



**Fundação Educacional do Município de Assis  
Instituto Municipal de Ensino Superior de Assis  
Campus "José Santilli Sobrinho"**

**Rafael Felisbino**

**Inteligência Artificial e Redes Neurais: Conceitos e Aplicações**

Assis, SP

2012

**Rafael Felisbino**

## **Inteligência Artificial e Redes Neurais: Conceitos e Aplicações**

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Bacharelado em Ciência da Computação.

Orientador: Prof. Dr. Luiz Carlos Begosso

Área de Concentração: Informática.

Assis, SP

2012

## FICHA CATALOGRÁFICA

FELISBINO, Rafael

Inteligência Artificial e Redes Neurais: Conceitos e Aplicações / Rafael Felisbino. Fundação Educacional do Município de Assis – FEMA – Assis, 2012.

47p.

Orientador: Prof. Dr. Luiz Carlos Begosso

Trabalho de Conclusão de Curso – Instituto Municipal de Ensino Superior de Assis – IMESA.

CDD 001.6

Biblioteca FEMA

# Inteligência Artificial e Redes Neurais: Conceitos e Aplicações

**Rafael Felisbino**

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Bacharelado em Ciências da Computação, analisado pela seguinte comissão examinadora:

Orientador: Prof. Dr. Luiz Carlos Begosso

Avaliador: Prof. Domingos de Carvalho Villela Junior

Assis, SP

2012

## DEDICATÓRIA

Dedico este trabalho de conclusão de curso ao meu amigo Pedro Carvalho Dias.

## **AGRADECIMENTOS**

Agradeço a deus, minha família, meus amigos e todos aqueles que me acompanharam por essa longa jornada de estudos.

## RESUMO

Nos últimos anos houve um grande crescimento da Inteligência Artificial, principalmente quando falamos de Redes Neurais Artificiais, as quais procuram ao máximo a aproximação de simular o mesmo esquema de nossos neurônios.

As Redes Neurais Artificiais vem propondo grandes benefícios a nós seres-humanos.

Para o desenvolvimento desse projeto conduziu-se uma ampla pesquisa na área da Inteligência Artificial, mais especificamente de Redes Neurais Artificiais. Desenvolveu-se um programa de reconhecimento de padrões para auxílio no entendimento do trabalho. Este programa foi desenvolvido na linguagem JAVA e utilizou o banco de dados PostgreSQL.

**Palavras Chave:** Inteligência Artificial, Redes Neurais, Redes Neurais Artificiais.

## ABSTRACT

Recent years have seen a large growth of Artificial Intelligence, especially when it comes to artificial neural networks, which seek the maximum approximation to simulate the same pattern as our neurons.

Artificial Neural Networks has been proposing major benefits to us-human beings.

This project will be done extensive research in the field of Artificial Intelligence, specifically talking about neural networks, and also will develop a pattern recognition program to aid in the understanding of the project. This program will be developed in JAVA using the PostgreSQL database.

**Keywords:** Artificial Intelligence, Neural Networks, Artificial Neural Networks.

## LISTA DE ILUSTRAÇÕES

Figura 01 – Neurônio Artificial projetado por McCulloch e Pitts.....	12
Figura 02 – Rede Perceptron proposta por Frank Rosenblatt.....	13
Figura 03 – Rede MLP .....	14
Figura 04 – Esquema dos constituintes da célula neural .....	20
Figura 05 – Rede Perceptron .....	22
Figura 06 – Organização em camadas .....	25
Figura 07 – Fase de Propagação.....	26
Figura 08 – Fase de Retropropagação.....	27
Figura 09 – Modelagem do Problema .....	29
Figura 10 – Casos de Uso.....	30
Figura 11 – Diagrama ER.....	31
Figura 12 – Diagrama de Classes.....	32
Figura 13 – Diagrama de Sequência.....	33
Figura 14 – Diagrama de Atividades .....	34
Figura 15 – Estrutura SGBD Relacional.....	36
Figura 16 – Criação do banco de dados na ferramenta pgAdmin III .....	37
Figura 17 – Tela inicial do programa .....	38
Figura 18 – Cadastramento Espécie .....	39
Figura 19 – Tela de Cadastramento de Espécie .....	40
Figura 20 – Consulta Espécie .....	41
Figura 21 – Tela de Consulta de Espécie .....	42
Figura 22 – Menu Testes .....	43
Figura 23 – Tela de Testes.....	44

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>12</b>
1.1 - Objetivos .....	15
1.2 - Justificativa .....	15
1.3 - Motivação .....	15
1.4 – Estrutura do Trabalho .....	16
<b>2. FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>17</b>
2.1 – Inteligência Artificial .....	17
2.2 – Redes Neurais Artificiais .....	19
2.2.1 – Rede Neural Biológica .....	19
2.2.2 – Rede Neural Artificial .....	21
2.2.3 – Perceptron Multicamadas .....	23
2.3 – Algoritmo “ <i>Backpropagation</i> ” .....	25
<b>3. DESENVOLVIMENTO DO PROJETO .....</b>	<b>28</b>
3.1 – Descrição do Problema .....	28
3.2 – Modelagem do Problema .....	28
3.3 – Especificação .....	29
3.3.1 – Diagrama de Casos de Uso .....	30
3.3.2 – Diagrama de ER .....	31
3.3.3 – Diagrama de Classe .....	32
3.3.4 – Diagrama de Sequência .....	33

3.3.5 – Diagrama de Atividade .....	34
3.4 – Implementações .....	36
3.4.1 – Criação do banco de dados .....	36
3.4.1.1 – Preparação do Banco de Dados PostgreSQL .....	37
3.4.2 – Desenvolvimento do programa .....	38
3.4.2.1 – Visão Geral do Programa .....	38
3.4.2 .2– Cadastramento de Espécies e Reinos.....	39
3.4.2.3 – Consultas de Espécies e Reinos .....	41
3.4.2.4 – Testes .....	42
<b>4. CONCLUSÃO .....</b>	<b>45</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>46</b>

## 1. INTRODUÇÃO

Neste capítulo será feito um breve resumo sobre a abordagem do presente trabalho. Espera-se ressaltar a importância da área de Inteligência Artificial (IA) para a atualidade e também para o futuro. Inicialmente, serão abordados aspectos históricos da Inteligência Artificial e das Redes Neurais Artificiais. Destacaremos a rede Perceptron e o algoritmo de treinamento *backpropagation*.

McCulloch-Pitts propuseram um modelo matemático do neurônio biológico, em 1943. O neurônio McCulloch-Pitts (M-P) é uma aproximação útil do neurônio real, servindo até hoje como um bloco construtivo básico para construções de algoritmos de redes neurais (Ludwig e Montgomery, 2007). A Figura 1 ilustra o neurônio artificial concebido por McCulloch e Pitts.

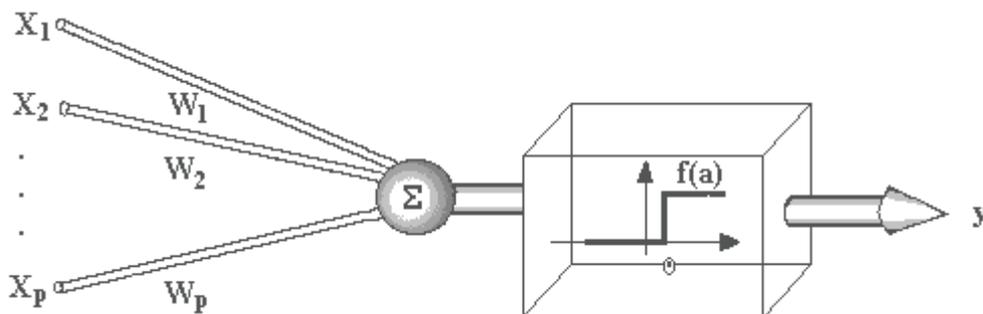


Figura 1 – Neurônio Artificial projetado por McCulloch e Pitts. (Tatibana e Kaetsu, 2012).

Inteligência Artificial ainda é um termo não muito fácil de definir, pois encontra um vasto campo de conceituações e nem todas elas são convergentes. Porém segundo o dicionário Michaelis, Inteligência é a “Faculdade de entender, pensar, raciocinar e interpretar”, ou simplesmente “entendimento, intelecto”. O termo Artificial é definido, também pelo senso comum, como “Produzido ou efetuado pela habilidade do homem para imitar a natureza”. Logo, Inteligência Artificial foi definida por Winston (1987), como “o estudo de conceitos que permitem aos computadores serem inteligente”.

Ludwig e Montgomery (2007) ressaltam que a Inteligência Artificial começou a ser reconhecida como ciência apenas em 1956 em uma conferência de verão em Dartmouth College, nos Estados Unidos, porém anos antes Alan Turing já havia feito alguns estudos sobre o termo “Inteligência Artificial”. Nesta conferência, foi proposto pelos autores a realização de “um estudo durante dois meses, por dez homens, sobre o tópico Inteligência Artificial”.

Em 1958 surge uma das primeiras redes neurais, a rede perceptron, proposta por Frank Rosenblatt. Esta rede consistia na estrutura utilizada no neurônio de M-P acrescida de uma regra de aprendizagem. O acréscimo desta regra de aprendizagem contribuiu para que a rede perceptron apresentasse importantes aspectos inteligentes (Braga, Ludemir e Carvalho, 2000). A Figura 2 ilustra a Rede Perceptron.

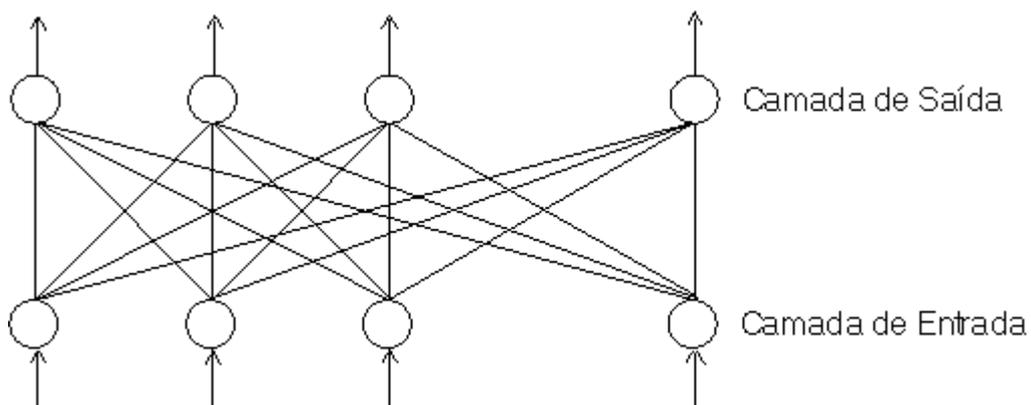


Figura 2 – Rede Perceptron proposta por Frank Rosenblatt. (Tatibana e Kaetsu, 2012).

Braga, Ludemir e Carvalho (2000), ressaltam que a rede perceptron simples não era capaz de solucionar problemas linearmente não separáveis, não era possível separar as classes linearmente, baseado nos estudos feitos por Minsky e Papert em 1960. Em 1986 Rumelhart, Hilton e Williams desenvolveram o algoritmo de treinamento backpropagation, mostrando que era possível treinar eficientemente

redes com camadas intermediárias, o que resultou no modelo de redes neurais mais utilizados atualmente, a rede Perceptron Multi-Camadas (Multilayer Perceptron - MLP), ao qual são capazes de solucionar problemas linearmente não separáveis (Tatibana e Kaetsu, 2012). A Figura 3 ilustra a estrutura de uma rede perceptron multicamadas.

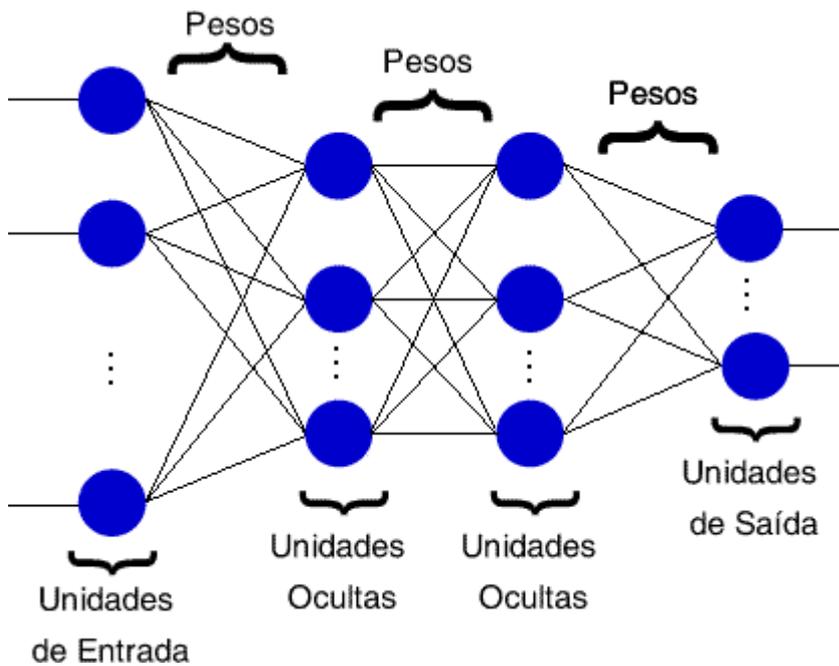


Figura 03 – Rede MLP ( extraído de LNCC – Laboratório Nacional de Computação Científica, 2012).

Esta ciência denominada de “Inteligência Artificial” vem crescendo muito com grandes perspectivas de alcançar o objetivo proposto pelos seus pesquisadores. Com a revolução industrial, o advento da tecnologia e o avanço dos processos automáticos, fizeram com que vários estudos fossem propostos por pessoas a quais buscavam ampliar as estratégias de redução de problemas de controle de sistemas que apresentavam características não lineares, assim como as perspectivas de estruturas de máquinas inteligentes que pudessem substituir o homem em várias atividades. Com a tecnologia que temos hoje, é possível criar máquinas as quais podem mostrar conhecimento, raciocínio e aprendizado, nos quais são os elementos

principais de um sistema inteligente. Pensando sobre estas possibilidades é que vários pesquisadores demonstraram interesse pela área.

O futuro da IA ainda é desconhecido, não se sabe até onde pode ir, mas os cientistas tentam ao máximo aproximar modelos computacionais do desempenho inteligente de um ser humano.

### **1.1 – Objetivos**

Esse trabalho de conclusão de curso tem por objetivo apresentar um estudo sobre as redes neurais artificiais, suas aplicações nas mais variadas áreas e, a partir desse estudo, elaborar um projeto e implementação de RNA capaz de executar a classificação de N indivíduos nas classes dos Seres Vivos.

### **1.2 - Justificativa**

A realização deste trabalho se da pelo fato da Inteligência Artificial ser um termo muito polêmico e ao mesmo tempo desafiador. As discussões sobre a capacidade da Inteligência Artificial em gerar artefatos computacionais com características próximas ou idênticas à do ser humano é o que permite imaginar inúmeras possibilidades de soluções de problemas não lineares na computação.

### **1.2 – Motivação**

A motivação para a escolha deste tema foi pelo fato de sua tecnologia conter habilidades e possibilidades muito atraentes, quais sejam: representação do conhecimento, raciocínio e aprendizado. Aliado a estas características a IA é aderente a uma tecnologia que está em grande crescimento em mercados industriais, bolsa de valores, sistemas de tempo real, etc.

## **1.4 – Estrutura do Trabalho**

O trabalho está organizado em quatro capítulos, conforme descrito a seguir.

O primeiro capítulo, a presente introdução, descrevem-se os objetivos, a motivação e as justificativas para o desenvolvimento do trabalho.

O segundo capítulo apresenta a fundamentação teórica necessária para compreender conceitos específicos do trabalho onde são abordados os seguintes assuntos: redes neurais artificiais (RNA), aplicações das RNA e redes perceptron.

O terceiro capítulo refere-se ao desenvolvimento da proposta aqui estabelecida, ou seja, a construção de um modelo de RNA capaz de executar a classificação de padrões. Será apresentada a modelagem do problema bem como sua especificação e a implementação.

Por fim, o capítulo quatro apresenta as conclusões e os trabalhos futuros.

## 2. FUNDAMENTAÇÃO TEÓRICA

### 2.1 – Inteligência Artificial

Os primeiros estudos sobre Inteligência Artificial surgiram na segunda guerra mundial, década de 40, quando havia uma grande necessidade de se desenvolver uma tecnologia voltada para a análise de balística, quebra de códigos e cálculos para projetar a bomba atômica. Surgiam então os primeiros grandes projetos de construção de computadores, assim chamados por serem máquinas utilizadas para fazer cálculos (cômputos). Nesta época o computador era apenas de âmbito militar e científico, porém logo após a II Guerra Mundial começou a ser gradualmente utilizado em empresas, indústrias, universidades e etc. A diversidade de aplicações estimulou pesquisas de software, hardware e linguagem de programação (Lima e Labidi, 2012).

Alan Turing, em 1950, deu o ponta pé inicial para a IA, quando publicou na revista filosófica *Mind* um artigo chamado “*Computing Machine and Intelligence*”. Neste artigo, Alan Turing, apresentou, pela primeira vez, o que hoje é conhecido por Teste de Turing. Com este teste, pretendia-se descobrir se uma máquina pode ou não pensar (UEM – Universidade Estadual de Maringá, 2012).

De acordo com Lima e Labidi (2012), foi em 1956, nos Estados Unidos, que John McCarthy reuniu em uma conferência proferida ao Dartmouth College, na Universidade de New Hampshire, vários pesquisadores de renome para estudar o que foi denominado por Minsky, McCarthy, Newell e Simon de Inteligência Artificial (IA), expressão utilizada para designar um tipo de inteligência construída pelo homem para dotar a máquina de comportamentos inteligentes.

Nesta mesma conferência, surgiram dois paradigmas da Inteligência Artificial: a Abordagem Cognitiva e a Abordagem Conexionista.

Abordagem Cognitiva, também denominada de descendente ou simbolista, dá ênfase aos processos cognitivos, ou seja, a forma como o ser humano raciocina.

Objetiva encontrar uma explicação para comportamentos inteligentes baseados em aspectos psicológicos e processos algorítmicos. Os pioneiros dessa corrente foram Jonh McCarthy, Marvin Minsky, Newell e Simon. As primeiras modelagens da inteligência surgiram na década de 50 e tiveram como base as regras de produção e a lógica dos predicados. A formalização da lógica facilitou o processo de formalização e representação dos conhecimentos a serem utilizados pelos programas de computador. Inicialmente, esses conhecimentos se restringiram a esquemas de raciocínios para jogos, aplicações matemáticas e simuladores.

Abordagem Conexionista, também denominada de biológica ou ascendente, dá ênfase no modelo de funcionamento do cérebro, dos neurônios e das conexões neurais. Os pioneiros dessa corrente foram McCulloch, Pitts, Hebb, Rosenblatt e Widrow.

Alguns campos de estudo no desenvolvimento da IA foram: Algoritmos Genéticos, Programação evolutiva, Lógica FUZZY, Sistemas baseados em conhecimento, Programação Genética, Raciocínio baseado em casos, Redes Neurais Artificiais (RNA), dentre outros.

Ferrari (2005) divide as principais fases de desenvolvimento da IA de acordo com os seus objetivos, métodos e limitações:

### **Clássica 1956-1970**

- Objetivo: simular a inteligência humana.
- Métodos: solucionadores gerais de problemas (GPS) e lógica.
- Limitação: subestimação da complexidade computacional dos problemas.

### **Romântica 1980-1990**

- Objetivo: simular a inteligência humana em situações pré-determinadas.
- Métodos: formalismo de representação de conhecimento adaptados aos tipos de problema, mecanismos de ligação procedural visando maior eficiência computacional.

- Limitação: subestimação de quantidade de conhecimento necessária mesmo para tratar de problemas muito simples.

### **Moderna 1980-1990**

- Objetivo: simular o comportamento de um especialista num domínio específico.
- Métodos: sistemas de regras, representação da incerteza, conexionismo.
- Limitação: subestimação da complexidade do problema de aquisição de conhecimento.

## **2.2 – Redes Neurais Artificiais**

### **2.2.1 – Rede Neural Biológica**

O cérebro humano é considerado como se fosse um processador, sendo composto por aproximadamente 10 bilhões de neurônios. Todas as funções e movimentos do organismo estão relacionados ao funcionamento destas pequenas células (neurônios). Os neurônios estão conectados uns aos outros através de sinapses, e juntos formam uma grande rede, chamada Rede Neural. Esta grande rede proporciona uma admirável capacidade de processamento e armazenamento de informação (Tatibana e Kaetsu 2012).

O sistema nervoso é formado por um conjunto extremamente complexo de neurônios. Nos neurônios a comunicação é realizada através de impulsos, quando um impulso é recebido, o neurônio o processa, e passado um limite de ação, dispara um segundo impulso que produz uma substância neurotransmissora o qual flui do corpo celular para o axônio (que por sua vez pode ou não estar conectado a um dendrito de outra célula). O neurônio que transmite o pulso pode controlar a frequência de pulsos aumentando ou diminuindo a polaridade na membrana pós-sináptica. Eles têm um papel essencial na determinação do funcionamento, comportamento e do raciocínio do ser humano. Ao contrário das redes neurais artificiais, redes neurais naturais não transmitem sinais negativos, sua ativação é

medida pela frequência com que emite pulsos, frequência esta de pulsos contínuos e positivos. As redes naturais não são uniformes como as redes artificiais, e apresentam uniformidade apenas em alguns pontos do organismo. Seus pulsos não são síncronos ou assíncronos, devido ao fato de não serem contínuos, o que a difere de redes artificiais (Tatibana e Kaetsu 2012).

Os principais componentes dos neurônios naturais são:

Os dentritos, que tem por função, receber os estímulos transmitidos pelos outros neurônios;

O corpo de neurônio, também chamado de *somma*, que é responsável por coletar e combinar informações vindas de outros neurônios;

E finalmente o axônio, que é constituído de uma fibra tubular que pode alcançar até alguns metros, e é responsável por transmitir os estímulos para outras células.

A Figura 4 ilustra os principais componentes de um neurônio natural.

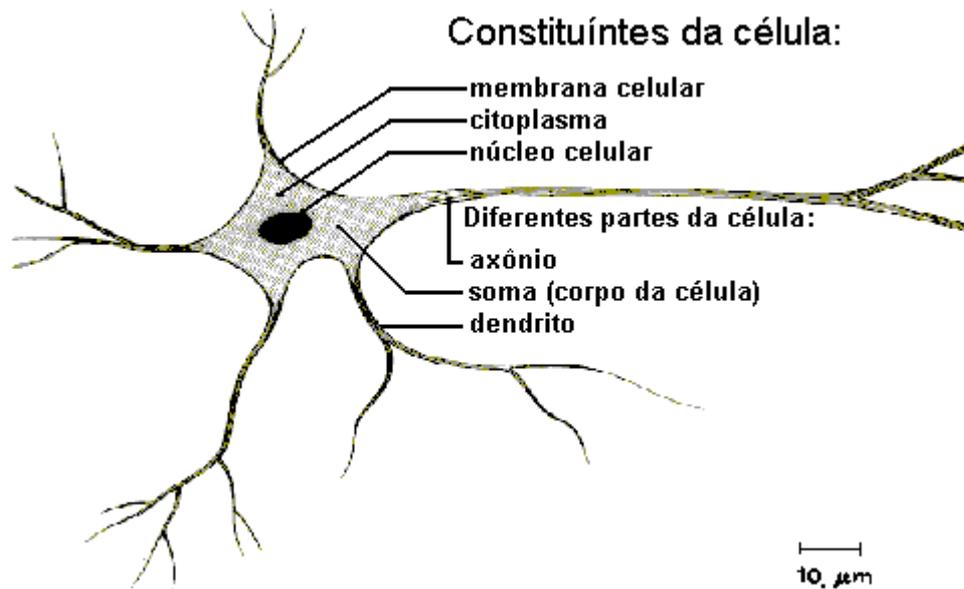


Figura 04 – Esquema dos constituintes da célula neural (extraído de Tatibana e Kaetsu 2012).

### 2.2.2 – Rede Neural Artificial

As Redes Neurais Artificiais são provavelmente a mais antiga técnica de IA. Surgiu na década de 40, por Walter Pitts e McCulloch, o primeiro matemático e o segundo neurofisiologista. A ideia era fazer uma analogia entre neurônios biológicos e circuitos eletrônicos, capazes de simular conexões sinápticas pelo uso de resistores variáveis e amplificadores. Foi ainda nesta década de 40, que o matemático Johann Von Neumann, propôs a arquitetura dos computadores eletrônicos atuais, em seu trabalho intitulado “*Preliminary Discussion of the Logic Design of an Electronic Computing Instrument*”. Uma máquina de processamento sequencial com CPU e memória separados e um ponteiro que registra o endereço do próximo comando a ser executado, (Ludwig e Montgomery, 2007).

Em 1949 Donald Hebb escreveu um livro intitulado “*The Organization of Behavior*” (A Organização do Comportamento) que perseguia a ideia de que o condicionamento psicológico clássico está presente em qualquer parte dos animais pelo fato de que esta é uma propriedade de neurônios individuais. Suas ideias não eram completamente novas, mas Hebb foi o primeiro a propor uma lei de aprendizagem específica para as sinapses dos neurônios (Ludwig e Montgomery, 2007).

Ludwig e Montgomery (2007) ressaltam que Marvin e Papert em 1951, construíram o primeiro neuro computador, denominado Snark. O Snark operava com sucesso a partir de um ponto de partida técnico, ajustando seus pesos automaticamente, entretanto, ele nunca executou qualquer função de processamento de informação interessante, mas serviu de inspiração para as ideias de estruturas que o sucederam.

De acordo com Braga, Ludemir e Carvalho (2000), em 1958, Frank Rosenblatt criou a rede Perceptron, e neste modelo dizia, que, se fossem acrescentadas de sinapses ajustáveis, as RNAs com nodos MP poderiam ser treinadas para classificar certos tipos de padrões. Rosenblatt descreveu uma topologia de RNA, estruturas de ligação entre os nodos e, o mais importante, propôs um algoritmo para treinar a rede

para executar determinados tipos de funções. O perceptron simples descrito por Rosenblatt possui três camadas: a primeira recebe as entradas do exterior e possui conexões fixas; a segunda recebe impulsos da primeira através de conexões cuja eficiência de transmissão (peso) é ajustável e, por sua vez, envia saídas para a terceira camada (resposta).

A Figura 5 mostra a estrutura de uma rede Perceptron.

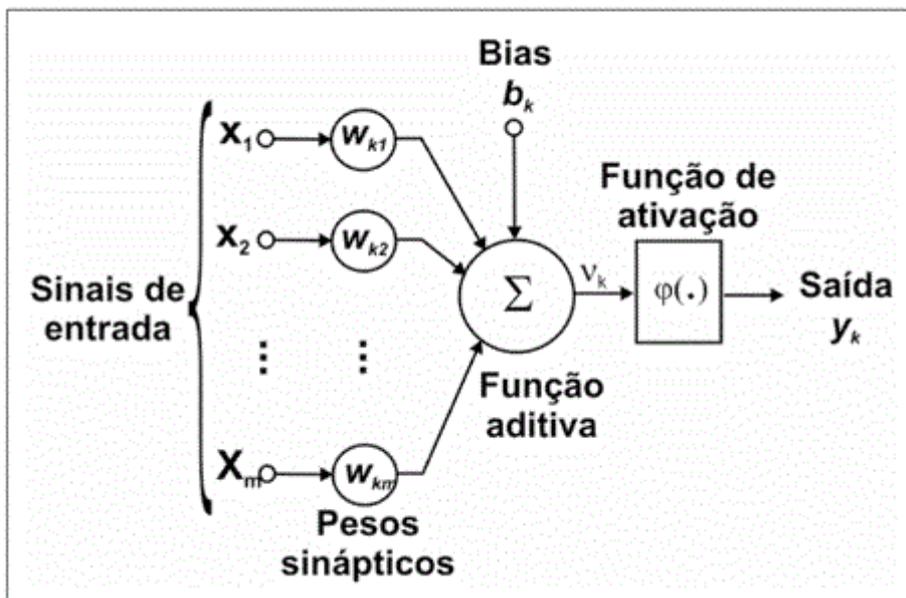


Figura 05 – Rede Perceptron (Extraído de Revista Escola de Minas 2005).

Em 1969, Minsky e Papert publicaram um artigo, no qual, provavam com formulas matemáticas que a rede perceptron simples, não era capaz de solucionar problemas não linearmente separáveis, oque desestimulou grandes pesquisadores na época, deixando o estudo de redes neurais esquecido durante um bom tempo. Poucos pesquisadores continuavam a trabalhar na área (Braga, Ludemir e Carvalho, 2000).

Nos anos 80, muitos dos pesquisadores foram bastante corajosos e passaram a publicar diversas propostas para a exploração de desenvolvimento de redes neurais bem como suas aplicações. Talvez o fato mais importante deste período tenha ocorrido quando Ira Skurnick, um administrador de programas da DARPA (*Defense*

*Advanced Research Projects Agency*) decidiu ouvir os argumentos da neuro computação e seus projetistas, e divergindo dos caminhos tradicionais dos conhecimentos convencionais, fundou em 1983 pesquisas em neuro computação. Este ato não só abriu as portas para a neuro computação, como também deu à DARPA o status de uma das líderes mundiais em se tratando de "moda" tecnológica. Outra "potência" que emergiu neste período foi John Hopfield, renomado físico de reputação mundial, se interessou pela neuro computação, e escreveu artigos que percorreram o mundo todo persuadindo centenas de cientistas, matemáticos, e tecnólogos altamente qualificados a se unirem esta nova área emergente. Apesar de um terço dos pesquisadores da área terem aderido à mesma pela influência de Hopfield, foi em 1986 que este campo de pesquisa "explodiu" com a publicação do livro "Parallel Distributed Processing" (Processamento Distribuído Paralelo) editado por David Rumelhart e James McClelland. Em 1987 ocorreu em São Francisco a primeira conferência de redes neurais em tempos modernos, a IEEE International Conference on Neural Networks, e também foi formada a International Neural Networks Society (INNS). A partir destes acontecimentos decorreu a fundação do INNS journal em 1989, seguido do Neural Computation e do IEEE Transactions on Neural Networks em 1990. Desde 1987, muitas universidades anunciaram a formação de institutos de pesquisa e programas de educação em neuro computação (Tatibana e Kaetsu, 2012).

### **2.2.3 – Perceptron Multicamadas**

De acordo com Tatibana e Kaetsu (2012) a rede neural artificial é um sistema de neurônios ligados por conexões sinápticas e dividido em neurônios de entrada, que recebem estímulos do meio externo, neurônios internos ou ocultos e neurônios de saída, que se comunicam com o exterior. A forma de arranjar perceptrons em camadas é denominado Perceptron de Multicamadas. O perceptron de multicamadas foi concebido para resolver problemas mais complexos, os quais não poderiam ser resolvidos pelo modelo de neurônio básico. Um único perceptron ou uma combinação das saídas de alguns perceptrons poderia realizar uma operação XOR, porém, seria incapaz de aprendê-la. Para isto são necessárias mais conexões,

os quais só existem em uma rede de perceptrons dispostos em camadas. Os neurônios internos são de suma importância na rede neural, pois se provou que sem estes se torna impossível a resolução de problemas linearmente não separáveis. Em outras palavras pode-se dizer que uma rede é composta por várias unidades de processamento, cujo funcionamento é bastante simples. Essas unidades, geralmente são conectadas por canais de comunicação que estão associados a determinado peso. As unidades fazem operações apenas sobre seus dados locais, que são entradas recebidas pelas suas conexões. O comportamento inteligente de uma Rede Neural Artificial vem das interações entre as unidades de processamento da rede.

A maioria dos modelos de redes neurais possui alguma regra de treinamento, onde os pesos de suas conexões são ajustados de acordo com os padrões apresentados. Em outras palavras, elas aprendem através de exemplos. Arquiteturas neurais são tipicamente organizadas em camadas, com unidades que podem estar conectadas às unidades da camada posterior (Tatibana e Kaetsu, 2012).

Tatibana e Kaetsu (2012) ressaltam que a rede neural passa por um processo de treinamento a partir dos casos reais conhecidos, adquirindo, a partir daí, a sistemática necessária para executar adequadamente o processo desejado dos dados fornecidos. Sendo assim, a rede neural é capaz de extrair regras básicas a partir de dados reais, diferindo da computação programada, onde é necessário um conjunto de regras rígidas pré-fixadas e algoritmos.

A Figura 6 demonstra a organização em camadas da rede MLP.

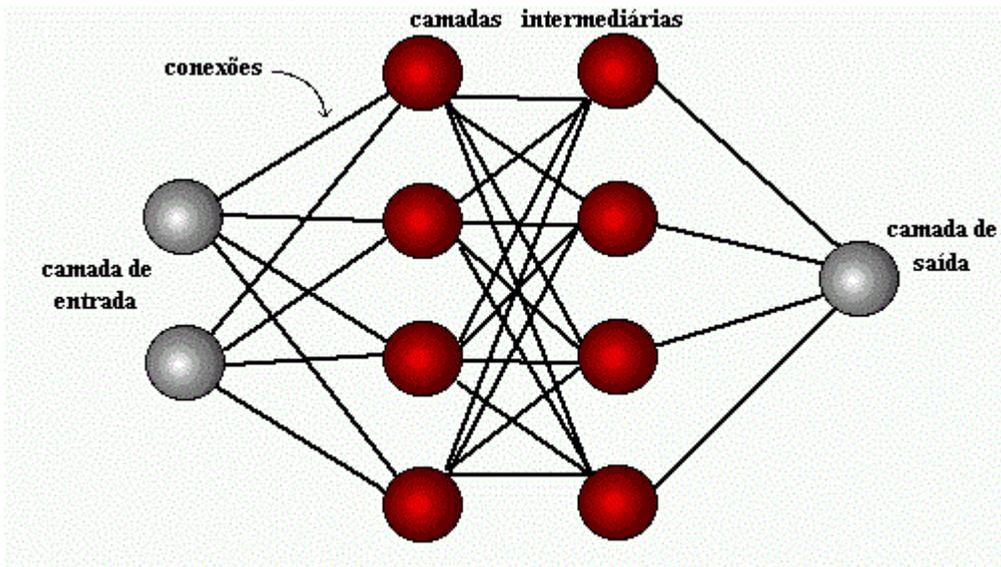


Figura 06 – Organização em camadas (extraído de Tatibana e Kaetsu, 2012).

Usualmente as camadas são classificadas em três grupos:

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

### 2.3 – Algoritmo “*Backpropagation*”

De acordo com Haykin (2001) o algoritmo de *backpropagation* foi proposto em 1986 por Rumelhart, Hinton e Williams. *Backpropagation* é um algoritmo para treinamento de redes MLP, baseado no Aprendizado Supervisionado por Correção de Erros.

Durante o treinamento com o algoritmo *backpropagation*, a rede opera em uma sequência de dois passos. Primeiro, um padrão é apresentado à camada de

entrada da rede. A atividade resultante flui através da rede, camada por camada, até que a resposta seja produzida pela camada de saída. No segundo passo, a saída obtida é comparada à saída desejada para esse padrão particular. Se esta não estiver correta, o erro é calculado. O erro é propagado a partir da camada de saída até a camada de entrada, e os pesos das conexões das unidades das camadas internas vão sendo modificados conforme o erro é retropropagado (USP – Universidade de São Paulo, 2012).

O algoritmo de *Backpropagation* é formado por dois passos:

**1° - Propagação:** Depois de apresentado o padrão de entrada, a resposta de uma unidade é propagada como entrada para as unidades na camada seguinte, até a camada de saída, onde é obtida a resposta da rede e o erro é calculado;

A Figura 7 mostra o caminho da propagação do algoritmo de *backpropagation*.

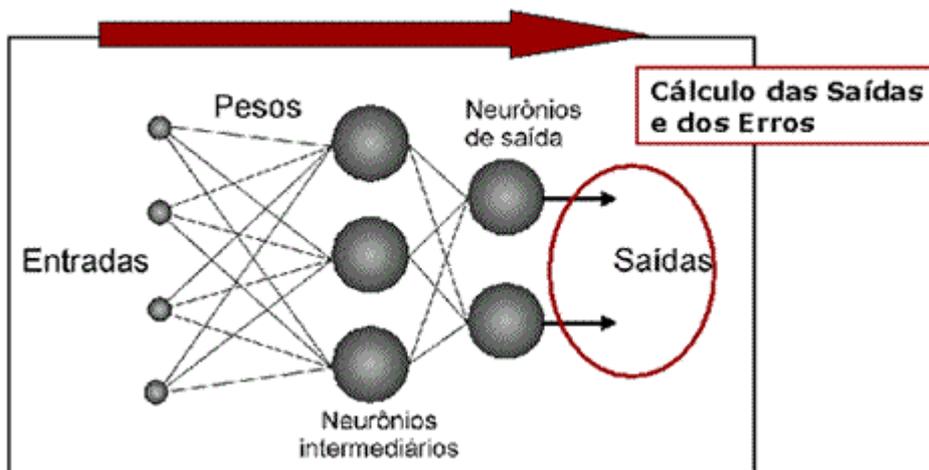


Figura 07 – Fase de Propagação (extraído de LNCC – Laboratório Nacional de Computação Científica, 2012).

**2° - Retropropagação ("*backpropagation*"):** Desde a camada de saída até a camada de entrada, são feitas alterações nos pesos sinápticos, conforme ilustrado na Figura 8.

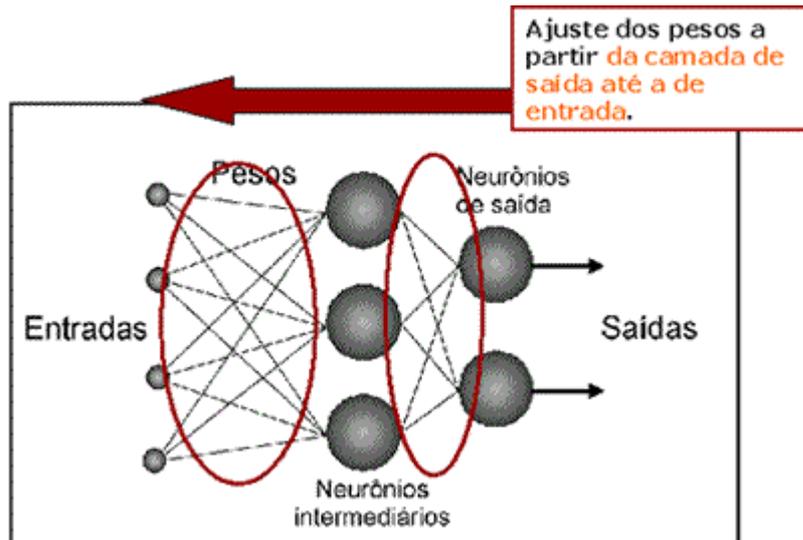


Figura 08 – Fase de Retropropagação (extraído de LNCC – Laboratório Nacional de Computação Científica, 2012).

Existe um problema com o algoritmo de *Backpropagation*, ele pode ficar muito tempo sendo executado, em treinamento. Vários fatores podem influenciar esta duração, porém o correto é sempre utilizar um critério de parada, geralmente utiliza-se um número máximo de ciclos ou também verificar se a rede apresenta uma boa capacidade de generalização. Porque se não ela pode causar o problema de *over-training*, ou seja a rede pode se especializar no conjunto de dados do treinamento e perde a capacidade de generalização.

### **3. DESENVOLVIMENTO DO PROJETO**

Este capítulo tem por objetivo apresentar a descrição e modelagem do programa para a classificação de espécies de seres vivos utilizando técnicas de inteligência artificial. Será apresentado, também, como desenvolver uma rede neural Perceptron para reconhecimento de padrões.

#### **3.1 – Descrição do Problema**

Neste projeto desenvolveu-se um programa para reconhecimento de N espécies em quatro tipos diferentes de reinos. Para este programa, foi utilizada a rede neural Perceptron simples, contendo três sinais de entrada e dois sinais de saída. As informações de entrada são codificadas em binário para melhor entendimento do algoritmo, após essa codificação foi feito todo o treinamento da rede utilizando o algoritmo de treinamento regra delta (Aprendizagem por correção de erros), aprendizado supervisionado, o qual é treinado com pares de conjuntos de entrada e de saída desejada. Quando é apresentada a rede um conjunto de entrada, esta retorna um conjunto de valores de saída, que é comparado ao conjunto de valores de saída desejado podendo assim identificar a entrada e gerar uma resposta igual à resposta esperada.

Para o desenvolvimento deste programa utilizou-se a linguagem JAVA, a IDE ECLIPSE e o banco de dados POSTGRESQL.

#### **3.2 – Modelagem do Problema**

Nesta seção será apresentada a modelagem do problema. O desenvolvimento do projeto foi dividido em 2 módulos para facilitar a implementação:

- Módulo 1: Criação do banco de dados;
- Módulo 2: Desenvolvimento do programa.

A Figura 09 mostra um esquema da modelagem do problema.

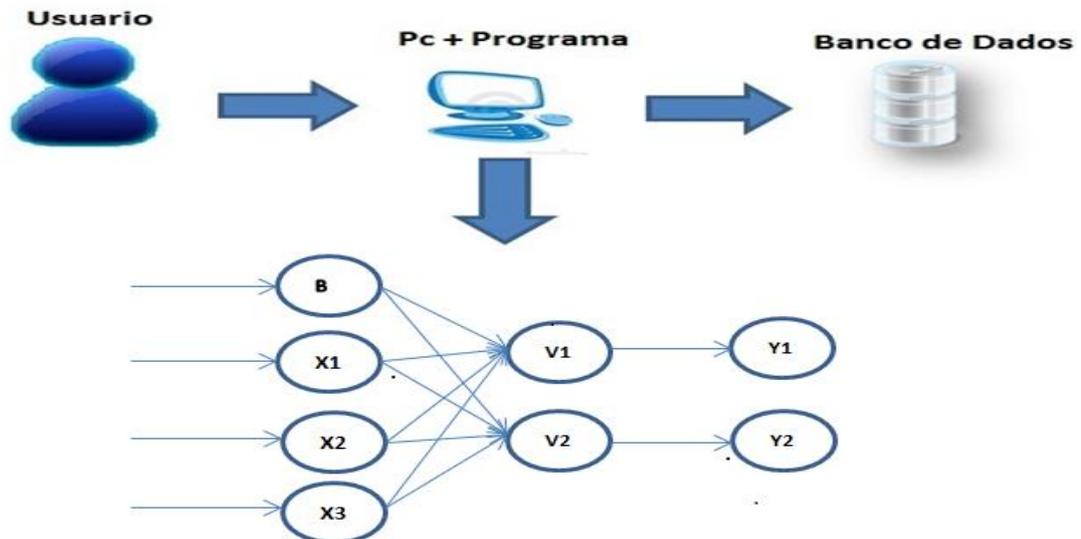


Figura 09 – Modelagem do Problema.

O usuário faz os cadastros de espécie e reino. O banco de dados armazena as entradas (x1,x2,x3) e saídas (y1,y2), em binário, de espécies e reinos. O algoritmo ainda adiciona um neurônio especial chamado de BIAS para aumentar os graus de liberdade, permitindo assim uma melhor adaptação por parte da rede neural e também utilizando este neurônio especial é possível apresentar uma saída não nula, ainda que todas suas entradas sejam nulas. Por fim, o programa, a partir das entradas produzidas pelo usuário, executa o treinamento da rede utilizando as funções de ativação e transferência (v1, v2) gerando as saídas (y1,y2).

### 3.3 – Especificação

Nesta seção, será feita a especificação do modelo de reconhecimento de padrões com o banco de dados relacional, e a construção dos diagramas UML (*Unified Modeling Language*), utilizando as ferramentas ArgoUML, DIA e DBDesigner.

Na especificação serão apresentados os diagramas de casos de uso, diagrama de classes, diagrama de sequência, diagrama de Entidade-Relacionamento e o diagrama de atividade.

### 3.3.1 – Diagrama de Casos de Uso

Diagramas de Casos de Uso representam um conjunto completo de ações realizadas por um sistema, e que originam um resultado a um ator.

Os casos de uso mostram o quê deve ser feito, mapeando as necessidades do cliente, admitindo mudanças nas regras e requisitos, apresentando visualmente itens que estão contidos dentro do escopo do software, formalizando o escopo a ser contratado e também facilitando em muito a comunicação entre a equipe e o cliente.

A Figura 10 mostra o diagrama de casos de uso que representa as funcionalidades executadas pelo sistema desenvolvido nesse trabalho..

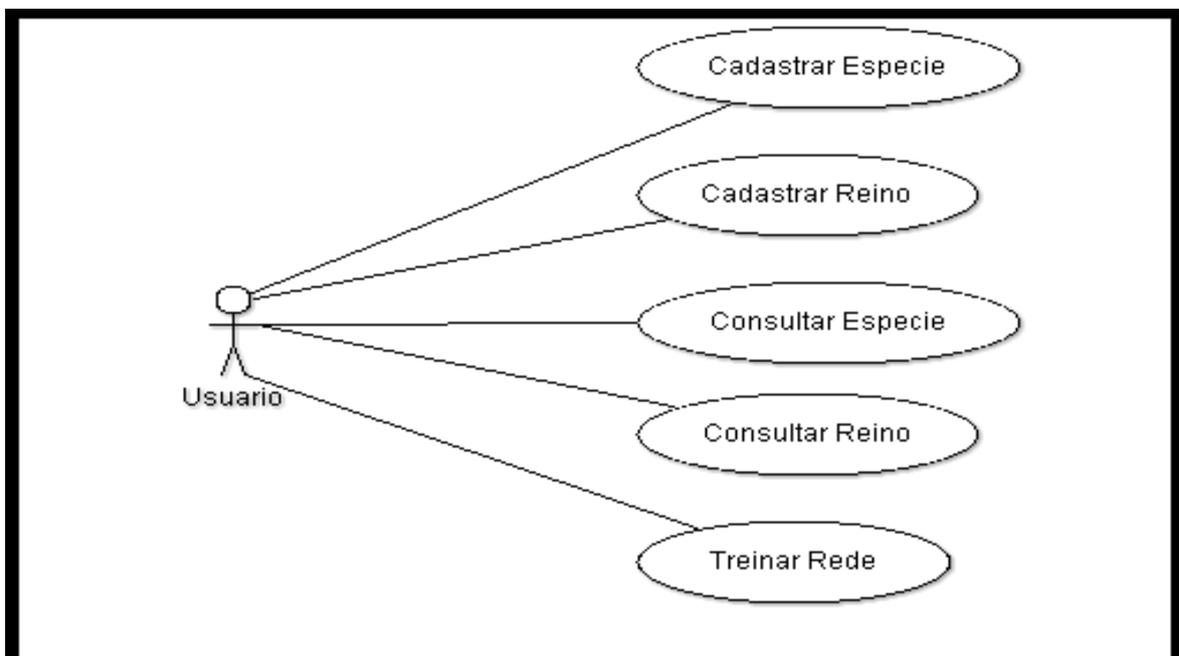


Figura 10 – Caso de Uso geral do sistema.

- **Cadastrar Espécie:** Encarregado de fazer a persistência dos dados obtidos do usuário para poder treinar a rede e comparar as saídas.
- **Cadastrar Reino:** Cadastramento necessário de espécies.
- **Consultar Espécie:** Caso de Uso responsável por pesquisar espécies, alterá-las ou excluí-las.
- **Consultar Reino:** Caso de Uso responsável por pesquisar por reinos, alterá-los ou excluí-los.
- **Treinar Rede:** Caso de Uso responsável por treinar a rede de acordo com a entrada obtida pelo usuário, utilizando-se do algoritmo perceptron simples para efetuar o treinamento.

### 3.3.2 – Diagrama de ER

O Diagrama de Entidade-Relacionamento (DER) é um modelo baseado na percepção do mundo real, que consiste em um conjunto de objetos básicos chamados entidades e nos relacionamentos entre esses objetos. O DER tem por objetivo facilitar o projeto de banco de dados, possibilitando a especificação da estrutura lógica geral do banco de dados, ou seja, é capaz de expressar graficamente a estrutura lógica geral de um banco de dados (Mello, 2012).

A Figura 11 mostra o Diagrama de ER.

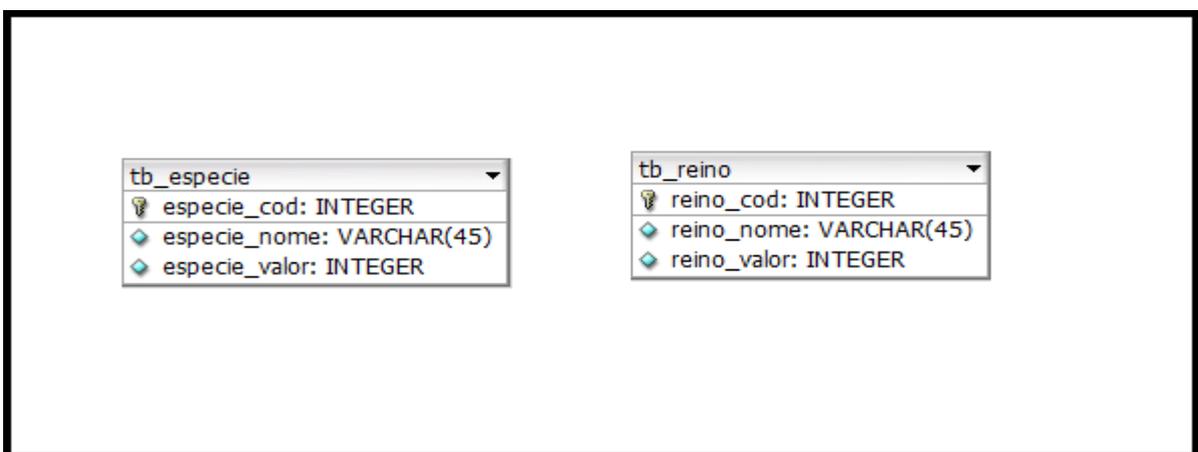


Figura 11 – Diagrama ER.

Este diagrama apresenta as tabelas armazenadas no banco da aplicação:

- **Tb\_especie:** Tabela responsável para o cadastramento e consultas de espécies.
- **Tb\_reino:** Tabela responsável para o cadastramento e consultas de reinos.

### 3.3.3 – Diagrama de Classe

Diagrama de classes é uma representação da estrutura e relações das classes que servem de modelo para objetos que são gerenciados pela aplicação (Lima, 2012).

A Figura 12 ilustra o Diagrama de Classes.

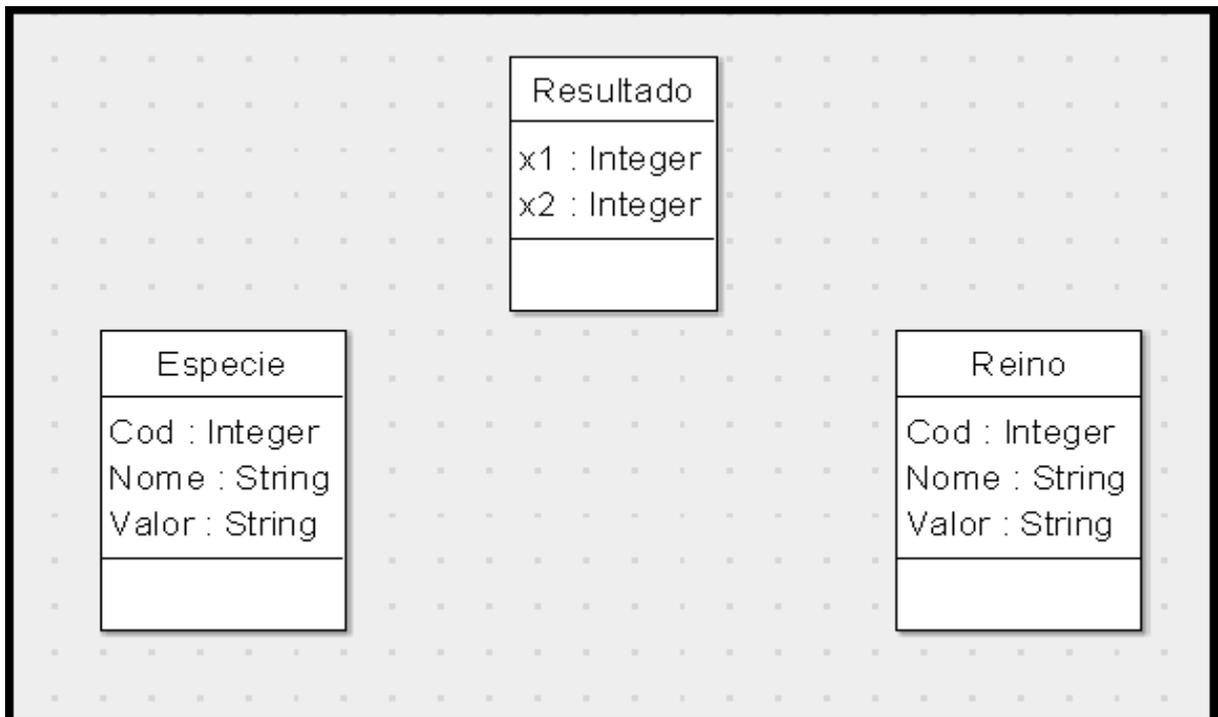


Figura 12 – Diagrama de Classes.

O diagrama de classes apresenta as três classes criadas para este projeto. As duas classes de cadastramento e a de resultado.

- **Classe Espécie:** Classe responsável para o cadastramento e consultas de espécies.
- **Classe Reino:** Classe responsável para o cadastramento e consultas de reinos.
- **Classe Resultado:** Classe responsável para armazenar os resultados e compará-los com as saídas esperadas.

### 3.3.4 – Diagrama de Sequência

Diagrama de Sequência descreve a maneira como os grupos de objetos colaboram em algum comportamento ao longo do tempo. Este diagrama, serve também para representar a sequência de algum processo (Lima, 2012).

A Figura 13 ilustra o Diagrama de Sequência do processo de treinamento.

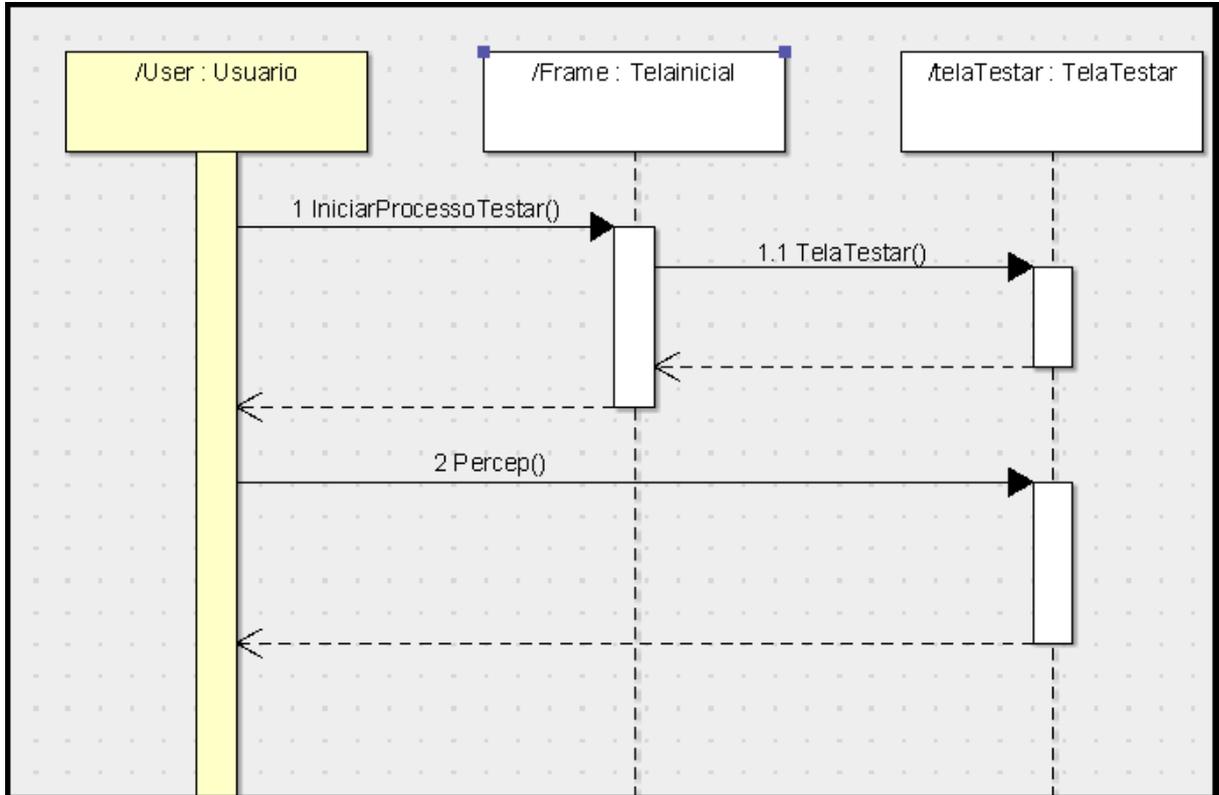


Figura 13 – Diagrama de Sequência.

Neste diagrama de sequência, o usuário solicita a tela de testes com o click do botão na tela principal, e em seguida a tela de testes é gerada. O usuário informa os dados de entrada e clica no botão treinar, no qual chama o método `percep()` que faz todo o treinamento da rede e gera o resultado esperado.

### **3.3.5 – Diagrama de Atividade**

Diagrama de atividades é utilizado para representar os fluxos conduzidos por processamentos, ou seja, o diagrama de atividades mostra o fluxo de atividades em um único processo, mostrando como uma atividade depende da outra.

A Figura 14 mostra o desenvolvimento do diagrama de atividade para o reconhecimento de padrões.

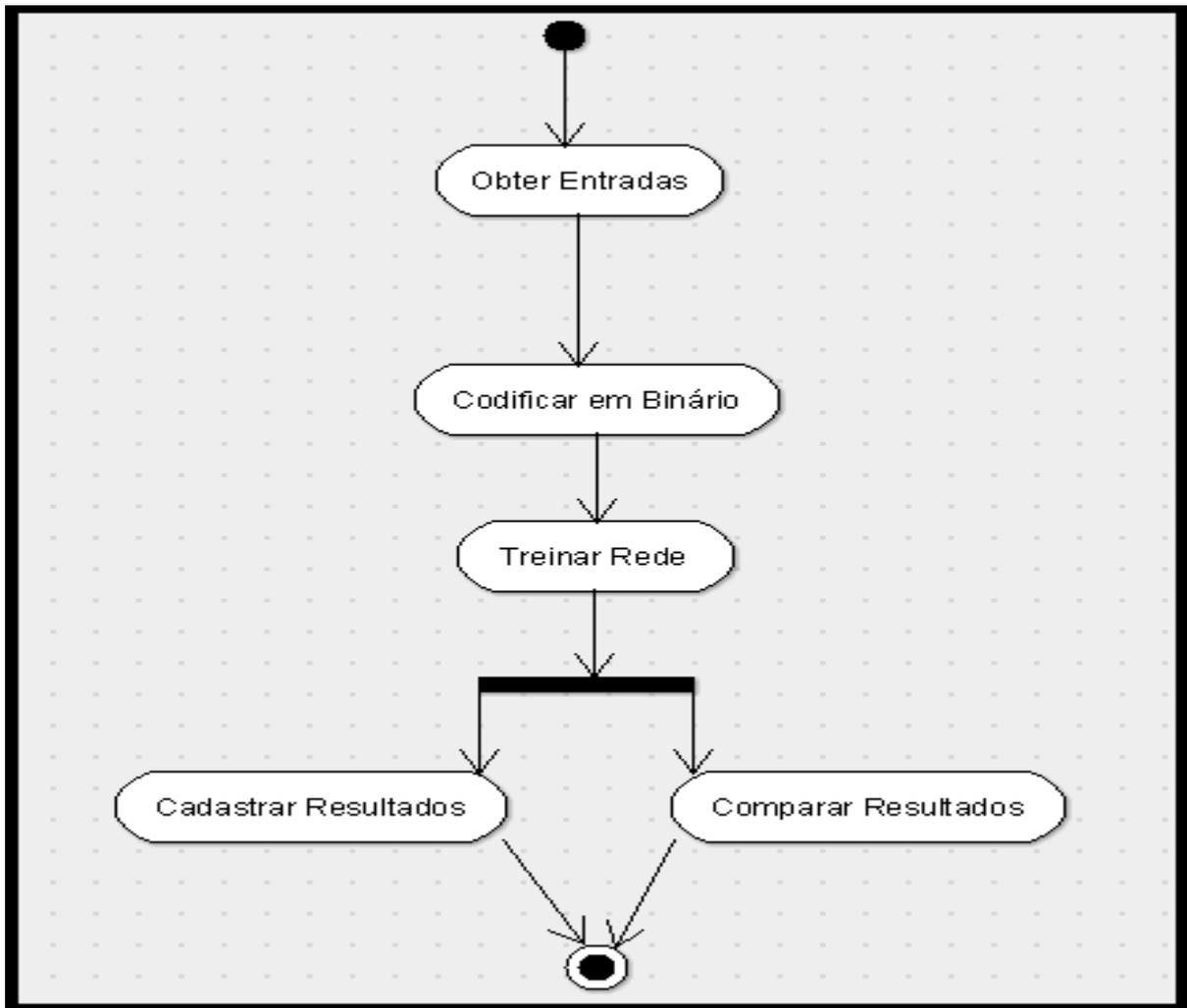


Figura 14 – Diagrama de Atividades.

O Diagrama de Atividades do processo de reconhecimento de padrões mostra a visualização de resultados obtidos, comparando-os com resultados esperados e armazenando os resultados obtidos.

### 3.4 - Implementações

Nesta seção serão apresentados, com detalhes, os módulos de programa implementados no projeto.

#### 3.4.1 – Criação do banco de dados

Neste módulo será feito a criação do banco de dados relacional para armazenar as informações, pesquisa de informações e atualizações das informações.

A Figura 15 mostra a arquitetura de um banco de dados relacional.

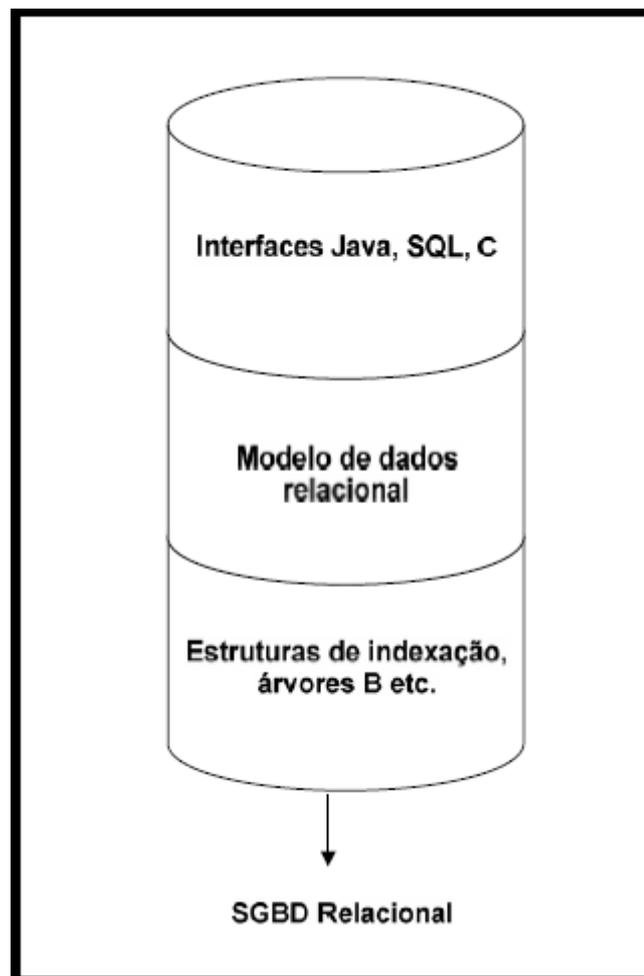


Figura 15 – Estrutura SGBD Relacional.

### 3. 4.1.1 – Preparação do Banco de Dados PostgreSQL.

Após a instalação do banco de dados PostgreSQL será necessário criar um novo banco para persistir suas informações e realizar suas transações. Este banco será criado pela ferramenta pgAdmin III, como mostrado na Figura 16.

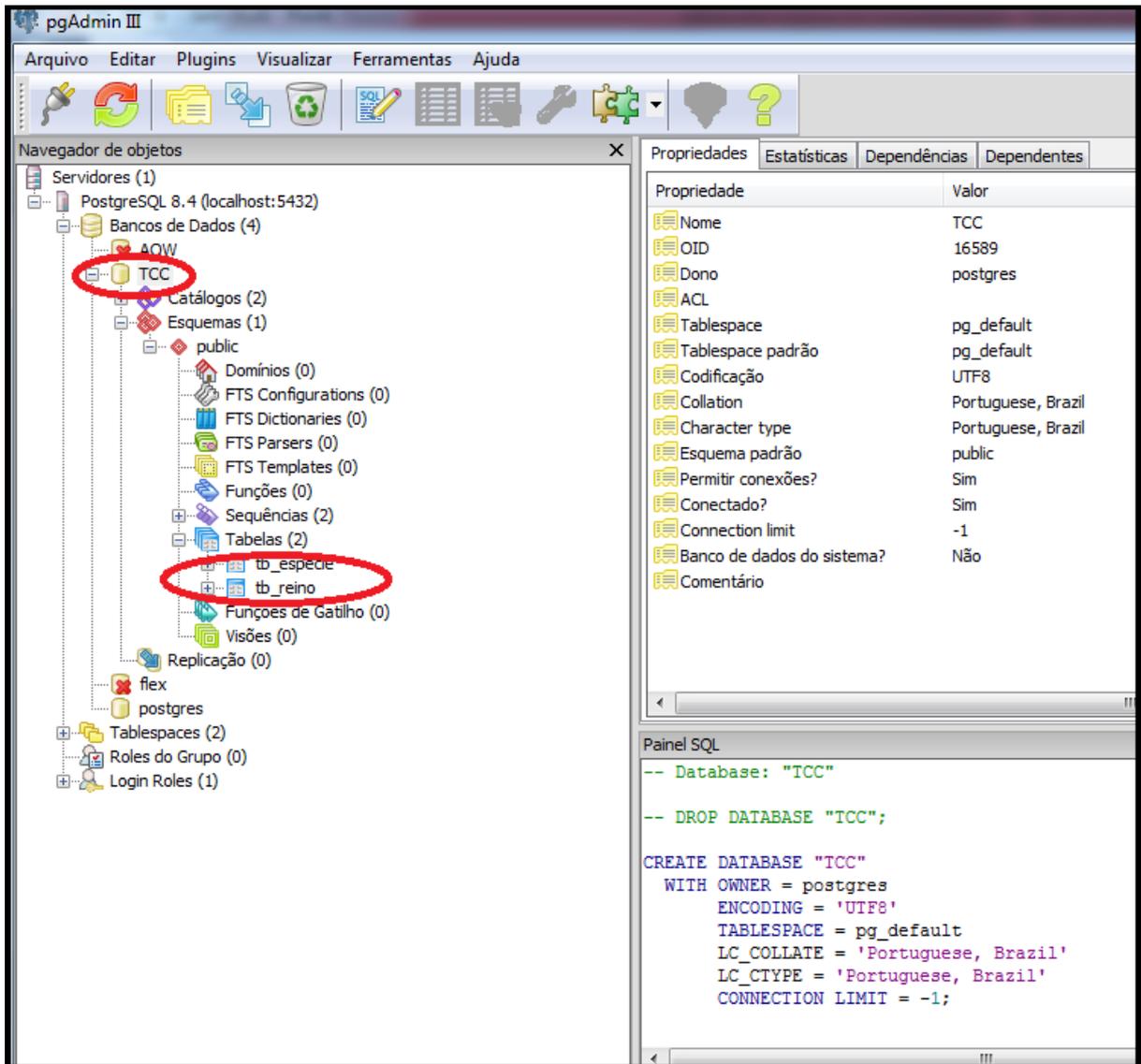


Figura 16 – Criação do banco de dados na ferramenta pgAdmin III.

Após a criação do banco e das tabelas, já é possível a interação do banco com o programa.

### 3. 4.2 – Desenvolvimento do programa

Nesta seção, será apresentado a implementação do programa para reconhecimento de padrões, com todas as informações necessárias para o processamento computacional.

#### 3. 4.2.1 – Visão Geral do Programa

A interface de inicialização do programa oferece três opções de menu e mais 5 opções de botão. As opções de ambos são cadastros, consultas e testes, conforme mostrado na Figura 17.

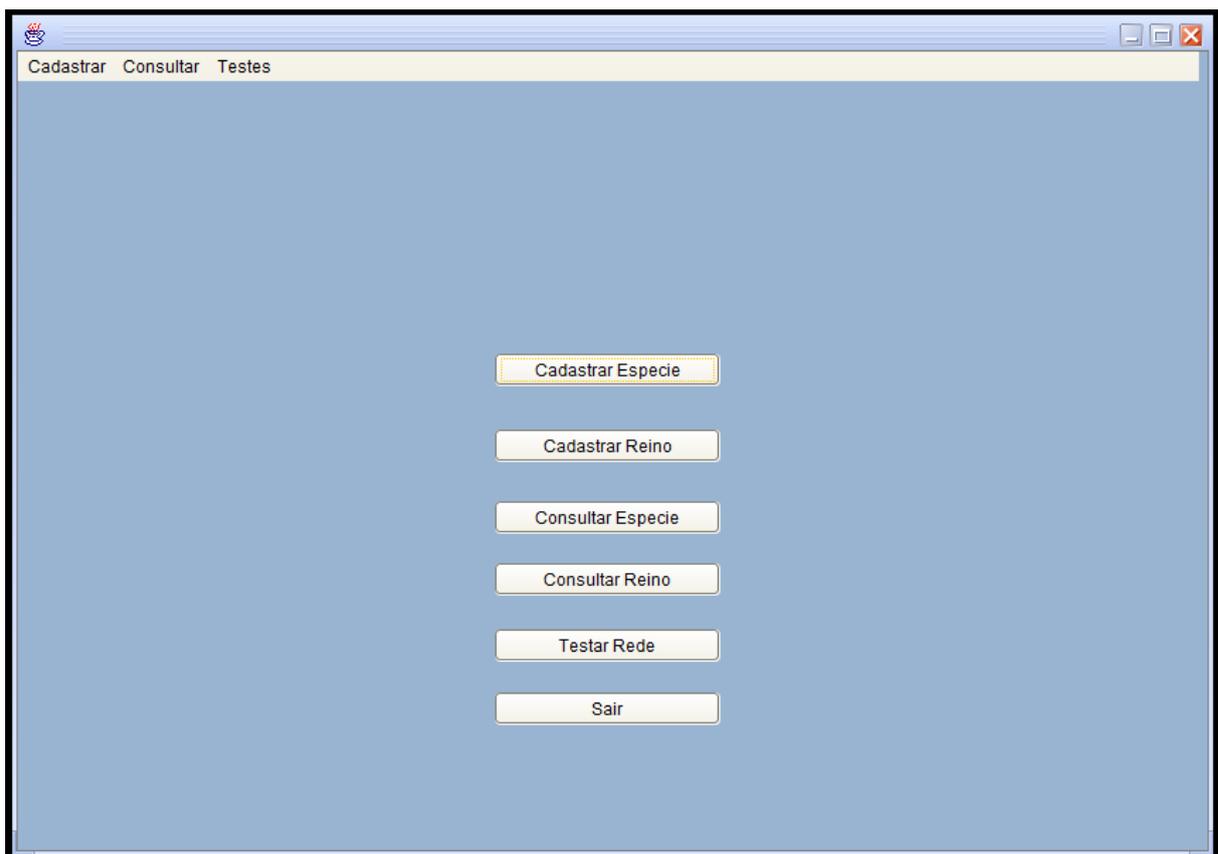


Figura 17 – Tela inicial do programa.

**Cadastrros:** nesta opção o usuário pode cadastrar espécies e reinos. Isso permite que a rede inicie o treinamento e gere as comparações dos resultados obtidos com as informações cadastradas.

**Consultas:** nesta opção o usuário pode consultar se os dados foram cadastrados da forma correta ou também promover atualizações.

**Testes:** é definida a opção de testes para analisar as entradas fornecidas pelo usuário, por meio das Redes Neurais do tipo Perceptron simples.

### 3. 4.2.2 – Cadastramento de Espécies e Reinos

Para o cadastramento de espécies ou reinos, o processo é semelhante para ambos. A seguir apresentam-se os passos necessários para o cadastramento de uma espécie. A Figura 18 mostra a interface de cadastramento de espécie.

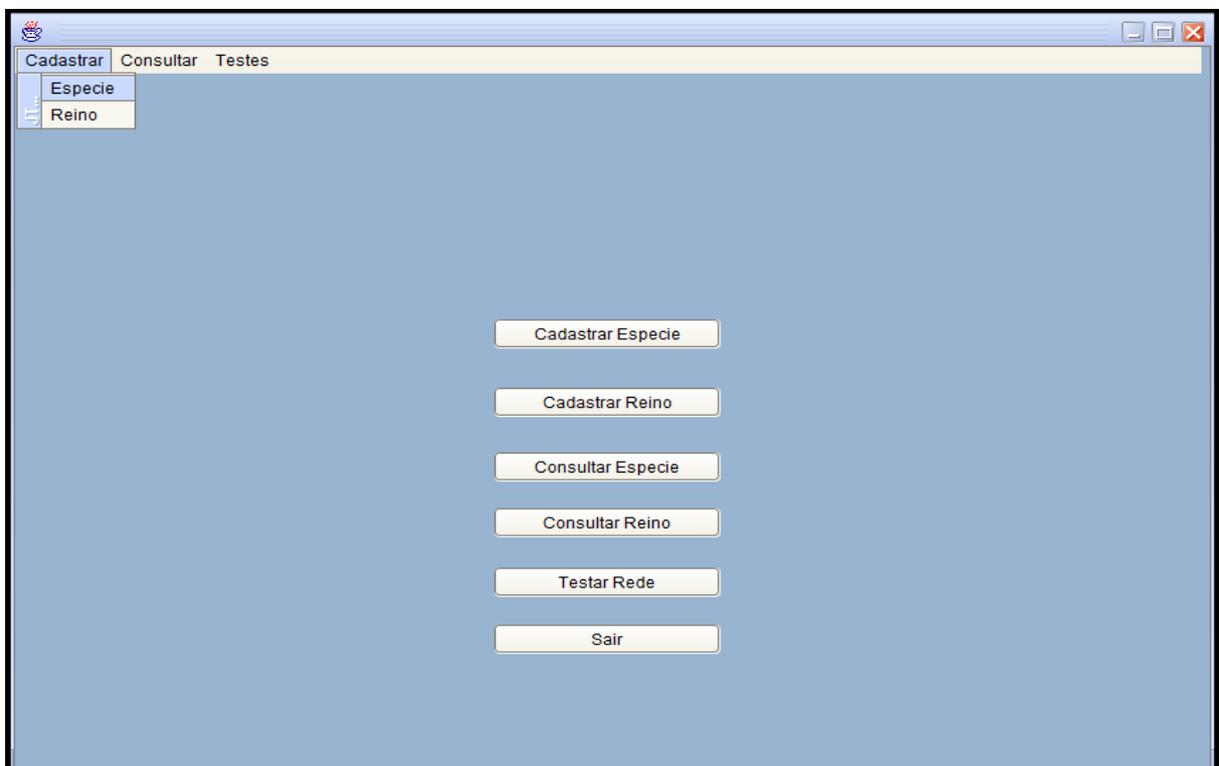
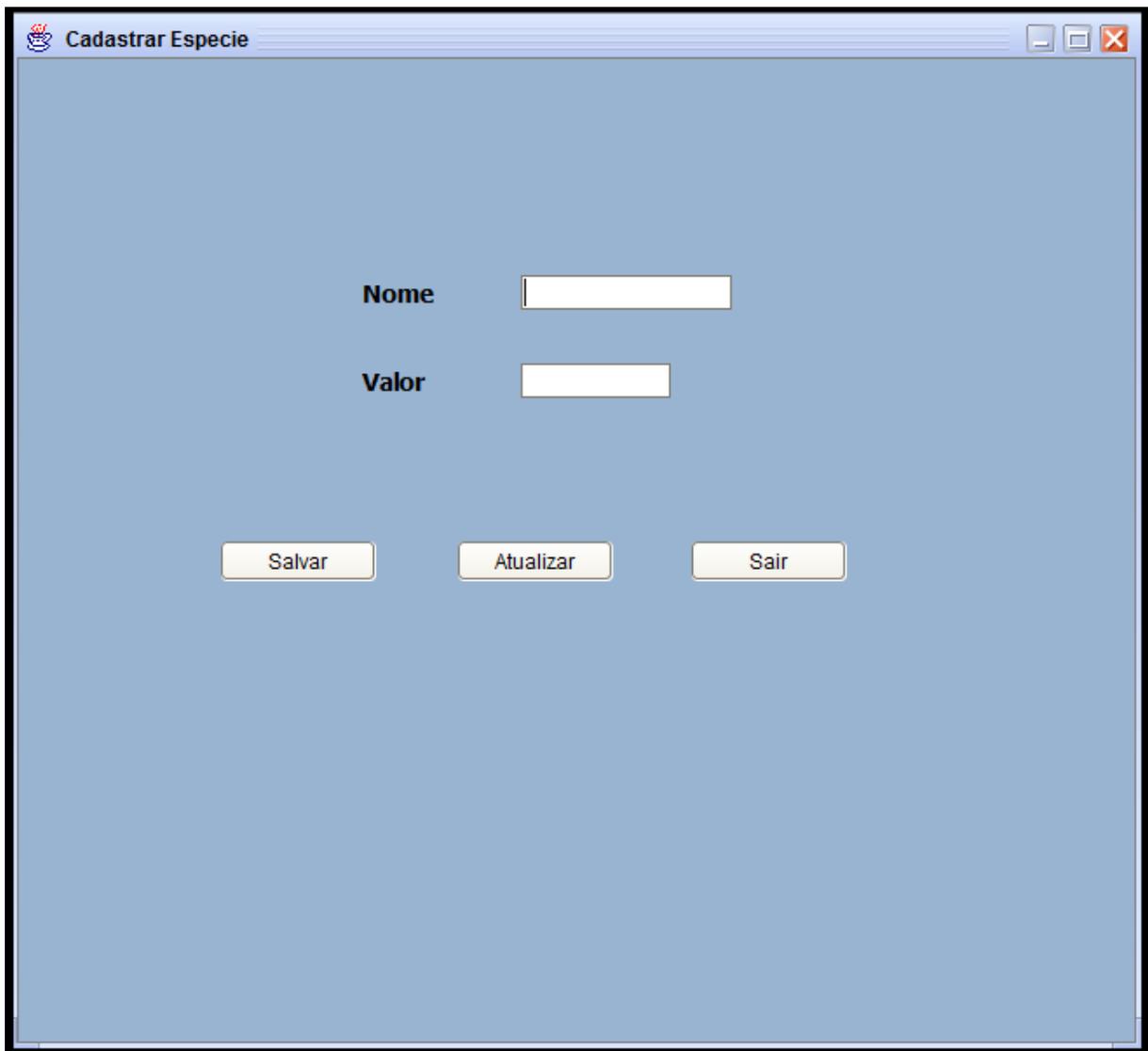


Figura 18 – Cadastramento Espécie.

Após a escolha no menu-cadastrar-espécie ou no botão cadastrar espécie, é criada a tela de cadastramento de espécie, no qual apresenta dois campos: nome da espécie e o campo valor em binário, no qual é o valor correspondente daquela espécie (x1, x2, x3).

A Figura 19 mostra a interface de cadastramento de espécie.



The image shows a software window titled "Cadastrar Espécie". The window has a blue background and a title bar with standard window controls (minimize, maximize, close). The main content area contains two labels, "Nome" and "Valor", each followed by a white rectangular input field. Below these fields are three buttons: "Salvar", "Atualizar", and "Sair", arranged horizontally.

Figura 19 – Tela de Cadastramento de Espécie.

Esses campos são de extrema importância, pois a rede compara os resultados obtidos com os resultados cadastrados. Nesta interface de cadastramento existe

também uma opção de atualizar um cadastro já existente no banco, podendo alterar seu nome ou valor.

### 3. 4.2.3 – Consultas de Espécies e Reinos

Para a consulta de espécies ou reinos, o processo é o mesmo, dessa forma descrevem-se os passos necessários para a consulta de uma espécie. A Figura 20 mostra a interface de consulta de espécie.

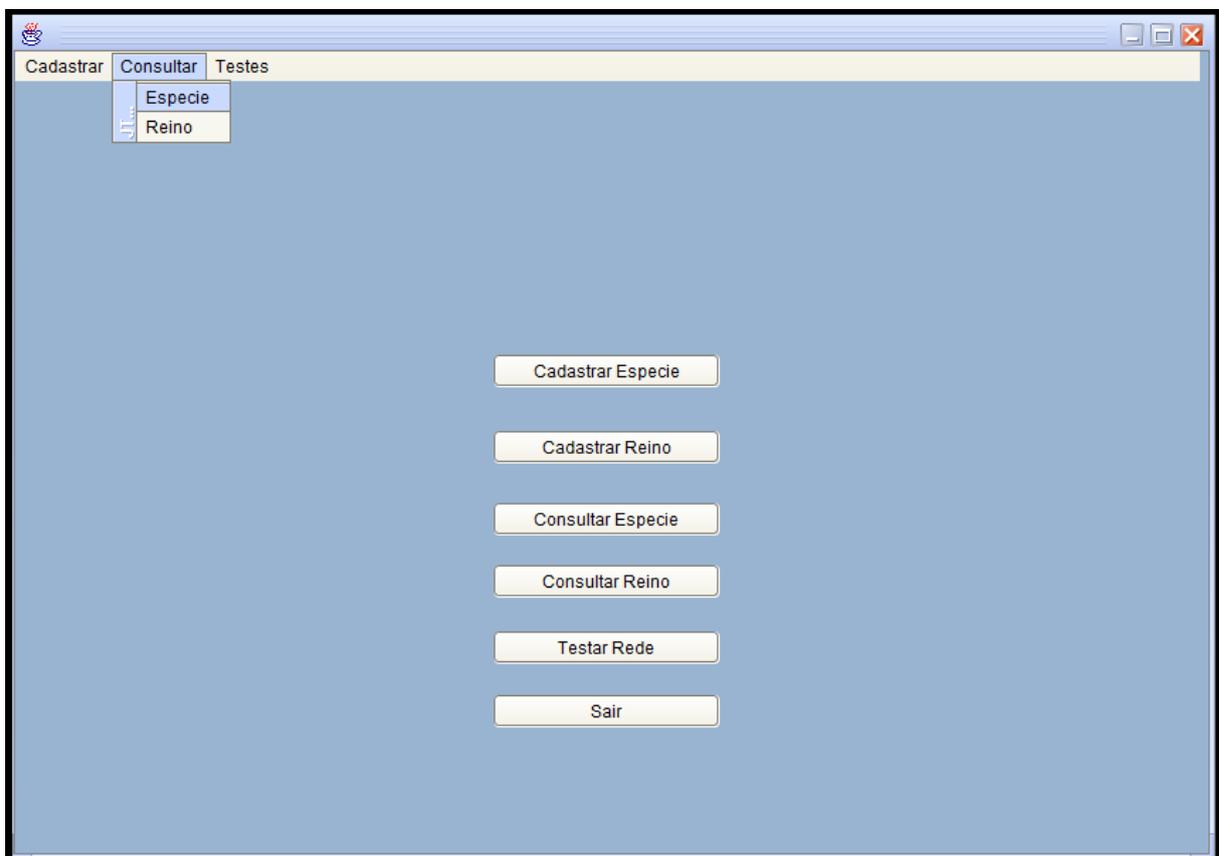


Figura 20 – Consulta Espécie.

Após a escolha no menu-consultar-especie ou no botão consultar especie, é criada a tela de consultar especie, no qual apresenta um campo, um botão e uma tabela, que contém as informações da pesquisa.

A Figura 21 mostra a interface de consulta de espécie.

**Pesquisar Espécie**

Pesquisar por :

Cod especie	Nome especie	Valor especie
2	homem	11
3	plantas	10
4	bacterias	00
5	fungos	01

Figura 21 – Tela de Consulta de Espécie.

Na interface de consulta existem também os botões: Excluir, no qual se exclui uma espécie selecionada; Alterar, que possibilita ao usuário alterar a espécie selecionada; e Sair, fecha a tela de consulta.

#### 3. 4.2.4 – Testes

Depois dos cadastramentos, e das consultas temos a fase de testes, ilustrada pela Figura 22.

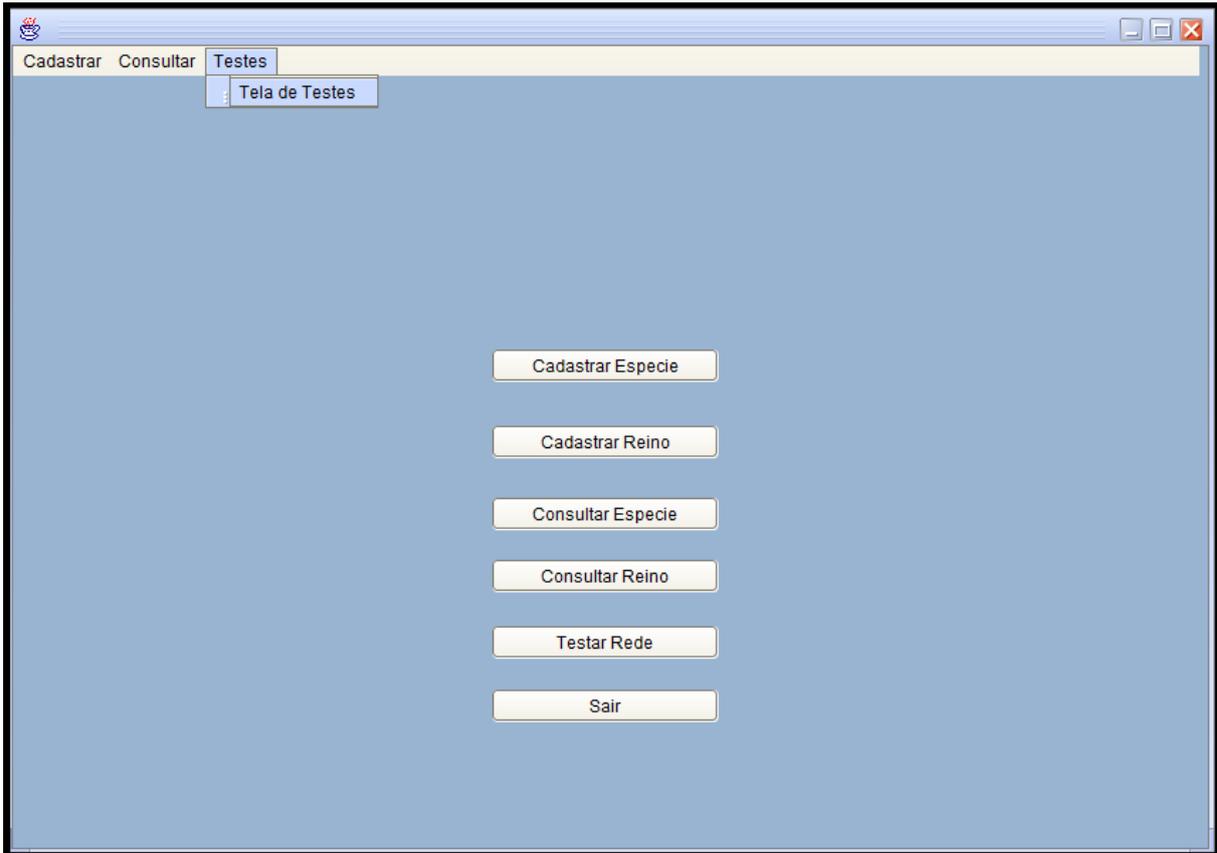


Figura 22 – Menu Testes.

Após a escolha no menu-testes ou no botão Testar Rede, é criada a tela de testes, na qual apresenta três campos a serem preenchidos pelo usuário ( $x_1$ ,  $x_2$ ,  $x_3$ ). Um botão treinar, que assume as entradas do usuário, passa para o método `percep()`, e o método `percep()` executa todo o treinamento da rede perceptron simples devolvendo, ao final, os resultados. Existem também as tabelas que são utilizadas para mostrar informações cadastradas e também os resultados obtidos.

A Figura 23 mostra a interface de testes.

Treinar Rede

x1

x2

x3

Resultados Obtidos:

Y1            Y2            ÉPOCA

Saida y1	Saida y2	Saida Epoca
0	0	0
1	1	1

Cod especie	Nome especie	Valor especie
2	homem	11 1
3	plantas	10
4	bacterias	00
5	fungos	01

Cod reino	Nome reino	Valor reino
2	animalia	11
3	plantae	10
4	monera	00
5	fungi	01

Figura 23 – Tela de Testes.

As tabelas Y1, Y2, ÉPOCA, mostram os resultados obtidos pela rede após cada ciclo de treinamento. Logo Y1, Y2 são as saídas obtidas e ÉPOCA os ciclos.

## 4. CONCLUSÃO

O presente trabalho de Conclusão de Curso realizou um estudo a partir da área de Inteligência Artificial, mais precisamente sobre as Redes Neurais Artificiais. O uso de Redes Neurais para reconhecimento de padrões vem sendo muito utilizado. Uma grande motivação para utilizá-las, se dá pelas suas características que buscam aproximar o sistema artificial ao sistema neural humano. Podem citar as seguintes características das RNA: capacidade de aprendizado e grande eficiência no processamento, o qual geram respostas mais rapidamente e um reconhecimento mais eficiente.

Para a escolha de um modelo de Rede Neural, deve ser feito um grande estudo, pois como demonstrado nesse trabalho, existem diferentes tipos de redes e diferentes tipos de algoritmos de treinamento.

Neste trabalho implementou-se uma rede neural artificial com o intuito de aplicar os conhecimentos obtidos durante sua condução. Construiu-se a rede neural perceptron simples com dois valores de entrada e dois valores de saída, para o reconhecimento de N indivíduos em classes de seres vivos. Para a correção dos pesos sinápticos e o nível de bias foi utilizado o algoritmo de treinamento regra delta, o que demonstrou ser um algoritmo muito rápido e eficiente para o problema.

Para trabalhos futuros novas implementações de reconhecimento de padrões mais complexas, como por exemplo de DNA, face poderiam ser criadas. Para isso, poderia ser utilizada um nível de camadas maior e um outro algoritmo de treinamento, o Algoritmo de BackPropagation, por exemplo.

## REFERÊNCIAS BIBLIOGRÁFICAS

BARRETO, Guilherme de Alencar. **Redes Neurais Artificiais: Uma Introdução Prática**. Depto. Engenharia de Teleinformática (DETI/UFC). Disponível em: <[http://www.deti.ufc.br/~guilherme/TI016/slides\\_PS\\_MLP.pdf](http://www.deti.ufc.br/~guilherme/TI016/slides_PS_MLP.pdf)>. Acesso em: 15 março 2012.

BRAGA, Antônio de Pádua; LUDEMIR, Teresa Bernarda; CARVALHO, André Carlos Ponce de Leon Ferreira. **Redes Neurais Artificiais: Teoria e Aplicações**. 1. Ed. Rio de Janeiro: Editora LTC – Livros Técnicos e Científicos S.A, 2000.

FERRARI, Fabricio. **Introdução à Inteligência Artificial**. Disponível em: <<http://www.ferrari.pro.br/home/documents/FFerrari-Introducao-IA.pdf>>. Acesso em: 03 maio 2012.

HAYKIN, Simon. **Redes Neurais: Princípios e Prática**. 2. Ed. Tradução de Paulo Martins Engel. São Paulo: Editora Artmed LTDA, 1999.

LIMA, Cynthia Moreira; Labidi, Sofiane. **Introdução à Inteligência Artificial**. Disponível em: < [http://cynthia\\_m\\_lima.sites.uol.com.br/ia.html](http://cynthia_m_lima.sites.uol.com.br/ia.html)>. Acesso em: 03 maio 2012.

LIMA, Fernando. **UML-Diagrama de Classes**. Disponível em: < [http://200.230.71.24/moodle/file.php/120/Engenharia\\_03\\_DiagramaClasses.pdf](http://200.230.71.24/moodle/file.php/120/Engenharia_03_DiagramaClasses.pdf)>. Acesso em: 13 junho 2012.

LIMA, Fernando. **UML-Diagrama de Sequência**. Disponível em: < [http://200.230.71.24/moodle/file.php/120/Engenharia\\_05\\_DiagramaSequencia.pdf](http://200.230.71.24/moodle/file.php/120/Engenharia_05_DiagramaSequencia.pdf)>. Acesso em: 13 junho 2012.

LNCC – Laboratório Nacional de Computação Científica. **Tutorial Redes Neurais**. Disponível em: < [http://www.lncc.br/~labinfo/tutorialRN/frm1\\_introducao.htm](http://www.lncc.br/~labinfo/tutorialRN/frm1_introducao.htm)>. Acesso em: 03 maio 2012.

LUDWIG, Oswaldo Jr; MONTGOMERY, Eduard. **Redes Neurais: Fundamentos e Aplicações em C**. 1. Ed. Rio de Janeiro: Editora Ciência Moderna LTDA, 2007.

MELLO, Mauricio. **Banco de Dados**. Disponível em: <<http://www.las.pucpr.br/mcfmello/BD/BD-Aula02-MER.pdf>>. Acesso em: 13 junho 2012.

UEM – Universidade Estadual de Maringá. **Redes Neurais**. Disponível em: <<http://www.din.uem.br/ia/intelige/neurais2/historico.html>>. Acesso em: 30 abril 2012.

UEM – Universidade Estadual de Maringá. **Teste de Turing**. Disponível em: <<http://www.din.uem.br/~ia/maquinas/turing.htm>>. Acesso em: 03 maio 2012.

USP – Universidade de São Paulo. **Redes Neurais Artificiais**. Disponível em: <<http://www.icmc.usp.br/~andre/research/neural/index.htm#intro>>. Acesso em: 15 março 2012.

TATIBANA, Cassia Yuri; KAETSU, Deisi Yuki. **Redes Neurais**. Disponível em: <<http://www.din.uem.br/ia/neurais/>>. Acesso em: 30 abril 2012.