UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO

Dennis Kerr Coelho

SISTEMA NEURAL PARA PREVISÃO DE TEMPO DE PERFURAÇÃO DE POÇOS DE PETRÓLEO

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciências da Computação

Prof. Dr. Mauro Roisenberg
Orientador

Florianópolis, Julho 2005

Sistema neural para previsão de tempo de perfuração de poços de petróleo

Dennis Kerr Coelho

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação Área de Concentração - Sistema de Conhecimento e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

_____

Raul Sidnei Wazlawick, Dr
Coordenador do Curso

_____

Mauro Roisenberg, Dr.
Orientador

Banca Examinadora

_____

Dante Augusto Couto Barone,Dr.

_____

Paulo José de Freitas Filho,Dr.

_____

Silvia Modesto Nassar, Dr.ª

Ao meu pai e minha mãe Jony Coelho e Joan Kerr Coelho.
Que me propiciaram uma boa educação e tudo necessário
Para que pudesse me dedicar totalmente aos estudos.

AGRADECIMENTOS

Inicialmente agradeço a meus pais que sempre me incentivaram. Agradeço a meus avós e meus familiares. Agradeço a minha namorada que sempre me deu forças. E finalmente agradeço a meu orientador que me ajudou durante todo o mestrado e sem quem não teria realizado um trabalho tão completo.

SUMÁRIO

LISTA DE FIGURAS

LISTA DE TABELAS

RESUMO

Esta dissertação tem como objetivo mostrar como a abordagem conexionista pode ser utilizada na avaliação e previsão do tempo total em operações de perfuração e completação de poços de petróleo em águas profundas. Os valores dos parâmetros utilizados para estimar o tempo total gasto da operação realizada no poço foram retirados de um banco de dados históricos de uma companhia petrolífera. As correlações e as características destes parâmetros foram detectadas utilizando-se de uma rede neural competitiva conectada a uma rede neural direta que foi treinada para estimar a média, o desvio padrão e o tempo total gasto na operação realizada no poço. São apresentados os experimentos realizados para validação do modelo e os resultados são utilizados para avaliar o desempenho e validade da proposta. Uma das vantagens da metodologia proposta, está no fato de ser uma ferramenta simples e prática para obtenção de uma estimativa do tempo total de uma operação realizada sobre um poço de petróleo baseado em parâmetros geométricos e tecnológicos, sem a necessidade de especificar todas as sub-operações de perfuração e completação como acontece nos métodos tradicionais de analise de risco.

Palavras-chave: Redes Neurais Artificiais, Análise de Risco e Reconhecimento de Padrões.

ABSTRACT

This dissertation's objective is to demonstrate how the connectionist approach can be used in the evaluation and prediction of the total time spent in drilling and completion operations of oil wells in deep waters. The parameter values utilized to estimate the total time spent in oil well operations were taken from an oil company's historical data bank. The correlation and characteristics of these parameters were detected using a competitive neural network connected to a feedforward neural network that was trained to estimate the average values, the standard deviation and the total time spent in the realization of the oil well operation.

The experiments created to validate the model are presented, and the results are utilized to evaluate the performance and validity of the proposition. One of the advantages of the proposed methodology is the fact that it's a simple and practical tool. It can be used to estimate the total time spent in an oil well operation, based on geometric and technological parameters, without the need to specify all the drilling and completion sub-operations, unlike the traditional methods of risk analysys.

Key Words: Neural Networks, Risk Analisis, Pattern Recognition.

Capítulo 1

1.1 Introdução

Apesar do petróleo ser a maior fonte de energia utilizada atualmente pelo homem, esta é uma fonte energética não-renovável, cujas reservas estimadas podem se esgotar em aproximadamente 4 ou 5 décadas. Ainda assim, a exploração petrolífera é uma atividade econômica na qual são investidos bilhões de dólares todo ano e onde estão envolvidos complexos problemas de decisão, de conhecimento, de risco, de incertezas e onde se busca atingir o objetivo de encontrar mais petróleo (SILVA, 2000). Segundo estudos publicados na internet (ALEKLETT & CAMPBELL, 2005) pela *ASPO News (The Association for the Study of Peak Oil),* as atividades de exploração ainda requisitarão significativas melhorias científicas, tecnológicas e gerenciais pelo menos durante o primeiro quarto do século XXI.

O Brasil está buscando a auto-suficiência na produção de petróleo, entretanto, o pico na produção brasileira de petróleo em terra foi atingido em 1997. Hoje em dia a maior parte das novas reservas brasileiras de petróleo estão localizadas em águas profundas, o que leva o país a investir em novas tecnologias e modelos para exploração e produção de petróleo. (SILVA, 2000).

A exploração de petróleo em águas profundas é uma tarefa complexa e sujeita a um grande número de falhas, pois acontece num ambiente extremamente hostil onde qualquer pequeno erro pode causar um grande problema. No processo de perfuração de poços de petróleo nestas condições são utilizadas as mais modernas tecnologias de perfuração de poços. Toda essa tecnologia tem um custo elevado, que aumenta ainda mais com as falhas ocorridas durante o processo. Só para se ter uma idéia, segundo THOMAS (1996), o aluguel de uma sonda de perfuração custa em torno de US$ 180.000 / dia. Esse custo alto do aluguel da sonda faz com que as operações envolvidas no processo de perfuração e exploração de petróleo sejam extensamente planejadas e simuladas antes de sua execução.

A previsão do tempo gasto com operações de perfuração e completação (seqüência de tarefas realizadas sobre um poço visando prepará-lo para a produção) de poços de petróleo ou gás está sujeita a uma série de incertezas e fatores de risco. Este risco está associado ao conhecimento limitado disponível sobre as características

geológicas da formação, das dificuldades técnicas e dos comportamentos imprevistos de operadores humanos (JACINTO, 2002). O planejamento e a análise de risco destas atividades são influenciados por eventos inesperados, tais como: “kick” (perda de controle do fluído de perfuração) e perda de circulação ou colapso do poço. Estes eventos levam a perdas de tempo, aumento de custos, declínio da produção ou até mesmo a perda do poço.

Técnicas de análise de risco e gerenciamento de operações de exploração petrolífera estão crescendo a nível mundial e várias companhias internacionais têm melhorado seu desempenho utilizando técnicas de análise de risco combinadas a novas tecnologias de análise de dados. Entre os estudiosos desta especificidade cita-se HARBAUGH et all (1995) e ROSE (2001).

Em um poço de petróleo são realizados vários tipos de intervenções: perfuração exploratória, completação, restauração etc. Essas intervenções são compostas por operações do tipo: Canhoneio, Instalação de BOP, Abandono de poço, Retirada de BOP etc. Atualmente a maior parte das técnicas de análise e previsão de custo (tempo) são baseadas em simulações numéricas das operações envolvidas no processo de perfuração e completação. Neste tipo de sistema, todas as operações envolvidas no processo de perfuração e completação são simuladas numericamente. O usuário do sistema necessita entrar com todas as operações que serão realizadas, bem como as “funções de tempo” (funções de distribuição probabilística do tempo gasto na operação). As “funções de tempo” são funções de probabilidade que no momento da simulação, retornam o valor de tempo para a dada operação estando baseadas numa distribuição de probabilidades.

O método tradicional possui algumas características que muitas vezes o impossibilita de ser utilizado numa análise rápida. Entre estas, ressalta-se a necessidade de conhecer antecipadamente todas as operações envolvidas na intervenção a ser simulada e, além disso, deve-se conhecer a distribuição de probabilidade do tempo de todas as operações.

O Sistema proposto nesta dissertação apresenta uma abordagem alternativa e simplificada para a estimativa do tempo total de intervenções de perfuração e completação de poços de petróleo, baseada em dados históricos. Esse Sistema tem como objetivo calcular o tempo de uma intervenção utilizando-se de uma extensa base de dados pertencente à companhia petrolífera. Para isso, o Sistema utiliza uma arquitetura

neural, arquitetura essa utilizada no processo de generalização dos dados históricos. A arquitetura adotada no Sistema possibilita que um cenário (uma bacia, um campo de petróleo ou mesmo todos os campos de uma empresa) seja apresentado ao Sistema e utilizados em seu treinamento. Tendo treinado o Sistema, a avaliação de uma intervenção pertencente ao cenário treinado ou a um cenário similar é uma tarefa simples e rápida que não exige um planejamento das operações.

1.2 Objetivos

1.2.1 Objetivo Geral

Propor uma arquitetura neural para extração de conhecimento histórico de bases de dados e sua utilização para possíveis previsões.

1.2.2 Objetivos Específicos

a) Estudar técnicas de descoberta de conhecimento em bases de dados e de reconhecimento de padrões;

b) Estudar métodos para utilização de redes neurais em aplicações de auxilio a tomada de decisão;

c) Estudar os processos envolvidos na perfuração e completação de poços de petróleo;

d) Desenvolver uma técnica que possibilite a previsão dos tempos de intervenções baseados em dados históricos e represente esta previsão como uma distribuição de probabilidades associadas à incerteza da previsão;

e) Avaliar os resultados obtidos da utilização da técnica proposta.

1.3 Metodologia

Tendo definido o problema, parte-se para a definição da metodologia a ser utilizada. Inicialmente foi feito um trabalho de pesquisa dentro do escopo do problema, esta pesquisa foi realizada buscando-se, na literatura, fontes que descrevessem as

técnicas de descoberta de conhecimento em bases de dados, Reconhecimento de padrões, Perfuração de poços de petróleo, Técnicas de análise de risco e Redes Neurais Artificiais.

Após esta pesquisa, iniciou-se a prototipação de possíveis arquiteturas, sempre buscando uma arquitetura que melhor se adaptasse aos requisitos do Sistema.

Ao definir-se a arquitetura ideal, iniciou-se a fase de projeto onde foram tratados alguns detalhes que influenciavam a precisão do Sistema. Com o projeto pronto foi realizada a fase de implementação.

Em seguida o Sistema passou por uma fase de testes objetivando a validação do modelo. Para isto foi utilizada uma fração da base de dados, fração esta não utilizada nos treinamentos. Esse conjunto de testes foi apresentado ao Sistema e seus resultados foram comparados com os valores reais sendo retiradas assim as conclusões sobre a precisão e confiabilidade do sistema.

1.4 Estrutura do Texto

No capítulo 2 trata-se da base teórica sobre extração de conhecimentos de bases de dados, reconhecimento de padrões e redes neurais.

No capítulo 3 é dada uma breve explicação sobre os processos envolvidos na perfuração e completação de poços de petróleo.

No capítulo 4 são descritos detalhadamente todo o processo envolvido na elaboração da proposta adotada e no projeto e na implementação do sistema.

No capítulo 5 são descritos os testes de validação realizados com o sistema, além do modelo proposto e seu grau de confiabilidade.

No capítulo 6 são apresentadas as conclusões ontidas do sistema e do modelo, bem como as sugestões para possíveis melhorias e trabalhos futuros.

CAPÍTULO 2 – REDES NEURAIS, MINERAÇÃO DE DADOS E RECONHECIMENTO DE PADRÕES

No processo de análise do problema, procurou-se um conjunto de metodologias e técnicas que, ao ser implementado em um sistema, fosse capaz de fazer uma previsão válida para tempos de intervenções envolvidas no processo de perfuração de poços de petróleo. Esta análise demonstrou estar-se diante de um problema de Descoberta de Conhecimento em Base de Dados (KDD – *Knowledge Discovery Database*), pois esta previsão deveria levar em conta aspectos gerais da operação, tais como fatores geológicos, tecnológicos e humanos, e ser baseada em dados históricos, ao invés do método tradicional de estimativa que envolve a simulação da seqüência das operações de engenharia de poço que devem ser realizadas no processo de perfuração e completação do poço. Estes atributos estão geralmente presentes em bases de dados das grandes companhias de petróleo, entretanto, uma análise mais profunda mostrou que esta aplicação vai além das principais tarefas normalmente associadas às aplicações de KDD, tais como, associação, classificação ou agrupamento de variáveis do problema, envolvendo também a Mineração de Dados e o Reconhecimento de Padrões existentes na base histórica e um processo de previsão ou estimativa de tempos de perfuração de novos poços.

2.1 Descoberta de Conhecimento em Bases de Dados

O avanço de tecnologia tornou acessíveis ferramentas que possibilitam o acúmulo de grande quantidade de dados. A conseqüência é a ampliação do uso dos *Data Warehouses*, grandes repositórios de dados, agregados de forma organizada e eficiente, e em geral, de natureza histórica. Essa grande quantidade de dados armazenados está cada vez sendo mais valorizada. Muitas empresas utilizam-se de profissionais especializados para vasculhar suas *Data Warehouses* em busca de padrões e tendências.

Entretanto, a análise dos dados de uma Data Warehouses por um especialista é um processo demorado, dispendioso e sujeito a erros. Assim sendo, torna-se necessária uma forma de análise, interpretação e aquisição de dados automatizada. Motivadas por essa tendência, as técnicas de Descoberta de Conhecimento em Bases de Dados estão tornando-se cada vez mais refinadas, complexas e indispensáveis (DATA

WAREHOUSE, 2000). Estas técnicas normalmente servem para auxiliar o especialista no cumprimento de suas tarefas.

2.2 O Processo KDD

O KDD é na verdade a aplicação de um conjunto de técnicas com o objetivo de extrair conhecimentos de uma base de dados. Enquanto um cérebro humano, comprovadamente, consegue fazer até oito comparações ao mesmo tempo (DATA WAREHOUSE, 2000), um sistema de KDD consegue fazer milhares de comparações e encontrar padrões imperceptíveis a um ser humano. O KDD é basicamente a aplicação de técnicas estatísticas, matemáticas e de IA para a extração de conhecimento de uma base de dados. Segundo DATA WAREHOUSE (2000), usualmente um processo de KDD segue alguns passos:

> a) **Limpeza dos dados**: para remover ruídos e inconsistência nos dados;
> b) **Integração de dados**: onde múltiplas fontes de dados podem ser combinadas;
> c) **Transformação nos dados**: onde dados são transformados em padrões mais simples de serem trabalhados;
> d) **Mineração nos dados**: métodos (estatísticos, de IA,....) são aplicados para extrair padrões dos dados;
> e) **Avaliação de padrões**: medidas que avaliam o quão interessante é cada padrão, ou seja, o quão relevante é o padrão em relação aos dados avaliados e
> f) **Apresentação do conhecimento**: técnicas de visualização e de representação do conhecimento.

2.3 Mineração de Dados

A Mineração de Dados (Data Mining) é considerada por muitos autores como a fase mais importante do processo de KDD. É nesta fase que geralmente é feita a descoberta de conhecimento ou de padrões, para a solução do problema.

A mineração de dados nada mais é do que um processo de reconhecimento de padrões existentes em uma base de dados. No tópico 2.4 tratara-se das técnicas utilizadas para reconhecimento de padrões.

2.4 Reconhecimento de Padrões

SCHALKOFF (1992), apresenta que as

> Técnicas de Reconhecimento de Padrões (RP) tem sido uma importante componente de sistemas inteligentes e usada em muitos sistemas de pré-processamento de dados e tomadas de decisão. Simplificando, reconhecimento de padrões (RP) é uma ciência que se concentra na descrição ou classificação (reconhecimento) de medidas.

Assim sendo, técnicas de reconhecimento de padrões podem se utilizadas como técnicas de pré-processamento de dados em sistemas de auxilio para tomada de decisão. Também podem ser utilizadas para o agrupamento e classificação de grandes quantidades de dados.

De acordo com SCHALKOFF (1992), existem três abordagens correlatas para reconhecimento de padrões:

a) Abordagem Estatística de Reconhecimento de Padrões.
b) Abordagem Sintática de Reconhecimento de Padrões.
c) Abordagem Neural de Reconhecimento de Padrões.

2.4.1 Abordagem Estatística de Reconhecimento de Padrões (AERP)

Uma abordagem estatística para reconhecimento de padrões assume uma base estatística, ou seja, algoritmos matemáticos normalmente baseados para classificação. Esta base estatística é utilizada na obtenção de um conjunto de características mensuráveis, que são extraídas dos dados de entrada, e são utilizados na criação de vetores de características para uma de cada *n* classes (SCHALKOFF, 1992).

Duas das técnicas que seguem uma abordagem estatística para reconhecimento de padrões são as técnicas de Análise de *Cluster* e Regressão. Essas técnicas são descritas nos tópicos 2.4.1.1 e 2.4.1.2.

2.4.1.1 Análise de *Cluster*

Segundo HAN, J. & KAMBER (2000), o processo de agrupar um conjunto de objetos físicos ou abstratos em classes de objetos similares é chamado de *clusterização*. Um *cluster* é uma coleção de dados com similaridades a outros dados do mesmo *cluster* e diferenças a objetos de outros *clusters*. Um *cluster* de dados pode ser tratado coletivamente em várias aplicações. O termo *cluster,* aglomerado será mantido em inglês nesta dissertação, pois se trata de um termo bem difundido dentro das Ciências da Computação.

Num escopo estatístico, análise de *cluster* foi extensamente estudada por vários anos, focada principalmente na análise de *cluster* baseada na distância. Ferramentas de análise de *cluster* baseadas em k-media, e alguns outros métodos são implementados na maioria dos pacotes de programas para estatística.

Em aprendizado de máquina, análise de *cluster* refere-se à aprendizagem não supervisionada. Diferente da classificação, a *clusterização* não somente trabalha com classes pré definidas, mas suas classes são criadas a partir do conjunto de treinamento. Por essa razão essa é uma forma de aprendizagem por observação, isto é, aprendendo por exemplos.

2.4.1.2 Regressão

Segundo HAN & KAMBER (2000), numa regressão linear os dados são modelados utilizando-se uma linha reta. Regressão linear é a forma mais simples de regressão. Regressão linear bi-variável modelam uma variável randômica, Y (chamada de variável resposta), sendo uma função linear de outra variável randômica, X (chamada de variável de predição).

$$Y = \alpha + \beta X.$$

Onde a variação de Y é assumida como sendo constante, e α e β são coeficientes de regressão especificando a intercessão com Y e a inclinação da linha. Os coeficientes podem ser resolvidos utilizando-se o método do mínimo quadrado, o qual minimiza o erro entre a linha atual que separa os dados e a linha estimada.

Regressão múltipla é uma extensão da regressão linear envolvendo mais de uma variável de predição. Isso habilita a variável de resposta Y a ser modelada por uma função linear de um vetor multidimensional de atributos.

2.4.2 Abordagem Sintática de Reconhecimento de Padrões (ASRP)

Muitas vezes a informação significativa de um padrão não está meramente na presença, ausência, ou no valor numérico de um conjunto de características. As inter-relações e interconexões entre elas possuem importantes informações estruturais, e sua identificação facilita a descrição estrutural ou classificação. Esta abordagem é a base da abordagem sintática de reconhecimento de padrões, ou seja, usando ASRP, se estará apto a quantificar e extrair informações estruturais e acessar a similaridade estrutural entre os padrões, utilizando-se para isso de gramáticas ou autômatos. (SCHALKOFF, 1992).

Uma das técnicas de reconhecimento de padrões que utiliza uma abordagem sintática é a técnica de Árvore de Decisão que está descrita no tópico 2.4.2.1.

2.4.2.1 Árvore de Decisão

Segundo HAN & KAMBER, uma árvore da decisão é uma estrutura em forma de árvore, onde cada nodo interno denota um teste em um atributo, cada aresta representa um resultado do teste e os nodos folha representam classes ou distribuições da classe. O nó mais alto em uma árvore é o nó da raiz. Uma árvore de decisão típica é mostrada na Fig. 2.1. Representa o conceito compra de computador, predizendo se um consumidor da AllEletronics tem tendência a comprar um computador. Os nodos internos são representados como retângulos e os nodos folhas como ovais. A fim de classificar uma amostra desconhecida os valores dos atributos de um exemplo são testados na árvore de decisão. Um caminho é traçado da raiz até o nodo folha que contenha a classe de predição para o dado exemplo. As árvores da decisão podem facilmente ser convertidas para regras de classificação.

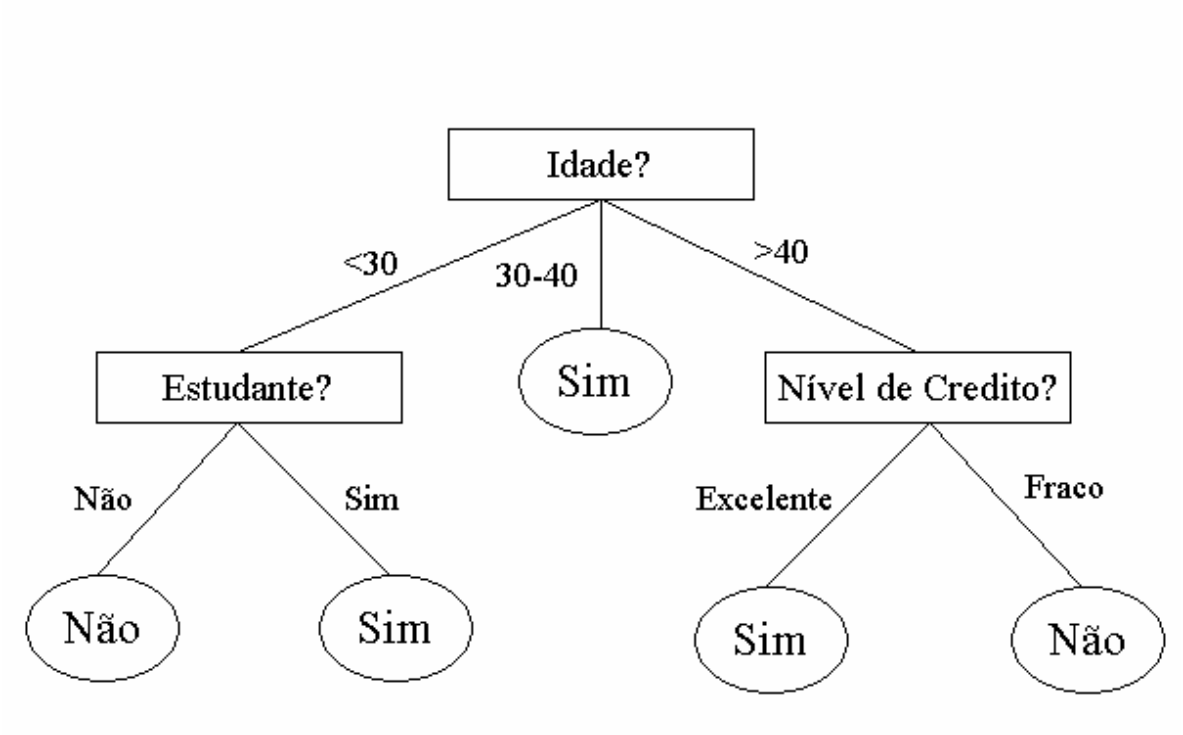


FIGURA 2.1 – Uma árvore da decisão para o conceito compra de computador, indicando se um cliente da AllElectronics é um provável comprador de computador. Cada nodo (não-folha) interno representa um teste em um atributo. Cada nodo folha representa uma classe (comprador de computador = sim ou comprador de computador = não)

2.4.3 Abordagem Neural para Reconhecimento de Padrões (ANRP)

Modernos computadores digitais podem não emular o paradigma computacional de sistemas biológicos. A alternativa para computação neural emerge das tentativas de conhecer como os sistemas neuronais biológicos armazenam e manipulam informação, isso liga uma classe de sistemas neurais artificiais chamados redes neurais. (SCHALKOFF, 1992).

A ANRP é uma abordagem não algorítmica, estratégia do tipo caixa-preta, treinável. O objetivo é “treinar” a caixa-preta neural para “aprender” a correta resposta ou saída (neste caso classificação) para cada exemplo do treinamento. Esta estratégia é atrativa para os projetistas de sistemas de reconhecimento de padrões desde que possuam, antecipadamente, uma grande quantidade de conhecimento detalhado do funcionamento interno do sistema sendo este requisito mínimo. Além disso, depois do treinamento espera-se que a estrutura interna da rede neural se auto-organize para

habilitar a extrapolação quando apresentada a um novo padrão, seguindo as bases da "experiência" adquirida dos padrões de treinamento. (SCHALKOFF, 1992).

Uma das técnicas de reconhecimento de padrões que utiliza uma abordagem neural é a técnica de Redes Neurais que está descrita no tópico 2.4.3.1.

2.4.3.1 Redes Neurais

Em termos gerais uma rede neural é um conjunto de entradas e saídas conectadas onde cada conexão tem um peso associado a ela. Durante a fase da aprendizagem, a rede aprende ajustando os pesos para poder predizer corretamente a classe do exemplo de entrada. A aprendizagem da rede neural também é tratada como aprendizagem conexionista devido às conexões entre unidades. (HAN & KAMBER, 2000)

Redes neurais necessitam de longos tempos de treinamento e são, conseqüentemente, mais apropriadas para as aplicações onde este tempo é aceitável. A técnica requer um número de parâmetros que normalmente são determinados empiricamente, como a topologia ou "estrutura" da rede (Idem).

As redes neurais foram criticadas por sua dificuldade de interpretação, por conta da dificuldade dos humanos de interpretarem o significado escondido nos pesos da rede. Estas características tornaram inicialmente as redes neurais menos aconselháveis para a mineração dos dados. (Idem).

As vantagens das redes neurais, entretanto, incluem sua tolerância elevada aos dados ruidosos e como sua habilidade em classificar padrões em que não foram treinadas. Além disso, diversos algoritmos têm sido desenvolvidos recentemente para a extração das regras ocultas nas redes neurais treinadas. Estes fatores contribuem para a utilidade das redes neurais para a classificação na mineração dos dados. (Idem).

2.4.4 Comparação entre as Abordagens de Reconhecimento de Padrões

A tabela 2.1 exibe as características de cada uma das abordagens utilizadas para reconhecimento dos padrões. Como podemos ver a abordagem neural pode ser utilizada para previsão do tempo de perfuração de poços de petróleo, pois nesse processo de previsão não estamos interessados no conteúdo semântico dos dados, mas sim na correlação entre um novo poço e os existentes na base de dados.

Tabela 2.1 – Comparação entre as abordagens de reconhecimento de padrões.

	Estatica (AERP)	**Sintatica (ASRP)**	**Neural (ANRP)**
1 - Geração de Padrão Básico	Modelos Probabilísticos	Gramáticas Formais	Estados Estável ou Matriz de Pesos
2 – Classificação de Padrões (Reconhecimento/ Descrição) Básico	Estimação/ Teoria da Decisão	Parser	Baseado em Propriedades das Redes Neurais
3 – Organização de Características	Vetor de Características	Primitivas e Relações Observadas	Entradas dos Neurônios ou Estados Armazenados
4 – Abordagem Típica Para Aprendizado (Treinamento) A - Supervisionado B – Não Supervisionado	 Densidade/ Distribuição (Usualmente Paramétrico) *Clusterizando*	 Formando Gramáticas (Heurística ou Inferência Gramatical) *Clusterizando*	 Determinando os parâmetro da Rede Neural *Clusterizando*
5 - Limitações	Dificuldade em expressar informação estrutural	Dificuldade em aprender regras estruturais	Freqüentemente a rede passa pouca informação semântica

Fonte: SCHALKOFF, Robert. **Pattern Recognition**. New York: John Wiley & Sons, Inc, 1992.

Sendo a proposta deste trabalho a utilização de uma abordagem neural para resolução do problema, a sessão 2.5 descreve o funcionamento das redes neurais. utilizadas neste trabalho.

2.5 Redes Neurais

O nosso cérebro é composto por aproximadamente 100 bilhões de neurônios, estas pequenas células componentes do nosso sistema nervoso, atuam sobre todas as funções e movimentos do nosso organismo. O neurônio tem um corpo celular chamado soma e diversas ramificações, as ramificações conhecidas como dendritos, conduzem sinais das extremidades para o corpo celular. Existe também uma ramificação geralmente única chamada axônio que transmite as informações do corpo celular para as suas extremidades, as extremidades dos axônios são conectadas com dendritos de outros neurônios pelas sinapses. Em muitos casos um neurônio é conectado com outros axônios ou com o corpo de outro neurônio. O conjunto de todos os neurônios conectados forma uma grande rede denominada Rede Neural. (BARRETO, 1999).

As sinapses transmitem estímulos através de diferentes concentrações de Na+ (Sódio) e K+ (Potássio) e o resultado disto pode ser estendido por todo o corpo humano. Esta grande rede proporciona uma fabulosa capacidade de processamento e armazenamento de informação. (HAYKIN, 2001).

Tendo como inspiração este sistema biológico surgiram varias técnicas de Redes Neurais Artificiais (RNAs), sendo que todas essas técnicas baseiam-se em modelos de neurônios artificiais implementados de forma a possuir algumas das características de neurônios biológicos, essas técnicas de RNAs estão em sua maioria fundadas sobre o fato da inteligência surgir através do peso das conexões sinapticas.

Existem várias topologias de redes neurais diferentes e também muitas formas de treiná-las. As principais topologias são as redes diretas e as redes recorrentes e cada topologia possui formas diferentes de treinamento. Nas seções seguintes serão abordadas as topologias de redes neurais utilizados no Sistema.

2.5.1 Redes Diretas

As redes diretas, ou “feedforward” são redes neurais cujo grafo não possui ciclos e geralmente estão representadas em camadas. Nestas redes, os neurônios de uma camada i transmitem seu sinal de saída para os neurônios de uma camada j, onde $j > i$.. Por exemplo, os neurônios que recebem sinais de excitação do meio externo estão na camada de entrada, os neurônios que estão na saída são chamados de camada de saída, e

os que não pertencem nem a camada de entrada e nem a camada de saída são os neurônios internos, pertencentes à camada intermediária.

Esta topologia é uma das mais conhecidas e utilizadas, principalmente pelo fato de que a implementação de métodos de aprendizado para estas redes ser mais simples. O algoritmo “”backpropagation” é um dos mais utilizados nestas redes. (DAYHOFF, 1990).

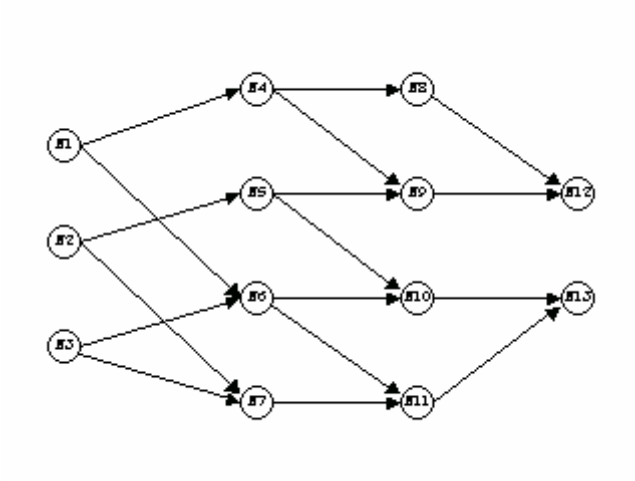


FIGURA 2.2 – Exemplo de Rede Direta

2.5.2 Aprendizagem Supervisionada

Neste tipo de aprendizado, um “professor” indica se o comportamento da rede é bom ou ruim. Este método de aprendizado é o mais comum no treinamento das RNAs e é chamado Aprendizado Supervisionado porque a o resultado desejado na saída da rede é fornecido por um “professor” externo. O objetivo é ajustar os parâmetros da rede de forma a encontrar uma ligação entre os pares de entrada e saída fornecidos.

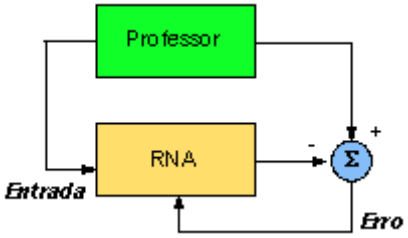


FIGURA 2.3 – Aprendizado supervisionado.

No aprendizado supervisionado, a cada exemplo colocado na entrada da rede, a saída obtida nos neurônios de saídas é comparada com os resultados desejados e o professor, através do erro obtido, supervisiona o ajuste dos pesos de forma com que a rede possa direcionar o seu aprendizado para a resposta correta, ou seja, os pesos são ajustados de forma a minimizar o erro.

O algoritmo de Retropropagação de Erros (Error Backpropagation) é o algoritmo mais conhecido de aprendizagem supervisionada para redes de camadas.

2.5.3 Retropropagação de Erros

Nesta seção o algoritmo de Retropopagação de Erros está descrito de forma simplificada, este algoritmo foi retirado de BARRETO (1992).

O Algoritmo:

1. "Seja A, o número de neurônios da camada de entrada, conforme determinado pelo comprimento dos vetores de entrada de treinamento,C, o número de neurônios da camada de saída. Escolha B, o número de neurônios da camada intermediária. Geralmente as camadas de entrada e intermediária têm, cada uma, um neurônios extra, usado como polarização ("bias"), portanto, usa-se os intervalos (0, ... ,*A*) e (0, ... ,*B*) para estas camadas, especificamente."
2. "Inicialize os pesos da rede. Cada peso deve ajustado aleatoriamente."
3. "Inicialize as ativações dos neurônios com *bias*, ou seja $x_0 = 1$ e $h_0 = 1$;"
4. "Escolha um par entrada-saída. Suponha que o vetor de entrada seja X_l, e que o vetor de saída desejada seja Y_l, Atribua níveis de ativação, aos neurônios da camada de entrada."
5. "Propague a ativação dos neurônios da camada de entrada para os da camada de intermediária, usando-se como sugestão, a função sigmóide unipolar."

$$H_k = \frac{1}{1+e^a}, \forall k = 1,\cdots,B$$

Onde:

$$a = \sum_{l=0}^{A} (W1_{lk} X_l)$$

6. "Propague a ativação dos neurônios da camada intermediária para os da camada de saída."

$$Y_k = \frac{1}{1+e^a}, \forall k, \cdots, C$$

Onde:

$$a = \sum_{l=0}^{B} (W2_{lk} H_l)$$

7. "Compute os erros dos neurônios da camada de saída, denotado por $\delta 2_k$."

$$\delta 2_k = Y_k (1 - Y_k)(T_k - Y_k), \forall k = 1, \cdots, C$$

8. "Compute os erros dos neurônios da camada intermediária, denotada por $\delta 1_k$."

$$\delta 1_k = H_k (1 - H_k) \sum_{l=1}^{C} \delta 2_l W 2_{lk}, \forall k = 1, \cdots, B$$

9. "Ajuste os pesos, entre a camada intermediária e a de saída. O coeficiente de aprendizagem é denotado por η."

$$\Delta W 2_{lk} = \eta \delta 2_{kl} H_l, \forall l = 0, \cdots, B; \forall k = 0, \cdots, C$$

10. "Ajuste os pesos entre a camada de entrada e a intermediária."

$$\Delta W 1_{lk} = \eta \delta 1_k X_l, \forall l = 0, \cdots, A; \forall k = 1, \cdots, B$$

11. "Vá para a etapa 4 e repita. Quando todos pares entrada-saída, tiverem sido apresentados à rede, uma época terá sido completada. Repita as etapas de 4 a 10, para tantas épocas quanto forem desejadas."

Maiores detalhes sobre o algoritmo de aprendizagem por retropropagação de erros podem ser encontrados em BARRETO (1999).

2.5.4 Rede Competitiva Simples

Redes Competitivas Simples (RCS) são redes neurais onde o aprendizado emerge da competição entre seus neurônios. O resultado dessa competição pode ser utilizado para classificar padrões.

> A rede básica do aprendizado competitivo consiste de duas camadas – uma camada de entrada e uma camada competitiva. Na camada competitiva, as unidades competem pela oportunidade de responder aos padrões de entrada. A vencedora representa a categoria de classificação para o padrão que entrou. A competição pode ser realizada por meio de um algoritmo que designa a unidade vencedora, ou, alternativamente, através da inibição entre as unidades da camada competitiva. No caso de inibição, a camada competitiva progride a um estado em que somente a unidade vencedora fica ativa (BARRETO, 1999).

O treinamento de uma RCS consiste na apresentação de cada exemplo à rede. O neurônio vencedor tem seus pesos atualizados de acordo com um passo alfa. Os exemplos são apresentados sucessivamente até que o erro da RCS seja aceitável.

A arquitetura de uma rede competitiva pode ser vista na Fig. 2.4.

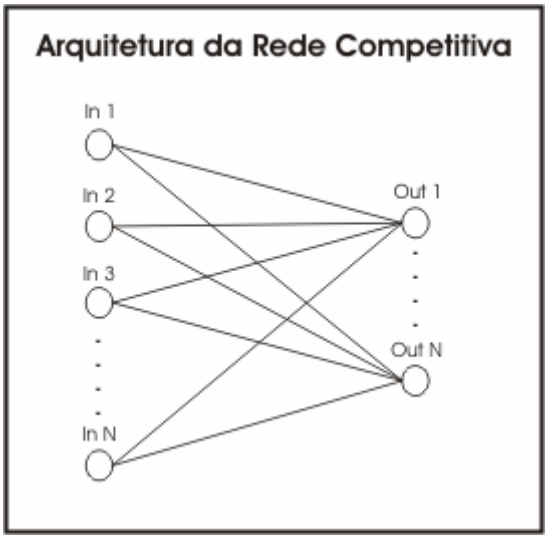


FIGURA 2.4 – Arquitetura da Rede Competitiva.

Como se pode ver na Fig. 2.4 os neurônios da camada de entrada são conectados aos neurônios da camada de saída através de pesos. Toda vez que um padrão é apresentado, as distâncias entre o padrão e os pesos de cada uma das saídas são calculadas. Assumindo os pesos e o novo padrão como vetores podemos ver uma representação gráfica na Fig. 2.5.

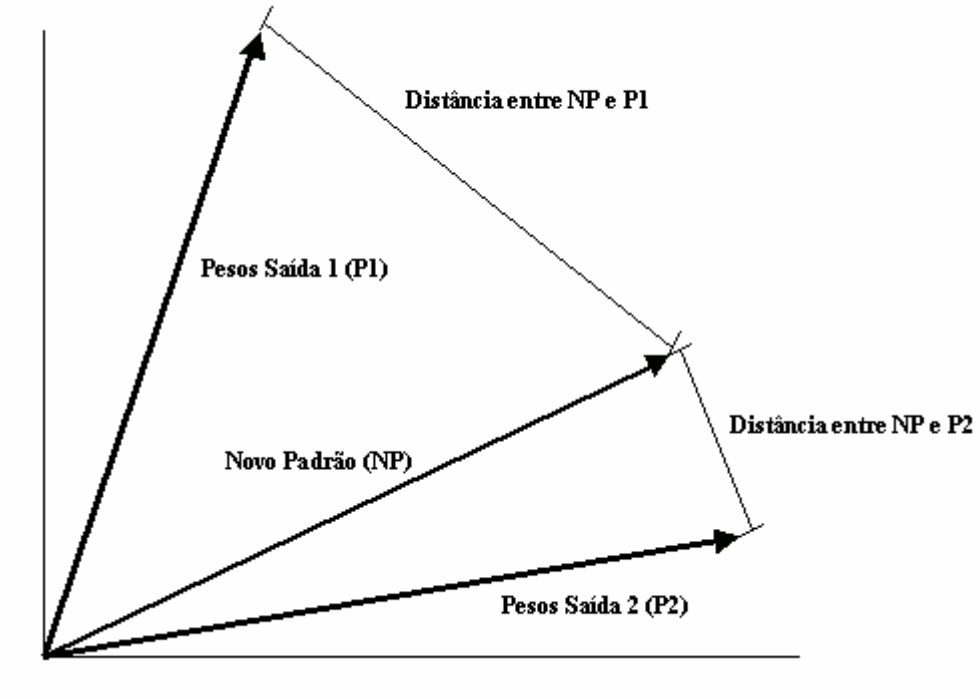


FIGURA 2.5 – Apresentação de um padrão a rede competitiva.

O neurônio de saída cuja distância é a menor torna-se o neurônio vencedor da competição. A saída do neurônio vencedor é 1 e a dos outros neurônios é reduzida a 0.

Durante o treinamento são apresentados sucessivos padrões a rede competitiva, para cada padrão apresentado é calculado um neurônio vencedor. O neurônio vencedor é modificado seguindo uma taxa de aprendizagem α, para que se aproxime ainda mais do padrão apresentado.

Na Fig. 2.6 pode-se ver uma representação esquemática da influência da taxa de aprendizado sobre os pesos do neurônio vencedor.

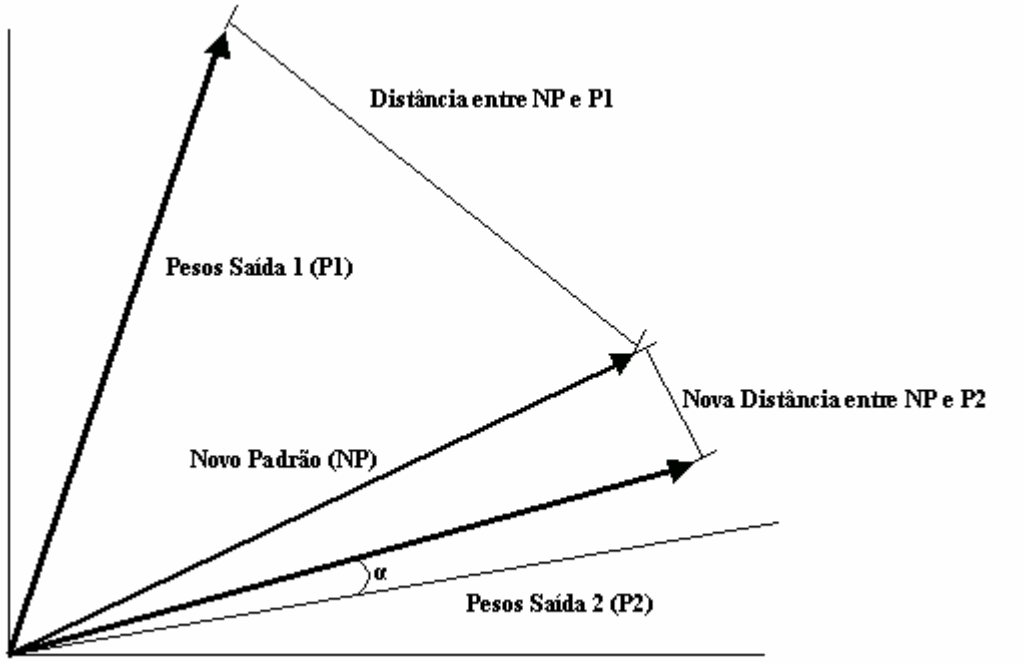


FIGURA 2.6 – Representação esquemática da aprendizagem numa rede competitiva.

CAPÍTULO 3 – PERFURAÇÃO E EXPLORAÇÃO DE PETRÓLEO

Os processos de perfuração e completação de poços de petróleo são processos complexos e caros, uma boa parte do custo está associada ao tempo das intervenções. O aluguel de uma sonda custa em torno de US$ 180.000 por dia, segundo THOMAS (1996). Neste capítulo, será apresentada uma descrição dos processos de perfuração e completação, assim como as operações envolvidas nesses processos.

3.1 Perfuração

A perfuração de um poço de petróleo é realizada através de uma sonda como pode ser visto na Fig. 3.1. No processo de perfuração rotativa, as rochas são perfuradas pela ação da rotação e do peso aplicado a uma broca existente na extremidade de uma coluna de perfuração. A coluna de perfuração consiste basicamente de comandos (tubos de paredes espessas) e de tubos de perfuração (tubos de paredes finas). Os fragmentos de rocha são removidos continuamente através de um fluido de perfuração, lama, injetado por bombas para o interior da coluna de perfuração. O fluido de perfuração é injetado na coluna de perfuração através da cabeça de injeção ou “swivel” e retorna à superfície através do espaço anular formado pelas paredes do poço e da coluna.

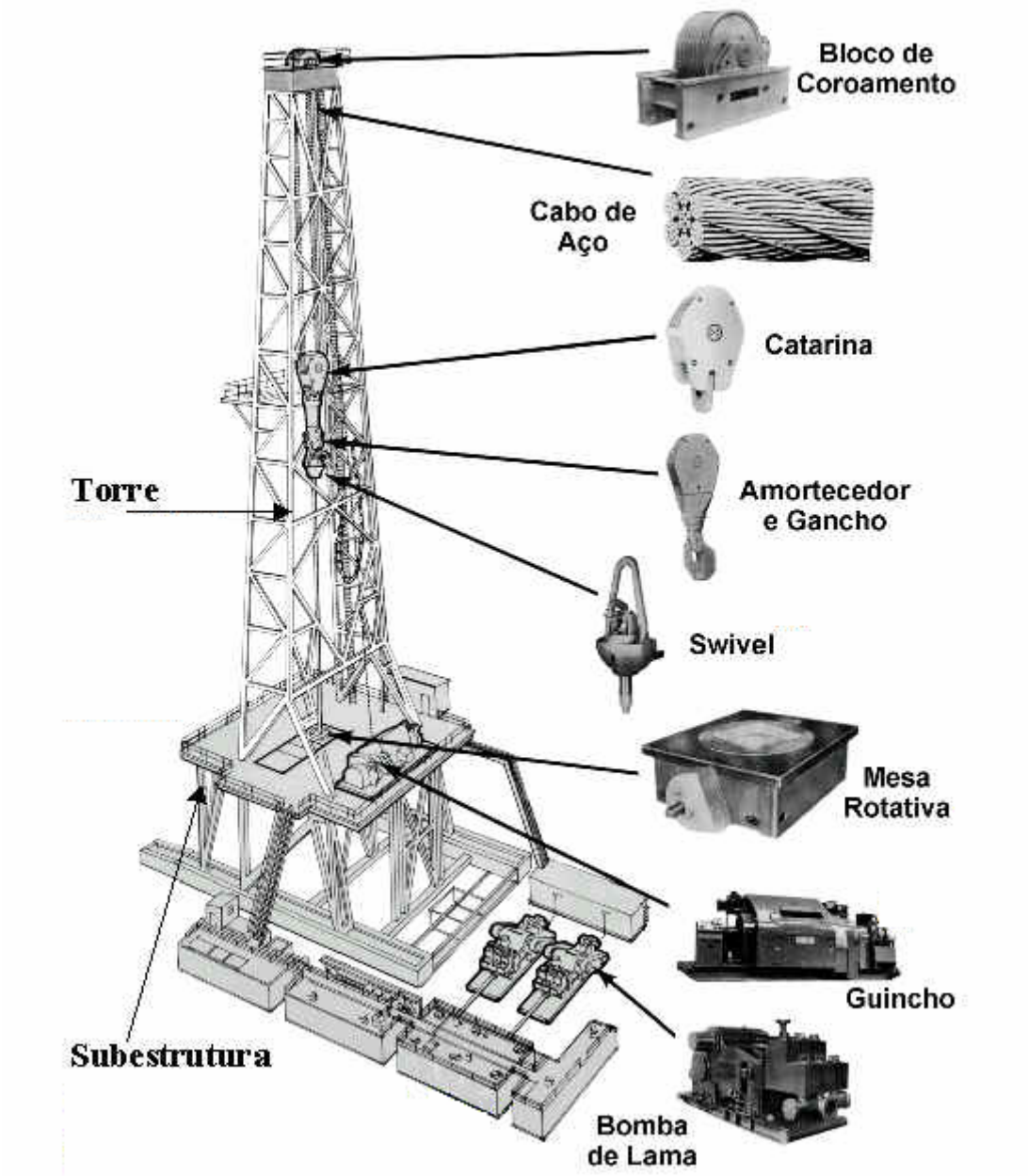


FIGURA 3.1 – Sonda (Extraído de DOMINGOS, Luís. **Perfuração no mar.** Disponível em: <http://histpetroleo.no.sapo.pt/perf_mar.htm>. acesso em: 1 mar. 2004).

Ao atingir certa profundidade a coluna de perfuração é retirada do poço e uma coluna de revestimento de aço, com diâmetro inferior ao da broca, é descida no poço. O espaço anular entre o tubo de revestimento e as paredes do poço é cimentado com a finalidade de isolar as rochas atravessadas, permitindo então o avanço da perfuração com segurança. Após a operação de cimentação, a coluna de perfuração é novamente descida no poço, tendo em sua extremidade uma nova broca, como mostra a Fig. 3.2, de diâmetro menor do que a do revestimento, para prosseguimento da perfuração. Dessa

maneira o poço é perfurado em diversas fases, caracterizadas pelos diferentes diâmetros das brocas. (THOMAS, 1996).

FIGURA 3.2 – Brocas (Extraído de

DOMINGOS, Luís. **Perfuração no mar.** Disponível em: <http://histpetroleo.no.sapo.pt/perf_mar.htm>. acesso em: 1 mar. 2004).

As operações típicas de perfuração são: Alargamento e repassamento, Conexão, Manobra e circulação, Revestimento, Cimentação, Perfilagem e Movimentação da sonda. A operação de alargamento e repassamento consiste em se perfurar o poço com uma broca de diâmetro maior que a utilizada para perfuração. A operação de conexão, manobra e circulação consiste na conexão ou desconexão dos tubos de perfuração à coluna e a circulação refere-se a circular o fluido de perfuração para retirar os cascalhos do espaço anular. A operação de revestimento do poço constitui-se na instalação das colunas de revestimento: condutor, revestimento de superfície, revestimento intermediário, revestimento de produção, liner, tié back. A cimentação é feita no espaço anular entre a tubulação de revestimento e as paredes do poço.A refilagem é feita após a perfuração e consiste em descer várias ferramentas com a finalidade de medir propriedades das formações. Ao fim das etapas de perfuração inicia-se o processo de completação. (JACINTO, 2002).

3.2 Completação

Ao terminar a perfuração de um poço é necessário deixá-lo em condições de operar de forma segura e econômica durante toda sua vida produtiva. Ao conjunto de operações destinadas a equipar o poço para produzir óleo ou gás (ou ainda injetar fluido nos reservatórios) denomina-se completação.

Quanto aos aspectos técnico e operacional deve-se buscar otimizar a vazão de produção (ou injeção) e tornar a completação a mais permanente possível, ou seja, aquela que minimize a necessidade de intervenções futuras para manutenção do poço (as chamadas operações de "workover").

Considerando que a completação tem reflexos em toda vida produtiva do poço e envolve altos custos faz-se necessário um planejamento criterioso das operações e uma análise econômica cuidadosa. (THOMAS, 1996).

A completação pode ser dividida:

a) Quanto ao posicionamento da cabeça do poço: terrestres ou marítimas (águas rasas ou águas profundas), impactando diretamente nos sistemas de cabeça de poço e no tipo de árvore de natal (conjunto de válvulas acopladas a cabeça do poço e que controlam a passagem do fluido) utilizada.

b) Quanto ao revestimento de produção: o poço aberto (zona produtora totalmente aberta), com liner rasgado ou canhoneado (posicionamento dos tubos previamente rasgado em frente à zona produtora ou então cimentado e posteriormente canhoneado nas zonas de interesse); com revestimento canhoneado (mais utilizado, compreendendo a descida do revestimento de produção até o fundo do poço); após a cimentação do espaço anular, o revestimento é canhoneado nos intervalos de interesse.

c) Quanto aos números de zonas exploradas: Simples (produção de modo controlado e independente de uma única zona de produção, através de uma única tubulação); Múltipla (permite produzir ao mesmo tempo duas ou mais zonas ou reservatórios diferentes, através de uma ou mais colunas de produção descidas no poço).

As etapas típicas de uma completação de um poço marítimo compõem-se da instalação dos equipamentos de superfície (cabeça de produção BOP), condicionamento do poço (condicionamento do revestimento de produção e substituição do fluido que se encontra no interior do poço, pelo fluido de completação); avaliação da qualidade da

cimentação (avaliação através de perfis acústicos que medem a aderência do cimento ao revestimento e do cimento à formação); canhoneio (perfuração do revestimento utilizando cargas explosivas, visando comunicar o interior do poço com a formação produtora); instalação da coluna de produção (descida da coluna de produção pelo interior do revestimento de produção) e colocação do poço em produção (induz-se a surgência no poço ou inicia-se o método de elevação artificial e efetua-se o teste inicial de produção para medir a vazão de produção e avaliar o desempenho do poço). (JACINTO, 2002).

As operações citadas acima se desdobram em várias sub-operações que não foram descritas, pois o intuito deste capítulo é dar uma noção geral do processo de perfuração e completação de poços de petróleo.

CAPÍTULO 4 – SISTEMA PROPOSTO

Estudos a respeito da acurácia ou incerteza das respostas fornecidas por RNAs, assim como sobre uma possível distribuição estatística para o erro da resposta, tanto para o conjunto de treinamento como para o conjunto de testes, são raros na literatura e envolvem conhecimentos bastante complexos, sendo que alguns estudos para arquiteturas específicas podem ser vistas em (HAYKIN 2001). Entretanto parece correto afirmar que a precisão da resposta fornecida por uma RNA depende do modelo ou arquitetura da RNA utilizada, bem como da representatividade dos dados nos conjuntos utilizados para treinamento e teste da Rede. (Idem)

Durante o desenvolvimento deste trabalho uma série de modelos de RNAs foram propostas, sendo realizados testes com a base de casos disponível, com o objetivo de avaliar de maneira mais pragmática a adequação e precisão de cada modelo para estimar o tempo total de intervenções e detectar aspectos relevantes da constituição do conjunto de treinamento. Com os resultados destas propostas direcionou-se a implementação da solução pela qual se optou.

Estas propostas preliminares, a proposta da arquitetura neural adotada no sistema, bem como uma descrição detalhada dos módulos que compõem o Sistema e seu funcionamento, são apresentadas neste capítulo.

4.1. Propostas Preliminares

Durante a fase de projeto foram desenvolvidos vários protótipos utilizando diversos modelos e arquiteturas diferentes de RNAs. Esses protótipos revelaram-se de extrema importância no processo de entendimento do problema, além de proporcionarem uma evolução da arquitetura que levou à arquitetura adotada no Sistema.

A base utilizada nos testes e no desenvolvimento do Sistema foi cedida por uma empresa do setor petrolífero. No caso dos testes, a base utilizada nos teste possuía os campos: *Tipo de Intervenção, Tipo de Fluído, Tipo de Poço, Afastamento Lateral, Lâmina D'Água, Campo, Sonda, Profundidade Final, Azimute, Tempo Total.* Estes campos foram obtidos através da filtragem dos campos de uma tabela de intervenções

realizadas pela empresa do setor petrolífero que cedeu a base de dados. Esta tabela possui 3050 intervenções cadastradas, além dos campos de controle interno da empresa, que não tem relevância para o estudo realizado.

Alguns modelos e testes que foram propostos e efetuados, bem como uma análise resumida dos resultados, podem ser vistos a seguir:

1. **Teste utilizando toda a base como conjunto de treinamento**

Este teste foi realizado utilizando o matlab e consistiu no treinamento de uma rede neural direta utilizando 70% da base de dados como conjunto de treinamento e 30% como conjunto de avaliação. A rede foi para prever somente o tempo total de perfuração, o objetivo do teste era avaliar a rede direta para esse tipo de previsão. As previsões feitas pela rede treinada possuíam um grande erro se comparadas com valores reais, sendo este valores: erro médio do tempo total de perfuração de 412,8436 horas e desvio padrão do erro de 390,2307 horas, sendo que o tempo médio de intervenções é de 409,7630 e o desvio padrão é de 504,8013 horas. Este erro deve-se a grande ambigüidade encontrada nos dados utilizados para o treinamento da rede.

2. **Teste utilizando os dados separados por tipo de intervenção.**

Este teste foi realizado com uma implementação própria de rede neural direta consistindo no treinamento de várias redes diretas, uma para cada tipo de intervenção. Os resultados obtidos neste teste foram pouco melhores do que o obtido no teste anterior, isso demonstrou que normalmente os exemplos de um tipo de intervenção não prejudicavam o aprendizado de outra intervenção, o real problema estava nas ambigüidades existentes dentro de uma mesma intervenção. Assim sendo, partiu-se para uma outra abordagem utilizada no teste 3.

3. **Teste eliminando dados cuja contribuição no erro final foi muito grande.**

O sistema do teste anterior foi modificado para retornar a participação de cada elemento do conjunto de treinamento no erro total de treinamento. Os elementos com maior participação no erro total eram eliminados e a rede direta treinada novamente.

Este teste obteve melhores resultados, pois os elementos que causavam as ambigüidades no conjunto de treinamento possuíam uma grande participação no erro total. Em média somente 20 elementos de um conjunto total de mais de 3000 eram responsáveis por aproximadamente 50% do erro de treinamento da rede.

Apesar dos resultados obtidos neste teste terem sido satisfatórios essa abordagem não deve ser utilizada, pois, simplesmente, induziu resultados satisfatórios eliminando elementos discrepantes, que ao serem eliminados, descaracterizam os dados uma vez que uma das maiores características do problema é sua grande variabilidade.

4. Teste agrupando os dados em conjuntos e utilizando a média e o desvio padrão.

Neste teste, houve uma mudança significativa no sistema, os dados de treinamento foram agrupados utilizando um algoritmo estatístico *k-media*, baseado na distância euclidiana entre os dados. Para cada agrupamento gerado foi calculado a média e o desvio padrão. A média e o desvio padrão do grupo transformaram-se nos objetivos de treinamento em conjunto com o tempo total. Os resultados alcançados para média e desvio padrão foram aceitáveis e para tempo total, melhoraram em relação ao teste anterior. Essa abordagem mostrou-se mais satisfatória que a anterior, pois além de alcançar resultados melhores possui uma média e um desvio padrão associado à resposta, esse fato auxilia na interpretação dos resultados e na visualização da sua acurácia.

5. Teste agrupando os dados em conjuntos e utilizando o grupo a que pertence como valor de entrada.

Com relação ao teste anterior, a única alteração realizada foi a adição do conjunto a que cada dado pertence. Assim sendo, ao passarem pela rede direta, os dados do conjunto de treinamento adquirem mais um parâmetro que é o conjunto a que o dado foi agrupado no passo anterior. Os resultados desse teste foram melhores se comparados com o anterior, todos os erros possuíam valores aceitáveis. Esse teste foi utilizado como base da arquitetura utilizada na versão final do sistema.

Alem disso, tendo concluído a etapa de avaliação dos protótipos foram detectados aspectos do sistema que influenciavam decisivamente nos resultados. Estes aspectos foram analisados e os resultados dessa análise foram utilizados, assim como os resultados do teste anterior, no desenvolvimento da arquitetura final do sistema e sua implementação. Estes aspectos são:

a) **Inconsistências nos dados**: a base de dados utilizada no projeto possuía uma grande quantidade de dados inconsistentes ou incompletos. Esses dados, quando

substituídos por valores padrão causam uma grande quantidade de erros, que, prejudicando as previsões, deixam os resultados menos confiáveis.

b) **Grande variabilidade dos dados**: os dados utilizados possuíam uma grande variabilidade, sendo que, nas mesmas situações o tempo de perfuração variou mais de 1000%. Com o objetivo de exemplificar esse fato, observa-se a tabela 4.1 onde é apresentado um exemplo real retirado da base de dados. Nesse exemplo, vimos duas *Perfurações Exploratórias* com mesmos parâmetros que, no entanto, possuem uma diferença de mais de 1600% entre seus tempos totais.

Tabela 4.1 – Exemplo de dados reais

TipoIntervencao	Fluido Poco	Tipo Poco	AfastamentoAlvo	Lamina Dagua	Cam Poco	Tipo Sonda	ProfFinal Sondador	Azimute Poco	Total
Perfuração Exploratória			0,00	-182,00	CVS	SS	5.425,00		178,50
Perfuração Exploratória			0,00	-182,00	CVS	SS	5.425,00		2.917,50

Fonte: Dados reais retirados da base

c) **Resposta desejada do sistema**: o software deve retornar não só uma previsão do tempo de perfuração, mas também a média e o desvio padrão associado a esse resultado. Como o tempo para o processo de perfuração possui uma grande variabilidade, assim, poços iguais levam tempos bastante diferentes para serem perfurados ou completados, uma rede neural preveria um valor intermediário entre os dois poços semelhantes. Esse valor intermediário entre os dois poços de nada serve sem uma idéia da variabilidade envolvida nesse número. Assim sendo, o agrupamento de poços semelhantes e o cálculo da média e dos desvios padrão para esse conjunto de poços, para serem utilizados posteriormente no treinamento, torna-se de grande importância para se ter uma noção da acurácia e variabilidade dos resultados fornecidos pela RNA. Além disso, deve-se retornar um valor de média e desvio padrão para que se possa realizar estudos sobre o risco envolvido na perfuração de poço, estes estudos podem ser feitos da mesma forma que são feitos atualmente com os resultados de programas de simulação numérica como o E&P Risk III.

4.2. Proposta Adotada

Tendo concluído a etapa de testes e análise de alguns modelos e arquiteturas de RNAs, bem como de tratamento dos conjuntos de treinamento e teste e identificando as características mais relevantes das arquiteturas e dados de treinamento, iniciou-se a implementação da proposta adotada no Sistema.

A implementação do sistema foi realizada com base no último protótipo avaliado, sendo esse o que apresentou melhores resultados. A maior diferença entre a arquitetura do último protótipo e a arquitetura final do Sistema foi a adoção de uma rede competitiva em substituição ao algoritmo estatístico *k-media* que classificava as entradas em grupos. Assim sendo, o núcleo do Sistema tornou-se uma arquitetura composta por dois modelos de redes, uma rede direta e uma rede competitiva.

A arquitetura neural do Sistema está ilustrada na Fig. 4.1. O diagrama de blocos da Fig. 4.1 apresenta as ligações entre as redes representadas na figura como caixas. Nos tópicos 4.3 e 4.4 será descrito o funcionamento de cada uma das redes, assim como suas saídas e entradas.

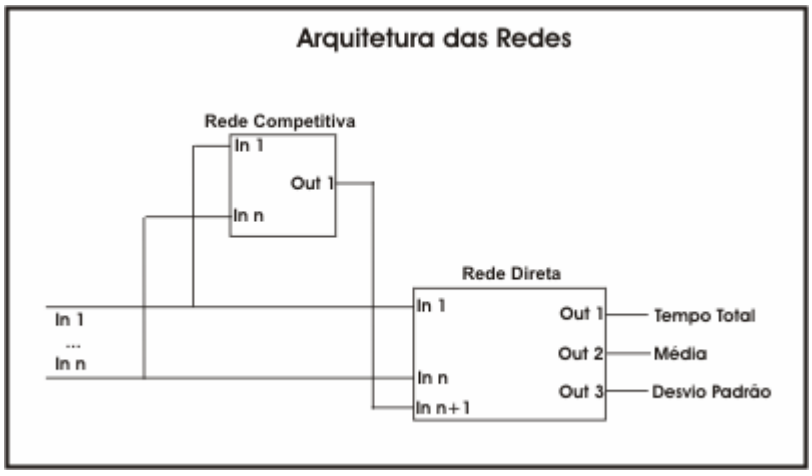


FIGURA 4.1 – Arquitetura de redes do Sistema

4.2.1 Rede Competitiva

Neste experimento, utilizou-se uma rede competitiva simples que possui duas camadas de neurônio totalmente conectadas entre si. Quando um dado é apresentado a essa rede sua saída é o índice do neurônio vencedor, ou seja, o neurônio com maior valor de ativação para a entrada apresentada.

A rede foi treinada através de um algoritmo de aprendizagem supervisionada, nesse sistema de aprendizado os exemplos são apresentados sucessivamente por um determinado número de vezes. A cada exemplo apresentado à rede, o neurônio vencedor é calculado, e seus pesos são modificados para aproximarem-se aos valores dos dados de entrada.

Inicialmente, todos os neurônios possuem uma espécie de *bônus* chamado de *bias*. Para cada vez que um neurônio vence a disputa com os outros seu *bias* é subtraído até um mínimo de zero. A utilização de *bias* é de grande importância para evitar que um neurônio sempre seja vencedor e seja criado um único grupo contendo todos os exemplos. A Fig. 4.2 contém um diagrama que demonstra como são as conexões entre os neurônios em uma rede competitiva. (KOHONEN, 1987).

No Sistema, a rede competitiva tem a incumbência de agrupar os dados em grupos de dados semelhantes. A informação de que grupo um dado pertence é importante pois para cada grupo será calculado uma média e um desvio padrão, que serão utilizados posteriormente para o treinamento da rede direta. Numa rede competitiva cada neurônio da camada de saída representa um grupo ou conjunto (*Cluster*).

Uma das vantagens da utilização de uma rede competitiva sobre o algoritmo estatístico *k-media,* utilizado nos testes 4 e 5 é que depois de treinada, a rede competitiva pode determinar o grupo a que pertence um novo dado de entrada mesmo que esse dado não tenha sido visto durante o treinamento. Além disso, a rede competitiva gera um agrupamento mais preciso, pois não necessariamente o elemento representativo do grupo deve estar presente no conjunto de treinamento, ao contrário do algoritmo matemático, onde o elemento que representava o grupo era um elemento do conjunto de treinamento. No caso da rede competitiva o elemento utilizado para comparação de um candidato com o conjunto é o centro de gravidade do próprio conjunto.

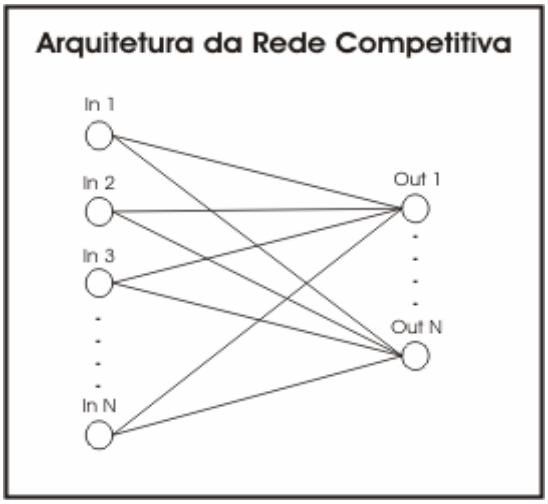


FIGURA 4.2 – Arquitetura da rede competitiva

4.2.2 Rede Direta

Neste experimento foi utilizada uma rede direta de três camadas. Esta rede foi treinada utilizando-se o algoritmo de Retropopagação de Erros (Backpropagation). A rede direta tem a incumbência de abstrair as variações entre diferentes entradas. Nessa arquitetura a rede direta é treinada para fornecer três tipos de saída: *tempo total*, *média* e *desvio padrão*. Como entrada ela recebe, além do dado, parâmetros relacionados à intervenção apresentada à rede e ao grupo a que o dado pertence, informação esta obtida da rede competitiva.

Após ter sido treinada essa rede direta tem a capacidade de abstrair as variações entre os dados e sua influência nas três variáveis de saída.

Na Fig. 4.3 é apresentado um diagrama que descreve a rede neural direta utilizada nesse projeto.

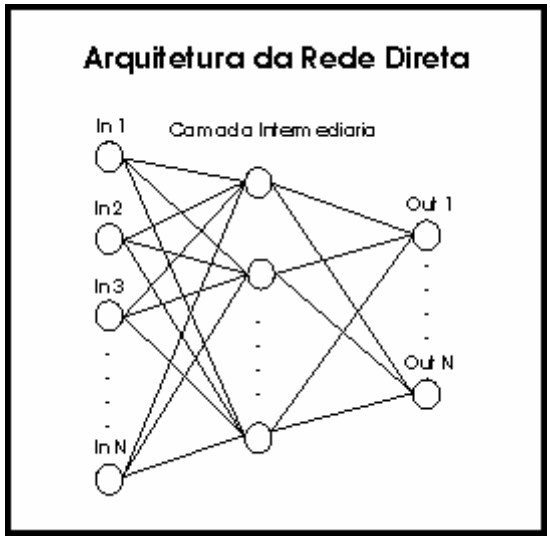


FIGURA 4.3 – Arquitetura da rede direta

4.3. Treinamento das Redes

No item anterior, foi descrito o funcionamento de cada uma das redes separadamente, porém, existem dependências entre as redes durante o processo de treinamento. Assim sendo, o treinamento das redes do sistema segue os seguintes passos:

1- **Tratamento dos Dados**: os dados são tratados para se adequarem ao formato exigido pelas redes neurais. Em primeiro lugar os dados que possuem campos em branco são excluídos. Em seguida os dados numéricos são normalizados entre 0,1 e 0,9. Os dados qualitativos são transformados em binário, ou seja, se existem X tipos para dada variável são alocados X neurônios para tal variável. O neurônio correspondente ao tipo da entrada recebe o valor 0,9 e os demais neurônios que representam essa variável recebem 0,1.

Normalmente, normalizam-se as variáveis de saída entre 0,1 e 0,9 para que os valores de saída não trabalhem nos extremos da função. Buscando padronizar a função de normalização, optou-se por normalizar também as entradas entre 0,1 e 0,9.

Na tabela 4.2, são apresentados dois dados, um no formato que o sistema lê do arquivo de dados e o outro no formato convertido para treinamento das redes.

2- **Divisão dos Conjuntos:** nessa etapa os dados serão divididos em dois conjuntos, um conjunto de treinamento e um conjunto de testes e validação. A divisão é feita de forma aleatória respeitando a proporção de 70% dos dados para treinamento e

30% para testes. Para trabalhos futuros fica a sugestão de utilização de métricas mais confiáveis que levem em consideração a natureza dos dados, como por exemplo, uma métrica que garanta que pelo menos um exemplar de cada grupo gerado pela rede competitiva esteja presente no conjunto de treinamento.

3- **Treinamento da Rede Competitiva:** tendo os dados preparados é iniciado o treinamento da rede competitiva por um número de épocas pré-determinado. Tendo concluído o treinamento o grupo a que pertence cada dado é armazenado para posterior utilização no treinamento da rede direta. Os pesos da rede competitiva são armazenados.

4- **Cálculo da Média e do Desvio Padrão:** para cada grupo gerado pela rede competitiva são calculados a Media e o Desvio Padrão do *tempo total* dos dados pertencente ao grupo. Esses valores são armazenados para utilização no treinamento da rede direta.

5- **Treinamento da Rede Direta:** a rede direta é treinada utilizando os dados mais o grupo a que ele pertence, sendo a saída desejada a média e o desvio padrão para o conjunto a que o dado pertence, e o tempo total referente ao dado. A rede direta é treinada por um número de épocas pré-determinado durante a etapa de seleção dos parâmetros de treinamento do Sistema, e após o treinamento seus pesos são armazenados.

Tabela 4.2 – Comparação entre dado original e dado convertido

Variáveis	Tipo Intervenção	Tipo Fluído	Tipo Poço	Afast. Lateral	Lamina D'Agua	Campo	Tipo Sonda	Profund. Final	Azimute	Tempo Total
Dado Original	Restauração	OL	P	2147,85	-117,00	BG	SM	4213,00	47,09	96,00
Dado Convertido e normalizado	0.9 0.1 0.1 0.1 0.1 0.1 0.1 0.1	0.9 0.1 0.1	0.9 0.1	0.42	0.86	0.9 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	0.9 0.1 0.1 0.1 0.1 0.1 0.1 0.1	0,90	0,20	0.11

4.4 Implementação

4.4.1 Plataforma de Desenvolvimento

Para o desenvolvimento do software foi utilizada a linguagens C++ e o compilador utilizado foi o C++ Builder.

A Linguagem C++ foi escolhida, pois possibilita o desenvolvimento de softwares de alto desempenho e o sistema necessita de tal desempenho, pois, realiza um grande número de operações que demandam muito tempo.

O C++ Builder foi utilizado por ser o método mais simples de se desenvolver interfaces gráficas em C++. O uso do C++ Builder agilizou todo o desenvolvimento das interfaces, o que possibilitou uma maior concentração do tempo no desenvolvimento da técnica utilizada.

4.4.2 Detalhes de Desenvolvimento

Foi desenvolvida uma biblioteca de entrada e tratamento de dados. Essa biblioteca lê um arquivo, exportado de uma base de dados no formato texto. Cada campo dos dados é classificado automaticamente com *quantitativo* ou *qualitativo*. A classificação é feita através de um algoritmo que analisa todos os campos de todos os dados e verifica para cada campo se ele possui somente valores numéricos ou possui também valores nominais. Se um campo possuir somente valores numéricos ele é classificado como *quantitativo.* Se possuir algum valor que não seja numérico o campo é classificado como *qualitativo.* O usuário pode no sistema alterar a classificação, mas um dado qualificado como *qualitativo* não pode ser classificado como *quantitativo*, pois não será possível converter palavras em números, mas uma classificação numérica pode ser considerada como um dado *qualitativo*. O método citado acima é um método simples para tipificação de dados. Uma análise mais completa a respeito de soluções mais elaboradas para tipificação de dados pode sem encontrada em SANTOS (2001).

4.4.3 Interface

A interface do sistema possui duas abas: uma aba é utilizada para avaliação de novas operações e outra aba utilizada para treinamento das redes do sistema.

4.4.3.1 Interface de Treinamento

A interface de treinamento do sistema foi implementada na forma de um guia de operação, “wizard”, para facilitar seu uso. O usuário do sistema tem somente que preencher os dados de cada tela e passa para a seguinte clicando no botão *Próximo*, ou voltar para tela anterior clicando no botão *Anterior*. A maior parte das telas já vem totalmente preenchidas, o usuário pode modificar os parâmetros na tentativa de melhorar os resultados, mas um usuário inexperiente tem como opção prosseguir o treinamento sem se preocupar com os parâmetros que ele desconhece. As etapas do guia de operação são:

Etapa 1 – entrada de dados: nessa etapa deve-se *importar* um arquivo que contenha os dados provenientes do banco de dados, ou *abrir* um arquivo gravado anteriormente na etapa 2. Os dados provenientes da base de dados devem estar num padrão texto, onde os campos são separados por “;”. Depois de selecionado o arquivo de dados, deve-se preencher o nome de cada um dos parâmetros e selecionar o tipo de cada campo, qualitativo ou numérico, o sistema reconhece automaticamente o tipo dos campos, mas o usuário tem a liberdade de alterá-los. Após, deve-se selecionar qual dos campos é o de saída. No nosso caso o campo que contenha o tempo total da intervenção.

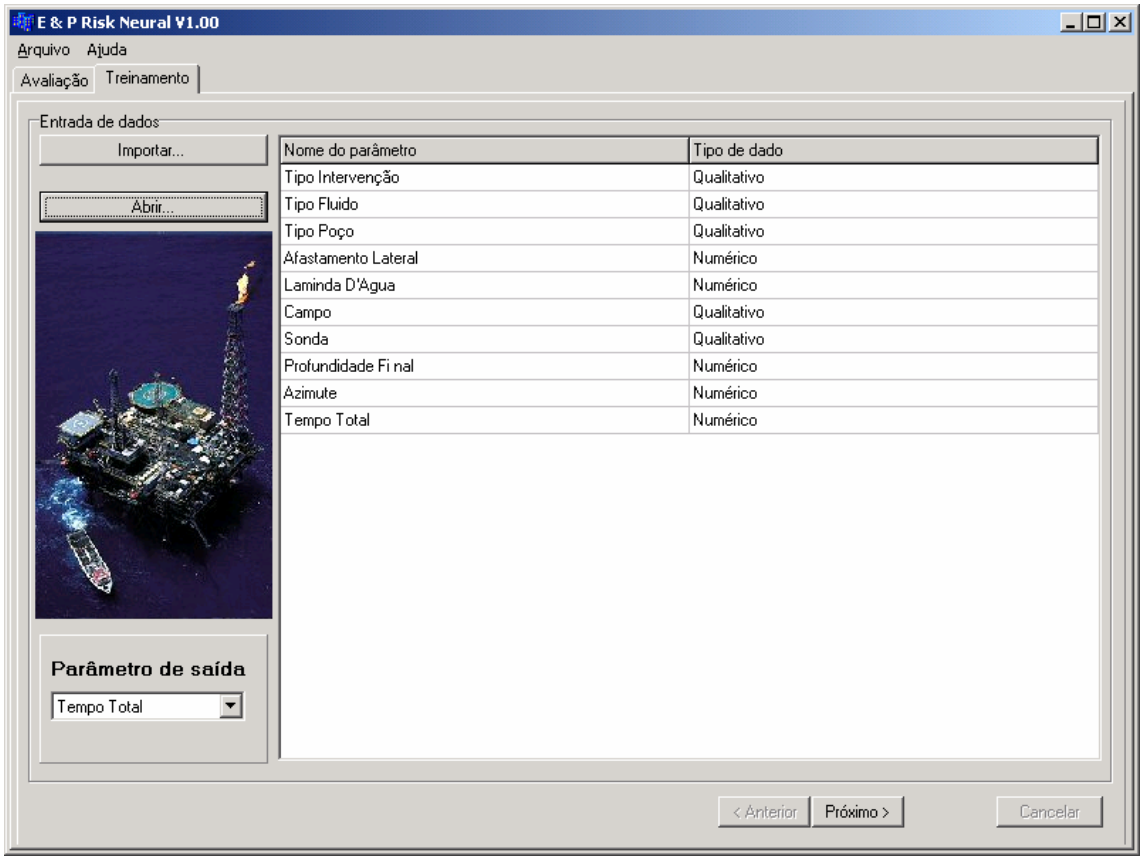


FIGURA 4.4 – Tela da etapa 1, Entrada de Dados

Etapa 2 – analise de dados: essa etapa é responsável pela analise de dados. A análise dos dados é feita automaticamente e visa excluir os dados incompletos e alertar o usuário de tipos que aparecem em menos de 1% dos dados. Este valor de 1% é meramente informativo e visa alertar o usuário que certo tipo pode não ser representativo.

Nesta etapa o usuário pode *salvar os dados* para na próxima vez que decidir treinar com essa mesma base, não precisar rotular os parâmetros novamente.

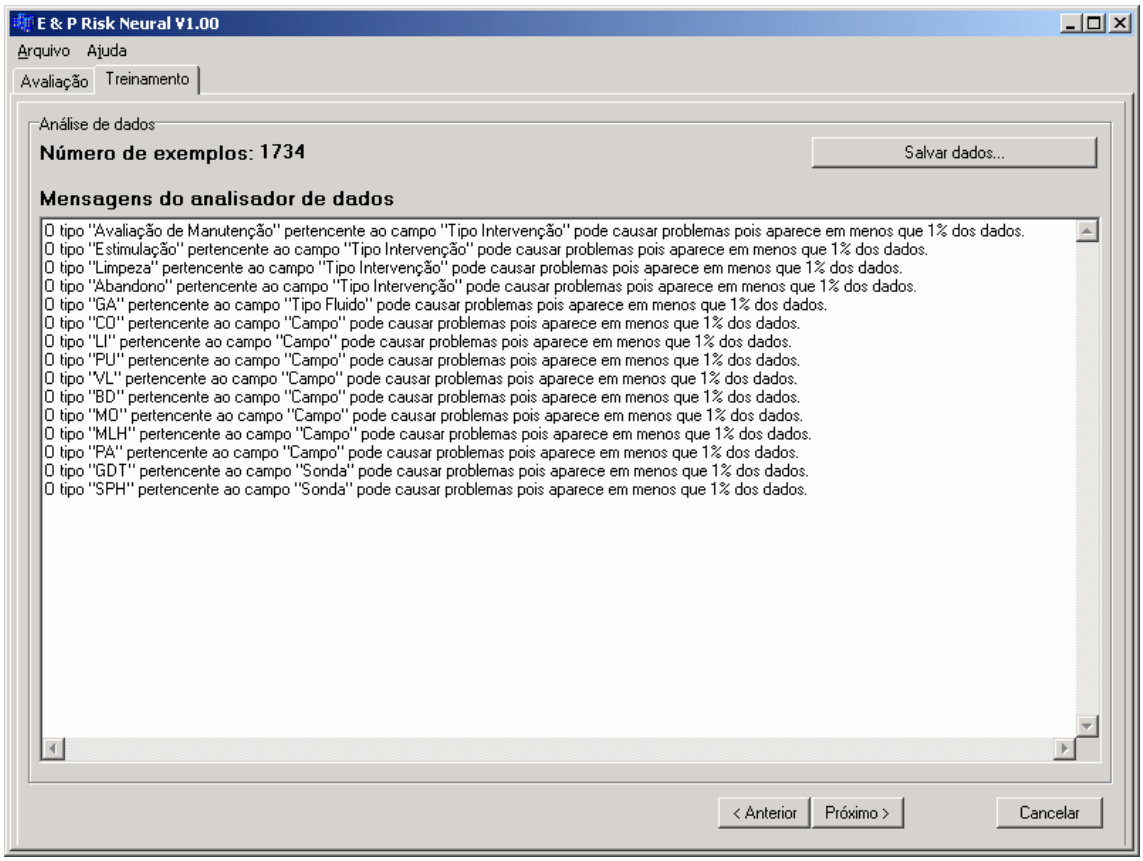


FIGURA 4.5 – Tela da etapa 2, Análise de dados

Etapa 3 – parâmetros de treinamento: nesse passo deve-se selecionar os parâmetros de treinamento. Existem dois modos, um modo básico e um modo avançado. No modo básico o usuário pode somente setar o número de épocas de treinamento da rede competitiva e o número de épocas de treinamento da rede direta. No modo avançado pode-se, além do que é possível no modo básico, setar o número de neurônios de saída, alfa da rede competitiva e número de neurônios na camada intermediária, alfa, eta, gain da rede direta. Esses parâmetros serão utilizados no próximo passo. Normalmente esses campos vêm selecionados com valores padrão, o número de neurônios na rede competitiva é calculado automaticamente como sendo 5% do número total de dados. Os outros campos estão preenchidos com valores utilizados na realização dos testes de validação da ferramenta.

Os outros campos estão selecionados com valores que, quando utilizados na base de dados de validação, obtiveram valores aceitáveis de ERRO, em torno de 20h para os parâmetros média e desvio padrão, com um tempo de treinamento pequeno, em torno de 10 minutos na máquina utilizada para testes.

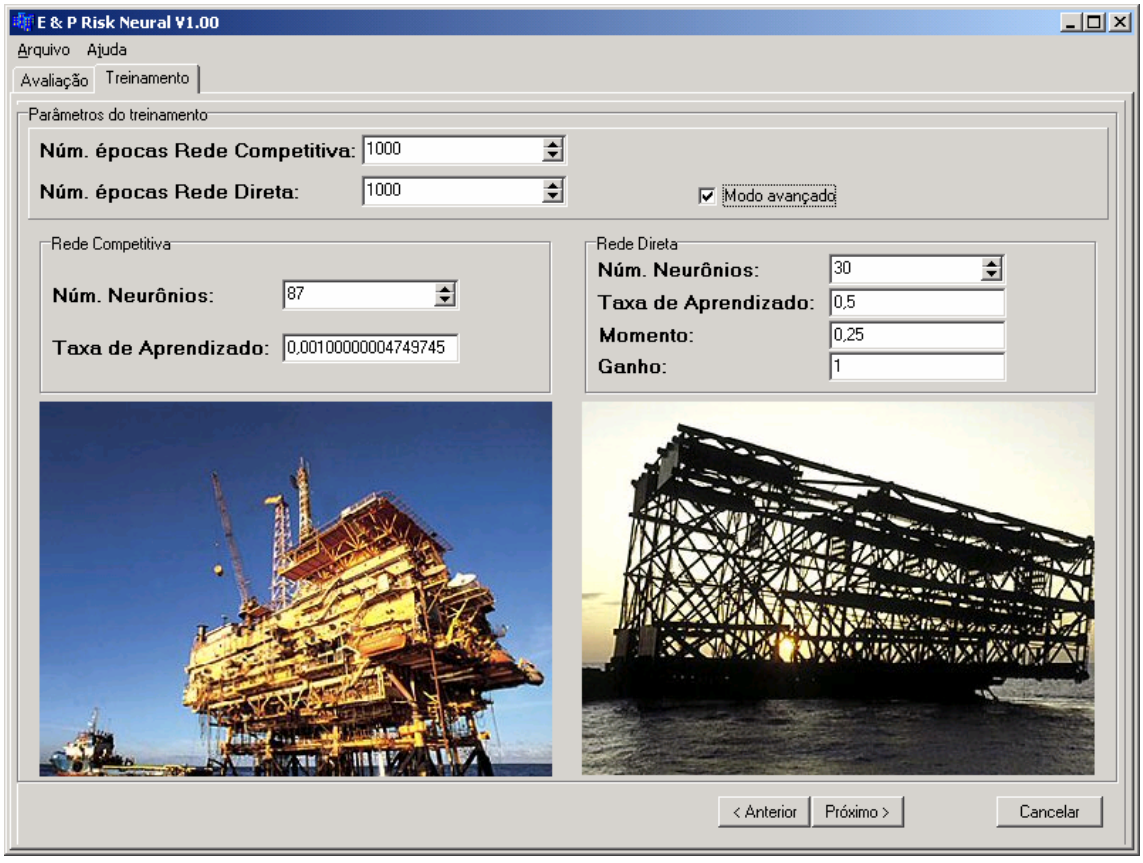


FIGURA 4.6 – Tela da etapa 3, Parâmetros de treinamento

Etapa 4 – treinamento: nesse passo, as redes neurais são treinadas levando em consideração os parâmetros de treinamento selecionados no passo anterior. Sendo possível a qualquer momento, parar o treinamento de uma das redes e iniciar o treinamento da próxima ou finalizar o treinamento.

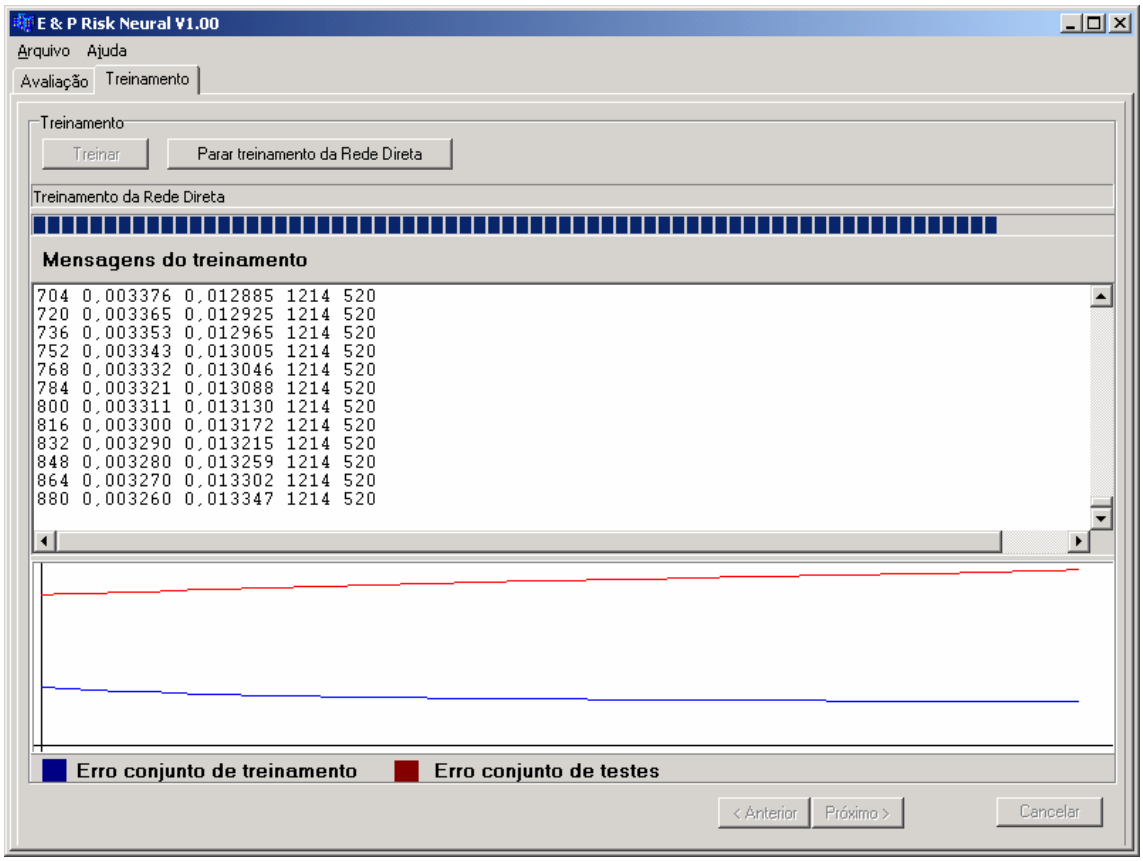


FIGURA 4.7 – Tela da etapa, 4 Treinamento

Etapa 5 – relatórios: neste passo são apresentados os relatórios de treinamento de cada uma das duas redes. Para a rede competitiva é exibido o número de exemplos classificados em cada conjunto, além do bias para o conjunto e o raio do conjunto. Para a rede direta são apresentados: erro médio quadrático, erro médio para o parâmetro Tempo Total, erro médio para o parâmetro Média e erro médio para o parâmetro Desvio Padrão. Isso para o conjunto de testes e para o conjunto de treinamento. Além disso, são exibidos três gráficos um para cada um dos parâmetros. Os gráficos desenham pontos, que possuem na sua coordenada x o valor real e na coordenada y o valor simulado para cada exemplo do conjunto de treinamento, em azul e do conjunto de testes, em vermelho. Além disso, nesse passo pode-se exportar um relatório em Excel e salvar as redes treinadas para sua utilização na previsão de um novo poço.

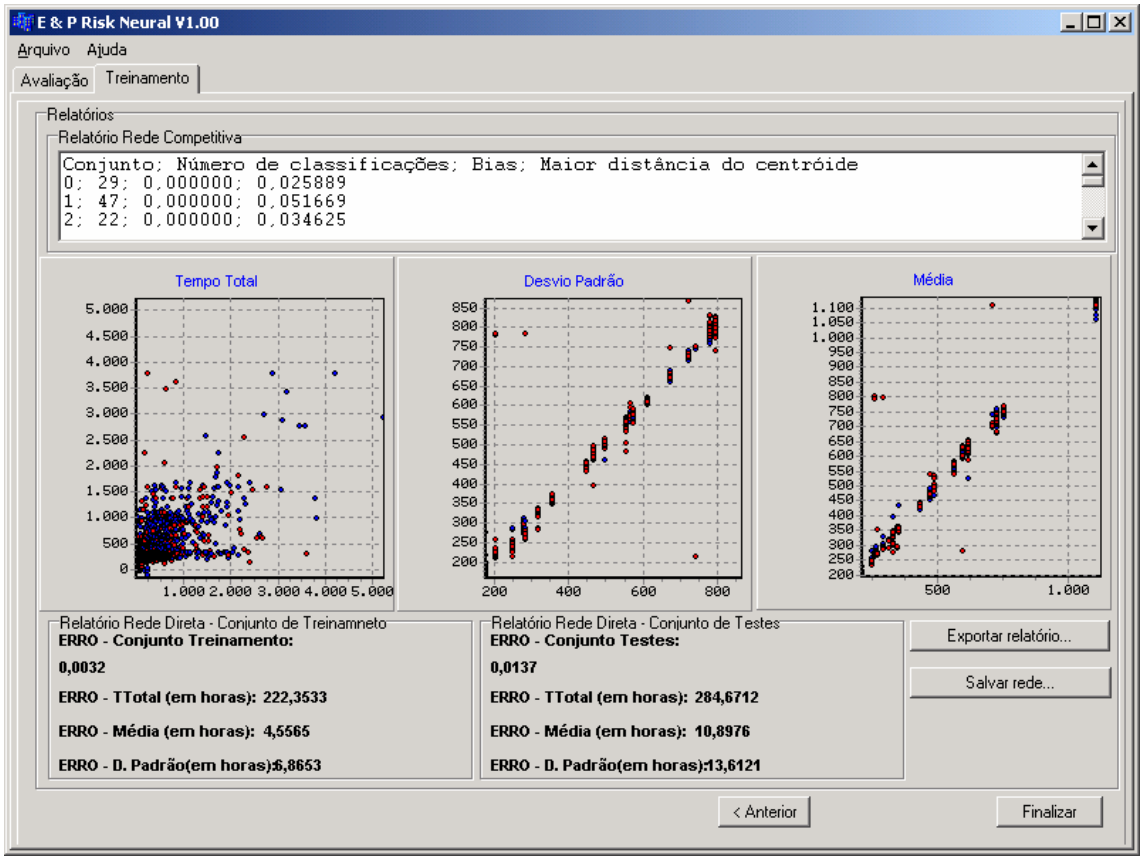


FIGURA 4.8 – Tela da etapa 5, Relatórios

4.4.3.2 Interface de Avaliação

Tendo treinado previamente as redes do sistema e gravado um arquivo na etapa 5 do treinamento com os dados referentes às redes, pode-se utilizar a interface para avaliar a ação de uma nova operação. Para avaliar esta nova operação deve-se carregar um arquivo previamente gravado com os dados das redes e em seguida deve-se preencher o valor dos parâmetros pedidos. Executando as redes através do botão *Executar redes* o sistema apresenta o tempo total, a média e o desvio padrão para a operação cujos dados foram preenchidos. Pode-se também gerar um relatório que conterá os dados da operação como também os resultado das redes.

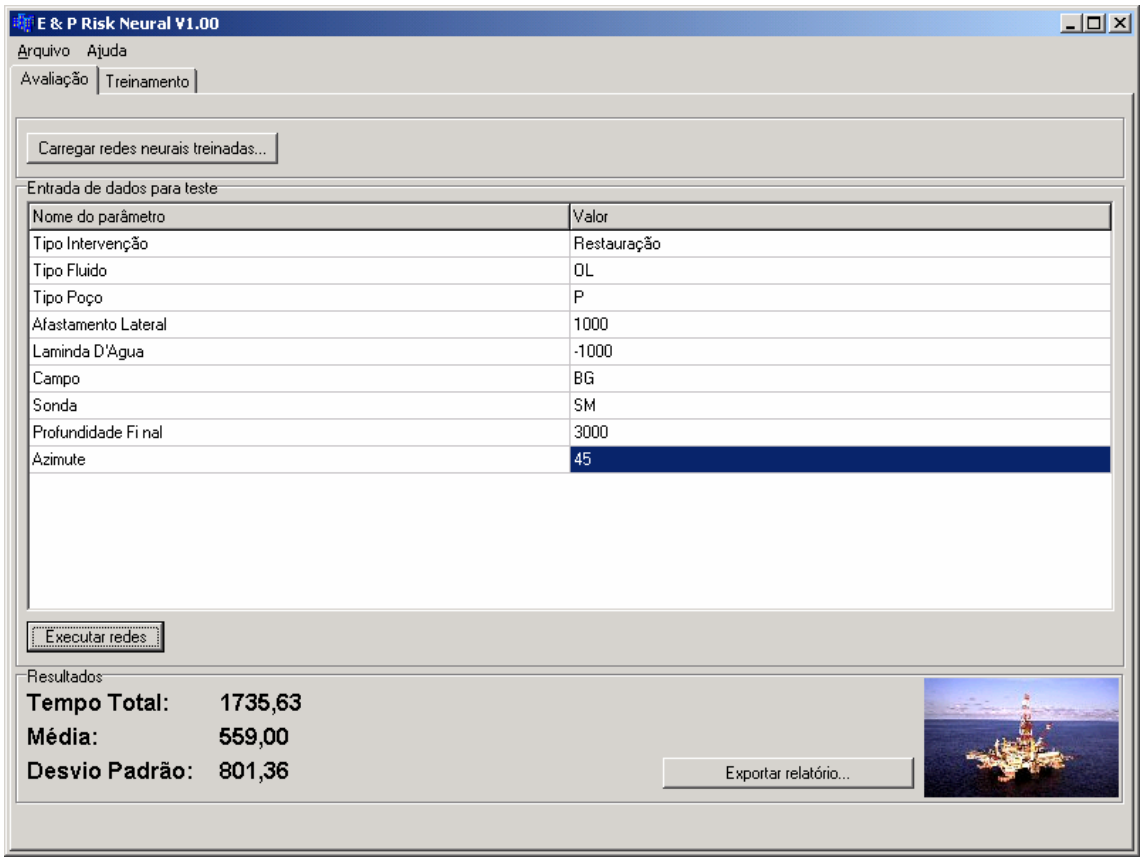


FIGURA 4.9 – Avaliação

CAPÍTULO 5 - TESTES E VALIDAÇÃO

Nesse capítulo, será apresentada uma descrição dos testes realizados durante a validação do sistema, bem como os resultados alcançados. Na realização destes testes foi utilizada a última versão do sistema implementado, bem como várias tabelas de dados retirados da base cedida por uma empresa do setor petrolífero. Este conjunto de testes foi realizado visando validar a solução adotada e observar eventuais pontos deficientes para futuras melhorias.

5.1. Dados para os Testes

Os arquivos contendo os dados utilizados nos testes foram gerados a partir da base de dados hsitóricos fornecida pela empresa e já descrita no capítulo anterior. Filtrando esta tabela foram obtidos os seguintes campos relevantes aos testes:

1. TipoIntervenção: possui o tipo da intervenção.
2. FluidoPoco: possui o Tipo de Fluido do poço.
3. TipoPoco: possui o tipo do poço, normalmente P (perfuração) e I (Injeção)
4. AfastamentoAlvo: possui o afastamento lateral entre a boca do poço e o reservatório de petróleo.
5. LaminaDagua: possui a altura da lâmina d'água no local de perfuração.
6. CamPoco: possui a sigla do campo onde o poço está inserido.
7. TipoSonda: possui a sigla que indica o tipo da sonda utilizada.
8. ProfFinalSondador: possui a profundidade final atingida pelo sondador.
9. AzimutePoco: posui o azimute do poço.
10. Total: possui o tempo total gasto na intervenção.

A partir dessa tabela foram gerados arquivos no formato texto que possuíam alguns desses campos e seus dados respectivamente. Os arquivos gerados foram,

Arquivo 1: possui os campos 1,2,3,4,5,6,7,8,9 e10.

Arquivo 2: possui os campos 1,4,5,6,7,8,9 e10.

Arquivo 3: possui os campos 1,4,5,6,7,8 e10

Inicialmente, foi criado o Arquivo 1, que possuía todos os campos considerados relevantes, com este arquivo foram realizados testes que nos levaram a perceber que muitos dados eram descartados pois seus campos não estavam preenchidos. Um dos maiores problemas ocorriam com o tipo de intervenção “perfuração exploratório”, pois os campos “FluidoPoco” e “TipoPoco” normalmente não eram preenchidos. O campo “FluidoPoco” não foi preenchido para esse tipo de intervenção porque durante a realização de uma perfuração exploratória não se sabe ao certo que tipo de fluido será encontrado. O campo “TipoPoco” também não fora preenchido em nenhuma situação para esse tipo de intervenção. Assim sendo, decidiu-se gerar um novo arquivo, o Arquivo 2, retirando-se os campos “TipoPoco” e “FluidoPoco”.

Enquanto se realizava a etapa de testes utilizando o Arquivo 2, percebeu-se que uma grande quantidade de dados era excluída do treinamento devido o não preenchimento do campo azimute. Dessa maneira, foi criado um terceiro arquivo para testes, o Arquivo 3, este possui o maior número de dados válidos para o processo de treinamento e validação.

Tabela 5.1 – Número de dados pertencentes aos arquiovos de dados

Arquivo de Dados	Número Inicial de Dados	Número Dados Excluídos	Número Dados Restantes Para Treinamento
Arquivo 1	3050	1316	1734
Arquivo 2	3050	849	2201
Arquivo 3	3050	187	2863

Na Tabela 5.1 está listado o número de dados excluídos para cada um dos arquivos de dados. Além disso, tem-se o número inicial de dados em cada tabela e o número final de dados para treinamento.

5.2. Simulações

Com base nos arquivos de dados gerados, foi realizada uma série de simulações onde os parâmetros de entrada do Sistema foram alterados buscando determinar o valor

dos parâmetros que leva ao melhor resultado final. Assim sendo, após a realização dos testes é feita uma análise de seus resultados.

Tabela 5.2 – Parâmetros referentes ao treinamento nas simulações realizadas e erro médio para cada parâmetro de saída

Número da Simulação	Arquivo Utilizado	Rede Competitiva		Rede Direta		Erro médio (em horas)		
		Neurônios	Épocas	Neurônios	Épocas	Tempo Total	Média	Desvio Padrão
1	1	87	1000	30	1000	287.29	30.34	46.14
2	1	87	1000	60	5000	285.04	25.41	34.42
3	1	30	1000	30	1000	282.38	13.08	18.27
4	2	111	1000	30	1000	275.34	30.71	45.49
5	2	111	1000	60	5000	288.69	25.21	36.24
6	2	30	1000	30	1000	299.02	9.96	15.70
7	3	144	1000	30	1000	342.85	45.94	59.48
8	3	144	1000	60	5000	344.32	33.55	45.17
9	3	30	1000	30	1000	306.21	12.25	10.25

A tabela 5.2 descreve as simulações realizadas, as informações relevantes ao treinamento das redes e os erros obtidos em cada um dos parâmetros de saída.

Tendo realizado as simulações descritas na Tabela 5.2, foram gerados arquivos contendo os dados referentes a cada simulação realizada. Um dos arquivos gerados possui os dados referentes a todas intervenções utilizadas para teste e validação. Entre estes dados estão tempo simulado, tempo real e erro para os três parâmetros de saída: Tempo Total, Media e Desvio Padrão. O erro é calculado de acordo com a fórmula apresentada na Equação 5.1.

E = |TS-TR| onde

E = Erro

TS = Tempo Simulado

TR = Tempo Real

Equação 5.1 – Equação do Erro

Como pode ser visto na tabela 5.2, o maior erro está contido no parâmetro tempo total. Este erro deve-se, principalmente, à grande variabilidade dos valores de tempo total do conjunto de treinamento.

Outro aspecto interessante a ser analisado é o que acontece quando se aumenta o número de neurônios na rede direta e a quantidade de épocas de treinamento. Este aumento faz com que o sistema diminua o erro para média e para o desvio padrão, como pode ser visto comparando as simulações 1 e 2, 4 e 5, 7 e 8. Porem isso só acontece para tempo total na comparação entre 1 e 2, nos demais casos, essa expectativa não se consolida. O motivo deste aumento é que o Tempo Total não obedece a uma função em relação as variáveis de entrada, ou seja, exemplos com parâmetros de entrada exatamente iguais geram tempos de perfuração bastante diferentes. Assim sendo, um número maior de épocas de treinamento causa uma diminuição do erro para o conjunto de treinamento, mas piora o conjunto de testes e validação. Isto ocorre, pois a rede aprende valores específicos, mas não aprende a função que rege os valores do Tempo Total.

As simulações 3, 6 e 9 foram as que tiveram melhor desempenho para seus respectivos arquivos de dados. Este desempenho foi atingido através da diminuição do número de neurônios competitivos (as simulações 3, 6 e 9 são realizadas com 30 neurônios na rede competitiva) e vem acompanhado de uma perda de precisão do resultado final. Uma análise mais detalhada destes efeitos pode ser vista no tópico a seguir.

5.3 Análise Detalhada

Para uma análise mais detalhada foram selecionadas as simulações 8 e 9 pelos seguintes motivos:

a) Nestas simulações foram utilizados os conjuntos de dados que possuíam o maior número de dados para facilitar a visualização dos efeitos causados pelos parâmetros de treinamento.

b) Estes experimentos possuem características que devem ser analisadas para um melhor entendimento do funcionamento do sistema.

Um dos aspectos mais importantes a ser analisado encontra-se na comparação entre as simulações 8 e 9. A simulação 9 tem um erro médio para os parâmetros Média e Desvio Padrão muito menor do que os da simulação 8, mas isso não quer dizer que as redes treinadas na simulação 9 resolvam o problema de maneira melhor do que as treinadas pela simulação 8. O problema é que a diminuição do erro na simulação 9 é baseada na diminuição do número de neurônios na camada competitiva. Essa ação gera agrupamentos de dados maiores o que faz com que esses agrupamentos possuam grandes desvios padrões e medias próximas às médias globais, ou seja, a média de todos os dados apresentados ao Sistema.

Seguindo a idéia de redução do número de conjuntos para obter um maior desempenho, chegaríamos a uma situação em que teríamos um único conjunto, e sua média e desvio padrão seriam iguais à média global do sistema. Assim sendo, todos os poços pesquisados retornariam o mesmo resultado e os parâmetros média e desvio padrão possuiriam um erro próximo a zero.

Então a melhor forma de configurar o sistema é buscar um número de neurônios na rede competitiva que gere agrupamentos pequenos, mas em contra partida não gere agrupamentos de um único individuo. Desta forma, o erro não será tão baixo mas o sistema possuirá mais capacidade de diferenciação entre os poços testados. A melhor maneira de diminuir o erro nos dois parâmetros analisados é até certo ponto o aumento do número de neurônios na rede direta.

A seleção de um número ótimo de agrupamentos é um problema complexo que vem sendo abordado em dissertações e teses de doutorado como de SCREMIN (2003), JOHSON e WICHER (1998) e MORRISON (1990).

As Fig. 5.1 e 5.2 apresentam os gráficos referentes ao parâmetro média das simulações 8 e 9. Nota-se que na simulação 9 existem muito menos valores de média.

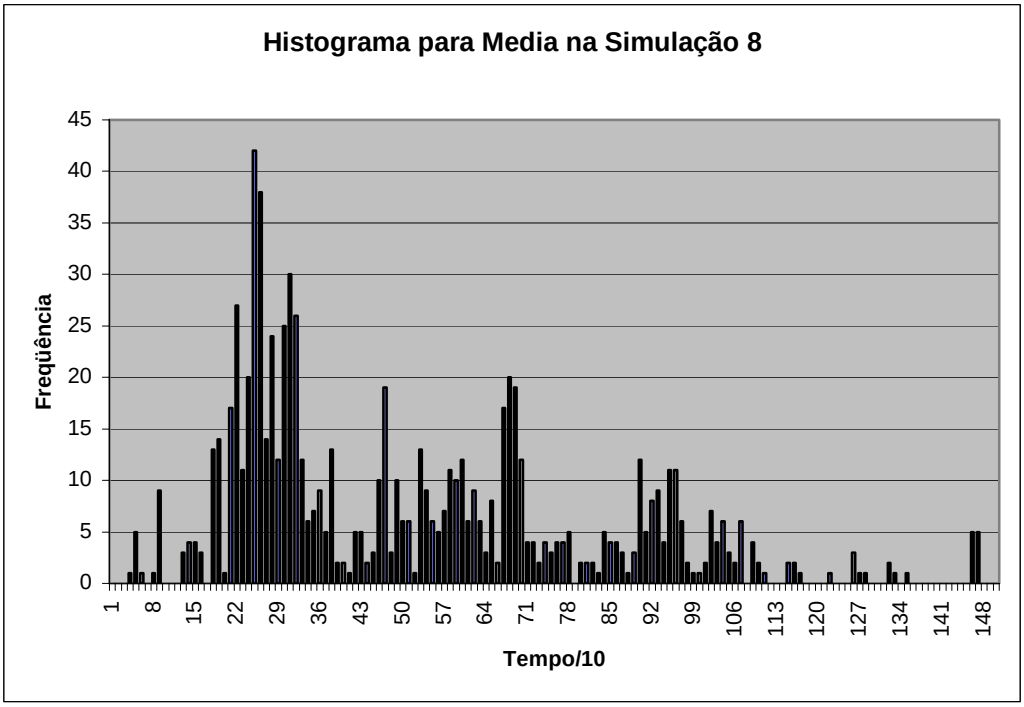


FIGURA 5.1 – Histograma de distribuição dos dados no tempo. O no eixo x estão valores de tempo/10, ou seja cada unidade representa 10 horas, e em y a freqüência com que aparecem nos dados utilizados para teste e validação.

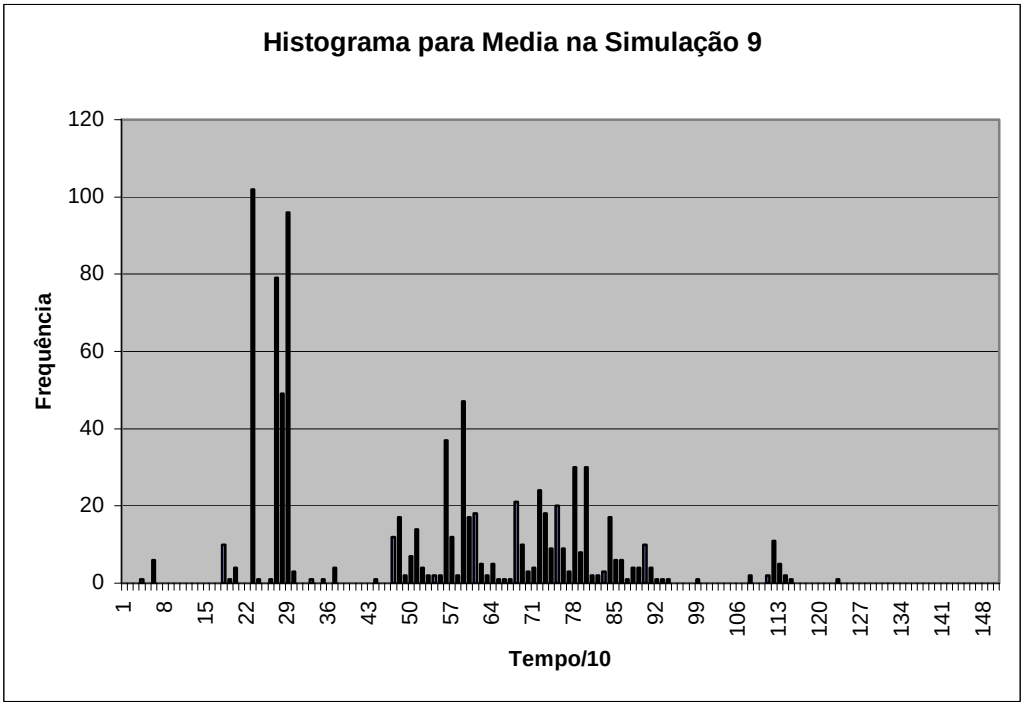


FIGURA 5.2 – Histograma de distribuição dos dados no tempo. O no eixo x estão valores de tempo/10, ou seja cada unidade representa 10 horas, e em y a freqüência com que aparecem nos dados utilizados para teste e validação.

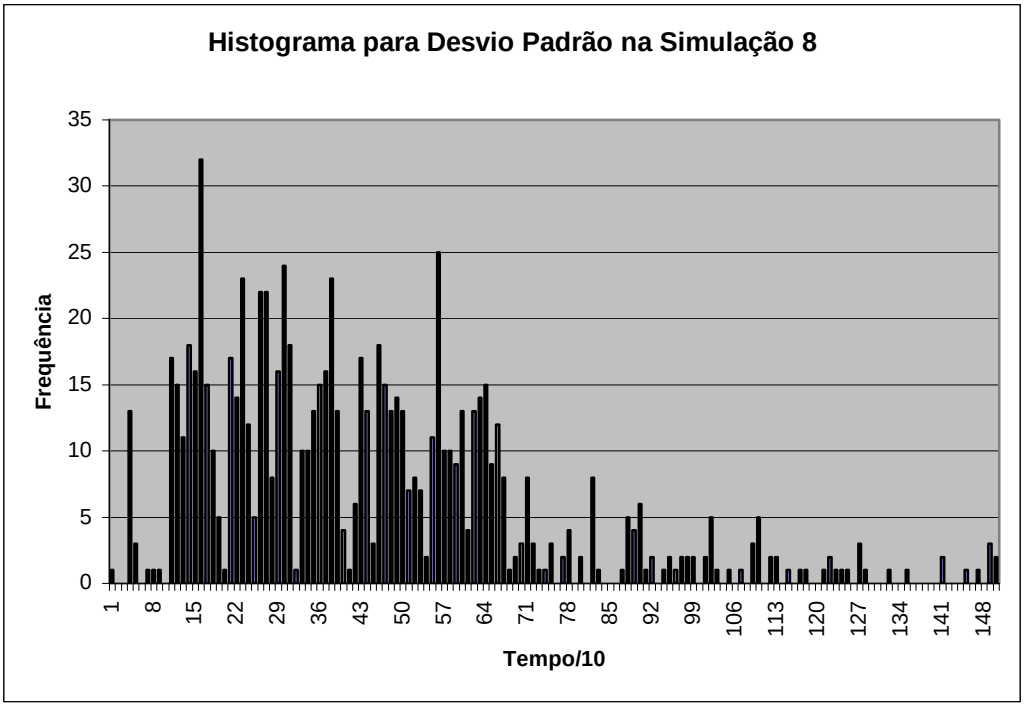


FIGURA 5.3 – Histograma de distribuição dos dados no tempo. O no eixo x estão valores de tempo/10, ou seja cada unidade representa 10 horas, e em y a freqüência com que aparecem nos dados utilizados para teste e validação.

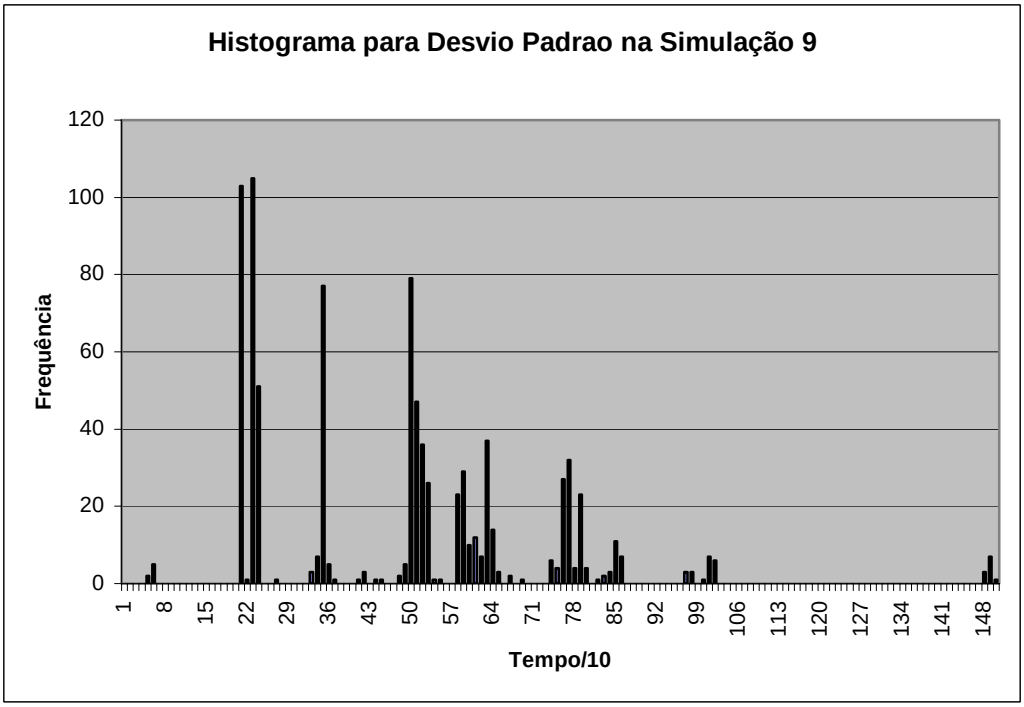


FIGURA 5.4 – Histograma de distribuição dos dados no tempo. O no eixo x estão valores de tempo/10, ou seja cada unidade representa 10 horas, e em y a freqüência com que aparecem nos dados utilizados para teste e validação.

As Fig. 5.3 e 5.4 apresentam os gráficos referente ao parâmetro desvio padrão das simulações 8 e 9. Nota-se que na simulação 9 existem poucas faixas de valores que concentram quase todos os resultados para variável desvio padrão. Além disso, quase não existem valores de desvio padrão pequenos na simulação 9.

É possível notar nas Fig. 5.1 e 5.2 que os resultados referentes à simulação 9 estão mais concentrados em faixas pequenas de valores. Isso tem como causa a pequena quantidade de conjuntos utilizada o que faz com que muitas intervenções diferentes sejam reconhecidas como pertencentes ao mesmo conjunto, já que esses conjuntos na simulação 9 são grandes. Já nos gráficos referentes à simulação 8, os resultados são mais variados pois os conjuntos são menores e assim só classificam dados realmente similares.

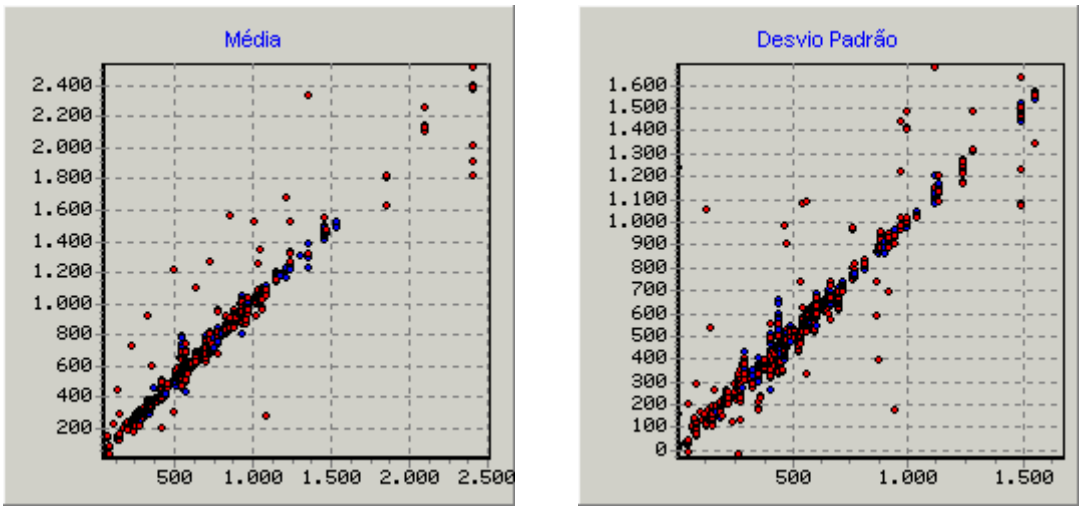


FIGURA 5.5 – Gráficos de distribuição de resultados para o teste 8. Os círculos que possuem na coordenada x, o tempo real, para a dada intervenção, e em y, o tempo simulado; os círculos azuis representam as intervenções pertencentes ao conjunto de treinamento, e em vermelho as pertencentes ao conjunto de teste.

Na Fig. 5.5, são apresentados gráficos para a média e o desvio padrão na simulação 8, neste tipo de gráfico, agrupamentos horizontais representam conjuntos. Como o número de conjuntos é grande e eles estão bem distribuídos, pois essa é uma característica do problema, quase não é possível notar-se agrupamentos. Neste tipo de gráfico os grupos (*Clusters*) podem ser visualizados como agrupamentos verticais de pontos, pois todos os elementos de um grupo possuem aproximadamente um mesmo

valor de saída, ocorrendo somente uma pequena variação que depende de sua distância em relação ao centróide.

Nesse tipo de gráfico, os pontos com menor erro estão dispostos sobre a diagonal, sendo que, quanto mais distante um ponto estiver da diagonal maior o seu erro.

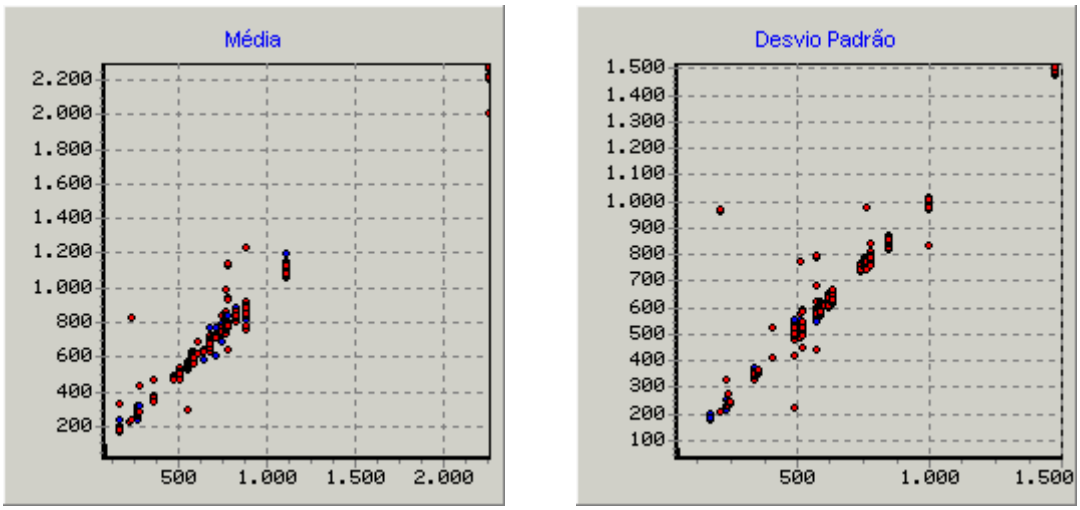


FIGURA 5.6 – Gráficos de distribuição de resultados para o teste 9. Os círculos que possuem na coordenada x, o tempo real, para a dada intervenção, e em y, o tempo simulado; os círculos azuis representam as intervenções pertencentes ao conjunto de treinamento, e em vermelho as pertencentes ao conjunto de teste.

Já na Fig. 5.6, que representa a simulação 9, pode-se notar distintamente os agrupamentos. Além disso, todos os dados estão concentrados em poucos valores, o que não reflete a grande variabilidade característica do problema.

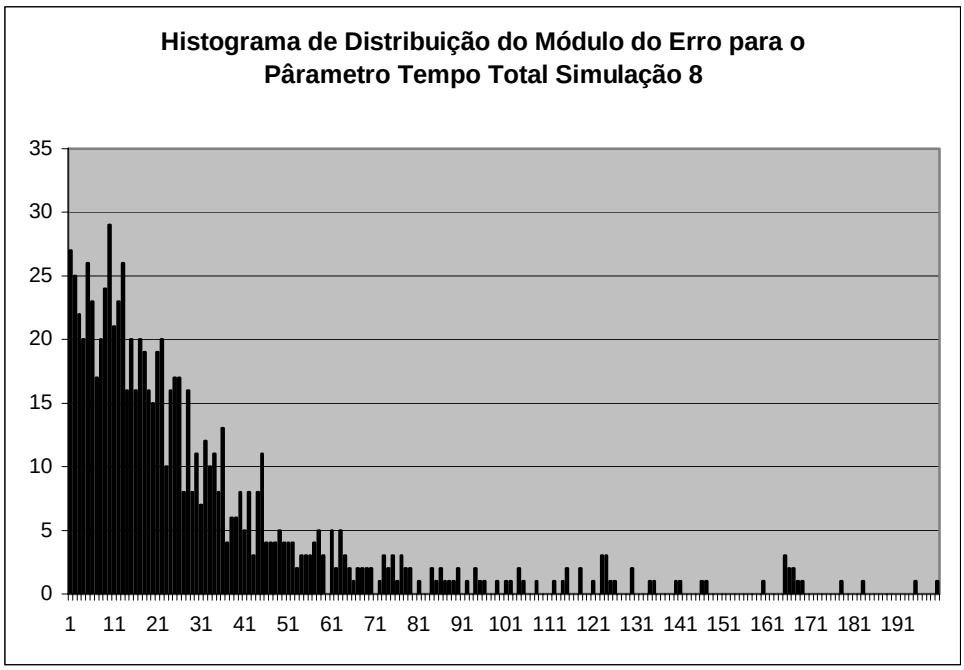
5.4 Análise de Erro

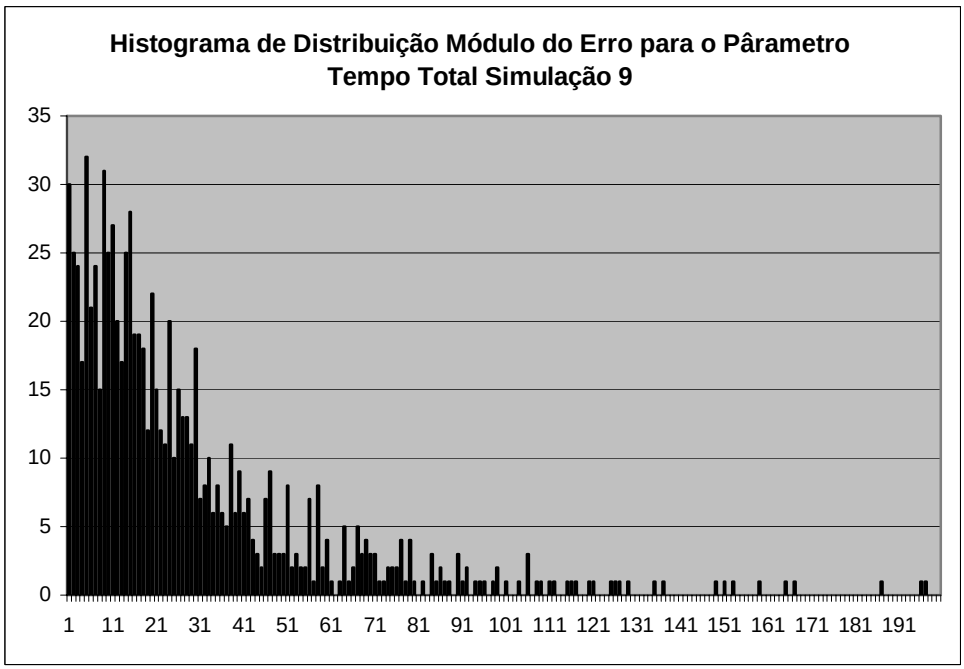
Na literatura de redes neurais existem poucos estudos que abordam o comportamento aleatório do erro. Este assunto quando tratado na literatura é abordado de forma superficial e em problemas específicos.

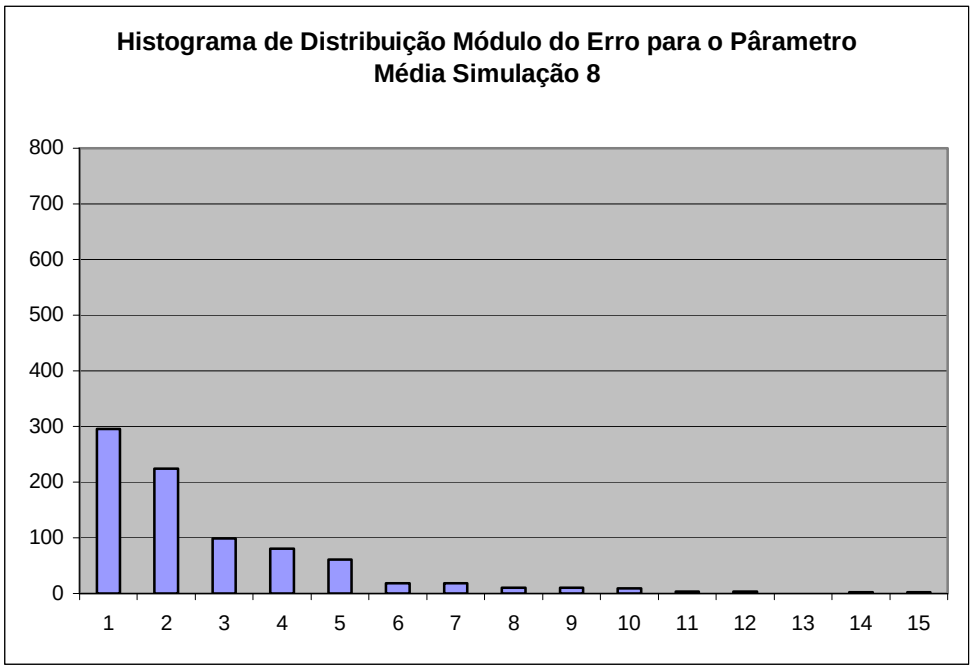
Dentro do Sistema, descobrir como é a distribuição do erro, é de suma importância por tratar-se de um sistema de análise de risco. Como na literatura não foi possível encontrar uma fórmula que determine a distribuição de erro nas duas redes neurais utilizadas, partimos para um método de análise das simulações. Para realizar

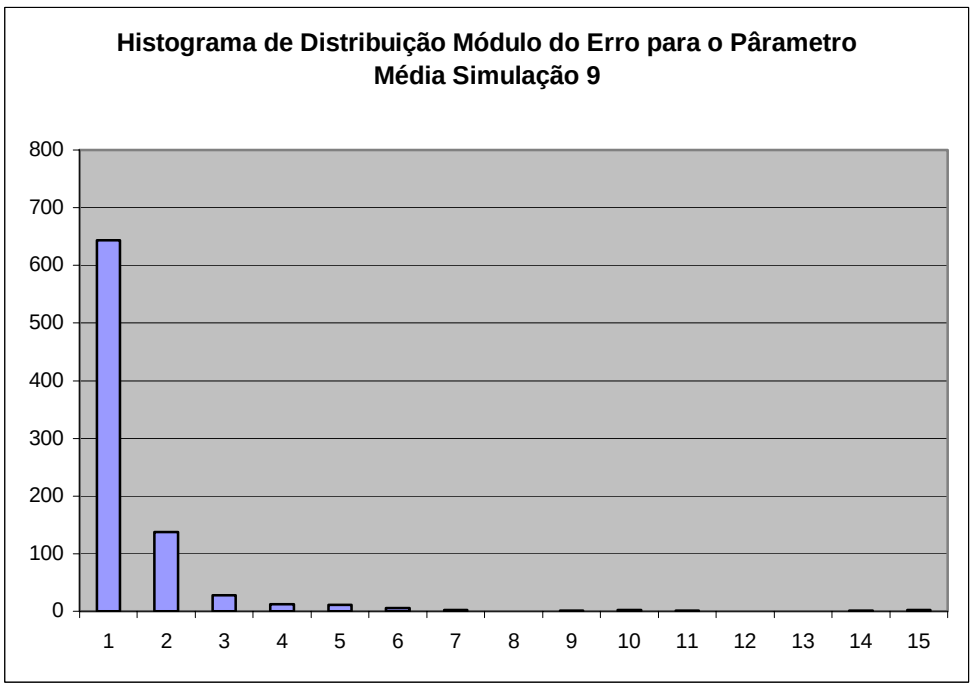
esta analise, foi utilizada uma ferramenta que realiza um teste de aderência a vários tipos de funções estatísticas.

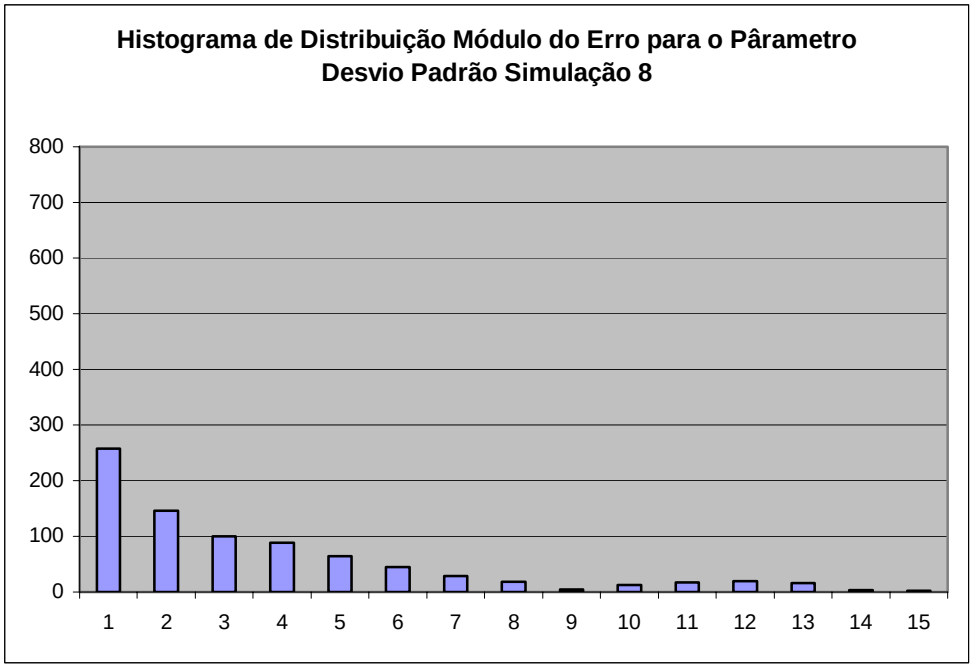
Para uma melhor visualização das simulações todos os erros dos testes foram agrupados em faixas de 10 horas. Com o resultado desse agrupamento foram gerados os gráficos que estão na Fig. 5.7, estes foram gerados para os três parâmetros de saída: Tempo Total, Média e Desvio Padrão.






Histograma de Distribuição Módulo do Erro para o Pârametro
Média Simulação 8
800
700
600
500
400
300
200
100
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15


Histograma de Distribuição Módulo do Erro para o Pârametro
Média Simulação 9
800
700
600
500
400
300
200
100
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15



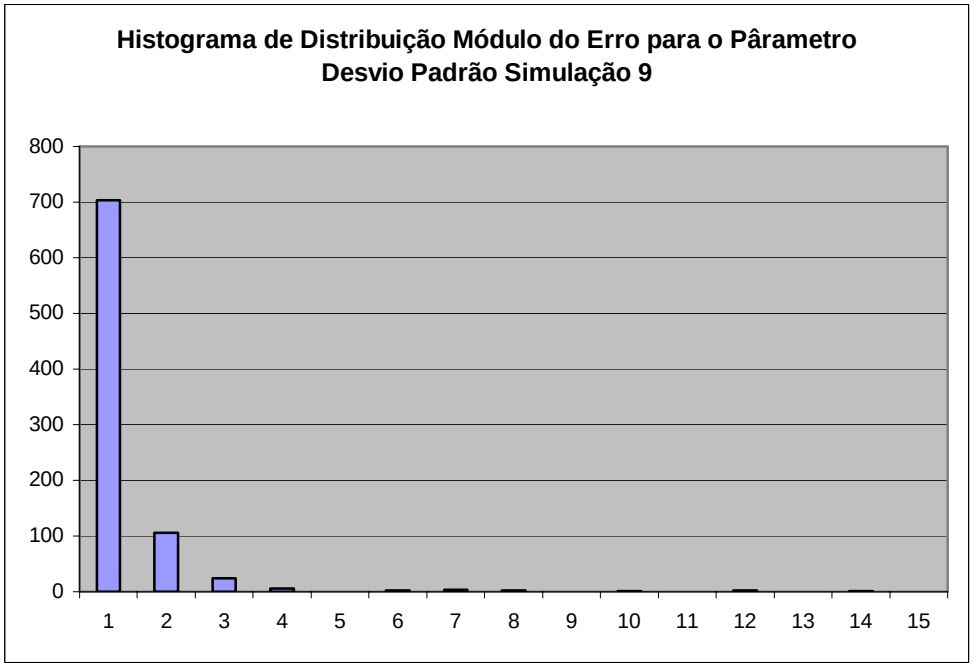


FIGURA 5.7 – Histogramas de distribuição do erro nas simulações 8 e 9. O no eixo x estão valores de tempo/10, ou seja cada unidade representa 10 horas, e em y a freqüência com que tal erro ocorre nos dados utilizados para teste e validação.

Para todos os casos analisados, a ferramenta de aderência retornou que a forma da função do erro era exponencial. Saber a forma da função do erro facilita para que num

trabalho futuro sejam estudadas uma fórmula do erro embutido no sistema e sua influência em cada resultado.

Sendo a forma da distribuição do erro uma exponencial, o usuário do sistema pode ter uma idéia do erro embutido nos resultados do sistema. Mas em trabalhos futuros, a fórmula do erro deve ser estudada mais detalhadamente, podendo-se criar uma maneira formal de se avaliar o erro embutido no sistema.

5.5 Testes com Dados Reais

Para finalizar a análise dos resultados, foram realizados testes utilizando dados reais, dados estes não utilizados no treinamento. Tais simulações foram realizadas sobre as redes treinadas na Simulação 8.

Tabela 5.4 – Testes realizados com dados reais

N°	Tipo da Intervenção	Afast. Lateral (Metros)	Lamina D'Água (Metros)	Campo	Sonda	Prof. Final (Metros)	Tempo Real (Horas)	Tempo Simulado (Horas)	Média (Horas)	Desvio Padrão (Horas)
1	Perfuração de Desenvolvimento	319,84	-960,00	MRL	SS	2905,00	629,50	818,79	706,11	335,03
2	Perfuração Exploratória	0,00	-322,00	AB	SS	2979,00	629,50	731,29	786,8	179,55
3	Completação	2372,75	-142,00	CH	SM	4188,00	455,50	464,28	438,58	294,01
4	Perfuração Exploratória	0,00	-118,00	RJS	SS	2780,00	703,50	940,93	962,44	655,31

Na tabela 5.4 estão dispostos os dados de intervenções reais que utilizadas bem como os resultados obtidos pelo sistema. Os campos *Intervenção, Afastamento Lateral, Lamina D'Água, Campo, Sonda e Profundidade Final* são os parâmetros da entrada do sistema. O campo *Tempo Real* refere-se ao tempo de duração real para a dada intervenção. Os campos *Tempo Simulado, Média e Desvio Padrão* possuem os retornos obtidos pela simulação.

Ao analisar-se os testes contidos na tabela 5.4, observa-se que os valores de tempo total simulado estão sempre próximos a média, sendo esta uma característica do sistema devido à grande variabilidade do problema.

Pode-se ver na tabela 5.4 que para a simulação 1 o tempo previsto para a intervenção foi de 818,79 horas, com média de 706,11 horas e desvio padrão de 335,03

horas. Seu tempo real da intervenção foi de 629,50 horas que é satisfatório num tipo de problema que se caracteriza pela grande variabilidade Se uma análise de confiabilidade fosse feita, sendo adotada uma margem de confiabilidade de 90% o valor real deveria estar entre 613,23 e 798,98 horas. Assim sendo, pode-se considerar satisfatório o resultado obtido pelo Sistema. Para o cálculo deste intervalo de confiança, foi utilizada a Equação 5.2.

$$T = M \pm \frac{(1.64 * \delta)}{\sqrt{n}}$$

Onde:

T é o intervalo de confiança

M é a média

DP é o desvio padrão

N é o número de elementos do *cluster*

Equação 5.2 – Intervalo de Confiança de 90%.

Na tabela 5.5 estão calculados os intervalos de confiança para cada um dos testes. Comparando-se a tabela 5.5 com a tabela 5.4 pode-se perceber que somente em um caso o Tempo Real não caiu dentro do intervalo de Confiança de 90%, mas esse fato não invalida o bom resultado obtido.

Tabela 5.5 – Intervalo de Confiança de 90% Para Cada Teste

Nº do Teste	Nº de elementos no *cluster*	Intervalo de Confiança 90%	
		Inicio	Fim
1	35	613,2361	798,9839
2	2	578,5839	995,0161
3	4	197,4918	679,6682
4	44	800,4216	1124,458

CAPÍTULO 6 - CONSIDERAÇÕES FINAIS

6.1. Conclusão

Durante a fase inicial do projeto foi realizada uma extensa bateria de testes visando à identificação de possíveis modelos a serem utilizados no Sistema. Esta fase teve uma contribuição muito grande para com as etapas posteriores, pois a análise de vários modelos, mesmo que estes não fossem utilizados no Sistema final, forneceu um bom conhecimento do problema. Esse conhecimento foi revelado pelo comportamento dos dados reais utilizados durante esta etapa de testes.

O protótipo final da etapa de testes foi utilizado como base para a arquitetura final do sistema essa arquitetura adotada foi composta por duas redes neurais, uma rede competitiva e uma rede direta sendo implementada com sucesso num Sistema de fácil utilização por usuários com conhecimentos de redes neurais. Assim sendo foi possível a realização de uma extensa bateria de testes visando a validação do sistema.

Os resultados obtidos com esses testes demonstraram que a arquitetura utilizada no sistema é capaz de prever, com um alto grau de confiabilidade os tempos de intervenções realizadas sobre poços de petróleo, mas, esse grau de confiabilidade do resultado depende de vários fatores. Alguns fatores que influenciam na confiabilidade dos resultados estão listados a seguir:

a) Volume de dados: para conseguir uma boa confiabilidade nos resultados deve-se possuir uma base de dados com o maior número de casos reais possíveis.

b) Erro nos dados: a base de dados deve possuir o mínimo possível de dados com valores incorretos, pois muitas vezes esses valores incorretos levam a contradições que dificultam o aprendizado das redes neurais.

c) Parâmetros de treinamento: uma má escolha dos parâmetros de treinamento pode prejudicar completamente no resultado final. Como foi visto no capítulo 5, uma escolha equivocada da quantidade de neurônios na rede competitiva tornou o resultado muito menos confiável

Quando os fatores listados acima são contornados, o sistema mostrou-se de grande valor na resolução do problema proposto, tendo chegado a resultados aceitáveis e coerentes com a realidade. Uma avaliação completa da veracidade dos resultados

obtidos com o sistema só poderá ser feita com a utilização deste em poços reais. Mas este fato não impede que o sistema cumpra seus objetivos, pois ele é um sistema de auxilio e seus resultados serão comparados com o de outros sistemas baseados em análises estatísticas e por especialistas no assunto.

O sistema cumpre com sucesso seus objetivos, no que diz respeito a prever os resultados de um poço de petróleo baseado em dados históricos, pois se utilizado corretamente e com parâmetros corretos realiza uma previsão precisa no que diz respeito às características dos dados utilizados no seu treinamento.

Mesmo cumprindo com sucesso seus objetivos, o sistema tem alguns problemas;

- O sistema não tem nenhuma forma de corrigir dados incompletos, ou seja, em operações em determinado campo não se aplica este não é preenchido, sendo o dado todo excluído.
- O sistema não guarda nenhuma informação sobre o treinamento no arquivo que contém os pesos da rede treinada. Assim sendo, os que tiverem acesso somente a este arquivo não tem como conhecer grau de confiabilidade da rede utilizada.
- Muitos dos parâmetros necessários para treinamento das redes dependem muito das características do conjunto de treinamento. Assim sendo dependem muito do conhecimento em redes neurais do usuário e mesmo assim são necessários vários testes até encontrar-se um valor aceitável. Isso torna o Sistema restrito a pessoas com algum conhecimento em redes neurais.

Boa parte dos problemas citada acima será resolvida caso sejam implementadas as sugestões para trabalhos futuros.

Mesmo com estes problemas o Sistema proposto nesta dissertação é uma boa alternativa aos métodos tradicionais.

6.2. Trabalhos Futuros

Durante a etapa de testes e validação, foram percebidas varias modificações que podem vir a serem feitas no sistema para tornar mais simples sua utilização além de torná-lo mais robusto e mais confiável. As possíveis melhorias ao sistema estão abaixo citadas como sugestão de trabalhos futuros.

a) **Tratamento inicial dos dados** – é necessário que o sistema possua uma maneira automática, melhor que a utilizada atualmente, para excluir dados incompletos ou incorretos e se possível preencher campos de dados onde tal campo não se aplica.

b) **Método automático de cálculo do número de neurônios na camada competitiva** - é necessário que o sistema possua um sistema automático de seleção de número de neurônios na camada competitiva. Esse sistema automático pode ser criado utilizando-se de técnicas de algoritmos genéticos.

c) **Novas informações nos arquivos de pesos das redes treinadas**: é necessária a inclusão de dados relativos ao treinamento das redes no arquivo que contem os pesos da rede, pois esse arquivo pode ser utilizado por varias pessoas, não necessariamente quem treinou a rede, assim sendo o usuário deste arquivo deve possuir uma maneira de checar o erro de treinamento embutido nos pesos utilizados.

d) **Garantir que conjuntos próximos tenham semelhanças** – no método utilizado atualmente os dados classificados em conjuntos próximos, como 1 e 2 por exemplo, não necessariamente devem possuir alguma semelhança. Para garantir essa semelhança geográfica pode-se ao invés de utilizar redes competitivas utilizar uma rede de korronen.

e) **Tempo de treinamento elevado** - o tempo de treinamento das redes para uma base de dados grande é elevado. Uma das soluções para esse problema seria a utilização de um algoritmo paralelo de treinamento, o que possibilitaria a utilização de máquinas multi-processadas ou mesmo um *cluster* para seu treinamento.

Além de todos os aspectos citados acima ainda existe um outro aspecto que pode ser abordado em trabalhos futuros, pois o sistema foi criado para resolução de um

problema de previsão de tempo de perfuração de poços de petróleo, mas nada impede o sistema ser utilizado em outros problemas de previsão. Por exemplo, o sistema pode ser utilizado no auxilio na tomada de decisão de um empréstimo bancário. Assim sendo, outras utilizações do sistema como possíveis adaptações são sugestões para trabalhos futuros.

REFERÊNCIAS BIBLIOGRÁFICAS

ALEKLETT, K. and CAMPBELL, C.J. The Peak and Decline of World Oil and Gas Production**. Published by the Association for the Study of Peak Oil and Gas**. Disponível em < www.asponews.org>. Acesso em: 25 maio 2005.

BARRETO, J. M. **Inteligência Artificial no Liminar do Século XXI.** 2 ed. Florianópolis: Duplic, 2000.

BISHOP, Christopher. **Neural Networks for Pattern Recognition**. Oxford: Clarendon Press,1995.

DAYHOFF, Judith. **Neural Network Architectures – An Introduction.** New York: Van Nostrand Reinhold Co, 1990

DOMINGOS, Luís. **Perfuração no mar.** Disponível em: <http://histpetroleo.no.sapo.pt/perf_mar.htm>. acesso em: 1 mar. 2004.

HAYKIN, Simon. **Redes Neurais Princípios e Prática**. Trad. Paulo Martins Engel. 2 ed. Porto Alegre: Bookman, 2001.

HAN, J. & KAMBER, M. **Data Mining:** Concepts and Techniques**.** Simon Fraser University: Morgan Kaufmann Publishers, 2000.

HARBAUGH, J. W., DAVIS, J. C. and WENDEBOURG . **Computing Risk for Oil Prospects:** Principles and Programs. Pergamon. UK. 1995.

JACINTO, Carlos Magno. **Modelagem e Simulação do Risco na Perfuração e Completação de Poços de Petróleo e Gás em Águas Profundas**. Dissertação (Mestrando em Ciências) – Programa de Pós-Graduação em Ciência, Universidade Federal Fluminense, Rio de Janeiro: 2002.

JOHSON, R.A.; WICHERN, D.W. **Applied multivariate statistical analysis.** Prentice-Hall: New Jersey, 1998. 3. ed.

KIMBALL, Ralph. **Data Warehouse Toolki**t. New Jersey: John Wiley & Sons, Inc, 1997.

KOHONEN, T. **Self-Organization and Associative Memory**. 2Nd Edition, Berlin: Springer-Verlag, 1987.

MORRISON, D. F. **Multivariate statistical methods**. McGraw-Hill: USA, 1990. 3. ed.

ROSE, P. Risk Analysis and Management of Petroleum Exploration Ventures. **AAPG Methods in Exploration Series**, No. 12. AAPG. USA. 2001.

RUD, Olívia Parr. **Modeling Data for Marketing, Risk, and Customer Relationship Management.** New Jersey: John Wiley & Sons, Inc, 2001.

SCHALKOFF, Robert. **Pattern Recognition**. New York: John Wiley & Sons, Inc, 1992.

SILVA, Reneu Rodrigues da. **Explorator:** Protótipo de Sistema Holístico em Exploração de Petróleo. 2000. Tese (Doutorado em Geologia) – Programa de Pós-Graduação em Geologia, Universidade Federal do Rio de Janeiro, Rio de Janeiro: 2000.

THOMAS, J. E. (Org.). **Fundamentos de Engenharia de Petróleo**. [S.l.: s. n.],1996.

SANTOS, José Gonçalo dos. **Sistema Especialista para Tipificação de Dados**. 2001. Dissertação (Mestrado em Sistema de Conhecimento) – Programa de Pós-Graduação em Ciência da Computção, Universidade Federal de Santa Catarina, Florianópolis: 2001.

SCREMIN, Marcos Antônio Antonelio. **Método para a seleção do número de componentes principais com base na lógica difusa**. 2003. Tese (Doutorado em Engenharia da Produção) – Programa de Pós-Graduação em Engenharia de Produção. Universidade Federal de Santa Catarina, Florianópolis: 2003.