferramentas de

reconhecimento de padrões sejam de extrema importância na avaliação de diferenças e similaridades em grupos de amostras, especialmente quando se trata de dados espectrométricos.

As impressões espectrais específicas de tipos de madeira podem ser analisadas por diferentes técnicas de exploração, sendo os métodos mais comuns na literatura aqueles que trabalham com a espectrometria vibracional (*near-infrared (NIR), middle infrared (MID) e Raman spectroscopy*)[2] pois oferecem uma grande gama de detalhes do espectro da madeira (LABATI *et al.*, 2009).

Em Harris e Bertolucci (1978) *apud* Lengowski (2012) a espectrometria pode ser definida como a análise das interações da radiação eletromagnética com a matéria. A espectrometria na região do infravermelho próximo (NIR) é uma das técnicas não destrutivas mais empregadas, devendo-se isso às vantagens como a determinação sem necessidade de tratamento da amostra, rapidez na obtenção de resultados, facilidade de transporte do instrumental, entre outras (LENGOWSKI, 2012; PAULA FILHO, 2012).

Em geral, na maioria dos trabalhos publicados, a análise dos dados é feita por meio de técnicas de análise multivariada como PCA (*Principal Component Analysis*) e PLS (*Partial Least Squares*), como pode ser visto em Carneiro (2008), também as ferramentas de MLR (*Mutiple Linear Regression*) e APS (*Successive Projections Algorithm*), utilizadas por Pontes (2009) e Dantas (2007). Estes métodos visam a formação de agrupamentos, no entanto exigem um conhecimento prévio por parte do manipulador (aluno, pesquisador, fiscal, etc.). O SIMCA (*Soft Independent Modeling of Class Analogy*) é utilizado na etapa de classificação, como abordou Pontes (2009).

1.2 OBJETIVO DO TRABALHO

1.2.1 Objetivo Geral

Verificar se a utilização de Redes Neurais Artificiais como técnica de reconhecimento de padrões pode auxiliar na identificação ou discriminação de

[2] Espectrometria no infravermelho próximo (NIR), no infravermelho médio (MID) e de Raman.

madeira mediante a análise dos seus espectros ópticos obtidos por aparelho de espectroscopia no infravermelho próximo (*NIR*), em absorbância.

1.2.2 Objetivos Específicos

- Aplicar uma rede neural, baseada no método de Levemberg-Marquardt, como ferramenta de identificação de algumas espécies de madeira;
- Analisar e comparar o desempenho da rede proposta para classificação de espécies de madeira mediante espectros de infravermelho próximo, sem filtragem estatística ou agrupamentos relacionados ao tipo de corte utilizado;

1.2.3 Por que redes neurais artificiais?

Arquitetar uma Rede Neural Artificial para a classificação de espécimes de madeira mediante seus espectros pode apresentar as seguintes vantagens:

- Método veloz de auxílio na identificação de espécimes por especialistas ou em campo junto a órgãos de fiscalização e policiamento ambiental;
- Possibilidade de indicar ao especialista uma direção, dentre as várias espécies a serem analisadas, para a identificação da amostra, visto que a rede pode não ser capaz de fornecer uma resposta exata, mas próxima da verdadeira.
- Destacar semelhanças/padrões entre as amostras analisadas;
- A rede desenvolvida pode ser executada em computadores portáteis em conjunto com o NIR (near-infrared spectroscopy), cuja utilização não necessita de tratamento da amostra, possui rapidez na obtenção de resultados além da facilidade de transporte do instrumental, ou seja, pode ser utilizada em campo junto a um computador portátil para análise dos resultados.

A principal característica de uma RNA é a sua capacidade de adquirir experiência e aperfeiçoar seu desempenho, baseando-se em dados reais para obter um modelo genérico com o objetivo de identificar padrões nesses dados (TATIBANA, 2006 *apud* BOÇOIS, 2012).

1.4 ESTRUTURA DO TRABALHO

O presente trabalho está dividido em seis capítulos.

Capítulo I - Introdução ao trabalho.

Capítulo II - Revisão de Literatura.

Capítulo III - Descrição do problema.

O Capítulo IV - Descreve, em detalhes, os conceitos das Redes Neurais Artificiais.

O Capítulo V - Descreve a implementação da Rede Neural e a análise dos resultados obtidos pela rede.

Capítulo VI - Conclusões obtidas pela análise dos resultados do Capítulo V e as sugestões para futuros trabalhos.

Os apêndices encontram-se nas páginas finais do trabalho.

CAPÍTULO II - REVISÃO DE LITERATURA

Neste capítulo é feito uma síntese de trabalhos envolvendo a aplicação de Redes Neurais Artificiais, inicialmente de forma global e, posteriormente, salientando o emprego na área de classificação de madeira.

2.1 REDES NEURAIS ARTIFÍCIAIS

Mendes (2008) desenvolveu uma rede neural, por meio do *Neural Network Toolbox* do software MATLAB, para a predição do tipo de cefaléia a partir de entradas retiradas de um questionário de anamnese. Para o treinamento e teste da rede foram utilizadas informações coletadas através de questionário respondido por 2177 pacientes identificados com algum tipo de cefaléia. O desempenho da rede, relativo à predição correta, atingiu um pico de 98,53% de acerto.

Pimentel (2005) apresentou um modelo empírico, direcionado ao refino de petróleo, baseado em RNA para determinar o comportamento estacionário de uma unidade de craqueamento catalítico fluido. O objetivo foi predizer as vazões das correntes de saída do processo (*gasolina, GLP, gás combustível, LCO e o óleo decantado*) levando em consideração treze variáveis de entrada. O modelo proposto utilizou uma arquitetura do tipo MLP (*Multi-Layer Perceptron*) usando o algoritmo de Levenberg-Marquardt para a otimização dos pesos da rede. A rede foi implementada por meio da ferramenta *Neural Network Toolbox* do software MATLAB. Os resultados apresentaram uma média de 77% de acerto para os problemas propostos.

Hervás *et al.* (1996) empregaram RNAs para a classificação qualitativa de carcaças de porcos ibéricos entre grupos de preço. Os espectros obtidos por um aparelho de infravermelho próximo (NIR) foram utilizados como fonte de informação para a classificação. Nesse trabalho foi feito o comparativo entre 4 arquiteturas de redes neurais: uma rede LVQ (*Linear Vector Quantization*); uma rede MLP; uma rede com aleatoriedade na direção de busca; e uma rede linear competitiva simples. A rede com arquitetura LVQ foi a que apresentou melhor desempenho, com aproximadamente 96% de acerto.

Boçois (2012) desenvolveu uma Rede Neural com arquitetura de Kohonen para a identificação de seis doenças dermatológicas (*Impetigo, Cromoblastomicose, Esporotricose, Herpes Labial, Larva Migrans Cutânea e Leishmaniose Tegumentar*) a partir das características da lesão apresentada pelo paciente. Para a identificação dos agrupamentos foi utilizado o método de visualização da *U-matriz* e o sistema descritor *RGB*. A rede obteve um acerto geral de 94,5% no conjunto de testes.

Baldin (2002) propôs uma RNA do tipo Perceptron, com aprendizagem *Backpropagation* utilizando o método do Gradiente Descendente para atualização de pesos, como ferramenta de auxílio na orientação vocacional. A ideia principal é treinar uma RNA com tendências ou qualidades de pessoas profissionalmente satisfeitas e, posteriormente, utilizar este treinamento para orientar os jovens na escolha de uma determinada profissão. Os melhores resultados, na etapa de teste, obtiveram reconhecimento de tendências superior a 90%.

2.2 TÉCNICAS PARA CLASSIFICAÇÃO DE MADEIRA

Paula Filho (2012) elaborou um protocolo comparativo entre diversos métodos de classificação, não necessariamente Redes Neurais, mediante a análise de imagens macroscópicas de madeira cortada. Foram construídas duas bases de imagens (obtidas em campo e em laboratório) a partir de amostras de madeira de espécies florestais encontradas no território nacional. Os atributos extraídos, de acordo com protocolo proposto, foram avaliados por técnicas como análise de cor, Gray-level Co-occurrence Matrices, histograma de borda, Fractais e filtro Gabor. Os resultados foram de 95,82% de acerto para imagens colhidas em campo e 99,49% para as imagens obtidas em laboratório.

Pinto Junior *et al.* (2006) proporam uma RNA para identificar classes de defeitos de madeira serrada de eucalipto em imagens digitais, utilizando-se como características os percentuais das bandas do vermelho, verde e azul detectados por uma câmera especial dotada de filtro RGB. Os resultados variam de 76,6% a 83,1% de acerto, dependendo do tamanho dos blocos de madeira analisados.

Labati *et al.* (2009) desenvolveram um mecanismo para classificação de alguns tipos de madeira, utilizadas na fabricação de painéis, a partir dos espectros de infravermelho captados por filtros ópticos de baixo custo acoplados a um conjunto

de fotodetectores. A ideia é treinar uma rede para identificar se a madeira utilizada é adequada para a fabricação de painéis e juntá-la a um sistema de reconhecimento eletrônico. Faz-se incidir um feixe de laser sobre a amostra e fotodetectores com filtros nas cores vermelho, verde e azul captam as frequencias de infravermelho associadas. Dentre as duas situações de teste propostas, o melhor desempenho atingiu precisão, no reconhecimento, superior a 97%.

Esteban *et al.* (2009) foi utilizada uma Rede Neural do tipo Perceptron com *Backpropagation* com regularização Bayesiana para a identificação de dois tipos de madeira de árvores nativas das Ilhas Canárias, *Juniperus cedrus* e *J. phoenicea*. Como essas duas espécies apresentam características anatômicas similares, a ideia da rede desenvolvida foi utilizar, em conjunto, atributos biométricos e anatômicos para classificação. O percentual de sucesso atingido no reconhecimento foi de 92 %.

Packianather e Drake (2000) aplicaram uma RNA com o objetivo de identificação de falhas em madeira laminada colada (em inglês *plywood*) produzidas por uma indústria de beneficiamento. A ideia é que um sistema computadorizado possa fazer a identificação de 12 tipos de falhas poupando recursos humanos para tal tarefa. O trabalho resultou numa precisão de 86,5% e numa diminuição no tempo de identificação se comparado com classificadores humanos.

Jordan *et al.* (1998) utilizou as respostas elásticas, devido à excitação ultrassônica da madeira, para treinar uma RNA com o objetivo de classificar sua origem entre os quatro diferentes espécimes de árvores utilizadas. Foi utilizada uma RNA do tipo MLP (*Multi-layer Perceptron*) com 3 camadas. O modelo teve desempenho aproximado de 97% na classificação das espécies.

Bremananth *et al.* (2009) as diferenças nas características presentes nas cascas de espécies distintas de árvores foram utilizadas para treinar uma RNA como ferramenta de classificação. As informações para treinamento da rede foram obtidas pela análise da imagem das suas cascas mediante técnicas como Gray-level Co-occurrence Matrices (*GLCM*) e passaram por um tratamento estatístico com PCA (*Principal Component Analisys*). Quatro espécies de árvores, nativas da América do Norte, foram utilizadas no trabalho. O resultado atingiu 97% de precisão.

CAPÍTULO III - DESCRIÇÃO DO PROBLEMA

3.1 IDENTIFICAÇÃO DA MADEIRA

De acordo com Zenid e Ceccantini (2007), desde a antiguidade o homem busca agrupar os diversos conjuntos de seres vivos no intuito de formar grupos com finalidade prática e/ou econômica. No aspecto biológico, a classificação é essencial e diversos cientistas, atuantes na área de plantas, necessitam de uma sistema de referência, pois ligado a um nome existe uma série de indivíduos com características em comum.

Ainda, segundo os autores Zenid e Ceccantini (2007), na área de tecnologia da madeira a identificação botânica da amostra em estudo permite o acesso às suas propriedades, sendo que estas estão, frequentemente, disponíveis em bancos de dados, livros, artigos entre outros, o que propicia um melhor conhecimento e aplicação do material. A utilização correta dos espécimes de madeira está sujeita a procedimentos que garantam a sua identificação, quer como árvores, serragem, pó, carvão ou toras brutas. Pode-se também ressaltar que a identificação de madeira é útil à indústria e comércio, onde favorece meios para localizar fraudes ou enganos.

Em geral, a identificação de um vegetal arbóreo, matéria-prima para a produção de madeira serrada, pode ser realizada em diferentes níveis de confiabilidade, mediante processos científicos (macroscópicos e microscópicos) ou de práticas empíricas (conhecimento popular).

Processos complexos, envolvendo diversas etapas, são utilizados na identificação científica da madeira. Na etapa macroscópica, FIGURA 5 - (B), a amostra é analisada em relação às suas características organolépticas ou sensoriais como: cor, desenhos, densidade e depois, após a superfície ser polida, características anatômicas como anéis de crescimento, raios, vasos e parênquimas são observados. Características como porosidade, largura e altura do raio, presença de estratificação, arranjo dos vasos e parênquimas também são avaliadas. (NISGOSKI, 1999; ZENID e CECCANTINI, 2007)

Na identificação microscópica, FIGURA 5 - (A), são analisadas as particularidades do tecido e células lenhosas tais como: composição dos raios; tipo e disposição das pontoações; presença de células oleíferas; espessamentos; tilos;

gomas; cristais; sílica; fibras septadas, etc. (CORE *et al.*, 1979 *apud* NISGOSKI, 1999; ZENID e CECCANTINI, 2007).

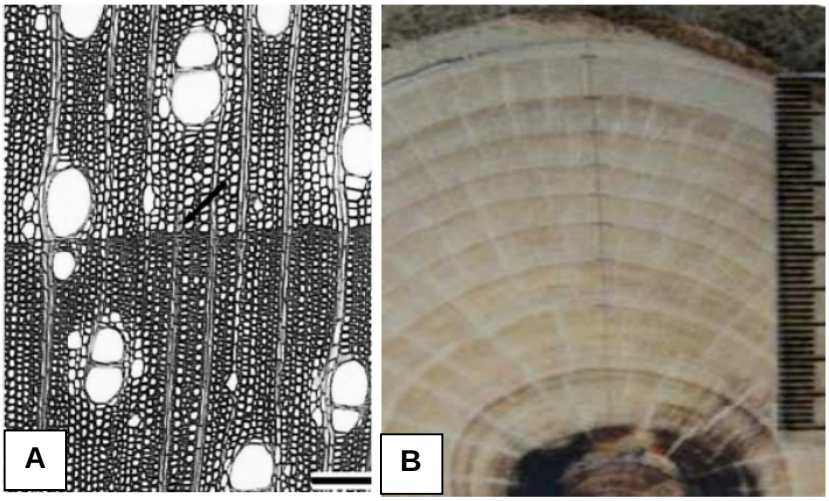


FIGURA 5: SEÇÃO MICROSCÓPICA (A) E MACROSCÓPICA (B) DE UM TRONCO DE *OCOTEA porosa* (*nees*), CONHECIDA COMO IMBUIA, UTILIZANDO O CORTE TRANSVERSAL.
FONTE: (COSMO *et al.*, 2009)

A utilização de processos macroscópicos ou microscópicos, simultaneamente ou não, envolvem a habilidade e o conhecimento do observador, porém é de grande importância ter à disposição uma coleção de madeiras (*xiloteca*) cujos exemplares tenham procedência conhecida. A multiplicidade de espécies existentes torna a identificação anatômica de madeiras uma tarefa relativamente complicada, por isso métodos auxiliares como manuais ilustrados, chaves dicotômicas, cartões-perfurados e bancos de dados eletrônicos são utilizados. Com o desenvolvimento tecnológico, métodos não anatômicos como fluorescência do extrato aquoso ou etílico, testes com cromoazurol e de saponina e, em especial, a espectroscopia no infravermelho próximo (NIR) são utilizados. (ZENID e CECCANTINI, 2007; PAULA FILHO, 2012)

De acordo com os autores Paula (1997) *apud* PAULA FILHO (2012) e Jordan *et al.* (1998) a madeira tem características de materiais anisotrópicos, isto é, não apresentam as mesmas propriedades em todas as direções (planos de corte), FIGURA 6.

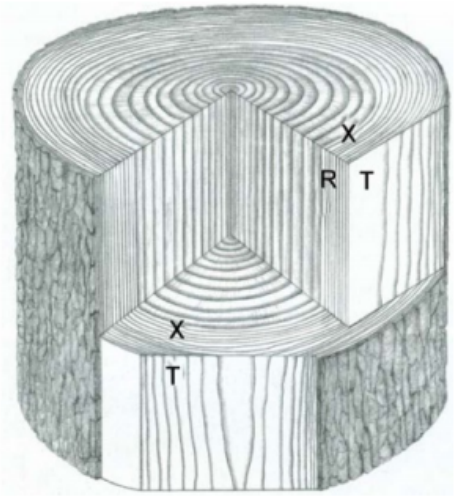


FIGURA 6: PLANOS DE CORTE DE UM TRONCO.
FONTE: (PAULA FILHO, 2012)

As direções distintas vistas na FIGURA 6 podem ser definidas como:

- Transversal (X): Perpendicular ao eixo da árvore;
- Longitudinal Radial (R): na direção axial, paralelo ao eixo maior do tronco e à direção dos raios lenhosos, perpendicular aos anéis de crescimento;
- Longitudinal Tangencial (T): na direção axial, paralelo ao eixo maior do tronco, em ângulo reto ou perpendicular aos raios lenhosos e, ainda, tangencial aos anéis de crescimento;

A variação do corte em X (transversal), R (radial) e T (tangencial) está relacionada ao fim pretendido para a madeira cortada podendo ser estético ou referido à resistência.

3.2 FLORA ARBÓREA BRASILEIRA

Fontes como o *Instituto Brasileiro de Florestas* (IBF) ressaltam que as variações climáticas, de relevos e solos presentes no território brasileiro proporcionam que a extensão do país seja a base mais rica em flora do mundo, possuindo mais de 56.000 espécies de plantas identificadas.

Segundo o site do *Sistema Nacional de Informações Florestais* (SNIF) o Brasil é considerado um país com uma grande diversidade em sua formação vegetal e ecossistemas, abrigando uma das floras mais diversas e viçosas do planeta. Existem no Brasil cerca de 7.880 espécies arbóreas catalogadas, mas estima-se a

existência de aproximadamente 11.120 espécies de árvores apenas na floresta Amazônica.

Ainda segundo o SNIF, do total das espécies conhecidas, 472 compõem a "Lista oficial das espécies da flora brasileira ameaçadas de extinção"[3], a última lista de espécies da flora ameaçadas foi divulgada em 2008, através da Instrução Normativa nº 6, de 23 de setembro de 2008. Neste documento, as espécies estão divididas em categorias como "*ameaçadas de extinção*" e "*com deficiência de dados*". Nela também são propostas políticas para cada categoria, com o objetivo de conservação das espécies. Os biomas com maior número de espécies ameaçadas são: a Mata Atlântica (276), o Cerrado (131) e a Caatinga (46). Já a Amazônia aparece com (24) espécies, o Pampa com (17) e o Pantanal com (2).

Dentre as espécies listadas, 25 são encontradas em território paranaense, conforme mostra a TABELA 1.

TABELA 1: ESPÉCIES PARANAENSES INCLUÍDAS NA "LISTA OFICIAL DAS ESPÉCIES DA FLORA BRASILEIRA AMEAÇADAS DE EXTINÇÃO"

Famílias	Espécies	Biomas	Locais
Araucariaceae	*Araucaria angustifolia* (Pinheiro-brasileiro, pinheiro do Paraná)	Mata Atlântica	MG, PR, RJ,RS, SC, SP
Arecaceae	*Bactris hatschbachii*	Mata Atlântica	PR, SP
Arecaceae	*Butia eriospatha* (Butiá)	Pampa	PR, RS, SC
Arecaceae	*Euterpe edulis* (Jussara, palmito)	Mata Atlântica	AL, BA, ES, GO, ,PB, PE, PR, RJ, RN, SE, SC, SP
Asteraceae	*Viguiera aspilioides*	Cerrado	PR
Asteraceae	*Viguiera paranensis*	Mata Atlântica	PR
Bromeliaceae	*Aechmea apocalyptica*	Mata Atlântica	PR, SC, SP
Bromeliaceae	*Dyckia hatschbachii* (Gravatá, bromélia)	Mata Atlântica	PR
Bromeliaceae	*Vriesea brusquensis* (Gravatá, monjola, bromélia)	Mata Atlântica	PR, SC
Bromeliaceae	*Vriesea muelleri* (Gravatá)	Mata Atlântica	PR, SC

[3] A lista oficial das espécies nativas em extinção pode ser obtida no link: http://www.mma.gov.br/estruturas/ascom_boletins/_arquivos/83_19092008034949.pdf

Bromeliaceae	*Vriesea pinottii* (Gravatá, monjola, bromélia)	Mata Atlântica	PR, SC
Dicksoniaceae	*Dicksonia sellowiana* (Xaxim, xaxim-imperial)	Mata Atlântica	MG, PR, RJ, RS, SC, SP
Lauraceae	*Ocotea odorifera* (Canela-sassafrás, sassafraz)	Mata Atlântica	ES, MG, PR, RJ, RS, SC, SP
Lauraceae	*Ocotea porosa* (Imbuia)	Mata Atlântica	PR, RS, SC
Lejeuneaceae	*Myriocoleopsis fluviatilis*	Mata Atlântica	PR, SC, SP
Moraceae	*Dorstenia tenuis* (Violeta-da-montanha, violeta-montes)	Mata Atlântica	PR, SC
Poaceae	*Thrasyopsis jurgensii*	Pampa / Mata Atlântica	PR, RS, SC
Pottiaceae	*Erytrhophyllastrum andinum*	Mata Atlântica	PR

FONTE:(MMA - Ministério do Meio Ambiente, 2008)

Existem, também, as espécies protegidas por lei nos âmbitos Municipal, Estadual ou Federal. Dentre elas podemos citar:

- Castanheira (*Bertholetia excelsa*) (Decreto Federal 5.975/2006);
- Seringueira (*Hevea* spp) (Decreto Federal 5.975/2006);
- Mogno (*Swieteniamacrophylla*) (Decreto Federal 6.472/2008).

Nas subseções seguintes será apresentada uma sinopse de cada espécie arbórea utilizada para o desenvolvimento deste trabalho. Não foi feita restrição alguma quanto a espécies brasileiras nativas ou protegidas pela legislação.

3.3 LAURACEAE

De acordo com Silva e Medeiros (2000) a família Lauraceae possui de 30 a 50 gêneros e cerca de 2.000 espécies, em geral são plantas lenhosas arbóreas, no entanto, também possui exemplares com porte de erva, herbáceo, como a Cassita. Esta família possui distribuição predominantemente tropical e subtropical, especialmente nas florestas centro e sul-americanas e, também, na região indo-

malaia (Indonésia). Em território brasileiro, as Lauraceae apresentam distribuição extensa com maior incidência nas restingas do litoral, nos cerrados e nas matas.

3.3.1 Canela Sassafrás (*Ocotea odorifera*)

A Canela Sassafrás, FIGURA 7, também é conhecida como Sassafrás, Sassafrás-amarelo ou Sassafrás-preto. Famosa pelo seu odor característico, essa árvore pode atingir aproximadamente 25 metros e ter um tronco com 70 cm de diâmetro. A localização predominante é a região da Mata Atlântica e pode ser encontrada nos estados do Espírito Santo, Minas Gerais, Paraná, Rio de Janeiro, Rio Grande do Sul, Santa Catarina e São Paulo. Segundo o Ministério do Meio Ambiente, esta espécie está presente na *lista oficial de espécies da flora brasileira ameaçadas de extinção*.

Devido à boa qualidade da madeira, a Canela Sassafrás possui utilização para fabricação de móveis, construção civil, embarcações, marcenaria, portas trabalhadas, caixotaria, molduras, fabricação de folhas faqueadas para revestimentos decorativos e caixilhos. Seu óleo, de fragrância característica, é utilizado na perfumaria, enceramento de piso, polimento, detergentes, sabões, agentes limpantes entre outras aplicações na indústria farmoquímica, principalmente na produção do safrol[4] (MILANESI, 2008).

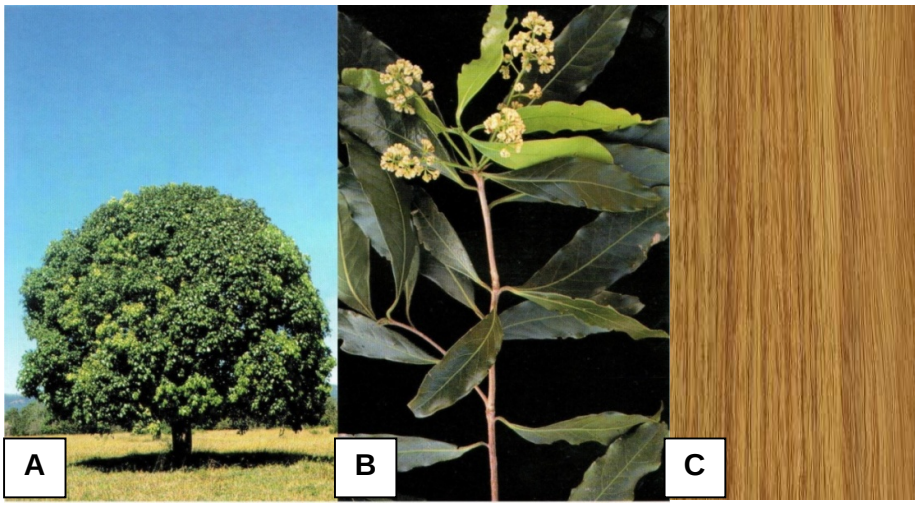


FIGURA 7: CANELA SASSAFRÁS (*Ocotea odorifera*). A: a árvore; B: a folha; C: a madeira.
FONTE: (LORENZI, 2002)(IBF - Instituto Brasileiro de Florestas)(ADAPTADO)

[4] É um fenil éter volátil encontrado naturalmente nos óleos essenciais de Canela-Sassafrás e utilizado na síntese de diversos compostos químicos como heliotropina, etilvanilina, butóxido de piperonila, etc.

3.3.2 Canela (*Nectandra* spp.)

A *Nectandra* spp, FIGURA 8, também é conhecida como Canela-preta, Canela-parda, Louro-preto, Canela-ferrugem ou Canela-puante. Existem várias espécies afins, as quais alguns autores chamam de variedade, por ser sua madeira de difícil distinção. Pode ser encontrada em todo território brasileiro, mas sua predominância é na região de florestas pluviais numa faixa que se estende do estado da Bahia até o extremo sul do Brasil. Não é raro encontrar árvores dessa espécie cujo tronco tenha diâmetro variando entre 50 a 60 cm e altura de até 30m, sua casca é lisa e possui tom acinzentado.

A madeira tem coloração que vai do amarelo-pardacento ao pardo-escuro, com manchas escuras; superfície lustrosa e áspera. Grã irregular, textura média, cheiro e gosto imperceptíveis. A Canela-parda possui excelentes propriedades para trabalhos com máquinas e ferramentas manuais, dando bom acabamento além de possuir alta resistência natural ao ataque de fungos e insetos. É empregada, geralmente, em acabamentos internos na construção civil, esquadrias, caixilhos, rodapés, móveis populares, embalagens leves, etc.

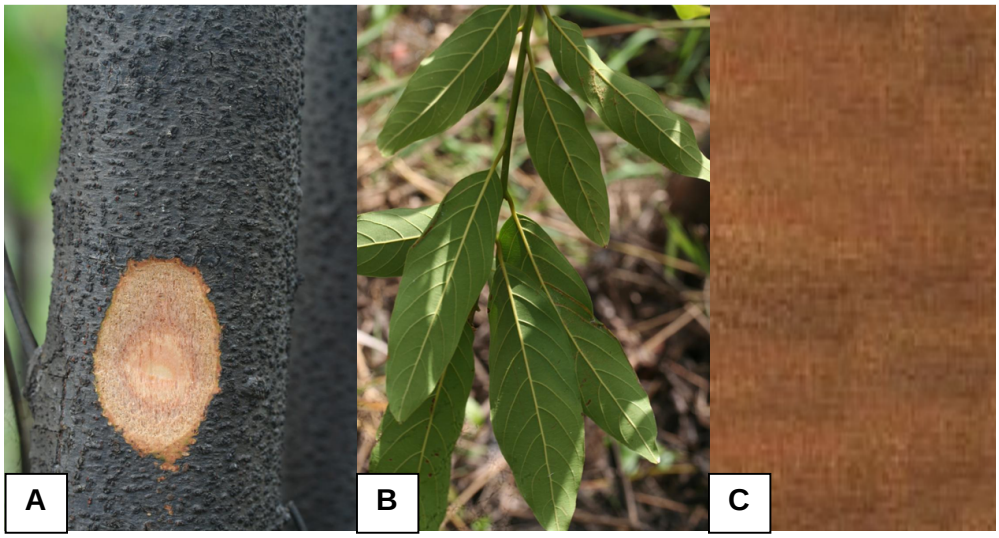


FIGURA 8 : CANELA (*Nectandra* spp.). A: o tronco; B: a folha; C: a madeira.
FONTE: (LAURACEAE, 2012; REMADE, 2009)

3.3.3 Imbuia (*Ocotea porosa*)

A *Ocotea porosa*, FIGURA 9, popularmente conhecida como Imbuia, é árvore nativa brasileira e protegida por lei, pode ser encontrada nos estados do Paraná, São Paulo, Santa Catarina, Rio Grande do Sul e Rio de Janeiro. Sua altura varia entre 15 e 20 metros, tendo como características marcantes o crescimento lento e o tronco grosso. (LORENZI, 2002)

Essa árvore sempre teve fundamental participação no desenvolvimento econômico e cultural nas regiões onde se encontra devido à suas boas propriedades tecnológicas (ex.: acabamento fácil, durabilidade e resistência à água). Com boa aceitação no mercado e podendo ser usada para diversos fins é muito procurada pela indústria moveleira para fabricação de móveis de luxo, folhas faqueadas decorativas, peças torneadas, painéis compensados e divisórias. A construção civil também aproveita das características de resistência presentes nas peças de Imbuia para servirem de vigas, caibros, ripas, marcos ou batentes, portas, janelas, molduras, lambris e similares. Também é utilizada por carvoarias e na arborização urbana (ÁVILA *et al.*, 2011 ; Instituto de Pesquisa e Estudos Florestais - IPF).

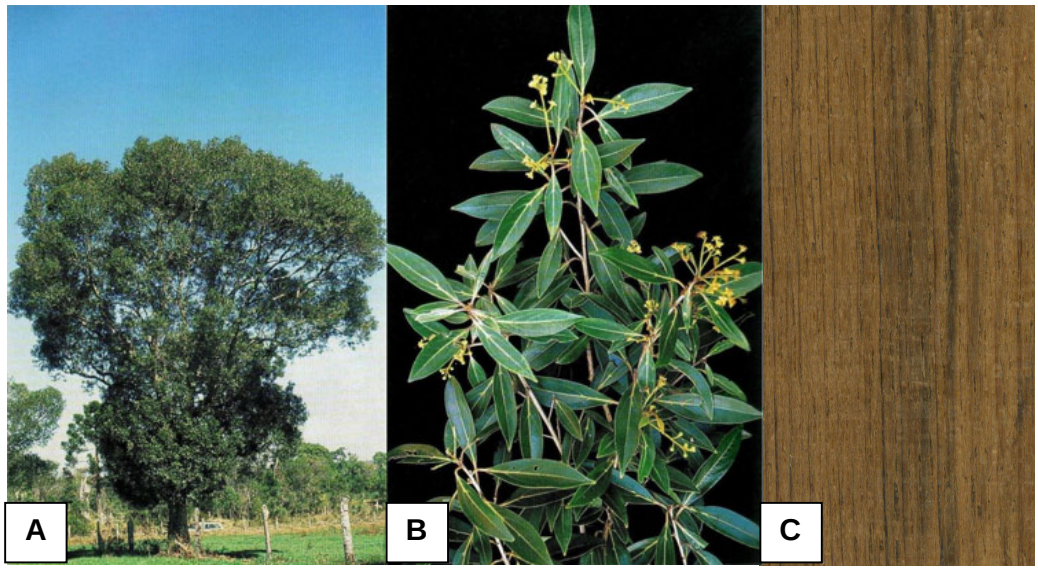


FIGURA 9: IMBUIA (*Ocotea porosa*). A: a árvore; B: a folha; C: a madeira.
FONTE: (LORENZI, 2002)(IBF - Instituto Brasileiro de Florestas)(ADAPTADO)

3.4 MYRTACEAE

A família Myrtaceae abrange duas grandes subfamílias: Myrtoideae e Leptospermoideae, sendo que a primeira apresenta maior concentração nas Américas do Sul e Central, a segunda registra maior incidência na Austrália. Apenas a subfamília Myrtoideae conta com cerca de 70 gêneros e 2400 espécies (ARAGÃO; CONCEIÇÃO , 2008).

Ainda segundo os autores, a taxonomia (classificação) é bastante complexa dentro da família Myrtaceae, o que torna elevado o número de indeterminação nos levantamentos florísticos e fitossociológicos.

3.4.1 Eucalipto (*Eucalyptus* spp.)

O Eucalipto (*Eucalyptus* spp.) é uma planta nativa da Austrália, com mais de 600 espécies FIGURA 10 e que chegou ao Brasil no início do século XX, inicialmente foi utilizada como dormente nas ferrovias, lenha para locomotivas e, posteriormente, como poste para rede elétrica (FOELKEL, 2009).

No fim da década de 20, a madeira de eucalipto começou a ser transformada em carvão mineral para o processo de fabricação do ferro-gusa, nas siderúrgicas. A partir desse ponto, novas formas de utilização foram desenvolvidas (FOELKEL, 2009).

Atualmente, o eucalipto tem um aproveitamento praticamente integral. Das suas folhas são extraídos óleos essenciais utilizados em produtos de limpeza e alimentícios, em perfumes e remédios. A casca oferece tanino, usado na produção de couro. O tronco produz madeira para sarrafos, lambris, ripas, vigas, postes, varas, esteios para minas, mastros para barco, tábuas para embalagens e móveis. Sua fibra é utilizada como matéria-prima para a fabricação de papel e celulose (FOELKEL, 2009).

Segundo Pinto Júnior (2010), o eucalipto está presente em mais de 450 municípios com franca expansão na maioria dos estados brasileiros com tradição na cultura deste gênero ou em estados considerados como novas fronteiras da silvicultura. Dentre as espécies de Eucalipto, pode-se citar o fim pretendido e as espécies que melhor se ajustam a ele.

Para lenha e carvão: espécies que dêem grande quantidade de lenha em prazo curto (*Eucalyptus grandis, E. urophylla, E. torilliana, E. camadulensis, E. tereticornis*) (TRUGILHO *et al.* 2005; FOELKEL, 2009) .

Para papel e celulose: espécies que apresentem cerne branco e macio (*Eucalyptus grandis, E. saligna, E. urophylla*) (SANTOS, 2005; TRUGILHO *et al.*, 2005).

Para postes, mourões, dormentes e estacas: espécies com cerne duro (para resistir ao tempo), (*Eucalyptus citriodora, E. robusta, E. globulus*) (PINTO JÚNIOR, 2010; SANTOS, 2005).

Para serrarias: espécies de madeira firme, em que não ocorram rachaduras (*Eucalyptus dunnii, E. viminalis, E. grandis*) (PINTO JÚNIOR, 2010).

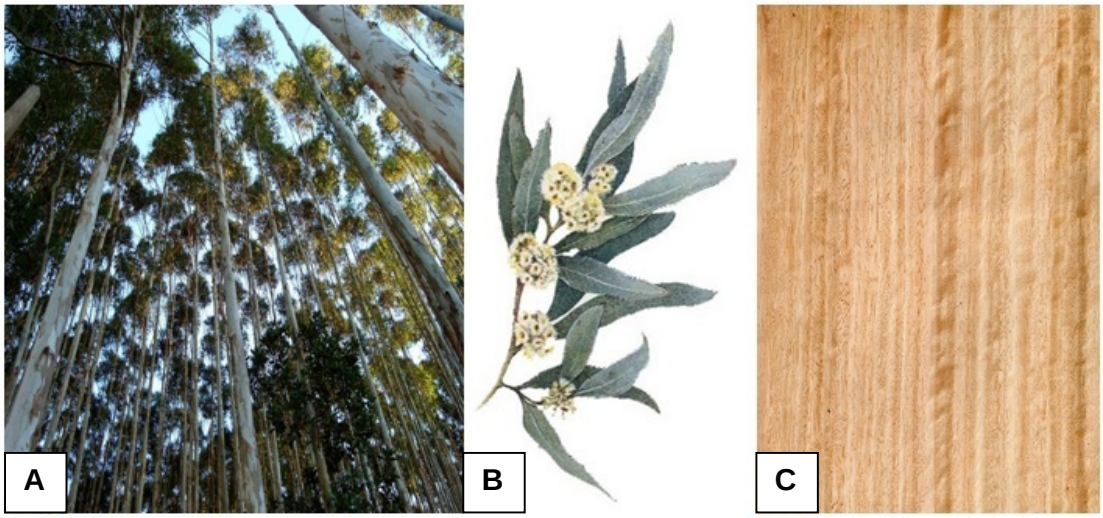


FIGURA 10: EUCALIPTO (*Eucalyptus* spp.). A: a árvore; B: a folha; C: a madeira.
FONTE:(FOELKEL, 2009; Instituto Brasileiro de Florestas(IBF)) (ADAPTADO)

3.5 OBTENÇÃO DOS DADOS

Os dados utilizados na implementação deste trabalho foram coletados junto ao Laboratório de Anatomia da Madeira da Universidade Federal do Paraná, no ano de 2012.

Os espectros das amostras estudadas foram coletados utilizando o espectrofotômetro NIR Bruker Tensor 37, FIGURA 11, que tem por base a interferometria de Fourier, em modo de absorbância com 64 varreduras (scans) e resolução de 4 cm^{-1}. A região espectral do infravermelho utilizada, medida em *número de ondas*[5], do inglês *wavenumbers*, variou de 3996 - 9995 (cm^{-1}) com passo (resolução) de 4 (cm^{-1}), totalizando 1556 leituras para cada amostra de cada espécie (STUART, 2004).

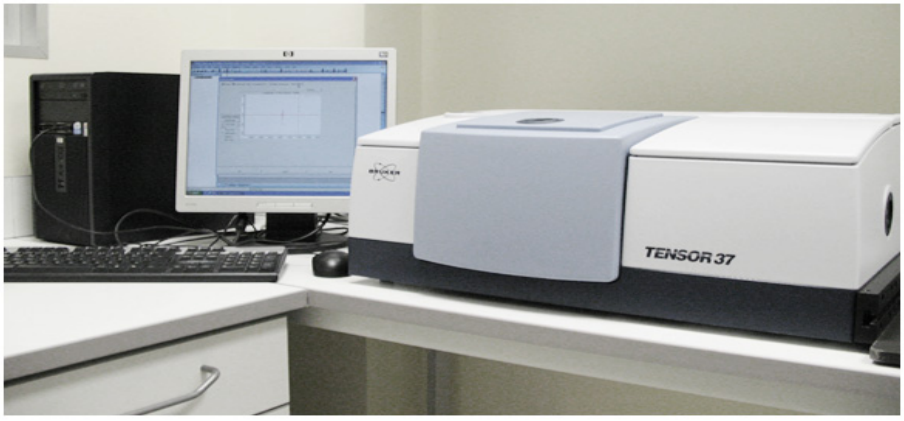


FIGURA 11: ESPECTRÔMETRO MARCA BRUKER, MODELO TENSOR 37
FONTE: NATIONAL (2013)

As amostras não receberam a adição de nenhum tipo de tratamento químico, apenas foram cortadas em um tamanho apropriado para propiciar a realização da coleta de informação pelo aparelho de NIR.

As informações registradas pelo NIR foram interpretadas pelo software *OPUS SPECTROSCOPY*, padrão nos equipamentos de espectrometria da empresa BRUKER, cuja função foi realizar as medições e processamento dos espectros. O *OPUS*, entre outras possibilidades, foi capaz de fornecer uma matriz de dados do tipo *número de onda* por *índice de absorbância*.

---

[5] o *wavenumber (número de onda)* é uma unidade, em (cm^{-1}), amplamente utilizada em espectrometria. Ele indica o número de ondas em um centímetro de comprimento.

Como não havia compatibilidade entre os softwares MATLAB e *OPUS*, fez-se necessária a utilização de uma ferramenta intermediária. Neste caso os dados obtidos pelo *OPUS* foram preparados pelo software *Unscrambler*, da empresa CAMO S.A., e repassados ao MATLAB, visto existir compatibilidade entre os dois.

3.5.1 Espectrometria no infravermelho

Segundo Hecht (2002), a região do infravermelho está dentro da faixa de frequência que varia de $3x10^{11}$ Hz até $4x10^{14}$ Hz, sendo detectado, inicialmente, pelo astrônomo inglês Sir William Herschel (1738-1822) enquanto estudava os níveis de energia associados com os comprimentos de onda da luz visível. Essa faixa de abrangência está abaixo da relativa à radiação eletromagnética do vermelho, por isso o nome infravermelho. Embora não exista universalidade quanto a isso, o infravermelho (IR), em geral, pode ser dividido em quatro regiões: o infravermelho próximo (*near infrared*), ou seja, próximo do visível e com comprimento de onda variando entre (780 - 3000 nm); o infravermelho médio (*intermediate/medium infrared*) variando entre (3000 - 6000 nm); o infravermelho distante (*far infrared*) variando entre (6000 - 15000 nm); o infravermelho extremo (*extreme infrared*) variando entre (15000 nm - 1 mm). O comprimento de onda é simbolizado pela letra grega lambda (λ) e é medido em nanômetros[6] (nm). A faixa do espectro vista pelo olho humano sem a ajuda de aparelhos vai de (400 - 750 nm), conforme mostra a FIGURA 12.

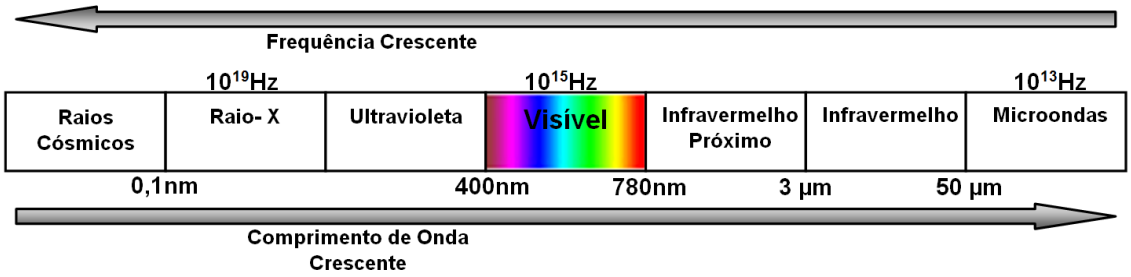


FIGURA 12: ESPECTRO ELETROMAGNÉTICO
FONTE: AUTOR (ADAPTADO DE THEOPHANIDES, 2012)

Ainda, de acordo com Hecht (2002), as moléculas de qualquer material, desde que sua temperatura esteja acima do zero absoluto (-273°C ou 0 Kelvin), irá

[6] Na espectrometria são usadas, em geral, as seguintes unidades de medida: 1Å = 10^{-10} m (angstrom); 1nm = 10^{-9} m (nanômetros); 1µm = 10^{-6} m (micrometros).

irradiar e absorver radiação infravermelha. Pode-se afirmar que o infravermelho é emitido num espectro contínuo por corpos quentes, como: lâmpadas, aquecedores, carvão em brasa, o Sol, animais de sangue quente, etc. Para ter-se uma ideia, o corpo humano irradia infravermelho num comprimento de onda que inicia em 3000 nm com picos na vizinhança de 10.000 nm. A possibilidade de captar essa radiação é o princípio de funcionamento dos detectores de infravermelho, amplamente empregados nas áreas militares e de segurança, bem como, é a ferramenta de sobrevivência de alguns animais com atividades noturnas.

Para Theophanides (2012), a espectroscopia é o estudo da interação de ondas eletromagnéticas com a matéria. Técnicas espectroscópicas baseadas na interação entre luz e matéria têm sido utilizadas para estudar materiais *in vivo*, *ex vivo* ou *in vitro*. Com a espectroscopia no infravermelho é possível obter informações de materiais isolados e biomateriais como: biopolímeros; materiais biológicos; tecido conectivo; células simples; fluidos biológicos em geral, entre outros.

A energia da radiação infravermelha é mensurada, geralmente, por dispositivos que respondem ao calor gerado pela absorção do (IR). Pequenas diferenças na temperatura do objeto e do seu arredor resultam em característica de emissão do (IR) que podem ser usados de muitas formas, como, por exemplo, detecção de tumores e identificação de compostos orgânicos (THEOPHANIDES, 2012).

O funcionamento básico de um espectroscópio, consiste em passar um feixe de luz através de uma solução/amostra e medir a quantidade de energia radiante que foi absorvida ou refletida. O feixe de luz infravermelha produzido pela fonte é dividido em dois, como ilustra a FIGURA 13, sendo que um passa pela amostra a ser analisada e o outro por uma amostra de referência[7]. A radiação incidente sobre a amostra terá uma parcela absorvida e o restante será refletido sobre uma grade de

[7] A amostra de referência é utilizada com o objetivo de prevenir que flutuações da energia elétrica da saída da fonte afetem os resultados finais, uma vez que tanto a amostra quanto a referência são afetadas da mesma forma. Ainda, possibilita prevenir a influência de variações no resultado final devido ao fato de que a fonte não necessariamente emite a mesma intensidade de luz para todos os comprimentos de onda (CIURCZAK, 2005).

difração[8] que os direcionará sobre um detector, fazendo com que os sinais sejam convertidos para o formato digital, possibilitando o processamento por sistemas computacionais. Como fonte de radiação luminosa são utilizadas, em geral, lâmpadas de tungstênio ou deutério (THEOPHANIDES, 2012; HECHT, 2001).

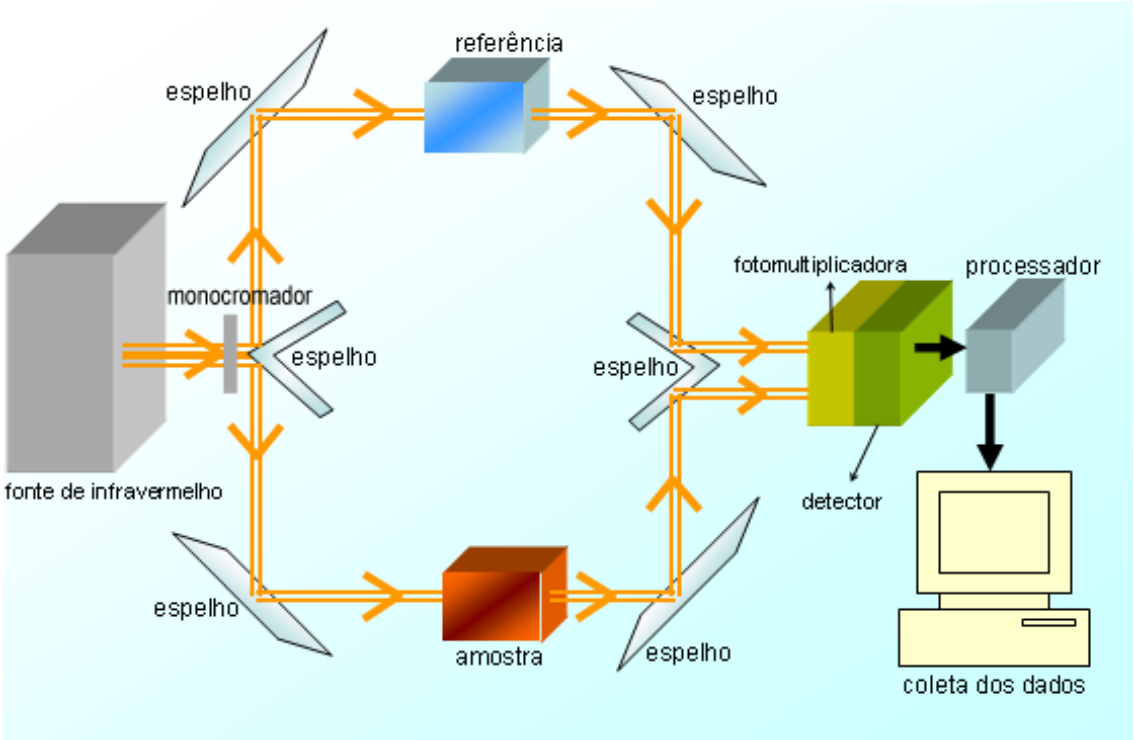


FIGURA 13: APARATO BÁSICO DE ESPECTROSCOPIA.
FONTE: (ESPECTROSCOPIA DE INFRAVERMELHO, 2013).

De acordo com Albrecht *et al*. (2008), a espectroscopia de refletância do infravermelho próximo (NIR) mede a intensidade de absorção da luz infravermelha para uma amostra em uma determinada faixa de comprimento de onda. Os espectros resultantes fornecem um conjunto de característica que compõe uma assinatura única com importantes informações bioquímicas como, por exemplo, o número de grupos funcionais de ligações químicas (-CH, -OH e -NH), como ilustra a FIGURA 14.

[8] A função da grade de difração é separar os diferentes comprimentos de onda do feixe e direcioná-los sobre os fotodetectores. O fotodetector ao ser irradiado produz uma corrente elétrica cuja intensidade depende da magnitude da emissão recebida

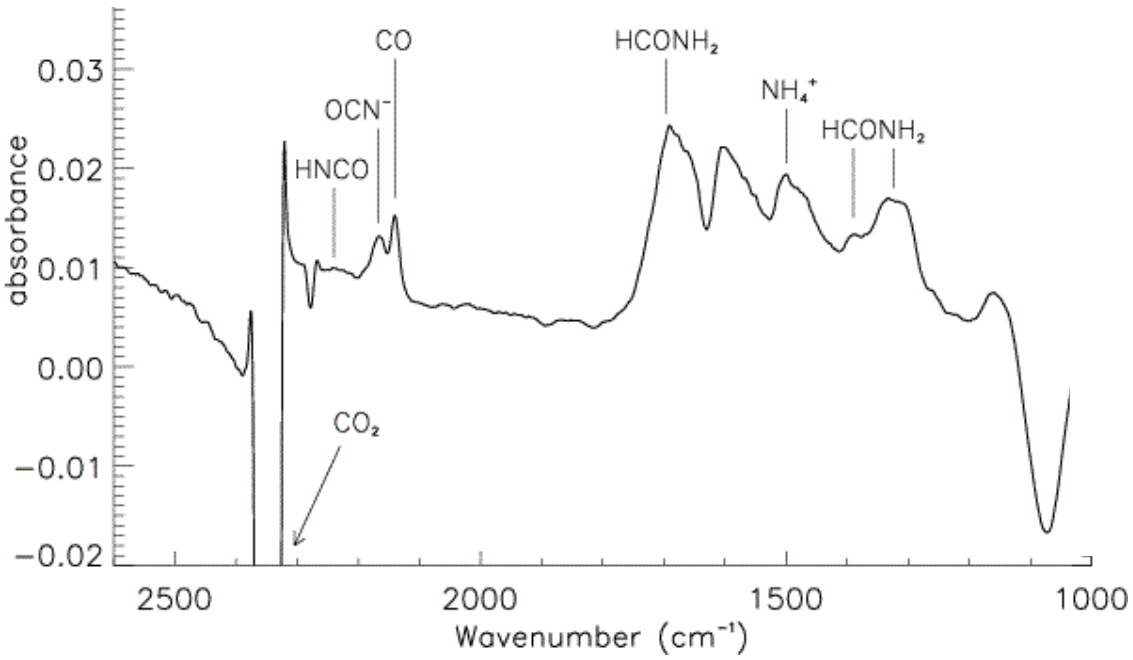


FIGURA 14: ESPECTRO DE UMA PLACA DE ATEROMA EM UMA ARTÉRIA CORONÁRIA, MOSTRANDO AS BANDAS DE ABSORBÂNCIA DE PROTEINAS, AMIDO, ETC.
FONTE: (THEOPHANIDES, 2012)

O objetivo da técnica, em resumo, é incidir radiação eletromagnética sobre um composto orgânico e medir a intensidade de absorção na região do infravermelho próximo. A intensidade da radiação absorvida pode indicar a natureza das moléculas orgânicas e da sua concentração.

Qualquer material pode irradiar e absorver radiação infravermelha via vibração térmica das suas moléculas constituintes. Pode-se entender o processo vibracional de uma molécula, associado à espectrometria do infravermelho, por uma representação em termos dos níveis de energia discreta quantizada E_0, E_1, E_2, etc., como ilustra a FIGURA 15.

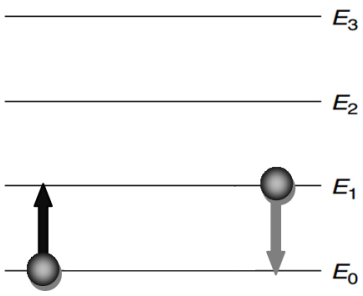


FIGURA 15: REPRESENTAÇÃO DOS NÍVEIS DE ENERGIA DISCRETA QUANTIZADA
FONTE: AUTOR (ADAPTADO DE STUART, 2004)

Um aglomerado de moléculas terá seus átomos distribuídos entre esses diversos níveis de energia. Hipoteticamente, para um determinado átomo na camada E_0, quando a sua molécula interage com a radiação devido à excitação pelos fótons[9], há a emissão ou absorção de um quantum (ou fóton) de energia (seta preta na FIGURA 15), ou seja, há uma mudança para a outra camada, E_1.

Pode-se relacionar a energia do quantum com frequência pela seguinte equação:

$$\Delta \boldsymbol{E} = \boldsymbol{hv} \qquad \textbf{(1)}$$

Onde:

h: constante de Planck, (*h = 6,626 x 10-34 J.s*);

v: frequência;

ΔE: variação de Energia;

Pode-se considerar a frequência de emissão ou absorção de radiação que ocorre na transição entre os níveis de energia E_0 e E_1 como:

$$\boldsymbol{v} = (\boldsymbol{E_1} - \boldsymbol{E_0})/\boldsymbol{h} \qquad \textbf{(2)}$$

O *número de onda*[5]($\bar{v}$), em cm^{-1}, pode ser obtido pela equação (3):

$$\bar{\boldsymbol{v}} = \frac{\boldsymbol{1}}{\boldsymbol{\lambda}} = \boldsymbol{v}/\boldsymbol{c} \qquad \textbf{(3)}$$

Onde:

λ: é o comprimento de onda;

v: é a frequência;

c: a velocidade da luz no vácuo ($c = 2{,}997925 \times 10^8\ m.s^{-1}$);

Associado à captação de energia quantizada absorvida está um mecanismo pelo qual o átomo retorna ao seu estado original pela emissão de um quantum de energia, ou seja, o átomo que estava na camada E_1 poderá retornar para sua camada de origem (seta cinza na FIGURA 15).

[9] O fóton é uma partícula elementar, com característica dúbia (partícula e onda), mediadora da força eletromagnética, presente na luz (HECHT,2001).

CAPÍTULO IV - REFERENCIAL TEÓRICO

4.1 REDES NEURAIS – UMA VISÃO GERAL

Segundo Haykin (2001), as Redes Neurais Artificiais tiveram seu desenvolvimento, inicialmente, entre 1943-1958 quando vários pesquisadores se sobressaíram por suas contribuições. Dentre eles estão:

- *McCulloch & Pitts (1943)* por introduzirem a ideia de Redes Neurais como máquinas computacionais;
- *Hebb (1949)* por postular a primeira regra de aprendizagem auto-organizada;
- *Rosenblatt (1958)* por propor o *Perceptron* como o primeiro modelo para aprendizagem com um professor (aprendizagem supervisionada);

Redes Neurais Artificiais (RNA) ou *Artificial Neural Network* (ANN), representam um sistema de processamento de informação que apresentam algumas características em comum com as Redes Neurais Biológicas. São modelos matemáticos que tentam representar a cognição humana ou o neurônio biológico de forma generalizada (FAUSETT, 1993).

> Uma rede neural é um processador em paralelo maciçamente distribuído, constituído de unidades de processamento simples que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso. Ela se assemelha ao cérebro humano em dois aspectos:
>
> 1. O conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem.
>
> 2. Forças de conexão entre neurônios, conhecidos como pesos sinápticos, são utilizados para armazenar o conhecimento adquirido (HAYKIN, 2001).

De acordo com Boçois (2012), os modelos matemáticos que levaram ao desenvolvimento das Redes Neurais consideraram os seguintes atributos:

- Processamento de informação em elementos simples chamados neurônios;
- Os sinais são transmitidos entre os neurônios por links de conexão;
- Para cada link de conexão existe um peso associado, o qual, em uma Rede Neural típica, multiplica os sinais transmitidos;
- Cada neurônio aplica uma função de ativação (não necessariamente linear) na sua camada de entrada para determinar seu sinal de saída.

O processo de aprendizagem de uma RNA, conhecido como algoritmo de aprendizagem, tem a função de modificar os pesos sinápticos da rede de uma forma ordenada para que ela atinja o objetivo estabelecido (HAYKIN, 2001).

Ainda segundo Haykin (2001), o poder computacional de uma rede neural provém de sua estrutura de processamento distribuída em paralelo e sua capacidade de generalização. Considera-se generalização como o fato da RNA produzir saídas adequadas para entradas que não estavam presentes no seu treinamento (fase de aprendizagem). Pode-se destacar algumas das suas principais características:

- *Não-linearidade:* permitindo a um neurônio adequar-se a sinais de entrada lineares ou não-lineares;
- *Mapeamento de Entrada-Saída:* por meio de amostras de treinamento os pesos sinápticos de uma Rede Neural Artificial são modificados de forma a minimizar a diferença entre a resposta desejada e a resposta real da rede (sinal de saída), produzida pelo sinal de entrada. O processo de modificação dos pesos acontece até que a rede alcance um estado estável onde não haja mais modificações significativas nos pesos sinápticos;
- *Adaptabilidade:* Redes Neurais Artificiais possuem a habilidade de adaptar seus pesos sinápticos a modificações do meio ambiente, podendo, facilmente, ser retreinada para lidar com modificações nas condições operativas iniciais;
- *Resposta a Evidências:* uma rede neural pode fornecer informação não somente sobre qual padrão particular selecionar, mas também sobre a confiança ou crença na decisão tomada;
- *Tolerância a Falhas:* Redes Neurais têm o potencial de ser inerentemente tolerante a falhas, sendo que seu desempenho se deteriora suavemente sob condições de operação desfavoráveis.

As redes neurais possuem inspiração na neurobiologia, na seção 4.1.1 será feita uma breve descrição do cérebro humano.

4.1.1 O cérebro humano

Haykin (2001), o cérebro é descrito com uma estrutura formada por três camadas, conforme mostra a FIGURA 16.

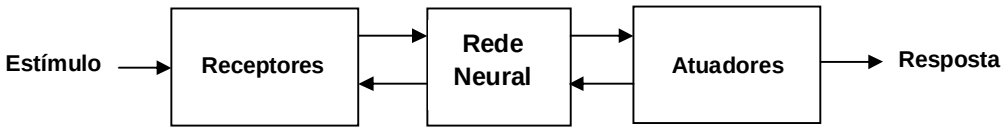


FIGURA 16: REPRESENTAÇÃO EM DIAGRAMA EM BLOCOS DO SISTEMA NERVOSO.
FONTE: (HAYKIN, 2001)

Imaginando-se, de forma hipotética, que um pedestre encontra uma onça pintada caminhando sobre uma calçada, o cérebro do pedestre, representado pela Rede Neural (nervosa), recebe uma informação visual através dos olhos (receptores), percebe-a e toma a decisão apropriada, devido à aprendizagem, que nessa situação poderia ser “procurar um abrigo”. O cérebro envia um sinal para o restante do corpo, indicando uma transmissão para frente, e os receptores (órgãos motores, glândulas hormonais, etc.) retornam a informação para o cérebro, realizando a realimentação do sistema, notificando que estão prontos (ou não) para a tarefa proposta. Os atuadores executam a conversão dos impulsos elétricos gerados pela rede neural em respostas discerníveis como saídas do sistema.

A ideia atualmente aceita dos neurônios como constituintes estruturais do cérebro foi proposta inicialmente pelo médico e histologista espanhol Santiago Ramón y Cajal em seu trabalho “Histologie du système nerveux de l’homme et des vertebres”[10] publicado em 1911 (MORENO-MARTINEZ; MARTIN-ARAGUZ, 2002; HAYKIN, 2001).

Em comparação com as portas lógicas em silício, os neurônios biológicos são mais lentos. Os eventos em um circuito de silício acontecem na ordem de nanosegundos (10^{-9}s), enquanto que eventos em um sistema neural biológico acontecem na ordem de milissegundos (10^{-3}s). Para compensar a lenta taxa de operação de um neurônio biológico, o cérebro possui um número gigantesco de células nervosas, com conexões maciças entre si e trabalhando paralelamente (HAYKIN, 2001; HAGAN *et. al.*, 1996).

[10]Tradução: Histologia do sistema nervoso do homem e dos vertebrados. (AUTOR)

No cérebro humano existe aproximadamente 86 bilhões de neurônios, sendo que o córtex cerebral[11] abriga 16 bilhões de neurônios e o cerebelo[12] 69 bilhões (ZORZETTO, 2012).

Considerado pelas suas características, o cérebro é uma estrutura altamente eficiente e complexa. Segundo Haykin (2001), a eficiência energética do cérebro é de aproximadamente 10^{-16} joules (em operações/segundo), em contrapartida os melhores computadores apresentam cerca de 10^{-6} joules (em operações/segundo), a diferença de consumo energético é muito significativa. De acordo com o artigo de Fischetti (2011), o supercomputador mais poderoso do mundo, até meados de 2012, era o K, da empresa Fujitsu (Japão). Este supercomputador é quatro vezes mais rápido e possuí uma capacidade de armazenamento dez vezes superior ao cérebro humano, no entanto, necessitava de 9,89MW (consumo equivalente ao de 10.000 casas populares) para funcionar enquanto o cérebro humano requer apenas 20W. Analisando de modo comparativo, o K consome o mesmo que 9,9 milhões de lâmpadas LED e o cérebro apenas 20 - sendo que a diferença de processamento é pequena quando comparada ao consumo. A FIGURA 17 ilustra um cérebro humano em relação ao supercomputador K.

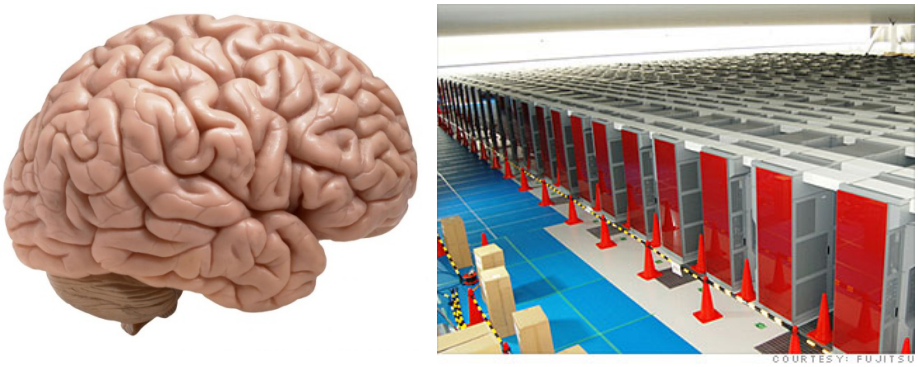


FIGURA 17: ILUSTRAÇÃO COMPARATIVA ENTRE O CÉREBRO HUMANO E O SUPERCOMPUTADOR K.
FONTES: (FUJITSU & WISEGEEK)

A plasticidade do sistema nervoso permite a sua adaptação ao seu meio mediante a criação de novas conexões sinápticas[13] e as modificações das sinapses

[11] Principal responsável por funções cognitivas como atenção, memória e linguagem(ZORZETTO, 2012).
[12] Responsável pela manutenção do equilíbrio e pelo controle do tônus muscular e dos movimentos voluntários, bem como pela aprendizagem motora (ZORZETTO, 2012).

existentes. Os neurônios possuem em torno de 1.000 e 10.000 conexões (sinapses) com os neurônios adjacentes. Podem-se dividir essas células fundamentais do tecido nervoso em três seções principais: o corpo da célula, os dendritos e o axônio, FIGURA 18, cada uma com funções específicas, porém complementares. Os dendritos são uma rede receptiva de fibras nervosas que carregam sinais elétricos para dentro do corpo celular. O corpo celular soma estes sinais e o axônio, que é uma comprida fibra nervosa, transmite o sinal para outros neurônios. O ponto de contato entre o axônio de uma célula e o dendrito de outra é chamado de *sinapse* (BOÇOIS, 2012; HAYKIN, 2001; HAGAN *et. al.*, 1996).

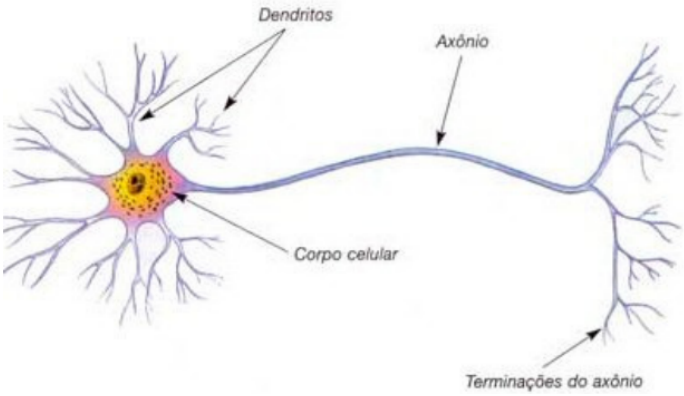


FIGURA 18: NEURÔNIO BIOLÓGICO.
FONTE: (BOÇOIS, 2012)

Em termos de transmissão de sinais o neurônio codifica suas saídas como uma série de pulsos breves de tensão, sendo estes usualmente conhecidos como *potenciais de ação* ou *impulsos*. Os *potenciais de ação* são originados no corpo celular do neurônio ou perto dele, propagando-se através de cada neurônio a uma velocidade e amplitude constantes obedecendo a uma hierarquia de níveis entrelaçados de organização que vai do *sistema nervoso central* até as *moléculas* (HAYKIN, 2001; FAUSETT, 1993).

Ainda não foi possível recriar a organização do cérebro biológico. Os neurônios artificiais que se utilizam para construir as Redes Neurais Artificiais são, sem dúvida, primitivos quando comparados às Redes Neurais Biológicas. No

[13] Sinapses: São unidades estruturais e funcionais elementares que medeiam as interações entre os neurônios (HAYKIN, 2001).

entanto, com a analogia neurobiológica como inspiração e uma variedade de ferramentas tecnológicas e teóricas é possível acreditar que os avanços nessas áreas serão significativos nas próximas décadas (HAYKIN, 2001).

4.2 REDES NEURAIS ARTIFICAIS

Uma Rede Neural Artificial é como uma coleção de neurônios artificiais disposta de forma a configurar um aspecto específico. Um neurônio artificial é uma unidade de processamento de informação fundamental para a operação de uma Rede Neural Artificial. Os neurônios biológicos, no cérebro, comunicam-se continuamente e em paralelo, transferindo informações através de sinapses, sendo que as novas informações são recebidas pelos dendritos e transmitidas pelos axônios. Tentando simular estas funções, as Redes Neurais Artificiais buscam programar sua dinâmica e comportamento básico, como ilustra a FIGURA 19 (BOÇOIS, 2012).

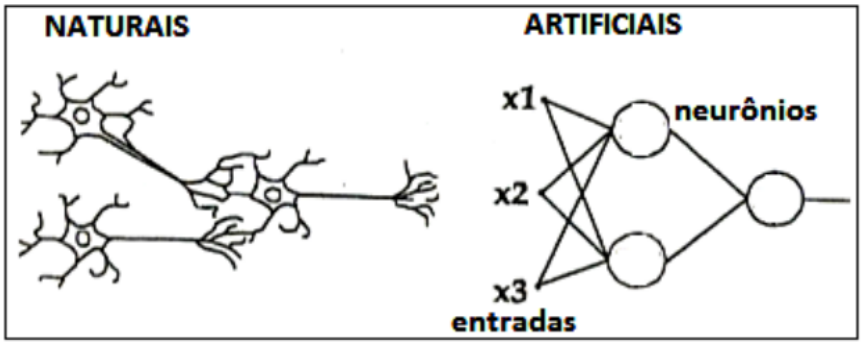


FIGURA 19: ANALOGIA ENTRE OS NEURÔNIOS NATURAIS E OS NEURÔNIOS ARTIFICIAS.
FONTE: (BOÇOIS, 2012)

Podem-se comparar as entradas de uma Rede Neural Artificial com os estímulos recebidos pelos neurônios biológicos e todos esses estímulos são encaminhados às unidades de processamento paralelamente. O poder computacional de uma Rede Neural, tanto a artificial quanto a biológica, está em sua arquitetura extremamente distribuída de forma paralela e, também, de sua habilidade em aprender e sua capacidade de generalizar (BOÇOIS, 2012).

A principal propriedade em uma Rede Neural é a sua capacidade de aprender a partir do meio e de aperfeiçoar o seu desempenho por meio da aprendizagem. A Rede Neural Artificial aprende a respeito do seu ambiente por meio

de um processo iterativo de ajustes aplicados a seus pesos sinápticos e níveis de bias. De forma genérica pode-se dizer que aprendizagem é um procedimento onde os parâmetros livres de uma RNA são adaptados através de estímulos oriundos do ambiente onde a rede está inserida, sendo que o tipo de aprendizagem é determinado pelo modo como ocorre a modificação dos padrões (parâmetros). Esses ajustes fazem com que a rede torne-se mais instruída sobre seu ambiente após cada iteração do processo de aprendizagem (HAYKIN, 2001; MENDES, 2008).

Segundo HAYKIN (2001), o *conhecimento* se refere à informação armazenada ou a modelos utilizados por uma pessoa ou máquina como forma de interpretação, previsão e reposta apropriada a situações do mundo exterior. A representação do conhecimento possui duas características principais, que são:

- A informação realmente torna-se explícita;
- A informação é codificada fisicamente para um uso subsequente.

Em aplicações do mundo real de “máquinas inteligentes” uma solução satisfatória depende de uma boa representação do *conhecimento*. O mesmo vale para as Redes Neurais Artificiais, pois representam uma forma de “máquina inteligente”, tanto que boa parte do desenvolvimento dessa área provém das engenharias. Deve-se considerar, também, que os parâmetros internos da rede para representação do *conhecimento* são muito diversificados, o que torna o desenvolvimento de uma solução satisfatória algo trabalhoso (HAGAN *et al.*,1996; HAYKIN, 2001).

A aprendizagem ocorre durante o treinamento da rede. Nesta etapa é apresentado à RNA um conjunto de dados (padrões de entrada), denominado conjunto de treinamento. O conjunto treinamento, em geral, recebe entre 50 e 90% do total de padrões (amostras) existentes, sendo que o restante será separado para teste e/ou validação, no caso de processos que utilizam a *validação cruzada*[14]. O subconjunto com os dados de validação é utilizado durante a fase de treinamento da rede, já o subconjunto de teste é apresentado à rede apenas na fase final (fase de teste), para averiguar a capacidade de generalização da rede, ou seja, o quanto ela "aprendeu" (BOÇOIS, 2012; TATIBANA; KAETSU, 2009).

[14] Validação Cruzada: É um critério de parada com o objetivo de impedir grande discrepância entre os erros de treinamento e teste.

De acordo com Boçois (2012), as amostras devem ser fornecidas à rede de modo aleatório, para estimular que ela aprenda as regras sem ser influenciada por um padrão sequencial. Segundo Haykin (2001), nos problemas de classificação de padrões por meio da aprendizagem com retropropagação, é conveniente que exista uma aleatoriedade (baralhamento) das amostras apresentadas à rede de múltiplas camadas, de uma iteração (época) para a seguinte, evitando que exemplos (amostras) sucessivos apresentados à rede em uma época pertençam à mesma classe.

De acordo com Boçois (2012), os seguintes passos podem ser seguidos para o desenvolvimento de aplicações envolvendo redes neurais:

- dos dados coletados, separar um conjunto para o treinamento e outro para o teste, se necessário tomar outro conjunto para validação;

- determinar a configuração da rede, ou seja, a topologia a ser utilizada e a determinação de parâmetros do algoritmo de treinamento;

- treinar a rede (nessa etapa ocorrem as modificações nos parâmetros livres como resultado da estimulação);

- testar a rede (nessa etapa é possível avaliar o desempenho da rede);

Pode-se sintetizar o processo de aprendizagem do seguinte modo: apresenta-se à rede uma amostra aleatória do conjunto de treinamento e alteram-se os parâmetros livres (pesos e bias) com o objetivo de diminuir a diferença entre a resposta desejada e a obtida pela rede, segundo o critério estatístico adotado. Repete-se o treinamento para todas as entradas (amostras) do *conjunto de treinamento* até que a rede atinja um ponto onde não ocorram modificações significativas nos pesos sinápticos.

4.2.1 Processo de treinamento

O diagrama mostrado na FIGURA 20 ilustra o modelo de uma rede neural artificial com duas camadas, sendo o LAYER 1 a camada intermediária e o LAYER 2 a camada de saída. Considera-se as camadas 1 e 2 compostas por apenas 1 neurônio artificial (HAGAN *et al.*, 1996).

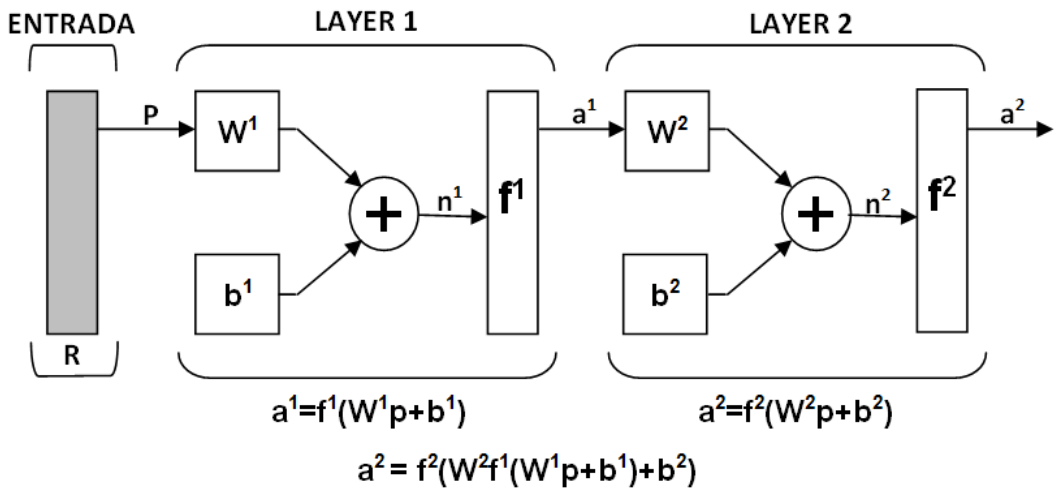


FIGURA 20: REPRESENTAÇÃO ESQUEMÁTICA DE UMA RNA.
FONTE: AUTOR (ADAPTADO DE HAGAN *et al.*, 1996)

Considerando:

R : representa o conjunto dos sinais de entrada [$p_1, p_2, \ldots, p_R$];

W^m: representa o conjunto de pesos para a camada *m* em relação ao neurônio na posição *i*, com $W^m = [w_1^m, w_2^m, \ldots, w_i^m]$;

b^m : bias;

f^m : função de ativação na camada *m*;

$\oplus$: representa o *somador*;

Uma entrada escalar ***p*** é multiplicada pelo peso ***w*** (escalar) para formar o sinal ***wp***, que é um dos termos enviados ao somador. O outro termo é o bias, ***b***, que também é passado ao somador. Em geral, alguns autores consideram que o bias é multiplicado por uma entrada constante de valor ***1***. A saída do somador, representada por ***n***, é transmitida para a função de transferência ***f***, a qual produz uma saída escalar ***a*** (*a = f (n)*). Há autores que chamam a *função de ativação* como *função de transferência* ou *função restritiva*, cuja função é restringir a amplitude do sinal de saída do neurônio, e o *bias* como *compensador* ou *equilibrador*.

Fazendo uma analogia com o neurônio biológico, Figuras 18 e 19, pode-se entender o peso ***w*** como a força do sinal de sinapse e o corpo celular como o conjunto somador e função de transferência.

O neurônio artificial representado pelo LAYER 1 possui ***R*** terminais de entrada [$p_1, p_2, ..., p_R$], que podem ser considerados como os dendritos do neurônio biológico, e apenas um terminal de saída, a^m (o axônio). A segunda camada, LAYER 2, possui apenas um terminal para os sinais entrada, a^1, e outro para os sinais de saída, a^2. O comportamento das sinapses é representado pelos pesos $[w_1^1, w_2^1, \dots, w_M^1]$ interligados com as entradas dos neurônios em cada camada, cujo valor pode ser excitatório ou inibitório (HAYKIN, 2001; BISHOP, 2006; HAGAN *et al.*,1996).

De forma genérica, pode-se considerar que o sinal de entrada $\boldsymbol{p_j}$ conectado ao neurônio ***i*** é multiplicado pelo peso[15] sináptico $\boldsymbol{w_{ij}}$, onde o índice **i** representa o neurônio recebedor da camada **m** e o índice **j** indica o neurônio fornecedor do sinal (camada **m-1**), resultando numa *sinapse* ou *elo de conexão*. O *somador* representa o somatório do produto entre os sinais de entrada [$p_1, p_2, ..., p_R$] e as sinapses $[w_1^m, w_2^m, \dots, w_M^m]$ adicionado ao bias. Para as camadas intermediárias os sinais de entrada serão $[a_1^m, a_2^m, \dots, a_n^m]$ no lugar de [$p_1, p_2, ..., p_R$]. O resultado do *somador* tem seu valor modificado pela *função de ativação*, produzindo o sinal de saída do neurônio, cuja amplitude pertence, geralmente, ao intervalo unitário fechado [0,1] ou [-1,1] (HAYKIN, 2001; BISHOP, 2006).

O modelo neuronal, conforme ilustra a FIGURA 20, também inclui um bias, representado por b^m, cuja função é aumentar ou diminuir a entrada líquida da função de ativação. Pode-se representar matematicamente um neurônio artificial utilizando as seguintes equações:

$$\boldsymbol{n_i^m} = \left(\sum_{j=1}^{S^{m-1}} \boldsymbol{w_{i,j}^m} \cdot \boldsymbol{a_j^{m-1}}\right) + \boldsymbol{b^m} \quad \textit{(somador, soma ponderada, função soma)} \qquad \textbf{(4)}$$

$$\boldsymbol{a^{m+1}} = \boldsymbol{f^{m+1}}\left(\boldsymbol{n_i^{m+1}}\right) = \boldsymbol{f^{m+1}}\left(\boldsymbol{W^{m+1}a^m} + \boldsymbol{b^{m+1}}\right) \quad \textit{(sinal de saída do neurônio)} \qquad \textbf{(5)}$$

Quando os sinais de entrada [$p_1, p_2, ..., p_R$] ou [$a_1^m, a_2^m, \dots, a_n^m$] são apresentados aos neurônios, ocorre a sua multiplicação pelos pesos sinápticos respectivos e a soma de todos esses resultados por $\boldsymbol{n_i^m}$, representando o sinal de excitação do neurônio. O *somador* resultante, representado pela equação (4), recebe o nome de *somador*, *soma ponderada* ou *função soma*. Os pesos

[15] os pesos iniciais são obtidos de forma aleatória.

$[w_1^m, w_2^m, \dots, w_M^m]$ representam o grau de importância que determinada entrada possui em relação aquele determinado neurônio (BOÇOIS, 2012).

Determinado o valor da *função soma,* $\boldsymbol{n_i^m}$, o neurônio deve restringir o resultado a um intervalo permissível de amplitude, o que é feito pela *função de ativação*.

A *função de ativação* é importante para o comportamento de uma Rede Neural Artificial, pois define a saída do neurônio artificial e, portanto, o caminho pelo qual a informação é conduzida. As *funções de ativação* mais comuns são as gaussianas, sigmóides, senoidais, lineares, tangentes hiperbólicas sigmoidais, logarítmicas sigmoidais, degrau, entre outras.

A FIGURA 21 ilustra as *funções de ativação* mais utilizadas.

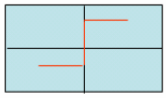
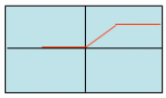
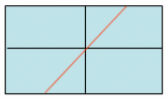
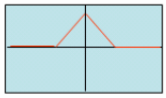
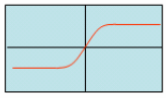
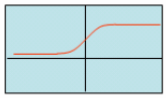
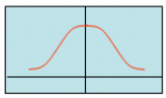
Função	Gráfico	Comando Matlab	Equação
Degrau		hardlim(n)	se n>=0 => y=1 senão y=-1
Linear c/ saturação		satlin(n)	se n < 0 => y = 0 se n > 1 => y = 1 se 0< = n <= 1 => y = n
Linear		purelin(n)	y = n
triangular		tribas(n)	se -1 <= n <= 1 => y = 1-abs(n) senão y=0
Tangente hiperbólica		tansig(n)	$y = \frac{2}{(1+e^{-2n})} - 1$
Logarítmica		logsig(n)	$y = \frac{1}{(1+e^{-n})}$
Base radial		Radbas(n)	$y = e^{-n}$

FIGURA 21: FUNÇÕES DE ATIVAÇÃO
FONTE: (LIMA, 2006)

Após o sinal ser transformado pela *função de ativação* ocorre o ajuste sináptico sobre os pesos [$w_1^m, w_2^m, \ldots, w_M^m$]. Este processo constitui o armazenamento do "conhecimento", do fato apresentado, conjuntamente em cada neurônio. Um neurônio individual não guarda o conhecimento todo em si, mas faz parte de uma rede, envolvendo todos os neurônios com a função de reter a informação (BOÇOIS, 2012).

De forma geral, entende-se o ajuste sináptico sobre pesos e bias como o resultado de um cálculo matemático cujo objetivo é somar ao peso atual um valor que corresponde à quantidade de erro gerada pela rede, corrigindo seu valor e produzindo um *sinal de saída* mais próximo do desejado. Uma das técnicas mais conhecidas para a atualização dos pesos é a *Regra Delta*, onde o valor do novo peso é determinado, segundo Haykin (2001), do seguinte modo:

$$\boldsymbol{w^m(k+1) = w^m(k) + \alpha * e_k * p_k} \qquad \textbf{(4)}$$

Considerando:

$\boldsymbol{w^m(k+1)}$: valor atualizado do peso para iteração (k+1);

$\boldsymbol{w^m(k)}$: valor anterior do peso, iteração (k);

$\boldsymbol{\alpha}$: constante de aprendizagem;

$\boldsymbol{e_k}$: erro = (*obtido - desejado*);

$\boldsymbol{p_k}$: sinal de entrada;

A constante de aprendizagem, também conhecida como taxa de aprendizagem, tem a função de determinar o comprimento dos passos em direção ao ponto de mínimo sobre a superfície de erro, por exemplo: a distância entre os pontos nas Figuras 22 e 23. Em geral a taxa de aprendizagem é um valor entre 0 e 1, podendo ser fixa ou variável. Segundo Bishop (2006), a taxa de aprendizagem tem grande influência no comportamento de uma Rede Neural Artificial. Valores pequenos tendem a deixar a rede mais estável embora com a aprendizagem mais lenta. Uma taxa de aprendizagem com valores próximos de 1 tendem a ser mais rápidas embora instáveis, ou seja, com mais oscilações sobre a superfície de erro até encontrar o mínimo. As Figuras 22 e 23 ilustram o comportamento do método do

gradiente descendente para uma função quadrática com taxa de aprendizagem, $\boldsymbol{\alpha}$, igual a 0,038 e 0,030.

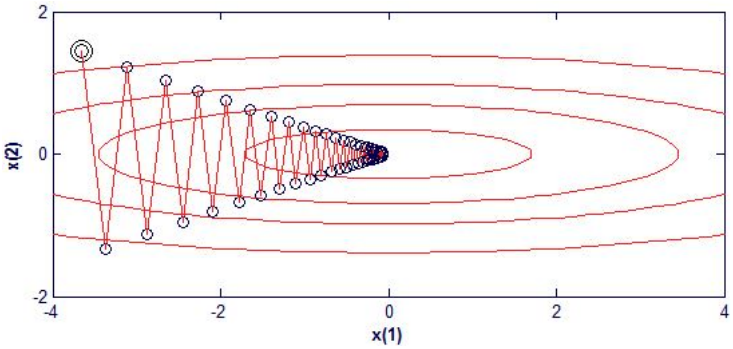


FIGURA 22: TRAJETÓRIA PARA O GRADIENTE DESCENDENTE COM $\boldsymbol{\alpha} = \mathbf{0,038}$
FONTE: AUTOR (ADAPTADO DE HAGAN *et al.,* 1996)

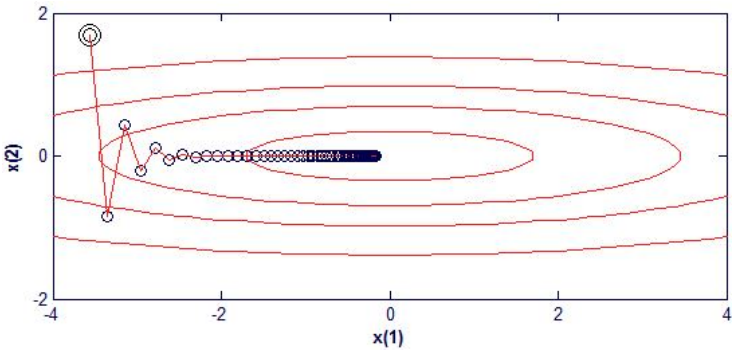


FIGURA 23: TRAJETÓRIA PARA O GRADIENTE DESCENDENTE COM $\boldsymbol{\alpha} = \mathbf{0,030}$
FONTE: AUTOR (ADAPTADO DE HAGAN *et al.* (1996))

Pode-se observar pelas figuras que uma pequena alteração na taxa de aprendizagem é capaz de produzir uma modificação significativa na trajetória em busca do ótimo.

4.2.2 Arquitetura de uma rede neural artificial

Segundo Boçois (2012), a maneira pela qual os neurônios de uma RNA estão estruturados, sua arquitetura, está diretamente relacionada ao algoritmo de aprendizagem usado para treinar a rede. Uma rede neural organizada em camadas é dita rede neural em camadas ou múltiplas camadas. Essas camadas referem-se às entradas apresentadas à rede (camada de entrada), as saídas fornecidas pela rede

(camada de saída) e, em alguns casos, há as camadas ocultas, que possuem neurônios ocultos com a função de intervir entre a camada de entrada e a de saída. Adiciona-se uma ou mais camadas ocultas para tornar a rede capaz de extrair estatísticas de ordem elevada embora um número grande de camadas ocultas não signifique deixar o algoritmo mais eficiente. Quando uma rede possui todos os neurônios de uma camada conectados a todos os neurônios da camada adjacente seguinte, a rede é dita totalmente conectada.

Em geral pode-se identificar três classes diferentes de arquitetura de rede neural: redes alimentadas adiante com camada única, redes alimentadas adiante com múltiplas camadas e redes recorrentes.

- *Redes alimentadas adiante com camada única*: Na sua forma mais simples, como representa a FIGURA 24, possuem a camada de entrada que se projeta na camada de saída, mas não o contrário. Ela é estritamente alimentada adiante ou acíclica e com camada única, ou seja, possui apenas a camada de saída, que contem os neurônios de computação. A camada de entrada não é contada, pois nela não é realizada qualquer computação.

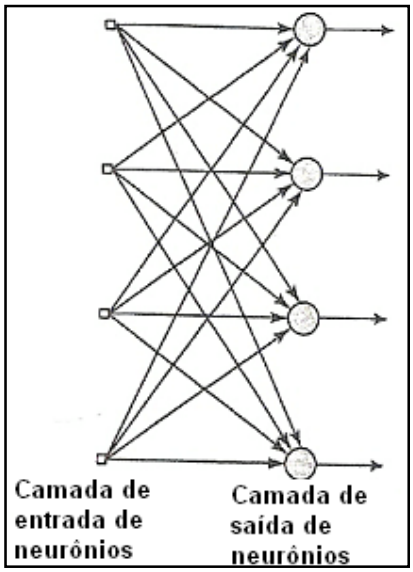


FIGURA 24: REDE ALIMENTADA ADIANTE COM UMA ÚNICA CAMADA
FONTE: (HAYKIN, 2001)

- *Redes alimentadas adiante com múltiplas camadas*: Se distinguem da classe anterior pela presença de camadas ocultas, que podem variar de um a n camadas, FIGURA 25. Os neurônios da camada de entrada fornecem informações para a 1ª camada oculta. Esta computa as informações e as saídas servem como entrada para a 2ª camada oculta e assim sucessivamente para o restante da rede. O conjunto das saídas da última camada (camada de saída) constitui a resposta global da rede para o padrão fornecido.

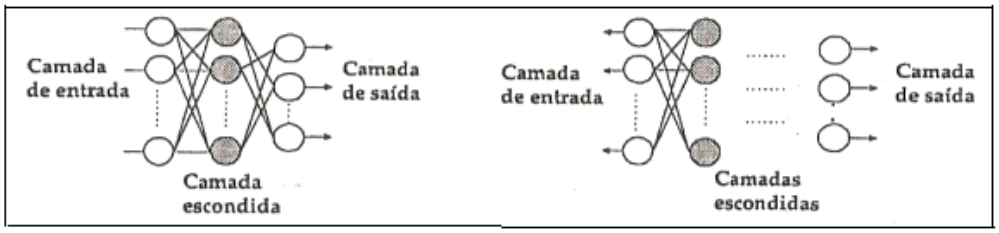


FIGURA 25: REDES ALIMENTADAS ADIANTE COM MÚLTIPLAS CAMADAS
FONTE: (TAFNER *et al.*, 1995)

- *Redes recorrentes*: Diferente das outras, possuem pelo menos um laço de realimentação, ou seja, o sinal de saída de um neurônio serve como entrada para a mesma camada, uma camada anterior ou entre camadas alternadas, FIGURA 26. Essa conexão (realimentação) pode ser realizada tanto para frente quanto para trás.

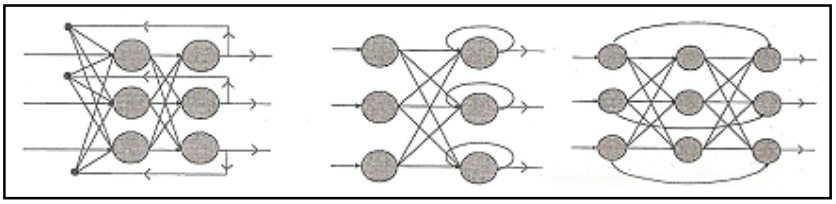


FIGURA 26: REDES RECORRENTES
FONTE: (TAFNER *et al.*, 1995)

4.3 MÉTODO DE APRENDIZAGEM *BACKPROPAGATION*

A demonstração das limitações de uma rede neural de camada única, pelo trabalho de Minsky e Papert (1969), foi o fator decisivo para o declínio do interesse no estudo de redes neurais na década de 1970. Embora vários pesquisadores tenham abandonado suas pesquisas em redes neurais, outros continuaram seus trabalhos e descobriram novos métodos mais eficientes, envolvendo redes neurais de múltiplas camadas, dando um novo incentivo para estudos nessa área a partir da década de 1980. Dentre os pesquisadores que persistiram no desenvolvimento das redes neurais artificiais, aqueles cujos trabalhos tiveram maior disseminação foram RUMELHART *et al.* (1986), McCLELLAND e RUMELHART (1988), HOPFIELD (1984) entre outros. Esses trabalhos mostraram a viabilidade da utilização de redes neurais para a solução de um grande número de problemas (FAUSSET, 1995).

O método de treinamento conhecido como "*backpropagation of errors*" (retropropagação de erros), é um melhoramento do método *perceptron de camada única*. A técnica consiste na aplicação do método do gradiente descendente, que é uma técnica de otimização irrestrita, de forma apropriada a minimizar o erro computado pela rede.

4.3.1 Gradiente Descendente (Descida mais íngreme)

O treinamento de uma RNA utilizando o método backpropagation envolve três estágios: a propagação dos padrões de entrada; a retropropagação do erro associado; o ajuste dos parâmetros pesos e bias. Esse processo pode ser ilustrado conforme a FIGURA 27.

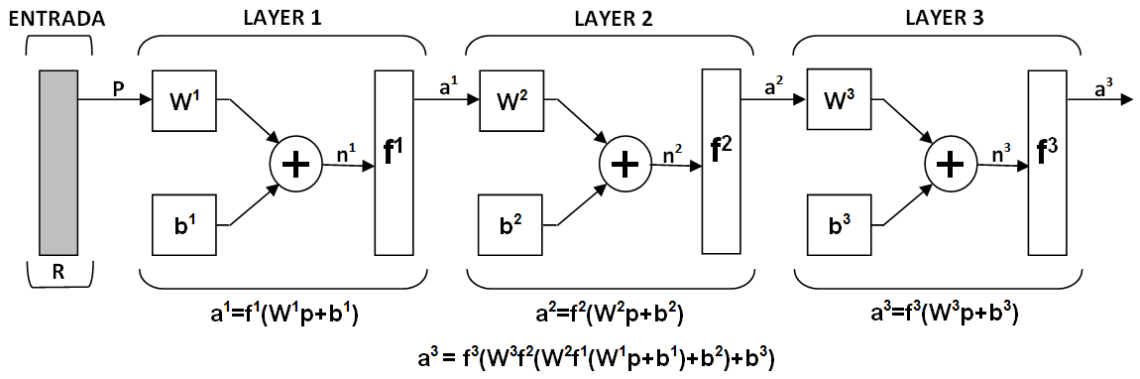


FIGURA 27: REPRESENTAÇÃO ESQUEMÁTICA DE UMA RNA COM 3 CAMADAS
FONTE: AUTOR (adaptado de HAGAN *et al.*, 1996)

Onde:

R: Conjuto de treinamento composto por [p_1, p_2, ..., p_R], relativo à uma amostra (padrão);

{*P, t*}: os sinais de entrada, para o primeiro layer, e saída desejada;

w, b: são pesos e bias respectivamente;

***n*:** representa o somatório do produto entre pesos e entradas adicionado ao bias;

***f*:** representa a função de ativação adotada;

a: representa a aplicação da função de ativação sobre n^k, bem como é o sinal de entrada para as camadas intermediárias e de saída;

Para redes com múltiplas camadas a saída de uma camada torna-se o sinal de entrada para o *layer* (camada) seguinte. O sinal é tratado pela função de ativação, ***f***, e pode ser expresso como:

$$a^{m+1} = \boldsymbol{f}^{m+1}(W^{m+1}a^m + b^{m+1}) \qquad \textbf{(6)}$$

Onde m = 0,1,..., M-1[16].

Para "ensinar" a rede é necessário treiná-la mediante um conjunto de dados **Q**, que está associado a sua respectiva saída, {*P, t*}. Como o processo é iterativo, os pesos e bias iniciais são modificados a cada iteração (*k*) com o intuito de minimizar, gradativamente, o erro quadrático médio (7) (HAGAN; MENHAJ, 1994).

$$\mathcal{E}(\boldsymbol{w}) = \frac{1}{Q}E(\boldsymbol{e_k}^2) = \frac{1}{Q}E(\boldsymbol{t_k} - \boldsymbol{a_k})^2 \qquad \textbf{(7)}^{17}$$

Onde ***w*** é o vetor de pesos e bias, ***Q*** a quantidade de padrões de entrada e ***t*** é o valor desejado de saída para um determinado padrão. Se a rede possuir várias saídas pode-se generalizar a equação (7) para forma:

[16] Considerar *m* como indicativo da camada e *M* como sendo a última camada (camada de saída).
[17] Alguns autores consideram o *Q*, durante o cálculo do *Mean Square Error* (MSE*)*, como o produto das dimensões do vetor erro.

$$\mathcal{E}(\boldsymbol{w}) = \frac{1}{Q}\boldsymbol{E}(\boldsymbol{e_k}^{\boldsymbol{T}}\boldsymbol{e_k}) = \frac{1}{Q}\boldsymbol{E}((\boldsymbol{t_k} - \boldsymbol{a_k})^{\boldsymbol{T}}(\boldsymbol{t_k} - \boldsymbol{a_k})) \quad \textbf{(8)}$$

Formalmente, segundo Haykin (2001) e Hagan *et al.* (1996), pode-se descrever o algoritmo do método do gradiente descendente como:

$$\boldsymbol{w_{i,j}^m}(\boldsymbol{k}+\mathbf{1}) = \boldsymbol{w_{i,j}^m}(\boldsymbol{k}) - \boldsymbol{\alpha}\frac{\partial\mathcal{E}}{\partial \boldsymbol{w_{i,j}^m}} \quad \textbf{(9)}$$

$$\boldsymbol{b^m}(\boldsymbol{k}+\mathbf{1}) = \boldsymbol{b^m}(\boldsymbol{k}) - \boldsymbol{\alpha}\frac{\partial\mathcal{E}}{\partial \boldsymbol{b^m}} \quad \textbf{(10)}$$

Onde $\boldsymbol{\alpha}$ é a taxa de aprendizagem e $\boldsymbol{w_{i,j}^m}$ representa o peso do neurônio ***i*** no layer ***m***, com sinal vindo do neurônio ***j*** da camada anterior. Já o $\boldsymbol{b^m}$ representa o bias da camada ***m***. Alguns autores utilizam a notação $\boldsymbol{b_i^m}$ para indicar o bias do neurônio ***i*** na camada ***m***.

As equações (9) e (10) necessitam do cálculo das derivadas parciais. Para redes de múltiplas camadas o erro, $\mathcal{E}$, não é uma função explícita dos pesos nas camadas escondidas, portanto essas derivadas não são calculadas tão facilmente.

Segundo Haykin (2001), a técnica de *backpropagation* é apropriada para implementar o método da *descida do gradiente* considerando o espaço de pesos de uma rede com múltiplas camadas e retro-alimentação.

De acordo com Hagan *et al.* (1996), como o erro é uma função indireta dos pesos nas camadas escondidas, lembrando que o erro mencionado é relativo à iteração, pode-se utilizar a *regra da cadeia* para obter a derivada. A ideia é utilizar uma função conhecida como alternativa para descobrir as derivadas implícitas. Considera-se:

$$\boldsymbol{n_i^m} = \sum_{\boldsymbol{j=1}}^{\boldsymbol{S^{m-1}}} \boldsymbol{w_{i,j}^m} \cdot \boldsymbol{a_j^{m-1}} + \boldsymbol{b^m} \quad \textbf{(11)}$$

Onde,

S^m : índice que expressa o número de neurônios na camada *m* ;

n_i^m : representa o somatório do produto entre pesos e entradas, adicionado o bias, do neurônio *i* em relação ao *layer m*;

$w_{i,j}^{m}$: representa o peso do neurônio *i* no *layer m*, com sinal vindo do neurônio *j* da camada anterior, (m-1);

a_j^{m-1}: representa a aplicação da função de ativação sobre n_i^{m-1}, também pode-se entender como o sinal de saída da camada anterior e sinal de entrada para a atual camada.

Assim, pode-se expressar o segundo termo das equações (9) e (10) como:

$$\frac{\partial \mathcal{E}}{\partial w_{i,j}^{m}} \equiv \frac{\partial \mathcal{E}}{\partial n_i^{m}} \cdot \frac{\partial n_i^{m}}{\partial w_{i,j}^{m}} \qquad \textbf{(12)}$$

$$\frac{\partial \mathcal{E}}{\partial b^{m}} \equiv \frac{\partial \mathcal{E}}{\partial n_i^{m}} \cdot \frac{\partial n_i^{m}}{\partial b^{m}} \qquad \textbf{(13)}$$

Aplicando o segundo termo das equações (12) e (13) sobre a equação (11) tem-se:

$$\frac{\partial n_i^{m}}{\partial w_{i,j}^{m}} = \frac{\partial(\sum_{j=1}^{S^{m-1}} w_{i,j}^{m} \cdot a_j^{m-1} + b^{m})}{\partial w_{i,j}^{m}} = a_j^{m-1} \qquad \textbf{(14)}$$

$$\frac{\partial n_i^{m}}{\partial b^{m}} = \frac{\partial(\sum_{j=1}^{S^{m-1}} w_{i,j}^{m} \cdot a_j^{m-1} + b^{m})}{\partial b^{m}} = 1 \qquad \textbf{(15)}$$

Para simplificar a notação considera-se,

$$s_i^{m} = \frac{\partial \mathcal{E}}{\partial n_i^{m}} \qquad \textbf{(16)}$$ [18]

A equação (16) retrata a *sensibilidade* de $\mathcal{E}$ para alterar o i-ésimo elemento (sinal) de entrada na camada *m*. O cálculo de s_i^{m} é importante, pois é com ele que faz-se a distribuição do *erro* pelas camadas escondidas na etapa de retropropagação.

Segundo Haykin (2001), apesar dos neurônios ocultos não serem acessíveis diretamente, eles compartilham a responsabilidade por qualquer erro cometido na saída da rede. A ideia de sensibilidade é saber como penalizar ou recompensar os neurônios ocultos pela sua parcela de responsabilidade no desempenho da rede, o que ocorre durante a etapa de retropropagação.

---

[18] s^{m} : representa a sensibilidade, e S^{m} é o índice que expressa o número de neurônios na camada *m*;

Considerando a equação (16), podem-se alterar as equações (12) e (13) da seguinte forma:

$$\frac{\partial \mathcal{E}}{\partial w_{i,j}^{m}} = s_i^m \cdot a_j^{m-1} \tag{17}$$

$$\frac{\partial \mathcal{E}}{\partial b^{m}} = s_i^m \cdot 1 = s_i^m \tag{18}$$

Por fim, pode-se expressar o algoritmo do gradiente descendente, mostrado nas equações (9) e (10), como:

$$w_{i,j}^{m}(k+1) = w_{i,j}^{m}(k) - \alpha s_i^m \cdot a_j^{m-1} \tag{19}$$

$$b^{m}(k+1) = b^{m}(k) - \alpha s_i^m \tag{20}$$

Seguindo a notação proposta por Hagan *et al.* (1996), expressa-se as equações (19) e (20) em sua forma matricial por:

$$W^{m}(k+1) = W^{m}(k) - \alpha s^{m}(a^{m-1})^{T} \tag{21}$$

$$b^{m}(k+1) = b^{m}(k) - \alpha s^{m} \tag{22}$$

Onde, por (16) temos:

$$s^{m} = \begin{bmatrix} \frac{\partial \mathcal{E}}{\partial n_1^m} \\ \frac{\partial \mathcal{E}}{\partial n_2^m} \\ \vdots \\ \frac{\partial \mathcal{E}}{\partial n_{S^m}^m} \end{bmatrix} \tag{23}$$

Considerando:

s^m: a sensibilidade da camada *m*;

$n_{S^m}^m$: representa o somatório do produto entre pesos e entradas, adicionado o bias, do neurônio *i* pertencente a camada *m*;

A forma matricial é importante pois facilita a manipulação algébrica quando pretende-se uma implementação computacional.

Para o cálculo das *sensitividades* é necessário aplicar, novamente, a *regra da cadeia*. Esse processo descreve a relação de recorrência na qual a *sensitividade*

para a camada (*m)* é calculada utilizando a *sensitividade* na camada (*m+1)*, lembrando que isso ocorre no processo de retropropagação, ou seja, do último para o primeiro layer. A FIGURA 28 ilustra o processo.

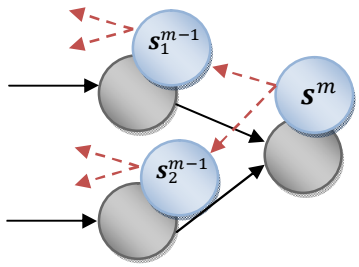


FIGURA 28: PROPAGAÇÃO DAS SENSIBILIDADES
FONTE: AUTOR.

A seguinte matriz jacobiana é usada para derivar a relação de recorrência das sensitividades.

$$\frac{\partial \boldsymbol{n}^{m+1}}{\partial \boldsymbol{n}^{m}} = \begin{pmatrix} \frac{\partial \boldsymbol{n}_1^{m+1}}{\partial \boldsymbol{n}_1^{m}} & \frac{\partial \boldsymbol{n}_1^{m+1}}{\partial \boldsymbol{n}_2^{m}} & \cdots & \frac{\partial \boldsymbol{n}_1^{m+1}}{\partial \boldsymbol{n}_{S^m}^{m}} \\ \frac{\partial \boldsymbol{n}_2^{m+1}}{\partial \boldsymbol{n}_1^{m}} & \frac{\partial \boldsymbol{n}_2^{m+1}}{\partial \boldsymbol{n}_2^{m}} & \cdots & \frac{\partial \boldsymbol{n}_2^{m+1}}{\partial \boldsymbol{n}_{S^m}^{m}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \boldsymbol{n}_{S^{m+1}}^{m+1}}{\partial \boldsymbol{n}_1^{m}} & \frac{\partial \boldsymbol{n}_{S^{m+1}}^{m+1}}{\partial \boldsymbol{n}_1^{m}} & \cdots & \frac{\partial \boldsymbol{n}_{S^{m+1}}^{m+1}}{\partial \boldsymbol{n}_1^{m}} \end{pmatrix} \tag{24}$$

Os elementos da matriz (24) podem ser calculados mediante a expressão (25).

$$\frac{\partial n_i^{m+1}}{\partial n_j^{m}} = \frac{\partial\left(\sum_{l=1}^{S^m} w_{i,l}^{m+1} \cdot a_l^m + b_i^{m+1}\right)}{\partial n_j^m} = w_{i,j}^{m+1} \frac{\partial a_j^m}{\partial n_j^m} =$$

$$= \boldsymbol{w}_{i,j}^{m+1} \frac{\partial f^m(\boldsymbol{n}_j^m)}{\partial \boldsymbol{n}_j^m} = \boldsymbol{w}_{i,j}^{m+1} \dot{\boldsymbol{f}}^m(\boldsymbol{n}_j^m) \tag{25}$$

Considerando,

$$\dot{\boldsymbol{f}}^m(\boldsymbol{n}_j^m) = \frac{\partial f^m(\boldsymbol{n}_j^m)}{\partial \boldsymbol{n}_j^m} \tag{26}$$

Assim, pode-se tomar a forma matricial de (26) como:

$$\frac{\partial \boldsymbol{n}^{m+1}}{\partial \boldsymbol{n}^{m}} = \boldsymbol{W}^{m+1}\dot{\boldsymbol{F}}^{m}(\boldsymbol{n}^{m}) \tag{27}$$

Onde o termo $\dot{\boldsymbol{F}}^{m}(\boldsymbol{n}^{m})$ pode ser calculado como:

$$\dot{\boldsymbol{F}}^{m}(\boldsymbol{n}^{m}) = \begin{bmatrix} \dot{f}^{m}(n_1^m) & 0 & \cdots & 0 \\ 0 & \dot{f}^{m}(n_2^m) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \dot{f}^{m}(n_{S^m}^m) \end{bmatrix} \tag{28}$$

Pode-se representar a relação de recorrência para a sensitividade usando a regra da cadeia na forma matricial, como mostra a equação (29).

$$s^m = \frac{\partial n^{m+1}}{\partial n^m} = \left(\frac{\partial n^{m+1}}{\partial n^m}\right)^T \cdot \frac{\partial \mathcal{E}}{\partial n^{m+1}} = \dot{\boldsymbol{F}}^{m}(\boldsymbol{n}^{m})(\boldsymbol{W}^{m+1})^T \frac{\partial \mathcal{E}}{\partial n^{m+1}} \Rightarrow$$

$$\Rightarrow \boldsymbol{s}_i^m = \dot{\boldsymbol{F}}^{m}(\boldsymbol{n}^{m})(\boldsymbol{W}^{m+1})^T \boldsymbol{s}^{m+1} \text{ , para m = } M\text{-}1,\dots,1 \tag{29}$$

Observa-se que a *sensitividade* é propagada para trás, da última para a primeira camada da rede, originando o nome "*backpropagation*". A expressão (30) ilustra essa idéia.

$$\boldsymbol{s}^M \rightarrow \boldsymbol{s}^{M-1} \rightarrow \cdots \rightarrow \boldsymbol{s}^2 \rightarrow \boldsymbol{s}^1 \tag{30}$$

Para finalizar o algoritmo do *backpropagation* é necessário explicitar o modo para se obter o $\boldsymbol{s}^m$ relativo a última camada, representado por $\boldsymbol{s}^M$, sendo necessário calcular os elementos do vetor mostrado em (23).

$$\boldsymbol{s}_i^M = \frac{\partial \mathcal{E}}{\partial \boldsymbol{n}_i^M} = \frac{\partial (\boldsymbol{t}-\boldsymbol{a})^T(\boldsymbol{t}-\boldsymbol{a})}{\partial \boldsymbol{n}_i^M} = \frac{\partial \sum_{j=1}^{S^M}(\boldsymbol{t}_j-\boldsymbol{a}_j)^2}{\partial \boldsymbol{n}_i^M} = -2(\boldsymbol{t}_j - \boldsymbol{a}_j)\frac{\partial \boldsymbol{a}_i}{\partial \boldsymbol{n}_i^M} \tag{31}$$

Onde,

$$\frac{\partial \boldsymbol{a}_i}{\partial \boldsymbol{n}_i^M} = \frac{\partial f^M(\boldsymbol{n}_i^M)}{\partial \boldsymbol{n}_i^M} = \dot{f}^M(\boldsymbol{n}_i^M). \tag{32}$$

Assim, pode-se expressar (31) como:

$$\boldsymbol{s}_i^M = -2(\boldsymbol{t}_i - \boldsymbol{a}_i)\dot{f}^M(\boldsymbol{n}_i^M) \tag{33}$$

Na forma matricial,

$$\boldsymbol{s^M = -2\dot{F}^M(n^M)(t-a)} \tag{34}$$

O método do gradiente descendente, embora robusto e confiável, converge lentamente para a solução ótima e possui uma tendência a ficar "perdido" nas regiões que apresentam mínimos locais, FIGURA 29, sendo pouco eficiente quando comparado a outros métodos como os de Gauss-Newton e Levemberg-Marquardt, como representado pela FIGURA 30. Embora o método do gradiente descendente possa vir a ser o único caminho para problemas que apresentem milhares de parâmetros, visto a sua simplicidade computacional (HAYKIN, 2001; GAVIN, 2011).

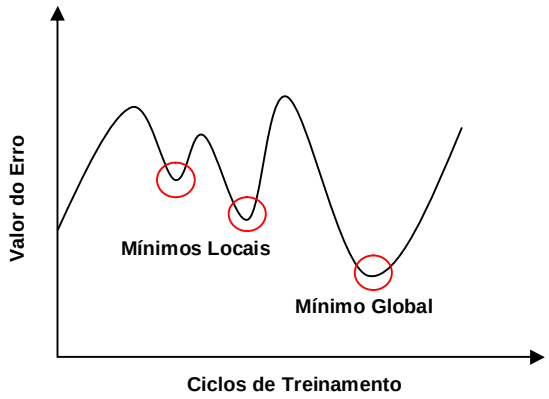


FIGURA 29: GRÁFICO DE UMA POSSÍVEL SUPERFÍCIE DE ERRO MOSTRANDO OS MÍNIMOS LOCAIS E O MÍNIMO GLOBAL.
FONTE: AUTOR.

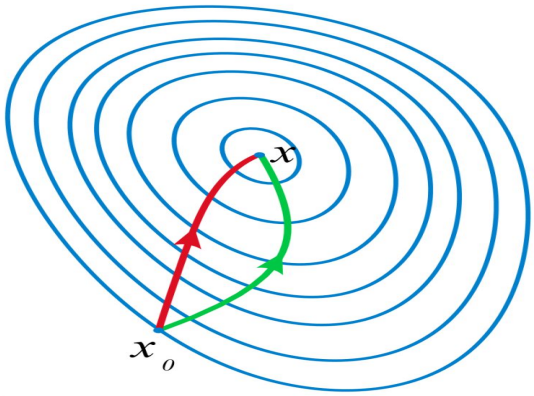


FIGURA 30: APROXIMAÇÃO PELO MÉTODO DE NEWTON (VERMELHO) E APROXIMAÇÃO PELO MÉTODO DA DESCIDA MAIS ÍNGREME (VERDE).
FONTE: ALEXANDROV, 2007

O algoritmo do Método do Gradiente Descendente pode ser resumido como:

---

1. *Inicializar os pesos das conexões e dos bias com valores aleatórios.*
2. *Apresentar todas as entradas para a rede e calcular as saídas, eq.(6), e erros correspondentes, eq. (7) e (8).*
3. *Calcular as sensibilidades com a relação de recorrência, eq. (24) e (29).*
4. *Atualizar pesos e bias utilizando o método do Gradiente Descendente, eq. (21) e (22).*
5. *Verificar os erros para os padrões de entrada e testar o critério de parada.*

No APÊNDICE 1 há um exemplo do método do *Gradiente Descendente*.

4.3.2 Modificações heurísticas do método *backpropagation*

• MOMENTO

Segundo Hagan *et al.* (1996), um procedimento para melhorar o algoritmo do backpropagation padrão, que utiliza o método do gradiente descendente, é utilizar um termo de *momento* com o objetivo de suavizar a trajetória de oscilação sobre a superfície de erro e, com isso, otimizar a convergência.

Para ilustrar o efeito de suavização, toma-se um exemplo simples constituído por um filtro de Passa Baixa[19] de 1ª ordem, conforme equação (35).

$$\boldsymbol{y(k) = \gamma y(k-1) + (1-\gamma)w(k)} \qquad \textbf{(35)}$$

Onde:

w(k): sinal de entrada;

y(k): sinal de saída;

γ: é coeficiente de momento, obedecendo ($0 \leq \gamma < 1$);

Neste exemplo, o sinal de entrada foi obtido por meio de uma equação de onda senoidal, (36).

$$\boldsymbol{w(k) = 1 + sen(\frac{2\pi k}{16})} \qquad \textbf{(36)}$$

As Figuras 31 e 32 ilustram a influência do termo de momento (γ) no comportamento do sinal de saída (realce em vermelho) em relação ao sinal de entrada (realce em preto). Pode-se perceber que a oscilação da saída do filtro é menor que a oscilação do sinal de entrada, conforme se espera de um filtro de Passa Baixa. Quando o termo de momento (γ) aumenta, a oscilação na saída do filtro é suavizada, ou seja, tem uma redução de amplitude conforme ilustra a FIGURA 31. Outro aspecto significativo é que o valor médio entre a entrada do filtro, *w(k)*, e a saída, *y(k)*, são muito próximos, respectivamente a 1,02 e 0,88, ou seja, o filtro tem a tendência de reduzir a quantidade de oscilações sem grandes discrepâncias do valor médio. Desconsiderando os possíveis erros de aproximação, os valores médios tendem a serem iguais. (BISHOP, 2006; HAGAN *et al.*,1996)

---

[19] Os filtros Passa Baixas eliminam altas frequências, sendo usados para eliminar ruídos.

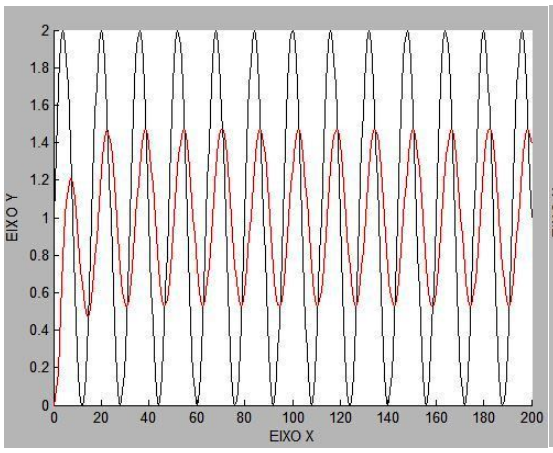


FIGURA 32: EFEITO DE SUAVIZAÇÃO DO MOMENTO PARA $\boldsymbol{\gamma}$ = 0,90
FONTE: AUTOR (ADAPTADO HAGAN *et al.*(1996))

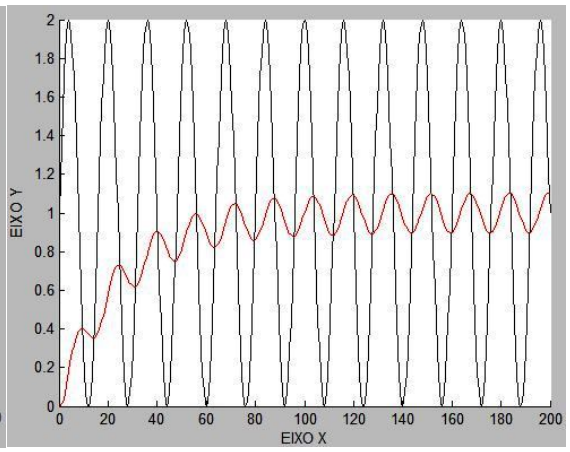


FIGURA 31: EFEITO DE SUAVIZAÇÃO DO MOMENTO PARA $\boldsymbol{\gamma}$ = 0,98
FONTE: AUTOR (ADAPTADO HAGAN *et al.*(1996))

Em caráter ilustrativo, no APÊNDICE 2 contém o código-fonte utilizado para criação das Figuras 31 e 32.

Adaptando-se essa ideia às redes neurais faz-se necessário que seja feita uma modificação nas equações responsáveis pelos ajustes dos pesos e bias, equações (19) e (20).

Adicionando o termo de *momento* obtém-se as seguintes equações modificadoras para a etapa de atualização dos pesos do método *backpropagation*.

$$\Delta \boldsymbol{w^m(k)} = \boldsymbol{\gamma \Delta w^m(k-1) - (1-\gamma)\alpha s^m (a^{m-1})^T} \quad \textbf{(37)}$$

$$\Delta \boldsymbol{b^m(k)} = \boldsymbol{\gamma \Delta b^m(k-1) - (1-\gamma)\alpha s^m} \quad \textbf{(38)}$$

De acordo com Hagan *et al.* (1996), a utilização do termo de *momento* permite a utilização de taxas de aprendizagem maiores sem perda da estabilidade da rede. Também tende a acelerar a convergência mantendo uma direção consistente.

• VARIAÇÃO DA TAXA DE APRENDIZAGEM

A FIGURA 33 representa um superfície de erro, onde o ponto em azul indica o ponto de partida, escolhido aleatoriamente, e o ponto em vermelho denota o ótimo global. Essa superfície tem regiões inclinadas e planas, o caminho em verde representa um possível percurso em busca do ótimo, algo que é obtido a cada iteração da rede, pontos em preto, pela atualização dos pesos e bias.

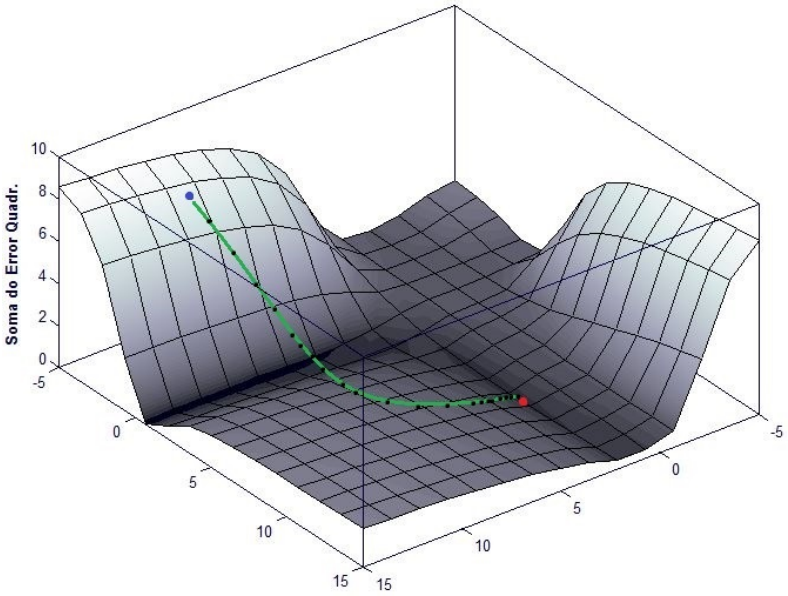


FIGURA 33: EXEMPLO DE UMA SUPERFÍCIE DE DESEMPENHO
FONTE: AUTOR (ADAPTADO DE HAGAN *et al.* (1996)

Segundo Hagan *et al.* (1996), como a forma da superfície pode ser diferente em regiões diversas, pode-se otimizar a convergência, durante a etapa de treinamento se a taxa de aprendizagem for aumentada nas regiões planas e diminuída nas inclinadas.

Ainda de acordo com Hagan *et al.* (1996), uma tática é variar a taxa de aprendizagem de acordo com o desempenho do algoritmo. As regras propostas são as seguintes:

1) Se o erro quadrático, ao longo de todo o conjunto de treinamento, aumentar mais que um percentual **θ** após a atualização dos pesos, então os novos pesos são descartados e a taxa de aprendizagem é multiplicada por algum fator $0< \rho <1$ e o *coeficiente de momento*, γ, assume o valor zero. (em geral 1%< **θ** <5%).

2) Se o erro quadrático diminuir depois da atualização dos pesos, então a taxa de aprendizagem é multiplicada por uma fator η > 1. Caso o *coeficiente de momento* seja zero, então ele receberá seu valor original. Do contrário, não se alterará o valor do *momento*.

3) Se o erro quadrático aumentar, mas em um percentual inferior a **θ**, então a atualização dos pesos é aceita mas a taxa de aprendizagem não será alterada. Caso o *momento* seja zero, então ele receberá seu valor original. Do contrário, não se alterará o valor do *momento*.

4.3.3. Método de Gauss-Newton e Levemberg Marquardt

O método de Levenberg-Marquardt é uma variação dos métodos de Guass-Newton e Gradiente Descendente, cujo objetivo é minimizar funções não-lineares. Sua aplicação como uma heurística de Redes Neurais Artificiais foi apresentada, inicialmente, no artigo de Hagan e Menhaj (1994).

Segundo Yu e Wilamowski (2010), o método de Newton trabalha com a matriz hessiana e as derivadas de segunda ordem da função erro, $\mathcal{E}(\boldsymbol{w})$, resultando em um custo computacional elevado. Com o intuito de simplificar o processo de cálculo, introduz-se a matriz Jacobiana, sendo esta computacionalmente mais simples de utilizar.

Define-se:

$$\bar{e}^{\,T} = [\bar{e}_1, \bar{e}_2, \dots, \bar{e}_N] = [\overbrace{e_{1,1}, e_{2,1}, \dots, e_{S^M,1}}^{\bar{e}_1}, e_{1,2}, \dots, e_{S^M,Q}]$$

$e_{S^M,Q}$: S^M indica a posição do elemento no vetor erro ($\bar{\boldsymbol{e}}$) e $\boldsymbol{Q}$ indica o padrão de treinamento (amostra) relacionado *(rede com saídas com múltiplos elementos)*.

$$w^{\,T} = [w_1, w_2, \dots, w_n] = [\overbrace{w^1_{1,1}, w^1_{1,2}, \dots, w^1_{S^1,R}, b^1_1, \dots, b^1_{S^1}}^{w_1}, w^2_{1,1}, \dots, b^M_{S^M}]$$

$w^m_{S^m,R}$: indica o peso (w) da camada/layer (m) do neurônio na posição (S^m) que recebe sinal do elemento (R) da camada anterior (m-1). Em resumo, $\boldsymbol{w}$ representa todos os pesos e bias, da rede, em um vetor.

De acordo com Hagan *et al.* (1996) o método de Newton para otimização de um índice de desempenho $\mathcal{E}(\boldsymbol{w})$ é descrito pela equação (39).

$$\boldsymbol{w_{k+1}} = \boldsymbol{w_k} - \boldsymbol{A_k^{-1} g_k} \tag{39}$$

Onde, A_k representa a matriz hessiana, (40), e g_k o gradiente, (41).

$$\boldsymbol{A_k} = \boldsymbol{\nabla^2}\mathcal{E}(\boldsymbol{w}) \tag{40}$$

$$\boldsymbol{g_k} = \boldsymbol{\nabla}\,\mathcal{E}(\boldsymbol{w}) \tag{41}$$

De modo análogo à equação (8), pode-se representar $\mathcal{E}(\boldsymbol{w})$ como a soma de funções quadráticas da forma,

$$\mathcal{E}(\boldsymbol{w}) = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \bar{\boldsymbol{e}}_{\boldsymbol{i}}^{\boldsymbol{2}}(\boldsymbol{w}) = \bar{\boldsymbol{e}}^{\boldsymbol{T}}(\boldsymbol{w})\bar{\boldsymbol{e}}(\boldsymbol{w}) \tag{42}$$

O gradiente de $\mathcal{E}(\boldsymbol{w})$ pode ser expresso na forma matricial como:

$$\boldsymbol{\nabla}\mathcal{E}(\boldsymbol{w}) = \boldsymbol{2J^T}(\boldsymbol{w})\bar{\boldsymbol{e}}(\boldsymbol{w}) \tag{43}$$

Sendo que a matriz jacobiana apresentada é expressa por (44).

$$\boldsymbol{J}(\boldsymbol{w}) = \begin{pmatrix} \frac{\partial \bar{e}_1(w)}{\partial w_1} & \frac{\partial \bar{e}_1(w)}{\partial w_2} & \cdots & \frac{\partial \bar{e}_1(w)}{\partial w_n} \\ \frac{\partial \bar{e}_2(w)}{\partial w_1} & \frac{\partial \bar{e}_2(w)}{\partial w_2} & \cdots & \frac{\partial \bar{e}_2(w)}{\partial w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial \bar{e}_N(w)}{\partial w_1} & \frac{\partial \bar{e}_N(w)}{\partial w_2} & \cdots & \frac{\partial \bar{e}_N(w)}{\partial w_n} \end{pmatrix} \tag{44}$$

Para obter a matriz hessiana deve-se considerar a equação (45), onde os índices k,j representam os elementos da matriz.

$$[\boldsymbol{\nabla^2}\mathcal{E}(\boldsymbol{w})]_{\mathbf{k,j}} = \frac{\partial^2 \mathcal{E}(w)}{\partial w_{\mathbf{k}} \partial w_{\mathbf{j}}} = \mathbf{2}\sum_{\mathbf{i=1}}^{\mathbf{N}} \left\{ \frac{\partial \bar{e}_{\mathbf{i}}(w)}{\partial w_{\mathbf{k}}} \frac{\partial \bar{e}_{\mathbf{i}}(w)}{\partial w_{\mathbf{j}}} + \bar{e}_{\mathbf{i}}(w) \frac{\partial^2 \bar{e}_{\mathbf{i}}(w}{\partial w_{\mathbf{k}} \partial w_{\mathbf{j}}} \right\} \tag{45}$$ [20]

A equação (45) pode ser representada de forma matricial pela equação (46).

$$\boldsymbol{\nabla^2}\mathcal{E}(\boldsymbol{w}) = \boldsymbol{2J^T}(\boldsymbol{w})\boldsymbol{J}(\boldsymbol{w}) + \boldsymbol{2S}(\boldsymbol{w}) \tag{46}$$

Segundo BISHOP(2006) se a rede, depois de treinada por um conjunto de dados, tem sua saída ($\mathbf{a_k}$) próxima do seu valor desejado ($\mathbf{t_k}$), então o termo ***S(w)***, que representa a matriz de sensibilidades, pode ser desprezado, resultando no

[20] A equação é obtida mediante a expansão da Série de Taylor aplicada a matrizes. Vide HAGAN *et al.* (1996), capítulo 8.

Método de Gauss-Newton (Quasi-Newton). Caso queira-se considerar o termo ***S(w)***, o desenvolvimento dar-se-á pelo Método de Newton, (47).

$$\boldsymbol{w_{k+1} = w_k - A^{-1}(w_k)g(w_k)} \tag{47}$$

Onde:

A: representa a matriz hessiana, (40);

g: representa o gradiente, (41);

Assim, segundo Hagan *et al.* (1996) e Haykin (2001), pode-se aproximar a matriz hessiana como,

$$\boldsymbol{\nabla^2 \mathcal{E}(w) \cong 2J^T(w)J(w).} \tag{48}$$

Substituindo as equações (48) e (43) na equação (39) obtém-se, de fato, o *Método de Gauss-Newton*, (49).

$$\begin{aligned} \boldsymbol{w_{k+1}} &= \boldsymbol{w_k - [2J^T(w_k)J(w_k)]^{-1}2J^T(w_k)\bar{e}(w_k)} \\ &= \boldsymbol{w_k - [J^T(w_k)J(w_k)]^{-1}J^T(w_k)\bar{e}}(w_k) \end{aligned} \tag{49}$$

A vantagem do *Método de Gauss-Newton* em relação ao *Método de Newton* é que não há necessidade de calcular as derivadas de segunda ordem exigidas pela matriz hessiana (HAGAN, 1996). Embora a aproximação da matriz hessiana feita por ***H = J^T J*** seja eficiente, não se pode garantir que ***H*** sempre será invertível, o que é uma condição necessária para a funcionalidade dos *Métodos de Newton* e *Gauss-Newton*. Lembrando que se ***H*** for definida positiva, então será invertível.

Segundo Hagan *et. al* (1996) e Haykin (2001) o problema da invertibilidade da matriz ***H*** pode ser superado por meio do seguinte artifício:

$$\boldsymbol{G = H + \mu I}. \tag{50}$$

Para comprovar a funcionalidade de (50) pode-se supor que os *autovalores* e *autovetores* de **H** são, respectivamente, $\{\lambda_1, \lambda_2, \dots, \lambda_n\}$ e $\{z_1, z_2, \dots, z_n\}$. Com isso pretende-se mostrar que **G** e **H** possuem os mesmo *autovetores*, conforme (51).

$$\boldsymbol{Gz_i = [H + \mu I]z_i = Hz_i + \mu z_i = \lambda_i z_i + \mu z_i = (\lambda_i + \mu)z_i} \tag{51}$$

Por (51) temos que os *autovetores* de **H** e **G** são os mesmos. Os autovalores de **G** são dados por $(\lambda_i + \boldsymbol{\mu})$, assim pode-se construir **G** como uma matriz definida positiva aumentando o valor de $\boldsymbol{\mu}$ até que $(\lambda_i + \boldsymbol{\mu}) > 0, \forall\, i$. Deste modo, a matriz será invertível.

A junção das equações (49) e (50) levam à ideia central do *Método de Levemberg-Marquardt*, ferramenta principal dessa dissertação. Pode-se representar o método como:

$$\boldsymbol{w_{k+1}} = \boldsymbol{w_k} - [\boldsymbol{J^T}(\boldsymbol{w_k})\boldsymbol{J}(\boldsymbol{w_k}) + \boldsymbol{\mu_k I}]^{-1}\boldsymbol{J^T}(\boldsymbol{w_k})\bar{\boldsymbol{e}}(\boldsymbol{w_k}) \tag{52}$$

$$\Delta\boldsymbol{w_k} = -[\boldsymbol{J^T}(\boldsymbol{w_k})\boldsymbol{J}(\boldsymbol{w_k}) + \boldsymbol{\mu_k I}]^{-1}\boldsymbol{J^T}(\boldsymbol{w_k})\,\bar{\boldsymbol{e}}(\boldsymbol{w_k}) \tag{53}$$

O algoritmo de *Levemberg-Marquardt* tem uma característica especial pois quanto maior torna-se $\boldsymbol{\mu_k}$, mais próximo do método do *Gradiente Descendente* ficam as equações (52) e (53), como mostra (54).

$$\boldsymbol{w_{k+1}} = \boldsymbol{w_k} - \frac{\mathbf{1}}{\boldsymbol{\mu_k}}\boldsymbol{J^T}(\boldsymbol{w_k})\bar{\boldsymbol{e}}(\boldsymbol{w_k}) = \boldsymbol{w_k} - \frac{\mathbf{1}}{\mathbf{2}\boldsymbol{\mu_k}}\boldsymbol{\nabla}\mathcal{E}(\boldsymbol{w}) \tag{54}$$

Com $\left(\frac{\mathbf{1}}{\mathbf{2}\boldsymbol{\mu_k}}\right)$ atuando com a taxa de aprendizagem.

De modo similar, quando $\boldsymbol{\mu_k}$ está mais próximo de zero as equações (52) e (53) aproximam-se do método de *Gauss-Newton*.

O algoritmo começa com um valor pequeno para $\boldsymbol{\mu_k}$, em geral $\boldsymbol{\mu_k} = 0{,}01$, se ao final da iteração a função erro $\mathcal{E}(\boldsymbol{w})$ não reduzir então $\boldsymbol{\mu_k}$ será multiplicado por um fator $\boldsymbol{\vartheta} > 1$ e o passo será repetido. A função $\mathcal{E}(\boldsymbol{w})$ deverá, eventualmente, reduzir, com passos apontando na direção inversa a do gradiente, como indica o método do *Gradiente Descendente*. Caso a iteração reduza o valor de $\mathcal{E}(\boldsymbol{w})$ então $\boldsymbol{\mu_k}$ será dividido pelo fator $\boldsymbol{\vartheta}$, aproximando o modelo ao método de *Gauss-Newton*, que tem maior velocidade de convergência. O método de *Levemberg-Marquardt* proporciona uma boa relação entre a velocidade do método de *Gauss-Newton* e a garantia de convergência do método do *Gradiente Descendente*.

A Tabela 2 mostra um comparativo entre os métodos apresentados.

TABELA 2: COMPARATIVO ENTRE OS ALGORITMOS DE APRENDIZAGEM

Algoritmo de aprendizagem	Regra de atualização	Taxa de convergência	Complexidade Computacional
Gradiente Descendente	$\boldsymbol{w(n+1) = w(n) - \eta g(n)}$	Estável e lento	Gradiente
Newton	$\boldsymbol{w(n+1) = w(n) - H^{-1}(n)g(n)}$	Instável e rápido	Gradiente e Hessiana
Gauss-Newton	$\boldsymbol{w(n+1) = w(n) - \left(J^T(n)J(n)\right)^{-1}J^T(n)e(n)}$	Instável e rápido	Jacobiano
Levemberg-Marquardt	$\boldsymbol{w(n+1) = w(n) - (J^T(n)J(n) + \delta I)^{-1}J^T(n)e(n)}$	Estável e rápido	Jacobiano

FONTE: (YU, 2011)

O desenvolvimento das equações (52) e (53) consistem no cálculo da matriz jacobiana. Modificando a matriz em (44), tem-se:

$$\boldsymbol{J(w)} = \begin{bmatrix} \frac{\partial e_{1,1}}{\partial w^1_{1,1}} & \frac{\partial e_{1,1}}{\partial w^1_{1,2}} & \cdots & \frac{\partial e_{1,1}}{\partial w^1_{S^1,R}} & \frac{\partial e_{1,1}}{\partial b^1_1} & \cdots \\ \frac{\partial e_{2,1}}{\partial w^1_{1,1}} & \frac{\partial e_{2,1}}{\partial w^1_{1,2}} & \cdots & \frac{\partial e_{2,1}}{\partial w^1_{S^1,R}} & \frac{\partial e_{2,1}}{\partial b^1_1} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \frac{\partial e_{S^M,1}}{\partial w^1_{1,1}} & \frac{\partial e_{S^M,1}}{\partial w^1_{1,2}} & \cdots & \frac{\partial e_{S^M,1}}{\partial w^1_{S^1,R}} & \frac{\partial e_{S^M,1}}{\partial b^1_1} & \cdots \\ \frac{\partial e_{1,2}}{\partial w^1_{1,1}} & \frac{\partial e_{S^M,1}}{\partial w^1_{1,2}} & \cdots & \frac{\partial e_{S^M,1}}{\partial w^1_{S^1,R}} & \frac{\partial e_{S^M,1}}{\partial b^1_1} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \end{bmatrix} \quad \textbf{(55)}$$

A taxa de erro ($e_{S^m,Q}$) é calculada em relação a cada conexão ($w^m_{S^m,R}$). Lembrando-se que:

$$\bar{e}^{\,T} = [\bar{e}_1, \bar{e}_2, \dots, \bar{e}_N] = \left[e_{1,1}, e_{2,1}, \dots, e_{S^M,1}, e_{1,2}, \dots, e_{S^M,Q}\right]$$

$e_{S^m,Q}$: S^m indica a posição do elemento no vetor erro ($\bar{\boldsymbol{e}}$) e $\boldsymbol{Q}$ indica o padrão de treinamento (amostra) relacionado *(rede com mais de uma saída)*.

$$w^{\,T} = \left[w_{1,} w_{2,} \dots, w_n\right] = \left[w^1_{1,1}, w^1_{1,2}, \dots, w^1_{S^1,R}, b^1_1, \dots, b^1_{S^1}, w^2_{1,1}, \dots, b^M_{S^M}\right]$$

$w^{m}_{S^m,R}$: indica o peso (*w*) da *camada/layer* (*m*) na posição (S^m) que recebe sinal do elemento (*R*) da camada anterior. Portanto, ***w*** representa todos os pesos/bias da rede em um vetor (*M*, maiúsculo, é o índice da última camada).

Os termos da matriz (55) são representados como:

$$[\boldsymbol{J}]_{\boldsymbol{h},\boldsymbol{l}} = \frac{\partial \bar{\boldsymbol{e}}_{\boldsymbol{h}}}{\partial \boldsymbol{w}_{\boldsymbol{l}}} = \frac{\partial \boldsymbol{e}_{\boldsymbol{k},\boldsymbol{q}}}{\partial \boldsymbol{w}_{\boldsymbol{l}}} \tag{56}$$

Utilizando ideia similar à utilizada na derivação do método backpropagation padrão (utilizando o método do Gradiente Descendente), pode-se calcular a equação (56). Na equação (12) tinha-se,

$$\frac{\partial \mathcal{E}}{\partial w^{m}_{i,j}} \equiv \frac{\partial \mathcal{E}}{\partial n^{m}_{i}} \cdot \frac{\partial n^{m}_{i}}{\partial w^{m}_{i,j}}$$

O termo $\frac{\partial \mathcal{E}}{\partial n^{m}_{i}}$ foi definido como sendo a *sensibilidade*, ou seja,

$$s^{m}_{i} = \frac{\partial \mathcal{E}}{\partial n^{m}_{i}}$$

Para calcular a matriz jacobiana, (55), considera-se uma ideia similar mas com a sensibilidade descrita como:

$$\tilde{\boldsymbol{s}}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{h}} \equiv \frac{\partial \bar{\boldsymbol{e}}_{\boldsymbol{h}}}{\partial \boldsymbol{n}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{q}}} = \frac{\partial \boldsymbol{e}_{\boldsymbol{k},\boldsymbol{q}}}{\partial \boldsymbol{n}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{q}}} \tag{57}$$

Assim, os elementos da matriz jacobiana são calculados do seguinte modo:

$$[\boldsymbol{J}]_{\boldsymbol{h},\boldsymbol{i}} = \frac{\partial \bar{\boldsymbol{e}}_{\boldsymbol{h}}}{\partial \boldsymbol{w}_{\boldsymbol{i}}} = \frac{\partial \boldsymbol{e}_{\boldsymbol{k},\boldsymbol{q}}}{\partial \boldsymbol{w}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{j}}} = \frac{\partial \boldsymbol{e}_{\boldsymbol{k},\boldsymbol{q}}}{\partial \boldsymbol{n}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{q}}} \times \frac{\partial \boldsymbol{n}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{q}}}{\partial \boldsymbol{w}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{j}}} = \tilde{\boldsymbol{s}}_{\boldsymbol{i},\boldsymbol{h}} \times \frac{\partial \boldsymbol{n}^{\boldsymbol{m}}_{\boldsymbol{l},\boldsymbol{q}}}{\partial \boldsymbol{w}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{j}}} = \tilde{\boldsymbol{s}}^{\boldsymbol{m}}_{\boldsymbol{l},\boldsymbol{h}} \times \boldsymbol{a}^{\boldsymbol{m}-\boldsymbol{1}}_{\boldsymbol{j},\boldsymbol{q}} \tag{58}$$

Quando o peso $\boldsymbol{w_i}$ for referente ao bias, o elemento na matriz jacobiana é calculado como:

$$[\boldsymbol{J}]_{\boldsymbol{h},\boldsymbol{i}} = \frac{\partial \bar{\boldsymbol{e}}_{\boldsymbol{h}}}{\partial \boldsymbol{w}_{\boldsymbol{i}}} = \frac{\partial \boldsymbol{e}_{\boldsymbol{k},\boldsymbol{q}}}{\partial \boldsymbol{b}^{\boldsymbol{m}}_{\boldsymbol{i}}} = \frac{\partial \boldsymbol{e}_{\boldsymbol{k},\boldsymbol{q}}}{\partial \boldsymbol{n}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{q}}} \times \frac{\partial \boldsymbol{n}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{q}}}{\partial \boldsymbol{b}^{\boldsymbol{m}}_{\boldsymbol{i}}} = \tilde{\boldsymbol{s}}^{\boldsymbol{m}}_{\boldsymbol{i},\boldsymbol{h}} \tag{59}$$

As equações (58) e (59) representam as chamadas "*Sensitividades de Marquardt*". Essas sensitividades são calculadas por uma relação de recorrência

similar à utilizada no método backpropagation padrão mas como uma pequena modificação na sensitividade da camada final, $\tilde{S}_{i,h}^{M}$.

Para calcular a sensitividade na camada de saída utiliza-se a equação (60).

$$\tilde{\boldsymbol{S}}_{\boldsymbol{i,h}}^{\boldsymbol{M}} = \frac{\partial \bar{\boldsymbol{e}}_{\boldsymbol{h}}}{\partial \boldsymbol{n}_{\boldsymbol{i,q}}^{\boldsymbol{M}}} = \frac{\partial \boldsymbol{e}_{\boldsymbol{k,q}}}{\partial \boldsymbol{n}_{\boldsymbol{i,q}}^{\boldsymbol{M}}} = \frac{\partial (\boldsymbol{t}_{\boldsymbol{k,q}} - \boldsymbol{a}_{\boldsymbol{k,q}}^{\boldsymbol{M}})}{\partial \boldsymbol{n}_{\boldsymbol{i,q}}^{\boldsymbol{M}}} = -\frac{\partial \boldsymbol{a}_{\boldsymbol{k,q}}^{\boldsymbol{M}}}{\partial \boldsymbol{n}_{\boldsymbol{i,q}}^{\boldsymbol{M}}} = -\dot{\boldsymbol{f}}^{\boldsymbol{M}}\left(\boldsymbol{n}_{\boldsymbol{i,q}}^{\boldsymbol{M}}\right) \quad \textbf{(60)}$$

Na forma matricial temos a equação (60) como:

$$\tilde{\boldsymbol{S}}_{\boldsymbol{q}}^{\boldsymbol{M}} = -\dot{\boldsymbol{F}}^{\boldsymbol{M}}(\boldsymbol{n}_{\boldsymbol{q}}^{\boldsymbol{M}}) \quad \textbf{(61)}$$

Como o processo de retropropagação inicia-se no último layer, calculado $\tilde{s}_{i,h}^{M}$ pode-se obter a sensibilidade das próximas camadas utilizando a equação (62).

$$\tilde{\boldsymbol{S}}_{\boldsymbol{q}}^{\boldsymbol{m}} = \dot{\boldsymbol{F}}^{\boldsymbol{m}}\left(\boldsymbol{n}_{\boldsymbol{q}}^{\boldsymbol{m}}\right)(\boldsymbol{W}^{\boldsymbol{m+1}})^{\boldsymbol{T}}\tilde{\boldsymbol{S}}_{\boldsymbol{q}}^{\boldsymbol{m+1}} \quad \textbf{(62)}$$

A *sensitividade de Marquardt* total para a relação padrão/camada pode ser representada por uma matriz aumentada, conforme (63).

$$\tilde{\boldsymbol{S}}^{\boldsymbol{m}} = \left[\tilde{\boldsymbol{S}}_{\boldsymbol{1}}^{\boldsymbol{M}} \middle| \tilde{\boldsymbol{S}}_{\boldsymbol{2}}^{\boldsymbol{m}} \middle| \cdots \middle| \tilde{\boldsymbol{S}}_{\boldsymbol{Q}}^{\boldsymbol{m}}\right] \quad \textbf{(63)}$$

Onde:

$\tilde{\boldsymbol{S}}_{Q}^{m}$: vetor que representa a sensibilidade de Marquardt para a camada ***m*** em relação ao padrão ***Q***.

O algoritmo do método de *Levemberg-Marquardt* pode ser resumidos como:

---

1. *Inicializar os pesos das conexões e dos bias com valores aleatórios.*
2. *Apresentar todas as entradas para a rede e calcular as saídas, eq.(*6*), e erros correspondentes, eq. (*7*) e (*8*).*
3. *Calcular as sensibilidades com a relação de recorrência, eq. (*61*) e (*62*). Calcular a matriz jacobiana, (*55*), (*58*) e (*59*).*
4. *Encontrar* Δw_k *em (*53*).*
5. *Recalcular os erros para os pesos atualizados,* $w_k + \Delta w_k$*. Se a nova soma dos erros quadráticos for menor que o obtido no passo 1, então divida μ por* ϑ *e volte ao passo 1. Se a soma dos erros quadráticos não reduziu, então multiplique μ por* ϑ *e volte ao passo 3.*
6. *Teste o critério de parada.*

O APÊNDICE 3 apresenta um exemplo simples do método de *Levemberg-Marquardt*.

CAPÍTULO V - IMPLEMENTAÇÃO E ANÁLISE DOS RESULTADOS

5.1 MATLAB – NEURAL NETWORK TOOLBOX

Para o desenvolvimento desse trabalho foi utilizada uma RNA do tipo múltiplas camadas com retropropagação (*feedforward backpropagation network*), treinada e simulada utilizando o software MATLAB versão 7.10 (R2010a). Todo o processo de treinamento e teste foi desenvolvido e realizado em um computador dotado de processador marca INTEL modelo i7 3770k, com 16GB de memória RAM.

O MATLAB é um software robusto, que possui diversas ferramentas pré-programadas, conhecidas como *toolbox*, abrangendo vários ramos das ciências exatas e tecnológicas. Nesse trabalho foi utilizada a ferramenta *Neural Network Toolbox 6.0,* desenvolvida pela equipe de pesquisadores doutores *Howard Demuth*, *Mark Beale* e *Martin Hagan*.

Existem dois modos distintos para a utilização do *Neural Network Toolbox 6.0* no MATLAB. Uma possibilidade é usar a interface gráfica para usuário *(GUI - Graphical User Interface), vide* FIGURA 34, consistindo das ferramentas "*Neural Fitting Tool (GUI)*", "*Neural Clustering Tool (GUI)*", "*Neural Pattern Recognition Tool (GUI)*" e o "*Neural Network Tool (GUI)*". A outra possibilidade é utilizar a estruturação da rede mediante *linhas de comando* (command line functions), forma mais indicada para configurações mais específicas da rede. Quando se pretende utilizar processamento em cluster, com filas de execução, o método de linhas de comando é o mais apropriado.

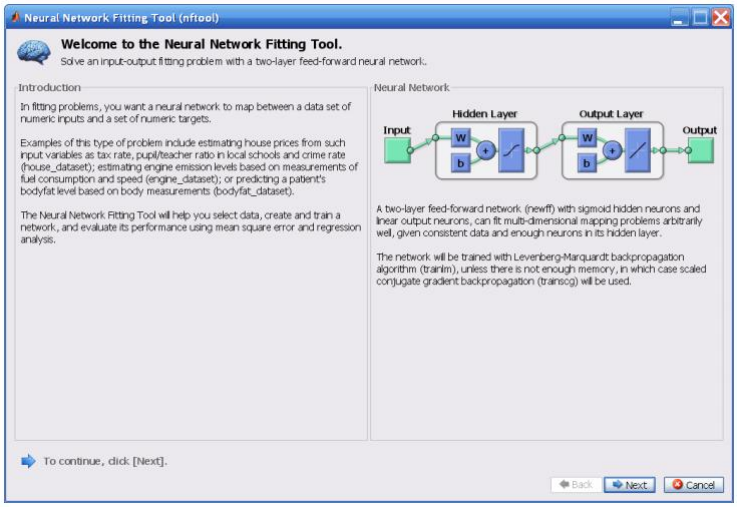


FIGURA 34: EXEMPLO DE UMA DAS INTERFACES GRÁFICAS, DISPONÍVEIS NO NEURAL NETWORK TOOLBOX 6.0, PARA MANIPULAÇÃO DE REDES NEURAIS.
FONTE: (DEMUTH *et al*., 2009)

Embora a interface com o usuário possibilite uma fácil manipulação do toolbox, não há a possibilidade de configurações mais profundas na estruturação da rede, limitando o usuário aos modelos definidos como *default* (padrão).

Visando uma melhor interação com a RNA a ser desenvolvida, optou-se pelas *linhas de comando* como forma de utilização do *toolbox* para redes neurais. A TABELA 3 mostra alguns dos comandos utilizados para criação, treino e simulação da RNA.

TABELA 3: COMANDOS COMUNS PARA CRIAÇÃO DE UMA RNA COM O *NEURAL NETWORK TOOLBOX*.

Comando	Default	Descrição
newff ()	####	Função para criar uma rede do tipo feed-forward backpropagation
net.trainParam.mu	0.001	Valor inicial para o mu
net.trainParam.mu_inc	10	Fator de incremento do mu
net.trainParam.mu_dec	0.1	Fator de decremento do mu
net.trainParam.mu_max	1e10	Valor máximo permitido para o mu
net.trainParam.max_fail	6	Máximo de falhas durante a validação
net.trainParam.epochs	1000	Máximo de iterações
net.trainParam.goal	0	Valor alvo para performance
net.divideFcn	dividerand (divisãoaleatória)	Define parâmetros para divisão dos dados
net.divideParam.trainRatio	0.6	Percentual de amostra destinado ao treinamento

net.divideParam.valRatio	0.2	Percentual de amostra destinado à validação
net.divideParam.testRatio	0.2	Percentual de amostra destinado ao teste
net.performFcn	mse (mean square error)	Define parâmetro de performance
train()	####	Função para treinamento da rede
sim()	####	Função para simulação da rede treinada

FONTE: (DEMUTH *et al.*, 2009)

No APÊNDICE 4 foi descrito o algoritmo implementado para treinamento e simulação da rede.

5.2 ARQUITETURA E PARÂMETROS DA REDE

A arquitetura utilizada consiste de uma Rede Neural Artificial com aprendizagem do tipo *backpropagation* utilizando a heurística de Levemberg-Marquadant para o treinamento e o método do Gradiente Descendente com Momento para a atualização dos pesos e bias. Esses métodos foram descritos com detalhes no quarto capítulo deste trabalho.

A camada de entrada é constituída por um vetor com dimensão P onde seus elementos representam o *grau de absorbância* para cada um dos 1556 *números de onda*[5] analisados. As entradas, valores de *números de onda*, pertencem ao intervalo [0,1]. A saída é ajustada, pela função de saída, para o intervalo discreto linear [1,2,3 ou 4]. Devido à forma de saída escolhida, optou-se por manter uma função de ativação linear na camada de saída da rede. Outra medida importante utilizada foi a aleatorização dos dados antes da sua apresentação à rede. De acordo com Haykin (2001), é recomendável tornar aleatória a ordem de apresentação dos exemplos de treinamento, pois isso favorece a busca estocástica dos pesos, evitando que a aprendizagem da rede fique "viciada". A FIGURA 35 ilustra a arquitetura da rede utilizada.

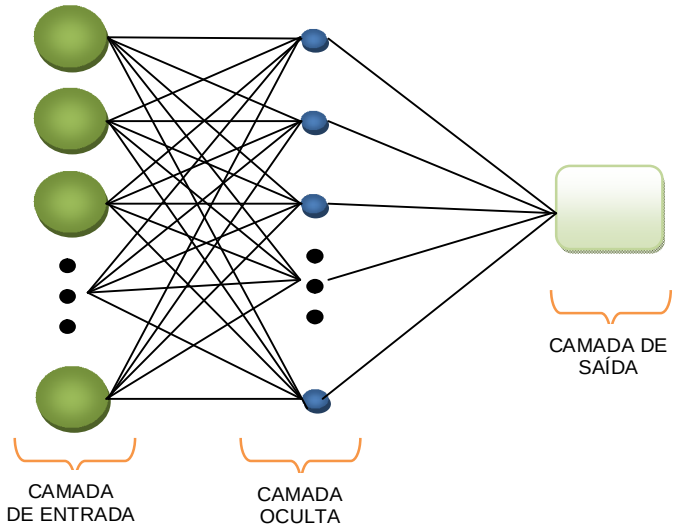


FIGURA 35: REPRESENTAÇÃO ESQUEMÁTICA DA ARQUITETURA DE REDE UTILIZADA.
FONTE: AUTOR

No modelo proposto, a rede possui apenas uma camada escondida. Segundo Haykin (2001), uma única camada oculta é suficiente para uma rede de múltiplas camadas computar uma aproximação uniforme para um dado conjunto de treinamento embora mais camadas possam, eventualmente, possibilitar melhores resultados. Como o objetivo deste trabalho é mostrar a viabilidade da aplicação de uma Rede Neural Artificial como classificador, a partir de um conjunto de dados obtidos por espectrofotômetro, optou-se, por questão de organização e tempo, trabalhar com uma única camada escondida e variar a quantidade de neurônios existentes nesta de 1 a 15. O tempo de treinamento aumenta de acordo com o número de neurônios na camada escondida, por isso foi escolhido um limite de 15 neurônios. A TABELA 4 e a FIGURA 36 apresentam o tempo de execução médio, obtido de dez repetições, por número de neurônios na camada escondida.

TABELA 4: TEMPO, TOTAL E MÉDIO, DE TREINAMENTO POR NÚMERO DE NEURÔNIOS NA CAMADA ESCONDIDA.

Função de Ativação log-sigmoidal	
Nº Neurônios	**Média**
1	00:00:17
2	00:01:09
3	00:02:12
4	00:04:37
5	00:12:48
6	00:20:13
7	00:38:09
8	00:39:45
9	01:09:30
10	01:22:02
11	03:29:29
12	03:39:00
13	03:57:59
14	04:17:24
15	05:38:04
Total	**25:32:38**

Função de Ativação tan-sigmoidal	
Nº Neurônios	**Média**
1	0:00:14
2	0:01:10
3	0:02:15
4	0:03:58
5	0:10:12
6	0:23:14
7	0:30:38
8	0:41:53
9	1:20:48
10	1:13:56
11	1:43:15
12	2:57:03
13	3:38:51
14	4:01:38
15	4:17:40
Total	**21:06:45**

FONTE: AUTOR

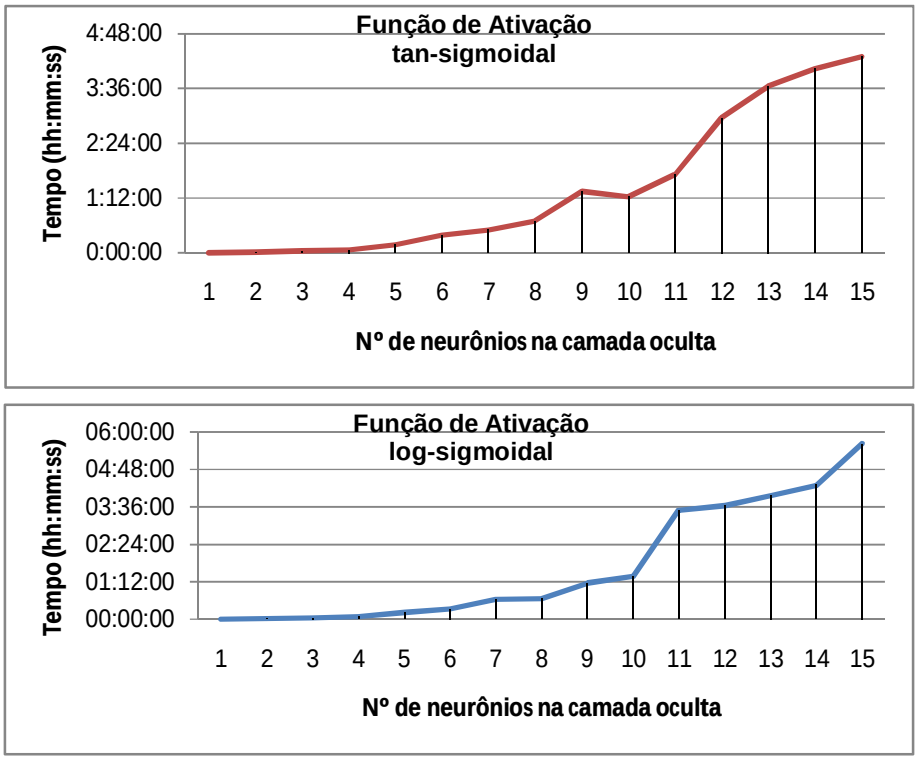


FIGURA 36: REPRESENTAÇÃO GRÁFICA DA (TABELA 4).
FONTE: AUTOR.

As amostras utilizadas para treinamento e teste da rede, como mencionado no capítulo 3, pertencem às famílias e espécies descritas na TABELA 5.

TABELA 5: DESCRITIVO DAS FAMÍLIAS E ESPÉCIES DAS AMOSTRAS UTILIZADAS PARA TREINAMENTO E TESTE DA REDE.

FAMÍLIA	NOME COMUM	ESPÉCIE
LAURACEAE	CANELA	(*Ocotea odorifera*)
	SASSAFRÁS	(*Nectandra* spp.)
	IMBUIA	(*Ocotea porosa*)
MYRTACEAE	EUCALIPTO	(*Eucalyptus* spp.)

FONTE: AUTOR

A quantidade de amostras utilizadas bem como sua separação dentro dos conjuntos de teste e treinamento pode ser vista na TABELA 6.

TABELA 6: QUANTITATIVO DAS AMOSTRAS UTILIZADAS

ESPÉCIE (NOME COMUM)	AMOSTRAS PARA TESTE	AMOSTRAS PARA TREINAMENTO	TOTAL DE AMOSTRAS
CANELA	288	431	**719**
IMBUIA	288	432	**720**
SASSAFRÁS	218	218	**436**
EUCALIPTO	211	316	**527**
TOTAL	**1005**	**1397**	**2402**

FONTE: AUTOR

O conjunto de treinamento possui 60% do total de amostras sendo que os 40% restantes foram utilizados na etapa de teste. Como as amostras de Canela-Sassafrás são em menor quantidade, comparado às outras, tomou-se 50% para treinamento e 50% para teste.

Como critério de parada o MATLAB dispõe-se de restrições quanto ao número de iterações/épocas, erro mínimo, gradiente mínimo, tempo máximo de execução e quantidade de testes falhos consecutivos de validação. Estes critérios são apresentados na TABELA 7.

TABELA 7: CRITÉRIOS DE PARADA ADOTADOS.

TIPO	DESCRIÇÃO	CRITÉRIO APLICADO
Nº de iterações	A rede para ao atingir o valor especificado.	Máximo de 1000 iterações
Erro mínimo	A rede para ao atingir o erro (MSE) mínimo especificado.	≥ 1E-10
Tempo máximo	A rede para ao atingir o tempo máximo especificado.	Infinito
Testes de Validação Cruzada	A rede para ao atingir o valor especificado de falhas consecutivas de validação.	10 falhas consecutivas
Gradiente Mínimo	A rede para ao atingir o valor mínimo do gradiente.	≥ 1E-10

FONTE: AUTOR

Uma observação pertinente é quanto ao valor mínimo para o gradiente. Segundo Hagan *et al.* (1996), valores pequenos para o gradiente, em geral, indicam que a rede está próxima do ótimo. Quando ocorre de o gradiente ser pequeno mas o erro (MSE) resultante alto, a rede pode estar perdendo a capacidade de generalização.

A técnica de validação cruzada faz com que o treinamento seja interrompido a cada ciclo, realizando-se uma estimação de erro sobre uma parte do conjunto de teste. Se o erro medido no conjunto de teste apresentar crescimento consecutivo por 10 interações (ciclos/epoch) enquanto o erro de treinamento diminui, o processo é encerrado. O objetivo da validação cruzada aplicada ao treinamento de uma RNA com aprendizagem do tipo *backpropagation* é descobrir o momento exato em que a rede começa a perder generalização, ou seja, ela "decora" os padrões de entrada.

Os pesos e bias são valores estocásticos pertencentes ao intervalo [-1,1]. Para a atualização dos pesos e bias optou-se por utilizar o *modo por lote* (*batch mode* ou *batching*), onde todas as entradas (padrões) são apresentadas à rede antes da atualização dos pesos e bias. De acordo com Demuth *et. al.* (2009) e Hangan (2001) o *batch mode* favorece estimativas mais exatas do gradiente pois considera todo o conjunto, diferente do *modo incremental* (incremental mode) onde os pesos são atualizados para cada entrada.

Quanto à atualização dos pesos e bias é importante salientar uma particularidade do comando ***newff()***, que utiliza uma função de aprendizagem responsável exclusivamente pelas suas atualizações. O treinamento se dá pelo método de Levenberg-Marquardt, com os pesos sendo gerados conforme a equação (52) mas passando, posteriormente, por um filtro. O algoritmo de filtragem, *learngdm.m*, adotado está baseado no método do gradiente descendente com momento, como foi descrito na seção 4.3.2.

De maneira geral, o algoritmo *learngdm* pode ser representado pela equação (63) (DEMUTH *et. al.* (2009)).

$$\boldsymbol{dW = mc \cdot dWprev + (1 - mc) \cdot lr \cdot gW} \qquad \textbf{(63)}$$

Onde:

dW: vetor de pesos "filtrados";

dWprev: vetor de pesos antes de passar pelo filtro;

mc: constante de momento (vide seção 3.3.2);

lr: constante de aprendizagem (vide seção 3.3.2);

gW: vetor gradiente

Como os valores para as constantes, momento e aprendizagem, são definidos empiricamente, adotou-se os padrões (default) do MATLAB como mostra a TABELA 8.

TABELA 8: VALORES PADRÕES DA FUNÇÃO "*learngdm.m*".

CONSTANTE	VALOR
Fator de Momento	0,9
Taxa de aprendizagem	0,01

FONTE: AUTOR.

O método de Levenberg-Marquardt disponível no MATLAB utiliza um parâmetro adaptativo $\boldsymbol{\mu_k}$, como mostra a seção 4.3.3, com seus fatores de incremento e decremento além de um limite superior para controle. Os valores adotados seguem o default do MATLAB conforme explicita a TABELA 9.

TABELA 9: VALORES PARA O PARÂMETRO ADAPTATIVO μ_k.

PARÂMETRO μ_k	DESCRIÇÃO	VALOR
μ_k	Valor inicial para o parâmetro adaptativo de Levemberg Marquadant	0,001
incremento para μ_k	fator de incremento para o μ_k	10
decremento para μ_k	fator de decremento para o μ_k	0,1
valor máximo para μ_k	limite superior para o valor de μ_k	1,00E+10

FONTE: AUTOR.

Durante os teste foram utilizadas as funções de transferência *logarítmica sigmoidal* e *tangente hiperbólica sigmoidal* na camada intermediária. Para a saída, composta por números naturais, foi mantida a função linear.

5.3 ANÁLISE DOS RESULTADOS

Inicialmente as amostras indicadas na TABELA 6 foram analisadas em conjunto, ou seja, as 4 espécies todas juntas (Canela, Imbuia, Sassafrás e Eucalipto). Assim o conjunto de treinamento atingiu a dimensão de 1397 (amostras) por 1556 (variáveis), totalizando uma matriz com 2.173.732 elementos. Esses elementos representam o valor de absorbância por *número de onda* para o intervalo 3996 - 9995 (cm^{-1}) com resolução de 4 (cm^{-1})[21].

Como mencionado na seção 3.5, os dados analisados são brutos, ou seja, não passaram por nenhum tipo de filtro estatístico além de conterem espectros relativos a diferentes tipos de corte[22] (transversal, tangencial e longitudinal). O objetivo é mostrar a potencialidade de uma rede neural, de forma independente, na classificação dessa informação.

5.3.1 Utilizando a função de ativação *tangente hiperbólica sigmoidal*

Utilizando a função de ativação *tangente hiperbólica sigmoidal* na camada intermediária, mas mantento a função linear na saída, a rede foi treinada e testada com a quantidade de neurônios variando de 1 a 15 por 10 ensaios, ou seja, repetiu-se o treinamento e teste para uma mesma arquitetura de rede.

[21] Na seção 2.3 são descritos conceitos de espectrometria, como absorbância.
[22] Para mais informações sobre os planos de corte, vide seção 2.1.

A importância da repetição dos ensaios deve-se a aleatoriedade da escolha dos pesos e bias, visto que estes tem grande influência no desempenho da rede. Pode ocorrer que para uma determinada escolha de pesos e bias, em um ensaio, a rede apresente um desempenho ruim, com erro grande. Realizando-se novos ensaios, pode ocorrer de os pesos e bias escolhidos aleatóriamente virem a melhorar o desempenho da rede, minimizando o erro.

As tabelas 10 e 11 mostram o erro obtido em cada etapa de ensaio para uma determinada quantidade de neurônios na camada escondida. Também foi destacado qual foi o valor máximo e mínimo do erro obtido em cada ensaio. Os valores em vermelho destacam os melhores resultados atingidos considerando os dez ensaios realizados.

O critério para o cálculo do erro foi o mesmo adotado por Demuth *et. al.* (2009), conforme a equação (64), onde $\boldsymbol{Q}$ representa a quantidade de padrões de entrada (amostra disponíveis).

$$\boldsymbol{mse} = \frac{1}{Q}\sum_{k=1}^{Q} e(k^2) = \frac{1}{Q}\sum_{k=1}^{Q}(t(k) - a(k))^2 \qquad \textbf{(64)}$$

TABELA 10: RELAÇÃO DE ERRO (MSE) OBTIDO EM CADA ENSAIO (1 A 5) PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Ensaio	**1**		**2**		**3**		**4**		**5**	
Nº Neurônios	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**
1	4,700E-03	9,300E-03	4,703E-03	9,343E-03	3,974E-03	1,043E-02	4,044E-03	8,726E-03	5,495E-03	9,813E-03
2	3,300E-03	1,570E-02	3,341E-03	1,574E-02	1,650E-03	1,569E-02	2,501E-03	1,291E-02	3,924E-03	1,022E-02
3	2,044E-04	1,120E-02	2,044E-04	1,119E-02	1,038E-09	7,725E-02	1,562E-09	8,259E-03	2,441E-09	7,912E-03
4	5,700E-03	4,140E-02	5,682E-03	4,138E-02	3,811E-06	5,377E-03	1,983E-03	5,688E-03	1,285E-03	4,483E-02
5	1,200E-03	1,560E-02	1,238E-03	1,557E-02	9,419E-04	6,649E-03	5,033E-04	3,311E-02	9,791E-04	2,092E-02
6	1,900E-03	8,400E-03	1,931E-03	8,388E-03	6,117E-04	3,612E-02	9,446E-09	4,913E-03	5,119E-04	1,055E-02
7	1,100E-03	1,250E-02	1,092E-03	1,254E-02	2,314E-03	8,067E-03	2,976E-03	1,734E-02	1,143E-03	5,112E-03
8	6,699E-04	1,070E-02	6,699E-04	1,072E-02	6,432E-08	2,372E-03	8,039E-04	5,989E-03	7,591E-04	4,647E-03
9	1,300E-03	6,000E-03	1,298E-03	5,957E-03	5,810E-04	4,582E-03	9,407E-04	8,095E-03	9,420E-04	1,629E-02
10	8,231E-04	1,170E-02	8,231E-04	1,171E-02	1,193E-03	4,838E-03	1,577E-03	6,993E-03	5,114E-04	7,398E-03
11	6,133E-04	1,070E-02	3,656E-04	4,375E-02	5,259E-04	8,826E-03	1,223E-03	7,909E-03	7,764E-04	6,798E-03
12	1,600E-03	1,750E-02	1,133E-03	1,038E-02	2,018E-03	4,736E-02	9,587E-04	1,068E-02	3,776E-03	1,353E-02
13	6,158E-04	8,200E-03	2,201E-03	9,379E-03	1,793E-10	4,803E-04	3,359E-02	1,356E-01	8,145E-04	7,137E-03
14	3,900E-03	1,500E-02	4,375E-04	8,609E-03	9,589E-04	9,561E-03	8,241E-04	4,575E-03	5,448E-04	9,332E-03
15	1,600E-03	1,090E-02	1,768E-03	1,188E-02	1,796E-03	1,043E-02	1,867E-03	8,022E-03	6,466E-03	3,395E-02
Máximo	5,700E-03		5,682E-03		3,974E-03		3,359E-02		6,466E-03	
Mínimo	2,044E-04		2,044E-04		1,793E-10		1,562E-09		2,441E-09	
Máximo		4,140E-02		4,375E-02		7,725E-02		1,356E-01		4,483E-02
Mínimo		6,000E-03		5,957E-03		4,803E-04		4,575E-03		4,647E-03

FONTE: AUTOR

TABELA 11: RELAÇÃO DE ERRO (MSE) OBTIDO EM CADA ENSAIO (6 A 10) PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Ensaio	**6**		**7**		**8**		**9**		**10**	
Nº Neurônios	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**
1	3,943E-03	1,031E-02	4,044E-03	8,726E-03	3,720E-03	9,117E-03	5,233E-03	1,019E-02	4,164E-03	1,315E-02
2	1,663E-03	8,671E-03	2,501E-03	1,291E-02	3,369E-07	2,247E-03	1,157E-03	1,614E-02	9,838E-06	2,396E-03
3	3,732E-03	1,101E-02	1,562E-09	8,259E-03	1,981E-01	5,658E-01	1,420E-03	1,762E-02	3,548E-03	1,154E-02
4	3,354E-03	1,837E-02	1,983E-03	5,688E-03	1,216E-03	6,566E-03	4,653E-03	1,322E-02	1,904E-03	1,753E-02
5	9,405E-04	6,343E-03	5,033E-04	3,311E-02	7,783E-04	1,875E-02	1,683E-03	1,252E-02	1,559E-03	1,149E-02
6	4,844E-04	1,682E-02	9,446E-09	4,913E-03	1,526E-05	9,875E-03	4,407E-04	8,270E-03	1,636E-03	6,851E-03
7	7,354E-04	4,571E-03	2,976E-03	1,734E-02	2,310E-03	9,684E-03	3,321E-04	2,525E-03	5,394E-04	3,072E-03
8	1,422E-10	3,384E-05	8,039E-04	5,989E-03	7,936E-03	1,955E-02	2,362E-04	5,825E-03	1,933E-03	5,482E-03
9	8,585E-04	7,220E-03	9,407E-04	8,095E-03	1,531E-03	1,099E-02	7,874E-04	9,346E-03	1,385E-03	8,905E-03
10	5,540E-04	7,973E-03	1,577E-03	6,993E-03	9,564E-02	4,713E-01	2,640E-03	1,136E-02	5,798E-03	1,778E-02
11	1,644E-03	1,200E-02	1,223E-03	7,909E-03	7,952E-03	3,130E-02	1,011E-03	1,111E-02	1,505E-03	1,366E-02
12	1,606E-03	8,500E-03	9,587E-04	1,068E-02	1,015E-03	4,853E-03	7,746E-04	4,719E-03	8,093E-04	5,213E-03
13	1,646E-03	1,298E-02	2,555E-03	4,542E-02	2,468E-03	1,387E-02	4,643E-03	1,137E-02	1,570E-04	1,285E-03
14	1,035E-03	5,366E-03	9,460E-04	1,206E-02	3,377E-04	9,155E-03	6,956E-02	1,427E-01	9,838E-04	1,246E-02
15	1,081E-03	8,074E-03	7,883E-04	1,337E-02	1,494E-03	2,318E-02	1,852E-03	1,383E-02	7,918E-04	7,707E-03

Máximo	3,943E-03		4,044E-03		1,981E-01		6,956E-02		5,798E-03	
Mínimo	1,422E-10		1,562E-09		3,369E-07		2,362E-04		9,838E-06	
Máximo		1,837E-02		4,542E-02		5,658E-01		1,427E-01		1,778E-02
Mínimo		3,384E-05		4,913E-03		2,247E-03		2,525E-03		1,285E-03

FONTE: AUTOR

Na TABELA 12 tem-se os valores médios, relativos aos dez ensaios, para as etapas de treinamento e teste. Para facilitar a vizualização da informação, tem-se a representação gráfica desses dados na FIGURA 37.

TABELA 12: VALORES MÉDIOS DOS ERROS RELATIVOS AOS DEZ ENSAIOS, OBTIDOS NAS ETAPAS DE TREINAMENTO E TESTE PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Erros Médios		
Nº Neurônios	**Treino**	**Teste**
1	0,00440	0,009911
2	0,00200	0,011263
3	0,02072	0,072999
4	0,00278	0,020005
5	0,00103	0,017406
6	0,00075	0,01151
7	0,00155	0,009274
8	0,00138	0,007132
9	0,00106	0,008548
10	0,01111	0,055802
11	0,00168	0,015397
12	0,00146	0,013343
13	0,00487	0,024576
14	0,00795	0,022885
15	0,00195	0,014134
Máximo	0,02072	0,07300
Mínimo	0,00075	0,00713

FONTE: AUTOR.

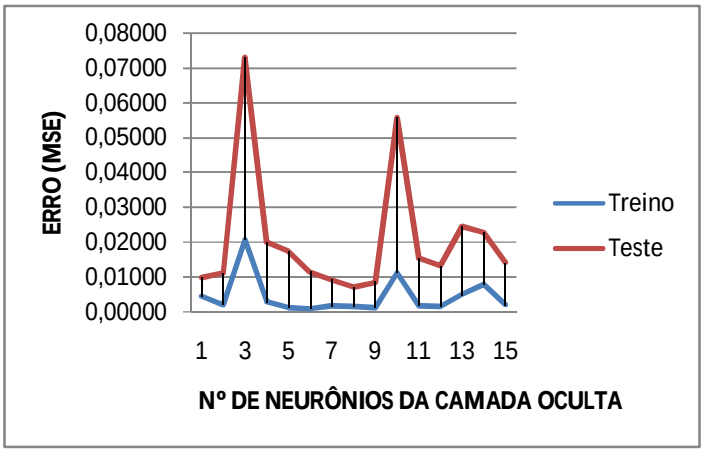


FIGURA 37: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE. FONTE: AUTOR.

Analisando a TABELA 12 observa-se que o melhor resultado médio durante o treinamento ocorreu com seis neurônios na camada intermediária, enquanto que na etapa de teste isso aconteceu para oito neurônios na camada escondida. Nas Tabelas 10 e 11, referentes aos ensaios, os melhores resultados para treinamento e teste ocorreram no sexto ensaio, para oito neurônios na camada intermediária.

Para cada espécie utilizada foi atribuido um valor de referência, variando de 1 a 4, que está relacionado à saída da rede. Esses valores foram escolhidos de forma arbitrária com o intuíto de facilitar a identificação das espécies após a classificação. A TABELA 13 ilustra esses valores.

TABELA 13: VALORES DE REFERÊNCIA PARA AS QUATRO ESPÉCIES UTILIZADAS NO TREINAMENTO E TESTE DE UMA RNA COM FUNÇÃO DE TRANSFERÊNCIA TAN-SIGMOIDAL.

Espécie	Valor de Referência
Canela	1
Imbuia	2
Sassafrás	3
Eucalipto	4

FONTE: AUTOR

As quantidades de amostras para teste e treinamento estão descritas na TABELA 6. As Figuras 40 e 41 representam as saídas da rede levando em consideração o melhor resultado médio durante a etapa de treinamento, o que ocorreu para uma rede com seis neurônios na camada escondida, conforme mostra a TABELA 12 (coluna treino, grifo em vermelho). Os erros encontrados foram de 7,5E-4 e 1,151E-2 respectivamente

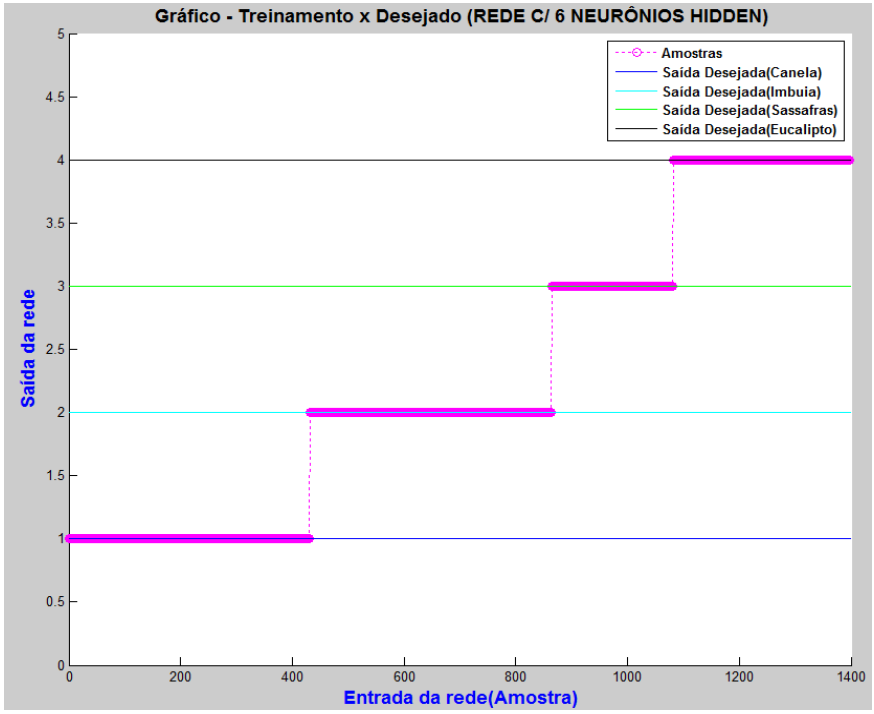


FIGURA 38: GRÁFICO - (TREINAMENTO x DESEJADO) PARA A REDE COM SEIS NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

Para obter o erro médio quadrático durante o treinamento, a rede recomputa os mesmos dados usados para treiná-las. Em uma rede ideal a saída computada sobre o conjunto de treinamento deveria apresentar erro igual a zero, embora na prática seja possível obter valores bem pequenos.

Erros próximos de zero, para o conjunto de treinamento, não significam uma ótima capacidade de generalização. Quando a rede "decora" os padrões de entrada, o erro de treinamento torna-se pequeno mas os obtidos na etapa de teste podem ser grandes.

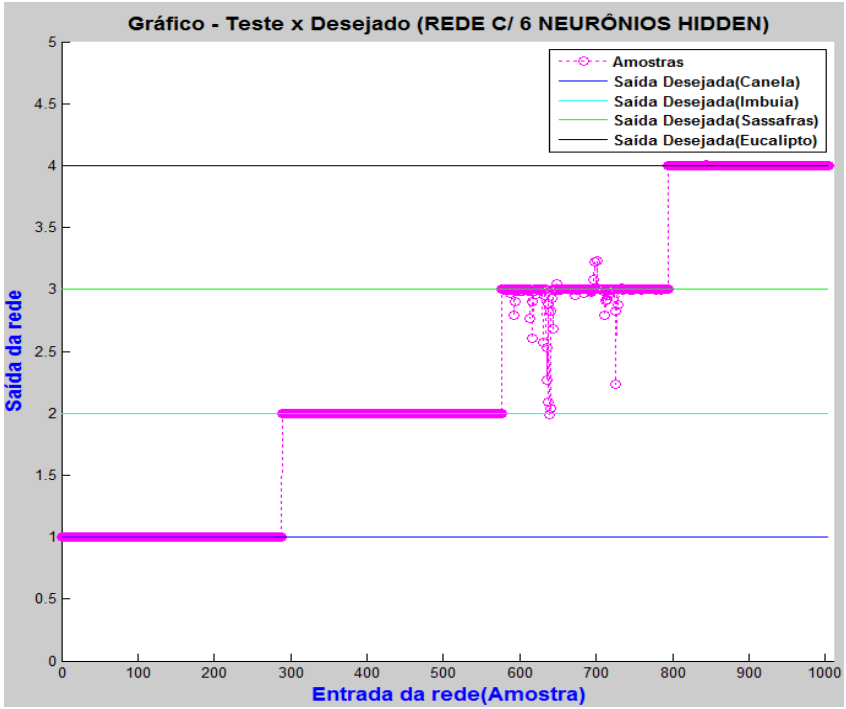


FIGURA 39: GRÁFICO - (TESTE x DESEJADO) PARA A REDE COM SEIS NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

Observou-se que o desempenho da rede para classificação das espécies foi bem satisfatório para as amostras de Canela, Imbuia e Eucalipto, embora tenha havido uma dispersão significativa com os exemplares de Sassafrás.

De acordo com as Tabelas 11 e 12, o melhor desempenho na fase de teste ocorreu para uma rede com oito neurônios na camada escondida, no sexto ensaio. Os valores do erro para treinamento e teste foram, respectivamente, de 1,422E-10 e 3,384E-05. As Figuras 42 e 43 ilustram o desempenho da rede, nesse caso, nas fases de treinamento e teste.

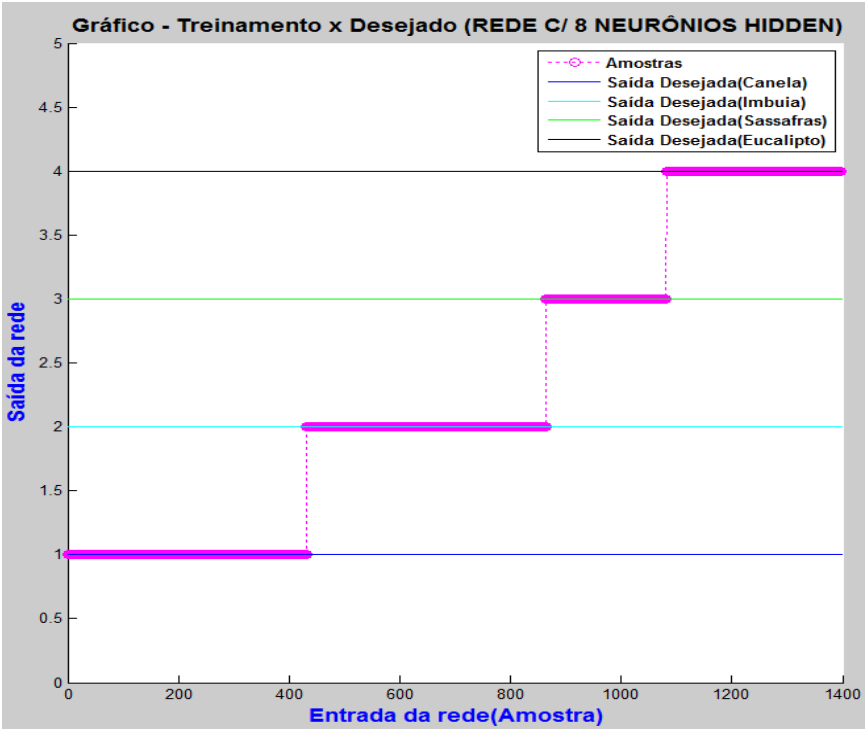


FIGURA 40: GRÁFICO - (TREINAMENTO x DESEJADO) PARA A REDE COM OITO NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

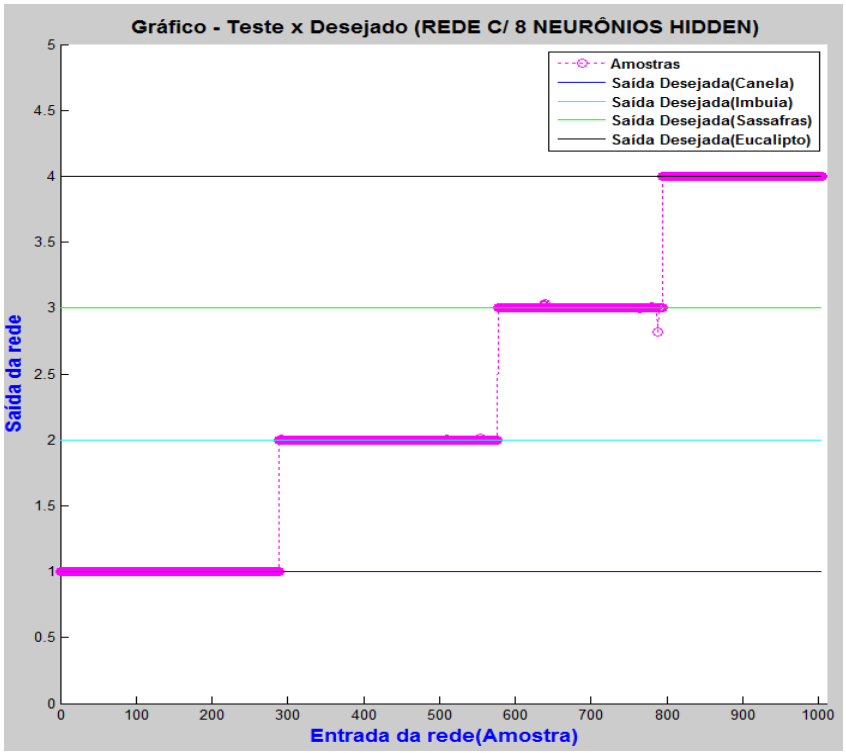


FIGURA 41: GRÁFICO - (TESTE x DESEJADO) PARA A REDE COM OITO NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

Pode-se observar que a rede com oito neurônios na camada oculta, FIGURA 41, apresentou melhor desempenho na etapa de teste, se comparado à rede com seis neurônios na camada escondida, FIGURA 39. Isso exemplifica uma situação onde um bom desempenho na etapa de treinamento não resulta em um melhor resultado na fase de teste. Foi possível observar que para ambos os caso ainda manteve-se uma pequena distorção nas amostras de Sassafrás, comparando-se às outras espécies. A amostra de Sassafrás com maior discrepancia apresentou o valor de 2,822, um desvio de 0,178 em relação à referência.

Considerando-se uma tolerância ao erro de 5%, a matriz de confusão para a classificação de quatro espécies utilizando uma RNA, do tipo proposto, com 8 neurônios na camada escondida encontra-se na Tabela 14. Considerando uma tolerância de 2% a matriz de confusão está descrita na Tabela 15.

TABELA 14: MATRIZ DE CONFUSÃO PARA UMA RNA COM 8 NEURÔNIOS NA CAMADA OCULTA E TOLERÂNCIA DE 5%.

Espécies	**Canela**	**Imbuia**	**Sassafrás**	**Eucalipto**	**Não Classificado**	**Total de Amostras**
Canela	288	0	0	0	0	288
Imbuia	0	288	0	0	0	288
Sassafrás	0	0	218	0	0	218
Eucalipto	0	0	0	211	0	211

FONTE: AUTOR

TABELA 15: MATRIZ DE CONFUSÃO PARA UMA RNA COM 8 NEURÔNIOS NA CAMADA OCULTA E TOLERÂNCIA DE 2%.

Espécies	**Canela**	**Imbuia**	**Sassafrás**	**Eucalipto**	**Não Classificado**	**Total de Amostras**
Canela	288	0	0	0	0	288
Imbuia	0	288	0	0	0	288
Sassafrás	0	0	217	0	1	218
Eucalipto	0	0	0	211	0	211

FONTE: AUTOR

Para 5% de tolerância houve um aproveitamento de 100% e para 2% houve um reconhecimento de 99,90% das amostras.

A TABELA 16 mostra o tempo de treinamento por ensaio e o tempo total para completar-se os dez ensaios realizados. O tempo médio por ensaio foi de 21h 06m 53s e o tempo total foi de 211h 08m 50s.

TABELA 16: TEMPO DE TREINAMENTO POR ENSAIO PARA A REDE COM FUNÇÃO TAN-SIGMOIDAL NA CAMADA INTERMEDIÁRIA

Ensaio	Tempo
1	25:43:29
2	38:7:18
3	12:56:42
4	13:40:25
5	27:17:22
6	11:33:24
7	26:40:38
8	21:20:31
9	15:54:10
10	17:54:47
Total	**211:8:50**

FONTE: AUTOR.

5.3.2 Utilizando a função de ativação logarítimica - sigmoidal

A próxima etapa de testes ocorreu com a alteração da função de ativação da camada intermediária para a log-sigmoidal, embora manteve-se a função linear na camada de saída da rede.

As Tabelas 17 e 18 mostram os erros obtidos em cada etapa de ensaio para uma determinada quantidade de neurônios na camada escondida. Os valores em vermelho destacam os melhores resultados atingidos durante os dez ensaios realizados.

TABELA 17: RELAÇÃO DE ERRO (MSE) OBTIDO EM CADA ENSAIO (1 A 5) PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Ensaios	**1**		**2**		**3**		**4**		**5**	
Nº Neurônios	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**
1	0,004691282	0,010392	0,004670729	0,009839	0,003930034	0,009517	0,003837	0,009161	0,003731	0,011354
2	2,60E-07	0,002719	0,00500321	0,011757	0,002089605	0,005431	0,001742	0,009275	0,0034	0,013765
3	0,003960905	0,007893	0,001865263	0,017617	0,001530498	0,003941	2,62E-07	0,003729	0,002383	0,005363
4	0,000601003	0,002517	0,004129858	0,011352	0,005476219	0,011162	0,001587	0,012097	0,002034	0,007185
5	0,001151252	0,00412	0,000152357	0,001849	2,29E-08	0,003538	0,000428	0,005731	0,00097	0,013766
6	0,00056563	0,003346	0,000531755	0,028396	0,001077295	0,010542	0,001687	0,006252	0,001499	0,014824
7	0,000735994	0,016731	0,001826203	0,020363	0,001400819	0,007265	0,001528	0,019385	0,001294	0,006461
8	3,15E-06	0,000542	0,001193371	0,051277	0,003727768	0,009625	0,001683	0,007039	0,001401	0,00612
9	0,00189387	0,003081	0,000854007	0,009095	0,001903778	0,006805	0,001011	0,006329	0,002751	0,01008
10	0,000623391	0,018646	0,000934511	0,006138	0,001129426	0,008438	0,001611	0,008586	0,001128	0,008001
11	0,000572479	0,010758	0,001969635	0,011543	0,00230826	0,014712	0,001429	0,00726	0,0011	0,013642
12	0,000802395	0,011978	0,000889299	0,005352	0,00124196	0,00909	0,000629	0,006799	0,001588	0,006233
13	0,001049249	0,011187	0,001195466	0,004784	0,00079121	0,019124	0,002216	0,016681	0,001102	0,007616
14	0,000975234	0,008425	0,001626045	0,012241	0,001871638	0,00758	0,001166	0,00919	0,001419	0,005807
15	0,001401675	0,01248	0,001606233	0,004673	0,000753847	0,01473	0,001281	0,008342	0,002238	0,014554
Máximo	0,004691282		0,00500321		0,005476219		0,003837		0,003731	
Mínimo	2,60338E-07		0,000152357		2,29107E-08		2,62E-07		0,00097	
Máximo		0,018646		0,051277		0,019124		0,019385		0,014824
Mínimo		0,000542		0,001849		0,003538		0,003729		0,005363

FONTE: AUTOR.

TABELA 18: RELAÇÃO DE ERRO (MSE) OBTIDO EM CADA ENSAIO (6 A 10) PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Ensaios	**6**		**7**		**8**		**9**		**10**	
Nº Neurônios	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**	**Treino**	**Teste**
1	0,004694	0,009886	0,003486	0,014324	0,005483	0,011412	0,004986	0,009399	0,003767	0,009348
2	0,00113	0,00871	0,001723	0,008111	0,004379	0,010046	0,004253	0,00995	0,001686	0,007021
3	0,009263	0,053149	0,002214	0,063576	0,001501	0,016958	0,003843	0,009693	0,003025	0,028698
4	0,002256	0,005745	0,001339	0,00541	0,000967	0,029331	0,009622	0,017045	0,00053	0,008055
5	0,001468	0,006091	0,000954	0,005692	0,000813	0,022749	0,002072	0,008853	0,000238	0,00799
6	0,001985	0,005587	0,000995	0,005808	0,001657	0,015329	0,0007	0,011923	0,001822	0,008146
7	0,001299	0,005167	0,00092	0,010022	0,001121	0,012087	0,000677	0,004785	1,07E-06	0,001132
8	0,000116	0,009892	0,000173	0,00572	0,001257	0,005501	0,01626	0,040457	0,00065	0,003572
9	0,001355	0,008482	0,000759	0,008253	0,002651	0,006004	0,002183	0,020254	0,000634	0,012208
10	0,004753	0,012254	0,001024	0,01002	0,002987	0,0182	0,001738	0,005764	0,001128	0,006671
11	0,007974	0,042748	0,002898	0,011115	0,001598	0,007433	0,001178	0,007542	0,000908	0,007118
12	0,000996	0,011475	0,001767	0,021268	0,000703	0,012494	0,001599	0,012543	0,010541	0,040009
13	0,002759	0,009147	0,000617	0,004578	0,002469	0,009413	0,000509	0,005939	0,002244	0,007731
14	0,001842	0,009379	0,003435	0,011305	0,001018	0,007161	0,000157	0,003573	0,001882	0,008386
15	0,003036	0,009915	0,001319	0,00884	0,000862	0,008968	0,001343	0,01059	0,00117	0,012121
Máximo	0,009263		0,003486		0,005483		0,01626		0,010541	
Mínimo	0,000116		0,000173		0,000703		0,000157		1,07E-06	
Máximo		0,053149		0,063576		0,029331		0,040457		0,040009
Mínimo		0,005167		0,004578		0,005501		0,003573		0,001132

FONTE: AUTOR.

Na TABELA 19 tem-se os valores médios, relativos aos dez ensaios, para as etapas de treinamento e teste. Para facilitar a vizualização da informação, tem-se a representação gráfica na FIGURA 44.

TABELA 19: VALORES MÉDIOS DOS ERROS RELATIVOS AOS DEZ ENSAIOS, OBTIDOS NAS ETAPAS DE TREINAMENTO E TESTE PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

	Erros Médios	
Nº Neurônios	**Treino**	**Teste**
1	0,004328	0,010463
2	0,002541	0,008679
3	0,002959	0,021062
4	0,002854	0,01099
5	0,000825	0,008038
6	0,001252	0,011015
7	0,00108	0,01034
8	0,002647	0,013974
9	0,001599	0,009059
10	0,001706	0,010272
11	0,002194	0,013387
12	0,002076	0,013724
13	0,001495	0,00962
14	0,001539	0,008305
15	0,001501	0,010521

Máximo	0,004328	0,021062
Mínimo	0,000825	0,008038

FONTE: AUTOR.

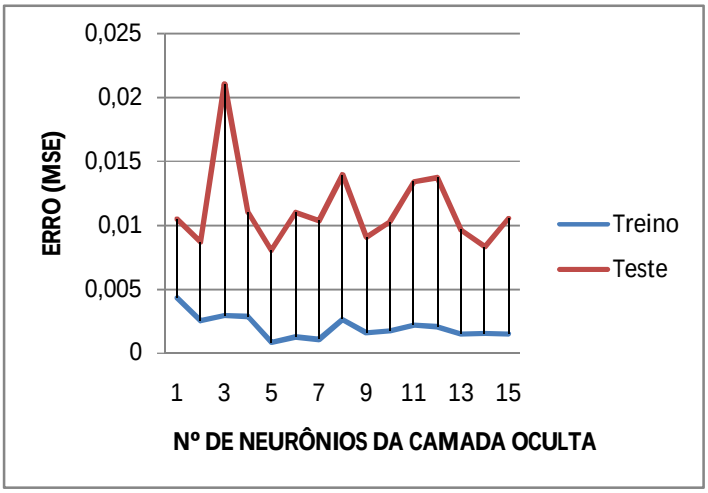


FIGURA 42: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE. FONTE: AUTOR.

Observa-se que o melhor desempenho médio ocorreu para uma rede com cinco neurônios na camada escondida. De acordo com a TABELA 17, no terceiro ensaio ocorreu o melhor desempenho para treinamento, com erro de 2,29E-08 e cinco neurônios na camada oculta. Entretanto, no primeiro ensaio, uma rede com oito neurônios mostrou-se mais eficiente quanto ao teste.

As figuras 43 e 44 representam graficamente as saídas, para treinamento e teste, da rede com cinco neurônios na camada intermediária.

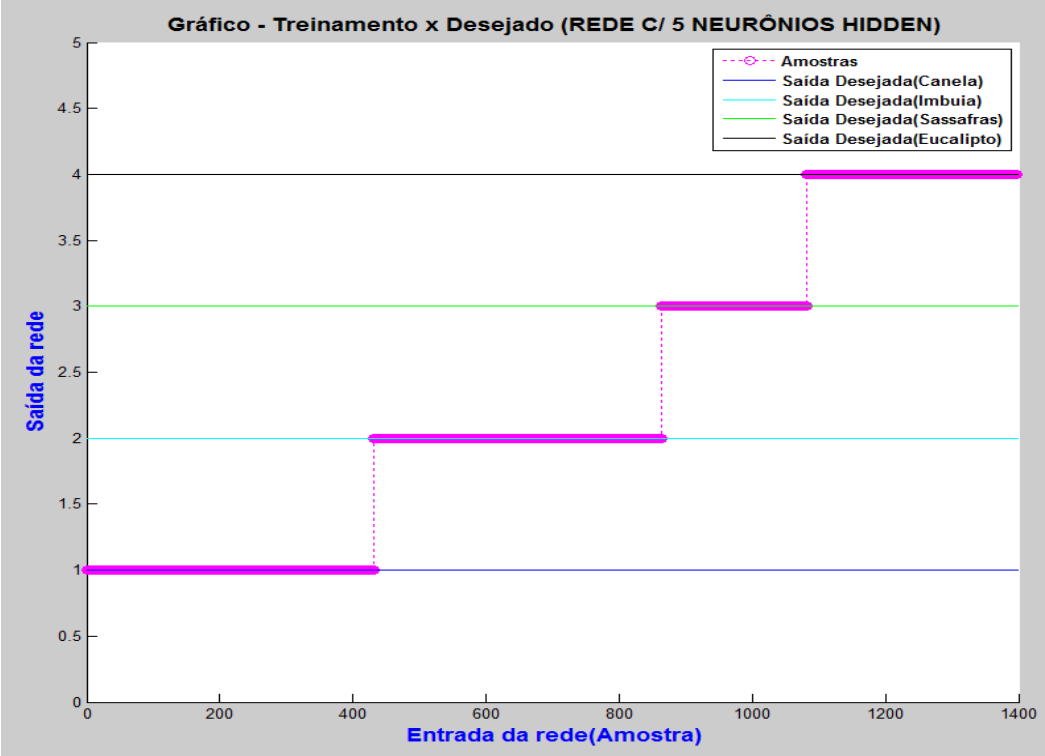


FIGURA 43: GRÁFICO - (TREINAMENTO x DESEJADO) PARA A REDE COM CINCO NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

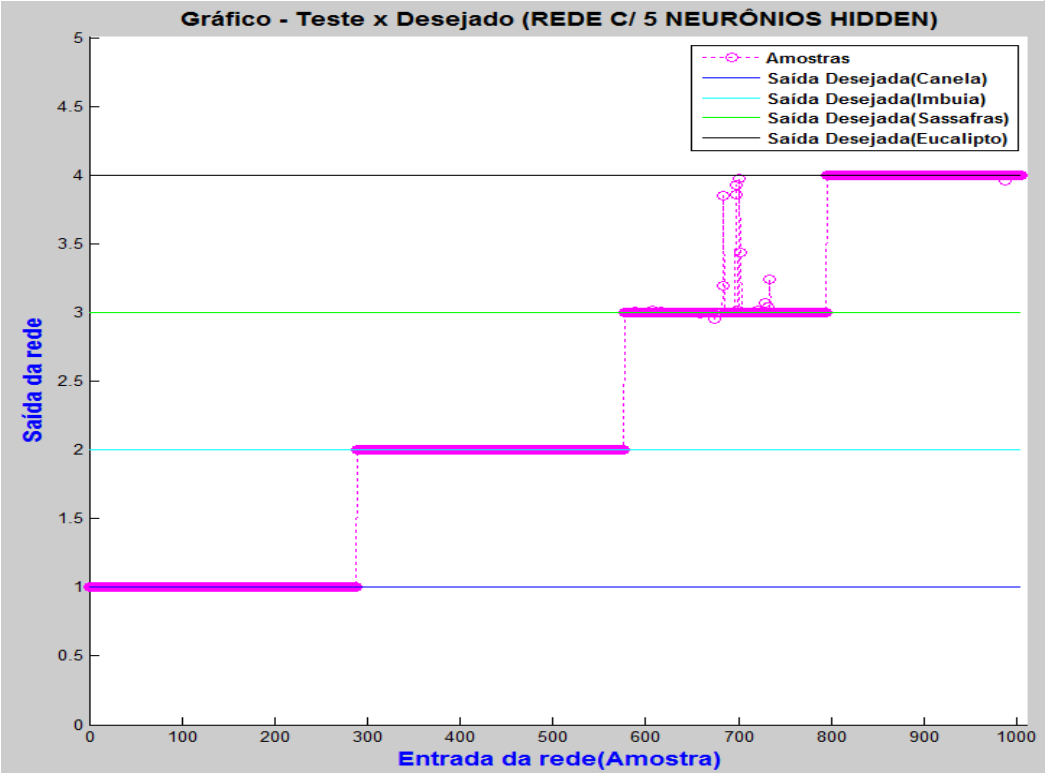


FIGURA 44: GRÁFICO - (TESTE x DESEJADO) PARA A REDE COM CINCO NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

Observa-se na FIGURA 44 que há uma discrepância em algumas amostras de Sassafrás enquanto as espécies Canela, Imbuia e Eucalipto não apresentam grandes variações.

A TABELA 20 apresenta o desvio padrão da saída da rede na etapa de teste, para cada espécie.

TABELA 20: DESVIO PADRÃO, POR ESPÉCIES, PARA A RNA COM CINCO E OITO NEURÔNIOS NA CAMADA OCULTA.

5 NEURÔNIOS	
Espécie	**Desvio Padrão**
Canela	9,38521E-06
Imbuia	0,000178503
Sassafrás	0,126213013
Eucalipto	0,002636818

8 NEURÔNIOS	
Espécie	**Desvio Padrão**
Canela	0,016950818
Imbuia	0,002137345
Sassafrás	0,045897917
Eucalipto	2,91881E-05

FONTE: AUTOR.

Uma representação gráfica, para facilitar a compreenção das informações contidas na TABELA 20, é ilustrada na FIGURA 47.

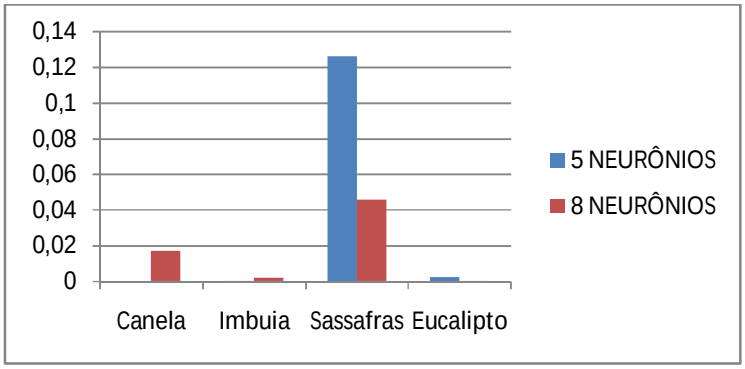


FIGURA 45: REPRESENTAÇÃO GRÁFICA DA (TABELA 20).
FONTE: AUTOR.

As figuras 46 e 47 representam a saída da rede para a arquitetura com oito neurônios na camada escondida.

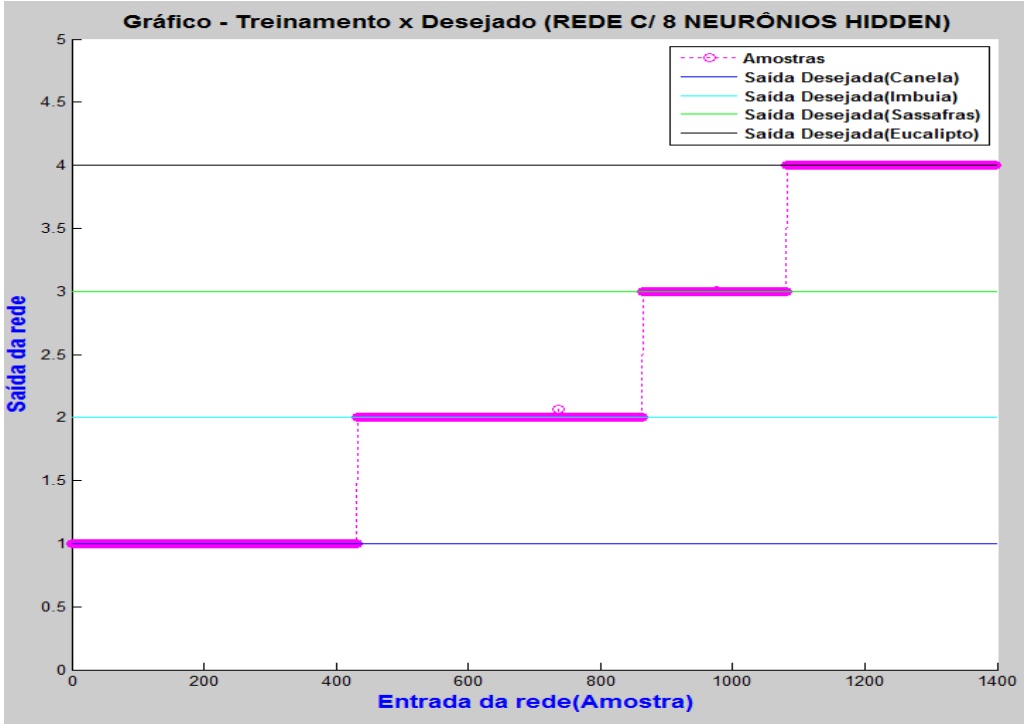


FIGURA 46: GRÁFICO - (TREINAMENTO x DESEJADO) PARA A REDE COM OITO NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

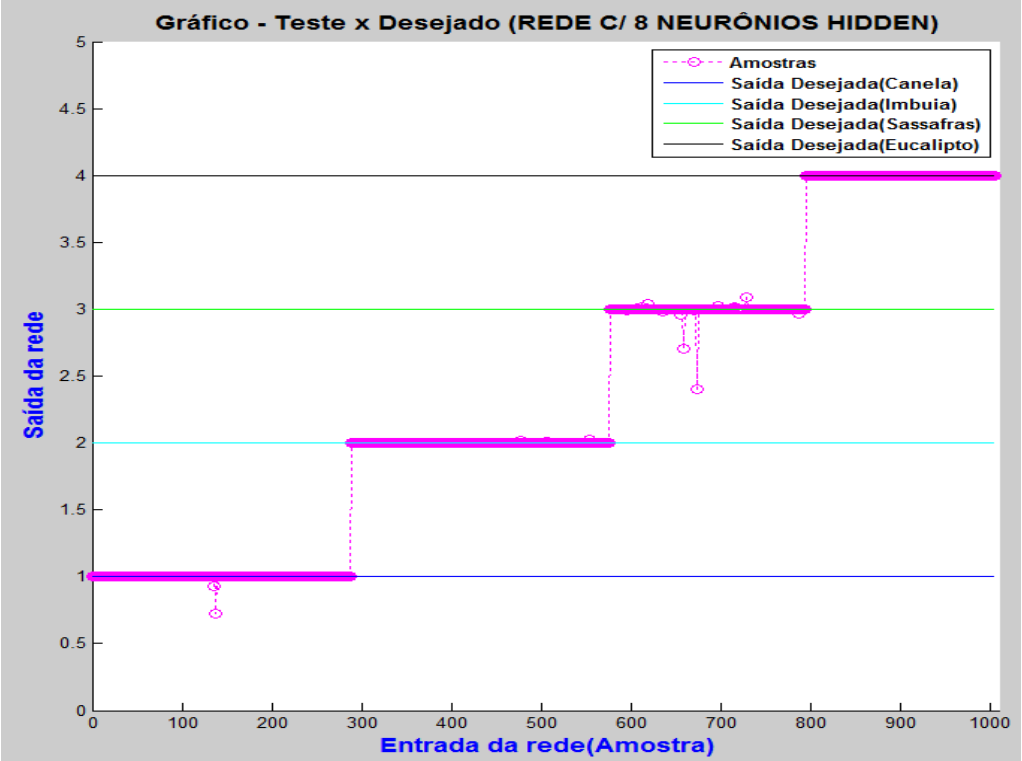


FIGURA 47: GRÁFICO - (TESTE x DESEJADO) PARA A REDE COM OITO NEURÔNIOS NA CAMADA ESCONDIDA.
FONTE: AUTOR.

Ao análisar a FIGURA 47, percebeu-se que a rede com oito neurônios na camada intermediária apresentou um pequeno desvio no reconhecimento de amostras de Canela e Sassafrás, tendo este último um desvio maior. Ao examinar a FIGURA 46 notou-se que o decréscimo de desempenho da rede com cinco neurônio na camada escondida foi quase que exclusivamente devido às amostras de Sassafrás.

Considerada uma tolerância ao erro de 5%, a matriz de confusão para a classificação das quatro espécies, utilizando uma RNA do tipo proposto com 8 neurônios na camada escondida, é descrita pela TABELA 21.

TABELA 21: MATRIZ DE CONFUSÃO PARA UMA RNA COM 8 NEURÔNIOS NA CAMADA OCULTA E TOLERÂNCIA DE 5%.

Espécies	Canela	Imbuia	Sassafrás	Eucalipto	Não Classificado	Total de Amostras
Canela	286	0	0	0	2	288
Imbuia	0	288	0	0	0	288
Sassafrás	0	0	216	0	2	218
Eucalipto	0	0	0	211	0	211

A TABELA 22 descreve a matriz de confusão, para o mesmo caso, mas tomando uma tolerância de 2%.

TABELA 22: MATRIZ DE CONFUSÃO PARA UMA RNA COM 8 NEURÔNIOS NA CAMADA OCULTA E TOLERÂNCIA DE 2%.

Espécies	Canela	Imbuia	Sassafrás	Eucalipto	Não Classificado	Total de Amostras
Canela	286	0	0	0	2	288
Imbuia	0	288	0	0	0	288
Sassafrás	0	0	215	0	3	218
Eucalipto	0	0	0	211	0	211

Para 5% de tolerância houve um aproveitamento de 99,60% e para 2% houve um reconhecimento de 99,50% das amostras.

A TABELA 23 mostra o tempo de treinamento em cada um dos dez ensaios realizados. O tempo médio por ensaio foi de 25h 32m 45s e o tempo total foi de 255h 27m 37s.

TABELA 23: TEMPO DE TREINAMENTO POR ENSAIO PARA A REDE COM FUNÇÃO LOG-SIGMOIDAL NA CAMADA INTERMEDIÁRIA

Ensaio	Tempo
1	28:35:43
2	24:52:49
3	21:25:00
4	30:54:35
5	27:59:38
6	25:00:15
7	23:05:04
8	24:50:21
9	23:11:51
10	25:32:15
Total	255:27:37

FONTE: AUTOR

5.3.3 Testes com conjuntos de três amostras

Para efeito de comparação foram feitos testes com a combinação de três das quatros espécies utilizadas. O objetivo foi verificar se havia alguma influência de uma determinada espécie sobre às outras. Optou-se por utilizar a função de ativação tangente hiperbólica-sigmoidal, pois foi a que apresentou melhores resultados nos teste anteriores[23].

A TABELA 24 apresenta a combinação de amostras utilizadas, foi mantida a mesma numeração de referência, conforme mostra a TABELA 13.

TABELA 24: COMBINAÇÃO DE ESPÉCIES.

Sigla	Nº de Ref.	Espécies
CIS	1-2-3	Canela, Imbuia, Sassafrás
CIE	1-2-4	Canela, Imbuia, Eucalipto
ISE	2-3-4	Imbuia, Sassafrás, Eucalipto
CSE	1-3-4	Canela, Sassafrás, Eucalipto

FONTE: AUTOR.

- **CIS**

A TABELA 25 informa o erro médio (MSE) por neurônios na camada escondida para o conjunto CIS (Canela, Imbuia e Sassafrás), a FIGURA 50

[23] Vide seção 4.3.1.

expressa estas informações no formato gráfico. O menor erro quadrático médio para o teste ocorreu no primeiro ensaio, para dez neurônios na camada escondida, com os valores de 1,09E-07 e 8,58E-6 nas etapas de treinamento e teste, respectivamente. Embora o melhor resultado tenha sido obtido com dez neurônios, a TABELA 25, indica que o menor erro médio ocorreu para três neurônios na camada escondida. As Figuras 49 e 50 ilustram a saída da rede para o melhor caso.

TABELA 25: VALORES MÉDIOS DOS ERROS RELATIVOS AOS CINCO ENSAIOS, OBTIDOS NAS ETAPAS DE TREINAMENTO E TESTE PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Erros Médios		
Nº de Neurônios	**Treino**	**Teste**
1	0,002057942	0,009091
2	0,001700925	0,011092
3	1,40382E-07	0,003328
4	0,001585404	0,008817
5	0,002069089	0,006913
6	0,00164232	0,013131
7	0,00214735	0,084058
8	0,001180134	0,033517
9	0,000396389	0,014108
10	0,002766824	0,01086
11	0,001267119	0,06296
12	0,000363641	0,027814
13	0,003440063	0,023832
14	0,002595686	0,018432
15	0,000532933	0,02077
Máximo	0,003440063	0,084058
Mínimo	1,40382E-07	0,003328

FONTE: AUTOR.

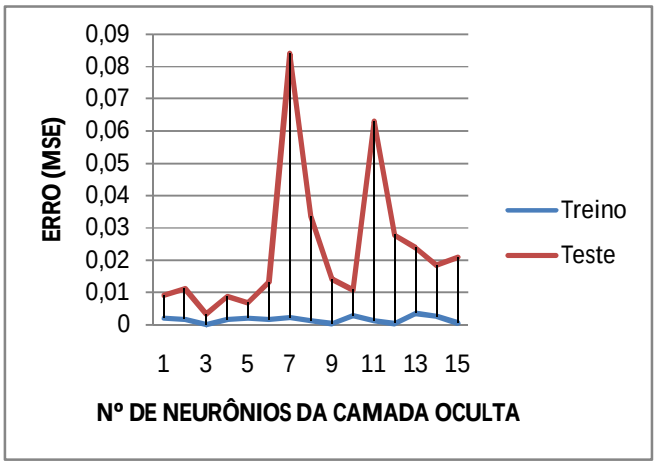


FIGURA 48: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE, PARA O CONJUNTO CIS.
FONTE: AUTOR.

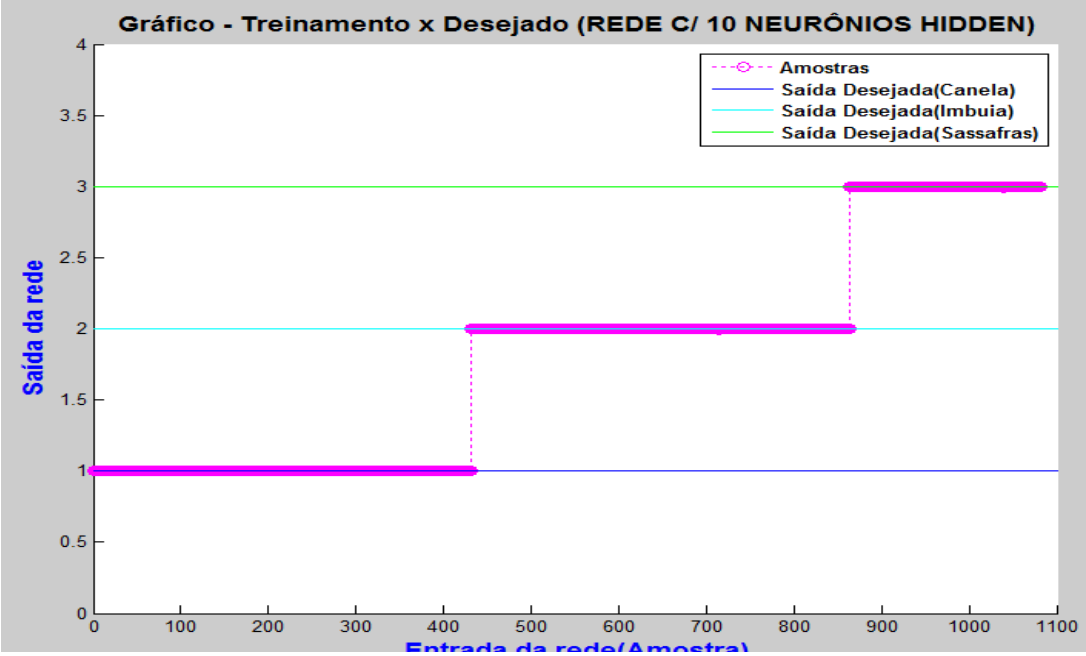


FIGURA 49: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO CIS (ERRO DE 1,09E-07).
FONTE: AUTOR.

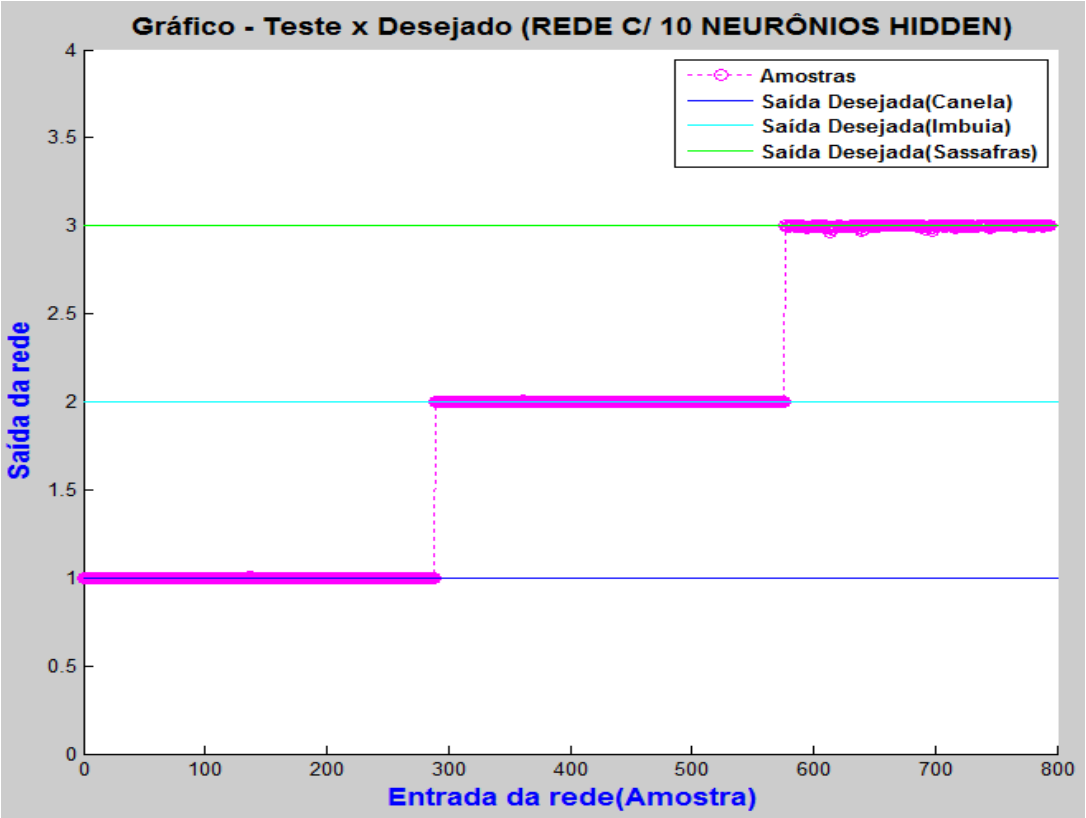


FIGURA 50: DESEMPENHO PARA O CONJUNTO DE TESTE CIS (ERRO DE 8,58E-06).
FONTE: AUTOR

Para uma tolerância de 2% o aproveitamento foi de 100%. A TABELA 26 ilustra a matriz de confusão para este caso.

TABELA 26: MATRIZ DE CONFUSÃO PARA O CONJUNTO CIS.

Espécies	Canela	Imbuia	Sassafrás	Não Classificado	Total de Amostras
Canela	288	0	0	0	288
Imbuia	0	288	0	0	288
Sassafrás	0	0	218	0	218

FONTE: AUTOR

• **CIE**

A TABELA 27 em conjunto com a FIGURA 51 mostram o erro médio (MSE) por neurônios na camada escondida para o conjunto CIE (Canela, Imbuia e Eucalipto). O menor erro quadrático médio para o teste ocorreu no segundo ensaio, para nove neurônios na camada escondida, com os valores de 1,33E-10 e 5,94E-06 nas etapas de treinamento e teste, respectivamente. As Figuras 52 e 53 ilustram a saída da rede para o melhor caso.

TABELA 27: VALORES MÉDIOS DOS ERROS RELATIVOS AOS CINCO ENSAIOS, OBTIDOS NAS ETAPAS DE TREINAMENTO E TESTE PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Erros Médios		
Nº de Neurônios	**Treino**	**Teste**
1	0,312228929	0,313986453
2	0,067454519	0,085652766
3	0,001235826	0,002618132
4	1,93707E-09	3,20441E-05
5	0,00026921	0,001361139
6	0,000136266	0,000516847
7	4,27347E-09	8,13253E-06
8	0,000238305	0,000888045
9	8,14667E-05	0,000339611
10	0,001600112	0,006619015
11	0,000964727	0,003021207
12	0,000523796	0,002032273
13	0,002549807	0,004336531
14	0,000145484	0,000955635
15	0,000122753	0,002581276
Máximo	0,312228929	0,313986453
Mínimo	1,93707E-09	8,13253E-06

FONTE: AUTOR.

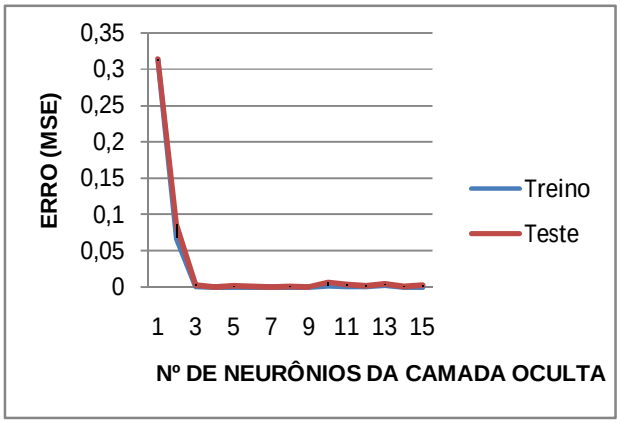


FIGURA 51: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE, PARA O CONJUNTO CIE.
FONTE: AUTOR.

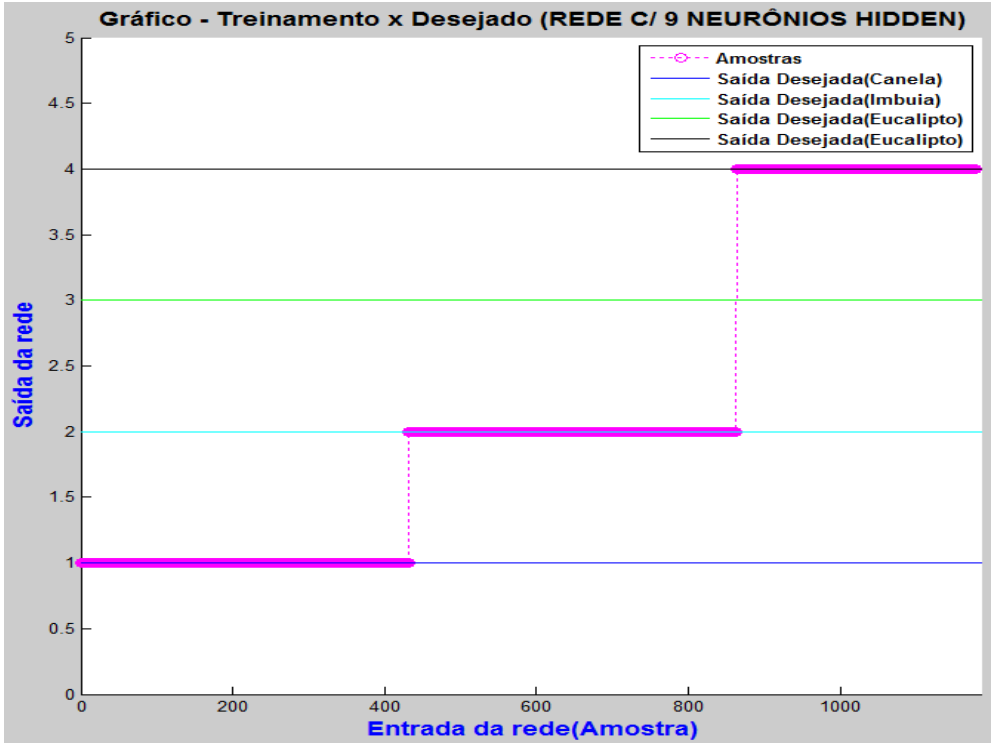


FIGURA 52: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO CIE (ERRO DE 1,33E-10).
FONTE: AUTOR

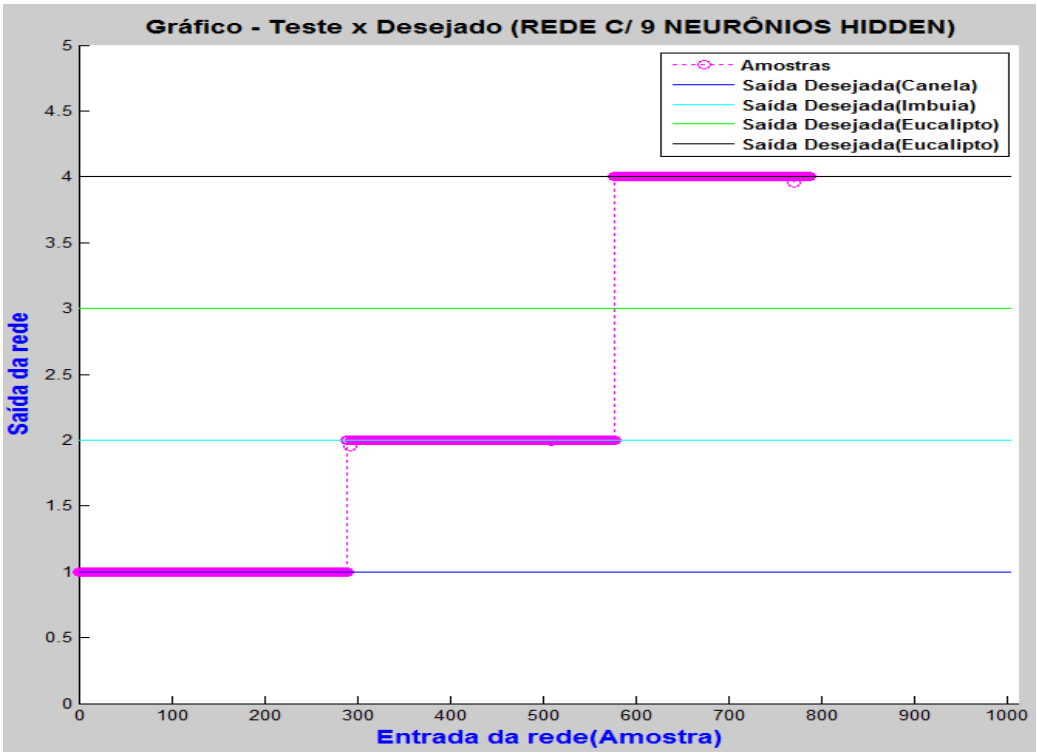


FIGURA 53: DESEMPENHO PARA O CONJUNTO DE TESTE CIE (ERRO DE 5,94E-06).
FONTE: AUTOR

Para uma tolerância de 2% o aproveitamento foi de 100%. A TABELA 28 ilustra a matriz de confusão para este caso.

TABELA 28: MATRIZ DE CONFUSÃO PARA O CONJUNTO CIE.

Espécies	Canela	Imbuia	Eucalipto	Não Classificado	Total de Amostras
Canela	288	0	0	0	288
Imbuia	0	288	0	0	288
Eucalipto	0	0	211	0	211

FONTE: AUTOR

- **ISE**

A TABELA 29 e a FIGURA 54 mostram o erro médio (MSE) por neurônios na camada escondida obtido para o conjunto ISE (Imbuia, Sassafrás e Eucalipto). O menor erro quadrático médio para o teste ocorreu no quarto ensaio, para treze neurônios na camada escondida, com os valores de 2,05E-10 e 1,80E-06 nas etapas de treinamento e teste, respectivamente. As Figuras 55 e 56 ilustram a saída da rede para o melhor caso.

TABELA 29: VALORES MÉDIOS DOS ERROS RELATIVOS AOS CINCO ENSAIOS, OBTIDOS NAS ETAPAS DE TREINAMENTO E TESTE PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Erros Médios		
Nº de Neurônios	**Treino**	**Teste**
1	0,715360093	0,569749
2	0,001105197	0,005518
3	0,000513398	0,003701
4	0,000260719	0,00301
5	0,001758749	0,00545
6	0,000367204	0,043449
7	2,21036E-06	0,002904
8	0,000187631	0,022645
9	8,05776E-06	0,001844
10	5,48383E-05	0,00235
11	0,000361444	0,00533
12	2,18318E-09	0,000123
13	0,000304043	0,0024
14	0,00036707	0,007135
15	0,000744354	0,006548
Máximo	1,304541034	1,23817
Mínimo	2,18318E-09	0,000123

FONTE: AUTOR

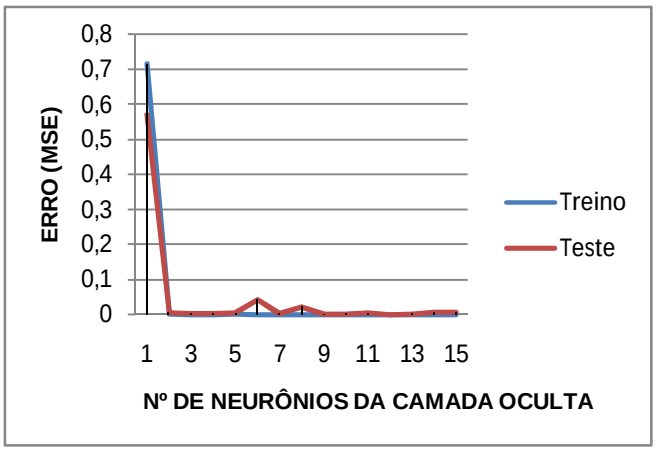


FIGURA 54: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE, PARA O CONJUNTO ISE.
FONTE: AUTOR.

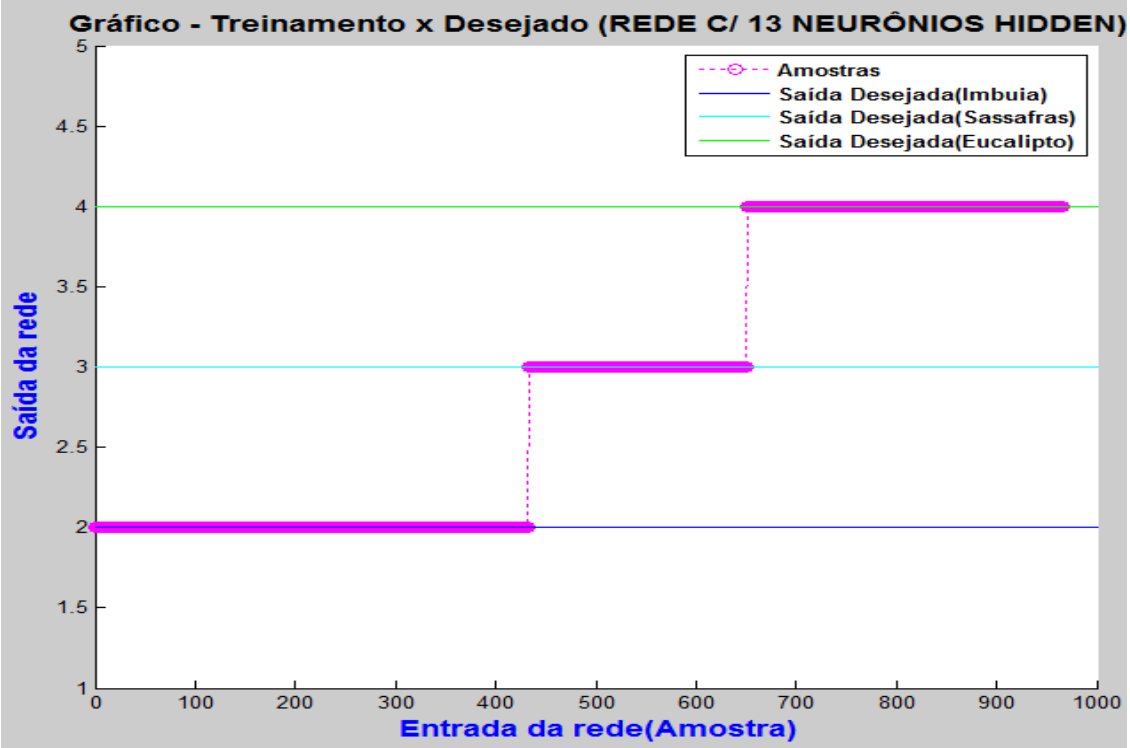


FIGURA 55: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO ISE (ERRO DE 2,05E-10).
FONTE: AUTOR

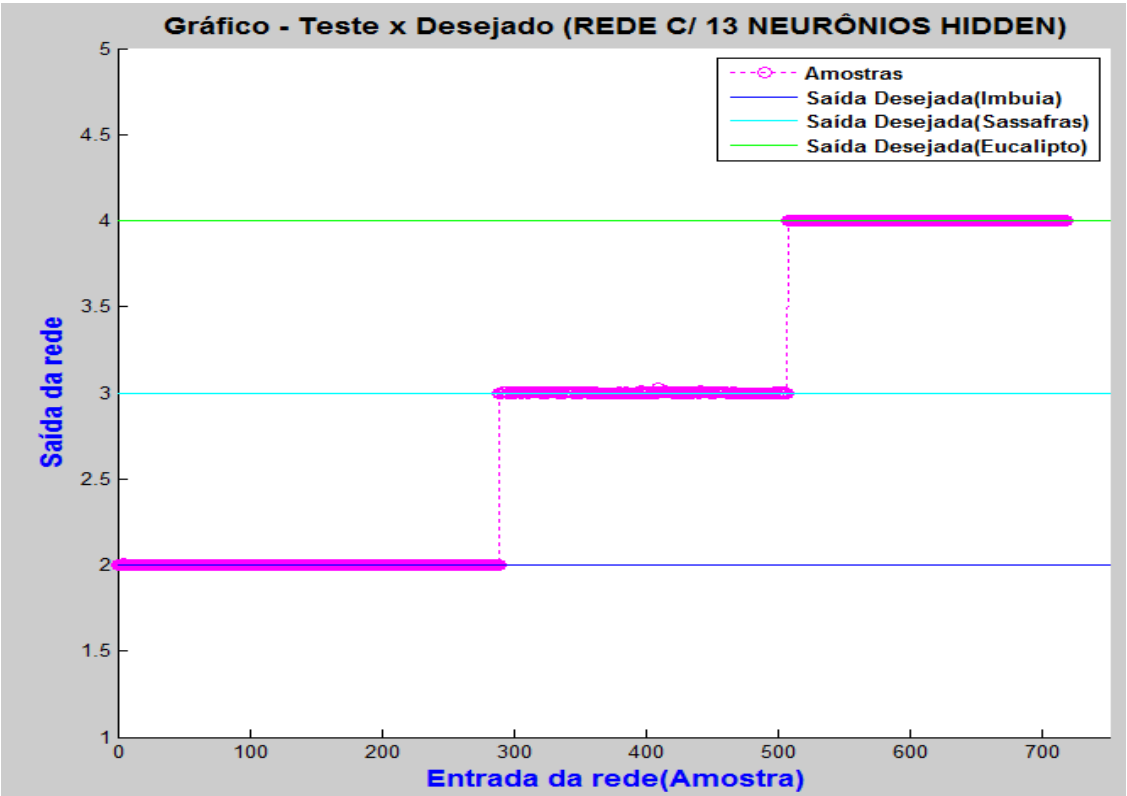


FIGURA 56: DESEMPENHO PARA O CONJUNTO DE TESTE ISE (ERRO DE 1,80E-06).
FONTE: AUTOR

Para uma tolerância de 2% o aproveitamento foi de 100%. A TABELA 30 ilustra a matriz de confusão para este caso.

TABELA 30: MATRIZ DE CONFUSÃO PARA O CONJUNTO ISE.

Espécies	Imbuia	Sassafrás	Eucalipto	Não Classificado	Total de Amostras
Imbuia	288	0	0	0	288
Sassafrás	0	218	0	0	288
Eucalipto	0	0	211	0	211

FONTE: AUTOR

• CSE

A TABELA 31 mostra o erro médio (MSE) por neurônios na camada escondida para o conjunto CSE (Canela, Sassafrás e Eucalipto), a FIGURA 57 expressa estas informações no formato gráfico. O menor erro quadrático médio para o teste ocorreu no oitavo ensaio, para quize neurônios na camada escondida, com os valores de 1,16E-08 e 1,97E-07 nas etapas de treinamento e teste, respectivamente. As Figuras 58 e 59 ilustram a saída da rede para o melhor caso.

TABELA 31: VALORES MÉDIOS DOS ERROS RELATIVOS AOS CINCO ENSAIOS, OBTIDOS NAS ETAPAS DE TREINAMENTO E TESTE PARA UMA DETERMINADA QUANTIDADE DE NEURÔNIOS NA CAMADA INTERMEDIÁRIA.

Erros Médios		
Nº de Neurônios	**Treino**	**Teste**
1	2,84E-03	2,50E-02
2	1,89E-03	1,05E-02
3	7,22E-09	1,52E-03
4	2,17E-03	1,32E-02
5	9,23E-04	3,06E-03
6	7,70E-04	6,25E-03
7	1,50E-03	1,10E-02
8	2,64E-03	1,61E-02
9	1,50E-03	7,33E-03
10	7,30E-04	9,66E-03
11	3,08E-03	1,60E-02
12	2,10E-03	7,91E-03
13	3,35E-08	1,93E-03
14	2,50E-03	1,29E-02
15	2,89E-03	1,83E-02
Máximo	3,08E-03	2,50E-02
Mínimo	7,22E-09	1,52E-03

FONTE: AUTOR.

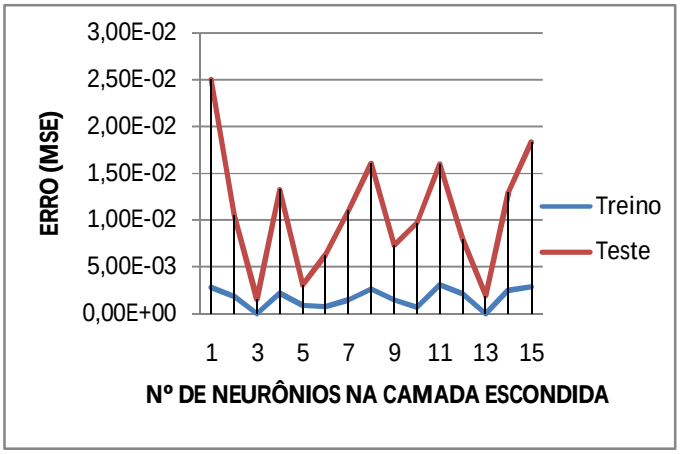


FIGURA 57: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE, PARA O CONJUNTO CSE.
FONTE: AUTOR.

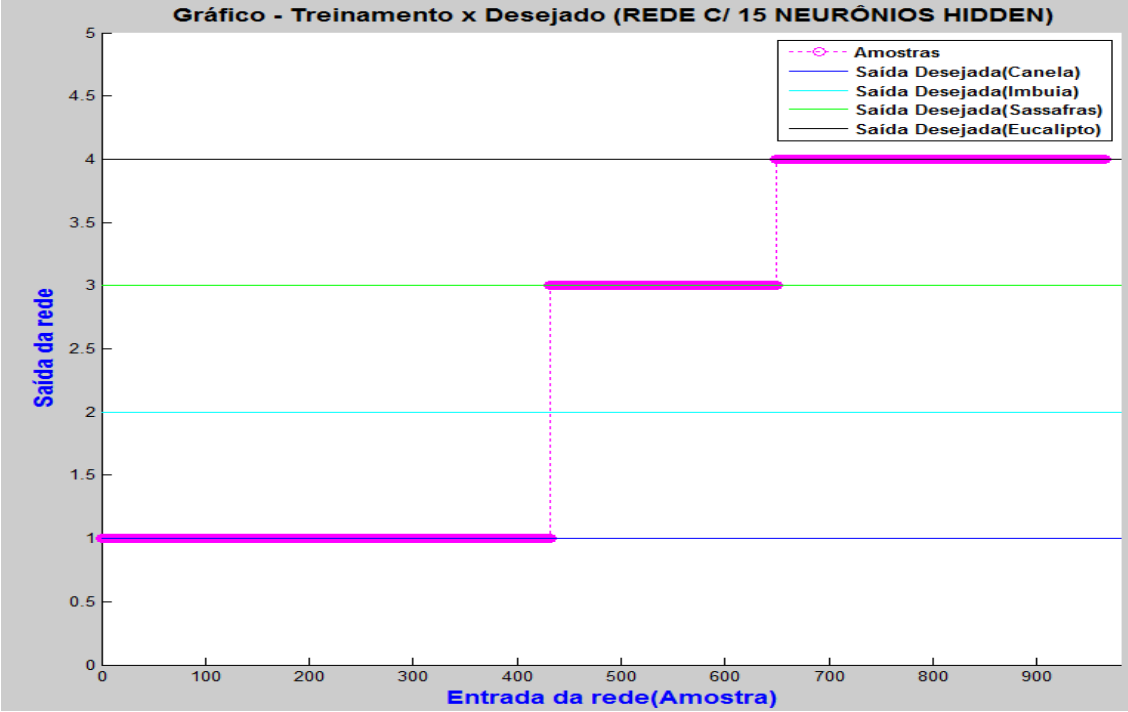


FIGURA 58: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO CSE (ERRO DE 1,16E-08).
FONTE: AUTOR

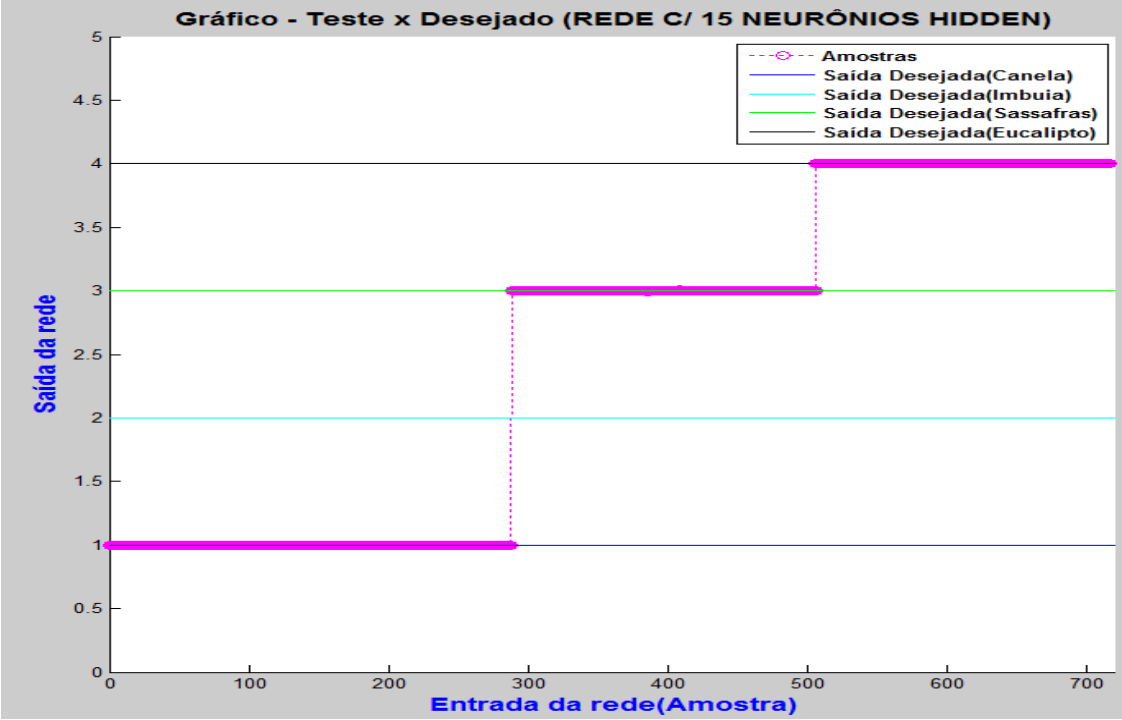


FIGURA 59: DESEMPENHO PARA O CONJUNTO DE TESTE CSE (ERRO DE 1,97E-07).
FONTE: AUTOR

Para uma tolerância de 2% o aproveitamento foi de 100%. A TABELA 32 ilustra a matriz de confusão para este caso.

TABELA 32: MATRIZ DE CONFUSÃO PARA O CONJUNTO CSE.

Espécies	Canela	Sassafrás	Eucalipto	Não Classificado	Total de Amostras
Canela	288	0	0	0	288
Sassafrás	0	218	0	0	288
Eucalipto	0	0	211	0	211

FONTE: AUTOR

Analisando os resultados da RNA proposta, para as combinações CIS, CIE, ISE e CSE, pode-se notar que a rede apresentou estabilidade, como foi ilustrado pelos gráficos de treinamento e teste, e ótimo desempenho, como mostraram as matrizes de confusão. Observou-se que não houve distorção a ponto de caracterizar uma influência de determinada amostra (espécie) sobre a outra. Percebeu-se, também, que não houve uma relação proporcional entre o número de neurônios na camada escondida e a redução do erro. Um exemplo disso pode ser visto na FIGURA 57.

Para ilustrar o comportamento da RNA ao tentar identificar uma amostra para a qual ela não foi treinada, tomou-se o conjunto de espécies formado por Canela, Imbuia e Sassafrás para treinamento. Na etapa de testes, além de exemplares análogos aos utilizados no treinamento, foram utilizadas amostras de Eucalipto. A FIGURA 60 ilustra o desempenho da rede na etapa de teste, as amostras reconhecidas com sucesso são representadas na cor magenta e os exemplares de Eucalipto na cor vermelha. Como eles não foram reconhecidos, mantiveram-se no canto inferior esquerdo do gráfico.

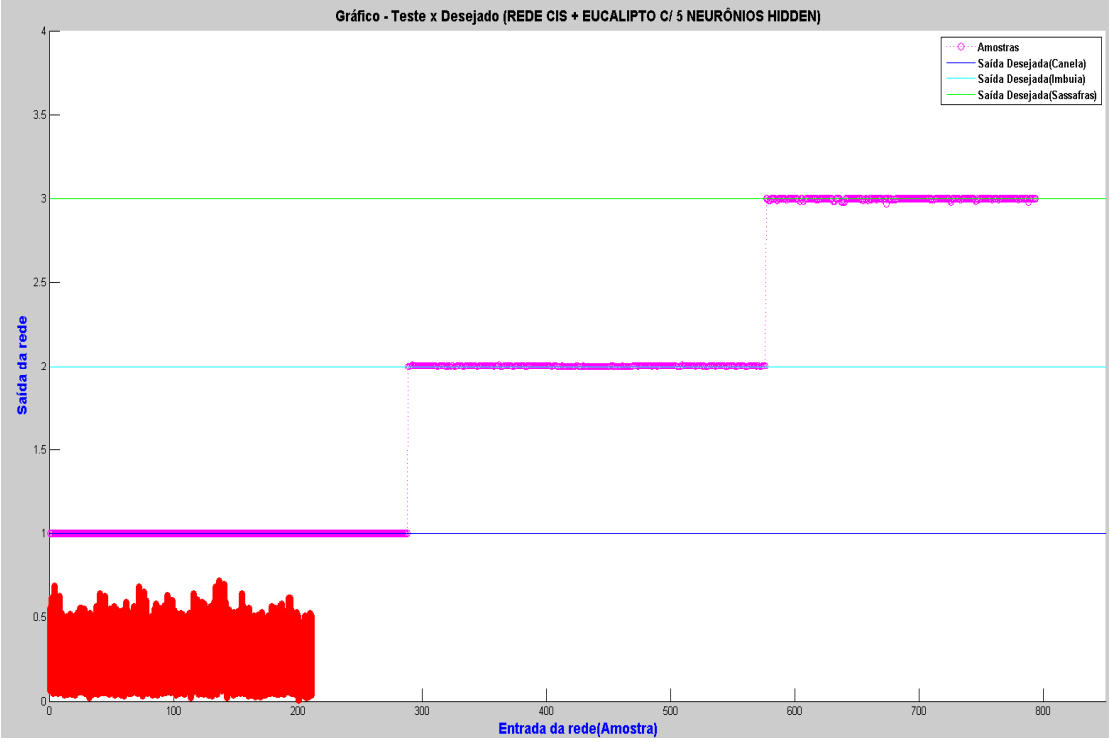


FIGURA 60: REPRESENTAÇÃO DA SAÍDA DA RNA PARA UMA AMOSTRA NÃO TREINADA (CIS + EUCALIPTO COM CINCO NEURÔNIOS NA CAMADA ESCONDIDA).
FONTE: AUTOR.

5.3.4 Análises sobre partições de números de onda

Segundo autores como Jaconi (2011), Stuart (2004) e Theophanides (2012), algumas ligações químicas apresentam comportamento vibracional distinto em determinadas faixas de comprimento de onda. Isso deve-se ao fato de que as ligações químicas, especialmente as do tipo covalente[24], possuem frequências vibracionais específicas, se uma molécula receber radiação eletromagnética com a mesma energia vibracional ocorrerá uma ressonância e a luz será absorvida. Esse processo constitui a base da espectrometria, na seção 3.5.1 há maiores informações sobre esse processo. A FIGURA 61 ilustra a relação entre faixas de comprimento de onda e o tipo de vibração de alguns grupos funcionais.

[24] A ligação covalente é um tipo de ligação química caracterizada pelo compartilhamento de um ou mais pares de elétrons entre átomos, causando uma atração mútua entre eles, que mantêm a molécula resultante unida. (STUART, 2004)

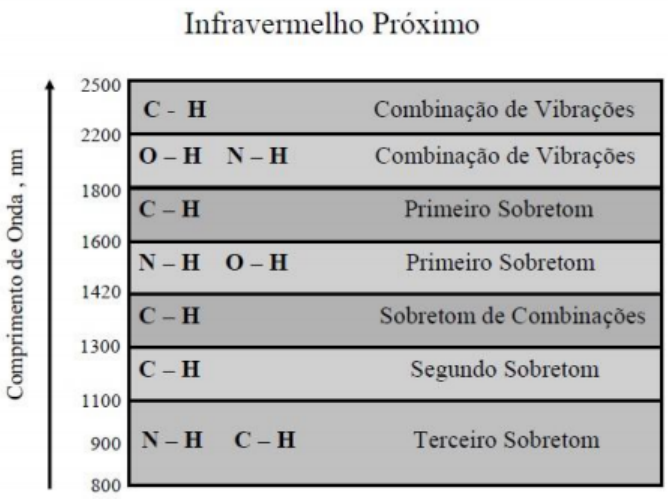


FIGURA 61: REGIÃO DE DETECÇÃO DE GRUPOS FUNCIONAIS EM NIRS.
FONTE: (JACONI, 2011)

Com o intuito de observar se havia alguma faixa de *números de onda* que pudesse ter maior relevância no processo decisório da rede neural proposta, dividiu-se o intervalo inicial, que variava de 9995 - 3996 (cm^{-1}) com resolução de 4 (cm^{-1}), em quatro partes. Como o processo de treinamento da rede neural é demorado, optou-se por não tomar uma quantidade grande de intervalos escolhendo-se quatro faixas com a mesma quantidade de *números de onda*. Foram realizados dez ensaios para cada parte. A TABELA 33 ilustra os intervalos escolhidos. As quantidades de amostras para treinamento e teste permaneceram as mesmas apresentadas na TABELA 6.

TABELA 33: FAIXAS DE NÚMEROS DE ONDA

Parte	Faixa de números de onda
Total	9995 - 3996 cm^{-1}
Parte 1	9995 -8498 cm^{-1}
Parte 2	8494 - 6997 cm^{-1}
Parte 3	6993 - 5497 cm^{-1}
Parte 4	5493 - 3996 cm^{-1}

FONTE: AUTOR.

• PRIMEIRA PARTE (9995 - 8498 cm^{-1})

A TABELA 34 mostra os erros médios quadráticos da rede para dez ensaios. A FIGURA 62 ilustra essas informações no formato gráfico.

TABELA 34: VALORES MÉDIOS DOS ERROS RELATIVOS AOS DEZ ENSAIOS (1ª PARTE).

	Erros Médios	
Nº de Neurônios	**Treino**	**Teste**
1	0,024563	0,151833
2	0,013986	0,137034
3	0,014626	0,142739
4	0,012988	0,137336
5	0,01598	0,160468
6	0,014022	0,155229
7	0,017352	0,161497
8	0,014613	0,176336
9	0,013816	0,173728
10	0,018735	0,182803
11	0,013957	0,159845
12	0,013331	0,152296
13	0,010328	0,151188
14	0,01168	0,178962
15	0,012869	0,175849

FONTE: AUTOR

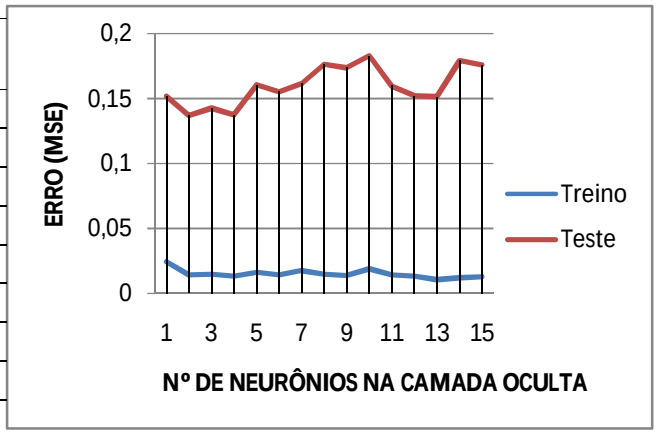


FIGURA 62: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE PARA A 1ª PARTE.
FONTE: AUTOR.

Observa-se que o erro na etapa de teste foi alto, FIGURA 62, representando um nível de reconhecimento pouco eficiente. Pode-se levar em consideração as similaridades químicas existentes entre as amostras em determinados intervalos de número de onda, dificultando a distinção realizada pela rede e aumentando o erro (STUART, 2004 ; THEOPHANIDES, 2012).

Considerando a FIGURA 62 foi possível observar que houve um grande distanciamento entre os erros obtidos nas etapas de treinamento e teste, demonstrando uma perda de generalização por parte da rede. As Figuras 63 e 64 representam a saída da rede nas etapas de treinamento e teste, respectivamente.

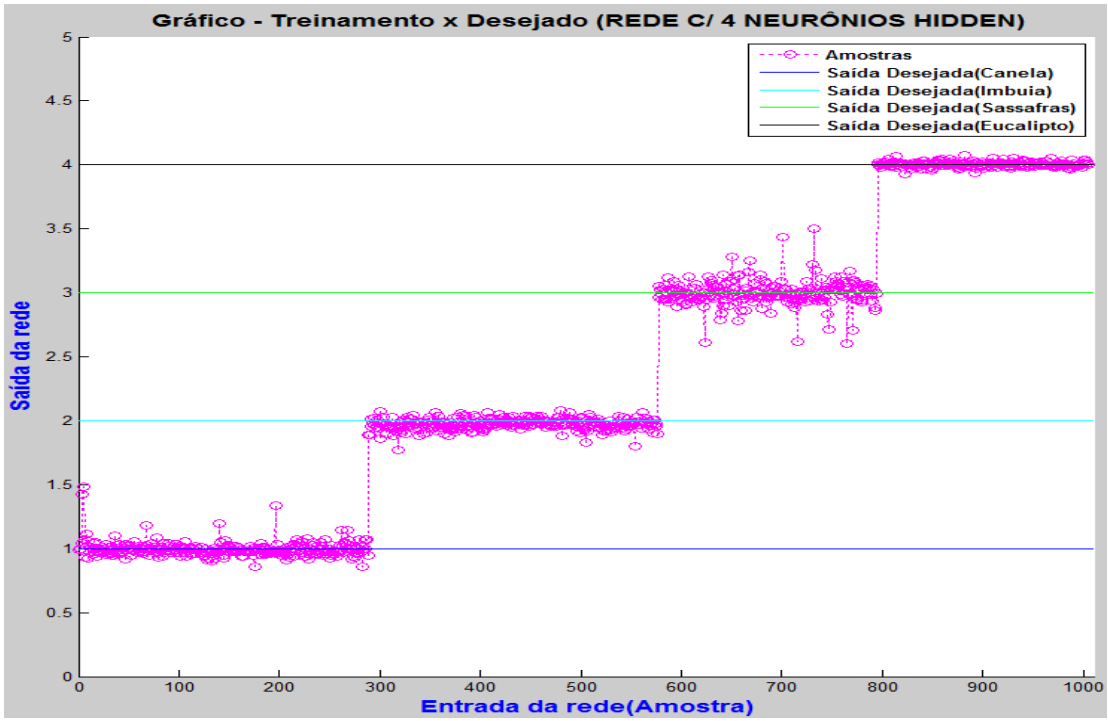


FIGURA 63: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO (1ª PARTE), OBTEVE-SE UM ERRO DE 4,15E-03.
FONTE: AUTOR

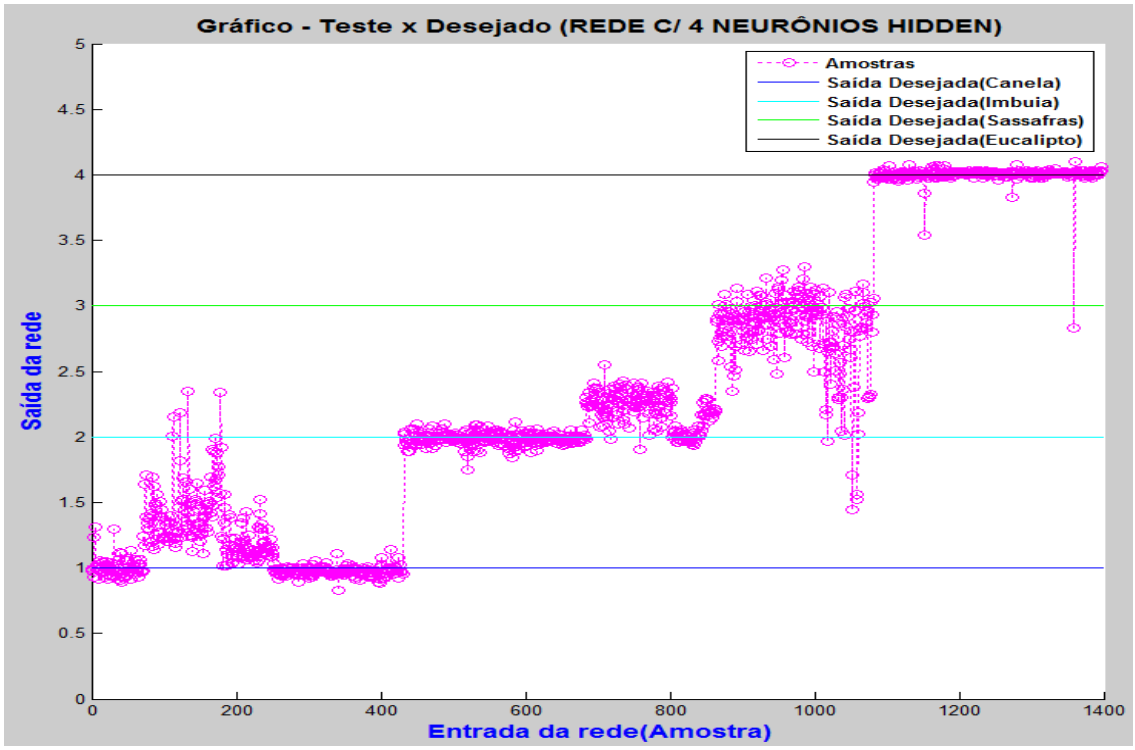


FIGURA 64: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO (1ª PARTE), OBTEVE-SE UM ERRO DE 5,23E-02.
FONTE: AUTOR.

• SEGUNDA PARTE (8494 - 6997 cm^{-1})

A TABELA 35 mostra os erros médios quadráticos da rede para dez ensaios. A FIGURA 65 ilustra essas informações no formato gráfico.

TABELA 35: VALORES MÉDIOS DOS ERROS RELATIVOS AOS DEZ ENSAIOS (2ª PARTE).

Erros Médios		
Nº de Neurônios	**Treino**	**Teste**
1	0,019581	0,106087
2	0,011151	0,118689
3	0,007586	0,121284
4	0,007662	0,129364
5	0,007701	0,12813
6	0,006867	0,135073
7	0,007924	0,136138
8	0,004936	0,125407
9	0,007745	0,141109
10	0,006439	0,12192
11	0,005938	0,131428
12	0,019766	0,15732
13	0,007932	0,144239
14	0,006598	0,134751
15	0,007633	0,128789

FONTE: AUTOR

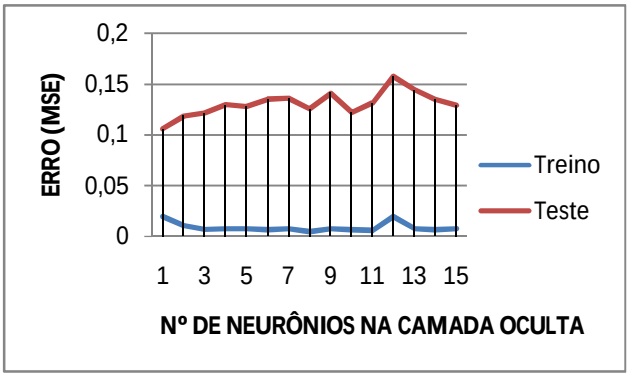


FIGURA 65: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE PARA A 2ª PARTE.
FONTE: AUTOR.

Notou-se que as saídas da rede para os conjuntos de treinamento e teste, Figuras 68 e 69, apresentaram maior estabilidade quando comparado com os resultados da primeira parte. Também houve uma redução dos valores médios de erro entre as etapas de treinamento e teste, FIGURA 65.

É plausível considerar que a rede conseguiu obter, nesse intervalo de número de onda, informações com maior distinção, possibilitando uma melhora na sua habilidade de generalização.

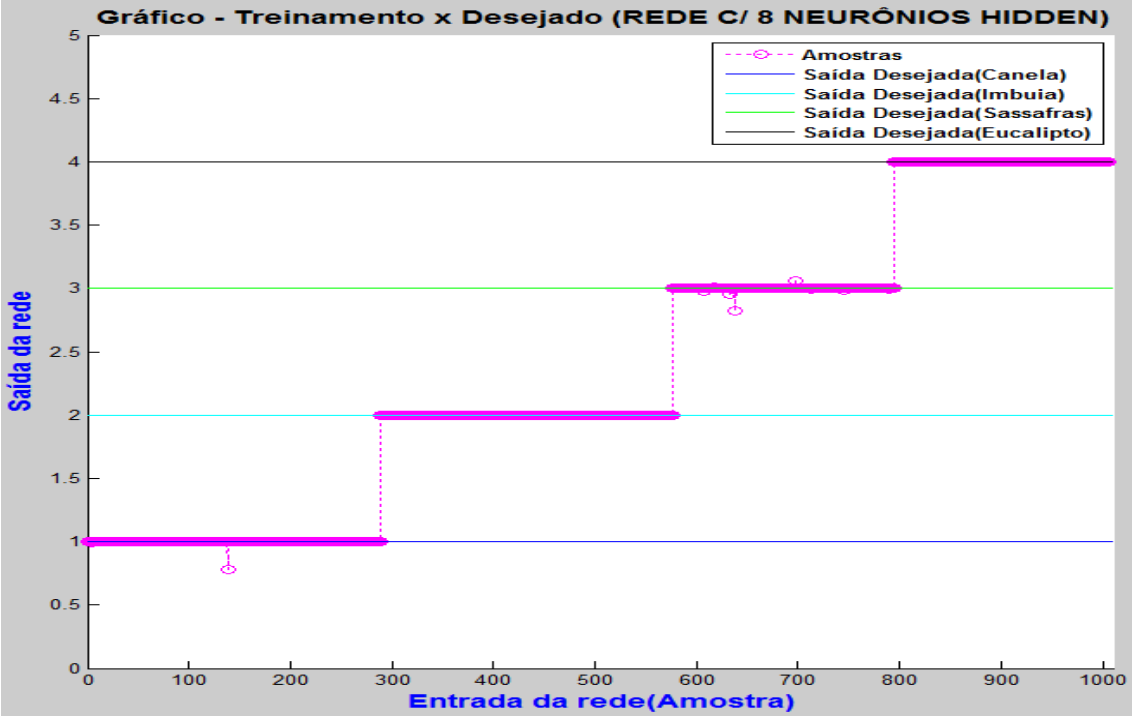


FIGURA 66: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO (2ª PARTE), OBTEVE-SE UM ERRO DE 8,77E-05.
FONTE: AUTOR.

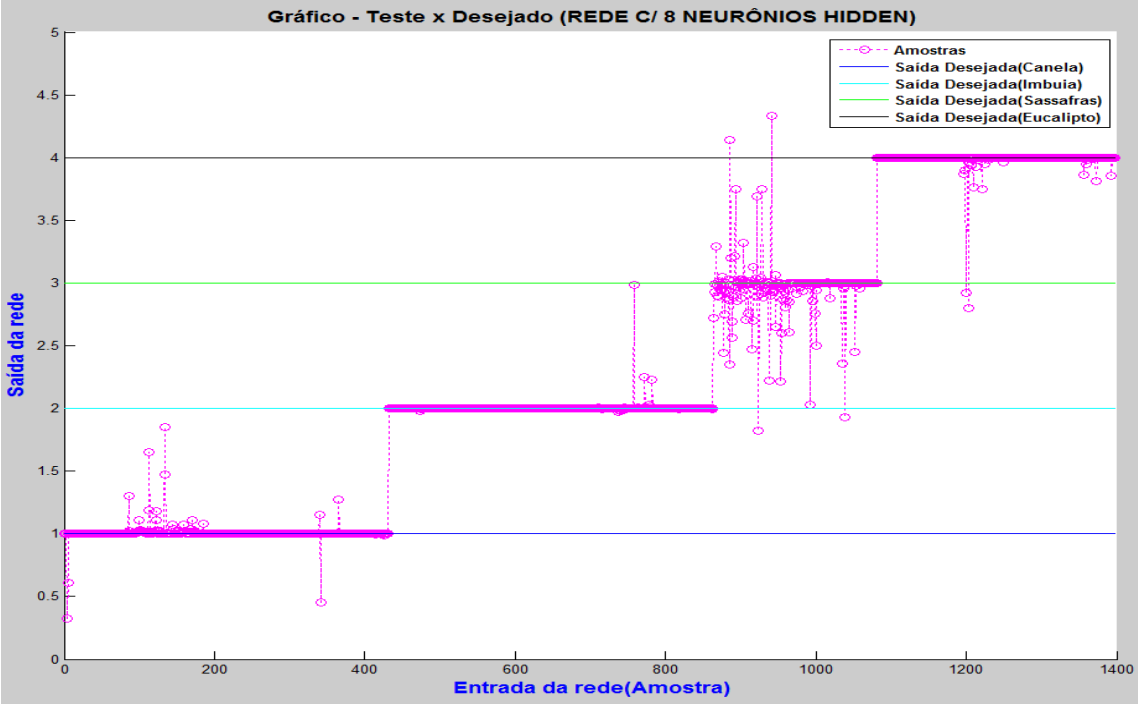


FIGURA 67: DESEMPENHO PARA O CONJUNTO DE TESTE (2ª PARTE), OBTEVE-SE UM ERRO DE 1,41E-02.
FONTE: AUTOR.

• TERCEIRA PARTE (6993 - 5497 cm^{-1})

A TABELA 36 mostra os erros médios quadráticos da rede para dez ensaios. A FIGURA 68 ilustra essas informações no formato gráfico.

TABELA 36: VALORES MÉDIOS DOS ERROS RELATIVOS AOS DEZ ENSAIOS (3ª PARTE).

Erros Médios		
Nº de Neurônios	**Treino**	**Teste**
1	0,012122	0,05039
2	0,005767	0,028593
3	0,003943	0,022995
4	0,003457	0,031595
5	0,003593	0,031815
6	0,002998	0,026548
7	0,002207	0,027549
8	0,003839	0,026971
9	0,002695	0,031808
10	0,003514	0,025372
11	0,002496	0,035279
12	0,002449	0,030997
13	0,002388	0,03093
14	0,003615	0,030612
15	0,003429	0,038258

FONTE: AUTOR.

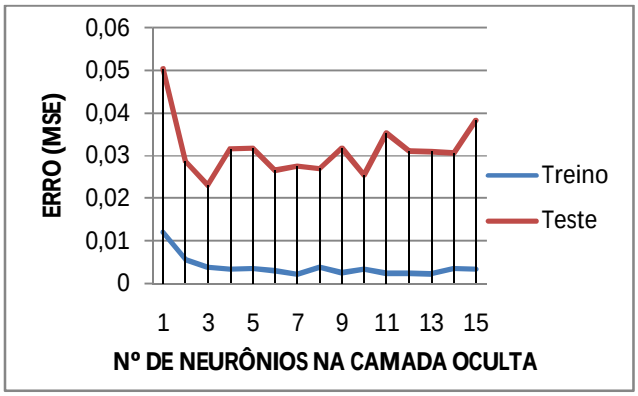


FIGURA 68: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE PARA A 3ª PARTE.
FONTE: AUTOR.

Em comparação com as duas primeiras faixas analisadas, percebe-se que a terceira parte atingiu melhor resultado, com um desvio significativamente menor entre os erros de treinamento e teste, conforme ilustra a FIGURA 68. Não houve variações significativas para os dados de treinamento, FIGURA 69, embora apenas a variedade Sassafrás tenha apresentado distorções significativas na etapa de teste, conforme mostra a FIGURA 70.

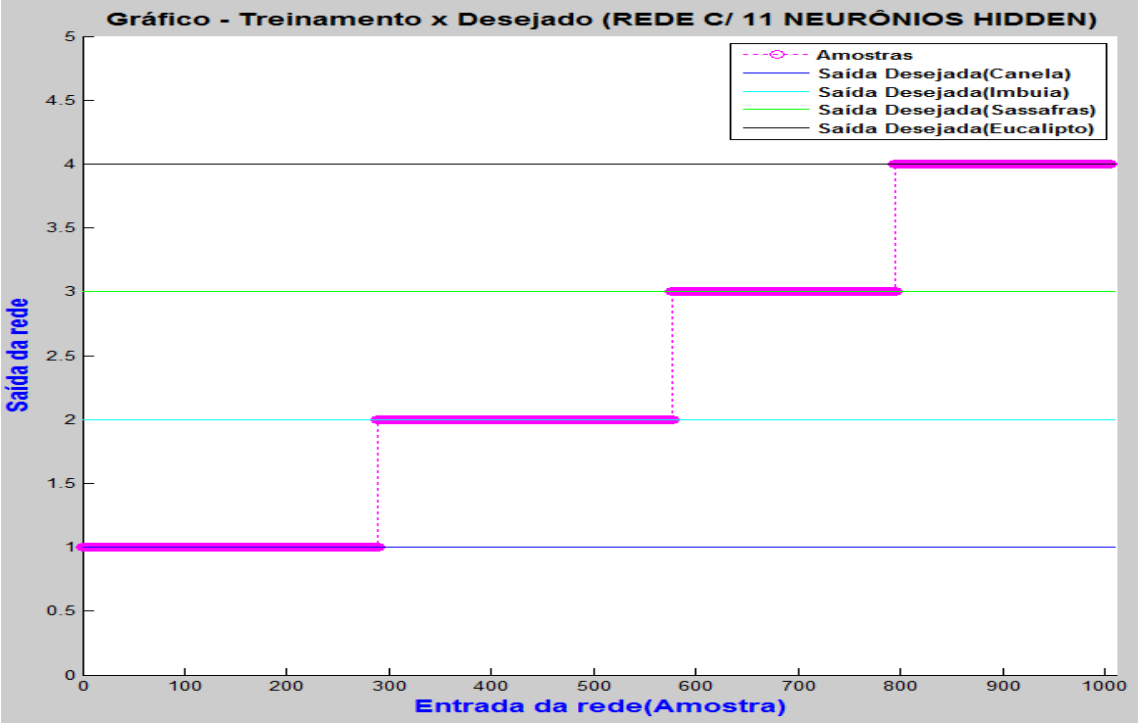


FIGURA 69: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO (3ª PARTE), OBTEVE-SE UM ERRO DE 8,39E-09.
FONTE: AUTOR.

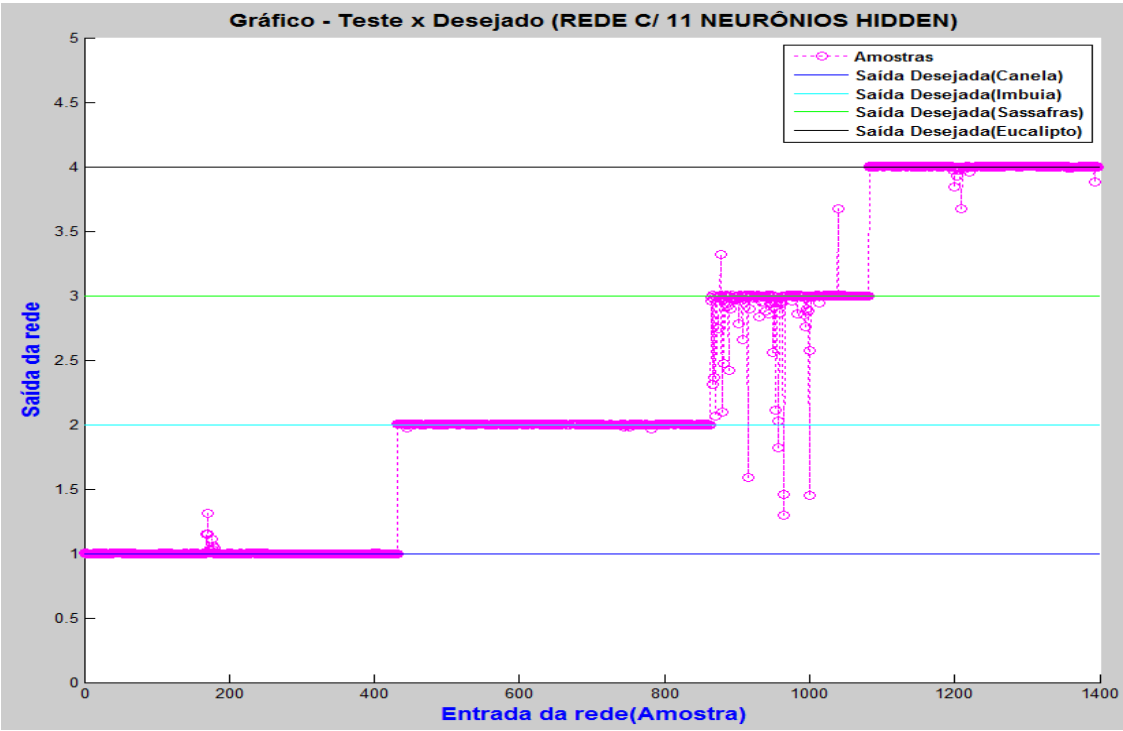


FIGURA 70: DESEMPENHO PARA O CONJUNTO DE TESTE (3ª PARTE), OBTEVE-SE UM ERRO DE 1,26E-02.
FONTE: AUTOR.

• QUARTA PARTE (5493 - 3996 cm^{-1})

A TABELA 35 mostra os erros médios quadráticos da rede para dez ensaios. A FIGURA 71 ilustra essas informações no formato gráfico.

TABELA 37: VALORES MÉDIOS DOS ERROS RELATIVOS AOS DEZ ENSAIOS (4ª PARTE).

Erros Médios		
Nº de Neurônios	**Treino**	**Teste**
1	0,006142	0,035949
2	0,002777	0,033458
3	0,017355	0,053618
4	0,002207	0,021574
5	0,002481	0,056796
6	0,002078	0,020283
7	0,00266	0,026503
8	0,001748	0,040319
9	0,001081	0,026085
10	0,001006	0,025923
11	0,002124	0,027313
12	0,001461	0,02425
13	0,001515	0,022379
14	0,001857	0,021045
15	0,001484	0,022537

FONTE: AUTOR.

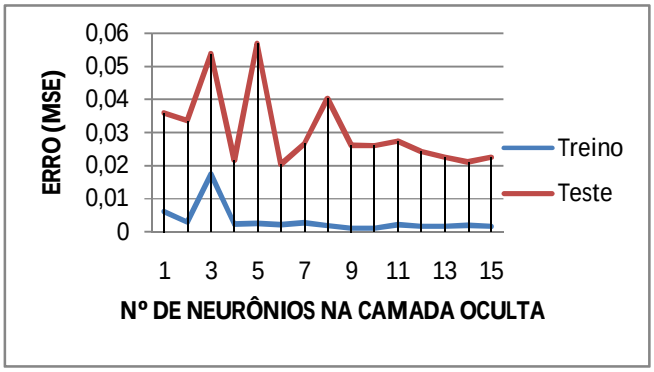


FIGURA 71: REPRESENTAÇÃO GRÁFICA DOS ERROS MÉDIOS NAS ETAPAS DE TREINAMENTO E TESTE PARA A 4ª PARTE.
FONTE: AUTOR.

Pela TABELA 37, representada graficamente pela FIGURA 71, observa-se que o erro é maior para uma quantidade menor de neurônios na camada escondida e tem uma redução de forma mais estável a partir de 9 neurônios na camada oculta. As Figuras 72 e 73 representam a saída da rede para os conjuntos de treinamento e teste, respectivamente. Nota-se que houve pequenas distorções, na etapa de teste, para as amostras de Sassafrás e Eucalipto, no entanto, a Canela e a Imbuia não apresentaram desvios significativos.

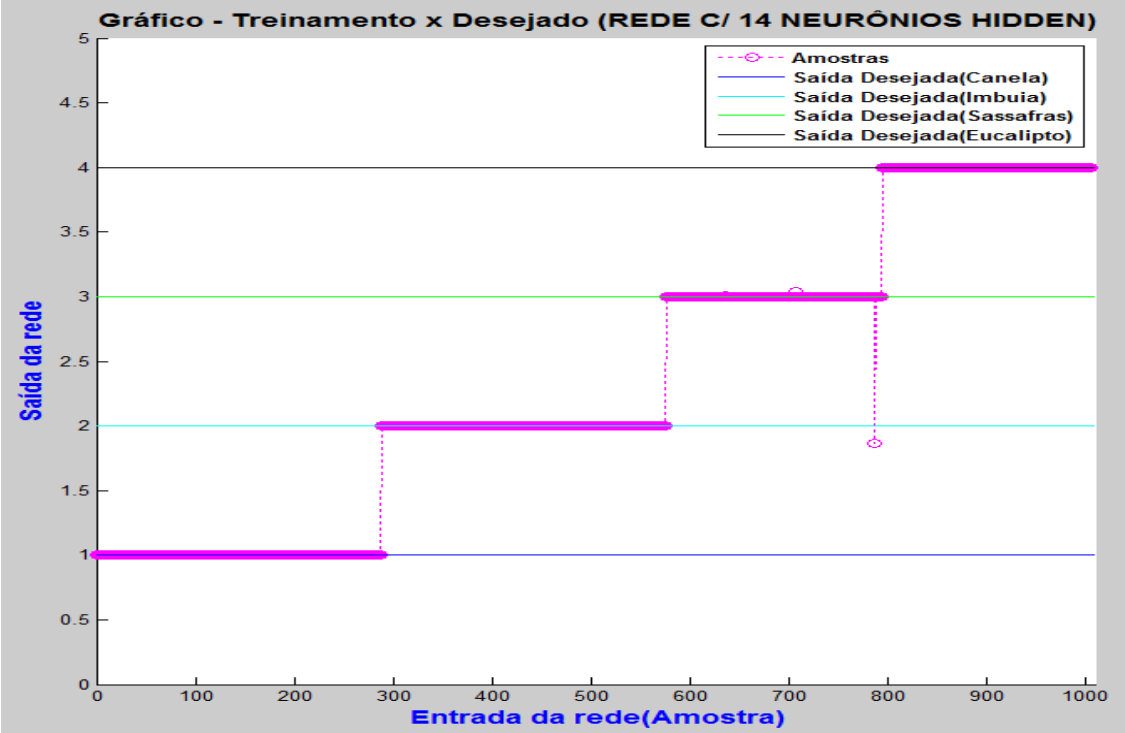


FIGURA 72 : DESEMPENHO PARA O CONJUNTO DE TREINAMENTO (4ª PARTE), OBTEVE-SE UM ERRO DE 1,28E-03.
FONTE: AUTOR.

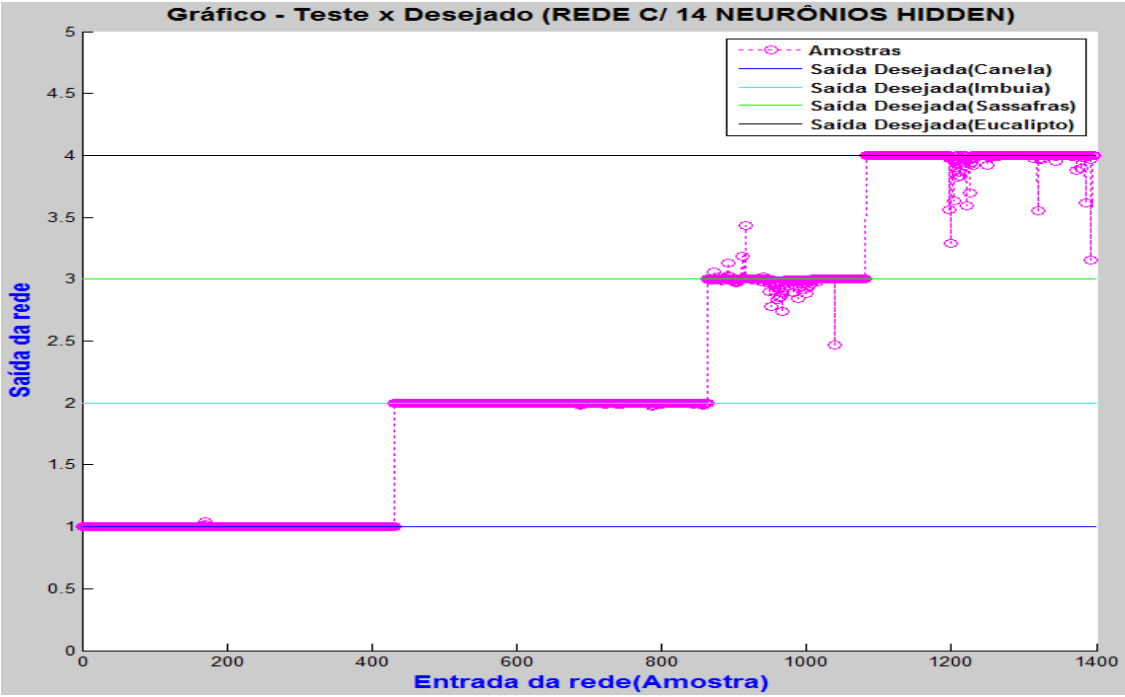


FIGURA 73: DESEMPENHO PARA O CONJUNTO DE TESTE (4ª PARTE), OBTEVE-SE UM ERRO DE 2,27E-03.
FONTE: AUTOR.

A TABELA 38 mostra qual foi o valor mínimo do erro médio quadrático, obtido pela rede nas quatro partes analisadas, durante as etapas de testes. Pode-se perceber pela FIGURA 74 que houve um decréscimo significativo no erro entre a primeira e segunda partes, sendo este menos acentuado entre a segunda e terceira, por fim, tendo uma queda um pouco maior entre a terceira e quarta partes.

TABELA 38: ERRO MÉDIO MÍNIMO OBTIDO NA ETAPA DE TESTE PARA AS QUATRO PARTES ANALISADAS.

Parte	Menor Erro Médio
P1	5,23E-02
P2	1,41E-02
P3	1,27E-02
P4	2,27E-03

FONTE: AUTOR

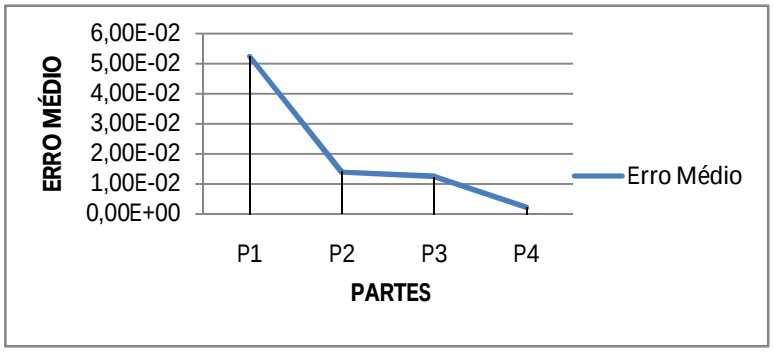


FIGURA 74: REPRESENTAÇÃO GRÁFICA DA (TABELA 35).
FONTE: AUTOR

É possível considerar que as três últimas partes, faixas de número de onda variando entre 8494 - 3996 cm^{-1}, possuem uma influência mais significativa na capacidade de reconhecimento da rede.

Para comprovar a eficiência das três últimas partes, foi feito um treinamento considerando um conjunto formado por P2, P3 e P4. As Figuras 75 e 76 ilustram a saída da rede nas etapas de treinamento e teste para esta situação. Comparando-se as Figuras 41 e 76, ambas ilustrando a saída da rede na etapa de teste para as quatro espécies, percebe-se certa similaridade embora uma tenha obtido erro de 3,38E-5 e a outra 1,09E-03. A diferença mais significativa entre ambas pode ser observada na saída relativa às amostras de Canela-Sassafrás.

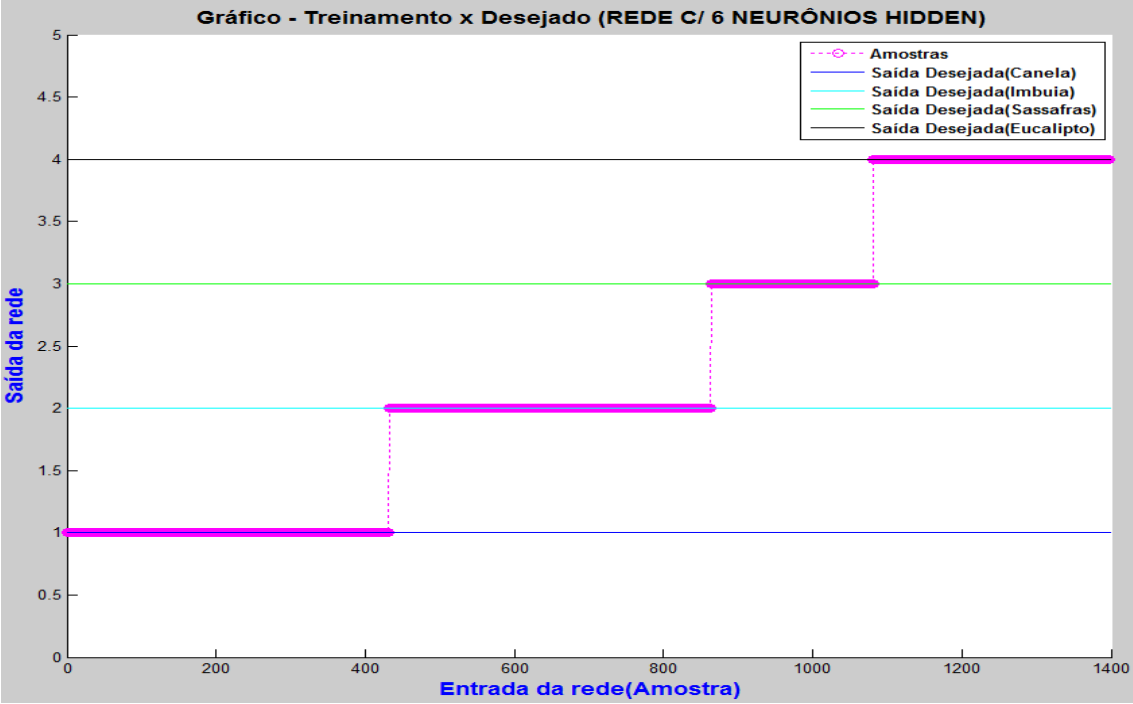


FIGURA 75: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO, OBTEVE-SE UM ERRO DE 1,11E-08.
FONTE: AUTOR.

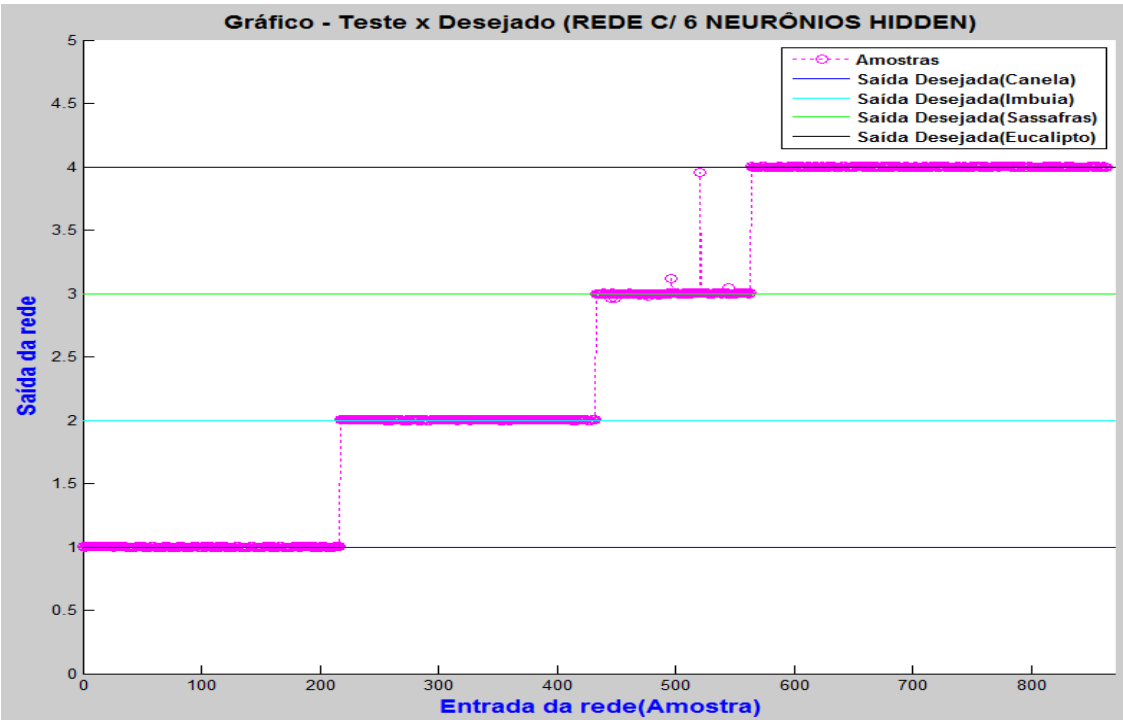


FIGURA 76: DESEMPENHO PARA O CONJUNTO DE TREINAMENTO, OBTEVE-SE UM ERRO DE 1,09E-03.
FONTE: AUTOR.

CAPÍTULO VI - CONSIDERAÇÕES FINAIS

6.1 CONCLUSÕES

Este trabalho propôs, em seus objetivos, mostrar a eficiência de uma rede neural do tipo *Multi-layer Perceptron* (MLP) com aprendizagem do tipo *Backpropagation* com a heurística de *Levemberg-Marquardt* como ferramenta de identificação de algumas espécies de árvores mediante a análise dos espectros de suas madeiras. Os espectros analisados não passaram por nenhum filtro estatístico (correção de linha base, suavização, normalização, derivação, etc.), sendo examinados em sua forma "bruta". Para observar o desempenho da rede proposta, não houve separação dos espectros quanto aos planos de corte (longitudinal radial, longitudinal tangencial, transversal) para uma mesma amostra, todos os dados foram analisados conjuntamente, ou seja, a rede esteve sujeita a diferentes tipos de leituras espectrométricas e mesmo assim conseguiu fazer uma generalização satisfatória, pois houve poucos desvios, como se pôde observar nos resultado obtidos.

Observou-se que função de ativação tangente hiperbólica-sigmoidal propiciou melhores resultados em relação à função logarítmica-sigmoidal. É importante frisar que mesmo para uma grande quantidade de padrões de entradas, os resultados foram satisfatórios com poucos neurônios na camada escondida. Assim, quando analisadas as quatro espécies (Canela, Imbuia, Sassafrás e Eucalipto), com função de transferência tan-sigmoidal e oito neurônios na camada oculta, obteve-se uma taxa de reconhecimento de 99,90% para uma margem de erro de 2%.

Das amostras analisadas, as de Canela-Sassafrás foram as que apresentaram maior desvio. É plausível considerar que esses desvios estejam relacionados à quantidade menor de amostras de Sassafrás, comparada às outras utilizadas para treinamento.

Ressalta-se, pelas combinações das três espécies (seção 4.3.3), que não houve interferências entre as amostras. Nessa etapa, para uma margem de erro de 2%, o reconhecimento foi de 100%, fortalecendo a ideia de independência entre as amostras.

Nota-se, pela seção 4.3.4, que houveram faixas de intervalos de número de onda que foram mais significativas para o reconhecimento da amostra.

O tempo de treinamento da rede aumentou de acordo com o número de neurônios na camada escondida, bem como pela quantidade de amostras utilizadas como entrada e a quantidade de variáveis presentes. Observou-se, no desenvolvimento deste trabalho, que houve um custo computacional considerável, visto que para a máquina utilizada (processador Intel I7 com 16 Gb de memória) houve uma demanda média de 91% de memória e 50% de processamento durante as etapas de treinamento com quatro espécies, simultaneamente.

Por fim, conclui-se que a utilização de RNA para identificação de madeira apresentou resultados promissores, visto ser flexível aos ruídos (falhas de leitura, sujeira nas amostras, erros no processo de corte, dentre outras) existentes e não ter exigido que os dados passem por prévio tratamento estatístico antes de serem utilizados. Embora o treinamento seja demorado e haja um custo computacional considerável, depois de treinada, a rede apresentou resultados de forma rápida e não exigiu esforço computacional significativo. De maneira análoga à utilizada neste trabalho, RNAs também podem ser aplicadas em conjunto a espectrofotômetros para análises de outros materiais.

6.2 TRABALHOS FUTUROS

Durante a realização deste estudo, verificaram-se algumas vertentes que podem possibilitar melhorias ao trabalho atual bem como investigações futuras e a elaboração de artigos científicos, tais como:

- Utilizar filtros estatísticos nos dados antes de utilizá-los para treinar a rede neural;
- Aplicar conceitos de Análise Multivariada para abstrair informação do conjunto de dados utilizados;
- Utilizar heurísticas diferentes, como uma rede RBF (*Radial Basis Function*) ou SVM (*Support Vector Machine*), para treinamento e simulação;
- Comparar o desempenho de uma rede neural com métodos tradicionalmente utilizados para a classificação de amostras mediante dados espectrométricos;
- Realizar estudos mais aprofundados quanto às faixas de números de onda com maior significância para a rede;
- Realizar teste com amostras de carvão;
- Realizar teste com outros compostos, orgânicos e inorgânicos;
- Ampliar a quantidade de amostras e determinar se há um limite máximo de espécies que possam ser treinadas em uma única RNA;
- Estudar uma possível utilização de um sistema baseado em Redes Neurais Artificiais como ferramenta para autoridades de fiscalização.

REFERÊNCIAS

ALBRECHT, R.; JOFFRE, R.; GROS, R.; PETIT, J. L.; TERROM, G.; PERISSOL, C. Efficiency of near-infrared reflectance spectroscopy to assess and predict the stage of transformation of organic matter in the composting process. **Bioresource Technology**, v. 99, p. 448-455, 2008. ISSN 0960-8524.

ALEXANDROV, O. **Comparison of gradient descent and Newton's method for optimization**. Disponível em: http://en.wikipedia.org/wiki/File:Newton_optimization _vs_grad_descent.svg#filelinks. Acesso em: 15/04/2013.

ARAGÃO, J. G.; CONCEIÇÃO, G. M. D. Myrtaceae: Espécies das subtribos Eugeniinae, Myrciinae e Myrtinae registradas para o estado do Maranhão. **Revista Sinapse Ambiental**, Minas Gerais, p. 7-17, Dezembro 2008.

ÁVILA, K. O.; RIBEIRO, M. L. M.; OLIVEIRA, I. H. S.; KNUTH, A. S.; GATTO, D. A. Importância da pesquisa sobre análise macroscópica para o engenheiro industrial madeireiro. **CIC 2001 - XX CONGRESSO DE INICIAÇÃO CIENTÍFICA UFPEL**, Pelotas, 2011.

BALDIN, M. S. Z. **Redes Neurais na orientação profissional**. 146 f. Dissertação (Mestrado em Métodos Numéricos). Universidade Federal do Paraná, 2002.

BISHOP, C. M. **Pattern Recognition and Machine Learning**. 2ª. ed. New York: Springer, 2006. cap. 4 e cap. 5.

BOÇOIS, A. **Diagnóstico de doenças dermatológicas usando a rede neural de Kohonen**. 98 f. Dissertação (Mestrado em Métodos Numéricos). Universidade Federal do Paraná, 2012.

BREMANANTH, R.; NITHYA, B.; SAIPRIYA, R. Wood Species Recognition System. **International Journal of Electrical and Computer Engineering**, p. 4 - 58, out. 2009.

BURDEN, R. L.; FAIRES, J. D. **Análise Numérica**. Tradução da 8ª Edição norte-americana. ed. São Paulo: CENGAGE, 2008. cap. 2 e cap. 10.

CARNEIRO, M. E. **Classificação de lâminas de madeira de Pinus spp por espetroscopia óptica.** 97 p. Dissertação (Mestrado em Eng. Florestal), Universidade Federal do Paraná, 2008.

CHEMWIKI. **Physical Chemistry**. Disponível em: <http://chemwiki.ucdavis.edu/Physical_Chemistry>. Acesso em: 18/04/2013

CIURCZAK, E. W.; JAMES, K. D. Pharmaceutical and Medical Applications of Near-Infrared Spectroscopy. **Pratical Spectroscopy Series**. v. 31. p. 45 - 48. 2005.

COSMO, N. L.; LIRA, P. K.; MORESCO, G. C.; SOFFIATTI, P.; SOUSA, T. R.; VASCONCELLOS, T. J.; LISI, C. S.; BOTOSSO, P. C. **Dendroecologia da espécie Ocotea porosa (Imbuia), Lauraceae, em áreas de Floresta Ombrófila Mista na região de Faxinal do Céu, Paraná**. 5º South American Dendrochronological Fieldweek. Faxinal do Céu - Pr, p. 8. 2009.

DANTAS, H. A. **Desenvolvimento de técnicas quimiométricas de compressão de dados e de redução de ruído instrumental aplicadas a óleo diesel e madeira de eucalipto usando espectroscopia NIR**. 158 f. Tese (Doutorado em Química). Universidade Federal de Pernanbuco. 2007.

DEMUTH, H.; BEALE, M.; HAGAN, M. **Neural Network Toolbox 6**. Natick, MA: The MathWorks, 2009.

DICAF, F. Portal da madeira. **Estrutura do tronco**, 2009. Disponível em: <http://portaldamadeira.blogspot.com.br/2009/10/lenho-xilologia.html>. Acesso em: 18/12/2012.

Espectroscopia de Infravermelho. In: **WIKIPÉDIA, a enciclopédia livre**. Flórida: Wikimedia Foundation, 2013. Disponível em: <http://pt.wikipedia.org/w/index.php?title=Espectroscopia_de_infravermelho&oldid=35513575>. Acesso em: 24 jul. 2013.

ESTEBAN, L. G.; FERNÁNDEZ, F. G.; PALACIOS, P. P.; ROMERO, R. M.; CANO, N. N. Artificial neural networks in wood identification: the case of two Juniperus species from the canary islands. **IAWA Journal**, Leiden, p. 87-94. 2009.

EXTRAÇÃO ilegal de madeira. **Portal de notícias G1**, Rio de Janeiro, 17 maio 2012. Disponível em:< http://g1.globo.com/mato-grosso/noticia/2012/05/prf-prende-5-homens-suspeitos-de-extrair-madeira-ilegalmente-em-mt.html > Acesso em: 18/12/12

FAUSETT, L. V. **Fundamentals of Neural Networks:** Architectures, Algorithms And Applications. 1ª Ed. ed. New York: Pearson. 1993. cap. 1 - 5.

FISCHETTI, M. Computers versus Brains. **SCIENTIFIC AMERICAN**, novembro 2011. Disponivel em: <http://www.scientificamerican.com/article.cfm?id=computers-vs-brains>. Acesso em: 20/11/2012.

FOELKEL, E. Curiosidades e singularidades acerca dos Eucaliptos. **Revista EUCALYPTUS**, Porto Alegre, n. 23, out. 2009.

FRANCA, M. S. J. Pau para toda obra. **Unesp ciência**, p. 10-13, 2012.

GAVIN, H. DUKE UNIVERSITY. **Experimental Systems**, 2011. Disponivel em: <http://people.duke.edu/~hpgavin/ce281/>. Acesso em: 16/02/2013.

HAGAN, M. T.; DEMUTH, H. B.; BEALE, M. **Neural Network Design**. Boston: Thomson Publishing INC., 1996. cap. 1-12.

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

HAYKIN, S. **Redes neurais: princípios e prática**. 2ª Ed. ed. Porto Alegre: Bookman, 2001. cap. 1-5.

HECHT, EUGENE. **Optics**. 4ª Ed. San Francisco. Addison-Wesley. 2001. cap.3-4.

HERVÁS, C.; GARRIDO, A.; LUCENA, B.; GARCÍA, N.; PEDRO, E. Near infrared spectroscopy for classification of Iberian pig carcasses using an artificial neural network. **Near Infrared Spectrosc**, Chichester- UK, v. 2, p. 177-184, janeiro 1996. ISSN 0967-0335.

HOPFIELD, J. J. Neurons with Graded Response Hava Collective Computational Properties like Those of Two-state Neurons. **Proceedings of the National Academy of Sciences**. Cambridge, v. 81, p. 3088-3092, mar. 1984

Instituto Brasileiro de Florestas (IBF). **Espécies Nativas Brasileiras**. Disponivel em: <http://www.ibflorestas.org.br/>. Acesso em: 10/01/2013.

Instituto de Pesquisa e Estudos Florestais (IPEF). **Identificação de Espécies Florestais (Ocotea porosa)**. Disponivel em: <http://www.ipef.br/identificacao /nativas/detalhes.asp?codigo=45>. Acesso em: 10/01/2013.

Instituto de Pesquisas Tecnológicas (IPT). **Curso do IPT em Mato Grosso colabora para dar suporte à fiscalização da atividade madeireira**. [s.l.], [s.d]. Disponivel em: <http://www.ipt.br/centros_tecnologicos/CT-FLORESTA/noticias/507-identificacao_de_madeiras.htm>. Acesso em: 18/12/2012.

Instituto de Pesquisas Tecnológicas (IPT). **Você sabia? Que o IPT foi pioneiro na identificação e aplicação das madeiras brasileiras?** [s.l.], [s.d]. Disponivel em: < http://www.ipt.br/institucional/campanhas/4voce_sabia_que_o_ipt_foi_pioneiro_na_identificacao_e_aplicacao_das_madeiras_brasileiras.htm>. Acesso em: 18/12/2012.

Instituto de Pesquisas Tecnológicas (IPT). **Identificação de madeiras.** [s.l.], [s.d]. Disponivel em: < http://www.ipt.br/centros_tecnologicos/CT-FLORESTA/noticias/507-identificacao_de_madeiras.htm>. Acesso em: 18/12/2012.

JACONI, A. **O uso da espectroscopia no infravermelho próximo na quantificação de carbono em solos sob o cultivo de cana-de-açucar**. Dissertação (Mestrado em Química). 115 f. Universidade de São Paulo. 2011.

JORDAN, R.; FEENEY, F.; NESBITT, N.; EVERTSEN, A. J.. Classification of wood species by neural network analysis of ultrasonic signals. **Journal Ultrasonics**, v. 36, p. 219-222, 1998. ISSN 0041-624X/98.

JORNAL AMBIENTE BRASIL. **Silvicultura do Eucalipto (Eucalyptus spp.)**. Disponivel em: <http://ambientes.ambientebrasil.com.br/florestal/silvicultura/ silvicultura_do_eucalipto_(eucalyptus_spp.).html>. Acesso em: 10/01/2013.

KHOURY JUNIOR, J. K.; Pinto, F. A. C.; Queiroz, D. M.; Lucia, R. M. D.; Resend, R. C. Redes neurais para reconhecimento de defeitos de madeira serrada de eucalipto em imagens digitais. **Scientia Forestalis**, Piracicaba, v. 70, p. 85-96, abril 2006.

LABATI, R. D.; GAMASSI, M.; PIURI, V.; SCOTTI, F. A Low-cost Neural-based Approach for Wood Types Classification. **CIMSA - International Conference on Computational Intelligence for Measurement Systems and Applications**, p. 12-16, Hong Kong, 13 nov. 2009.

LAURACEAE. **Atrium**, 2012. Disponivel em: <http://atrium.andesamazon.org/image_info.php?img=images/collections/jehouseholder_002169_01_p.jpg&id=56967#image56967>. Acesso em: 15/01/2013.

LENGOWSKI, E. C. **Caracterização e predição da cristalinidade de celulose através de espectroscopia no infravermelho e análise multivariada**. 108 p. Dissertação (Mestrado em Eng. Florestal). Universidade Federal do Paraná, 2012.

LIMA, J. P. H. **Redes Neurais Artificiais Aplicadas à otimização de processos de deposição de filmes finos poliméricos**. 134 p. Dissertação (Mestre em Engenharia Elétrica) Universidade de São Paulo. 2006.

LORENZI, H. **Árvores Brasileiras:** Manual de Identificação e Cultivo de Plantas Arbóreas Nativas do Brasil. [S.l.]: Instituto Plantarum de Estudos da Flora, v. 01, 2002.

McCLELLAND, J. L.; RUMELHART, D. E. **Explorations ins Parallel Distributed Processing**. Cambridge: MIT Press, 1988.

MENDES, K. B. **O uso de Redes Neurais Artificiais no diagnótico preditivo dos tipos mais freqüentes de cefaléia**. Curitiba: Dissertação(Mestrado em Métodos Numéricos), Universidade Federal do Paraná, 2008. 92 p.

MILANESI, L. D. S. **Estudo sobre a biologia da Canela-Sassafrás [Ocotea odorifera (Vell.) Rohwer] na Mata Atlântica do Rio Grande do Sul**. Porto Alegre: UFRGS - Instituto de Biociências, 2008.

MMA - Ministério do Meio Ambiente. **Lista Oficial das Espécies da Flora Brasileira Ameaçadas de Extinção**, 2008. Disponivel em: <http://www.mma.gov.br/estruturas/ascom_boletins/_arquivos/83_19092008034949.pdf>. Acesso em: 10/01/2013.

MORENO-MARTINEZ, J.; MARTIN-ARAGUZ, A. Santiago Ramón y Cajal: su actividad como médico militar (1873-1875). **Revista de Neurologia**, Barcelona, v. 35, p. 95-97, jul. 2002. ISSN PMID: 12389201.

NATIONAL Nanotechnology Research Center (unam). **Espectometry**. Disponível em: <www.nano.org.tr>. Acesso em: 18/04/2013

NISGOSKI, S. **Identificação e caracterização anatômica macroscópica das principais espécies utilizadas para laminação na região de curitiba - PR**. Dissertação (Mestrado em Eng. Florestal). 113 p. Universidade Federal do Paraná, 1999.

OLIVEIRA, J. E. D. *et al.* **Desenvolvimento Sustentável no noroeste de Mato Grosso**. [S.l.]: WWF - Brasil, 40 p. 2012.

PACKIANATHER, M. S.; DRAKE, P. R. Neural Networks for Classifying Images of Wood Veneer. **International Journal Advanced Manufacturing Technology**. p. 424-433. London, out. 2000.

PAULA FILHO, P. L. **Reconhecimento de espécies florestais através de imagens macroscópicas**. 116 f. Tese (Doutorado em Ciência da Computação). Universidade Federal do Paraná, 2012.

PIMENTEL, W. R. O. **Aplicação de redes neurais artificiais e de quimiometria na modelagem do processo de craqueamento catalítico fluído**. 181 f. Tese (Doutorado em Eng. Química). Universidade Estadual de Campinas, 2005.

PINTO JÚNIOR, J. E. Aspectos socioeconômicos, ambientais e legais da eucaliptocultura. **Sistemas de Produção**, v. 4, n. 2, p. 15-25, agosto 2010. ISSN ISSN 1678-8281.

PONTES, M. J. C. **Algoritmo das Projeções Sucessivas para Seleção de Variáveis Espectrais em Problemas de Classificação.** 202 f. Tese (Doutorado em Química). Universidade Estadual de Campinas, 2009.

REMADE. **Canela-parda**, 2009. Disponivel em: <http://www.remade.com.br/br/madeira_especies.php?num=150&title=&especie=Canela-Parda>. Acesso em: 10/01/2013.

RUMELHART, D. E.; McCLELLAND, J. L. **Parallel Distributed Processing: Explorations in the Microstructure of Cognition**. Vol. 1. Cambridge: MIT Press, 1986.

SANTOS, S. R. D. **Influência da qualidade da madeira de híbridos de Eucalyptus grandis x Eucalyptus urophylla e do processo Kraft de polpação na qualidade da polpa branqueada**. 115 p. Dissertação (Mestrado em Recusos Florestais), Universidade de São Paulo, 2005.

SILVA, M. D.; MEDEIROS, J. D. D. Anatomia da madeira de canela-preta: Ocotea Catharinensis mez.(Lauraceae). **INSULA**, Florianópolis, n. 29, p. 67-87, 2000.

SNIF - Sistema Nacional de Informações Florestais. **Espécies Florestais**. Disponivel em: <http://www.florestal.gov.br/snif/recursos-florestais/especies-florestais>. Acesso em: 10/01/2013.

STUART, B. **Infrared Spectroscopy:** Fundamentals and Applications. Hoboken: Wiley, 2004.

TAFNER, M. ; XEREZ, M.; FILHO, I. W. R. **Redes Neurais Artificiais : Introdução e princípios de neurocomputação.** Blumenau : EKO, 1995.

TATIBANA, C. Y.; KAETSU, D. Y. Uma Introdução às Redes Neurais. **Departamento de Informática**, 2009. Disponivel em: <http://www.din.uem.br/~ia/neurais/>. Acesso em: 19/12/2012.

THEOPHANIDES, T. Introduction to Infrared Spectroscopy. **Infrared Spectroscopy - Materials Science, Engineering and Technology**, abril 2012. ISSN ISBN 978-953-51-0537-4.

TRUGILHO, P. F; SILVA, J. R. M.; MORI, F. A.; LIMA, J. T.; MENDES, L. M.; MENDES, L. F. B. Rendimentos e características do carvão vegetal em função da posição radial de amostragem em clones de Eucaluptus. **Revista Cerne**, Lavras - MG, v. 11, n. 2, p. 178-186, abr./jun. 2005.

YU, H. **Advanced Learning Algorithms of Neural Networks**. 112 f. (Tese de Doutorado Ciência da Computação) Auburn University, 2011.

YU, H.; WILAMOWSKI, B. M. **Industrial Electronics Handbook**. 2ª ed. Auburn: AU, 2010.

ZENID, G. J.; CECCANTINI, G. C. T. **Identificação Macroscópica de madeiras**. São Paulo: IPT- SP, 2007.

ZORZETTO, R. Recontagem de neurônios põe em xeque ideias da neurociência. **Pesquisa FAPESP** , São Paulo, n. 192, fevereiro 2012.

APÊNDICES

APÊNDICE 1 - EXEMPLO DO MÉTODO DO GRADIENTE DESCENDENTE

Por motivos didáticos, este exemplo foi desenvolvido em planilha do software Microsoft Excel.

	P1	P2	T
A1	0	2	1
A2	1	2	1
A3	1	3	1
A4	1	0	0
A5	2	1	9

	Padrão de entrada		Saída
A1	0	2	1

Tx. Aprend	1

f1 = sigmoidal

f2 = sigmoidal

v = pesos para 1ª camada
w = pesos para 2ª camada

b1	1	2	3
1	0,900	0,900	-0,900

b2	1
1	-0,900

v	1	2	3
1	-0,900	0,900	-0,900
2	0,900	-0,900	0,900

w	1
1	0,900
2	-0,900
3	0,900

Primeira Camada

				Saída (a1)
n1,1 =	v11*P1+v12*P2+b1 =	2,700	f(n1,1) =	0,937
n1,2 =	v21*P1+v21*P2+b2 =	-0,900	f(n2,1) =	0,289
n1,3 =	v31*P1+v31*P2+b3 =	0,900	f(n3,1) =	0,711

Segunda Camada (Saída)

				Saída (a2)
n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	0,323	f(n2,1) =	0,5800633

erro = 0,4199367

Retropropagação (achar as sensitividades)

S21 =	-0,050	vide EQ. 33
S11 =	-0,003	vide EQ. 29
S12 =	0,009	vide EQ. 29
S13 =	-0,009	vide EQ. 29

Atualização dos pesos. Vide EQ. 21 e 22

b1	1	2	3
1	0,903	0,891	-0,891

b2	1
1	-0,850

v	1	2	3
1	-0,900	0,900	-0,900
2	0,905	-0,918	0,918

w	1
1	0,946
2	-0,886
3	0,935

	P1	P2	T
A1	0	2	1
A2	1	2	1
A3	1	3	1
A4	1	0	0
A5	2	1	9

	Padrão de entrada		Saída
A2	1	2	1

Tx. Aprend	1

f1 = sigmoidal

f2 = sigmoidal

b1	1	2	3
1	0,903	0,891	-0,891

b2	1
1	-0,850

v	1	2	3
1	-0,900	0,900	-0,900
2	0,905	-0,918	0,918

w	1
1	0,946
2	-0,886
3	0,935

Primeira Camada

				Saída (a1)
n1,1 =	v11*P1+v12*P2+b1 =	1,813	f(n1,1) =	0,860
n1,2 =	v21*P1+v21*P2+b2 =	-0,046	f(n2,1) =	0,489
n1,3 =	v31*P1+v31*P2+b3 =	0,046	f(n3,1) =	0,511

Segunda Camada

				Saída (a2)
n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	0,009	f(n2,1) =	0,5022229

erro = 0,4977771

Retropropagação (achar as sensitividades)

S21 =	-0,120	vide EQ. 33
S11 =	-0,014	vide EQ. 29
S12 =	0,027	vide EQ. 29
S13 =	-0,028	vide EQ. 29

Atualização dos pesos. Vide EQ. 21 e 22

b1	1	2	3
1	0,916	0,864	-0,863

b2	1
1	-0,730

v	1	2	3
1	-0,886	0,873	-0,872
2	0,933	-0,971	0,974

w	1
1	1,050
2	-0,827
3	0,997

	P1	P2	T
A1	0	2	1
A2	1	2	1
A3	1	3	1
A4	1	0	0
A5	2	1	9

	Padrão de entrada		Saída
A3	1	3	1

Tx. Aprend	1

f1 = sigmoidal

f2 = sigmoidal

b1	1	2	3
1	0,916	0,864	-0,863

b2	1
1	-0,730

v	1	2	3
1	-0,886	0,873	-0,872
2	0,933	-0,971	0,974

w	1
1	1,050
2	-0,827
3	0,997

Primeira Camada

n1,1 =	v11*P1+v12*P2+b1 =	2,828	f(n1,1) =	0,944
n1,2 =	v21*P1+v21*P2+b2 =	-1,177	f(n2,1) =	0,236
n1,3 =	v31*P1+v31*P2+b3 =	1,189	f(n3,1) =	0,766

Saída (a1)

Segunda Camada

n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	0,83	f(n2,1) =	0,6962872

Saída (a2)

erro = 0,3037128

Retropropagação (achar as sensitividades)

S21 =	-0,032	vide EQ. 33
S11 =	-0,002	vide EQ. 29
S12 =	0,005	vide EQ. 29
S13 =	-0,006	vide EQ. 29

Atualização dos pesos. Vide EQ. 21 e 22

b1	1	2	3
1	0,918	0,859	-0,857

b2	1
1	-0,698

v	1	2	3
1	-0,885	0,869	-0,866
2	0,938	-0,986	0,992

w	1
1	1,080
2	-0,819
3	1,021

	P1	P2	T
A1	0	2	1
A2	1	2	1
A3	1	3	1
A4	1	0	0
A5	2	1	9

	Padrão de entrada		Saída
A4	1	0	0

Tx. Aprend	1

f1 = sigmoidal

f2 = sigmoidal

b1	1	2	3
1	0,918	0,859	-0,857

b2	1
1	-0,698

v	1	2	3
1	-0,885	0,869	-0,866
2	0,938	-0,986	0,992

w	1
1	1,080
2	-0,819
3	1,021

Primeira Camada

n1,1 =	v11*P1+v12*P2+b1 =	0,034	f(n1,1) =	0,508
n1,2 =	v21*P1+v21*P2+b2 =	1,728	f(n2,1) =	0,849
n1,3 =	v31*P1+v31*P2+b3 =	-1,723	f(n3,1) =	0,151

Saída (a1)

Segunda Camada

n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	-0,69	f(n2,1) =	0,3338992

Saída (a2)

erro = -0,333899

Retropropagação (achar as sensitividades)

S21 =	0,167	vide EQ. 33
S11 =	0,045	vide EQ. 29
S12 =	-0,02	vide EQ. 29
S13 =	0,022	vide EQ. 29

Atualização dos pesos. Vide EQ. 21 e 22

b1	1	2	3
1	0,873	0,877	-0,879

b2	1
1	-0,865

v	1	2	3
1	-0,930	0,886	-0,888
2	0,938	-0,986	0,992

w	1
1	0,995
2	-0,961
3	0,996

	P1	P2	T
A1	0	2	1
A2	1	2	1
A3	1	3	1
A4	1	0	0
A5	2	1	0

	Padrão de entrada		Saída
A5	2	1	0

Tx. Aprend	1

f1 = sigmoidal

f2 = sigmoidal

b1	1	2	3
1	0,873	0,877	-0,879

b2	1
1	-0,865

v	1	2	3
1	-0,930	0,886	-0,888
2	0,938	-0,986	0,992

w	1
1	0,995
2	-0,961
3	0,996

Primeira Camada — Saída (a1)

n1,1 =	v11*P1+v12*P2+b1 =	-0,048	f(n1,1) =	0,488
n1,2 =	v21*P1+v21*P2+b2 =	1,664	f(n2,1) =	0,841
n1,3 =	v31*P1+v31*P2+b3 =	-1,664	f(n3,1) =	0,159

Segunda Camada — Saída (a2)

n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	-1,03	f(n2,1) =	0,263235

erro = -0,263235

Retropropagação (achar as sensitividades)

S21 =	0,132	vide EQ. 33
S11 =	0,033	vide EQ. 29
S12 =	-0,02	vide EQ. 29
S13 =	0,018	vide EQ. 29

Atualização dos pesos. Vide EQ. 21 e 22

b1	1	2	3
1	0,840	0,894	-0,897

b2	1
1	-0,997

v	1	2	3
1	-0,995	0,920	-0,923
2	0,905	-0,969	0,974

w	1
1	0,931
2	-1,072
3	0,975

Verificando os erros para todos os padrões de entrada.

	P1	P2	T
A1	0	2	1
A2	1	2	1
A3	1	3	1
A4	1	0	0
A5	2	1	0

b1	1	2	3
1	0,840	0,894	-0,897

b2	1
1	-0,997

v	1	2	3
1	-0,995	0,920	-0,923
2	0,905	-0,969	0,974

w	1
1	0,931
2	-1,072
3	0,975

PADRÃO (A1)

Primeira Camada — Saída (a1)

n1,1 =	v11*P1+v12*P2+b1 =	2,651	f(n1,1) =	0,934
n1,2 =	v21*P1+v21*P2+b2 =	-1,044	f(n2,1) =	0,260
n1,3 =	v31*P1+v31*P2+b3 =	1,052	f(n3,1) =	0,741

Segunda Camada — Saída (a2)

n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	0,316	f(n2,1) =	0,5783557

erro = 0,4216443

PADRÃO (A2)

Primeira Camada — Saída (a1)

n1,1 =	v11*P1+v12*P2+b1 =	2,651	f(n1,1) =	0,934
n1,2 =	v21*P1+v21*P2+b2 =	-0,124	f(n2,1) =	0,469
n1,3 =	v31*P1+v31*P2+b3 =	0,128	f(n3,1) =	0,532

Segunda Camada — Saída (a2)

n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	-0,1114	f(n2,1) =	0,4721688

erro = 0,5278312

PADRÃO (A3)

Primeira Camada — Saída (a1)

n1,1 =	v11*P1+v12*P2+b1 =	2,561	f(n1,1) =	0,928
n1,2 =	v21*P1+v21*P2+b2 =	-1,093	f(n2,1) =	0,251
n1,3 =	v31*P1+v31*P2+b3 =	1,102	f(n3,1) =	0,751

Segunda Camada — Saída (a2)

n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	0,33	f(n2,1) =	0,5817653

erro = 0,4182347

PADRÃO (A4)				
Primeira Camada				Saída (a1)
n1,1 =	v11*P1+v12*P2+b1 =	-0,155	f(n1,1) =	0,461
n1,2 =	v21*P1+v21*P2+b2 =	1,814	f(n2,1) =	0,860
n1,3 =	v31*P1+v31*P2+b3 =	-1,820	f(n3,1) =	0,139
Segunda Camada				Saída (a2)
n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	-1,3529	f(n2,1) =	0,2053937
			erro =	-0,205394
PADRÃO (A5)				
Primeira Camada				Saída (a1)
n1,1 =	v11*P1+v12*P2+b1 =	-0,244	f(n1,1) =	0,439
n1,2 =	v21*P1+v21*P2+b2 =	1,765	f(n2,1) =	0,854
n1,3 =	v31*P1+v31*P2+b3 =	-1,769	f(n3,1) =	0,146
Segunda Camada				Saída (a2)
n2,1 =	w11*f1+w12*f2+f3*w13+b2 =	-1,3611	f(n2,1) =	0,2040594
			erro =	-0,204059

ERRO QUADRÁTICO MÉDIO

ERRO = 0,14303

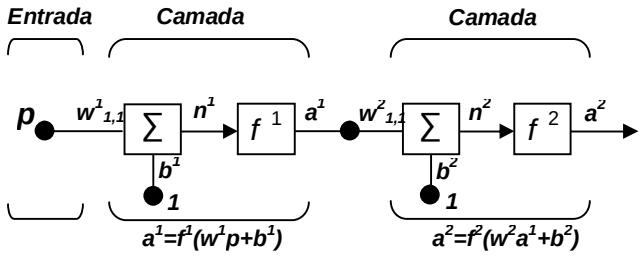
APÊNDICE 2 - CÓDIGO UTLIZADO PARA CRIAÇÃO DAS FIGURAS (33) E (34)

Código em linguagem MATLAB.

```
%=====
% VARIÁVEIS DEFINIDAS
%=====
i = 2;
mu = 0.9;  % termo de momento
w(1,1)= 1;
y(1,1) = 0;
x(1,1) = 0;
%=====
% LOOP DE CÁLCULO
%=====
for k = 2:0.5:200
    w(1,i) = 1 + sin((2*pi*k)/16); % função base
    y(1,i) = mu*y(i-1) + (1-mu)*w(i);  % filtro de 1ª ordem
    x(1,i) = k;
    i = i + 1;
end
%=====
% GRÁFICO
%=====
hold on
plot(x,w, 'k-')
plot(x,y, 'r-')
xlabel('EIXO X')
ylabel('EIXO Y')
hold off
```

APÊNDICE 3 - EXEMPLO SIMPLES DO MÉTODO DE LEVEMBERG-MARQUARDT

Funcionamento do método de Levemberg-Marquardt considere uma rede com a arquitetura representada pela figura abaixo.



Considerar a funções de ativação:

$$f^1(n) = (n)^2, \qquad f^2(n) = n$$

As suas derivadas como:

$$\dot{f}^1(n) = 2n, \qquad \dot{f}^2(n) = 1$$

Padrões de entrada:

$[p_1=1,\ t_1=1]$ e $[p_2=2,\ t_2=2]$

Pesos:

$W_1 = 1,\ b_1 = 0$

$W_2 = 2,\ b_2 = 1$

O primeiro passo é propagar as entradas através da rede e calcular os erros.

$$a_1^0 = p_1 = [1]$$

$$n_1^1 = W^1 a_1^0 + b^1 = [1][1] + [0] = [1]$$

$$a_1^1 = f^1(n_1^1) = ([1])^2 = [1]$$

$$n_1^2 = W^2 a_1^1 + b^2 = [2][1] + [1] = [3]$$

$$a_1^2 = f^2(n_1^2) = ([3]) = [3]$$

$$e_1 = (t_1 - a_1^2) = ([1] - [3]) = [-2]$$

$$a_2^0 = p_2 = [2]$$

$$n_2^1 = W^1 a_2^0 + b^1 = [1][2] + [0] = [2]$$

$$a_2^1 = f^1(n_2^1) = ([2])^2 = [4]$$

$$n_2^2 = W^2 a_2^1 + b^2 = [2][4] + [1] = [9]$$

$$a_2^2 = f^2(n_2^2) = ([9]) = [9]$$

$$e_2 = (t_1 - a_1^2) = ([2] - [9]) = [-7]$$

$$\boldsymbol{e_1^*} = \begin{bmatrix} -2 \\ -7 \end{bmatrix}$$

$$\boldsymbol{MSE} = \boldsymbol{e} = \mathbf{0,5} * \boldsymbol{e_1^*} * \boldsymbol{e_1^*}^T = 0{,}5 * 53 = 26{,}5$$

O próximo passo é encontrar e retropropagar as sensitividades usando as equações (61) e (62).

$$\tilde{S}_1^2 = -\dot{\boldsymbol{F}}^2(n_1^2) = -[1]$$

$$\tilde{S}_1^1 = \dot{\boldsymbol{F}}^1(n_1^1)(W^2)^T \tilde{S}_1^2 = \left[2n_{1,1}^1\right][2][-1] = [2(1)][2][-1] = [-4]$$

$$\tilde{S}_2^2 = -\dot{\boldsymbol{F}}^2(n_2^2) = -[1]$$

$$\tilde{S}_2^1 = \dot{\boldsymbol{F}}^1(n_2^1)(W^2)^T \tilde{S}_2^2 = \left[2n_{1,2}^2\right][2][-1] = [2(2)][2][-1] = [-8]$$

$$\tilde{S}^1 = \left[\tilde{S}_1^1 \middle| \tilde{S}_2^1\right] = [4 \quad -8]$$

$$\tilde{S}^2 = \left[\tilde{S}_1^2 \middle| \tilde{S}_2^2\right] = [1 \quad -1]$$

Agora, calcular a matriz jacobiana utilizando as equações (55), (58) e (59).

$$J(\boldsymbol{w}) = \begin{bmatrix} \frac{\partial \bar{e}_1}{\partial w_1} & \frac{\partial \bar{e}_1}{\partial w_2} & \frac{\partial \bar{e}_1}{\partial w_3} & \frac{\partial \bar{e}_1}{\partial w_4} \\ \frac{\partial \bar{e}_2}{\partial w_1} & \frac{\partial \bar{e}_2}{\partial w_2} & \frac{\partial \bar{e}_2}{\partial w_3} & \frac{\partial \bar{e}_2}{\partial w_4} \end{bmatrix} = \begin{bmatrix} \frac{\partial e_{1,1}}{\partial w_{1,1}^1} & \frac{\partial e_{1,1}}{\partial b_1^1} & \frac{\partial e_{1,1}}{\partial w_{1,1}^2} & \frac{\partial e_{1,1}}{\partial b_1^2} \\ \frac{\partial e_{1,2}}{\partial w_{1,1}^1} & \frac{\partial e_{1,2}}{\partial b_1^1} & \frac{\partial e_{1,2}}{\partial w_{1,1}^2} & \frac{\partial e_{1,2}}{\partial b_1^2} \end{bmatrix}$$

$$[\boldsymbol{J}]_{1,1} = \frac{\partial \bar{e}_1}{\partial w_1} = \frac{\partial e_{1,1}}{\partial w_{1,1}^1} = \frac{\partial e_{1,1}}{\partial n_{1,1}^1} \times \frac{\partial n_{1,1}^1}{\partial w_{1,1}^1} = \tilde{s}_{1,1}^1 \times \frac{\partial n_{1,1}^1}{\partial w_{1,1}^1} = \tilde{s}_{1,1}^1 \times a_{1,1}^0 = (-4)(1) = -4$$

$$[\boldsymbol{J}]_{1,2} = \frac{\partial \bar{e}_1}{\partial w_2} = \frac{\partial e_{1,1}}{\partial b_1^1} = \frac{\partial e_{1,1}}{\partial n_{1,1}^1} \times \frac{\partial n_{1,1}^1}{\partial b_1^1} = \tilde{s}_{1,1}^1 \times \frac{\partial n_{1,1}^1}{\partial b_1^1} = \tilde{s}_{1,1}^1 = -4$$

$$[\boldsymbol{J}]_{1,3} = \frac{\partial \bar{e}_1}{\partial w_3} = \frac{\partial e_{1,1}}{\partial w_{1,1}^2} = \frac{\partial e_{1,1}}{\partial n_{1,1}^2} \times \frac{\partial n_{1,1}^2}{\partial w_{1,1}^2} = \tilde{s}_{1,1}^2 \times \frac{\partial n_{1,1}^2}{\partial w_{1,1}^2} = \tilde{s}_{1,1}^2 \times a_{1,1}^1 = (-1)(1) = -1$$

$$[\boldsymbol{J}]_{1,4} = \frac{\partial \bar{e}_1}{\partial w_4} = \frac{\partial e_{1,1}}{\partial b_1^2} = \frac{\partial e_{1,1}}{\partial n_{1,1}^2} \times \frac{\partial n_{1,1}^2}{\partial b_1^2} = \tilde{s}_{1,1}^2 \times \frac{\partial n_{1,1}^2}{\partial b_1^2} = \tilde{s}_{1,1}^2 = -1$$

$$[\boldsymbol{J}]_{2,1} = \frac{\partial \bar{e}_2}{\partial w_1} = \frac{\partial e_{1,2}}{\partial w_{1,1}^1} = \frac{\partial e_{1,2}}{\partial n_{1,2}^1} \times \frac{\partial n_{1,2}^1}{\partial w_{1,1}^1} = \tilde{s}_{1,2}^1 \times \frac{\partial n_{1,2}^1}{\partial w_{1,1}^1} = \tilde{s}_{1,2}^1 \times a_{1,2}^0 =$$

$$= (-8)(2) = -16$$

$$[\boldsymbol{J}]_{2,2} = \frac{\partial \bar{e}_2}{\partial w_2} = \frac{\partial e_{1,2}}{\partial b_1^1} = \frac{\partial e_{1,2}}{\partial n_{1,2}^1} \times \frac{\partial n_{1,2}^1}{\partial b_1^1} = \tilde{s}_{1,2}^1 \times \frac{\partial n_{1,2}^1}{\partial b_1^1} = \tilde{s}_{1,2}^1 = -8$$

$$[\boldsymbol{J}]_{2,3} = \frac{\partial \bar{e}_2}{\partial w_3} = \frac{\partial e_{1,2}}{\partial w_{1,1}^2} = \frac{\partial e_{1,2}}{\partial n_{1,2}^2} \times \frac{\partial n_{1,2}^2}{\partial w_{1,1}^2} = \tilde{s}_{1,2}^2 \times \frac{\partial n_{1,2}^2}{\partial w_{1,1}^2} = \tilde{s}_{1,2}^2 \times a_{1,2}^1 = (-1)(4) = -4$$

$$[\boldsymbol{J}]_{2,4} = \frac{\partial \bar{e}_2}{\partial w_4} = \frac{\partial e_{1,2}}{\partial b_1^2} = \frac{\partial e_{1,2}}{\partial n_{1,2}^2} \times \frac{\partial n_{1,2}^2}{\partial b_1^2} = \tilde{s}_{1,2}^2 \times \frac{\partial n_{1,2}^2}{\partial b_1^2} = \tilde{s}_{1,2}^2 = -1$$

Assim, a matriz jacobiana será:

$$\boldsymbol{J}(\boldsymbol{w}) \begin{bmatrix} -4 & -4 & -1 & -1 \\ -16 & -8 & -4 & -1 \end{bmatrix}$$

Pela equação (53) podemos achar Δw, considerando **μ** = 0,1.

$$\Delta w = -\left[\begin{bmatrix} -4 & -16 \\ -4 & -8 \\ -1 & -4 \\ -1 & -1 \end{bmatrix} * \begin{bmatrix} -4 & -4 & -1 & -1 \\ -16 & -8 & -4 & -1 \end{bmatrix} + 0{,}1 * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\right]^{-1} * \begin{bmatrix} -4 & -16 \\ -4 & -8 \\ -1 & -4 \\ -1 & -1 \end{bmatrix}$$

$$* \begin{bmatrix} -2 \\ -7 \end{bmatrix} = \begin{bmatrix} -0{,}3524 \\ -0{,}1255 \\ -0{,}0881 \\ -0{,}0030 \end{bmatrix}$$

Como,

$$w_{antigo} = [1 \quad 0 \quad 2 \quad 1]$$

E,

$$w_{novo} = w_{antigo}^{T} + \Delta w$$

Logo,

$$w_{novo} = \begin{bmatrix} 1 \\ 0 \\ 2 \\ 1 \end{bmatrix} + \begin{bmatrix} -0{,}3524 \\ -0{,}1255 \\ -0{,}0881 \\ -0{,}0030 \end{bmatrix} = \begin{bmatrix} 0{,}6476 \\ -0{,}1255 \\ 1{,}9119 \\ 0{,}9970 \end{bmatrix}$$

Recalculando o erro.

$$n_1^1 = W^1 a_1^0 + b^1 = [0{,}6476][1] + [-0{,}1255] = [0{,}5221]$$

$$a_1^1 = f^1(n_1^1) = ([0{,}5221])^2 = [0{,}2726]$$

$$n_1^2 = W^2 a_1^1 + b^2 = [1{,}9119][0{,}2726] + [0{,}997] = [1{,}5182]$$

$$a_1^2 = f^2(n_1^2) = [1{,}5182]$$

$$e_1 = (t_1 - a_1^2) = ([1] - [1{,}5182]) = [-0{,}5182]$$

$$n_2^1 = W^1 a_2^0 + b^1 = [0{,}6476][2] + [-0{,}1255] = [1{,}1697]$$

$$a_2^1 = f^1(n_2^1) = ([1{,}1697])^2 = [1{,}3682]$$

$$n_2^2 = W^2 a_2^1 + b^2 = [1{,}9119][1{,}3682] + [0{,}997] = [3{,}6128]$$

$$a_2^2 = f^2(n_2^2) = [3{,}6128]$$

$$e_2 = (t_1 - a_1^2) = ([2] - [3{,}6128]) = [-1{,}6129]$$

$$\boldsymbol{e}_2^* = \begin{bmatrix} -0{,}5182 \\ -1{,}6129 \end{bmatrix}$$

$$\boldsymbol{MSE} = \boldsymbol{e} = 0{,}5 * \boldsymbol{e}_2^* * \boldsymbol{e}_2^{*\boldsymbol{T}} = 0{,}5 * 2{,}87 = 1{,}405$$

(...)

Repete-se o processo até atingir o erro desejado.

- Se o erro quadrático for menor que o anterior deve-se dividr **μ** por **v** e reiniciar o processo.
- Se o erro quadrático for maior que o anterior deve-se multiplicar **μ** por **v**, atualizar os pesos novamente e, se houver redução do erro, proceder como em (1). Do contrário repitir (2) novamente.

APÊNDICE 4 - ALGORITIMO PRINCIPAL PARA TREINAMENTO DA REDE

Código em linguagem MATLAB.

```
%====
% INICIO DO PROGRAMA
%====

p = 1;                                                  % contador de ensaios

while p <= 10

load cis.mat                                            % abre as informações sobre o conjunto CIS
eval(['mkdir C:\MATLAB\3_esp\ensaio_' int2str(p) ]);    % cria o diretório ensaio_nº na pasta C:\MATLAB\3_esp\

%====
% TAMANHO DA MATRIZES
%====

[itreino jtreino] = size(d_treino);  % informa tamanho da matriz de treinamento
[iteste jteste] = size(d_teste);     % informa tamanho da matriz de teste

%====
% INICIO DA ETAPA DE TREINAMENTO
%====

for i=1:15    % varia o número de neurônios na camada escondida

tic
eval(['net_' int2str(i) ' = newff(treino_a, d_a,[' int2str(i) '],{''tansig'',''purelin''},''trainlm'');']);
                                                        % treinamento da rede, definido as funções de
                                                          transferência e aprendizagem
eval(['net_' int2str(i) '.trainParam.lr_inc = 1.03;']);  % valor de incremento da taxa de aprendizagem
eval(['net_' int2str(i) '.trainParam.max_fail = 10;']);  % número máximo de falha de validação toleradas
eval(['net_' int2str(i) '.trainParam.epochs = 1000;']);  % número máximo de épocas/iterações permitidas
eval(['net_' int2str(i) '.trainParam.goal = 1e-10;']);   % erro pretendido
```

```
eval(['net_' int2str(i) '.divideFcn = ''dividerand'';']);       % ativa a divisão do conjunto de treinamento
eval(['net_' int2str(i) '.divideParam.trainRatio = 0.7;']);     % define 70% do conjunto de treinamento para
                                                                  treinamento
eval(['net_' int2str(i) '.divideParam.testRatio = 0.1;']);      % define 10% do conjunto de treinamento para teste
eval(['net_' int2str(i) '.divideParam.valRatio = 0.2;']);       % define 20% do conjunto de treinamento para
                                                                  validação
eval(['net_' int2str(i) '.performFcn = ''mse'';']);             % critério de desempenho utilizado (Erro Quadrático
                                                                   Médio)
eval(['[net_' int2str(i) ',net_treino_info_' int2str(i) ' ,net_output_' int2str(i) ',net_erros_' int2str(i)
',net_pf_' int2str(i) ',net_Af_' int2str(i) '] = train(net_' int2str(i) ', treino_a, d_a);'])  % salva informações
                                                                                                  de treinamento

%====
% INICIO DA ETAPA DE TESTE
%====

eval(['net_' int2str(i) '_outputs_treino = sim(net_' int2str(i) ', treino);']);   % simula a rede fornecendo o
                                                                                    conjunto de treinamento
eval(['net_' int2str(i) '_outputs_teste = sim(net_' int2str(i) ', teste);']);     % simula a rede fornecendo o
                                                                                    conjunto de teste
eval(['net_' int2str(i) '_erro_medio_treino = sum((d_treino - net_' int2str(i) '_outputs_treino).^2) ./ jtreino;']);
                                                                                  % calcula o erro quadrático médio
eval(['net_' int2str(i) '_erro_medio_teste = sum((d_teste - net_' int2str(i) '_outputs_teste).^2) ./ jteste;']);
                                                                                  % calcula o erro quadrático médio
eval(['tempo_' int2str(i) ' = toc;']);                                            % tempo de execução para a rede
                                                                                    com i neurônios


%====
% CONTROLE DE EXECUÇÃO
%====

iteracao = i      % indicador de iteração em execução

%====
% SALVA ITERAÇÃO
%====
eval(['save C:\MATLAB\3_esp\ensaio_' int2str(p) '\final_cis_' int2str(p) ]);   % salva as informações da iteração no
                                                                                  arquivo              "final_cis_nº"
```

```
end                     % final da função "for"

p = p+1;                % contador para p

save ('p.mat', 'p')     % salva informações do contador para o próximo ensaio

clear                   % limpa workspace para o novo ensaior

load p.mat              % inicializa o contado p

end                     % final do while

%====
% FIM DO PROGRAMA
%====
```