



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

CARLOS ROBERTO ROSSINI JUNIOR

**UTILIZAÇÃO DA TECNOLOGIA ADAPTATIVA PARA O
DESENVOLVIMENTO DE UMA REDE NEURAL ARTIFICIAL**

**Assis/SP
2018**



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

CARLOS ROBERTO ROSSINI JUNIOR

**UTILIZAÇÃO DA TECNOLOGIA ADAPTATIVA PARA O
DESENVOLVIMENTO DE UMA REDE NEURAL ARTIFICIAL**

Projeto de pesquisa apresentado ao curso de Ciência da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito para à obtenção do Certificado de Conclusão.

Orientando(a): Carlos Roberto Rossini Junior

Orientador(a): Dr. Almir Rogério Camolesi

**Assis/SP
2018**

FICHA CATALOGRÁFICA

ROSSINI, C. R. J.

UTILIZAÇÃO DA TECNOLOGIA ADAPTATIVA PARA O DESENVOLVIMENTO DE UMA REDE NEURAL ARTIFICIAL / Carlos Roberto Rossini Junior. Fundação Educacional do Município de Assis –FEMA – Assis, 2018.

Número de páginas. 40

Orientador: Dr. Almir Rogério Camolesi

1. Rede Neural Artificial. 2. Tecnologia Adaptativa. 3. Inteligência Artificial

CDD: 006.32
Biblioteca da FEMA

UTILIZAÇÃO DA TECNOLOGIA ADAPTATIVA PARA O DESENVOLVIMENTO DE UMA REDE NEURAL ARTIFICIAL

CARLOS ROBERTO ROSSINI JUNIOR

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador: _____
Dr. Almir Rogério Camolesi

Examinador: _____
Esp. Domingos de Carvalho Villela Junior

DEDICATÓRIA

Dedico este trabalho a Deus que me deu forças para realização deste trabalho. Aos meus pais Carlos e Regina que sempre me apoiaram, a minha irmã Annelize pelos conhecimentos transmitidos e a todos meus amigos que me ajudaram nos momentos difíceis.

AGRADECIMENTOS

Agradeço primeiramente ao meu orientador, professor e amigo Almir Rogério Camolesi, que me transmitiu o conhecimento e apoio necessário para a elaboração deste projeto, além de sua amizade e companheirismo.

Agradeço a minha família, que me apoiaram nos momentos mais importantes que trilhei nesta jornada, em especialmente a minha irmã Annelize, que estava ao meu lado o tempo todo e aos meus pais Carlos Roberto e Maria Regina que tiveram comigo nos momentos mais críticos.

Agradeço a todos os meus amigos que me proporcionaram um pouco de diversão nos momentos mais cansativos.

“Tudo posso naquele que me fortalece.”

(Filipenses 4:13)

RESUMO

Na busca de ferramentas computacionais capazes de simular o cérebro humano, foi criada a rede neural artificial, que possui a capacidade de realizar as tarefas de aprender, errar e de realizar novas descobertas.

Desta forma, muitos trabalhos foram realizados nesta área de conhecimento, de maneira que surgissem diversos novos modelos de redes neurais artificiais.

Este trabalho irá utilizar os conceitos da tecnologia adaptativa para a construção de uma rede neural artificial de múltiplas camadas, na tentativa de verificar as possíveis vantagens e desvantagens e na representação de algoritmos capazes de executar as ações adaptativas em uma rede neural artificiais de múltiplas camadas.

Através do estudo realizado, será desenvolvido um simulador de rede neural artificial de múltiplas camadas adaptativas, utilizando a linguagem de programação Lua, que possuirá como objetivo representar o objeto de estudo do trabalho e como uma ferramenta de ensino para a tecnologia adaptativa e de redes neurais artificiais de múltiplas camadas.

Palavras-chave: Inteligência Artificial, Redes Neurais Artificiais, Tecnologia Adaptativa

ABSTRACT

In search of computer tools able to simulate the human brain, was create the artificial neural network, that can realize the taks of learning, failing and discovering new things.

Therefore, a lot of works were done in this area of knowledge, in such way that emerged new models of artificial neural network.

This work, is going to use the adaptive technology concepts for a build of a Multi Layer Perceptron, in attempt to verify the possible advantages and disadvantages and the representation of algorithms capable of performing the adaptive actions in a neural network.

Through of the study realized, will be developed an adaptive multilayer artificial neural network simulator using the Lua programming language, which will aim to represent the object of work study and as a teaching tool for adaptive technology and artificial neural networks of multiple layers.

Keywords: Adaptive technology, Artificial Neural Network, Artificial Intelligence.

LISTA DE ILUSTRAÇÕES

Figura 1: Rede Neural Artificial	11
Figura 2: Neurônio Biológico.....	18
Figura 3: Neurônio Artificial.....	19
Figura 4: Representação do Neurônio Artificial	20
Figura 5: Representação MLP	23
Figura 6: Autômato Finito Adaptativo.....	25
Figura 7: Dominó Adaptativo	27
Figura 8: Percentual de Reconhecimento.....	29
Figura 9: Formato Inicial da Rede Neural	33
Figura 10: Adição de Camada de Neurônio.....	34
Figura 11: Remoção de Camada de Neurônio	35

LISTA DE ABREVIATURAS E SIGLAS

CNN	CONVOLUTIONAL NEURAL NETWORK
MCP	REDE NEURAL ARTIFICIAL
MLP	MULTILAYER PERCEPTRON
RNA	REDE NEURAL ARTIFICIAL

SUMÁRIO

1. INTRODUÇÃO	11
1.1. OBJETIVOS	13
1.2. JUSTIFICATIVA	13
1.3. MOTIVAÇÃO.....	13
1.4. PERSPECTIVA DE CONTRIBUIÇÃO.....	13
1.5. METODOLOGIA DE PESQUISA	14
2. REDES NEURAS ARTIFICIAIS	16
2.1. INTELIGÊNCIA ARTIFICIAL	16
2.2. APRENDIZADO DE MÁQUINA.....	17
2.3. REDES NEURAS	17
2.4. NEURÔNIO ARTIFICIAL.....	19
2.5. PERCEPTRON	20
2.5.1. APRENDIZADO DO PERCEPTRON.....	21
2.6. PERCEPTRON DE MÚTIPLAS CAMADAS	22
2.6.1. Rede Neural Convolutacional.....	24
3. TECNOLOGIA ADAPTATIVA	25
4. ESTUDO DE CASO	28
4.1. REDE NEURAL ARTIFICIAL DE MULTIPLAS CAMADAS ADAPTATIVA 28	
4.1.1. FUNÇÕES ADAPTATIVAS	29
4.2. IMPLEMENTAÇÃO DO SIMULADOR	33
5. CONCLUSÃO E TRABALHOS FUTUROS	36
5.1. TRABALHOS FUTUROS	37
6. REFERÊNCIAS	38

1. INTRODUÇÃO

Um problema matemático complexo pode ser rapidamente solucionado através de um algoritmo desenvolvido em um computador moderno, enquanto um humano, para conseguir solucionar este mesmo problema, pode possuir uma extrema dificuldade e demora. Em contrapartida, um humano possui uma facilidade muito maior em realizar uma identificação de um objeto em uma imagem, algo que um computador moderno possui uma extrema dificuldade de a realizar.

A inteligência artificial é um dos ramos mais estudados dentro da ciência da computação na atualidade, este fator está relacionado aos grandes números de pesquisas realizadas sobre o tema e o seu forte envolvimento dentro de grandes empresas. Em grande parte, as pesquisas relacionadas a inteligência artificial buscam utilizar os algoritmos genéticos ou as redes neurais artificiais como objeto de estudo.

Barretos (2002), demonstra a rede neural artificial como uma técnica para solucionar problemas ligados a Inteligência Artificial, pois elas possuem características que se assemelham com a inteligência humana, sendo elas a capacidade de aprender, errar e de realizar novas descobertas.

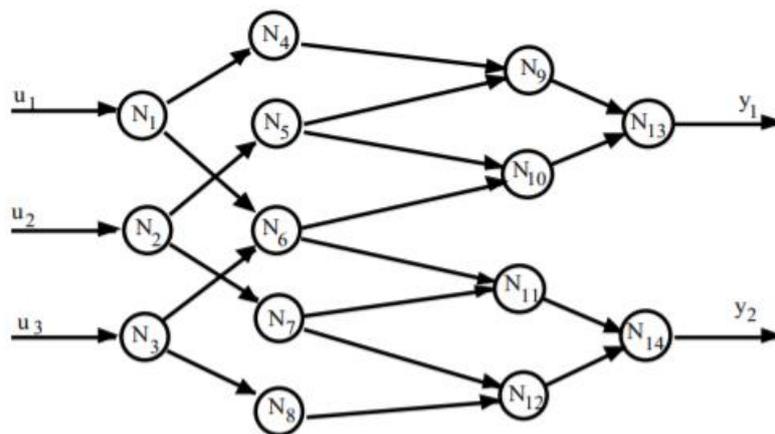


Figura 1: Rede Neural Artificial
Fonte: Barretos, 2002.

É possível considerar uma rede neural artificial como um sistema paralelo e distribuído, composto por diversas unidades de processamento simples, as quais irão realizar cálculos de funções matemáticas (BRAGA, LUDEMIR, PONCE, 2000). Também é demonstrado a rede neural artificial como uma ótima ferramenta para o desenvolvimento de aplicações

complexas, principalmente pela sua característica de conseguir realizar o aprendizado de acordo com o seu ambiente, desta forma, é considerada uma ótima ferramenta para o desenvolvimento de aplicações complexas.

Rowley (1998) apresentou um trabalho em que se utiliza das redes neurais artificiais para a produção de um algoritmo capaz de realizar reconhecimento facial, tendo como objetivo demonstrar a eficiência das redes neurais artificiais para problemas relacionados a visão computacional. Em seu trabalho, é possível perceber a eficiência da utilização das redes neurais para o desenvolvimento de uma solução de reconhecimento facial, visto que são demonstrados diversos casos em que a face a ser analisada está em diversos ângulos, posições e submetida a diferentes tipos de iluminação, e o algoritmo de reconhecimento, que utiliza de uma rede neural artificial em sua montagem, consegue realizar o reconhecimento da face em que ele foi previamente treinado a reconhecer.

Basu et al. (2012) realizou um estudo que demonstrou a precisão de reconhecimento de imagens de acordo com a quantidade de camadas de neurônios de uma rede neural artificial de múltiplas camadas. Neste trabalho, a rede neural artificial implementada tinha o objetivo de reconhecer numerais escritos à mão, porém esta *implementação demonstrou que até uma certa quantidade de camadas de neurônio, a precisão de reconhecimento do numeral aumentava, porém ao passar de um limite, de acordo com a adição de camadas iria diminuindo o percentual de precisão.*

Dutra e Matucci (2008) realizam um trabalho que demonstra a utilização de uma rede neural artificial em conjunto de uma árvore de decisão adaptativa, sendo os resultados positivos a utilização de uma rede neural artificial em conjunto da tecnologia adaptativa.

Neto (1993) apresenta a tecnologia adaptativa como uma técnica que pode ser utilizada para a modelagem de aplicações com comportamento modificável. A tecnologia adaptativa é composta por um dispositivo subjacente (dispositivo dirigido por regras) e um mecanismo adaptativo, o qual irá possibilitar a modificação do dispositivo subjacente.

Em um trabalho realizado por Tchemra (2009), é desenvolvido uma tabela de decisão adaptativa, a qual foi utilizada de uma tabela de decisão convencional como dispositivo subjacente em conjunto de funções adaptativas, o que permitiu que esta tabela de decisão se auto modificar em tempo de execução do código.

1.1. OBJETIVOS

Este trabalho tem como principal objetivo apresentar um estudo da utilização da tecnologia adaptativa na construção de uma rede neural artificial.

Para demonstrar as vantagens da utilização da tecnologia adaptativa para o desenvolvimento de uma rede neural artificial, foi desenvolvido um simulador que demonstra graficamente uma rede neural artificial e o seu comportamento após a execução de funções adaptativas.

1.2. JUSTIFICATIVA

Conforme Basu et al. (2012) demonstra, a arquitetura de uma rede neural artificial irá interferir diretamente em seu desempenho, tanto em velocidade, de acordo com a quantidade de camadas que a rede neural irá possuir, como também da sua taxa de precisão de acerto.

Este trabalho demonstra a importância dos estudos da tecnologia adaptativa, mostrando os cenários possíveis de realizar a sua implementação e dos benefícios de a utilizar.

1.3. MOTIVAÇÃO

A principal motivação deste trabalho é de contribuir com uma ferramenta que pode auxiliar nos futuros estudos relacionados a tecnologia adaptativa e demonstrar os pontos benéficos de sua utilização em conjunto das redes neurais artificiais.

Atualmente, a inteligência artificial é responsável por resolver diversos problemas computacionais. Esta pesquisa pretende realizar uma contribuição neste meio, desde realizando um estudo da arte, como também o desenvolvimento de uma ferramenta que utiliza destes conceitos.

1.4. PERSPECTIVA DE CONTRIBUIÇÃO

Pretende-se que esta pesquisa possa proporcionar um material de estudo e de referência para futuras pesquisas relacionadas a utilização da tecnologia adaptativa em conjunto da rede neural artificial.

Esta pesquisa pretende realizar um desenvolvimento de um simulador de uma rede neural artificial capaz de utilizar de funções adaptativa, o qual pode ser utilizado como um material didático para o ensino desde as redes neurais artificiais como também do ensino da tecnologia adaptativa.

1.5. METODOLOGIA DE PESQUISA

Neste trabalho foi empregado a metodologia de pesquisa experimental. Inicialmente, foi realizado um estudo detalhado referente aos objetos de estudo, sendo ele, a utilização da tecnologia adaptativa para o desenvolvimento de uma rede neural perceptron de multicamadas.

Como método de pesquisa relacionado as redes neurais artificiais, foi iniciada a pesquisa dos primeiros e principais trabalhos relacionados na área e por fim serão desenvolvidos algoritmos das técnicas estudadas.

Por fim, foi desenvolvida uma ferramenta que irá contemplar os objetos de estudo a fim de se realizar o estudo de caso.

1.6. ESTRUTURA DO TRABALHO

O trabalho é composto pelas seguintes partes:

- **Capítulo 1 – Introdução:** Apresentação da área de pesquisa do trabalho com os objetivos, justificativas, motivações, perspectivas de contribuições e metodologia de pesquisa.
- **Capítulo 2 – Redes Neurais Artificiais:** Introduz o estudo da arte e os principais conceitos referentes as redes neurais artificiais com foco na rede neural artificial de múltiplas camadas.
- **Capítulo 3 – Tecnologia Adaptativa:** Apresenta os conceitos e as técnicas referente a tecnologia adaptativa.
- **Capítulo 4 – Estudo de Caso:** Demonstra a utilização dos conceitos utilizados no trabalho e detalha o desenvolvimento de um simulador de rede neural artificial de múltiplas camadas adaptativa.

- **Capítulo 5 – Conclusão e Trabalhos Futuros:** Conclui-se as principais vantagens e desvantagens da utilização de um mecanismo adaptativo na construção de uma rede neural artificial.

2. REDES NEURAIIS ARTIFICIAIS

2.1. INTELIGÊNCIA ARTIFICIAL

Turing (1950) escreveu um dos primeiros artigos relacionado a inteligência artificial. Neste artigo, Turing pretende demonstrar que uma máquina conseguiria executar o papel de um humano, portanto, Turing propõe um jogo denominado como “*O Jogo da Imitação*” para demonstrar a sua ideia. Basicamente, o jogo é composto por três jogadores, onde o jogador (A) seria um interrogador, enquanto o jogador (B) e (C) seriam um de sexo feminino e outro de sexo masculino.

O interrogador têm como o objetivo descobrir o sexo do jogador (B) e (C) através de perguntas, porém o interrogador terá que estar em sala diferente dos outros jogadores, e não poderão ouvir a voz dos jogadores, então é sugerido que as perguntas e respostas sejam escritas a mão. Os jogadores que serão interrogados também possuem algumas regras, onde um deles terá que ajudar o interrogador enquanto o outro terá que atrapalhar o interrogador a chegar na resposta correta.

Também é proposto que o jogador (A) seja uma máquina e os outros jogadores não poderiam saber desta informação, desta forma, caso os jogadores não perceberem que o jogo estava sendo realizado por uma máquina, é possível concluir que uma máquina consegue realizar o papel de um humano.

Shirai e Tsujji (1982), definem duas perspectiva em que se pode abordar a inteligência artificial. Sendo a perspectiva científica, a qual busca compreender o mecanismo de compreensão humana, permitindo o computador realizar uma simulação que verifica teorias sobre a inteligência e a perspectiva de prover o computador das capacidades intelectuais humana.

Marvin Minsky foi um dos principais contribuidores do campo da Ciência da Computação e principalmente no ramo de inteligência artificial. Sabbatini (1998) demonstra os principais estudos realizados por Minsky na área da inteligência artificial e como eles foram importantes para o desenvolvimento do ramo, sendo um deles o estudo que demonstra as principais capacidades e limitação de uma rede neural artificial do tipo perceptron para a utilização de reconhecimento de padrões.

2.2. APRENDIZADO DE MÁQUINA

O aprendizado de máquina é considerado como uma das áreas da inteligência artificial que possui como objetivo o desenvolvimento de técnicas computacionais sobre a construção de sistemas capazes de adquirir conhecimento de forma automática. (BARANAUSKAS, MONARD, 2003).

Batista (2003) demonstra em sua tese a possibilidade de realizar o aprendizado de máquina através de indução de informação, sendo eles:

- **Aprendizado supervisionado:** Neste método de aprendizado, é oferecido ao sistema computacional um conjunto n de entradas $E = \{E1, E2, \dots, En\}$, sendo que para cada entrada existe uma saída $S = \{S1, S2, \dots, Sn\}$ correspondendo ao valor desejado.
- **Aprendizado não supervisionado:** Nesta forma, é entregue apenas o conjunto de entradas para o sistema computacional, deixando o algoritmo sozinho para encontrar alguma estrutura ou relação aos dados de entrada fornecido.

A utilização de algoritmos de aprendizado de máquina é fundamental na construção de uma rede neural artificial, pois são estes algoritmos que irão permitir que uma rede neural artificial tenha a capacidade de aprender e desta forma, regular os seus valores de pesos para gerar o resultado correto.

2.3. REDES NEURAIS

As redes neurais artificiais possuem a finalidade de simular algumas funções de uma rede neural biológica, possibilitando a criação de um sistema que possuirá comportamento inteligentes, sendo eles o de aprender, errar e o de realizar descobertas. Para Barretos (2002) as redes neurais artificiais consistem em um modo de solucionar problemas ligados a inteligência artificial, a qual é fundamentada através dos conceitos biológicos do sistema nervoso.

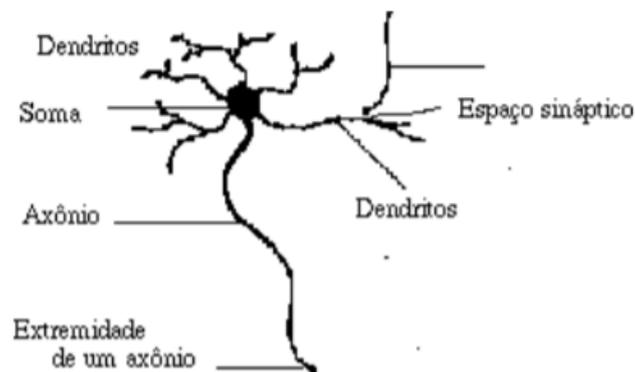


Figura 2: Neurônio Biológico
Fonte: Barretos, 2002

Para Morais (2010), um dos principais atrativos das redes neurais artificiais é a capacidade de aprender através de exemplos e de generalizar a informação aprendida. Desta forma, uma rede neural artificial consegue realizar seu aprendizado através de um número reduzido de exemplos e suas respostas serão coerentes para informações desconhecidas. Iyoda (2000) demonstra os principais problemas solucionados através da utilização de redes neurais artificiais, dentre eles é destacado o reconhecimento de padrões, aproximação de funções e sistemas de controles.

Existem diversas arquiteturas de uma rede neural artificial, onde cada uma conta com vantagens e desvantagens em certas características e para determinados problemas, é recomendado uma arquitetura específica para a solução do problema.

Grande parte das arquiteturas de redes neurais artificiais utilizam como base dos neurônios artificiais de McCulloch e Pitts (1943). Com base nos neurônios artificiais, Rosenblatt (1958) criou um dos tipos mais simples de redes neurais artificiais, o Perceptron.

Em um trabalho realizado por Minsky e Papert (1988) são descritas as principais limitações de um Perceptron, principalmente relacionados a problemas de reconhecimento de padrões. Desta forma, é demonstrado uma rede neural de multicamadas, o qual consegue suprir muitas das limitações anteriormente demonstradas.

Muitas arquiteturas de rede neural artificial foram criadas, porém a que mais possui destaque é a arquitetura de multicamadas, principalmente por sua característica de possuir uma ampla classe de problemas que ela consegue solucionar. (IYODA, 2000)

2.4. NEURÔNIO ARTIFICIAL

McCulloch e Pitts (1943) escreveram um artigo que contribuiu diretamente para o avanço dos estudos das redes neurais artificiais. Neste artigo, é proposto um modelo matemático, que com base nos conhecimentos que eles possuíam na época, simula o funcionamento de um neurônio biológico.

Este formalismo, denominado como Modelo McCulloch e Pitts (MCP) foi de extrema importância para o início das pesquisas relacionadas as redes neurais artificiais, sendo amplamente utilizado como base em muitas outras pesquisas do mesmo âmbito.

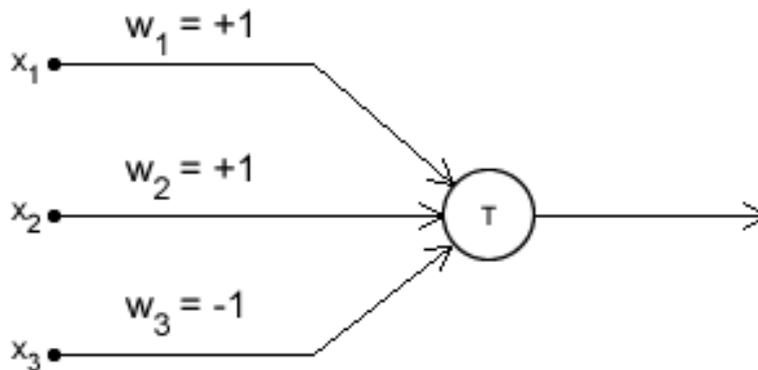


Figura 3: Neurônio Artificial
Fonte: SINHA, 2018

Para a realização da simulação de um neurônio artificial, o modelo MCP utiliza de um conjunto de valores de entrada com um respectivo valor de peso para cada valor.

Desta forma, para verificar se um neurônio será ativado, é realizado o produto escalar dos valores de entrada em função do valor de peso, denominado como função soma, dado pela seguinte fórmula:

$$S = \sum_{i=1}^n x_i w_i$$

Neste modelo, apenas valores lógicos são gerados, caso valor resultado da função soma for menor que o valor definido em T , o resultado que será gerado na saída irá ser 0, da

mesma forma, caso o valor seja maior ou igual, o neurônio artificial será ativado, resultado em uma saída de valor igual a 1.

2.5. PERCEPTRON

Rosenblatt (1958) criou o perceptron, um dos tipos mais simples de implementação de rede neural artificial que utiliza como a base o modelo MCP. Em sua implementação, os pesos descritos no modelo MCP representam o grau de importância da sua entrada correspondente para aquele neurônio.

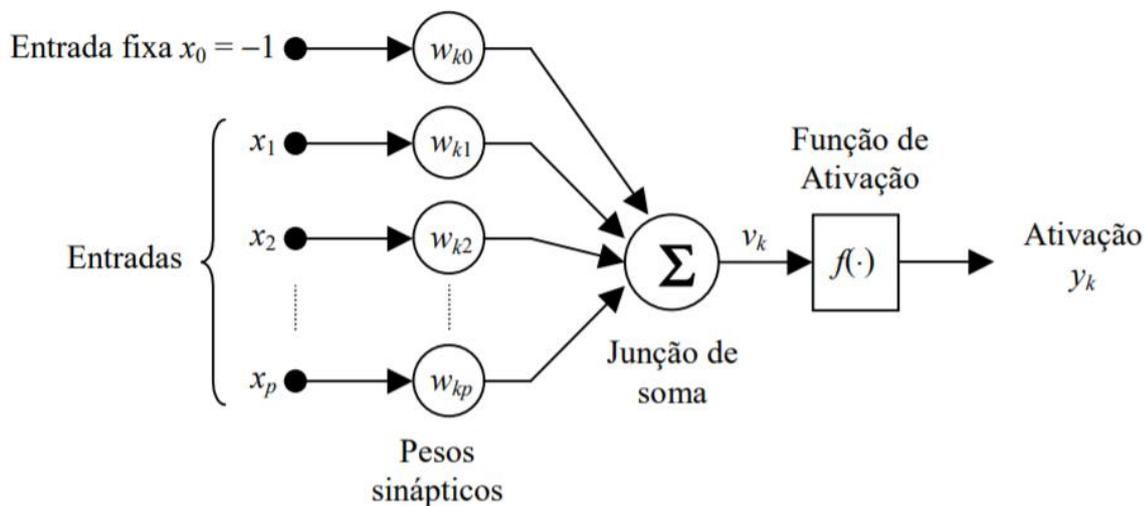


Figura 4: Representação do Neurônio Artificial
Fonte: IYODA, 2000

O perceptron descrito por Rosenblatt é capaz de retornar apenas valores binários como resultado, que é calculado através da função soma. Até este momento, a rede neural descrita por Rosenblatt se assemelha em muito com o modelo MCP, desta forma, é inserida a função de ativação ou função degrau (do inglês *Step Function*).

O funcionamento da função soma pode ser demonstrado através do algoritmo que será demonstrado

Algoritmo 1: Função Soma

```

function funcaoSoma(entradas, pesos)
    local soma = 0
    for i = 1, #entradas, 1 do
        soma = soma + entradas[i] * pesos[i]
    end
    return soma
end

```

Este algoritmo implementa uma função que recebe como valor um vetor contendo todos os valores de entradas e um vetor com os seus respectivos pesos.

Com a função soma calculada, é possível realizar a função de ativação, sendo ela:

$$f(x) = \begin{cases} 1 & \text{se } x \geq 0 \\ 0 & \text{se } x < 0 \end{cases}$$

Basicamente, caso o retorno da função de ativação retornar o valor 1, significaria que o neurônio foi ativado, enquanto para o retorno do valor 0, o neurônio não foi ativado.

Apesar do modelo perceptron ser simples, outros tipos de redes neurais artificiais mais complexas foram originadas utilizando como base este modelo, sendo uma delas a rede perceptron de múltiplas camadas.

2.5.1. APRENDIZADO DO PERCEPTRON

Gallant (1990) demonstra que existem diversas técnicas que buscam realizar o aprendizado de um perceptron, sendo que de acordo com cada algoritmo, possui uma característica majoritária, sendo elas a velocidade de aprendizado do perceptron e a sua escalabilidade.

As técnicas de aprendizagem que utilizam do método de aprendizagem supervisionada possuem alguns passos em comum, sendo eles o estímulo da rede neural pelo o seu ambiente, recebendo valores de entrada, os quais irão realizar o estímulo da rede neural e resultará em um valor.

O valor resultante é comparado com o valor do resultado já verificado previamente, caso os valores sejam diferentes, é realizada alteração nos pesos do perceptron.

Com base nos passos descritos por Gallant (1990), foi realizado um algoritmo que pretende realizar o balanceamento dos pesos de um perceptron de maneira mais simples possível.

Algoritmo 2: Função de Balanceamento

```
function balancearRede(valorCalculado, respostaEsperada, pesos, entradas,
taxaAprendizado)
if valorCalculado ~= respostaEsperada then
    erro = respostaEsperada - valorCalculado
    for i = 1, #pesos, 1 do
        pesos[i] = pesos[i] + (taxaAprendizado * entradas[i] * erro)
    end
end
end
```

2.6. PERCEPTRON DE MÚTIPLAS CAMADAS

Minsky e Papert (1988) apresentam uma crítica referente ao Perceptron descrito por Rosenblatt, onde demonstrava que a utilização de apenas um neurônio não era o suficiente para resolver certos problemas computacionais, sendo um deles a representação da porta *XOR*.

Hinton, Willians e Rumelhart (1985) apresentam uma rede neural artificial de múltiplas camadas com um algoritmo de treinamento eficaz. Este trabalho resultou na volta do interesse dos pesquisadores em relação aos estudos de redes neurais artificiais, pois ele demonstrava uma rede neural artificial que supria algumas das críticas realizadas por Minsky e Papert.

Uma rede neural do tipo perceptron de múltiplas camadas (do inglês *Multilayer Perceptron Neural Network*) (MLP) possui muitas características semelhantes a uma rede perceptron simples descrito por Rosenblatt (1958), porém demonstra como sua principal diferença, a possibilidade de possui diversas camadas de neurônios que realizam sinapse entre eles até a chegada dos valores na camada de saída, gerando um resultado de saída da rede.

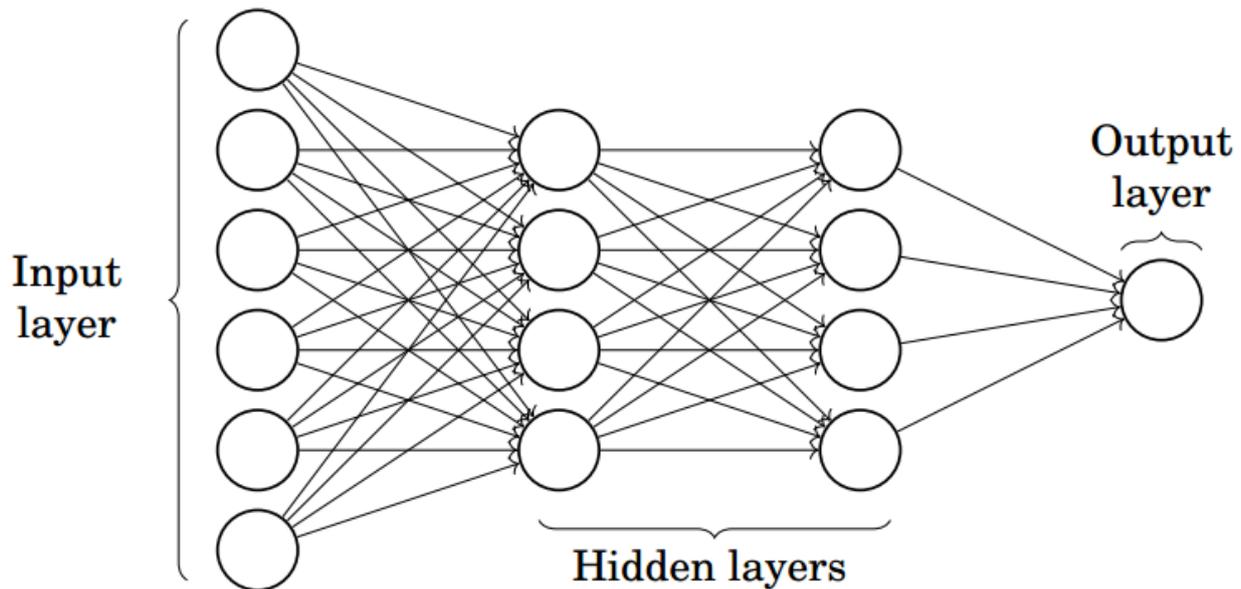


Figura 5: Representação MLP
Fonte: PAVLOVSKY, 2017

Pavlovsky (2017), demonstra que é possível classificar as camadas em três tipos, sendo elas:

- **Camada de entrada (*Input Layer*):** Esta camada recebe um valor de entrada externo e irá distribuir este valor para cada neurônio da próxima camada com um respectivo peso.
- **Camada oculta (*Hidden Layer*):** Diferente da camada de entrada, a camada oculta permite que diversas outras camadas inseridas na rede possuam sua mesma finalidade. A camada oculta irá receber diversos valores com seus respectivos pesos, realizará sua função de ativação e enviará seu valor para a camada de neurônio seguinte.
- **Camada de saída (*Output Layer*):** É considerada como a última camada da rede MLP, onde receberá os valores resultante da última camada oculta com seus respectivos pesos. Desta forma, será possível executar a sua função de ativação e retornar uma resposta da rede. Esta camada pode possuir mais de um neurônio, isto depende do tipo de problema que se deseja solucionar com a rede.

Um dos principais algoritmos de treinamento para uma rede MLP é o algoritmo de retropropagação de erro (do inglês *Backward propagation of errors*). Esta técnica é considerada

como aprendizagem supervisionada, onde os ajuste de pesos da rede são alterados de acordo com os erros que são realizados (BOURLARD, WELLEKENS, 1989).

As principais áreas de aplicação de uma rede MLP são na solução de problemas relacionados a tradução de textos, classificações de dados, reconhecimento de padrões, reconhecimento de voz e reconhecimento de imagens.

Muitas outras redes neurais foram surgindo, muitas possuindo como base a rede MLP, algumas trazendo proposta melhores de solução de alguns problemas.

2.6.1. Rede Neural Convolutacional

Uma rede neural convolutacional, normalmente chamada de CNN, é uma variação das redes de perceptrons de múltiplas camadas, tendo sido inspirada no processo biológico de processamentos de dados visuais. (VARGAS, PAES, VASCONCELOS, 2016)

De acordo com Samer, Rishi e Rowen (2015), a CNN possui expiração biológica do córtex visual humano, já que possui uma grande quantidade de células responsáveis por detectar luz em pequena escala, sobrepondo sub-regiões do campo visual, que são chamadas de campos receptivos, o qual é um processo similar a entrada e processamento de valores na CNN.

Este tipo de rede neural artificial é umas das principais para o desenvolvimento de aplicações que necessitam realizar classificação, detecção e reconhecimento de imagens e vídeos.

Hiraruki (2003), demonstrou a eficiência da utilização da CNN para o desenvolvimento de aplicação que necessita realizar o reconhecimento de imagens. Em seu trabalho, Hiraruki desenvolveu uma Rede Neural Artificial capaz de realizar reconhecimento facial, demonstrando uma taxa de aproximadamente 85% de acertos.

3. TECNOLOGIA ADAPTATIVA

A tecnologia adaptativa consiste em uma técnica que permite uma aplicação mudar seu comportamento. Conforme demonstrado por Camolesi (2007), uma aplicação complexa é aquela que possui os mecanismos necessários para realizar as suas modificações. Essas modificações são realizadas através de um conjunto de funções elementares, as quais permitem realizar a mudança de comportamento em tempo de execução de código através de estímulos de entrada.

Neto (1993) demonstra que para qualquer dispositivo que seja dirigido por regras (dispositivo subjacente) é possível implementar uma camada adaptativa, o qual irá garantir meios que permitem o dispositivo modificar-se.

É possível utilizar da tecnologia adaptativa em diversos dispositivos clássicos, sendo alguns deles o autômato de pilha, redes de Markov, árvores de decisão e tabelas de decisão (TCHEMRA, 2009). Neto (2007), apresenta os conceitos teóricos para a aplicação da tecnologia adaptativa nestes dispositivos.

Pistori (2003) demonstra a utilização de funções elementares adaptativas dentro de um autômato finito. No referente caso, a cada símbolo a lido, é adicionado a sequência de 3 transições no local da transição vazia, sendo elas em sequência: b, ϵ, c .

A seguinte figura demonstra o autômato finito em seu formato inicial e a sua modificação após a execução de funções adaptativas.

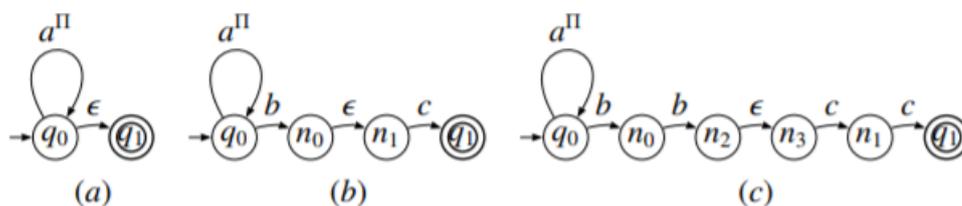


Figura 6: Autômato Finito Adaptativo

Fonte: Pistori, 2003

Um dos primeiros dispositivos a possuir uma camada de funções adaptativas foi o autômato de pilha estruturado, desenvolvido por Neto (1993).

É demonstrado que dentro das funções adaptativa, é necessário possuir três funções primitivas, sendo elas:

- **Consulta:** As ações elementares do tipo consultar irá verificar no dispositivo subjacente regras que condizem a regra fornecida na busca.
- **Exclusão:** Estas ações irão remover um conjunto de regras do programa, de acordo com a regra de comparação fornecida, o dispositivo terá que continuar com o seu funcionamento normalmente após a exclusão da regra.
- **Inclusão:** Será adicionado ao dispositivo subjacente um conjunto de regras, as quais serão inseridas em conjuntos com as regras já existentes do dispositivo.

Caso um dispositivo subjacente tenha implementado uma camada de função adaptativa que possui estes três tipos de função, este dispositivo terá a característica de ser adaptável.

Pistori (2003) demonstra um estudo detalhado em relação ao estado da arte da tecnologia adaptativa, tanto em âmbito nacional como internacional. Também é destacado as principais áreas de aplicações possíveis para a tecnologia adaptativa, como sendo umas das principais a inteligência artificial, que possui diversos ramos que pode se beneficiar com a utilização da tecnologia adaptativa, como as árvores de decisão, tabela de decisão e nos processos de aprendizagem de máquina.

Luz e Neto (2003) apresentam um trabalho em que foi possível melhorar a compressão de código em um compilador. Através da adição de uma ação adaptativa no momento de uma análise de regras de compressão, permitiu que o algoritmo conseguisse filtrar melhor as melhores regras, permitindo que o código gerado, fosse mais comprimido que anteriormente.

Rossini e Camolesi (2017) desenvolveram um jogo de dominó adaptativo. Neste jogo, existiam jogadas que permitiam realizar alguma operação adaptativa no conjunto de regras do dominó, o qual era utilizado como o dispositivo subjacente.

Para realizar a mudança de regras, foi adicionado um mecanismo adaptativo o qual era capaz de modificar as regras do jogo em tempo de execução de código.

A principal contribuição deste trabalho foi da utilização da técnica de padrão de projeto, sendo ela a injeção de dependência, para a inserção do mecanismo adaptativo em um dispositivo subjacente já construído, permitindo uma fácil integração do mecanismo adaptativo.



Figura 7: Dominó Adaptativo
Fonte: ROSSINI e CAMOLESI, 2017

4. ESTUDO DE CASO

Neste capítulo será demonstrado o desenvolvimento do estudo de caso desenvolvido neste projeto de conclusão de curso. Primeiramente, será demonstrado as análises dos estudos da utilização de técnicas adaptativas para o desenvolvimento de uma rede neural artificial, a qual neste trabalho, foi escolhido utilizar de uma rede do tipo MLP como foco de desenvolvimento.

Para demonstrar os conceitos estudados, foi desenvolvido um simulador de rede MLP, a qual é possível executar operação adaptativa na rede e alterar sua arquitetura em tempo de execução.

4.1. REDE NEURAL ARTIFICIAL DE MÚLTIPLAS CAMADAS ADAPTATIVA

Através dos estudos realizados em relação as redes neurais artificiais, é demonstrado que o próprio formalismo de uma rede neural artificial já possui aspecto adaptativo. Basicamente, devido a sua característica de aprendizagem, é possível utilizar de uma mesma estrutura de uma rede neural artificial para resolver problemas diferentes.

Specht (1990) demonstra o grande impacto dos algoritmos de treinamento na rede neural artificial MLP, onde os pesos sofrem constantes alterações com o objetivo de tornar a rede funcional para a solução do problema proposto.

Em alguns casos, como evidenciado por Basu et al. (2012), em se tratando de um rede neural artificial MLP, existem casos que a quantidade de camadas influencia diretamente na precisão de acerto da rede. No caso realizado representado pela a figura, após a quantidade de 65 camadas de neurônios ocultos, o percentual de precisão começa a decair.

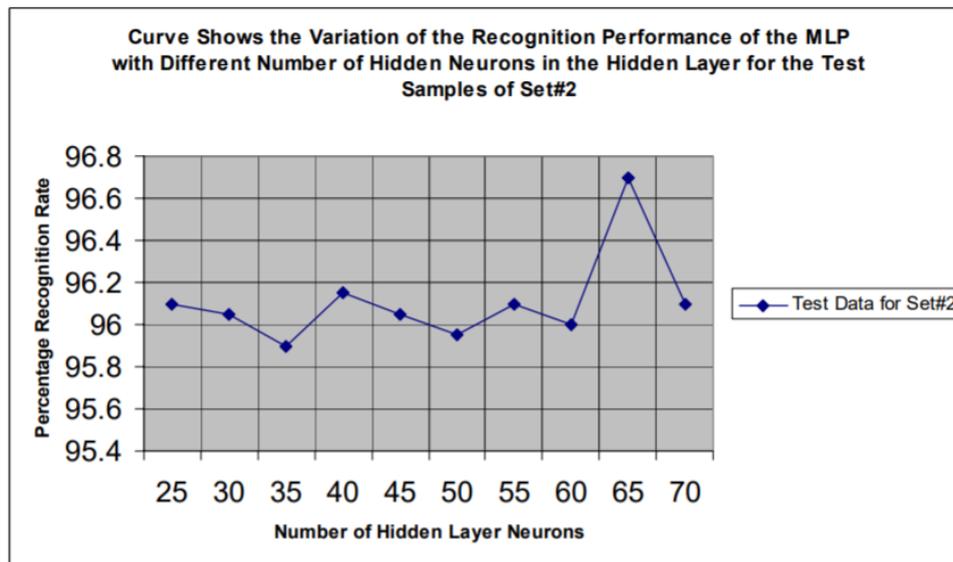


Figura 8: Percentual de Reconhecimento
Fonte: Basu et al, 2012

4.1.1. FUNÇÕES ADAPTATIVAS

Com base nestas análises, são proposta três funções adaptativas capazes de realizar modificação na arquitetura da rede neural artificial MCP.

- **Alterar função de ativação de neurônio:** Esta função terá como objetivo alterar a função de ativação de um neurônio, desta forma, em um sistema baseado em uma rede neural artificial MCP será possível alterar as funções de ativação dos neurônios em tempo de execução de código, caso tenha mais de uma função desta mesma finalidade programada.
- **Remover camada oculta:** Com esta função, será possível remover toda a camada de neurônios. Esta função pode ser utilizada em casos em que se precisa que a rede neural artificial MLP funcione de maneira mais rápida possível, e desta forma, é retirado uma camada de neurônios.
- **Adicionar camada oculta:** Nesta função, será possível adicionar uma nova camada oculta dentro da rede neural artificial MLP. Isto permite que caso necessite uma rede neural com um maior número de camadas, para principalmente aumentar o percentual de precisão.

Estas funções apenas foram desenvolvida para serem utilizadas nas camadas ocultas da rede neural artificial MCP.

4.1.1.1. FUNÇÃO DE ALTERAÇÃO DE FUNÇÃO DE ATIVAÇÃO

Para utilizar a função de alteração de ativação de neurônio, é necessário realizar a passagem dos valores necessários por parâmetro, sendo eles: rede (a rede neural que será executada a função adaptativa), posição da camada, posição do neurônio (caso for necessário mudar todos os neurônios, é necessário informar o valor como 0) e a função de ativação (a qual irá substituir a função anterior).

Algoritmo 3: Alteração de Função de Ativação

```
function alterarFuncaoAtivacao(rede, posCamada, posNeuronio, funcao)
    if posNeuronio == 0 then
        for i = 1, #rede.camadasOcultas[posCamada].neuronios, 1 do
            rede.camadasOcultas[posCamada].neuronios[i].fAtivacao = funcao
        end
    else
        rede.camadasOcultas[posCamada].neuronios[posNeuronio].fAtivacao
            = funcao
    end
end
```

A linguagem lua por ser considerada como uma linguagem funcional, é possível realizar a passagem de função por parâmetros e guarda as mesmas em variáveis.

4.1.1.2. FUNÇÃO DE REMOÇÃO DA CAMADA OCULTA

Esta função possibilita a remoção de uma camada de neurônio, sendo do tipo oculta. É possível remover a camada independente da sua posição.

A execução da função é possível através da passagem da rede neural artificial e da posição em que se deseja remover a camada de neurônios.

Existem três possíveis casos para a remoção da camada oculta que precisam ser analisados para a construção do algoritmo, sendo eles:

- **Remoção na primeira camada de neurônios oculto:** Este caso se refere para quando a camada que será excluída é a camada que recebe diretamente os valores da camada de entrada. Para realizar a remoção da camada, é necessário modificar o caminho em que os valores irão seguir na camada de entrada, para ao invés de seguirem em direção a camada que foi excluída, eles serem encaminhados para a camada posterior.
- **Remoção de camada intermediária de neurônio:** É referente à quando a camada de neurônio não se encontra após a camada de entrada e anteriormente a camada de saída. Desta forma, é necessário realizar a modificação nos axônios na camada anterior a excluída e modificar os dendritos do neurônio posterior.
- **Remoção da última camada de neurônios oculto:** A última camada oculta é responsável por enviar os valores finais para a camada de saída. Na realização da remoção da última camada de neurônios oculto, é necessário modificar os axônios da camada anterior a removida, fazendo uma ligação entre esta camada e a camada de saída.

Algoritmo 4: Remoção da Camada Oculta

```

function removerCamadaOculta(rede, posCamada)
    if #rede.camadasOcultas > 1 then
        if posCamada == 1 then
            for i = 1, #rede.camadaEntrada.neuronios, 1 do
                rede.camadaEntrada.neuronios[i].axonios = {}
                local qtd = #rede.camadasOcultas[posCamada + 1].neuronios
                for j = 1, qtd, 1 do
                    table.insert( #rede.camadaEntrada.neuronios,
                                   { j, math.random(0,10) })
                end
            end
        end
        return true
    end
end

```

No algoritmo anterior, foi apenas demonstrado parte do trecho em que se encontra o código referente a condição em que será removido a primeira camada de neurônio oculto.

Através da execução do código, foi possível perceber que a função adaptativa de remoção de camada poderia ocasionar problemas para a rede neural artificial.

Conforme demonstrado, uma camada de neurônios ocultos pode possuir diversos números de neurônios, não precisando possuir necessariamente a mesma quantidade de neurônios das camadas que fazem ligação direta com ele.

Ao realizar a remoção da camada oculta, a topologia da rede será alterada e as novas conexões de neurônios que serão realizadas necessitarão de um valor de peso. Caso a rede neural já esteja treinada para solucionar um certo tipo de problema, é muito provável que ela não irá conseguir executar corretamente este problema, pelo fato da rede neural ter sido modificada, desta forma será necessário realizar o treinamento da rede neural

artificial novamente para esta nova topologia, para que possa equilibrar os seus pesos e voltar a gerar resultados corretos.

4.2. IMPLEMENTAÇÃO DO SIMULADOR

Para o desenvolvimento do simulador foi utilizado a linguagem de programação Lua¹ com a utilização da framework Love2d² para o desenvolvimento da parte gráfica.

O simulador inicia com uma rede neural artificial do tipo MLP já inicializado, contendo a camada de entrada, duas camadas ocultas e a camada de saída.

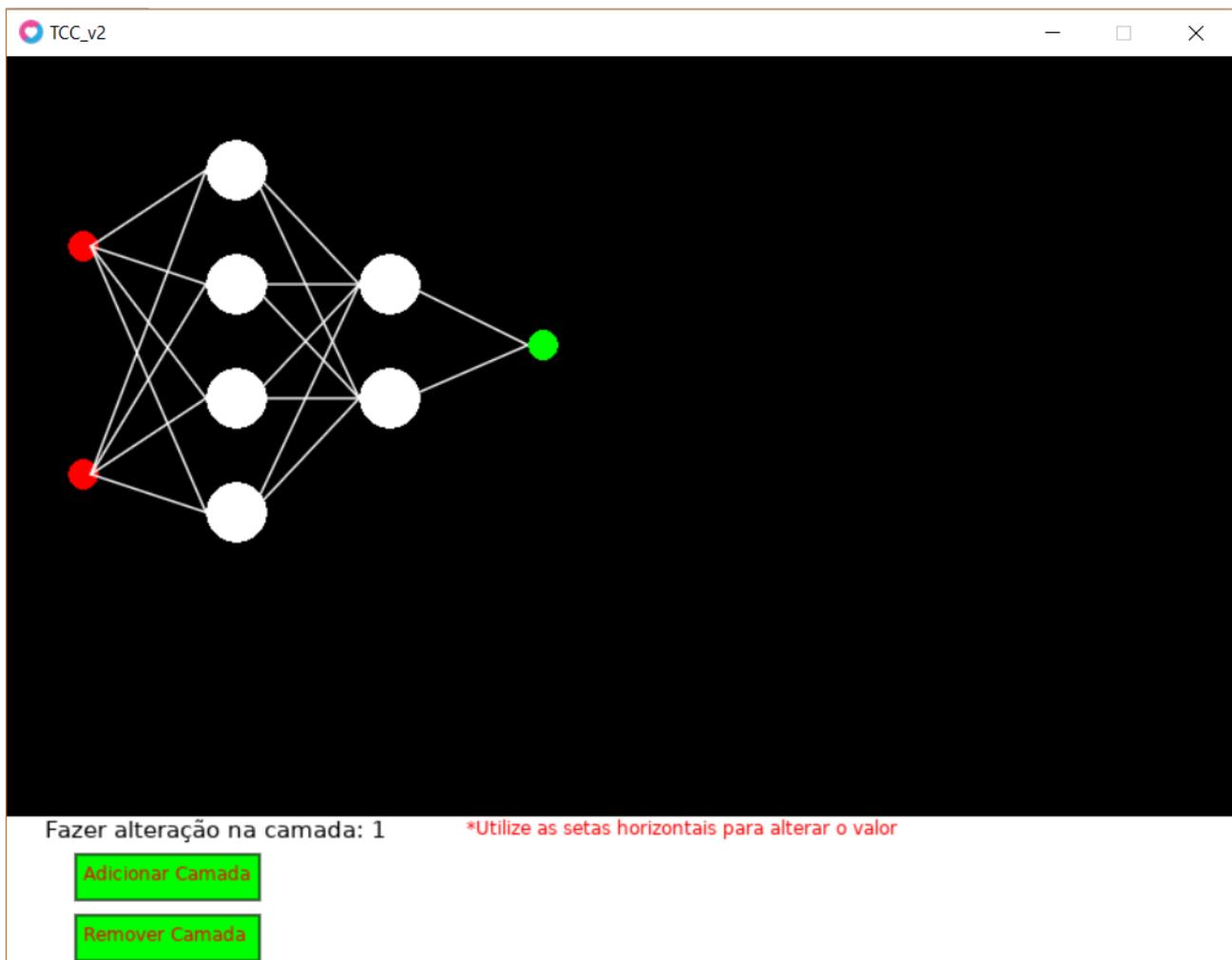


Figura 9: Formato Inicial da Rede Neural

¹ <https://www.lua.org/>

² <https://love2d.org/>

A aplicação conta com uma interface de usuário simples, onde é possível alterar a camada em que será realizada a função adaptativa.

O usuário tem as opções de adicionar e remover as camadas, apenas clicando nos botões.

Na figura seguinte, será demonstrado o caso em que será adicionado uma nova camada com 2 neurônios na 1^o posição das camadas ocultas.

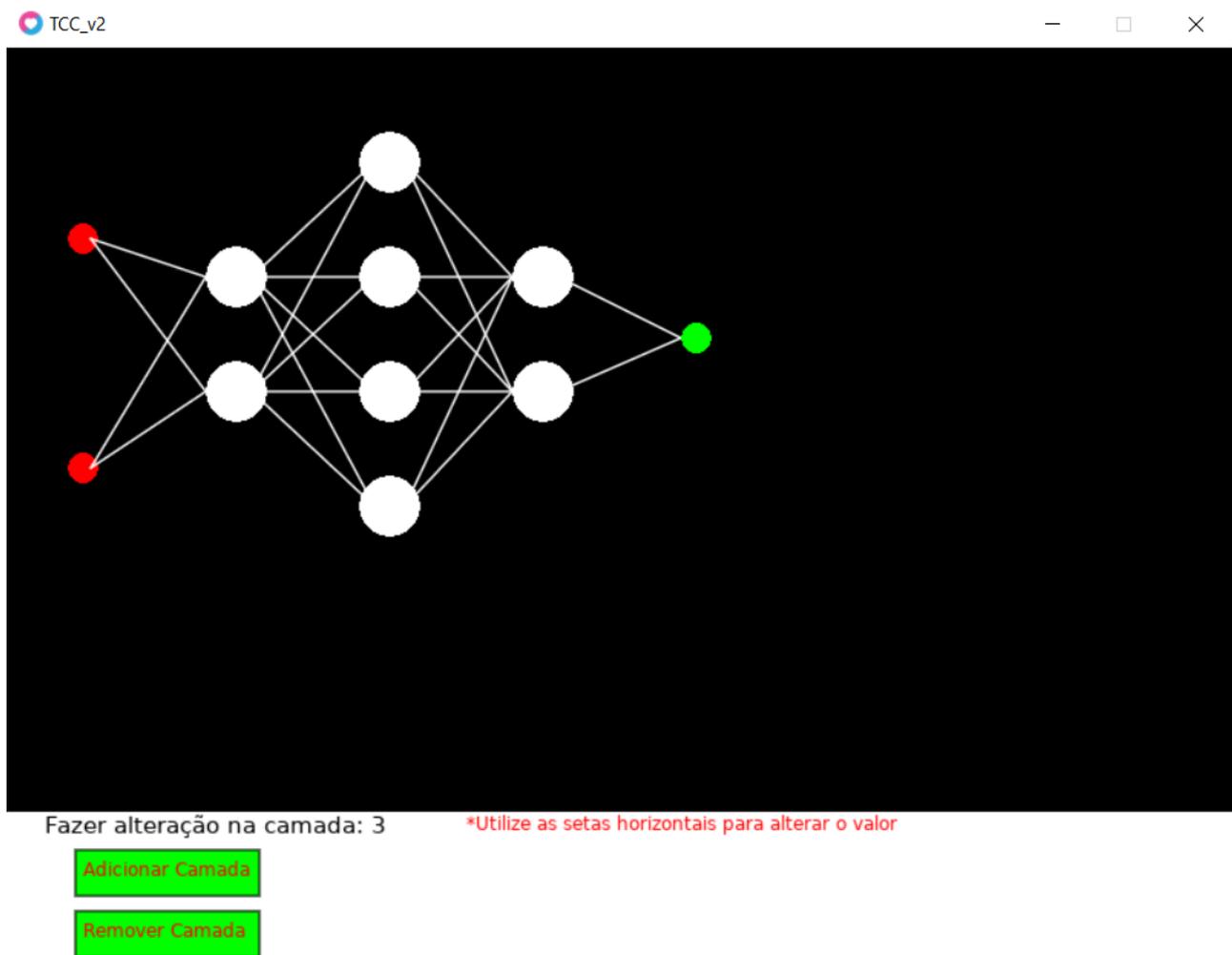
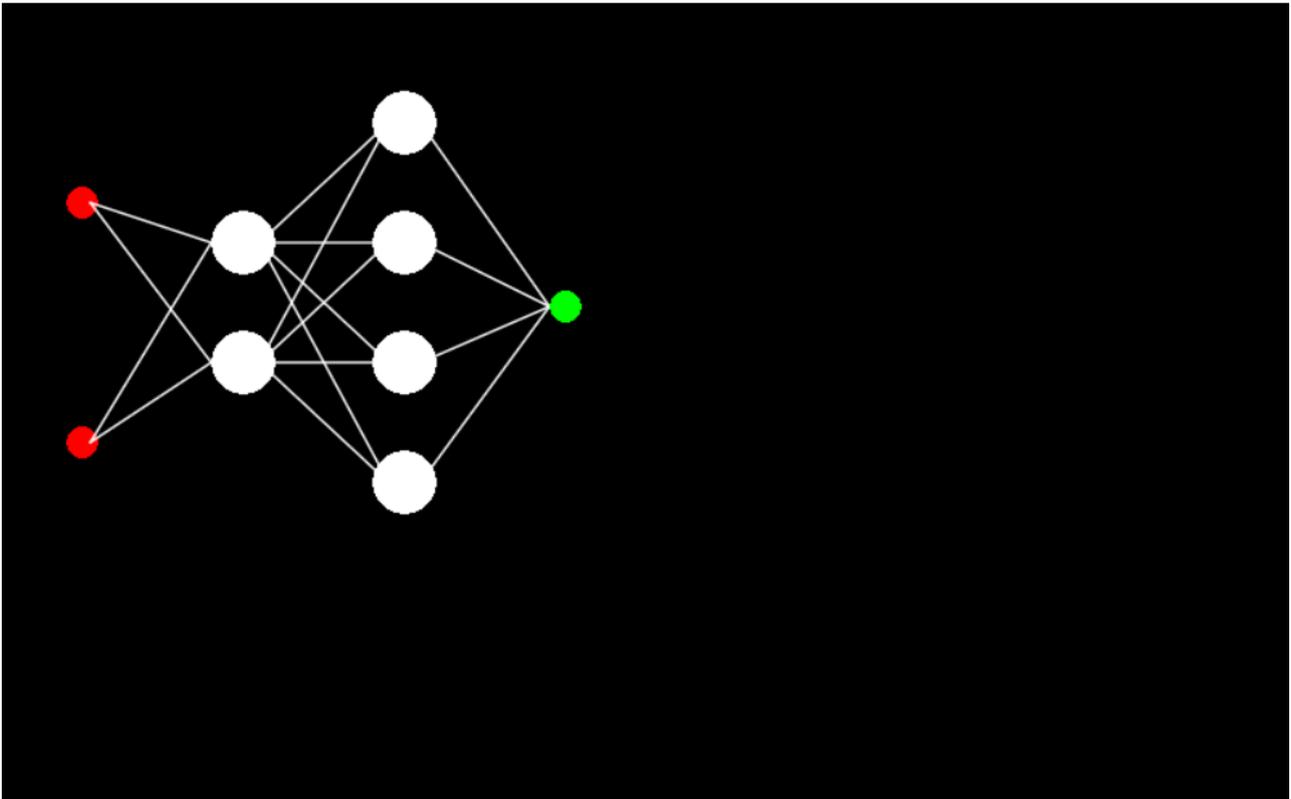


Figura 10: Adição de Camada de Neurônio

Uma nova camada de neurônios foi adicionada a rede, contendo apenas 2 neurônios. Esta camada recebe todas as ligações da camada anterior, a qual seria a camada de entrada, e realiza a ligação com a camada em que ela ocupou o lugar.

Na próxima figura, será demonstrado o resultado da rede neural ao remover a 3^o camada de neurônios ocultos, o qual será necessário realizar a ligação de neurônios com a camada de saída.

TCC_v2



Fazer alteração na camada: 3

**Utilize as setas horizontais para alterar o valor*

Adicionar Camada

Remover Camada

Figura 11: Remoção de Camada de Neurônio

5. CONCLUSÃO E TRABALHOS FUTUROS

Neste projeto, foi apresentado os principais conceitos relacionados as redes neurais artificiais e da tecnologia adaptativa, o qual este estudo gerou o desenvolvimento de um simulador de uma rede neural artificial de múltiplas camadas adaptativo. Este simulador apresenta como característica adaptativa, a possibilidade de realizar alteração em sua estrutura de rede neural artificial em tempo de execução.

Conforme demonstrado por Basu, et Al. (2012), o percentual de precisão de uma rede neural artificial de múltiplas camadas podem variar de acordo com a quantidade de camadas de neurônios em que a rede a possui. Esta pesquisa oferece um mecanismo adaptativo que irá permitir realizar a alteração na estrutura da rede neural artificial, especificamente na quantidade de camadas de neurônios que a rede irá possuir.

Deve-se observar que a utilização do mecanismo adaptativo proposto neste trabalho na rede neural artificial de múltiplas camadas possui algumas limitações de acordo com o contexto.

Ao mudar a estrutura de uma rede neural artificial, é muito provável que será necessário treinar a rede novamente, para que possa continuar gerando resultados corretos. Em um ambiente, onde os valores de peso já são conhecido para quando alguma alteração da estrutura da rede neural artificial for realizada, o mecanismo adaptativo proposto pode ser considerado uma ótima ferramenta, porém caso não se tenha o valor destes pesos que serão necessários para adicionar ou ajustar na rede neural, a utilização do mecanismo adaptativo pode atrapalhar na geração de resultados corretos da rede neural artificial, sendo necessário realizar o processo de treinamento novamente.

As principais contribuições deste trabalho foram a de utilizar conceitos da tecnologia adaptativa na linguagem Lua e no desenvolvimento de algoritmos que permitem a adaptabilidade em uma rede neural artificial.

5.1. TRABALHOS FUTUROS

Um dos principais problemas encontrado no desenvolvimento do simulador de rede neural artificial de múltiplas camadas foi de atribuição de valor de pesos para as novas conexões que eram geradas.

Uma sugestão de proposta de trabalho futuro, é de encontrar algum método de minimizar o treinamento necessário que a rede neural de múltiplas camadas terá que realizar após executar as funções adaptativas.

É proposto também, a busca de novas possibilidades de implementação de funções adaptativas nas redes neurais artificiais, e da utilização de um estudo de caso utilizando o mecanismo adaptativo proposto neste trabalho.

Com relação ao simulador, novas versões podem ser geradas, que poderiam demonstrar a execução da rede neural de múltiplas camadas com determinados valores e a melhoria da interface de usuário.

6. REFERÊNCIAS

- BARRETO, J. M. **Introdução as redes neurais artificiais**. V Escola Regional de Informática. Sociedade Brasileira de Computação, Regional Sul, Santa Maria, Florianópolis, 2002.
- BASU, S. et al. **An MLP based Approach for Recognition of Hand written 'Bangla' Numerals**. Computer Sce. & Engg. Dept., MCKV Institute of Engineering, Liluah, India, 2012.
- BATISTA, G. E. A. P. **Pré-processamento de dados em aprendizado de máquina supervisionado**. Tese de Doutorado. Universidade de São Paulo, 2003.
- BELL, G.; HEY, T.; SZALAY, A. **Beyond the Data Deluge**. Science 325, p. 1297-1298, 2009
- BENGIO, Y. **Learning deep architectures for ai**. Foundation and Trends in Machine Learning, v. 2, 2009.
- BOURLARD, H.; WELLEKENS, C.J. **Links between Markov models and multilayer perceptrons**. In Advances in neural information processing systems (pp. 502-510), 1989.
- CAMOLESI, A. R. **Proposta de um Gerador de Ambientes para Modelagem de Aplicações Usando Tecnologia Adaptativa**. Tese de doutorado, Escola Politécnica da USP, 2007.
- DUTRA, R. G.; Martucci, M. **Adaptive Fuzzy Neural Tree Network**. **IEEE Latin America Transactions**. Volume 6, Issue 5, ISSN: 1548-0992, pp. 453-460, September 2008.
- GALLANT, S. I. **Perceptron-based learning algorithms**. IEEE Transactions on neural networks 1.2 (1990): 179-191.
- HIRARUKI, M. H. **Aplicação da Rede Neural Artificial para Reconhecimento de Atributos Faciais**. Trabalho de pós-graduação, Universidade Federal de São Carlos, 2003.
- IYODA, E. M. **Inteligência Computacional no Projeto Automático de Redes Neurais Híbridas e Redes Neurofuzzy Heterogêneas**. Tese de Mestrado, Faculdade de Engenharia Elétrica e Computação, 2000.

LUZ, J. C.; NETO, J. J. **Tecnologia Adaptativa Aplicada à Otimização de Código em Compiladores**. IX Congreso Argentino de Ciencias de la Computación, La Plata, Argentina, 6-10 de Outubro, 2003.

MCCULLOCH, W. S.; PITTS, W. H. **A logical calculus of the ideas immanent in nervous activity**. Bulletin of mathematical biophysics, v. 5, p 115-133, 1943.

MINSKY, M. L; PAPER, S. **Perceptrons**. MIT Press, Cambridge, MA, v. 15, 1988.

MONARD, M. C; BARANAUSKAS, J. A. **Conceitos sobre aprendizado de máquina**. Sistemas Inteligentes-Fundamentos e Aplicações, 2003.

MORAIS, C. M. **Reconhecimento de padrões e redes neurais artificiais em predições de estruturas secundárias de proteínas**. Teste de Doutorado, Universidade Federal de Rio de Janeiro, 2010.

NETO, J. J. **Contribuições à Metodologia de Construção de Compiladores**. Teste de livre docência. Escola Politécnica da USP, São Paulo, 1993.

NETO, J. J. **Um levantamento da evolução da Adaptatividade e da Tecnologia Adaptativa**. IEEE Latin America Transactions, 5 v., n.7, 2007.

PAVLOVSKY, V. **Introduction To Artificial Neural Networks**. 2017. Disponível em: <www.vaetas.cz/posts/introduction-artificial-neural-networks>. Acesso em 27 jul. 2018.

PISTORI, H. **Tecnologia Adaptativa em Engenharia de Computação: Estado da Arte e Aplicações**. Tese de Doutorado, USP, São Paulo, 2003.

Rosenblatt, F. **The perceptron: a probabilistic model for information storage and organization in the brain**. Psychological review 65.6, 1958.

ROSSINI, C. R. J.; CAMOLESI, A. R. **O uso de conceitos de injeção de dependência no desenvolvimento de aplicações adaptativas**. Em: Memórias do XI Workshop de Tecnologia Adaptativa - WTA 2017. EPUSP, São Paulo. ISBN: 978-85-86686-90-0, pp. 94-99. 26 e 27 de Janeiro, 2017.

ROWLEY, H. A. **Neural network-based face detection**. IEEE Transactions on pattern analysis and machine intelligence, v. 20, n. 1, p. 23-38, 1998.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. **Learning internal representations by error propagation**. California Univ San Diego La Jolla Inst for Cognitive Science, 1985.

- SABBATINI, R. M. E. **A Mente, Inteligência Artificial e Emoções**. 1998. Disponível em: <www.cerebromente.org.br/n07/opiniao/minsky/minsky.htm>. Acesso em 27 jul. 2018.
- SAMER, C. H.; RISHI, K.; ROWEN. **Image Recognition Using Convolutional Neural Networks**. Cardence Whitepaper, p 1-12, 2015.
- SHIRAI, Y; TSUJII, J. **Inteligência Artificial, Conceitos, Técnicas e Aplicações**. Trab. Sob a direção de António Realinho, 1988.
- SPECHT, D. F. **Probabilistic neural networks**. Neural networks, v. 3, n. 1, p. 109-118, 1990.
- TCHEMRA, A.H. **Tabela de Decisão Adaptativa na Tomada de Decisão Multicritério**. Tese de Doutorado, Escola Politécnica da USP, 2009.
- TURING, A. M. **Computing Machinery and Intelligence**. Mind 49: 433-460, 1950.
- VARGAS, A. C. G.; PAES, A.; VASCONCELOS, C. N. **Um Estudo sobre Redes Neurais Convolucionais e sua Aplicação em Detecção de Pedestres**. In: CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES, 29. (SIBGRAPI), São José dos Campos. Porto Alegre: Sociedade Brasileira de Computação, 2016.