

**UNIVERSIDADE DE SÃO PAULO  
ESCOLA POLITÉCNICA**

**ERIC JUN URABAYASHI**

**Desenvolvimento de uma rede neural artificial para a  
previsão do comportamento constitutivo de uma liga de  
alumínio AFNOR 7020**

**São Paulo  
2005**

ERIC JUN URABAYASHI

**Desenvolvimento de uma rede neural artificial para a  
previsão do comportamento constitutivo de uma liga de  
alumínio AFNOR 7020**

Trabalho de Formatura apresentado  
à Escola Politécnica da  
Universidade de São Paulo

Área de concentração: Engenharia  
de Materiais  
Orientador: Prof. Dr. Ronald Lesley  
Plaut

São Paulo  
2005

## DEDICATÓRIA

Dedico este meu trabalho a todos os profissionais e estudantes envolvidos no estudo de materiais como ligas de alumínio.

## AGRADECIMENTOS

Ao Prof. Dr. Ronald Lesley Plaut , pela oportunidade.

À Lia, por me apoiar incondicionalmente em todas as ocasiões em que precisei.

A Deus, que me observa percorrer o caminho que Ele já traçou para mim.

## RESUMO

Devido ao avanço tecnológico e a modernização da fabricação de materiais como os metais, necessitou-se da utilização de programas computadorizados que controlem, monitorem e projetem eficientemente variáveis críticas do processo, como tensão, deformação, taxa de deformação e temperatura. Para isso, criou-se uma tendência a introduzir, entender e utilizar programas que supram essa necessidade. Assim, apresenta-se neste trabalho uma introdução sobre os conceitos fundamentais de redes neurais e a utilização do programa Neurosolutions como principal ferramenta desenvolvedora de uma rede neural, mostrando, inicialmente, suas funções e posteriormente, uma aplicação em uma liga de alumínio. Discute-se também a eficiência do programa e sua real aplicação na indústria como ferramenta vital no processo de produção.

## LISTA DE FIGURAS

Figura 2.1. Modelo de McCulloch e Pitts.....	13
Figura 2.2 - Esquema de um neurônio artificial.....	14
Figura 2.3 -Organização em camadas.....	15
Figura 3.1 Combinação de dados.....	25
Figura 3.2 NeuroSolutions for Excel.....	27
Figura 3.3 Comandos de inserção e seleção de dados.....	28
Figura .3.4 Divisão de dados.....	29
Figura 3.5 Divisão de conjuntos.....	30
Figura 3.6 Seleção da rede.....	31
Figura 3.7 Seleção de camadas intermediárias.....	32
Figura 3.8 – Precisão das camadas intemediárias.....	33
Figura 3.9 – Configuração da camada de saída.....	33
Figura 3.10 Rede construída.....	34
Figura 3.11 Seleção de iterações para treinamento da rede.....	35
Figura 3.12 Curvas de treino e validação.....	36
Figura 3.13 Curvas de treino.....	37
Figura 3.14 Curvas de validação.....	37
Figura 3.15 Seleção do teste.....	38
Figura 3.16 Curvas de saída.....	39
Figura 3.17 Rede Testada.....	40
Figura 3.18 Tensão Teórica x Tensão gerada pela rede neural.....	41

Figura 3.19 Teste da influência da variável.....	42
Figura 3.20 Dados de Produção em tempo de execução.....	43
Figura 4.1 Taxa de deformação x Tensão para a T=500°C.....	45

## LISTA DE TABELAS

Tabela 3.1 Valores de Tensão da liga AFNOR 7020.....	26
--	----

## SUMÁRIO

<b>1.INTRODUÇÃO.....</b>	<b>10</b>
<b>2.REDES NEURAIS.....</b>	<b>11</b>
<b>2.1 Introdução as redes neurais.....</b>	<b>11</b>
<b>2.2 O Neurônio Artificial e a Rede Neural.....</b>	<b>12</b>
<b>2.2.1 O Neurônio Artificial.....</b>	<b>13</b>
<b>2.3 A Rede Neural Artificial (Multilayer Perceptron).....</b>	<b>14</b>
<b>2.4. Desenvolvimento de Aplicações.....</b>	<b>17</b>
<b>2.4.1. Coleta de dados e separação em conjuntos.....</b>	<b>17</b>
<b>2.4.2. Configuração da rede.....</b>	<b>18</b>
<b>2.4.3. Treinamento.....</b>	<b>18</b>
<b>2.4.4. Teste.....</b>	<b>20</b>
<b>2.4.5. Integração da rede.....</b>	<b>20</b>
<b>2.5. Aplicações de Redes Neurais Artificiais.....</b>	<b>21</b>
<b>3.NEUROSOLUTIONS.....</b>	<b>24</b>
<b>3.1 A liga de Alumínio AFNOR 7020.....</b>	<b>24</b>
<b>3.2 Utilização do programa.....</b>	<b>26</b>
<b>4.CONCLUSÃO.....</b>	<b>44</b>
<b>REFERÊNCIAS.....</b>	<b>46</b>

## 1.INTRODUÇÃO

---

O presente documento representa um trabalho de formatura, requisito necessário e fundamental à conclusão do curso de Engenharia de Materiais da Escola Politécnica da Universidade de São Paulo (EPUSP)

Este trabalho apresenta um breve estudo sobre redes neurais, mostrando suas funções, aplicações, e principalmente, o funcionamento e utilização de um programa denominado Neurosolutions como principal software no desenvolvimento de redes neurais, através de aplicações em ligas de alumínio. Para atingir esse objetivo, apoiou-se em artigos publicados e na versão shareware do programa em questão.

## 2.REDES NEURAIS

---

O presente capítulo visa explicar sucintamente o conceito de Redes Neurais, mostrando sua origem, principais modelos utilizados e aplicações.

### 2.1 Introdução às redes neurais

As redes neurais artificiais consistem em um método de solucionar problemas de inteligência artificial, construindo um sistema que tenha circuitos que simulem o cérebro humano, inclusive seu comportamento, ou seja, aprendendo, errando e fazendo descobertas, em um modelo inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência.

Mas o que é uma rede neural?

Uma rede neural artificial é composta por várias unidades de processamento, cujo funcionamento é bastante simples. Estas unidades geralmente são conectadas por canais de comunicação que estão associados a determinado peso. As unidades fazem operações apenas sobre seus dados locais, que são entradas recebidas pelas suas conexões. O comportamento inteligente de uma Rede Neural Artificial vem das interações entre as unidades de processamento da rede.

Como opera uma rede?

De forma geral, a operação de uma célula da rede se resume em:

- Dados são apresentados à entrada;
- Cada dado é multiplicado por um peso que indica sua influência na saída da unidade;
- É feita a soma ponderada dos dados que produz um nível de atividade;
- Se este nível excede um limite (threshold) a unidade produz uma saída;

## 2.2 O Neurônio Artificial e a Rede Neural

Assim como o sistema nervoso, que é a sede da inteligência, o lugar onde se formam as idéias e do qual partem as ordens para a execução de movimentos, e regulação de funções e estímulos, é composto por bilhões de células nervosas, a rede neural artificial é formada por pequenos módulos, ou objetos, que simulam o funcionamento de um neurônio. Estes módulos devem funcionar de acordo com os elementos em que foram inspirados, recebendo e retransmitindo, e gerando informações.

### 2.2.1 O Neurônio Artificial

Atualmente, o modelo mais utilizado para explicar o funcionamento de uma rede neural é o do fisiologista Warrem MacCulloch. Ele interpretou o funcionamento do neurônio biológico como sendo um circuito de entradas binárias combinadas por uma soma ponderada (com pesos) produzindo uma entrada efetiva:

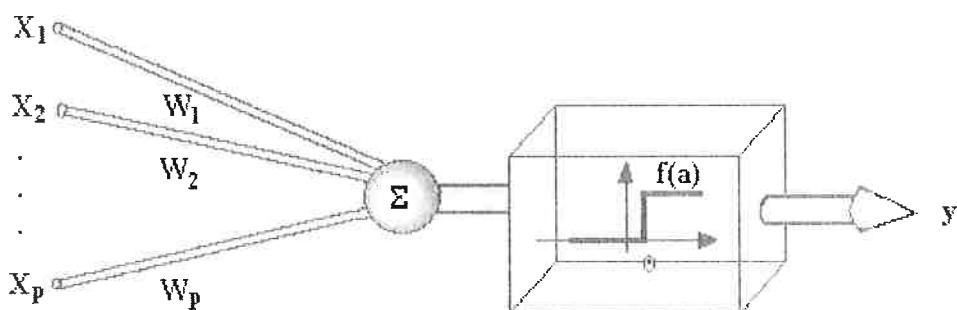


Figura 2.1. Modelo de McCulloch e Pitts

No modelo geral de neurônio (fig. 01), as entradas  $WiUi$  são combinadas rebatendo em uma função  $F$ , para produzir um estado de ativação do neurônio. As entradas chegam através dos dendritos, que são receptores dos estímulos transmitidos por outros neurônios, e tem um peso atribuído pela sinapse, que são junções interneuronais.

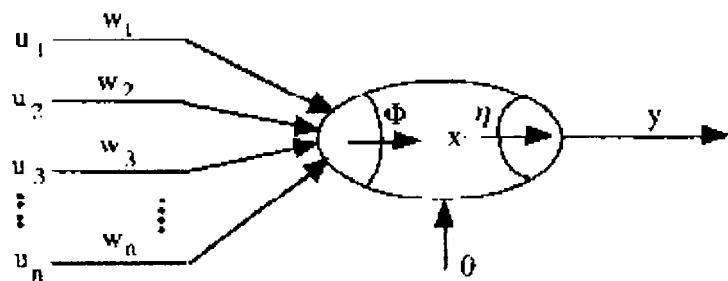


Figura 2.2 - Esquema de um neurônio artificial

A função básica de um neurônio é somar as entradas e retornar uma saída, como mostra a figura 2.2.

### 2.3 A Rede Neural Artificial (Multilayer Perceptron)

A rede neural artificial é um sistema de neurônios ligados por conexões sinápticas e dividido em neurônios de entrada, que recebem estímulos do meio externo, neurônios internos, que efetuam todo processo de classificação e ponderação dos estímulos, e neurônios de saída, que se comunicam com o exterior.

Existem várias formas de montar essa rede, mas a mais utilizada é a Multilayer Perceptron, que foi concebida para resolver problemas mais complexos, os quais não poderiam ser resolvidos pelo modelo de neurônio básico. Um único perceptron ou uma combinação das saídas de alguns perceptrons poderia realizar uma operação, porém,

seria incapaz de aprendê-la. Para isto são necessárias mais conexões, as quais só existem em uma rede de perceptrons dispostos em camadas, como mostra a figura 2.3. Dessa forma, os neurônios internos são de suma importância na rede neural pois se provou que se torna impossível a resolução de problemas linearmente não separáveis, já que quanto mais camadas a rede possuir, mais refinado será o resultado de saída da rede. Assim pode-se dizer que uma rede é composta por várias unidades de processamento. Essas unidades, geralmente são conectadas por canais de comunicação que estão associados a determinado peso. As unidades fazem operações apenas sobre seus dados locais, que são entradas recebidas pelas suas conexões. A inteligência da Rede Neural Artificial vem das interações entre as unidades de processamento da rede.

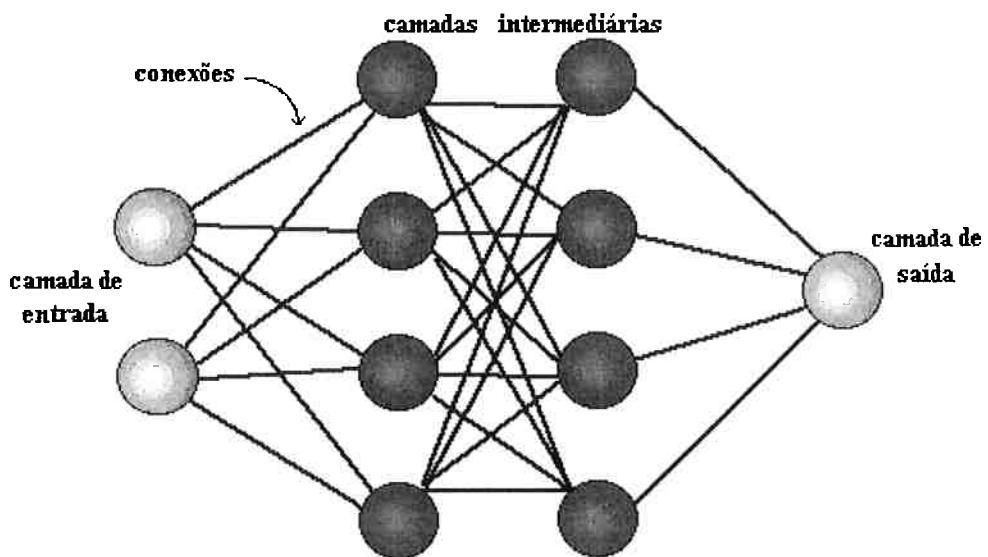


Figura 2.3 -Organização em camadas

A maioria dos modelos de redes neurais possui alguma regra de treinamento, onde os pesos de suas conexões são ajustados de acordo com os padrões apresentados, aprendendo de acordo com os exemplos apresentados. Arquiteturas neurais são tipicamente organizadas em camadas, com unidades que podem estar conectadas às unidades da camada posterior.

A rede neural passa por um processo de treinamento a partir dos casos reais conhecidos, adquirindo, a partir daí, a sistemática necessária para executar adequadamente o processo desejado dos dados fornecidos. Então, a rede neural é capaz de extrair regras básicas a partir de dados reais, diferindo da computação programada, onde é necessário um conjunto de regras rígidas pré-fixadas e algoritmos.

Usualmente as camadas são classificadas em três grupos:

- Camada de Entrada: onde os padrões são apresentados à rede;
- Camadas Intermediárias ou Ocultas: onde é feita a maior parte do processamento, através das conexões ponderadas; podem ser consideradas como extratoras de características;
- Camada de Saída: onde o resultado final é concluído e apresentado.

De acordo com a arquitetura da rede neural, existe uma classificação conforme implementação, topologia, características de seus nós, regras de treinamento, e tipos de modelos.

## 2.4. Desenvolvimento de Aplicações

Existem certos procedimentos para a construção de uma rede neural consistente:

### 2.4.1. Coleta de dados e separação em conjuntos

Os dois primeiros passos do processo para o desenvolvimento de redes neurais artificiais são suma importância para diminuir a probabilidade de erro do sistema: a coleta de dados relativos ao problema; e a sua separação em um conjunto de treinamento, um conjunto de validação, e um conjunto de testes. Esta tarefa requer uma análise cuidadosa sobre o problema para minimizar ambigüidades e erros nos dados. Além disso, os dados coletados devem ser significativos e cobrir amplamente o domínio do problema; devem cobrir operações normais ou rotineiras, exceções e as condições nos limites do domínio do problema.

Normalmente, os dados coletados são separados em três categorias: dados de treinamento, usados para o treinamento da rede, dados de validação, utilizado para verificar a eficiência da rede quanto a sua capacidade de generalização durante o treinamento, e podendo ser empregado como critério de parada do treinamento, e dados de teste, utilizados para verificar sua performance sob condições reais de utilização.

Geralmente, os dados são colocados em ordem aleatória para prevenção de tendências associadas à ordem de apresentação dos dados. É recomendável, se necessário, pré-processar estes dados, através de normalizações, escalonamentos e conversões de formato para torná-los mais apropriados à sua utilização na rede.

#### 2.4.2. Configuração da rede

O terceiro passo é a definição da configuração da rede, que pode ser dividido em três etapas:

- Seleção do paradigma neural apropriado à aplicação.
- Determinação da topologia da rede a ser utilizada - o número de camadas, o número de unidades em cada camada, etc.
- Determinação de parâmetros do algoritmo de treinamento e funções de ativação. Este passo tem um grande impacto na performance do sistema resultante.

#### 2.4.3. Treinamento

O quarto passo é o treinamento da rede. Nesta fase, seguindo o algoritmo de treinamento escolhido, serão ajustados os pesos das conexões. É importante considerar, nesta fase, alguns aspectos tais como a inicialização da rede, o modo de treinamento e o tempo de treinamento.

Uma boa escolha dos valores iniciais dos pesos da rede pode diminuir o tempo necessário para o treinamento. Normalmente, os valores iniciais dos pesos da rede são números aleatórios uniformemente distribuídos, em um intervalo definido. A escolha errada destes pesos pode levar a uma saturação prematura.

Quanto ao modo de treinamento, na prática é mais utilizado o modo padrão devido ao menor armazenamento de dados, além de ser menos suscetível ao problema de mínimos locais, devido à pesquisa de natureza estocástica que realiza. Por outro lado, no modo “batch” se tem uma melhor estimativa do vetor gradiente, o que torna o treinamento mais estável. A eficiência relativa dos dois modos de treinamento depende do problema que está sendo tratado[1].

Quanto ao tempo de treinamento, vários fatores podem influenciar a sua duração, porém sempre será necessário utilizar algum critério de parada. Normalmente, o critério de parada utilizado é um número máximo de ciclos. Mas, devem ser considerados a taxa de erro médio por ciclo, e a capacidade de generalização da rede. Pode ocorrer que em um determinado instante do treinamento a rede comece a degenerar, causando o problema de “over-training”, ou seja a rede se especializa em um conjunto de dados de treinamento e perde a capacidade de generalização[1].

O treinamento deve ser interrompido quando a rede apresentar uma boa capacidade de generalização e quando a taxa de erro for suficientemente pequena, ou seja, menor que um erro admissível. Assim, deve-se encontrar um ponto ótimo de parada com erro mínimo e capacidade de generalização máxima[1].

#### 2.4.4. Teste

O quinto passo é o teste da rede. Durante esta fase o conjunto de teste é utilizado para determinar a performance da rede com dados que não foram previamente utilizados. A performance da rede, medida nesta fase, é uma boa indicação de sua performance real.

Devem ser considerados ainda outros testes como análise do comportamento da rede utilizando entradas especiais e análise dos pesos atuais da rede, pois se existirem valores muito pequenos, as conexões associadas podem ser consideradas insignificantes e assim serem eliminadas (prunning). De modo inverso, valores substantivamente maiores que os outros poderiam indicar que houve over-training da rede.[1]

#### 2.4.5. Integração da rede

Finalmente, com a rede treinada e avaliada, ela pode ser integrada em um sistema do ambiente operacional da aplicação. Além disso, o sistema deve, periodicamente, monitorar sua performance e fazer a manutenção da rede quando for necessário ou indicar a necessidade de retreinamento[1].

## 2.5. Aplicações de Redes Neurais Artificiais

Aplicações de redes neurais são inúmeras. A primeira introdução pode ser vista em prognósticos de mercados financeiros. Grupos de investimento conhecidos utilizam redes neurais para analisar parte do mercado financeiro e fazer suas seleções. Outras aplicações bem sucedidas das técnicas de redes neurais artificiais são: controle de processos industriais, aplicações climáticas, e identificação de fraude de cartão de crédito. Um caso famoso foi o Mellon Bank que instalou um sistema de detecção de fraudes de cartão de crédito implementado com técnicas de redes neurais e os prejuízos evitados pelo novo sistema conseguiram cobrir os gastos de instalação em seis meses. Vários outros bancos começam a utilizar sistemas baseados em redes neurais para controlar fraudes de cartão de crédito. Estes sistemas têm a capacidade de reconhecer uso fraudulento com base nos padrões criados no passado com uma precisão melhor que em outros sistemas.[1]

Outro exemplo da utilização de redes neurais para melhoria na tomada de decisões é no diagnóstico médico. Em seu aprendizado, são submetidos uma série de diagnósticos de pacientes, de várias características, com vários sintomas e os resultados de seus testes. Também serão fornecidos os diagnósticos médicos para cada doença. Então quando forem apresentados os dados de um novo paciente, com seus sintomas, a rede fornecerá um diagnóstico para os novos casos. Isto essencialmente criará um sistema com o conhecimento de vários médicos, e fornecerá um diagnóstico inicial em tempo real à um médico. É importante mencionar que com

isso o que se pretende é implementar uma ferramenta de auxílio ao médico, e não um programa que o substitua. [1]

Outras aplicações[1]:

- análise e processamento de sinais;
- controle de processos;
- robótica;
- classificação de dados;
- reconhecimento de padrões em linhas de montagem ;
- filtros contra ruídos eletrônicos;
- análise de imagens;
- análise de voz;
- avaliação de crédito;
- análise de aroma e odor- um projeto que está em desenvolvimento, buscando a análise de odor via nariz eletrônico;
- análise e diagnóstico de descargas parciais pelo reconhecimento do padrão acústico- trata-se de uma tese de mestrado cujo objetivo é criar um sistema com capacidades de classificar o padrão acústico de uma descarga parcial ;

Programas utilizados

- Neuroimitator V3.1 para Windows 3.1 da Neuroma-RD Ltd: Ferramenta que permite à pesquisadores construir uma arquitetura arbitrária de rede neural e decidir parâmetros de neurônios envolvidos e conexões sinápticas.

- NeuroSolutions v5.0: Um ambiente de simulação orientado a objeto para Redes Neurais para estudos de sistemas distribuídos complexos difíceis de serem estudados apenas na teoria. Suas ferramentas de visualização permitem que o usuário verifique o comportamento da rede sem que seja necessário esperar até o fim de seu treinamento, além de permitir que parâmetros sejam alterados durante a execução da mesma.
- Matlab v4.0.6: Uma ferramenta que utiliza a interface gráfica para construir, desenhar, visualizar, implementar, e simular uma rede neural[2].
- Statistica: Ferramenta podesa cuja principal função é a seleção do INTELLIGENT PROBLEM SOLVERS, que usando métodos de inteligência artificial, permite resolução de problemas com altíssimo grau de exigência, envolvidos em análises avançadas de Redes Neurais, tais como selecionar a melhor arquitetura de rede, melhor subconjunto de variáveis, etc[3].

No caso, utilizou-se o programa NeuroSolutions, para desenvolver uma rede neural.

### **3.NEUROSOLUTIONS**

---

Este capítulo visa mostrar o funcionamento do programa, através de um exemplo prático, no caso, a liga AFNOR 7020, explicando sua utilização. São feitas também breves análises sobre o problema da utilização de um número mínimo de dados experimentais para a realização do experimento.

#### **3.1 A liga de Alumínio AFNOR 7020**

A liga de alumínio AFNOR 7020 de alta deformação é utilizada extensivamente em tanques de veículos de lançamento de satélites. Esse material é composto 4,5% de Zn, 1,2% de Mg, 0,4% de Fe, 0,25% de Mn, 0,3% de Si, 0,25 de Cr, 0,08% de Zr e balanceado com Al[5].

Devido à excêntrica aplicação e o manuseio adverso, é necessário conhecer perfeitamente suas características a altas temperaturas, tais como o comportamento constitutivo do material, para prever e projetar a tensão do material dada uma temperatura, uma deformação e uma taxa de deformação. A figura 3.1 mostra claramente que a combinação de três variáveis, no caso deformação, taxa de deformação e temperatura fornece uma tensão distinta[5].

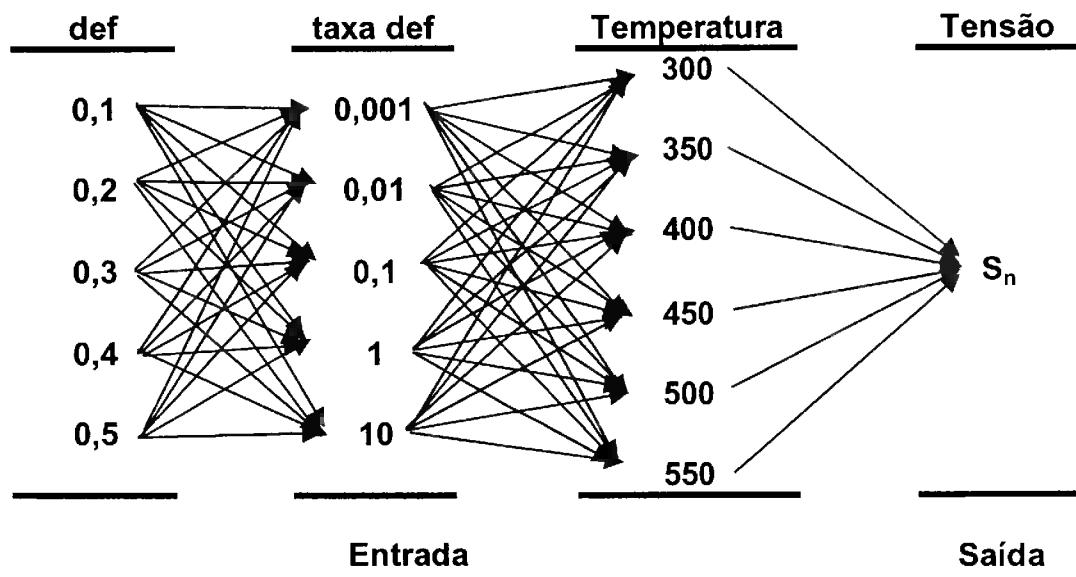


Figura 3.1 Combinação de dados

Assim, para a construção da rede neural, necessitou-se da utilização de dados experimentais na construção das camadas de entrada e saída da rede neural. A tabela 3.1 mostra todos os valores experimentais de tensão em função da taxa de deformação, deformação e temperatura do material em questão.

Tabela 3.1 Valores de Tensão da liga AFNOR 7020

def	taxa def	Temperatura					
		300	350	400	450	500	550
0,1	0,001	52	45,87	23,42	15,82	14,56	10,95
0,1	0,01	74,16	50,69	34,78	28,45	21,37	15,85
0,1	0,1	85,87	67,21	50,42	45	30,14	22,44
0,1	1	107,4	90,76	71,32	62,57	40,85	31,05
0,1	10	119,15	100,15	98,47	80,51	53,2	41,98
0,2	0,001	50,29	43,84	23,55	15,59	14,8	10,71
0,2	0,01	73,43	49,5	34,38	27,88	21,6	16,06
0,2	0,1	84,49	65,38	49,54	44,14	30,36	23,16
0,2	1	107,12	91,99	70,49	61,91	41,12	32,15
0,2	10	120,7	102,6	99	80,41	53,67	42,92
0,3	0,001	50,09	41,9	22,87	15,67	14,92	10,36
0,3	0,01	71,95	48,68	33,49	27,09	21,38	15,85
0,3	0,1	83,25	63,76	48,34	42,19	29,82	23,14
0,3	1	107,26	90,17	68,8	59,21	40,47	32,29
0,3	10	120,22	102,46	96,52	80,37	53,46	43,05
0,4	0,001	49,3	40,62	22,19	15,05	14,72	9,94
0,4	0,01	70,66	47,66	32,96	26,37	21,08	15,58
0,4	0,1	80,76	61,24	47,77	41,45	29,33	22,95
0,4	1	105,32	87,92	67,57	58,46	39,66	31,81
0,4	10	120,42	101,06	93,29	80,25	52,11	41,45
0,5	0,001	49,88	40,02	21,24	14,74	14,83	9,77
0,5	0,01	70,72	47,27	32,37	25,71	20,84	15,29
0,5	0,1	81,89	60,3	47,67	40,36	28,84	22,65
0,5	1	109,18	88,59	67,84	56,99	39,29	31,79
0,5	10	120,6	101,61	96,54	80	52,68	42,26

### 3.2 Utilização do programa

Dentro do Neurosolutions, existe um aplicativo chamado Neurosolutions for Excel, que será utilizado neste trabalho como principal ferramenta na obtenção de resultados. Esse aplicativo usa a interface do Microsoft Excel, criando um campo no menu chamado NeuroSolutions para que os dados sejam inputados, onde cada variável é posto em coluna, como mostra a figura 3.2

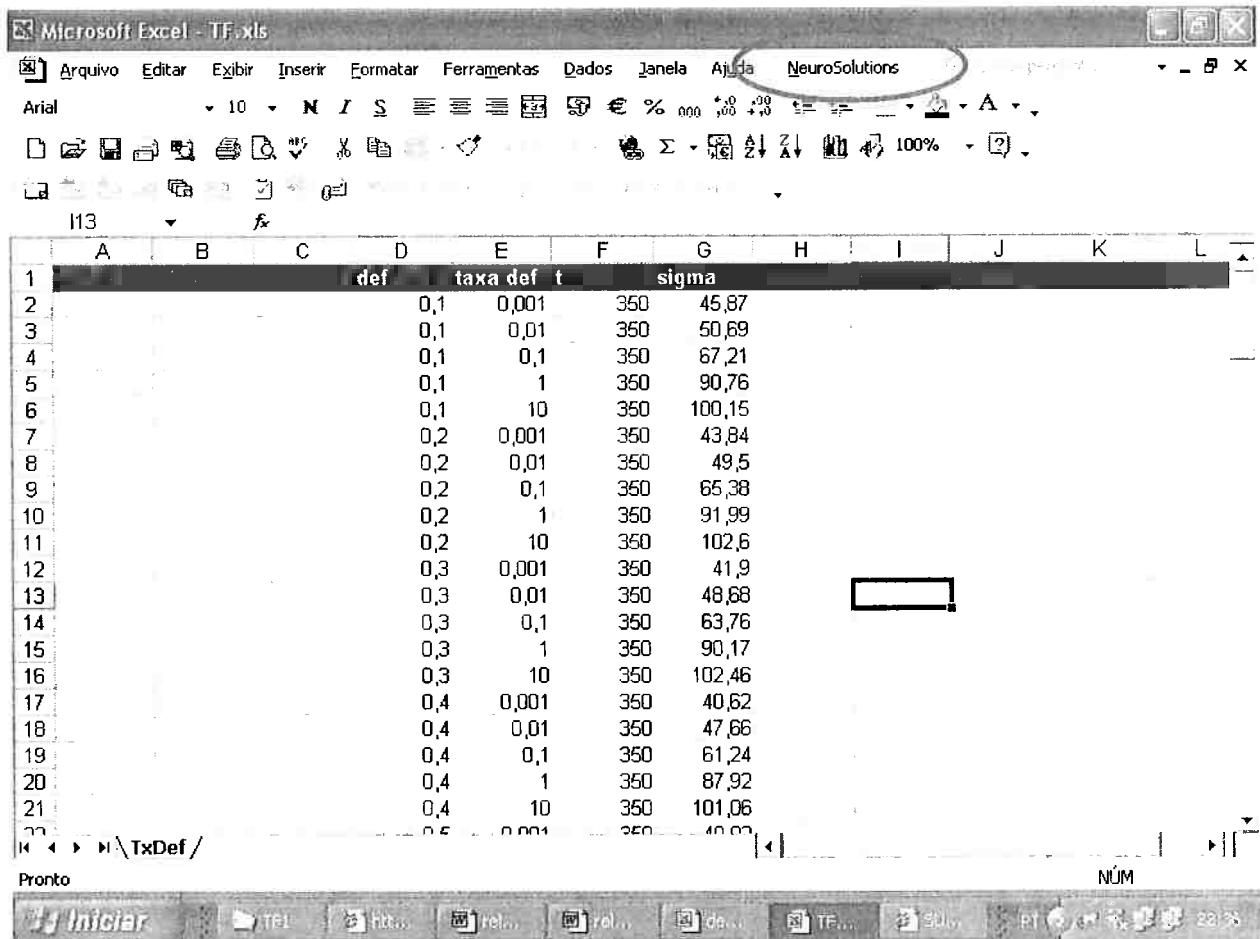


Figura 3.2 NeuroSolutions for Excel

Dentro do menu NeuroSolutions, selecionam-se dados de entrada (deformação, taxa de deformação e temperatura) e saída (tensão), respectivamente, através do caminho Neurosolutions – Tag Data – Column(s) as Input, Neurosolutions – Tag Data – Column(s) as Desired, iniciando a fase de coleta de dados e separação em conjuntos, como mostra as figura 3.3. Para que não criar uma rede “viciada”, os dados foram misturados aleatoriamente, através do caminho Neurosolutions – Preprocess Data – Randomize Rows.

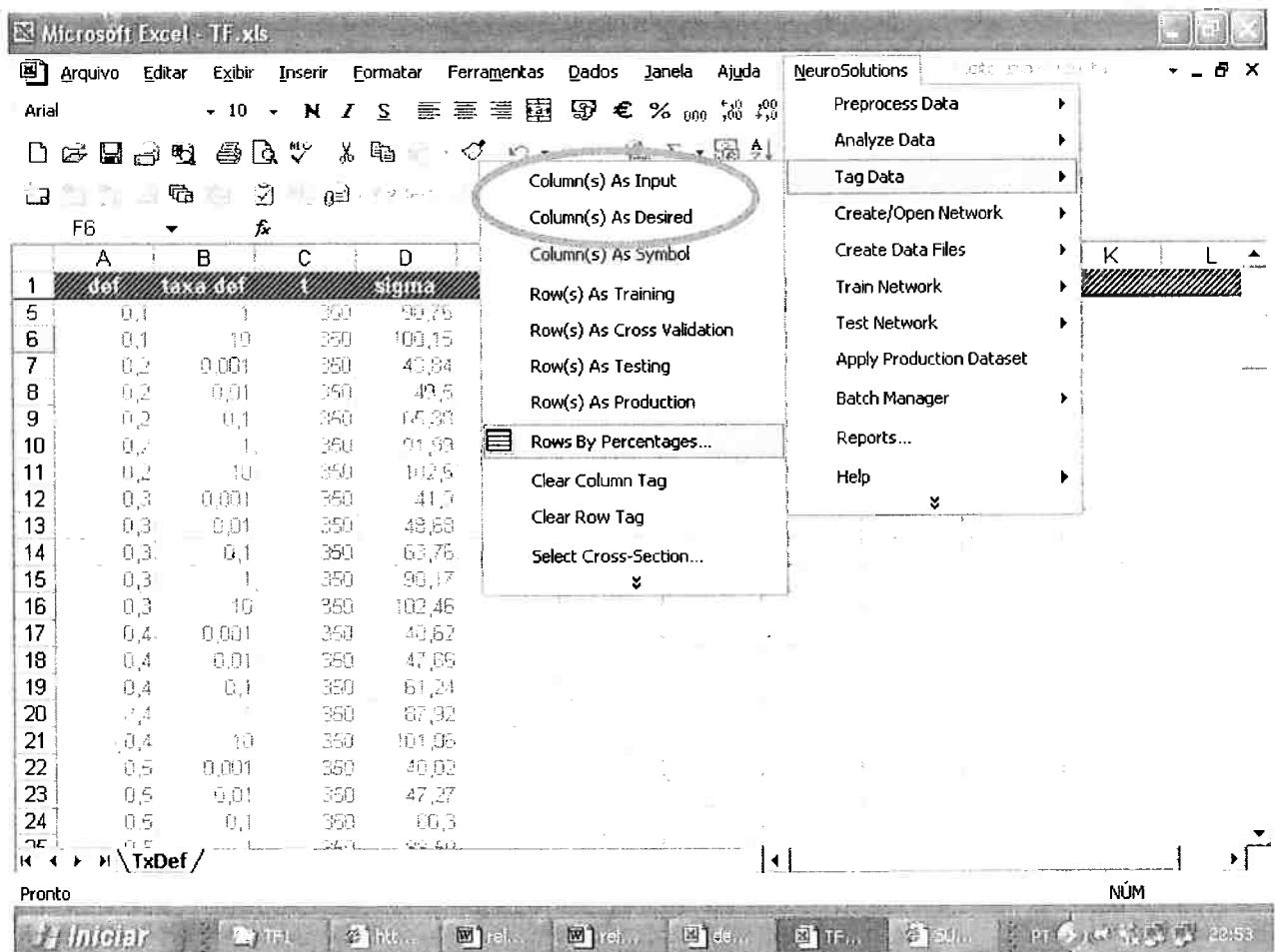


Figura 3.3 Comandos de inserção e seleção de dados

Assim, para separar os dados em conjuntos (treinamento, validação e teste), seleciona-se o Menu Neurosolutions – Tag Data – Rows By Percentages, dividindo percentualmente em conjuntos, no caso, 60% serão utilizados para treinar a rede, 20% para validação e 20% para teste como mostra a figura 3.4

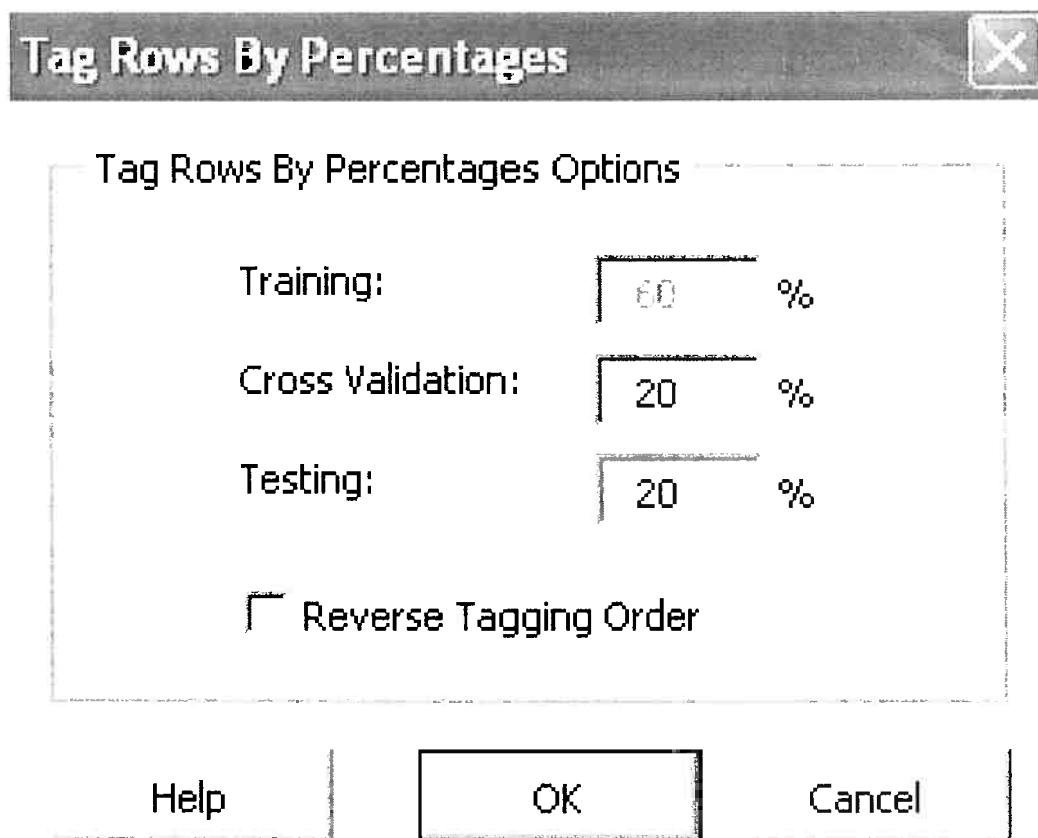


Figura .3.4 Divisão de dados

A figura 3.5 mostra que para uma visualização melhor, padronizaram-se os conjuntos tal que os dados de treino foram marcados em preto, os de validação em vermelho e os de teste em azul. Caso seja necessário, marcam-se alguns dados como dados de produção (no caminho NeuroSolutions – Tag Data – rows as Production marcados em verde), para realização da rede em tempo de execução, que será explicado mais adiante. Devido à limitação da tela, não foi possível mostrar todos os dados separados.

The screenshot shows a Microsoft Excel spreadsheet titled "Microsoft Excel - TF.xls". The data is organized into three distinct sections:

- Dados de Treino**: Located in the top section, spanning rows 83 to 95. It contains four columns of data: F, G, H, and I.
- Dados de Validação**: Located in the middle section, spanning rows 96 to 107. It also contains four columns of data: F, G, H, and I.
- Dados de Teste**: Located in the bottom section, spanning rows 108 to 124. It contains four columns of data: F, G, H, and I.

Arrows point from the labels to their respective data sections. The Excel ribbon at the top includes tabs for Arquivo, Editar, Exibir, Inserir, Formatar, Ferramentas, Dados, Janela, Ajuda, and NeuroSolutions. The status bar at the bottom shows "Pronto" and "NÚM".

Figura 3.5 Divisão de conjuntos

O passo seguinte é a configuração da rede. Através da opção "Create / Open Network" do menu "NeuroSolutions", inicia-se o processo de construção da rede neural. Neste caso trata-se de uma nova rede, na qual se deve selecionar a opção "New Custom Network". No caso, seleciona-se a rede conhecida, a Multilayer perceptron, a mais utilizada para resolver problemas complexos, conforme citada anteriormente, como mostra a figura 3.6.

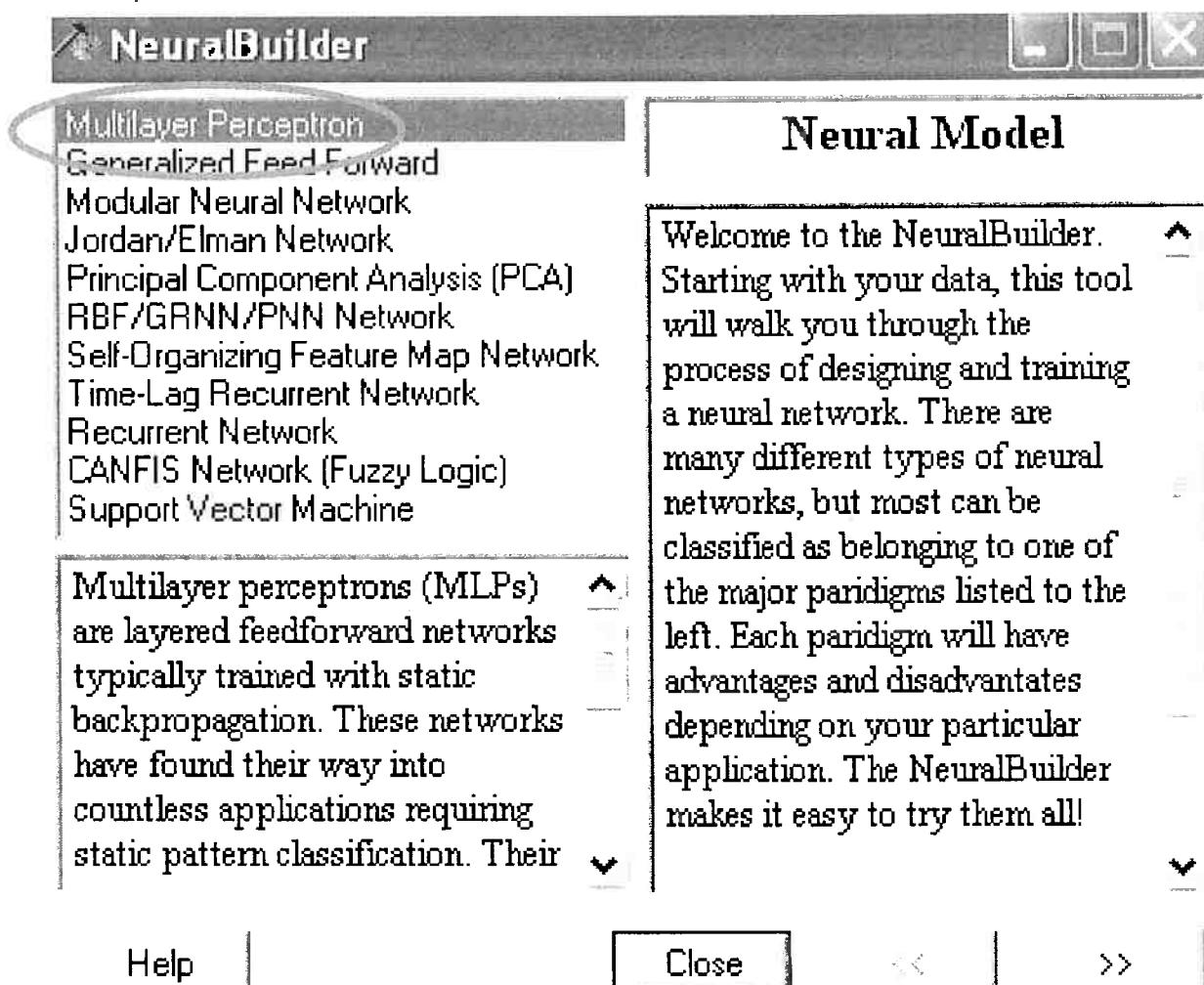


Figura 3.6 Seleção da rede

Após a escolha da rede, deve-se selecionar o número de camadas intermediárias a ser utilizado. No exemplo, escolheu-se usar quatro camadas como mostra a figura 3.7.

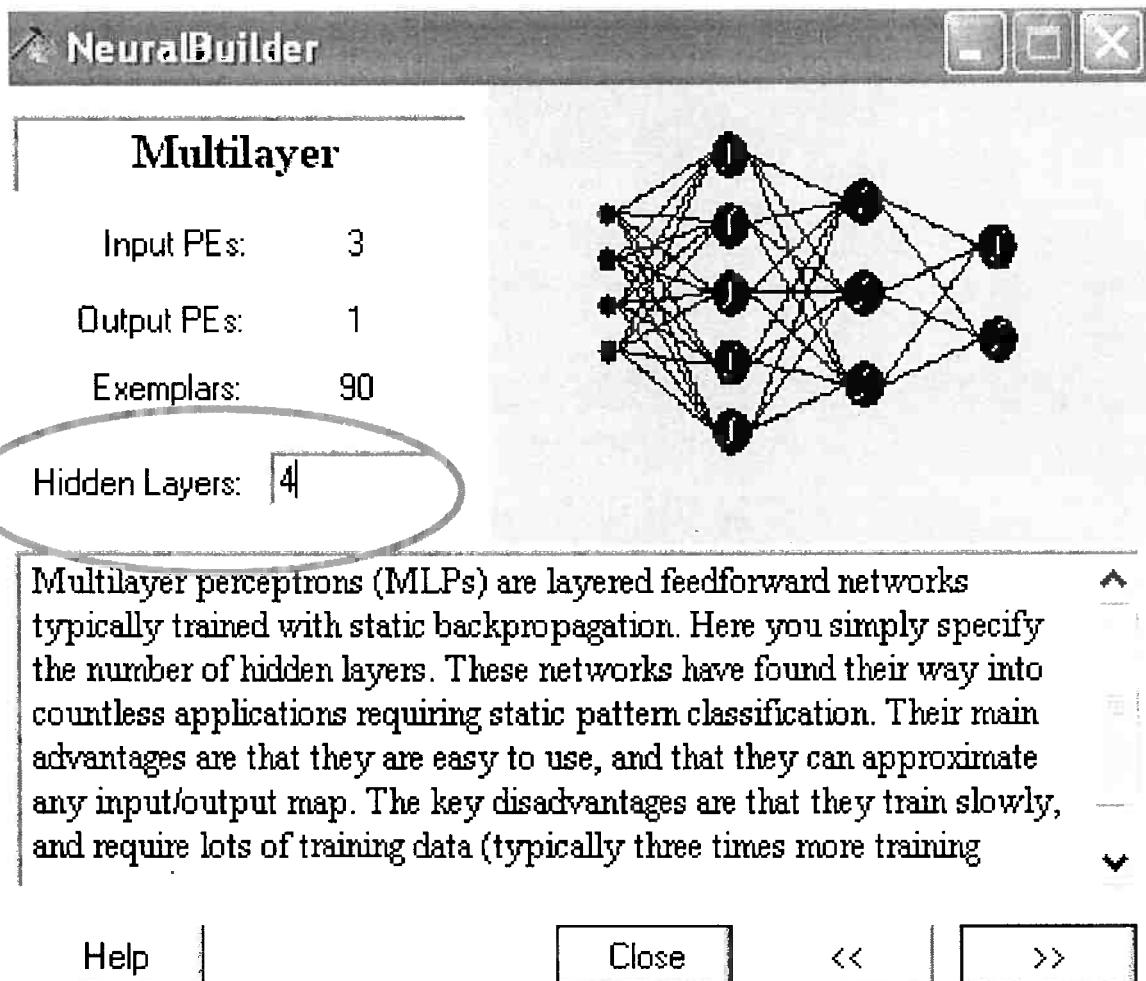


Figura 3.7 Seleção de camadas intermediárias

Observa-se que, durante a construção da rede, quanto maior o número de camadas utilizadas, menor é o desvio-padrão dos dados gerados pela rede treinada, consequentemente, a rede terá maior precisão, segundo as figuras 3.8 e 3.9.

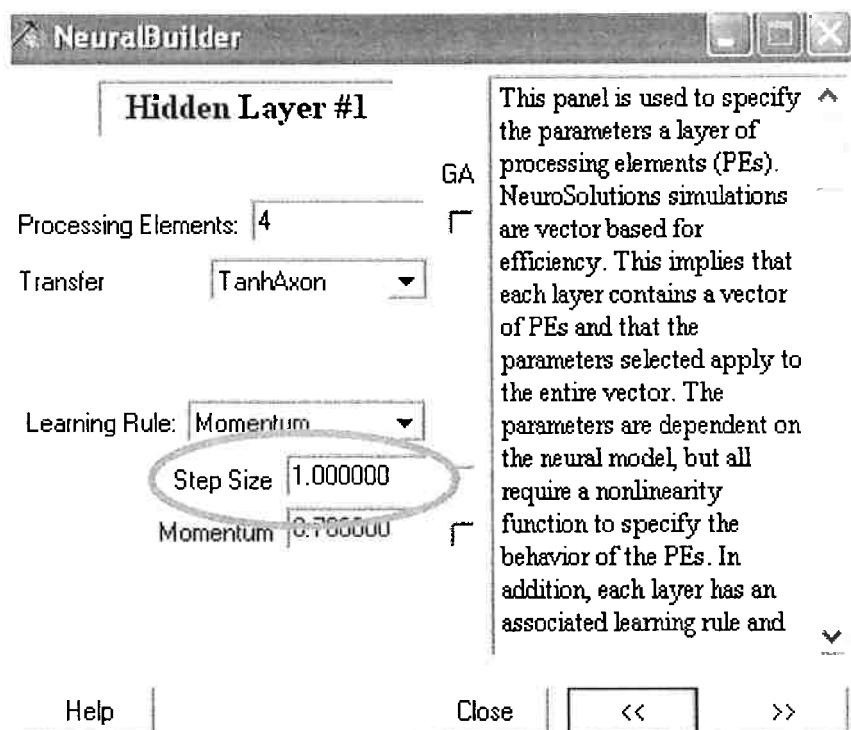


Figura 3.8 – Precisão das camadas intermediárias

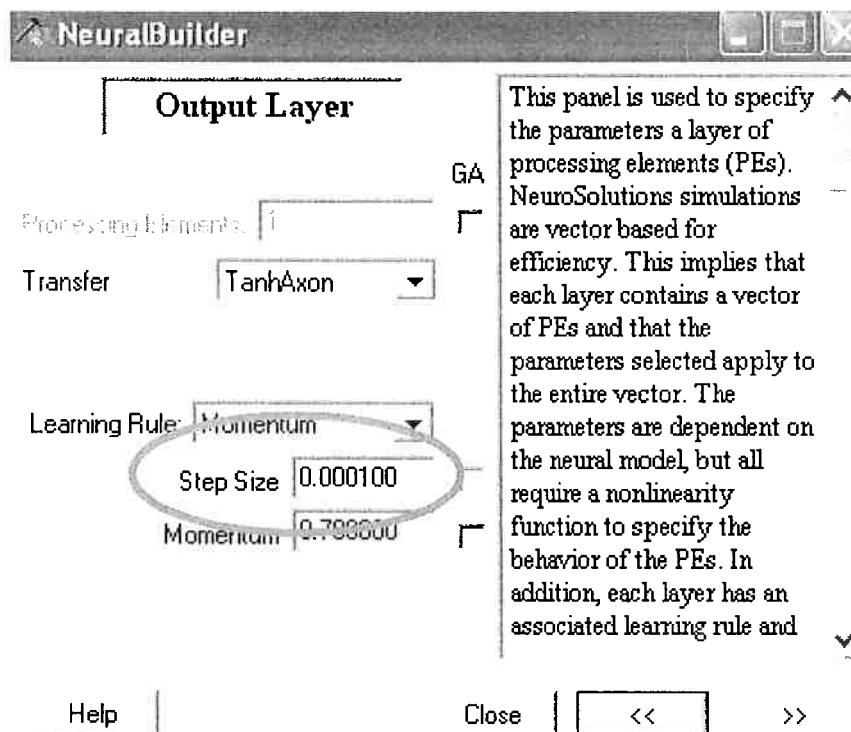


Figura 3.9 – Configuração da camada de saída

Após a configuração das camadas intermediárias e da camada de saída, verifica-se a rede construída através de uma imagem, onde as esferas alaranjadas representam as camadas intermediárias, como mostra a figura 3.10.

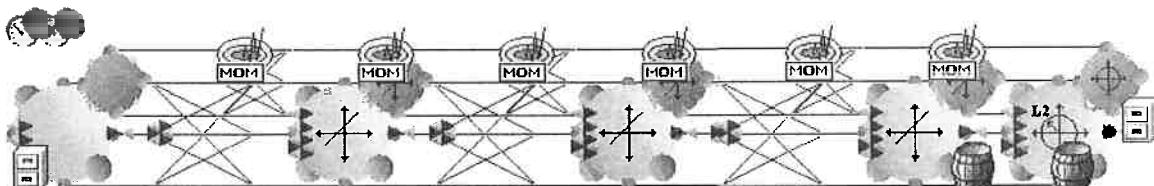


Figura 3.10 Rede construída

Após a fase de coleta de dados, separação em conjuntos e a configuração da rede, o próximo passo é treiná-la, a partir dos dados escolhidos, de tal forma que não haja um excesso de treinamento. Para isso, seleciona-se do menu o item NeuroSolutions – Train Networks – Train N Times. No caso, foi indicado um máximo de 1000 iterações em 3 treinos, respectivamente, inseridos nos campos Numbers of Epochs e numbers of Runs, como mostra a figura 3.11. Para que a rede seja válida, o item “Use Cross Validation” deve ser selecionado, a fim de criar o efeito comparativo da rede.

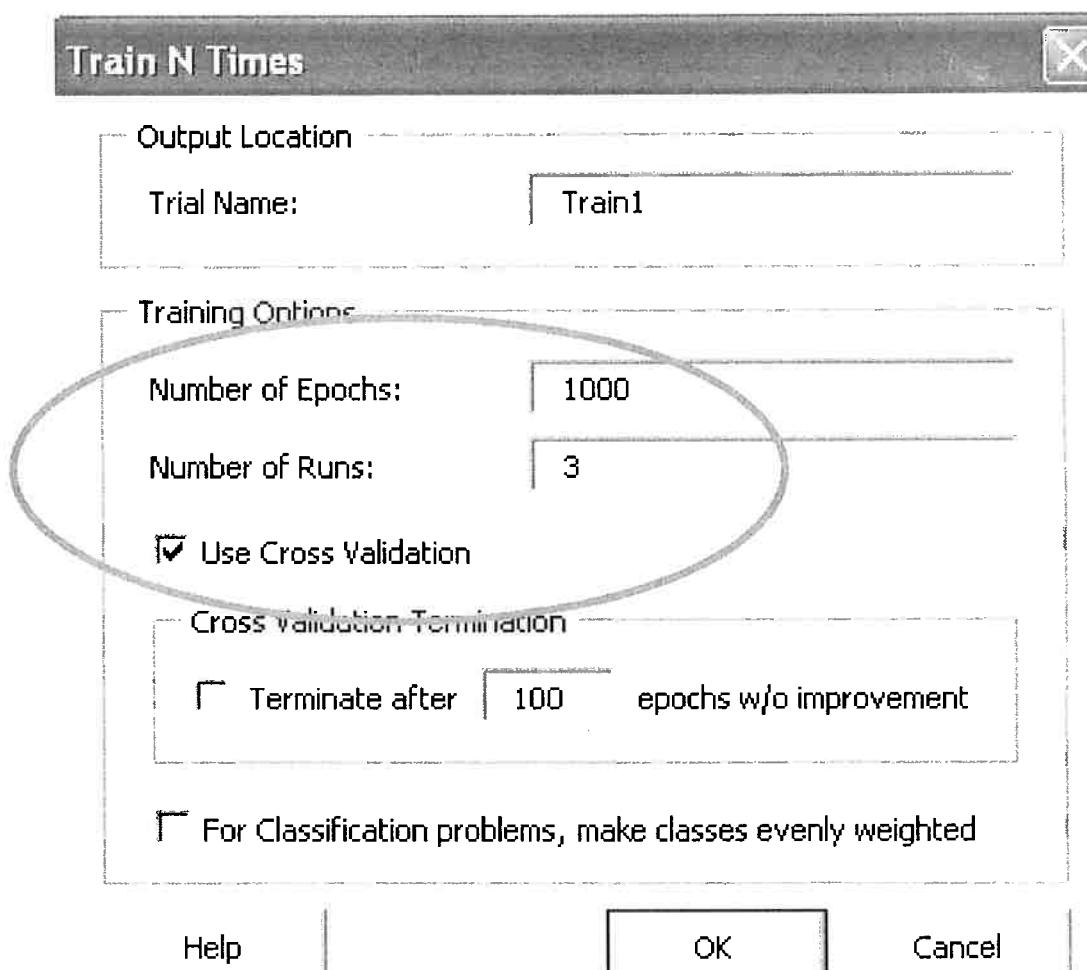


Figura 3.11 Seleção de iterações para treinamento da rede

Após o treino, analisou-se a rede treinada. Para que a rede seja válida, a curva de treinamento deve ter uma boa aderência na curva de validação, mostrando a efetividade da rede treinada. Dessa forma, sendo a curva de treino em linha contínua, e os limites superiores e inferiores dessa curva em linha pontilhada, marcados em azul e a curva de validação e seus limites, superior e inferior, em vermelho, na figura 3.12, observou-se a boa proximidade entre elas.

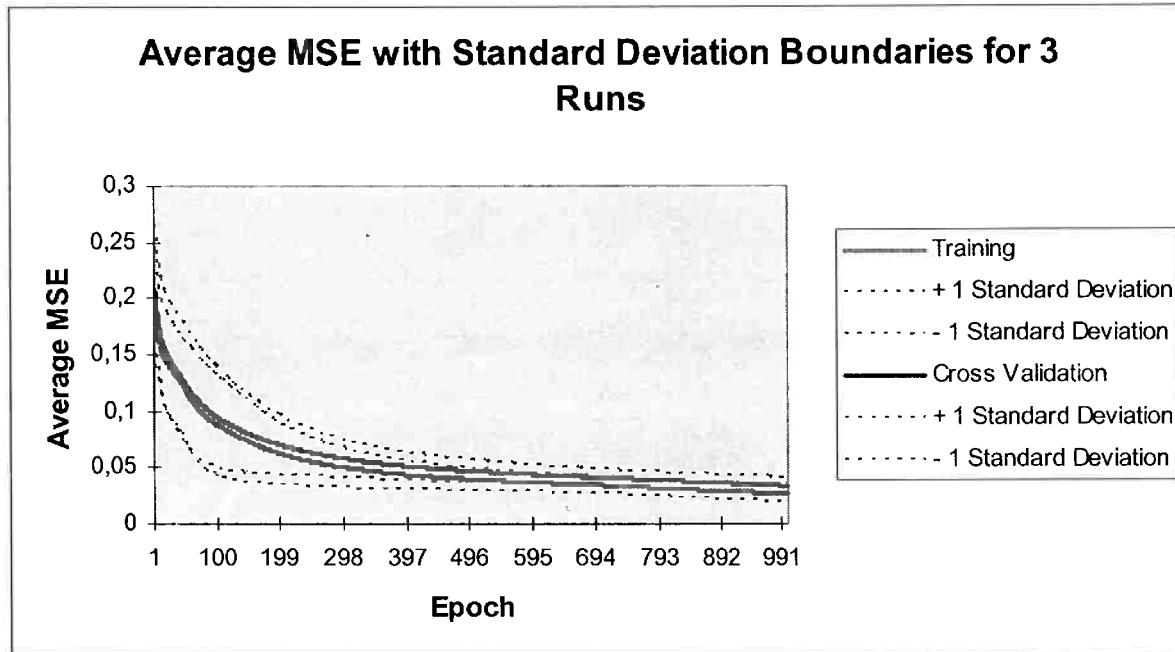


Figura 3.12 Curvas de treino e validação

No caso, não houve uma aderência perfeita entre as curvas, e sim uma proximidade entre elas, devido ao número relativamente pequeno de amostras fornecidas ao sistema. Assim, tanto a curva de treino, figura 3.13, quanto a de validação, figura 3.14, observou-se que na primeira rotina (Curva Run #1 em azul) divergiu das demais, colaborando para que a somatória das rotinas de treinamento e de validação não atingissem uma aderência adequada conforme esperado.

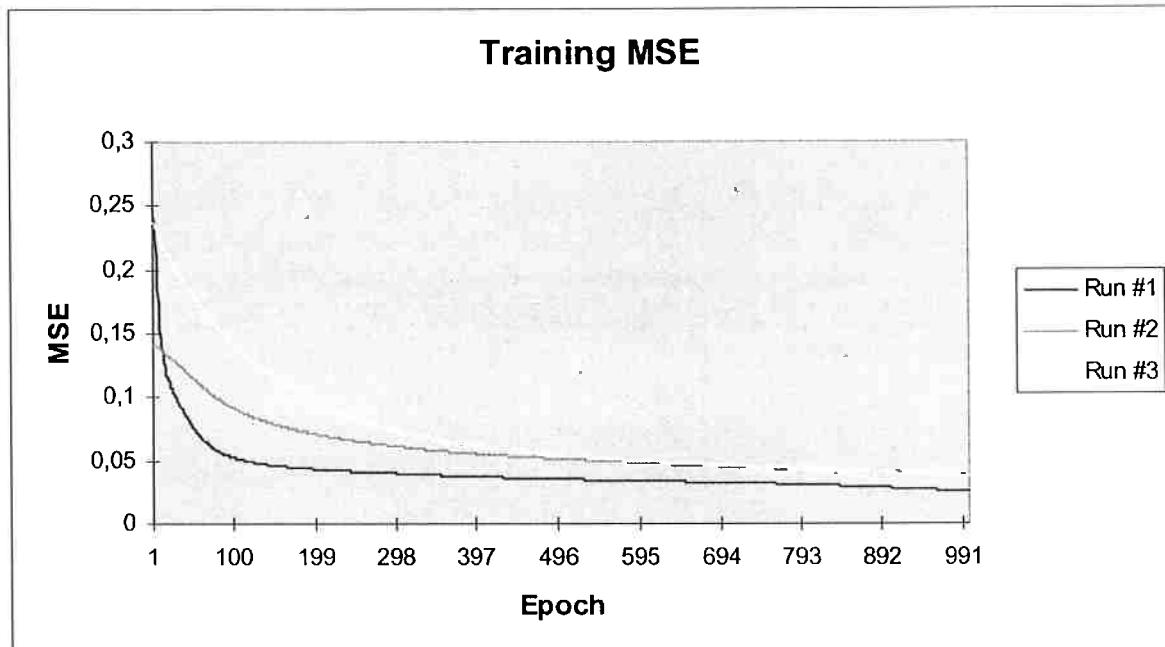


Figura 3.13 Curvas de treino

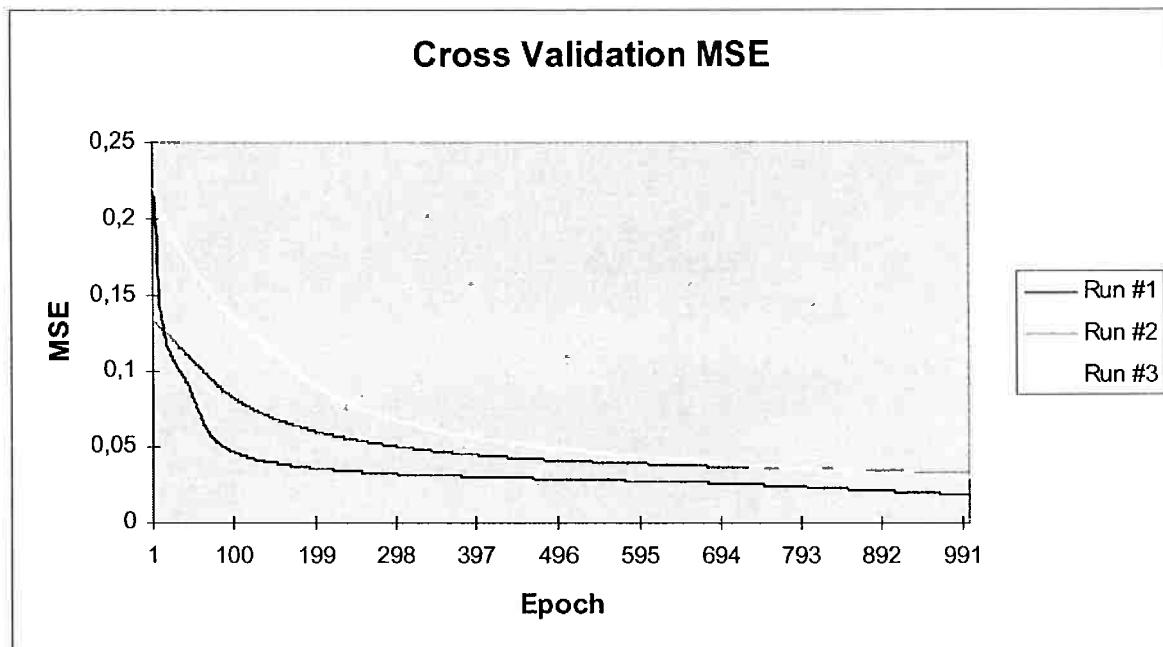


Figura 3.14 Curvas de validação

O passo seguinte é o teste da rede. Para testarmos a rede, é necessário selecionar no menu NeuroSolutions – Test Network – Test. No caso, figura 3.15,

selecionou-se o teste utilizando os melhores pesos para cada variável (Weights – Load Best) e o tipo Regressão Linear Múltipla (Report Type – Regression), cuja vantagem é obter uma relação matemática entre uma das variáveis (a variável dependente, Tensão) e o restante das variáveis que descrevem o sistema (variáveis independentes, deformação, temperatura e taxa de deformação)[4].

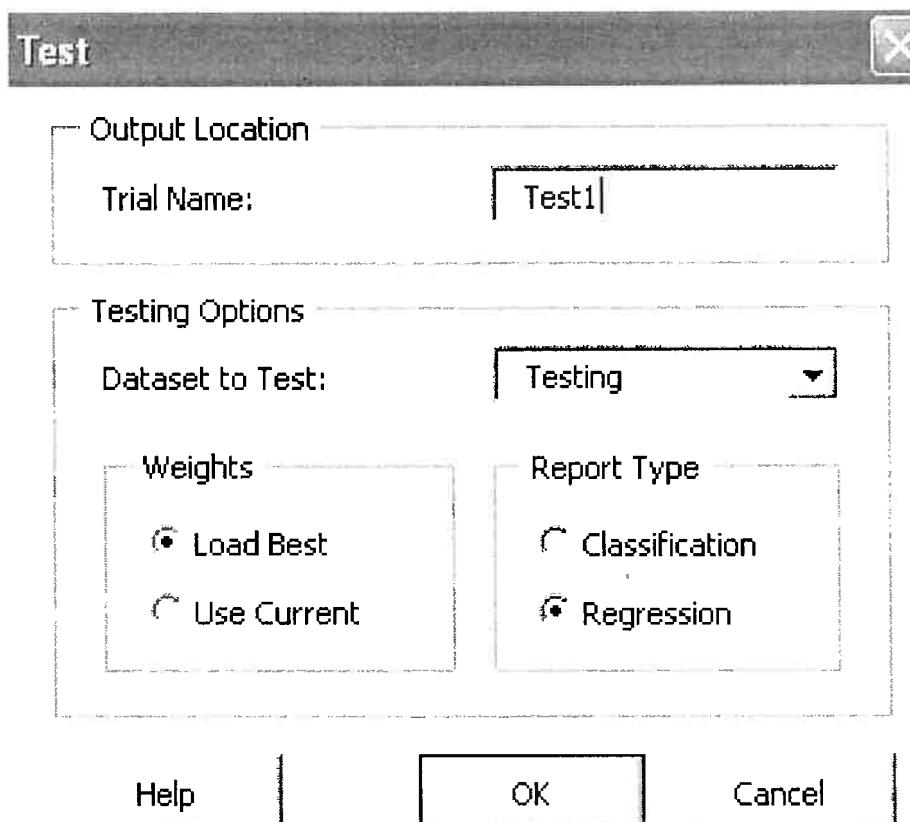


Figura 3.15 Seleção do teste

Dessa forma, comparam-se as curvas de saída, tanto da experimental (sigma) quanto da gerada pela rede (sigma Output), verificando a aderência relativamente boa, dado a falta de dados experimentais, como mostra a figura 3.16.

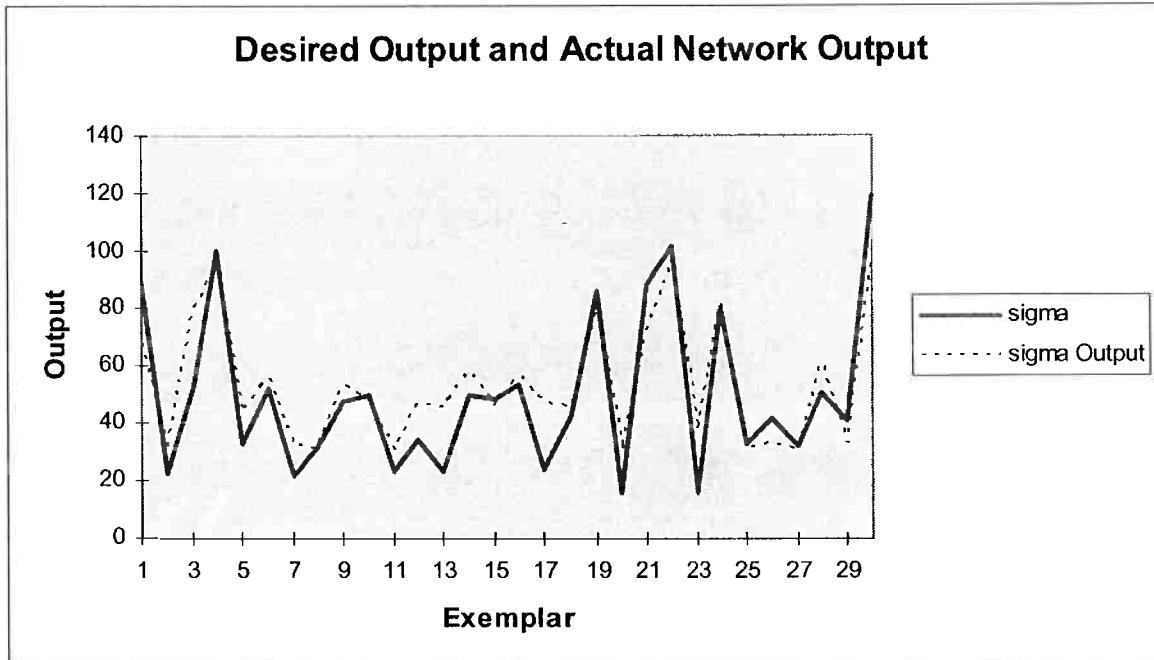


Figura 3.16 Curvas de saída

Na figura 3.17 pode-se verificar com uma precisão pontual a diferença entre a tensão experimental ( $\sigma$ ) e a tensão gerada pela rede ( $\sigma$  Output), tal que para alguns dados, o erro é alto, ultrapassando 50%. No gráfico 3.18, é possível observar esse efeito através da dispersão de pontos onde no eixo X é tensão teórica e no eixo Y, tensão gerada pela rede neural. Tal fato deve-se principalmente, a número relativamente baixo de amostras, conforme citado anteriormente.

def	taxa def	t	sigma	Sigma Output	Erro(%)
0,5	1	350	88,59	67,87	23,39
0,1	0,1	550	22,44	30,59	36,32
0,1	0,001	300	52	77,57	49,17
0,1	10	350	100,15	95,85	4,29
0,4	0,01	400	32,96	43,58	32,22
0,4	10	500	52,11	55,7	6,89
0,3	0,01	500	21,38	31,77	48,60
0,1	1	550	31,05	30,76	0,93
0,4	0,01	350	47,66	53,13	11,48
0,2	0,1	400	49,54	47,37	4,38
0,4	0,1	550	22,95	30,5	32,90
0,1	0,01	400	34,38	46,49	35,22
0,3	0,001	400	22,87	44,86	96,15
0,2	0,01	350	49,5	58,11	17,39
0,3	0,1	400	48,34	45,7	5,46
0,3	10	500	53,46	56,26	5,24
0,2	0,001	400	23,55	46,4	97,03
0,1	10	550	41,98	45,37	8,08
0,1	0,1	300	85,87	79,17	7,80
0,1	0,01	550	15,85	30,58	92,93
0,4	1	350	87,92	70,79	19,48
0,5	10	350	101,61	95,83	5,69
0,3	0,001	450	15,67	37,28	137,91
0,5	10	450	80	81,35	1,69
0,3	1	550	32,29	30,65	5,08
0,2	1	500	41,12	32,25	21,57
0,2	1	550	32,15	30,7	4,51
0,1	0,01	350	50,69	60,9	20,14
0,1	1	500	40,85	32,5	20,44
0,1	10	300	119,15	96	19,43

Figura 3.17 Rede Testada

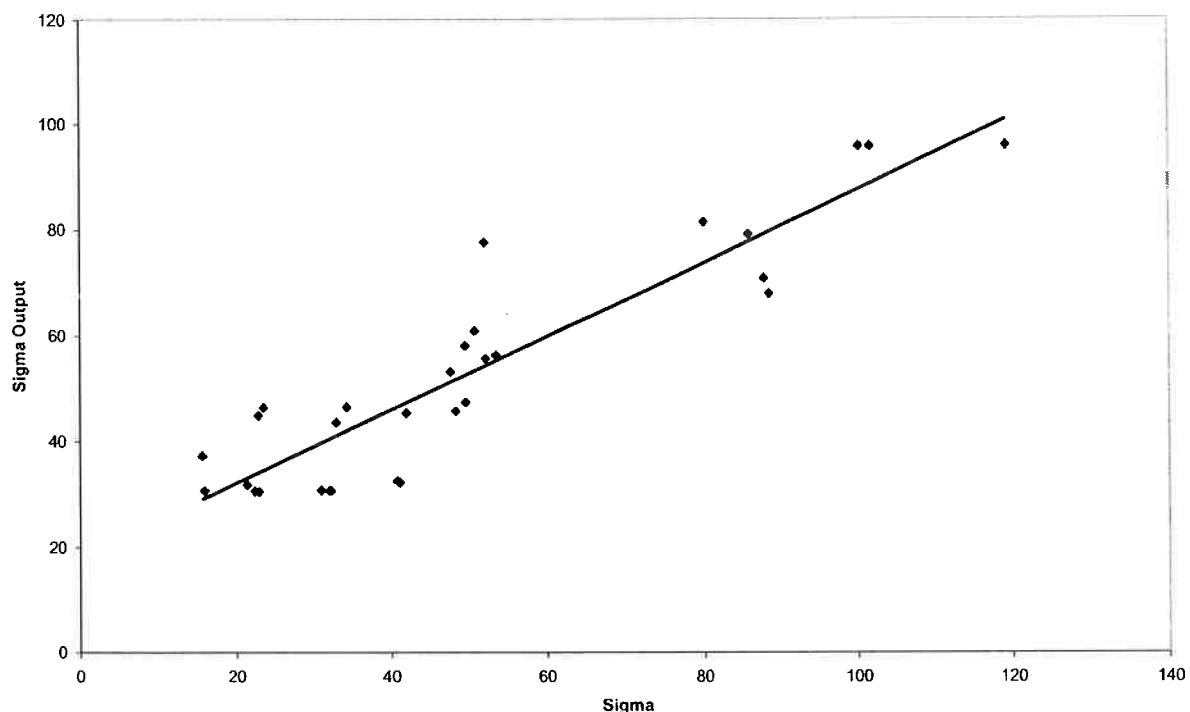


Figura 3.18 Tensão Teórica x Tensão gerada pela rede neural

Continuamente, realiza-se outro teste para verificar qual dos parâmetros tem maior influência no resultado final. Para isso, segue-se o caminho selecionando no menu neuroSolutions – Test Network – Sensitivity About the Mean. Assim, a figura 3.19 mostra que o parâmetro deformação, representado pelas letras def, tem maior influência sobre o resultado final, no caso a tensão gerada pela rede neural.

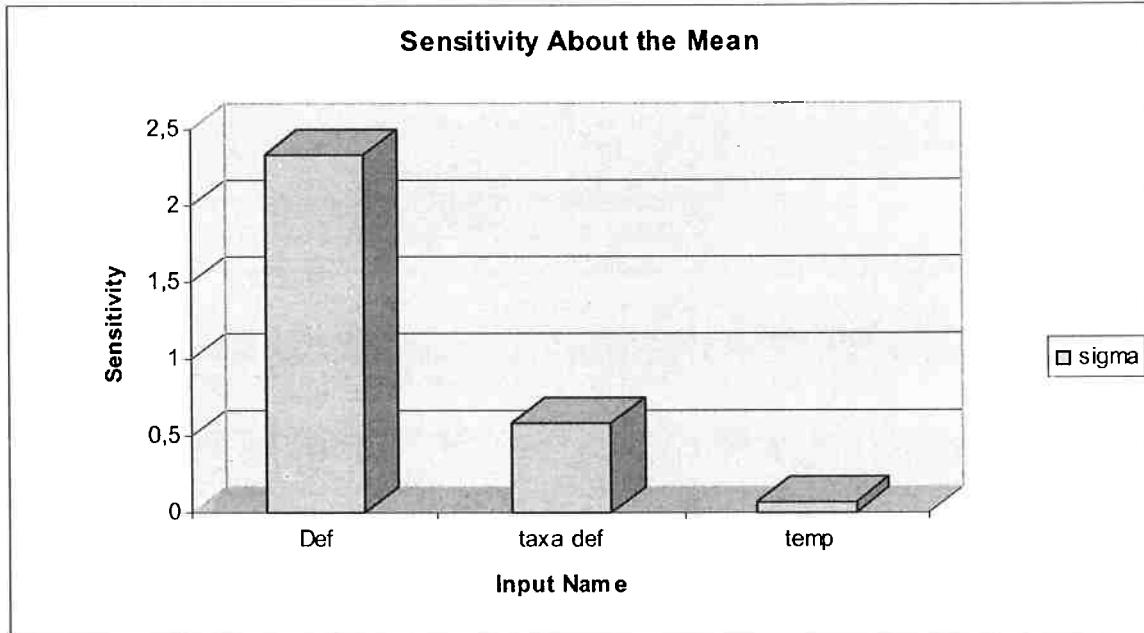


Figura 3.19 Teste da influência da variável

Depois de testar a rede, é possível realizar fase de integração da rede neural em um ambiente operacional, na qual monitora-se os dados de saída, gerando dados de saída em tempo de execução. Para isso segue-se o caminho Neurosolutions – Apply Production Dataset. Porém, a versão Shareware do programa NeuroSolutions não permite a utilização dessa função. Para isso, foi solicitado ao professor Ronald, que realizasse essa parte do experimento. A figura 3.20 mostra claramente o resultado teórico e o fornecido pela rede, mostrando que o problema do número de dados, citado anteriormente, também afeta o resultado final do experimento.

Def	taxa def	temp	sigma Production	sigma Teórico	Erro(%)
0,2	0,01	350	<b>51,18</b>	49,5	3,39
0,2	0,1	350	<b>53,68</b>	65,38	17,89
0,2	10	300	<b>99,70</b>	120,7	17,39
0,2	10	400	<b>97,68</b>	99	1,34
0,3	0,01	300	<b>65,23</b>	71,95	9,34
0,3	10	500	<b>54,05</b>	53,46	1,10
0,4	10	350	<b>99,73</b>	101,06	1,32
0,5	0,01	300	<b>69,35</b>	70,72	1,94

Figura 3.20 Dados de Produção em tempo de execução

## 4.CONCLUSÃO

---

---

Hoje, as redes neurais estão sendo aproveitadas em grande escala, tanto no ramo matemático como no ramo científico, isto é, podendo ser utilizadas em qualquer área, mostrando ser uma ferramenta poderosa no desenvolvimento de projetos que operam com inteligências artificiais.

Em termos didáticos, existem muitos programas construtores de redes neurais, no entanto, utilizou-se o programa *Neurosolutions*, pois apresenta aplicativos prontos que facilitam o desenvolvimento da rede. Dessa forma, pode-se conhecer e manusear o programa mostrando suas qualidades e importâncias profissionais.

Apesar da rede ter sido treinada com sucesso (boa aderência entre as curvas de treinamento e de validação), alguns resultados obtidos divergiram dos teóricos, mostrando o problema de se utilizar uma rede neural artificial que não comprehende ou classifica perfeitamente, devido ao número de amostras ser relativamente pequenas, pois para a montagem de uma boa rede neural com utilização em tempo de execução, necessita-se de mais de 1000 de amostras. No caso, utilizou-se 150 amostras, o que prejudica a precisão dos resultados. Esse efeito pode ser observado na figura 4.1, onde fixou-se a temperatura em 500°C e uma deformação de 0,3.

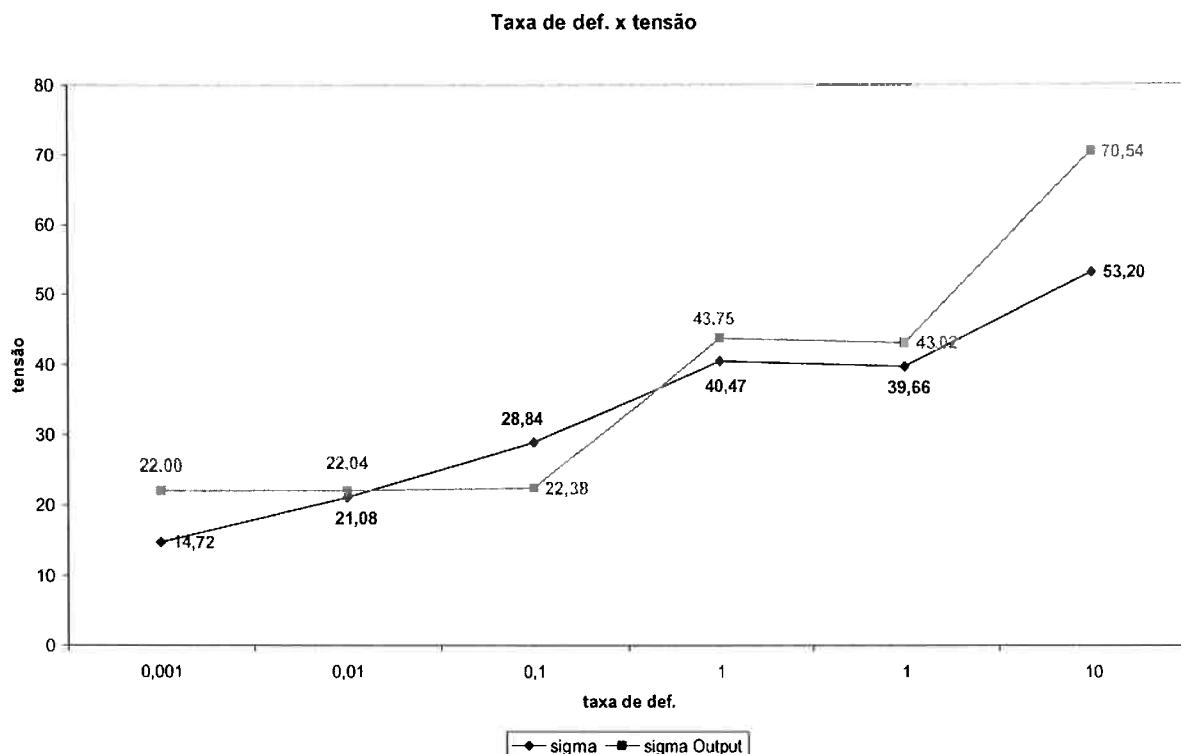


Figura 4.1 Taxa de deformação x Tensão para a T=500°C

Observando a utilização do programa para prever o comportamento constitutivo dos materiais, o modelo de rede neural pode ser usado como uma poderosa ferramenta didática e profissional para tal função.

## REFERÊNCIAS

- [1] Uma Introdução às Redes Neurais. Disponível em <<http://www.din.uem.br/ia/neurais>> Acesso em 10 set. 2005.
- [2] Business Intelligence. Disponível em <[http://www.fenas.com.br/produtos\\_statistica.htm](http://www.fenas.com.br/produtos_statistica.htm)> Acesso em 10 set. 2005.
- [3] Neural Network Toolbox for Matlab . Disponível em <<http://www.mathworks.com/products/neuralnet/>> Acesso em 10 set. 2005.
- [4] Estatística Multivariada. Disponível em <[http://www.criticanarede.com/cien\\_estatistica.html](http://www.criticanarede.com/cien_estatistica.html)> Acesso em 10 set. 2005.
- [5] Murty, S.V.S. Narayana; Rao, Nageswara B.; Kashyap, B.P. **Development and validation of processing map for AFNOR 7020 aluminium alloy.** Materials Science and Technology : June 2004, vol 20, 772