A FEA e a USP respeitam os direitos autorais deste trabalho. Nós acreditamos que a melhor proteção contra o uso ilegítimo deste texto é a publicação online. Além de preservar o conteúdo motiva-nos oferecer à sociedade o conhecimento produzido no âmbito da universidade pública e dar publicidade ao esforço do pesquisador. Entretanto, caso não seja do interesse do autor manter o documento online, pedimos compreensão em relação à iniciativa e o contato pelo e-mail bibfea@usp.br para que possamos tomar as providências cabíveis (remoção da tese ou dissertação da BDTD).

UNIVERSIDADE DE SÃO PAULO
FACULDADE DE ECONOMIA, ADMINISTRAÇÃO E CONTABILIDADE
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
MESTRADO PROFISSIONALIZANTE "MODELAGEM MATEMÁTICA EM FINANÇAS"

APLICAÇÃO DE REDES NEURAIS EM RISCO OPERACIONAL: ANÁLISE DISCRIMINANTE EM PERDAS TRABALHISTAS

Andre Beraha

Orientador: Prof. Dr. Gerson Francisco

São Paulo, 2005

T006.32
B482a

DEDALUS - Acervo - FEA
20600028675

T006.32 B482a
T88289
20600028675
FEAUSP
Powered by RfidProStar - www.tagprocess.com.br

APLICAÇÃO DE REDES NEURAIS EM RISCO OPERACIONAL: ANÁLISE DISCRIMINANTE EM PERDAS TRABALHISTAS

Andre Beraha

USP - FEA - SBD
DATA DA DEFESA 07 / 11 / 05

Dissertação apresentada à Faculdade de Economia, Administração e Contabilidade e ao Instituto de Matemática e Estatística da Universidade de São Paulo para obtenção do Título de Mestre.

Orientador: Prof. Dr. Gerson Francisco
Prof. Dr. Renato Vicente

São Paulo, 2005

88289

88289

FICHA CATALOGRÁFICA

Elaborada pela Seção de Processamento Técnico do SBD/FEA/USP

Beraha, Andre
Aplicação de redes neurais em risco operacional : análise discriminante em perdas trabalhistas / Andre Beraha. -- São Paulo, 2005.
62 p.

Dissertação (Mestrado Profissionalizante) -- Universidade de São Paulo, 2005
Bibliografia

1. Redes neurais 2. Análise discriminante 3. Análise de risco
I. Universidade de São Paulo. Faculdade de Economia, Administração e Contabilidade. II. Universidade de São Paulo. Instituto de Matemática e Estatística. III. Título.

CDD – 006.32

AGRADECIMENTOS

A Deus.

À meus familiares

Ao Prof. Dr. Renato Vicente, pelo imenso apoio que me deu para o desenvolvimento deste trabalho.

Ao Prof. Dr. Gerson Francisco, pela confiança depositada em mim e neste trabalho.

Aos Professores do Mestrado Profissionalizante "Modelagem Matemática em Finanças", pela grande contribuição que deram na minha formação acadêmica ao compartilhar seus conhecimentos.

À Samya Paiva, pelos conhecimentos que adquiri com ela em relação a Risco Operacional.

Ao Banco ABN AMRO Real, que cedeu sua base de dados para este estudo.

RESUMO

Com a publicação do Novo Acordo de Capital da Basiléia, as instituições financeiras incluíram no seu processo de gerenciamento e mensuração de riscos um novo componente importante: o Risco Operacional. As instituições financeiras têm despendido nos últimos anos um esforço considerável na Modelagem do Risco Operacional, tanto em termos de Cálculo de Capital quanto em modelos quantitativos de controle e gestão desse tipo de risco específico. Neste estudo, será implementada a técnica de Redes Neurais sobre uma importante categoria de evento de Risco Operacional denominada Práticas Empregatícias, que tem como efeito as Perdas Trabalhistas. A Análise Discriminante baseada em Redes Neurais a ser tratada neste estudo tem o intuito de modelar a probabilidade que um indivíduo possui de gerar Perdas Trabalhistas, dado um conjunto de características dele. Além disso, o modelo deve apresentar uma boa performance de classificação desse indivíduo nas duas classes disponíveis (gerar ou não perda trabalhista). Os resultados deste estudo motivam futuras pesquisas a utilizarem modelos quantitativos para controle, mensuração e gerenciamento de Risco Operacional.

Palavras-chave: Risco Operacional, Perdas Trabalhistas e Redes Neurais.

ABSTRACT

With the publishing of the New Basel Capital Accord, the financial institutions have included an important new component in their process of management and measurement of risks: Operational Risk. In the last few years, financial institutions have devoted tremendous efforts towards Modeling Operational Risk both in terms of Calculation of Capital and of quantitative models of control and management regarding this type of specific risk. In this study the technique of Artificial Neural Networks will be applied to an important category of Operational Risk event called Employment Practices, whose one of effects are Labor Losses. The Discriminating Analysis based on Artificial Neural Networks addressed in this study is designed to model the probability an individual has of generating Labor Losses given a set of individual features. In addition, the model should have a good classification performance of the individuals in to the two classes available (generating or not generating Labor Losses). The results of this study encourage future research in using quantitative models to control, measure and manage Operational Risk.

Key words: Operational Risk, Labor Losses and Artificial Neural Networks.

SUMÁRIO

1. INTRODUÇÃO *1*

1.1. Motivação **1**

1.2. Objetivos do Trabalho **3**

1.2.1. Objetivo Geral 3

1.2.2. Objetivos Específicos 3

1.2.3. Estrutura do Trabalho 4

2. RISCO OPERACIONAL *6*

2.1. O Novo Acordo de Capital da Basiléia **6**

2.2. Definição e Estrutura de Risco Operacional **11**

3. REDES NEURAIS ARTIFICIAIS *14*

3.1. Introdução **14**

3.1.1. Características Gerais 16

3.1.2. Topologias de Redes 19

3.1.3. Projeto de uma Rede Neural Artificial (RNA) 22

3.2. Processos de Aprendizado **24**

3.3. Modelo *Backpropagation* **28**

4. APLICAÇÃO E ANÁLISE DOS RESULTADOS *33*

4.1. Introdução **33**

4.2. Dados **34**

4.3. Tratamento dos Dados **35**

4.4. Modelagem dos Dados **36**

4.4.1. Análise Exploratória dos Dados 36

4.4.2. Seleção dos Modelos e Treinamento 37

4.5. Análise dos Resultados **39**

4.5.1. Análise dos Erros e Performance 39

4.5.2. Teste *Surrogate* 43

4.6. Refinamento do Modelo **46**

4.6.1. Votação em Comitê 46

4.6.2. Análise dos Erros e Performance 47

5. CONCLUSÕES *49*

REFERÊNCIAS BIBLIOGRÁFICAS *51*

APÊNDICES *53*

LISTA DE FIGURAS

Figura 1.1: Distribuição do Capital Econômico. 1
Figura 2.1: Estrutura de Risco Operacional 11
Figura 3.1: Modelo Matemático do Neurônio Artificial 17
Figura 3.2: Representação da Função Logística Sigmoidal 18
Figura 3.3: Estrutura de uma rede com múltiplas camadas escondidas 20
Figura 3.4: Estrutura de uma rede parcialmente conectada 20
Figura 3.5: Estrutura de uma rede completamente conectada 21
Figura 3.6: Estrutura de uma rede não recorrente ou feedforward 22
Figura 3.7: Etapas de um projeto de uma RNA 23
Figura 4.1: Análise Exploratória dos Dados 37

LISTA DE TABELAS

Tabela 2.1: Linhas de Negócio e Fatores βi 10

Tabela 2.2: Categorias de Causa, Evento e Efeito 13

Tabela 4.1: Modelos de Rede Implementados 39

Tabela 4.2: Ponto de Corte para Fronteira de Decisão 40

Tabela 4.3: Performance dos Modelos 41

Tabela 4.4: Performance dos Surrogates 44

Tabela 4.5: Estatística do Teste Surrogate 45

Tabela 4.6: Performance Votação em Comitê 47

1. INTRODUÇÃO

1.1. Motivação

Com o Novo Acordo de Capital da Basiléia, um novo componente passa a fazer parte do processo de gerenciamento e mensuração de riscos das instituições financeiras: o Risco Operacional.

Com a introdução explícita do Risco Operacional no Novo Acordo de Capital da Basiléia, muitos esforços têm sido alocados pelas instituições financeiras, tanto do Brasil quanto do exterior, no desenvolvimento de modelos matemáticos para cálculo de capital e gerenciamento do Risco Operacional.

Para se ter uma noção da relevância do Risco Operacional na atualidade, é interessante comparar o percentual de alocação de capital econômico para esse tipo de risco em relação aos demais. Como se pode perceber pelo gráfico a seguir, o Risco Operacional é responsável por 17% do total alocado de capital econômico, portanto existe muito incentivo para que pesquisas sejam realizadas com o intuito de desenvolver modelos quantitativos para mensuração e gestão desse risco.

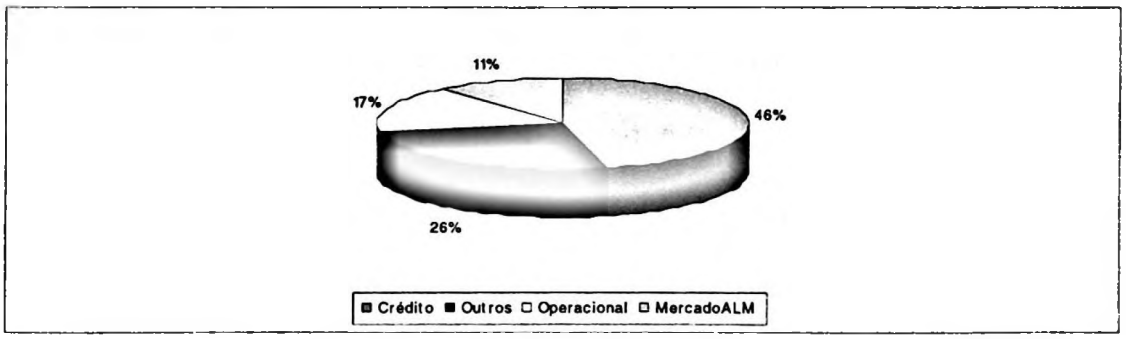

Figura 1.1: Distribuição do Capital Econômico.

[1] RMA Survey.

Especialmente no caso do mercado bancário brasileiro, um tipo de perda operacional que merece um destaque especial é a perda trabalhista, que se constitui de perdas derivadas de processos trabalhistas movidos por ex-funcionários.

De acordo com um levantamento realizado com dados internos de uma determinada instituição financeira, as perdas trabalhistas correspondem a 26% do total de perdas operacionais anuais e a aproximadamente 44% do total de perdas acumuladas.

Esse volume considerável é primordialmente causado em função de características próprias da legislação trabalhista brasileira, que rege que um bancário deve, no máximo, trabalhar 6 (seis) horas por dia, sendo o adicional considerado hora extra. Na prática, muitos bancos tratam essas perdas operacionais como um "custo de negócio", assumindo o risco operacional de não-pagamento das horas extras.

Embora as instituições assumam esse risco operacional, a provisão contábil dessas perdas trabalhistas é constituída somente quando é iniciada a ação legal. Devido a essa restrição contábil e ao volume significativo de perdas operacionais de caráter trabalhista, é latente a necessidade de gerenciar e controlar essa categoria de Risco Operacional por meio de modelos matemáticos que procurem estimar a probabilidade de um indivíduo, de acordo com certas características intrínsecas, gerar uma perda trabalhista à instituição.

A modelagem dessa probabilidade condicional será feita neste estudo específico por meio de uma técnica preditora e discriminante denominada Redes Neurais Artificiais, que tem sido aplicada em Finanças, principalmente em problemas de previsão de inadimplência de crédito, com resultados encorajadores.

Ao estimar essa probabilidade baseada em um modelo de Redes Neurais, provisões gerenciais poderão ser constituídas no momento do desligamento de um funcionário, e

não somente quando uma ação legal é iniciada, possibilitando um gerenciamento e uma mensuração mais sofisticada da exposição a essa categoria específica de Risco Operacional.

Atualmente existe um período de tempo, entre o desligamento do funcionário e o início da ação trabalhista, em que a exposição a risco não é mensurada, e, com este estudo, pretende-se preencher essa lacuna com a utilização de um modelo preditor e discriminante baseado em Redes Neurais Artificiais.

1.2. Objetivos do Trabalho

1.2.1. Objetivo Geral

Este estudo tem por objetivo desenvolver uma Rede Neural Artificial capaz de modelar a probabilidade de um indivíduo, de acordo com certas características intrínsecas dele, gerar perdas trabalhistas na instituição financeira a partir do momento de seu desligamento. Este modelo ainda deve possuir uma boa performance no que se refere à classificação dos indivíduos em duas classes: os que não geram perdas trabalhistas e os que geram.

1.2.2. Objetivos Específicos

Este trabalho possui também alguns objetivos específicos como:

- aprimorar o processo de provisão gerencial das perdas operacionais de caráter trabalhista por meio da modelagem da probabilidade condicional, utilizando Redes Neurais;
- sofisticar a tomada de decisão no momento de reestruturações que envolvam reduções de pessoal por intermédio da discriminação dos indivíduos em duas classes;
- analisar a aplicabilidade das Redes Neurais Artificiais para a modelagem de uma categoria específica de Risco Operacional;
- testar algumas arquiteturas de Redes Neurais adequadas ao problema proposto;
- selecionar e validar a topologia que melhor modela o problema em questão;
- refinar o modelo selecionado utilizando a técnica de votação em comitês.

1.2.3. Estrutura do Trabalho

A estrutura deste estudo será, a seguir, descrita.

No capítulo 2, os conceitos relacionados às definições de Risco Operacional serão desenvolvidos, além de contextualizar esse tipo de risco no âmbito do Novo Acordo de Capital da Basiléia. Como ponto de partida, na seção 2.1, será feita uma breve explicação sobre as mudanças propostas no Novo Acordo de Capital da Basiléia, com ênfase nos aspectos relacionados a Risco Operacional. A seção 2.2, por sua vez, será dedicada à explicação dos conceitos e da estrutura de Risco Operacional, englobando um tipo de perda operacional específica, que é a trabalhista, objeto deste estudo.

O capítulo 3 tratará dos principais aspectos teóricos relacionados às Redes Neurais Artificiais. A seção 3.1 abordará as características gerais das Redes Neurais Artificiais e a

topologias de Rede que será aplicada neste estudo. Na seção seguinte, 3.2, o foco será descrever o processo de aprendizado a ser utilizado na implementação prática . Na seção 3.3, aspectos relacionados à Rede Neural com algoritmo de treinamento *Backpropagation* serão abordados detalhadamente, uma vez que esse tipo de rede será usado na modelagem do problema proposto por este estudo.

Tendo sido contextualizados em caráter teórico tanto o Risco Operacional quanto as Redes Neurais Artificiais, o capítulo 4 tratará da aplicação e implementação de um modelo de Rede Neural, utilizando dados reais. Na seção 4.1, haverá uma breve introdução do problema, com a abordagem a ser adotada. Na seção seguinte, a 4.2, serão descritos os dados originais, e, na seção 4.3, a ênfase basear-se-á na explicação sobre a normalização das variáveis de entrada e saída da Rede, como a organização e a separação dos subconjuntos de dados. A implementação da Rede Neural no software MatLab será discutida na seção 4.4, enquanto que a seção 4.5 será dedicada à análise dos resultados. Finalmente, na seção 4.6, o modelo de Rede escolhido será refinado por meio da técnica de votação em comitês.

Finalizando, no capítulo 5, serão apresentadas as conclusões deste estudo. Alguns algoritmos usados no MatLab serão anexados no Apêndice deste trabalho.

2. RISCO OPERACIONAL

Este capítulo discorre sobre o Novo Acordo de Capital da Basiléia, contextualizando e enfocando uma de suas principais mudanças em relação ao primeiro Acordo de Capital de 1988: a inclusão explícita da necessidade de gerenciamento e mensuração do Risco Operacional.

Neste capítulo, também serão abordados os conceitos, definições e estruturas que envolvem o Risco Operacional, com isso, será possível contextualizar, no arcabouço teórico, um dos tipos de perdas operacionais, a perda trabalhista, cuja modelagem de sua probabilidade condicional é objeto deste estudo.

2.1. O Novo Acordo de Capital da Basiléia

O Comitê da Basiléia foi fundado em 1975, com representantes de instituições financeiras e bancos centrais do G10[1], com o objetivo maior de promover a estabilidade e a segurança do sistema financeiro em âmbito mundial.

Em 1988, foi referendado o primeiro Acordo de Capital da Basiléia, com foco no estabelecimento de um capital regulatório mínimo para o Risco de Crédito, sendo esse risco calculado com base no método do Ativo Ponderado pelo risco[2]. Em 1996, o primeiro Acordo sofreu uma emenda, com a inclusão do requerimento de capital para Risco de Mercado. Com

[1] Grupo composto por 10 países (Bélgica, Canadá, França, Alemanha, Itália, Japão, Luxemburgo, Holanda, Grã-Bretanha e Estados Unidos).
[2] Para cada grupo de ativos, foi estabelecido um fator de ponderação de risco.

a publicação e revisão do primeiro Acordo, os bancos centrais no mundo inteiro passaram a adotar, em alguns casos com pequenas variações, as diretrizes presentes nele.

Em 2001, o Comitê da Basiléia elaborou uma proposta, que vem sendo revisada nos últimos anos, sobre o Novo Acordo de Capital, o qual tem data prevista para sua implementação até o final de 2006.

O Novo Acordo de Capital está estruturado em três pilares básicos: Requerimentos de Capital Mínimo, Processo de Supervisão Bancária e Transparência ou Disciplina de Mercado.

As características mais relevantes do Pilar 1, Requerimentos de Capital Mínimo, são a inclusão explícita de alocação de capital para Risco Operacional; o refinamento da alocação de capital para Risco de Crédito, com a introdução de variáveis como *ratings*, garantias, taxas de recuperação, linhas de negócio etc., e a possibilidade de utilização de modelos internos e flexíveis para cálculo de capital regulatório.

A inclusão de Risco Operacional foi motivada pela ocorrência de alguns casos nos quais bancos foram à falência ou tiveram seus resultados fortemente impactados por eventos de Risco Operacional. O Comitê percebeu que existia um tipo de risco específico o qual deveria ter um tratamento especial no que se refere à alocação de capital e ao gerenciamento.

Os objetivos do Pilar 1 do Novo Acordo são proporcionar uma melhoria no processo de gerenciamento de riscos nas instituições financeiras, estimular o desenvolvimento de modelos de alocação de capital compatíveis com as peculiaridades de cada instituição e mais sensíveis a risco e diminuir a variância dos *spreads* de crédito.

Com a inclusão do Risco Operacional, a Exigência de Capital Mínimo proposta pelo Novo Acordo é regida pela seguinte equação:

$$\frac{CT}{(CRM + CRO) \times 12,5 + CRC} \geq 8\% \qquad (2.1)$$

Onde:

CT = Capital Regulatório Total definido conforme o Acordo de Capital de 1988.

CRM = Capital Regulatório Exigido para Risco de Mercado.

CRO = Capital Regulatório Exigido para Risco Operacional.

CRC = Capital Regulatório Exigido para Risco de Crédito.

O Pilar 2, Processo de Supervisão Bancária, tem como característica importante incentivar os reguladores a basearem a supervisão bancária numa abordagem mais voltada a processos e modelos. Os objetivos desse pilar estão calcados no incremento da qualidade do processo de supervisão bancária e no aumento da sinergia dessa atividade com o processo de gerenciamento de riscos praticado pelas instituições financeiras.

Já o Pilar 3, Transparência ou Disciplina de Mercado, enfoca e incentiva o *disclosure* de informações ao mercado sobre o nível de risco ao qual a instituição está exposta. Esse pilar tem como intuito alinhar os procedimentos contábeis com o reporte de riscos, ou seja, por meio das demonstrações contábeis, deve ser possível identificar de maneira mais clara e direta o nível de risco ao qual a instituição está exposta. Além disso, esse pilar incentiva uma maior homogeneização do reporte das informações de risco ao mercado.

Em resumo, em relação ao primeiro Acordo de Capital, o Novo Acordo da Basiléia traz as seguintes mudanças e inovações:

- **Risco de Crédito:**
  - antes calculado apenas com base no tipo de produto (por meio do método do ativo ponderado pelo risco);

- agora calculado também com base nas características individuais das contrapartes e da operação, como *ratings*, garantias, prazo, taxas de recuperação, produto e linhas de negócio;
- possibilidade de uso de modelo interno.

- **Risco Operacional:**
  - antes não era contemplado explicitamente;
  - agora foi incluído explicitamente no Novo Acordo;
  - Mensuração de Capital pode ser feita com modelos internos;
  - publicação de diretrizes sobre melhores práticas no gerenciamento de Risco Operacional.
  - Foco no aumento da qualidade do processo de supervisão bancária.
  - Foco no aumento da transparência ao mercado das informações sobre o nível de risco ao qual as instituições estão expostas.

Especificamente em relação à alocação de capital para Risco Operacional, o Novo Acordo de Capital prevê três tipos de abordagens.

A Abordagem Básica, na qual o capital exigido é função de um parâmetro α que multiplica a média da receita bruta dos últimos três anos.

$$K = GI.\alpha \tag{2.2}$$

Onde:

K = Capital Exigido para Risco Operacional por meio da abordagem básica.

GI = Média da Receita Bruta dos três últimos anos.

α = fator definido em 15% pelo Comitê da Basiléia.

A Abordagem Padronizada, que permite uma maior segmentação por linha de negócio, possui um parâmetro β associado a cada linha de negócio, sendo cada parâmetro multiplicado pela respectiva receita bruta da linha de negócio, perfazendo o capital exigido por linha de negócio. O capital exigido total será calculado por meio do somatório do capital exigido para cada uma das linhas de negócio definidas pelo Novo Acordo.

$$K = \sum_{i=1}^{8} (GIi.\beta i) \tag{2.3}$$

Onde:

K = Capital Exigido para Risco Operacional por meio da Abordagem Padronizada.

GIi = Média da Receita Bruta dos três últimos anos para cada linha de negócio.

βi = fatores definidos para cada linha de negócio[3], conforme tabela a seguir:

Linhas de Negócio	Fatores βi, em %
Corporate Finance	18
Negociação e Vendas	18
Varejo	12
Banco Comercial	15
Corretagem no Varejo	12
Pagamento e Recolhimento	18
Serviços de Consultoria	15
Administração de Ativos	12

Tabela 2.1: Linhas de Negócio e Fatores βi

[3] As linhas de negócio, com seus respectivos fatores β_i, foram referendadas com base em um estudo de maio de 2003 [5].

A Abordagem Avançada permite a seleção entre três tipos de modelos internos baseados em histórico de perdas, *holding period* e intervalos de confiança. Em todos os modelos internos, o Novo Acordo possibilita ajustes qualitativos no valor do capital calculado, que devem ser função de mudanças no ambiente operacional da instituição, do ambiente de controles internos e expectativas futuras. O Novo Acordo estipula que os bancos possuam uma base histórica de perdas de no mínimo três anos, utilizem o *holding period* de um ano, com um intervalo de confiança de 99,9% para o cálculo do *Value at Risk*. Esses modelos são:

- Distribuição de Perdas;
- Método de Mensuração Interna;
- Método de *Scorecard*.

2.2. Definição e Estrutura de Risco Operacional

Em 2001, o Comitê de Supervisão Bancária da Basiléia definiu o Risco Operacional como sendo "O risco de perdas resultante de processos internos inadequados ou falhos, pessoas e sistemas, ou de eventos externos".

A fim de conceituar melhor essa definição, é interessante analisar cada componente da estrutura de Risco Operacional proposta pelo Comitê da Basiléia:

Figura 2.1: Estrutura de Risco Operacional

As causas, na estrutura de Risco Operacional, correspondem aos fatores de risco ou possíveis vulnerabilidades que existem no ambiente operacional e podem levar à materialização de riscos operacionais. Esse componente é extremamente importante no gerenciamento de riscos operacionais, uma vez que os controles a serem criados ou ajustados, a fim de minimizar a probabilidade de ocorrência do risco operacional, devem ser desenhados e concebidos para atuar nesse componente de causa. Um ponto importante a ser destacado é que podem estar associados a um mesmo risco mais de um tipo de causa.

Um outro componente da estrutura é o evento, que nada mais é do que a materialização do risco, ou seja, o risco concretizado (probabilidade igual a 1). A cada evento de risco operacional, estão associados um ou mais efeitos, que podem ser perdas financeiras ou outros impactos como o reputacional ou o estratégico, os quais podem gerar perdas de receita.

Para tornar essa estrutura menos subjetiva, facilitar o gerenciamento e dar um caráter mais homogêneo ao reporte do Risco Operacional pelos bancos, o Comitê da Basiléia estipulou uma série de categorias de classificação para os componentes de causa, evento e efeito.

A categorização de causas facilita o gerenciamento proativo do risco, enquanto que a categorização dos eventos é extremamente útil, pois cada tipo de evento possui uma distribuição de probabilidade específica. As categorizações dos efeitos, por sua vez, facilitam o processo decisório com relação à aquisição de proteção por meio de seguros para certos tipos de efeitos ou mesmo para provisionamento de outros tipos de perdas. A tabela a seguir ilustra essas categorias[4]:

[4] Mais informações e detalhes sobre as categorias estão no documento sobre o Novo Acordo de Capital da Basiléia [2].

Causas	Eventos	Efeitos
Processos Internos	Fraude Interna	Prejuízo
Sistemas	Fraude Externa	Restituição
Pessoas	Práticas Empregatícias/Segurança no Ambiente	Perda de Recursos
Fatores Externos	Execução, Entrega e Gestão de Processo	Perdas ou Danos a Ativos
	Danos a Ativos Físicos	Obrigações Legais
	Clientes, Produtos e Práticas de Negócio	Ações Regulatórias
	Interrupção de Negócios/Falhas de Sistemas	Estratégico/Negócio
		Reputacional

Tabela 2.2: Categorias de Causa, Evento e Efeito

As perdas trabalhistas, cuja modelagem de sua probabilidade condicional por meio de Redes Neurais Artificiais é objeto deste estudo, enquadram-se na categoria de efeito definida como Obrigações Legais. Por sua vez, a categoria de evento de risco que gera como conseqüência esse tipo de perda é definida como Práticas Empregatícias/Segurança no Ambiente.

3. REDES NEURAIS ARTIFICIAIS

Este capítulo está dedicado à descrição de alguns aspectos teóricos relevantes relacionados às Redes Neurais Artificiais, que darão subsíduo ao entedimento do capítulo 4, dedicado à implementação prática

No primeiro momento, as características gerais das Redes Neurais Artificiais (que, agora, passarão a ser denominadas RNAs) serão contextualizadas e a topologia a ser usada na implementação prática (rede *feedforward*), tal como as funções de ativação e propagação usadas na modelagem serão abordados.

Aspectos relacionados à organização de um projeto de RNAs, do ponto de vista teórico, serão descritos logo em seguida, a fim de embasar a aplicação prática

Em seguida, serão explicados detalhadamente os principais aspectos relacionados ao processo de aprendizado supervisionado de RNAs por correção do erro.

Uma seção específica deste capítulo será dedicada ao desenvolvimento dos aspectos teóricos envolvidos no Modelo *Backpropagation,* que será implementado para a modelagem do problema proposto neste estudo.

3.1. Introdução

De acordo com definição de Haykin [6], uma Rede Neural Artificial pode ser entendida como:

"Um processador maciçamente e paralelamente distribuído, constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para uso. Ela assemelha-se ao cérebro em dois

aspectos: (1) o conhecimento é adquirido pela rede a partir de seu ambiente através de um processo de aprendizagem; (2) forças de conexão entre neurônios (os pesos sinápticos) são utilizadas para armazenar o conhecimento adquirido".

As RNAs são técnicas computacionais que apresentam um modelo matemático baseado na estrutura neural de organismos inteligentes e que adquirem conhecimento pela experiência.

A fim de tornar a definição de uma RNA mais concreta e palpável, cabe aqui descrever sucintamente um pouco o funcionamento do sistema nervoso, que é a inspiração biológica das RNAs.

O sistema nervoso é formado por um conjunto complexo de células nervosas denominadas neurônios. Os neurônios, por sua vez, são formados pelos:

- dendritos, que são um conjunto de terminais de entrada com a função de receber os estímulos transmitidos pelos outros neurônios;
- . corpo central ou soma, que tem a função de coletar e combinar as informações vindas de outros neurônios;
- axônios, que são longos conjuntos de saída responsáveis pela transmissão dos estímulos para outras células.

A comunicação entre os neurônios ocorre por meio de sinapses. A sinapse pode ser entendida como a região em que dois neurônios entram em contato, e é pela sinapse que os impulsos nervosos são transmitidos entre os neurônios.

Esse processo ocorre da seguinte maneira: os impulsos recebidos por um neurônio X são processados em um determinado instante, e atingindo um dado limiar de ação, esse mesmo neurônio dispara, produzindo uma substância neurotransmissora que sai da soma e se dirige ao axônio, que, por sua vez, pode estar ligado a um dendrito de um neurônio Y. Dependendo do tipo de excitação, um impulso é ou não gerado para o axônio desse neurônio Y.

3.1.1. Características Gerais

A RNA pode ser considerada, em termos genéricos, como sendo uma estrutura de processamento de dados distribuído e em paralelo, formada por vários elementos de processamento unidos por conexões frequentemente unidirecionais (mas não necessariamente). O elemento de processamento, ou neurônio artificial, pode apresentar N entradas, entretanto há apenas uma saída que poderá se conectar a outros neurônios.

O processamento que ocorre em cada neurônio é local e depende dos valores dos sinais de entrada em suas conexões e dos valores guardados em sua memória local.

Em 1943, Walter Pitts e Wax Ten McCulloch [7] desenvolveram o modelo matemático do neurônio artificial, que pode ser visualizado esquematicamente a seguir:

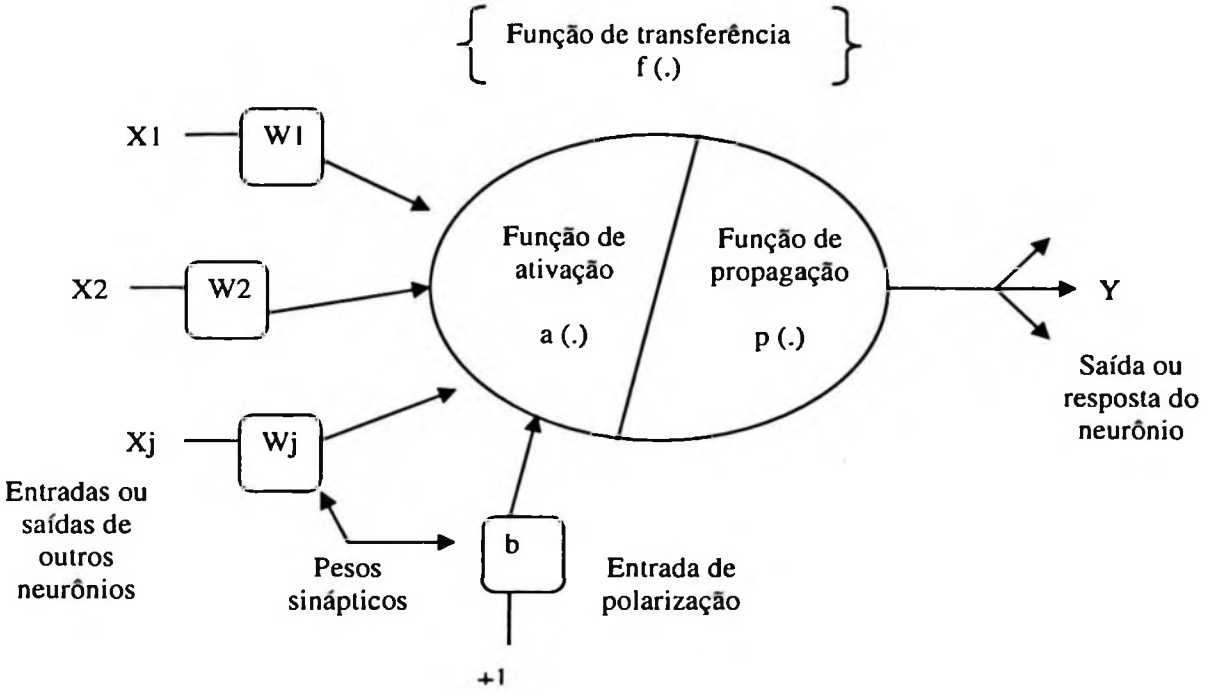

Figura 3.1: Modelo Matemático do Neurônio Artificial

Neste modelo matemático proposto, as conexões e pesos sinápticos representam os dendritos e sinapses respectivamente, a função de transferência atua como o corpo central do neurônio biológico, enquanto a saída representa o axônio.

As entradas x_j atuam como os estímulos que chegam ao neurônio em um determinado momento e são ponderadas por pesos w_j, antes do processamento pelo corpo central.

A função de ativação, denominada neste estudo como a (.), é responsável pelo processamento das entradas ponderadas e mede o estado de ativação. Algumas das funções de ativação mais utilizadas são a esférica, linear, polinomial e mahalanobis[5] sendo que neste estudo utilizou-se a função linear, conforme equação a seguir:

[5] Maiores detalhes sobre funções de ativação ver [13]

- Linear

$$a(x,w)=\sum_{j=1}^{n} x_j w_j \tag{3.1}$$

A função de propagação p(.) tem como intuito propagar o estado de ativação como estímulo para os demais neurônios, sendo responsável por definir o valor de estímulo de saída do neurônio.

A função de propagação deve respeitar algumas propriedades importantes, como ser monotônica, limitada e continuamente diferenciável. Os tipos mais usados de função de propagação são a degrau, degrau simétrico, logística sigmoidal e tangente hiperbólica[6], sendo que neste estudo utilizou-se a função logística sigmoidal na camada de saída. Cabe ressaltar que as funções degrau e degrau simétrico não são diferenciáveis em $x=0$.

- Logística Sigmoidal

$$y=\frac{1}{1+e^{-(n+b)}} \tag{3.2}$$

A função logística sigmoidal aceita entradas com valores no intervalo entre $(-\infty,\infty)$, ficando os valores de saída da função restritos ao intervalo (0, 1).

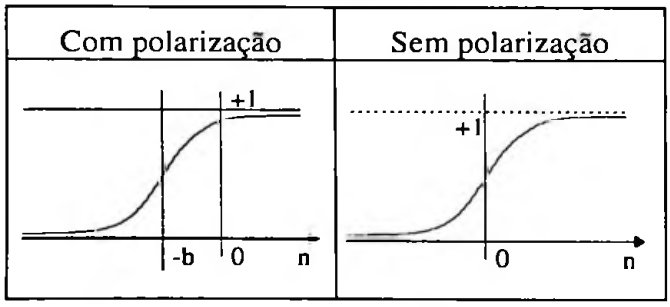

Figura 3.2: Representação da Função Logística Sigmoidal

[6] Maiores detalhes sobre funções de propagação ver [13]

A função logistica sigmoidal possui em sua derivada uma propriedade interessante: o fato de poder ser expressa em termos das própria função. Essa propriedade facilita muito os cálculos computacionais.

Segue a ilustração desta propriedade, num caso com polarização nula (b = 0):

$$f(n) = \frac{1}{1+e^{-n}} \qquad (3.3$$

$$f'(u) = \frac{df(u)}{du} = f(u)[1 - f(u)] \qquad (3.4)$$

De maneira resumida, a função de ativação processa os estímulos de entrada ponderados pelos pesos sinápticos, calculando o chamado estado de ativação. Esse estado de ativação é transmitido para a função de propagação, que, por sua vez, calcula o valor de saída do neurônio. A composição das funções de ativação e de propagação geram as chamadas funções de transferência ou de mapeamento matemático do neurônio.

$$f(.) = a(.) \,.\, p(.) \qquad (3.5)$$

Um ponto importante a se destacar é que a entrada de polarização, conhecida também como *bias,* desloca a função de transferência a partir de sua origem e tem o intuito de aumentar a velocidade de convergência do algoritmo de aprendizado, que será tratado em seções posteriores deste estudo.

3.1.2. Topologias de Redes

As RNAs podem ter sua topologia ou arquitetura variando quanto ao número de camadas e neurônios por camada, ao tipo de conexão e ao tipo de conectividade.

Em relação ao número de camadas, uma RNA pode ter apenas uma camada, sendo denominada de *Perceptron*, ou ter múltiplas camadas. Neste estudo, será implementada uma rede com uma camada escondida . A figura a seguir ilustra uma rede com múltiplas camadas escondidas.

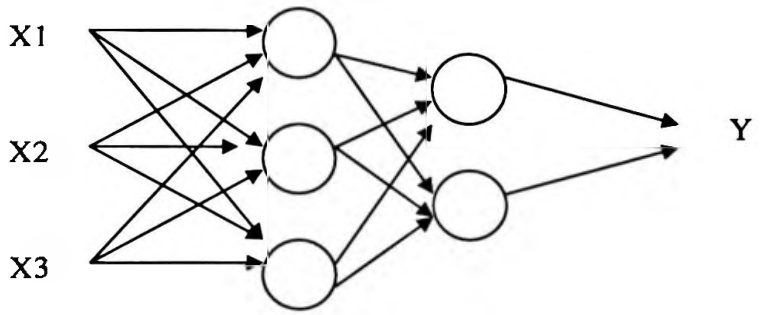

Figura 3.3: Estrutura de uma rede com múltiplas camadas escondidas

Em relação ao tipo de conectividade uma rede pode ser completamente conectada, ou seja, todas as saídas dos neurônios de uma camada são conectadas com todos os neurônios da camada seguinte, ou parcialmente conectada.

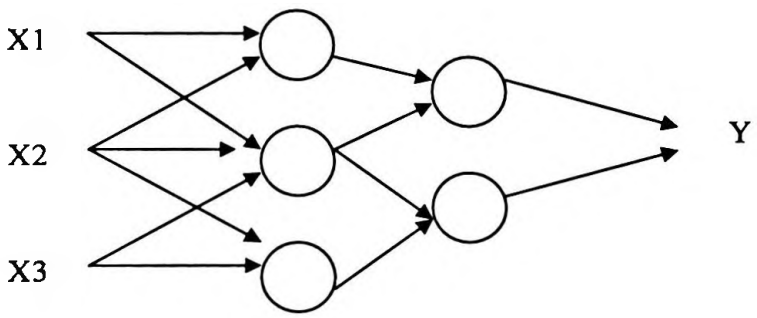

Figura 3.4: Estrutura de uma rede parcialmente conectada

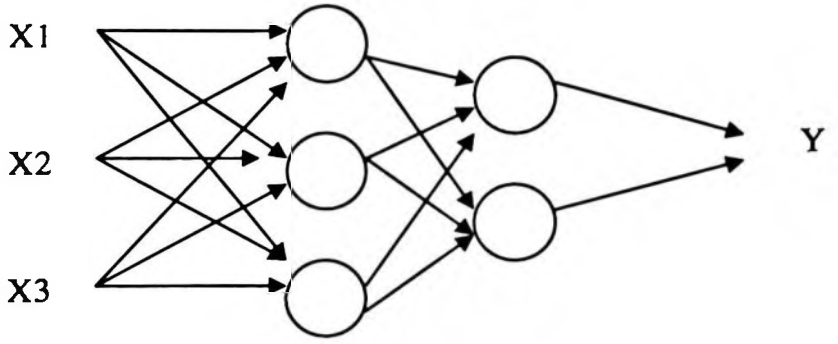

Figura 3.5: Estrutura de uma rede completamente conectada

O aumento do número de camadas gera um incremento no tempo de processamento e complexidade da rede, possibilitando a modelagem de problemas não linearmente separáveis. Já o aumento do número de neurônios por camada ocasiona um incremento do grau de liberdade da função de transferência, e quanto maior o número de variáveis livres menor será a capacidade de generalização da rede após seu treinamento. Muitos neurônios podem gerar também memorização dos dados de teste ou *overfitting*.

Em relação ao tipo de conexão, existem dois modelos básicos de RNAs: o não recorrente ou *feedforward* e o recorrente[7].

O modelo *feedforward*, que será utilizado neste estudo, é também conhecido como "sem memória", uma vez que não possui realimentação de suas saídas para suas entradas. Além disso, nesse modelo, não existem conexões unindo os neurônios da mesma camada nem de um neurônio para outro da camada anterior, as entradas são aplicadas a todos os neurônios da camada de entrada, e todas as saídas dos neurônios de uma camada são conectadas com todos os neurônios da camada seguinte (estrutura totalmente conectada)

[7] Maiores detalhes sobre redes recorrentes ver [13]

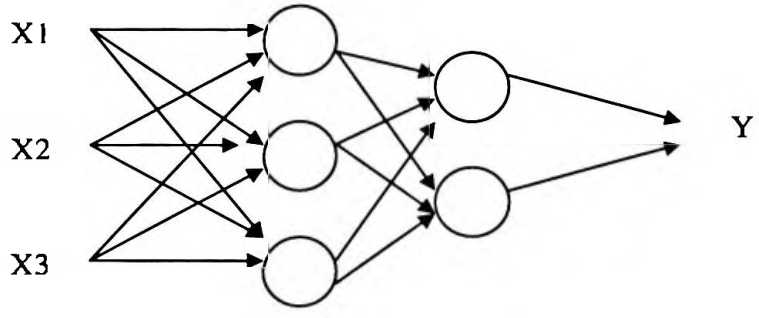

Figura 3.6: Estrutura de uma rede não recorrente ou *feedforward*

Uma outra forma de categorizar a topologia de RNAs é por meio do processo de aprendizado ao qual é submetida. Esse tópico será abordado numa seção posterior deste capítulo.

3.1.3. Projeto de uma Rede Neural Artificial (RNA)

Um modelo ou projeto baseado em RNAs deve seguir algumas etapas que são geralmente usuais. Caudill [11] propõe que um projeto utilizando RNAs deva possuir as seguintes etapas:

- Definição do Problema.
- Coleta de Dados (Treinamento e Teste).
- Pré-Processamento dos Dados.
- Projeto da Topologia da Rede.
- Treinamento da Rede.
- Teste e Validação da RNA.
- Pós-Processamento dos Dados.

Figura 3.7: Etapas de um projeto de uma RNA

Dessas etapas, a maior dificuldade recai sobre a definição do projeto de topologia da RNA, ou seja, estipular o número necessário de camadas ocultas e o número de neurônios em cada uma delas. Como não existe um método específico para isso, muitas vezes a topologia é definida de maneira heurística. Existe disponível literatura propondo o uso de algorítmos genéticos para seleção de topologia de RNA's [24].

Embora essa tarefa de solucionar a melhor arquitetura de RNA seja relativamente árdua, o Teorema de Kolmogorov [12] demonstra que qualquer função contínua de muitas variáveis (para um domínio de entrada fechado e limitado) pode ser representada como a superposição de um pequeno número de funções de apenas uma variável. Em termos de teoria de RNAs, esse teorema diz que qualquer função contínua a qual mapeia X para Y por meio de variáveis de entrada x_i para uma variável de saída y pode ser representada por uma RNA de três camadas, sendo duas escondidas. A primeira camada escondida deve possuir d (2d +1) neurônios e a segunda (2d+1) neurônios, onde d é o número de variáveis de entrada.

3.2. Processos de Aprendizado

"Aprendizado é o processo pelo qual os parâmetros livres da rede neural são ajustados através de um processo contínuo de estímulo do ambiente no qual a rede está inserida." Simon Haykin [13], adaptado de Mendel e McClaren.

Uma das características mais relevantes e importantes das RNAs é a sua habilidade de aprender com o ambiente, melhorando, dessa forma, seu desempenho.

Esse aprendizado ou treinamento das RNAs é realizado por meio de um processo iterativo de ajustes dos pesos sinápticos, a fim de minimizar a função erro definida entre a saída produzida pela rede e a saída desejada.

Esse treinamento consiste na aplicação seqüencial de um vetor de entradas e no ajuste dos pesos por meio de um algoritmo de aprendizado predefinido, ou seja, a RNA se modifica em função da necessidade de aprender com as informações que lhe são apresentadas.

A Rede deve ser treinada utilizando um conjunto de dados de treinamento e deve ser capaz de generalizar quando lhe é apresentado um outro conjunto de teste, com dados deferentes do conjunto de treinamento.

Um ponto importante a se destacar é que se deve tomar cuidado para evitar o *overfitting* ou "supertreinamento/memorização" dos dados durante o aprendizado com o conjunto de treinamento. Se ocorrer o *overfitting*, a rede não será capaz de generalizar, ou seja, reconhecer padrões e classificá-los de forma correta quando for exposta a dados que não estão contidos no conjunto de treinamento.

Existem três abordagens de aprendizado possíveis: aprendizado não supervisionado, supervisionado e híbrido[8].

No aprendizado supervisionado, ou com professor, que será implementado neste estudo, é usado um agente externo que mostra à rede a saída desejada para um determinado vetor de entrada. O ajuste dos pesos ocorre por meio da comparação entre o vetor de saída gerado pela rede e o vetor de saída desejado e conhecido. O processo de treinamento ou aprendizado inicia-se quando o vetor de entrada é apresentado à RNA. Em seguida, calcula-se o vetor de saída da rede, e ele é comparado com o vetor de saída desejado, se encontrado, portanto, o erro. Esse erro é realimentado por meio da RNA, os pesos sinápticos são recalculados por um algoritmo predeterminado a fim de minimizar a função erro escolhida previamente. O treinamento deve ser interrompido quando os erros estabilizarem-se em níveis baixos, sempre tomando cuidado para evitar o fenômeno de memorização dos dados.

Num processo de aprendizado, os pesos sinápticos de uma RNA são ajustados por meio de algoritmos de treinamento, que podem ser definidos como um conjunto de regras bem definidas para a resolução de um problema de aprendizagem [13].

A grande diferenciação que hoje existe entre os diversos algoritmos de aprendizado refere-se à regra de aprendizado adotada por cada um deles, ou seja, a forma como os pesos sinápticos são ajustados.

Foram descritas por Haykin [13] cinco regras de aprendizado: por correção do erro, baseada em memória, Hebbiana, competitiva e de Boltzmann[9]

---

[8] Maiores detalhes sobre o processo de aprendizado híbrido e o não supervisionado ver [13]
[9] Maiores detalhes sobre regras de aprendizado ver [13]

- Aprendizado por correção do erro

Essa regra de aprendizado, que será aplicada neste estudo, baseia-se na abordagem de aprendizado supervisionado, em que o sinal de saída calculado pela rede para um neurônio k é comparado com o sinal de saída desejado, produzindo um erro. Esse erro é usado para ajustar os pesos a fim de minimizar a distância entre o sinal de saída desejado e o produzido pela RNA. A equação desse erro é expressa por:

$$e_k(n) = d_k(n) - y_k(n) \tag{3.6}$$

Onde: $e_k(n)$ é o sinal do erro do neurônio k.

$d_k(n)$ é o sinal de saída desejado.

$y_k(n)$ é o sinal de saída produzido pelo neurônio k.

A equação que rege a variação dos pesos sinápticos nessa regra de aprendizado é:

$$\Delta w_{kj} = \eta e_k(n) x_j(n) \tag{3.7}$$

Onde: $e_k(n)$ é o sinal do erro do neurônio k.

x_j é o sinal pré-sináptico.

η é uma constante positiva que define a taxa de aprendizado.

O processo de treinamento de uma RNA pode ter duas dinâmicas de atualização dos pesos sinápticos:

- *Batch:* em que os pesos são ajustados somente ao final de cada época[10]. Uma característica positiva dessa dinâmica de aprendizado é que o treinamento é pouco

[10] Época é a nomenclatura dada ao processamento de todo o conjunto de observações.

influenciado pela ordem de apresentação dos padrões da RNA, entretanto a velocidade de convergência é um pouco mais lenta.

- Incremental ou *on-line:* os pesos são ajustados ao final do processamento de cada observação. Um ponto relevante é que a ordem de apresentação dos padrões influi na velocidade de convergência do algoritmo de aprendizado.

Na implementação prática deste estudo, a ser descrita no capítulo seguinte, foi usada a dinâmica *batch.*

Em relação à velocidade de convergência do processo de aprendizado, alguns parâmetros são geralmente utilizados com este intuito:

- Momento: esse parâmetro tem o intuito de acelerar o treinamento evitando oscilações. A inclusão do momento faz com que a RNA ignore as variações de alta freqüência na superfície da função erro, reduzindo a probabilidade do processo de convergência estacionar num ponto de mínimo local da função. O parâmetro do momento faz com que as variações nos pesos sinápticos sejam iguais à soma de uma fração da última variação ocorrida e uma nova alteração determinada pela taxa de aprendizado, conforme a seguinte equação:

$$w_{kj}(n) = \alpha w_{kj}(n-1) + \eta \delta_k(n) x_j(n) \qquad (3.8)$$

Onde: n é o n-ésimo exemplo apresentado à rede.

w_{kj} é o peso sináptico do neurônio k.

α é uma constante positiva que define o parâmetro de momento.

x_j é o sinal pré-sináptico.

η é uma constante positiva que define a taxa de aprendizado.

δ_k é o gradiente da função erro.

Taxa de Aprendizado: esse parâmetro influencia a magnitude das variações nos pesos sinápticos. Utilização de taxas de aprendizado muito pequenas podem levar à lentidão no processo de convergência e ao aumento de probabilidade de parada em pontos de mínimos locais da função erro. Por outro lado, altas taxas de aprendizado podem gerar oscilações na RNA, dificultando o processo de treinamento. Normalmente, o valor da taxa de aprendizado é determinado em 0,01, porém essa taxa pode ser reduzida caso haja um aumento no erro médio quadrático de previsão da rede ou vice-versa.

- Entrada de polarização ou *bias*: esse parâmetro desloca a função de transferência a partir de sua origem, aumentando a velocidade de convergência do algoritmo de aprendizado.

3.3. Modelo *Backpropagation*

O modelo ou algoritmo *backpropagation* foi desenvolvido em 1986 por Rumelhart, Hinton e Williams por meio da generalização da regra de aprendizado *Widrow-Hoff*, criada entre 1960 e 1962 por Bernard Widrow e Marcian Hoff para as redes do tipo *feedforward perceptron.*

A regra *Widrow-Hoff* é mais comumente conhecida como *Regra Delta* ou LMS (minimização do erro quadrático médio). Nessa regra, os pesos sinápticos são ajustados a fim de minimizar o erro, ou seja, a regra busca encontrar um conjunto de pesos e polarizações que atendam a esse objetivo de minimização.

A alteração dos pesos wji, segundo a Regra Delta, é calculada de acordo com a seguinte equação:

$$\Delta w_{ji}(n) = -\eta \frac{\partial E(n)}{\partial w_{ji}(n)} \tag{3.9}$$

Onde: n é o n-ésimo exemplo apresentado à rede.

w_{ji} é o peso sináptico do neurônio j.

η é uma constante positiva que define a taxa de aprendizado.

$d_j(n)$ é o sinal de saída desejado.

$y_j(n)$ é o sinal de saída produzida pelo neurônio j.

C é o conjunto que inclui todos os neurônios da camada de saída da rede.

$E(n) = \frac{1}{2}\sum_{j \in C}(d_j(n) - y_j(n))^2$ é a função erro instantâneo.

$\frac{\partial E(n)}{\partial w_{ji}(n)}$ é a derivada parcial da função erro em relação ao peso.

No modelo *backpropagation*, o ajuste dos pesos e polarizações é feito por intermédio da retropropagação do erro calculado através da saída. O processo de minimização da função erro é realizado continuamente a cada iteração, sendo o ajuste dos pesos e das polarizações feito no sentido oposto ao do gradiente da função no ponto corrente, ou seja, proporcionalmente ao negativo da derivada do erro quadrático médio em relação aos pesos atuais.

O algoritmo *backpropagation* é, em resumo, um algoritmo de treinamento supervisionado e determinístico, que implementa o método do gradiente descendente na soma dos quadrados dos erros.

O *backpropagation* é usualmente utilizado em redes com uma ou mais camadas escondidas de neurônios não-lineares, usando a função de propagação sigmoidal ou tangente hiperbólica e uma camada de saída de neurônios lineares.

No processo de treinamento *on-line* utilizando o algoritmo *backpropagation,* são realizados os passos a seguir [13]:

Passo 1: inicializar os pesos e polarizações com números pequenos e aleatórios e estipular os demais parâmetros de treinamento com a topologia de rede escolhida.

Passo 2: apresentar à rede um padrão de entrada [x(n)] e a saída desejada [d(n)], com n = 1, 2,..., N. Para cada exemplo de treinamento [x(n), d(n)], devem ser realizados os passos 3 e 4.

Passo 3: calcular a saída. O vetor de entradas x(n) é aplicado aos neurônios da camada de entrada, e o vetor de saída desejado é apresentado ao neurônio da camada de saída. Os estados de ativação e sinais de saída são calculados pela rede camada por camada. O nível de atividade interna da rede para o neurônio j na camada l pode ser definido por:

$$v_j^{(l)}(n) = \sum_{i=0}^{p} w_{ji}^{(l)}(n)\, y_i^{(l-1)}(n) \tag{3.10}$$

Onde: $y_i^{(l-1)}(n)$ é o sinal de saída do neurônio i na camada l-1 na iteração n.

$w_{ji}^{(l)}(n)$ é o peso sináptico do neurônio j na camada l, que é alimentado pelo neurônio i.

P é o número de entradas aplicadas ao neurônio j, excluindo a entrada de polarização.

Se i=0, $y_0^{(l-1)}(n) = -1$ e $w_{j0}^{(l)}(n) = \theta_j^l(n)$, que é o peso da polarização ou *threshold* aplicado no neurônio j da camada l. Assumindo utilização da função logística sigmoidal, o sinal de saída do neurônio j na camada l é definido por:

$$y_j^{(l)}(n) = \frac{1}{1+e^{(-v_j^{(l)}(n))}} \tag{3.11}$$

Se o neurônio j estiver nas primeiras camadas escondidas (l=1), então $y_j^0(n) = x_j(n)$, sendo $x_j(n)$ o j-ésimo elemento do vetor x(n). Se o neurônio j estiver na camada de saída (l=L), então $y_j^{(L)}(n) = y_j(n)$.

Calculado o valor da saída da rede, calcular o erro:

$$e_j(n) = d_j(n) - y_j(n) \tag{3.12}$$

Passo 4: calcular os gradientes locais do erro ($\delta's$) recursivamente, começando na camada de saída e trabalhando para trás, no sentido da primeira camada, conforme as seguintes equações:

$\delta_j^{(L)}(n) = e_j^{(L)}(n) y_j(n)[1 - y_j(n)]$ Se o neurônio j estiver na camada de saída L. (3.13)

$\delta_j^{(l)}(n) = y_j^{(l)}(n)[1 - y_j(n)]\sum_k \delta_k^{(l+1)}(n) w_{kj}^{(l+1)}(n)$ Se o neurônio j estiver na camada escondida l. (3.14)

O ajuste dos pesos sinápticos na camada l, de acordo com a regra delta generalizada, será dado pela seguinte equação:

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha[w_{ji}^{(l)}(n) - w_{ji}^{(l)}(n-1)] + \eta\delta_j^{(l)}(n) y_i^{(l-1)}(n) \tag{3.15}$$

Onde: α é a constante de momento.

η é a taxa de aprendizado.

Passo 5: apresentar à rede novas épocas de exemplos de treinamento até os parâmetros livres de treinamento se estabilizarem e a função do erro médio quadrático (3.41) atingir um ponto de mínimo.

$$E(n) = \frac{1}{2}\sum_{j \in C}(d_j(n) - y_j(n))^{\frac{1}{2}} \tag{3.16}$$

$$E\,(\text{médio}) = \frac{1}{N}\sum_{n=1}^{N} E(n) \tag{3.17}$$

4. APLICAÇÃO E ANÁLISE DOS RESULTADOS

4.1. Introdução

A abordagem prática utilizando RNAs para modelagem da probabilidade condicional de um indivíduo gerar perda trabalhista e a discriminação desse indivíduo em duas classes foram baseadas em algumas etapas que serão descritas a seguir.

Os modelos de RNAs que serão implementados neste estudo estimam as probabilidades condicionais (ou *a posteriori*) $P(C_k \mid x)$ expressas por meio do Teorema de Bayes da seguinte forma:

$$P(C_k \mid x) = \frac{p(x \mid C_k)P(C_k)}{p(x)} \tag{4.1}$$

Onde: C_k é a k-ésima classe, existindo, neste estudo, apenas duas classes (indivíduos que geraram e os que não geraram perdas).

x é o vetor de entradas

$p(x \mid C_k)$ é a densidade condicional à classe

$p(x)$ é função de densidade incondicional, que atua como um fator normalizador, garantindo que $\sum_{k=1}^{c} P(C_k \mid x) = 1$

$P(C_k)$ é a probabilidade *a priori*, que pode ser calculada com base no conjunto de dados.

$$p(x) = \sum_{k=1}^{c} p(x \mid C_k)P(C_k)$$

88289

A RNA busca modelar a densidade condicional à classe $p(x|C_k)$ por meio de funções parametrizáveis. A seleção destas funções no modelo levará a escolha de uma determinada distribuição[11].

Os modelos de RNAs que foram implementados no MatLab, utilizando a biblioteca NetLab, baseiam-se em redes não recorrentes com apenas uma camada escondida, sujeitas a treinamento supervisionado *off-line ou batch* com aprendizado por correção do erro utilizando o modelo *backpropagation* por meio do método/algoritmo de otimização Gradiente Conjugado Escalado[12].

A função de propagação utilizada na camada oculta é a tangente hiperbólica e, na camada de saída, a logística. A função de ativação usada em todas as camadas é a linear.

4.2. Dados

Os dados coletados para modelagem compreendem o período de 1998 a 2004, contemplando oito variáveis de entrada relacionadas a todos os indivíduos que se desligaram da instituição nesse período, além de uma variável de saída indicando se o indivíduo gerou ou não perda trabalhista.

O total de dados é constituído por uma matriz com 15.709 linhas, representando os indivíduos, e 9 colunas totalizando o somatório das variáveis de entrada e de saída.

Segundo o conjunto total de dados, a probabilidade *a priori* de um indivíduo gerar perda trabalhista é de 18,52% e de não gerar perda é de 81,48%.

As variáveis de entrada utilizadas na modelagem são:

[11] Maiores detalhes ver [12]

- **Tempo**: período em que o indivíduo trabalhou na instituição.
- **Idade**: idade do indivíduo quando se desligou da instituição.
- **Dependentes**: número de dependentes (filhos, por exemplo) do indivíduo quando se desligou da instituição.
- **Sexo**: sexo do indivíduo.
- **Desligamento**: motivo do desligamento (demissão por justa causa, sem justa causa, pedido de demissão, aposentadoria etc.).
- **Nível:** nível de instrução do indivíduo na data do desligamento (Ensino Médio, Superior completo etc.).
- **Estado**: estado civil do indivíduo na data do desligamento (solteiro, casado etc.).
- **Faixa salarial**: faixa salarial do indivíduo na data do desligamento.

4.3. Tratamento dos Dados

Com o intuito de diminuir a distância absoluta entre os valores assumidos pelas variáveis de entrada e por muitas delas serem qualitativas, os dados sofreram um pré-processamento.

As variáveis qualitativas de entrada foram normalizadas para o intervalo [-1,1], de acordo com o seu respectivo número de classes, e as quantitativas foram normalizadas para o intervalo [0,1], de acordo com a sua amplitude.

---

[12] Detalhes do algorítmo de otimização Gradiente Conjugado Escalado no Apêndice deste estudo

A variável de saída assume valor 1, se o indivíduo gerou perda trabalhista, ou 0, se não gerou.

Com o objetivo de realizar o processo de modelagem, o conjunto total de dados foi dividido em três subconjuntos:

- Treinamento: correspondente a 60% do total a ser usado no treinamento da rede.
- Validação: correspondente a 20% do total a ser usado para acompanhar o desempenho da rede a partir de uma amostra independente.
- Teste: correspondente a 20% do total a ser usado para testar a rede com um subconjunto de dados independente da criação do modelo.

Esses três subconjuntos foram criados a partir de um sorteio aleatório do conjunto total.

4.4. Modelagem dos Dados

4.4.1. Análise Exploratória dos Dados

Antes da seleção dos modelos de rede, realizou-se uma análise exploratória dos dados, plotando em gráficos com duas dimensões os valores de algumas variáveis, a fim de se verificar se o problema em questão pode ser considerado linearmente separável.

O resultado, que pode ser facilmente visualizado nos gráficos a seguir, indica que o problema não pode ser considerado linearmente separável, descartando-se, portanto, o

uso do *Perceptron* e partindo-se para uma modelagem de redes com uma camada escondida.

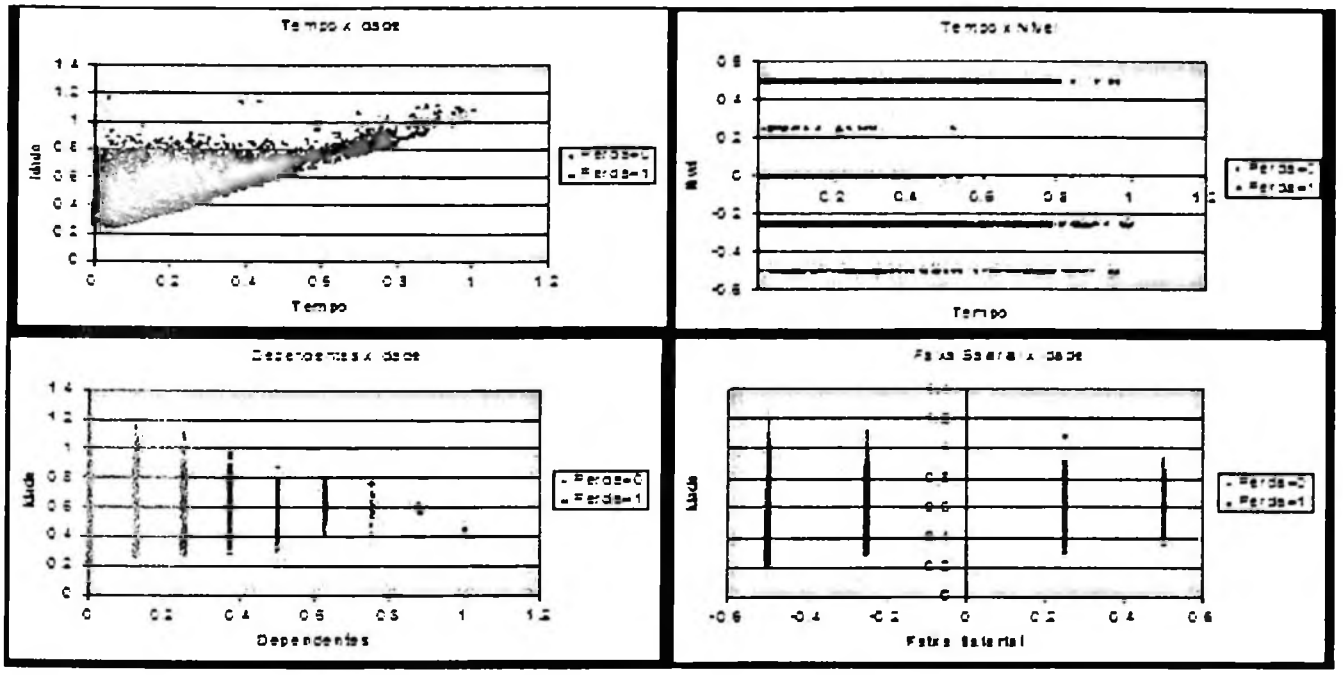

Figura 4.1: Análise Exploratória dos Dados

4.4.2. Seleção dos Modelos e Treinamento

O modelo de rede selecionado para a resolução do problema foi baseado em uma estrutura não recorrente, com apenas uma camada escondida. A função de ativação utilizada em todas as camadas foi a linear, enquanto que a função de propagação escolhida foi a tangente hiperbólica, exceto na camada de saída, em que foi acoplada a função logística, devido à necessidade de se modelar a probabilidade de o indivíduo gerar perdas trabalhistas.

O treinamento da rede foi supervisionado *off-line,* baseado em correção do erro utilizando o modelo *backpropagation* por meio do algoritmo de otimização Gradiente Conjugado Escalado.

A função erro que foi alvo de minimização durante o processo de treinamento foi a de Entropia Cruzada, definida por:

$$E = -\sum_{n} \{t^n \ln y^n + (1 - t^n) \ln(1 - y^n)\} \tag{4.2}$$

Onde: n é o n-ésimo exemplo apresentado à rede.

y representa a probabilidade *a posteriori* P (Classe 1 | x).

$1 - y$ representa a probabilidade *a posteriori* P (Classe 2 | x).

x representa o vetor de entrada.

t = 1, se o vetor de entrada pertencer à Classe 1, e t = 0, se o vetor de entrada pertencer à Classe 2.

Sua derivada em relação à y^n é definida por:

$$\frac{\partial E}{\partial y^n} = \frac{(y^n - t^n)}{y^n(1 - y^n)} \tag{4.3}$$

O critério usado para a parada no treinamento foi o número de iterações, e, para detectar, *overffiting*, a performance da rede treinada, tanto em função do erro médio quadrático quanto do percentual de acertos de classificação, foi calculada também expondo-a ao subconjunto de validação.

Algumas redes foram implementadas no MatLab, variando-se, em cada uma delas, o número de neurônios na camada escondida, o número de iterações no processo de

treinamento e as variáveis de entrada. O número de neurônios na camada escondida foi estipulado também, a fim de evitar *overfitting*, de acordo com algumas indicações encontradas na teoria [14], como a média aritmética das entradas e saídas da rede ou (2 d+1), sendo *d* o número de entradas na rede.

A tabela a seguir ilustra as redes que foram implementadas, variando, em cada uma, o número de neurônios na camada escondida, número de iterações no processo de treinamento e variáveis de entrada:

Rede	Neurônios	Função de Saída	Otimização	Iterações	Variáveis
1	1	Logística	Gradiente Conjugado Escalado	500	Idade;Tempo
2	4	Logística	Gradiente Conjugado Escalado	500	Todas
3	2	Logística	Gradiente Conjugado Escalado	500	Idade;Tempo
4	2	Logística	Gradiente Conjugado Escalado	500	Todas
5	1	Logística	Gradiente Conjugado Escalado	500	Idade;Tempo;Dependentes
6	2	Logística	Gradiente Conjugado Escalado	500	Idade;Tempo;Dependentes
7	1	Logística	Gradiente Conjugado Escalado	1000	Idade;Tempo
8	4	Logística	Gradiente Conjugado Escalado	1000	Todas
9	2	Logística	Gradiente Conjugado Escalado	1000	Idade;Tempo
10	2	Logística	Gradiente Conjugado Escalado	1000	Todas
11	1	Logística	Gradiente Conjugado Escalado	1000	Idade;Tempo;Dependentes
12	2	Logística	Gradiente Conjugado Escalado	1000	Idade;Tempo;Dependentes
13	17	Logística	Gradiente Conjugado Escalado	1000	Todas
14	17	Logística	Gradiente Conjugado Escalado	1500	Todas
15	17	Logística	Gradiente Conjugado Escalado	2000	Todas

Tabela 4.1: Modelos de Rede Implementados

4.5. Análise dos Resultados

4.5.1. Análise dos Erros e Performance

Foram estipuladas duas medidas de performance para as redes implementadas: a função de erro médio quadrático, já definida anteriormente (ver 3.45), e o percentual de acerto de classificação para as duas classes de indivíduos: o que gera perda trabalhista e o que não gera.

As duas medidas de performance foram calculadas tanto para o subconjunto de validação quanto para o de teste, refletindo pequenas variações, o que é um bom indicador para demonstrar que não ocorreu *overfitting*.

Para se estipular o ponto de corte que define a fronteira de decisão para classificação dos indivíduos entre as duas classes (a que gera e a que não gera perda trabalhista), realizou-se de forma heurística algumas simulações com diferentes números, acompanhando-se a sensibilidade da performance de acerto de classificação em ambas as classes no subconjunto de validação.

Conforme pode ser visto na tabela a seguir, o número de 18,52% proporcionou um bom balanceamento entre a performance de acerto de classificação de ambas as classes, sendo portanto escolhido como ponto de corte.

	Ponto de Corte	18,52%	Ponto de Corte	20,00%	Ponto de Corte	19,00%	Ponto de Corte	18,00%	Ponto de Corte	17,00%
Grupo	Acerto Perda = 0	Acerto Perda = 1	Acerto Perda = 0	Acerto Perda = 1	Acerto Perda = 0	Acerto Perda = 1	Acerto Perda = 0	Acerto Perda = 1	Acerto Perda = 0	Acerto Perda = 1
1	50.54%	79.74%	52.24%	77.37%	51.23%	79.01%	49.85%	80.68%	48.23%	81.75%
2	61.67%	72.99%	64.92%	69.34%	62.80%	71.72%	60.76%	73.72%	58.26%	75.73%
3	51.23%	79.01%	54.43%	75.16%	52.04%	77.55%	50.19%	79.93%	48.03%	82.12%
4	64.53%	70.44%	67.00%	67.34%	65.42%	68.61%	63.60%	70.80%	60.99%	72.81%
5	50.08%	79.93%	52.27%	78.10%	50.69%	79.74%	49.42%	80.84%	47.73%	81.20%
6	51.35%	78.47%	54.74%	74.82%	52.35%	77.55%	50.45%	79.56%	48.38%	80.84%
7	50.27%	80.11%	52.16%	77.92%	50.93%	79.56%	49.42%	80.84%	47.68%	81.75%
8	64.53%	70.80%	67.35%	68.61%	65.50%	70.26%	63.84%	72.09%	61.68%	73.72%
9	51.20%	79.01%	54.24%	75.55%	52.04%	77.55%	50.23%	79.93%	48.07%	81.93%
10	51.04%	81.39%	53.28%	80.11%	51.54%	80.84%	50.31%	82.35%	48.50%	83.21%
11	49.96%	80.29%	52.23%	78.47%	50.62%	79.93%	49.34%	81.02%	47.65%	81.20%
12	51.35%	78.47%	54.78%	74.82%	52.35%	77.37%	50.54%	79.56%	48.42%	80.84%
13	62.49%	72.63%	66.15%	68.80%	63.72%	71.72%	61.45%	73.54%	59.41%	75.73%
14	65.50%	72.63%	67.85%	69.34%	66.35%	72.02%	64.69%	73.18%	62.68%	74.27%
15	60.41%	73.91%	63.69%	69.53%	61.49%	72.03%	58.91%	75.16%	56.63%	77.92%

Tabela 4.2: Ponto de Corte para Fronteira de Decisão

Em resumo, quando a saída da rede for acima do valor de 18,52%, classifica-se o indivíduo como 1 (gera perdas trabalhistas), caso contrário, é classificado como 0 (não gera perdas).

O quadro a seguir ilustra a performance dos modelos de redes implementados, tanto para o subconjunto de validação quanto para o de teste:

				Subconjunto de Validação			Subconjunto de Teste		
Rede	[illegible]	Iterações	Variáveis	Erro Médio Quadrático	Acerto Perda = 0	Acerto Perda = 1	Erro Médio Quadrático	Acerto Perda = 0	Acerto Perda = 1
1	1	500	Idade;Tempo	36.79%	50.54%	79.74%	38.10%	51.32%	79.80%
2	4	500	Todas	36.20%	61.87%	72.99%	37.34%	62.02%	74.55%
3	2	500	Idade;Tempo	36.76%	51.23%	79.01%	37.98%	51.86%	79.31%
4	2	500	Todas	36.12%	64.53%	70.44%	37.51%	63.48%	72.74%
5	1	500	Idade;Tempo;Dependentes	36.12%	50.03%	79.93%	38.03%	51.24%	80.30%
6	2	500	Idade;Tempo;Dependentes	36.76%	51.35%	78.47%	37.90%	52.31%	78.49%
7	1	1000	Idade;Tempo	36.79%	50.27%	80.11%	38.10%	51.24%	80.30%
8	4	1000	Todas	36.00%	64.53%	70.50%	37.25%	64.94%	73.56%
9	2	1000	Idade;Tempo	36.76%	51.20%	79.01%	37.98%	51.91%	79.15%
10	2	1000	Todas	36.55%	51.04%	81.39%	37.87%	51.95%	82.43%
11	1	1000	Idade;Tempo;Dependentes	36.79%	49.96%	80.29%	38.03%	51.20%	80.62%
12	2	1000	Idade;Tempo;Dependentes	36.76%	51.35%	78.47%	37.90%	52.35%	78.33%
13	17	1000	Todas	36.09%	62.49%	72.63%	37.14%	63.72%	74.22%
14	17	1500	Todas	36.08%	65.50%	72.63%	37.12%	67.15%	72.98%
15	17	2000	Todas	36.17%	60.41%	73.91%	37.49%	60.76%	74.86%

Tabela 4.3: Performance dos Modelos

É possível perceber que todas as redes implementadas apresentam pequenas variações em relação a seu erro médio quadrático, que fica em torno do intervalo de [37,25%/ 38,10%], no subconjunto de teste. O erro médio quadrático relativamente alto é um indício de que talvez não estejam incluídas no vetor de entrada algumas potenciais variáveis com poder explicativo.

A escolha do ponto de corte de 18,52% para classificação dos indivíduos entre as duas classes produziu efeitos diferentes em cada uma das redes, tendo destaque positivo

para a performance de acertos de classificação para os indivíduos que geraram perdas, bem superior à performance de acertos dos indivíduos que não geraram perdas.

As redes 8 e 14, em especial, apresentaram bons percentuais de acerto para os indivíduos que geraram perdas, 73,56% e 72,58% respectivamente, e foram às redes que melhor performance tiveram em relação às outras no tocante aos acertos dos indivíduos que não geraram perdas, 64,94% e 67,15%.

Além disso, essas duas redes apresentaram o menor erro quadrático entre todas as outras, 37,25% e 37,32%.

Outro ponto importante a se destacar é que a escolha do ponto de corte produziu performances de acertos de classificação estáveis nos subconjuntos de validação e teste.

Uma comparação interessante a se fazer para avaliar o poder discriminador das redes 8 e 14 é considerar que, caso o processo de classificação dos indivíduos fosse um passeio aleatório, com probabilidade de 50% de o indivíduo ser classificado como 1 (gera perdas) e de 50% de ser classificado como 0 (não gera perdas), a esperança de acertos de classificação para ambas as classes seria de 50%, inferior à performance de acertos das duas redes em questão.

Os valores razoavelmente altos relacionados ao erro médio quadrático apresentado por todas as redes e o menor percentual de acertos de classificação da classe de indivíduos que não geraram perdas em relação à classe de indivíduos que geraram perdas podem ser um indício de que as saídas produzidas pela rede para esta primeira classe de indíviduos não possuem correlação com o vetor de entradas. Para aprofundar essa análise foi realizado o Teste *Surrogate*, detalhado a seguir.

4.5.2. Teste *Surrogate*

O Teste *Surrogate* foi proposto por Theiler, em 1992 [16], e consiste em detectar a não-linearidade nos dados.

As hipóteses do Teste *Surrogate* no contexto deste estudo são:

H0: ausência de correlação entre as saídas produzidas pela rede e o vetor de entradas

H1: existência de correlação entre as saídas produzidas pela rede e o vetor de entradas

O Teste foi realizado seguindo algumas etapas. Primeiramente, a fim de excluir a correlação entre os dados, o vetor com as variáveis reais de saída do subconjunto de treinamento foi aleatoriamente ordenado 60 vezes, mantendo-se fixa a matriz com as variáveis de entrada. Dessa forma, geraram-se 60 novos subconjuntos de treinamento.

A rede 8, que apresentou a melhor performance, foi escolhida como base, sendo treinada 60 vezes (uma vez com cada subconjunto de treinamento).

Para cada uma das 60 simulações, calculou-se a performance da rede no subconjunto original de teste, tanto em relação ao erro médio quadrático quanto em relação ao percentual de acertos para cada uma das classes. O quadro a seguir ilustra com detalhes cada uma das simulações com a rede 8.

Rede 8 - *Surrogate*					
Subconjunto de Teste					
Simulação	Acerto Perda = 0	Acerto Perda = 1	Simulação	Acerto Perda = 0	Acerto Perda = 1
1	53.15%	50.43%	31	65.35%	34.51%
2	47.95%	46.26%	32	42.32%	59.03%
3	42.46%	54.04%	33	65.02%	34.80%
4	45.09%	53.11%	34	49.19%	53.72%
5	54.72%	46.53%	35	55.12%	42.09%
6	53.05%	44.30%	36	19.45%	78.46%
7	30.84%	67.25%	37	77.06%	21.81%
8	59.04%	43.54%	38	56.35%	42.88%
9	54.51%	47.68%	39	76.36%	20.56%
10	68.13%	34.67%	40	50.26%	52.43%
11	56.66%	46.19%	41	52.81%	49.75%
12	49.86%	52.14%	42	43.73%	52.68%
13	65.93%	35.42%	43	28.73%	70.55%
14	60.45%	38.46%	44	52.12%	50.00%
15	58.58%	44.08%	45	30.28%	69.08%
16	66.08%	33.61%	46	51.83%	49.47%
17	66.48%	35.52%	47	60.18%	42.02%
18	59.87%	40.83%	48	64.01%	35.65%
19	54.95%	49.59%	49	46.02%	53.43%
20	40.26%	60.07%	50	52.44%	47.66%
21	37.62%	62.03%	51	54.73%	44.87%
22	52.08%	54.32%	52	31.19%	70.56%
23	56.89%	44.35%	53	30.25%	71.16%
24	51.64%	48.46%	54	61.58%	36.52%
25	78.44%	23.45%	55	64.05%	34.67%
26	47.85%	50.68%	56	49.94%	48.22%
27	40.44%	58.47%	57	60.22%	42.13%
28	28.63%	70.81%	58	49.20%	53.13%
29	74.48%	25.39%	59	49.46%	52.80%
30	55.75%	44.54%	60	56.30%	43.02%

Tabela 4.4: Performance dos *Surrogates*

A estatística do Teste *Surrogate* é definida por:

$$S = \frac{|V_{original} - \mu_{surrogate}|}{\sigma_{surrogate}} \tag{4.4}$$

Onde: $V_{original}$ é o valor obtido com série original, nesse caso, o percentual de acerto de classificação das classes com a série original.

$\mu_{surrogate}$ é a média obtida nos *surrogates*, nesse caso, a média do percentual de acerto de classificação das classes dos *surrogates*.

$\sigma_{surrogate}$ é o desvio padrão obtido nos *surrogates*, nesse caso, o desvio padrão do percentual de acerto de classificação das classes dos *surrogates*.

As estatísticas (S) do Teste *Surrogate* para o percentual de acerto de classificação para os indivíduos que geraram e os que não geraeam perdas trabalhistas são:

	S	μ surrogate	σ surrogate	V original
Acerto Perda = 0	0.986824539	52.62%	12.48%	64.94%
Acerto Perda = 1	2.092435245	47.73%	12.35%	73.56%

Tabela 4.5: Estatística do Teste *Surrogate*

Com 95% de confiança, não se rejeita a hipótese nula para o caso em que os indivíduos não geraram perdas trabalhistas. Isso indica que as saídas produzidas pela rede não possuem correlação com o vetor de entradas, portanto o poder de discriminação do modelo para esta classe é discutível.

Já para o caso em que os indivíduos geraram perdas trabalhistas, com 95% de confiança, rejeita-se a hipótese nula , ou seja, existe correlação entre as saídas produzidas pela rede e o vetor de entradas, demostrando que o modelo conseguiu discriminar melhor esta classe.

Os resultados do Teste *Surrogate* indicam que o conjunto de dados relativos aos indivíduos que não geraram perdas é insuficiente e, possivelmente, devem existir mais variáveis com poder discriminador e explicativo, as quais não foram incluídas no modelo por não estarem disponíveis.

O resultado do Teste, rejeitando-se H0 para o caso dos indivíduos que geraram perdas trabalhistas, é coerente com o fato de que a rede 8 apresentou uma performance de acertos de classificação superior à performance de acertos em relação à outra classe de indivíduos. Para os indivíduos que geraram perdas, a rede foi capaz de reconhecer mais os padrões não-lineares nos dados.

4.6. Refinamento do Modelo

4.6.1. Votação em Comitê

Como as redes são inicializadas com pesos sinápticos aleatórios e diferem quanto a sua arquitetura, existe uma possibilidade de que, durante o treinamento, o algoritmo de otimização tenha encontrado na função erro um ponto de mínimo local e não global.

A utilização de redes votando em comitê pode atenuar esse efeito e isso pode ser provado da seguinte forma:

Seja L o número de modelos de rede treinados $y_i(x)$, com $i = 1,...,L$. A função de mapeamento de cada rede pode ser escrita em termos da função-alvo mais um erro:

$$y_i(x) = h(x) + \varepsilon_i(x) \tag{4.5}$$

A esperança da soma dos erros quadráticos do modelo de rede $y_i(x)$ pode ser definida por $E[\varepsilon_i^2]$, portanto o erro médio das redes atuando individualmente é dado por:

$$E_{médio} = \frac{1}{L}\sum_{i=1}^{L} E[\varepsilon_i^2] \tag{4.6}$$

Simulando um comitê no qual a saída será dada pela média das saídas das L *redes:*

$$y_{COM}(x) = \frac{1}{L}\sum_{i=1}^{L} y_i(x) \tag{4.7}$$

A esperança do erro produzido pela votação do comitê será dada por:

$$E_{COM} = E[(\frac{1}{L}\sum_{i=1}^{L} \varepsilon_i)^2] \tag{4.8}$$

Assumindo que os erros ε_i têm média 0 e são não correlacionados, tem-se:

$$E_{COM} = \frac{1}{L^2}\sum_{i=1}^{L} E(\varepsilon_i^2) = \frac{1}{L} E_{medio} \tag{4.9}$$

Neste estudo, a fim de refinar o modelo, as duas redes com melhor desempenho (rede 8 e 14) foram usadas num comitê, no qual a saída do comitê é a média das saídas das duas redes.

4.6.2. Análise dos Erros e Performance

O quadro a seguir ilustra a performance da votação em comitê com as redes 8 e 14:

	Subconjunto de Validação			Subconjunto de Teste		
Comite	Erro Médio Quadrático	Acerto Perda = 0	Acerto Perda = 1	Erro Médio Quadrático	Acerto Perda = 0	Acerto Perda = 1
Rede 8 + Rede 14	35.84%	64.53%	73.54%	37.10%	64.86%	73.89%

Tabela 4.6: Performance Votação em Comitê

Como se pôde perceber, houve uma redução muito sensível em relação ao erro médio quadrático e um aumento também sensível da performance de acertos de classificação dos indivíduos que geraram perdas trabalhistas, quando comparamos com os resultados das redes 8 e 14 atuando individualmente, conforme Tabela

Nesse caso específico, não foi possível reduzir na ordem de $\frac{1}{2}$ o erro médio das redes 8 e 14, atuando isoladamente quando as duas foram utilizadas numa votação em comitê.

O que explica essa redução sensível é que os erros desses dois modelos de redes são correlacionados, violando uma das premissas utilizadas na seção imediatamente anterior.

5. CONCLUSÕES

Verificou-se neste estudo que a utilização de Redes Neurais Artificiais (RNAs) para modelagem de um problema de estimação de probabilidade de um tipo específico de perda derivada de Risco Operacional, a perda trabalhista, apresentou um desempenho razoável.

O erro médio quadrático do modelo ajustado por meio de votação em comitê com as duas redes de melhor desempenho é de 37,10%, o que pode ser considerado alto para utilização em modelos de provisão gerencial. Entretanto, esse modelo preditor já é um primeiro passo considerável no que refere ao controle, gestão e precificação dessa categoria de evento de Risco Operacional, uma vez que é pequena a utilização de ferramentas quantitativas que dêem subsídio para o cálculo da exposição a esse risco.

O que se pode concluir principalmente após a realização do Teste *Surrogate* é que o conjunto de dados relacionados aos indivíduos que não geraram perdas é insuficiente para que o modelo implementado possa reconhecer padrões de forma satisfatória, portanto devem-se focar esforços futuros na coleta de novos dados e principalmente de novas variáveis com poder explicativo e discriminador para essa classe de indivíduos, a fim de que o erro médio quadrático dos novos modelos com RNAs seja reduzido.

Os resultados do Teste *Surrogate* para o conjunto de dados relacionados aos indivíduos que não geraram perdas e a performance de acertos de classificação para essa classe com a utilização de RNAs, em torno de 64,86%, também são um ponto de motivação para aplicação e teste de outras técnicas preditivas e discriminantes, como a Análise de Cluster.

Em relação ao poder discriminador do modelo ajustado por meio de votação em comitê com as duas redes de melhor desempenho, pode-se concluir que o resultado foi relativamente bom para a classe de indivíduos que geraram perdas, acerto de 73,89%. Esse desempenho do modelo em relação à classificação de indivíduos que geraram perdas o credencia para um possível uso no auxílio para a tomada de decisões que envolvam grandes reestruturações e aquisições que, porventura, provoquem redução de pessoal.

Outra possível aplicação desse poder discriminador da RNA implementada neste estudo é dar subsídio ao cálculo de uma provisão gerencial que seria constituída apenas pelos indivíduos classificados como geradores de perdas trabalhistas por meio da Rede. Esse modelo de provisão já seria muito mais sensível ao risco, uma vez que a provisão contábil só é constituída no momento em que uma ação trabalhista é iniciada.

Finalmente, conclui-se que as experiências deste estudo utilizando RNAs para modelagem de um problema envolvendo Risco Operacional foram promissoras, sendo ponto de partida para refinamento dos resultados em futuros estudos e já servindo para a resolução parcial de alguns aspectos práticos relacionados ao controle e gestão desse tipo de risco.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] The RMA Journal, mar. 2003.

[2] Basel Committee on Banking Supervision. **Overview of the New Basle Capital Accord.** Consultative Document, Basle, apr. 2003.

[3] Basel Committee on Banking Supervision. **International Convergence of Capital Measurements and Capital Standards, A Revised Framework.** Basle, june 2004.

[4] Basel Committee on Banking Supervision. **QIS 2 – Operational loss risk data.** Basle, may 2001.

[5] Basel Committee on Banking Supervision. ***The 2002 Loss Data Collection Exercise for Operational Summary of the Data Collected.*** Basle, mar. 2003.

[6] HAYKIN, S. **Redes Neurais, Princípios e Prática.** 2. ed. Porto Alegre: Bookman, 2001.

[7] MCCULLOCH, W. W. **A Logical Calculus of the Ideas Imminent in Nervous Activity.** ***Bulletin of Mathematical Biophysics.*** 1943. v. 5.

[8] MINSKY, M.L.; PAPER, S.A. **Perceptrons: an introduction to Computacional Geometry.** 3. ed. Massachusssetts: MIT press, 1988.

[9] HOPFIELD, J.J. **Neural Network and Physicals Sistems with Emergent Collective Computation Abilities.** National Academy of Science, Anais, United States, 1982.

[10] MCCLELLAND, J.L.; RUMELHART, D.E. **Exploration in Parallel Distribuited Processing: A Handbook of Models, Programs and Exercises.** The Massachussetts Institute of Technology, 1988.

[11] CAUDILL, M. **Expert Network.** Bite, oct. 1991.

[12] BISHOP, C.M. **Neural Networks for Pattern Recognition.** Oxford, 1995.

[13] HAYKIN, S. **Neural Networks: A Comprehensive Foundation.** Prentice Hall International Edition, 1994.

[14] SILVA, E.; OLIVEIRA, A. C.**Dicas para Configuração de Redes Neurais.**

[15] DIEBOLT, C.; KYRTSOU, C.Ä **Survey on Cycles and Chaos** (Part II).

[16] THEILER, J.; EUBANK, S.; LONGTIN, A.; GALDRIKIAN, B. **Testing for Nonlinearity in Time Series: the method of surrogate data.** *Physica D*, n. 58.

[17] ZANETTI, L.A.; ALMEIDA, F. C. **Exploração do Uso de redes Neurais na Previsão do Comportamento de Ativos Financeiros.** III SemeAd.

[18] ALVES, C.A.M. **Uma Ferramenta de Extração de Regras de Redes Neurais.** Mestrado em Ciências em Engenharia Civil (UFRJ). *COPPE*, Rio de Janeiro, 2001.

[19] COATS, P.K.; FANT, F. **Recognizing Financial Distress Patterns Using a Neural Netowk Tool.** *Financial Management.* n. 3. autumm 1993. v. 22

[20] HORNIK, K. **Approximation Capabilities of MultiLayer Feedforward Networks.** *Neural Networks.* 1991. v. 4.

[21] VICENTE, R. **Redes Neurais para Inferência Estatística.** IME-FEA/USP, jun. 2002.

[22] MØLLER, M. **A scaled conjugate gradient algorithm for fast supervised learning.** *Neural Networks* 6 (4), 525-533, 1993.

[23] FLETCHER, R.. **Practical Methods of Optimization** (Second ed.). New York: John Willey, 1987

[24] FILHO, E.F.M.. **Evolutionary design of MLP neural networks architectures.** *Neural Networks.* 1997. Proceedings. IVth Brazilian Symposium on 3-5 Dec 1997. pages 58-65

APÊNDICES

- **Algoritmos do MatLab**

Segue um trecho de um dos algoritmos implementados neste estudo:

```
% Rede 1
% Entradas: Idade + Tempo
% Saída: Logística
% Otimização: Gradiente Conjugado Escalado
% Iterações: 500
% Neurônios: 1 (Média das Entradas)
load 'data.txt'
intreino=data(1:9425,1:2);
invalida=data(9426:12567,1:2);
inteste=data(12568:15709,1:2);
outtreino=data(1:9425,9);
outvalida=data(9426:12567,9);
outteste=data(12568:15709,9);
net1=mlp(2,1,1,'logistic');
options=foptions;
options(1)=1;
options(14)=500;
[net1,options,erro]=netopt(net1,options,intreino,outtreino,'scg');
outmodelvalida1=mlpfwd(net1,invalida);
deltasvalida1=(outmodelvalida1-outvalida);
deltaqmvalida1=sqrt(sum(deltasvalida1.*deltasvalida1)/size(outvalida,1))
outmodel1=mlpfwd(net1,inteste);
deltas1=(outmodel1-outteste);
deltaqm1=sqrt(sum(deltas1.*deltas1)/size(outteste,1))
% Rede 2
% Entradas: Todas
% Saída: Logística
% Otimização: Gradiente Conjugado Escalado
% Iterações: 500
```

```
% Neurônios: 4 (Média das Entradas)
intreino=data(1:9425,1:8);
invalida=data(9426:12567,1:8);
inteste=data(12568:15709,1:8);
net2=mlp(8,4,1,'logistic');
options=foptions;
options(1)=1;
options(14)=500;
[net2,options,erro]=netopt(net2,options,intreino,outtreino,'scg');
outmodelvalida2=mlpfwd(net2,invalida);
deltasvalida2=(outmodelvalida2-outvalida);
deltaqmvalida2=sqrt(sum(deltasvalida2.*deltasvalida2)/size(outvalida,1))
outmodel2=mlpfwd(net2,inteste);
deltas2=(outmodel2-outteste);
deltaqm2=sqrt(sum(deltas2.*deltas2)/size(outteste,1))
% Rede 3
% Entradas:Idade + Tempo
% Saída: Logística
% Otimizaçao: Gradiente Conjugado Escalado
% Iteraçoes: 500
% Neurônios: 2
intreino=data(1:9425,1:2);
invalida=data(9426:12567,1:2);
inteste=data(12568:15709,1:2);
net3=mlp(2,2,1,'logistic');
options=foptions;
options(1)=1;
options(14)=500;
[net3,options,erro]=netopt(net3,options,intreino,outtreino,'scg');
outmodelvalida3=mlpfwd(net3,invalida);
deltasvalida3=(outmodelvalida3-outvalida);
deltaqmvalida3=sqrt(sum(deltasvalida3.*deltasvalida3)/size(outvalida,1))
outmodel3=mlpfwd(net3,inteste);
deltas3=(outmodel3-outteste);
deltaqm3=sqrt(sum(deltas3.*deltas3)/size(outteste,1))
```

- **Algoritmo de Otimização Gradiente Conjugado Escalado**

Fonte: [21]

A amplitude de cada passo de otimização para uma superfície quadrática pode ser expressa por:

$$\alpha_j = -\frac{d_j.\nabla E(w_j)}{d_j.Hd_j} \qquad (1)$$

Para evitar o cálculo da Hessiana acima ou a utilização de otimizações unidimensionais a cada passo, foi criado por Møller [22] o algorítmo de gradientes conjugados escalados.

O algorítmo realiza uma aproximação numérica para o termo que envolve a Hessiana em (1). Seja $\varepsilon = \varepsilon_0 / \|d_j\|$, onde ε_0 é uma quantidade positiva e pequena, observa-se que:

$$\nabla E(w_j + \varepsilon d_j) \approx \nabla E(w_j) + \varepsilon H d_j \qquad (2)$$

A aproximação numérica para (1) é então:

$$Hd_j \approx \frac{\nabla E(w_j + \varepsilon d_j) - \nabla E(w_j)}{\varepsilon} \qquad (3)$$

Para que esta otimização funcione duas condições devem ser válidas:

(1) A função a ser minimizada é quadrática

(2) A Hessiana é positiva definida

Para superfícies genéricas, nada garante que a Hessiana é positiva definida, e um passo de otimização usando (3) pode aumentar o valor da função a ser minimizada em algumas direções. A fim de usar a aproximação numérica (3), a equação 1 pode ser alterada para:

$$\alpha_j = -\frac{d_j.\nabla E(w_j)}{d_j.Hd_j + \gamma_j \|d_j\|^2} \tag{4}$$

Onde o parâmetro γ_j regula a escala de cada passo, e é inicializado como $\gamma_j = 1$. O tamanho do parâmetro de escala é inversamente proporcional à amplitude do passo de otimização.

Fletcher [23] introduziu uma medida de grau de similaridade local entre a função e uma superfície quadrática:

$$\Delta_j = \frac{E(w_j) - E(w_j + \alpha_j d_j)}{E(w_j) - E_Q(w_j + \alpha_j d_j)} \tag{5}$$

Onde $E_Q(w_j + \alpha_j d_j) = E(w_j) + \alpha_j d_j \nabla E(w_j) + \frac{1}{2}\alpha_j^2 d_j.Hd_j$

Ao substituir (1) em (5), a equação pode ser simplificada para:

$$\Delta_j = \frac{2[E(w_j) - E(w_j + \alpha_j d_j)]}{\alpha_j d_j.d_j} \tag{6}$$

Através destes parâmetros de comparação, são feitas escolhas heurísticas para o parâmetro de escala a cada passo da otimização, conforme o que segue:

a. Se $\Delta_j > 0.75$ então $\gamma_{j+1} = \frac{\gamma_j}{2}$;

b. Se $0.25 < \Delta_j < 0.75$ então $\gamma_{j+1} = \gamma_j$;

c. Se $\Delta_j < 0.25$ então $\gamma_{j+1} = 4\gamma_j$;

d. Se $\Delta_j < 0$ então $\gamma_{j+1} = 4\gamma_j$, não sendo o passo de otimização realizado

Finalmente, o algorítmo de gradientes conjugados escalados pode ser entendido como sendo a sequência das seguintes etapas:

1. Escolha um ponto inicial no espaço dos parâmetros w_1;
2. Inicialize o parâmetro de escala em $\gamma_1 = 1$;
3. Calcule o gradiente no ponto inicial e faça a primeira direção de orimização ser $d_1 = -\nabla E(w_1)$;
4. Para descobrir a amplitude de cada passo na direção d_1 utilizar (3) e (4);
5. Calcule $w_2 = w_1 + \alpha_1 d_1$;
6. Pare se os critérios de convergência forem satisfeitos;
7. Calcule o parâmetro de comparação Δ;
8. Atualize o parâmetro de escala γ de acordo com as condições acima;
9. Calcule $\nabla E(w_2)$;
10. A nova direção é dada por $d_2 = -\nabla E(w_2) + \beta_1 d_1$, com β_1 definido pela relação de *Polak-Ribiere;*
11. Vá para o passo 3 para o novo d e para o novo parâmetro de escala γ.