

UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE CIÊNCIAS NATURAIS E EXATAS  
CURSO DE GRADUAÇÃO EM FÍSICA

Tiago de Souza Farias

**APLICAÇÃO DE INTELIGÊNCIA ARTIFICIAL EM MODELAGEM  
CIENTÍFICA**

Santa Maria, RS  
2016

**Tiago de Souza Farias**

**APLICAÇÃO DE INTELIGÊNCIA ARTIFICIAL EM MODELAGEM CIENTÍFICA**

Trabalho Final de Graduação apresentado ao Curso de Graduação em Física da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Bacharel em Física**.

ORIENTADOR: Prof. Jonas Maziero

Santa Maria, RS  
2016

**Tiago de Souza Farias**

**APLICAÇÃO DE INTELIGÊNCIA ARTIFICIAL EM MODELAGEM CIENTÍFICA**

Trabalho Final de Graduação apresentado ao Curso de Graduação em Física da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de **Bacharel em Física**.

**Aprovado em 15 de dezembro de 2016:**

---

**Jonas Maziero, Dr. (UFSM)**  
(Presidente/Orientador)

---

**Lucio Strazzabosco Dorneles, Dr. (UFSM)**

---

**Eleonir João Calegari, Dr. (UFSM)**

---

**Fábio Mallmann Zimmer, Dr. (UFSM)**

Santa Maria, RS  
2016

## AGRADECIMENTOS

*Aos meus pais, por tudo.*

*Aos meus avós pelo apoio que recebi durante toda a minha vida.*

*Agradeço também ao meu orientador Jonas Maziero, sem o apoio dele este trabalho não teria sido possível.*

*Agradeço a todos os meus professores, desde o ensino fundamental até o departamento de física da universidade, pois me ajudaram a trilhar o caminho do conhecimento.*

*Aos meus grandes amigos, Diego, Mateus, Helena e Giovani, que compartilharam comigo momentos importantes de minha vida na UFSM.*

*Agradeço ao pessoal dos laboratórios TMC e QIEP, onde passei grande parte de meu tempo universitário.*

*Aos amigos mais recentes, Mauro, Diovana, Maira e Denilson.*

*Por último, mas não menos importante, agradeço a população brasileira pelo suporte ao qual eu pude crescer profissionalmente.*

*Assim como há odores que os cães podem cheirar e não podemos, assim como há sons que os cães podem ouvir e não podemos, também há comprimentos de onda de luz que não podemos ver e sabores que não podemos provar. Por que então, dado o nosso cérebro ligado da maneira que está, a observação "Talvez haja pensamentos que não podemos pensar", lhe surpreende? Evolução, até agora, possivelmente nos impediu de ser capaz de pensar em algumas direções; Podem haver pensamentos impensáveis.*

*(Richard Hamming)*

## RESUMO

# APLICAÇÃO DE INTELIGÊNCIA ARTIFICIAL EM MODELAGEM CIENTÍFICA

AUTOR: Tiago de Souza Farias

ORIENTADOR: Jonas Maziero

Inteligência Artificial (I.A.) é uma área crescente de pesquisa para produzir agentes inteligentes que possam resolver problemas de difícil tratamento humano e auxiliar em atividades computacionais. Rede neural é uma categoria de I.A. inspirada nos conceitos de inteligência em neurônios biológicos, simulando o comportamento de atividade cerebral. A característica principal de redes neurais é sua habilidade em aprender e se adaptar a um problema de interesse. Esta área vem apresentando grande sucesso nos últimos anos em vários campos de pesquisa, como computação visual e teoria de jogos. Apesar do número cada vez maior de aplicações em ciência computacional, a aplicação de I.A. em pesquisa científica ainda é muito nova e pouco conhecida. Este trabalho considera aplicações de I.A. em modelagem científica. Foram analisados modelos físicos simples, como aproximação de funções, problema do pêndulo inverso e a previsão climática como o problema complexo. Os resultados foram analisados de acordo com o tipo de rede neural e de suas topologias, como número de neurônios. Mostramos que inteligência artificial é capaz de resolver certos problemas, mas que ainda há restrições do quão bem ela pode ir. Por fim, a conclusão discute a qualidade dos resultados e indica como as limitações presentes de I.A. podem ser contornadas.

**Palavras-chave:** Inteligência artificial. Rede neural. Modelagem científica. Física. Previsão. Clima.

## ABSTRACT

# APPLICATION OF ARTIFICIAL INTELLIGENCE IN SCIENTIFIC MODELLING

AUTHOR: Tiago de Souza Farias

ADVISOR: Jonas Maziero

Artificial Intelligence (A.I.) is a growing field of research aiming to make intelligent agents which can solve hard problems and auxiliate in computational activity. Neural network is an A.I. field inspired by concepts of intelligence in biological neurons, simulating the brain behavior. The main feature of neural networks is their ability to learn and adapt itself to a problem of interest. This field have accomplished remarkable success in the past few years, like visual computation and game theory. Despite this growing number of applications in computer science, A.I. applied to scientific is just beginning and remains unknown to most people.

This work considers A.I. applications in scientific modeling. Function approximation and inversed pendulum were treated as a simple model to study and weather forecasting as a hard problem of main interest. The results were analyzed with respect to the type of neural network and their topology, like the number of neurons. We have shown that artificial intelligence is able to solve certain kinds of problems, but there are restrictions of how far it can go. Lastly, the conclusion discuss the quality of the results and points out how the limitations of A.I. can be circumvented.

**Keywords:** Artificial intelligence. Neural network. Scientific modelling. Physics. Weather. Forecasting

## LISTA DE FIGURAS

Figura 2.1 – Representação de um neurônio biológico .....	13
Figura 2.2 – Representação de um neurônio artificial. $x$ representa as informações de entrada, $y$ é a informação de saída e $f$ é a operação não linear .....	14
Figura 2.3 – Exemplo de uma possível rede neural .....	14
Figura 2.4 – Exemplo de uma rede neural sofisticada .....	14
Figura 2.5 – Exemplo de um neurônio de várias operações, onde $\sigma y$ representa o produto de $y$ e $\sigma$ .....	16
Figura 2.6 – Rede neural que aproxima a função XOR .....	16
Figura 2.7 – Exemplo de custo em relação ao peso .....	18
Figura 2.8 – Circuito elétrico equivalente do neurônio integra-e-dispara .....	22
Figura 2.9 – Circuito elétrico equivalente do neurônio de Hodgkin-Huxley .....	23
Figura 3.1 – Figura representando três condições do pêndulo invertido. (a) e (c) fora do equilíbrio. (b) no equilíbrio instável .....	26
Figura 3.2 – Captural de tela do ambiente virtual para o equilíbrio instável do pêndulo inverso .....	27
Figura 4.1 – Aproximação de $\sin^2(x)$ para vários períodos de treinamento. (a) 5 iterações, (b) 10 iterações, (c) 20 iterações, (d) 100 iterações .....	29
Figura 4.2 – Função erro em relação ao período de treinamento .....	30
Figura 4.3 – Codificação do função $y = \sin^2(x)$ pela rede neural .....	30
Figura 4.4 – Tempo de pulso para cada neurônio na frequência mais baixa .....	31
Figura 4.5 – Decodificação por tempo da rede neural .....	31
Figura 4.6 – (a) Frequência de pulsos para cada neurônio. (b) Decodificação por frequência da rede neural .....	32
Figura 4.7 – Tempo de equilíbrio instável em relação ao período de treinamento .....	32
Figura 4.8 – Tempo de equilíbrio instável em relação ao período de treinamento .....	33
Figura 4.9 – Tempo de equilíbrio instável em relação ao período de treinamento .....	33
Figura 4.10 – Relação temperatura x tempo para a previsão da rede neural com um mês de dados de treinamento .....	34
Figura 4.11 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural com um mês de dados de treinamento .....	35
Figura 4.12 – Relação temperatura x tempo para a previsão da rede neural com três meses de dados de treinamento .....	35
Figura 4.13 – Relação temperatura x tempo para a previsão da rede neural, 6 meses de dados .....	36
Figura 4.14 – Relação temperatura x tempo para a previsão da rede neural do mês de abril/2016 .....	36
Figura 4.15 – Relação temperatura x tempo para a previsão da rede neural do mês de maio/2016 em Porto Alegre .....	37
Figura 7.1 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural para maio/2016 - Santa Maria, 3 meses de dados .....	43
Figura 7.2 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural para maio/2016 - Santa Maria, 6 meses de dados .....	44
Figura 7.3 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural para maio/2016 - Porto Alegre, 1 mês de dados .....	44
Figura 7.4 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural	



para abril/2016 - Santa Maria, 1 mês de dados (maio) ..... 45

## LISTA DE QUADROS

Quadro 2.1 – Função XOR .....	16
-------------------------------	----

## LISTA DE ABREVIATURAS E SIGLAS

<i>I.A.</i>	Inteligência Artificial
<i>XOR</i>	Ou exclusivo
<i>STDP</i>	Plasticidade dependente de pulsos temporais
<i>LTD</i>	Depressão de longa duração
<i>LTP</i>	Potenciação de longa duração

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b> .....	<b>11</b>
<b>2</b>	<b>DESENVOLVIMENTO</b> .....	<b>13</b>
2.1	NEURÔNICO BIOLÓGICO .....	13
2.2	NEURÔNIO ARTIFICIAL .....	13
2.3	TEORIA DE CONEXIONISMO .....	15
2.4	NEURÔNIOS DE PRIMEIRA GERAÇÃO .....	15
2.5	NEURÔNIOS DE SEGUNDA GERAÇÃO .....	15
2.6	PARADIGMAS DE APRENDIZADO .....	17
<b>2.6.1</b>	<b>Treinamento supervisionado</b> .....	<b>17</b>
2.6.1.1	<i>Técnicas de minimização</i> .....	18
2.6.1.2	<i>Redes neurais de várias camadas</i> .....	19
<b>2.6.2</b>	<b>Treinamento reforçado</b> .....	<b>20</b>
<b>2.6.3</b>	<b>Treinamento não supervisionado</b> .....	<b>21</b>
2.7	NEURÔNIOS DE TERCEIRA GERAÇÃO .....	21
<b>2.7.1</b>	<b>Modelos de neurônios de terceira geração</b> .....	<b>22</b>
<b>2.7.2</b>	<b>Aprendizado em neurônios de terceira geração</b> .....	<b>24</b>
2.8	ALGORITMO GERAL PARA SIMULAR REDES NEURAIS .....	25
<b>3</b>	<b>METODOLOGIA</b> .....	<b>26</b>
<b>4</b>	<b>RESULTADOS</b> .....	<b>28</b>
4.1	APROXIMAÇÃO DE FUNÇÃO .....	28
<b>4.1.1</b>	<b>Segunda geração</b> .....	<b>28</b>
<b>4.1.2</b>	<b>Terceira geração</b> .....	<b>28</b>
4.2	PÊNDULO INVERSO .....	29
<b>4.2.1</b>	<b>Segunda geração</b> .....	<b>31</b>
<b>4.2.2</b>	<b>Terceira geração</b> .....	<b>32</b>
4.3	PREVISÃO DO TEMPO .....	33
<b>4.3.1</b>	<b>Segunda geração</b> .....	<b>33</b>
<b>5</b>	<b>CONCLUSÃO</b> .....	<b>38</b>
<b>6</b>	<b>REFERÊNCIAS</b> .....	<b>40</b>
<b>7</b>	<b>APÊNDICE</b> .....	<b>43</b>

# 1 INTRODUÇÃO

O desejo de criar um ser artificial inteligente acompanha a história da civilização humana, desde mitos gregos como Galateia (GALATEIA, 2016) até a variedade enorme de máquinas pensantes presentes na ficção científica como HAL-9000 (2001, 1968) ou R2-D2 (STAR WARS, 1977), mas durante um longo tempo este desejo permaneceu somente na imaginação humana.

O conceito formal de inteligência artificial surgiu em 1950 com Alan Turing (TURING, 1950). Claude Shannon, que então trabalhava na IBM, reuniu (MOOR, 2006) diversos pesquisadores para formalizar os primeiros conceitos de programas inteligentes.

Definir inteligência ainda é debatido pela comunidade científica, cujo conceito evoluiu ao longo do tempo. Entende-se como inteligência artificial um agente interagente capaz de realizar operações que maximizem seu objetivo. As grandes conquistas da I.A. são louvadas em teoria de jogos, como o torneio de xadrez em 1997 do computador Deep Blue contra o russo Kasparov (Campbell, Hoane, Hsu, 2002), ou mais recentemente o torneio de Go com o computador AlphaGo (SILVER et al., 2016) contra o sul-coreano Lee Sedol. Outros destaques estão em computação visual, como classificação e reconhecimento de padrões em imagens (LECUN et al., 1998), ou até mesmo no desenvolvimento de veículos autônomos (MAHMUD, 2012).

Além das áreas computacionais, I.A. pode ser usada para desenvolver novas tecnologias, como antenas (HORNBY et al., 2006), células solares (SABERIAN et al., 2014) e novos materiais (NOSENGO, 2016). Apesar destas grandes contribuições, a área de pesquisa científica permaneceu pouco explorada devido ao alto nível de dificuldade que representa em descobrir e interpretar modelos de realidade física. Além disso, o uso de I.A. em ciência despede, em parte, da compreensão científica, pois o modelo de explicações é escondido do pesquisador, como uma caixa preta que guarda informações. Entretanto, o desenvolvimento de uma inteligência artificial dedicada a produção científica representa uma ferramenta muito poderosa a pesquisadores.

A aplicação de I.A. em ciência começou seu desenvolvimento muito recentemente, como aplicações em transição de fase (TANAKA, TOMIYA, 2016), mecânica quântica (YU, 2015) e dinâmica de fluidos (BAYMANI, KERAYECHAN, EFFATI, 2016).

Rede neural artificial é uma categoria de inteligência artificial inspirada no modelo biológico de neurônios para formar uma rede de conexões. Essa inspiração é baseada no fato de que o comportamento de animais, controlados por seus cérebros, é um comportamento ótimo, isto é, as ações dos indivíduos biológicos são baseadas em processos de otimização. Parte do sucesso da I.A. na última década somente foi possível devido ao investimento em novas técnicas de redes neurais, sendo um bom ponto de partida para o estudo de problemas científicos.

Neste trabalho, foram analisados por inteligência artificial: aproximação de função, pêndulo inverso e previsão do tempo.

A aproximação de função tem como objetivo elucidar o comportamento de redes neurais e demonstrar um teorema matemático chamado de teorema do conexionismo.

O pêndulo inverso é um sistema físico em que uma rede neural deve buscar um solução de equilíbrio, isto exige da I.A. conhecimento de conceitos físicos, como rotação, velocidade e força.

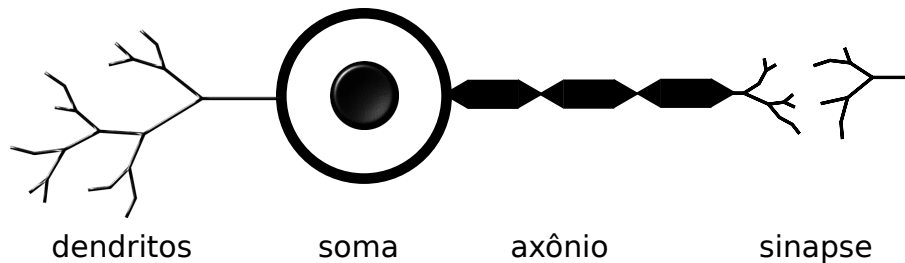
O clima é um sistema muito complicado, pois muitos fatores contribuem para a sua dinâmica. Aplicado pela I.A., este sistema continua um problema em aberto, apesar de alguns resultados indicarem a possibilidade de soluções (ABHISHEK et al., 2012).

## 2 DESENVOLVIMENTO

### 2.1 NEURÔNICO BIOLÓGICO

Um neurônio biológico (representado pela figura 2.1) é composto de um núcleo chamado soma, que processa a informação enviada pelos dendritos (ramificações que recebem informações de outros neurônios). O soma se comunica com outros neurônios propagando informação através do axônio, que conecta um neurônio a outro pela sinapse, estabelecendo uma rede de comunicação unidirecional baseada em impulsos elétricos.

Figura 2.1 – Representação de um neurônio biológico



### 2.2 NEURÔNIO ARTIFICIAL

O comportamento biológico dos neurônios pode ser mimetizado pelo neurônio artificial (figura 2.2), cuja forma mais simples pode ser representada por uma operação matemática não linear sobre informações recebidas. Esta operação é chamada função ativação. Formalmente diz-se que o neurônio mapeia a informação recebida para um espaço de informações esperado. Um neurônio artificial pode ser representado pictoricamente por um círculo com ramificações, que é a unidade básica de um grafo. Em geral, um dendrito artificial realiza uma operação linear chamada peso  $w$ . O dendrito multiplica a informação recebida por um valor, ou conjunto de valores (pesos) antes de ser mapeada pela operação não linear de um neurônio, equivalente a um neurônio biológico, cujo dendritos podem facilitar ou dificultar a passagem de informação.

Um conjunto de vários neurônios artificiais ligados cria um grafo chamado rede neural artificial, de onde emerge a inteligência artificial, como veremos mais tarde. Uma rede neural artificial pode ser representada pictoricamente por um conjunto de neurônios interligados (figura 2.3).

Geralmente a informação, desta forma representada, flui da esquerda para a direita. Independentemente da operação realizada, todos os neurônios, obrigatoriamente, realizam opera-

Figura 2.2 – Representação de um neurônio artificial.  $x$  representa as informações de entrada,  $y$  é a informação de saída e  $f$  é a operação não linear

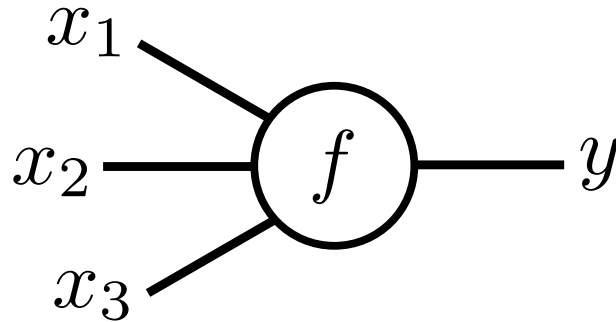
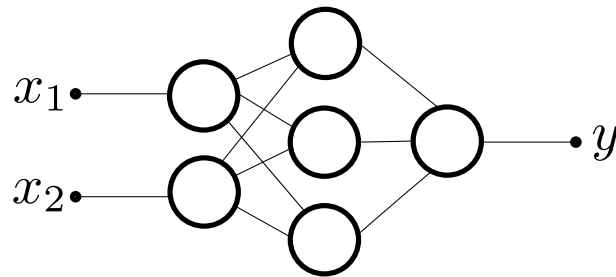


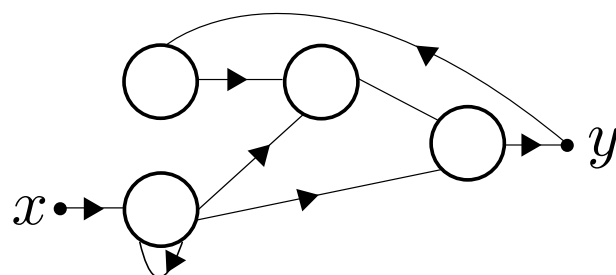
Figura 2.3 – Exemplo de uma possível rede neural



ções não lineares, caso contrário, seria possível reduzir uma rede neural por sucessivas operações lineares que seriam equivalentes a um neurônio apenas. Aqui alguns nomes são importantes de serem mencionados: cada coluna do grafo representa uma camada da rede, a primeira e última camada são chamadas de camada de entrada e camada de saída respectivamente. Todas as outras camadas são conhecidas como camadas ocultas.

Redes neurais separadas em camadas com conexões diretas simples são chamadas de rede de propagação direta. Alguns autores preferem chamar estas redes neurais de aprendizagem profunda (deep learning), onde o nível de profundidade está associada ao número de camadas ocultas (hidden layers). Assim como os neurônios, as redes neurais podem possuir muitas formas (figura 2.4). Estas formas variam de programador para programador, dependendo da técnica, criatividade e necessidade de cada um.

Figura 2.4 – Exemplo de uma rede neural sofisticada





### 2.3 TEORIA DE CONEXIONISMO

Também chamado de teorema da aproximação universal (HORNIK, 1991), a teoria do conexionismo estabelece que uma rede neural artificial consegue aproximar qualquer função contínua para uma quantidade suficiente de neurônios. Argumenta-se que este teorema é responsável por dar origem a inteligência, seja biológica ou artificial.

### 2.4 NEURÔNIOS DE PRIMEIRA GERAÇÃO

Os primeiros neurônios artificiais desenvolvidos para simulação computacional são chamados atualmente de neurônios de primeira geração, que consistem de funções ativação não contínuas, estes neurônios são denominados perceptron e obedecem o seguinte comportamento:

$$y(x) = \begin{cases} 1 & \text{se } wx + b < \theta \\ 0 & \text{caso contrário} \end{cases} \quad (2.1)$$

Onde  $b$  é chamado viés<sup>1</sup> e  $\theta$  é um valor limite fixo escolhido pelo programador de acordo com o problema a ser resolvido.

### 2.5 NEURÔNIOS DE SEGUNDA GERAÇÃO

Os neurônios de segunda geração são caracterizados por funções ativação contínuas. Esta geração representa o foco de aplicabilidade atual e é muito estudada pela ciência computacional.

Entre as funções ativação mais populares, destacam-se o neurônio sigmóide, que realiza o seguinte mapa:

$$y(x) = \sigma(x) = \frac{1}{1 + e^{-wx}} \quad (2.2)$$

O neurônio softplus:

$$y(x) = \text{softplus}(x) = \ln(1 + e^{wx}) \quad (2.3)$$

O neurônio softmax:

$$y(x) = \text{softmax}(x) = \frac{e^{wx_j}}{\sum_j e^{wx_j}} \quad (2.4)$$

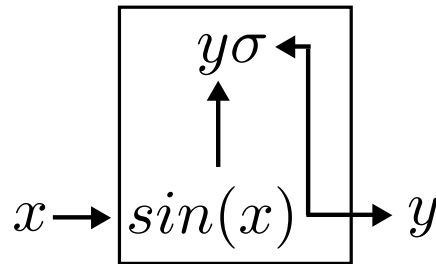
Entretanto, a complexidade de um neurônio pode ser bastante elevada, como mostrado

---

<sup>1</sup>viés é um valor optativo para que a função ativação não seja nula em  $x=0$ .

pela figura 2.5. Que considerada uma série de operações lineares e não lineares, onde o fluxo

Figura 2.5 – Exemplo de um neurônio de várias operações, onde  $y\sigma$  representa o produto de  $y$  e  $\sigma$



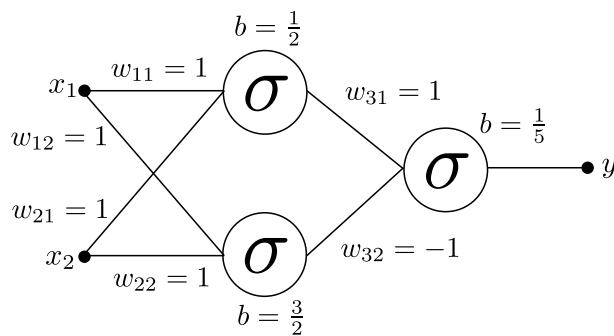
de informação pode ter mais de uma direção.

Como exemplo, a figura 2.6 mostra uma rede neural que aproxima a função XOR (ou exclusivo), um problema clássico de aproximação:

Quadro 2.1 – Função XOR

$x_1$	$x_2$	XOR
0	0	0
1	0	1
0	1	1
1	1	0

Figura 2.6 – Rede neural que aproxima a função XOR



Operacionalmente, a rede neural da figura 6 pode ser escrita como:

$$s_1 = \sigma(x_1 + x_2 + \frac{1}{2}) \tag{2.5}$$

$$s_2 = \sigma(x_1 + x_2 + \frac{3}{2}) \tag{2.6}$$

$$y = \sigma(s_1 + s_2 + \frac{1}{5}) \tag{2.7}$$

Devido ao comportamento sigmóide dos neurônios, os valores de  $y$  para os dados de entrada XOR estão centrados ao redor de 0.5, para corrigir isso, basta adicionar a condição:

$$XOR = \begin{cases} 0 & \text{se } y < 0.5 \\ 1 & \text{caso contrário} \end{cases} \quad (2.8)$$

## 2.6 PARADIGMAS DE APRENDIZADO

Ao iniciar uma rede neural aplicada em um certo sistema, os valores peso não são conhecidos. Existem muitas formas dos pesos serem obtidos, que dependem do paradigma de aprendizado da rede neural: aprendizado supervisionado, aprendizado reforçado e aprendizado não supervisionado. Os três paradigmas são igualmente úteis, pois dependem do problema a ser resolvido, entretanto, em algumas vezes, também podem ser usados em conjunto (Silver, D. et al., 2016).

### 2.6.1 Treinamento supervisionado

No caso do aprendizado supervisionado, buscam-se valores dos pesos de acordo com exemplos fornecidos. De certa forma, o comportamento do sistema é ensinado a rede neural, que então aprende os padrões e os aplica em casos do sistema que não estão contidos nos exemplos. Este paradigma é usado quando o sistema possui exemplos fornecidos e deseja-se fazer novas previsões. Em geral, o que se faz é buscar métodos de otimização que tentam minimizar ou maximizar certos parâmetros da rede neural que fornecem pesos ótimos.

Muitas vezes é útil associar à rede neural uma função chamada função custo que depende da informação produzida pela rede neural e seu valor esperado. Esta função pode assumir muitas formas, as mais utilizadas são:

-Custo quadrático:

$$C = \sum_j \frac{1}{2} (y_j - t_j)^2 \quad (2.9)$$

-Entropia binária cruzada:

$$C = - \sum_j \frac{1}{N} [t_j \ln(y_j) + (1 - t_j) \ln(1 - y_j)] \quad (2.10)$$

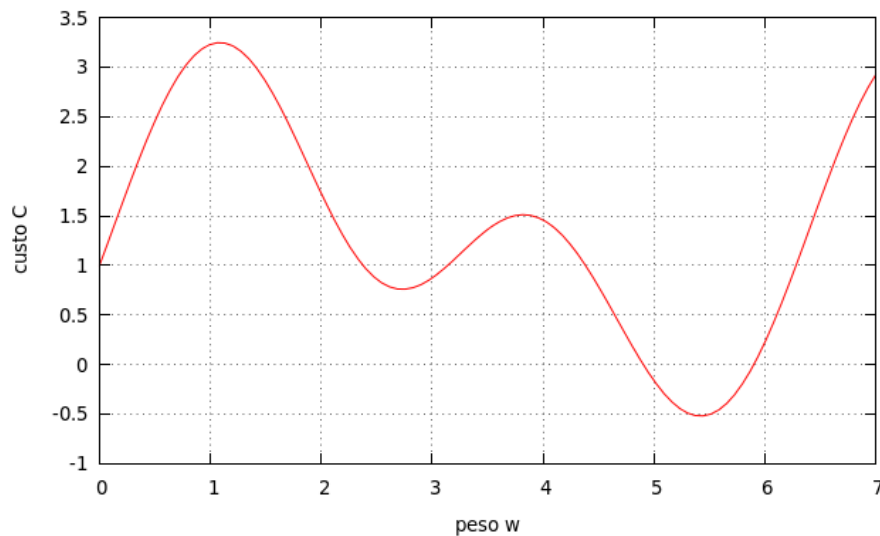
-Custo exponencial:

$$C = \tau e^{\frac{1}{\tau} \sum_j (y_j - t_j)^2} \quad (2.11)$$

Onde a soma ocorre sobre os exemplos fornecidos, sendo  $t$  o valor esperado,  $N$  o número de exemplos e  $\tau$  um parâmetro de controle fixo.

Para qualquer função custo, existe uma relação entre o custo e o valor do peso, como pode ser representado, de forma geral, na figura 2.7. Ou seja, existe um valor do peso cujo custo é mínimo. Portanto, o problema passa a ser minimizar a função custo em relação ao peso.

Figura 2.7 – Exemplo de custo em relação ao peso



### 2.6.1.1 Técnicas de minimização

Como mencionado anteriormente, encontrar os valores peso depende de um processo de minimização, as técnicas mais comuns são:

-Método do gradiente

$$w_n = w_{n-1} - \eta \frac{\partial C}{\partial w} \quad (2.12)$$

O valor  $w$  é atualizado iterativamente pela derivada, sendo  $\eta$  um parâmetro chamado velocidade de aprendizado.

-Algoritmo genético

O algoritmo genético consiste em evoluir várias redes neurais com pesos diferentes através de um processo de seleção. Este algoritmo consiste em:

- 1) Criar um conjunto de redes neurais (indivíduos) de pesos diferentes;
- 2) Selecionar indivíduos com menor função custo;
- 3) Gerar novos indivíduos com pesos obtidos a partir de alguma estatística escolhida (e.g. mé-

dia aritmética) das redes neurais selecionadas<sup>2</sup>;

4) Repetir passo 2 até algum critério (e.g. pequena variação da função custo entre as gerações) ser obedecido.

### 2.6.1.2 Redes neurais de várias camadas

Para redes neurais com mais de uma camada, faz-se necessário técnicas adicionais, pois os valores esperados das camadas anteriores a última não são conhecidos, a técnica mais comumente usada é chamada retropropagação (backpropagation).

Em retropropagação os erros obtidos da função custo são propagados no sentido contrário ao fluxo de informações da rede, onde os pesos internos são atualizados de acordo com a intensidade destes erros, chamado de deltas.

Para a camada anterior a última n:

$$\frac{\partial C}{\partial w_{n-1}} = \frac{\partial C}{\partial y_{n-1}} \frac{\partial y_{n-1}}{\partial w_{n-1}} \quad (2.13)$$

De forma mais geral, para camadas anteriores a última:

$$\delta_i = \frac{\partial y}{\partial y_i} \quad (2.14)$$

$$\frac{\partial C}{\partial w_j} = \frac{\partial C}{\partial y_n} \prod_{i=n-1}^{i=j+1} \delta_i \frac{\partial y_j}{\partial w_j} \quad (2.15)$$

Retropropagação pode usar o método do gradiente para atualizar o peso  $w_i$  através da variação do custo em relação ao próprio  $w_i$ .

Um problema comum de retropropagação são os deltas diminuírem a medida que se propagam pela rede, para resolver este problema, geralmente se implementa uma taxa de aprendizado adaptativo ou uma arquitetura diferente da propagação direta.

#### -Taxa de aprendizado adaptativo

Ao invés de escolher uma taxa de aprendizado fixo, varia-se esta taxa com o tempo, por exemplo, usando arrefecimento<sup>3</sup>:

$$\eta(t) = \frac{\eta(0)}{1 + t/T} \quad (2.16)$$

<sup>2</sup>Para reduzir o custo computacional do algoritmo genético (já que requer a produção de muitas gerações) é comum evoluir as redes neurais antes do processo de cruzamento.

<sup>3</sup>O nome arrefecimento provem da ciência de materiais, onde é a variação de temperatura é usada para alterar propriedades físico-químicas de materiais. De forma análoga, pode-se usar o tempo para “esfriar” a taxa de aprendizado, deixando a convergência para uma solução melhor comportada.

Sendo  $t$  a iteração evolutiva e  $T$  o número total de iterações.

#### -Convolução

Ao invés de construir uma rede neural com uma arquitetura de propagação direta, que possui conexões completa entre as camadas, é possível fazer algumas considerações para reduzir o número de pesos a serem atualizados. A convolução consiste em agrupar conexões de neurônios entre camadas por grupos. Por exemplo, em um processo de classificação de imagens, a convolução cria grupos de neurônios onde cada grupo analisa uma parte da imagem apenas. Esta técnica já demonstrou ser bastante efetiva quando comparada com propagação direta (MATUSUGU et al., 2003), além de custo computacional ser efetivamente menor.

#### -Rede neural recorrente

Esta arquitetura de rede neural adiciona memória, i.e., a capacidade dos neurônios não esquecerem o processo evolutivo que foram submetidos. A função ativação é modificada da seguinte forma:

$$y_n = f(wx_n + uy_{n-1}) \quad (2.17)$$

Onde  $u$  é um peso especial que relaciona a evolução do neurônio com sua capacidade de memorizar o processo evolutivo. Este peso  $u$  pode seguir as mesmas regras de processo evolutivo que o peso  $w$ . Esta técnica mostrou ser eficaz especialmente em problemas de análise verbal (GRAVES et al., 2006).

Alguns autores (STORK, 1989) argumentam que retropropagação não é compatível com um modelo realista de uma rede neural biológica. Embora esta argumentação apresenta evidência experimental, retropropagação ainda assim representa uma das melhores formas de atualizar os pesos neurais. Esta divergência entre redes biológicas e redes artificiais não é incomum. Contudo, a rede neural representa apenas uma aproximação de organismos naturais, não se excedendo em comparações ou imitações. Tal temática possui diferentes perspectivas, que variam de acordo com a necessidade do programador: aqueles que pesquisam rede neural com justificativa biológica e aqueles que que a pesquisam com justificativa matemática no conexionismo. Ambas pesquisas são válidas da perspectiva pragmática, pois contribuem para o progresso da inteligência artificial.

### 2.6.2 Treinamento reforçado

No paradigma do treinamento reforçado, a rede neural já possui conhecimento do sistema, entretanto não há um valor esperado absoluto. Um exemplo clássico é o jogo xadrez, onde pode haver jogadas igualmente favoráveis (ou desfavoráveis), então a função custo é descartada

por uma função que maximize o objetivo (o quão próximo da vitória uma jogada representa).

Neste tipo de treinamento, a rede neural considera vários estados  $R$  (soluções) para um problema, onde cada estado possui recompensas ou punições. A evolução da rede neural, quantificada pelo fator  $Q$ , ocorre quando a rede neural visita os estados de maior recompensa e evita aqueles de maior punição. O algoritmo Aprendizado-Q (AZAR et al., 2011) possui a seguinte evolução:

$$Q_{estado} = R_{estados} + \mu \max(Q_{proximosestados}) \quad (2.18)$$

$$w_n = w_{n-1} + \eta(Q_n - Q_{n-1}) \quad (2.19)$$

### 2.6.3 Treinamento não supervisionado

Por último, no paradigma de treinamento não supervisionado, uma rede neural não possui exemplos de teste nem objetivo conhecido. Este paradigma procura encontrar padrões entre características dos componentes de um sistema por similaridade. O algoritmo de Kohonen (KOHONEN, 1982) é um algoritmo comum aplicado a este paradigma. Os neurônios com pesos mais adequados são atualizados por:

$$w_n = w_{n-1} + \mu h(x - w_{n-1}) \quad (2.20)$$

Com  $h$  uma função de cooperação entre neurônios vizinhos  $i$  e  $j$ . Por exemplo:

$$h = e^{-(r_i - r_j)} \quad (2.21)$$

Com  $r$  a posição topológica do neurônio.

Essencialmente o mapa de Kohonen compete os neurônios e atualiza o peso do neurônio que possui ativação que caracteriza maior semelhança com os dados de entrada.

## 2.7 NEURÔNIOS DE TERCEIRA GERAÇÃO

As divergências entre redes biológicas e neurônios de segunda geração motivaram a modelagem de uma terceira geração de neurônios, que considera o comportamento biológico de forma mais realista.

Experimentos (OBEL et al., 2012) revelam que em um cérebro orgânico, os impulsos (correntes elétricas de curta duração) recebidos por um neurônio pelos dendritos, devido a algum estímulo, aumenta sua voltagem subitamente, então o neurônio emite um pulso elétrico através do axônio. O axônio amplifica o pulso até a sinapse, que representa o fim do axônio,

onde o pulso é copiado para os vários dendritos de outros neurônios. Essa cópia acontece pelas moléculas neurotransmissoras que carregam o sinal através do meio interneural até um neuroreceptor adequado, que então ativa um ciclo de sódio/potássio alterando a diferença de potencial no soma dos neurônios que receberam o pulso, iniciando o ciclo novamente.

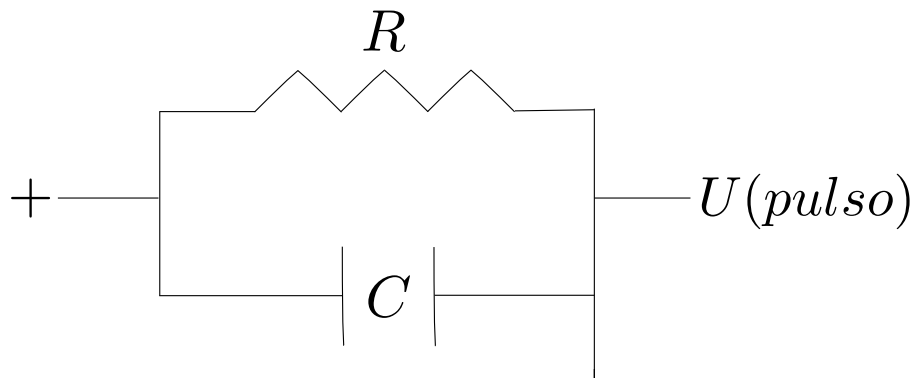
### 2.7.1 Modelos de neurônios de terceira geração

A principal característica da terceira geração de neurônios artificiais é a presença explícita do tempo, que insere o modelo de pulsos de informação baseados em redes naturais.

-Modelo integra-e-dispara (ABBOTT, 1999):

Neste modelo uma rede neural é aproximada por um circuito elétrico com resistores e capacitores:

Figura 2.8 – Circuito elétrico equivalente do neurônio integra-e-dispara



Onde cada capacitor do circuito equivale a um neurônio. O comportamento deste circuito possui duas etapas: Quando a diferença de potencial do capacitor está abaixo de um certo valor limite, na ausência de corrente, a voltagem do capacitor é amortecida com o tempo devido a presença do resistor. Na presença de corrente, quando o capacitor atinge uma voltagem limite, o capacitor emite pulsos e imediatamente volta a ter uma diferença de potencial nominal.

Pelo diagrama do circuito, podemos obter as equações do modelo. A corrente do circuito é dada por:

$$I(t) = C \frac{\partial U(t)}{\partial t} + \frac{1}{R} U(t) \quad (2.22)$$

A equação acima é integrada de acordo com condições específicas do tempo de resposta de um neurônio, i.e., quando a voltagem atinge um valor limite  $U_l$ , esta é restaurada a uma tensão específica  $U_r$ .

Em uma versão mais sofisticada, este modelo inclui um período de tempo, chamado de

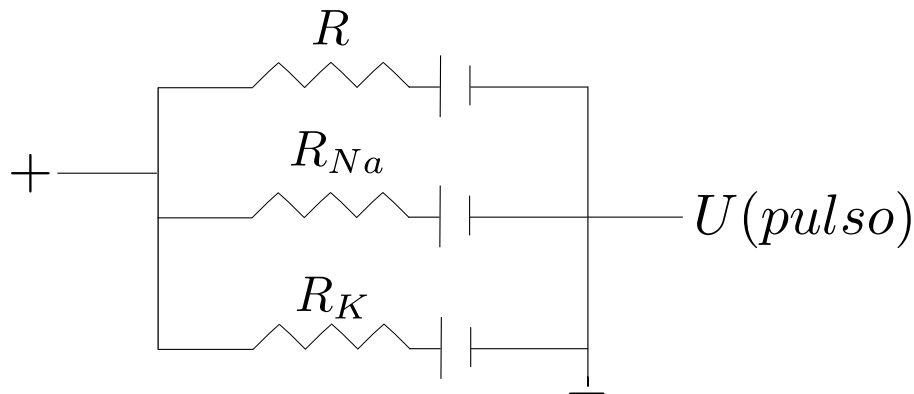


período refratório, onde o neurônio, após emitir pulsos, permanece por este período em estado de repouso.

- Modelo Hodgkin-Huxley (HODGKIN, HUXLEY, 1952):

Este modelo, elaborado em 1952, é considerado um dos mais completos modelos para redes neurais artificiais, possuindo grande compatibilidade com observações experimentais, pois foi construído a partir do conhecimento biofísico obtido em experimentos com redes neurais orgânicas de lulas. Assim como no modelo anterior, este pode ser representado por um circuito elétrico, com baterias como componente adicional.

Figura 2.9 – Circuito elétrico equivalente do neurônio de Hodgkin-Huxley



A corrente obtida do diagrama é:

$$I(t) = \sum_j I_j(t) \quad (2.23)$$

Cada  $I_j$  é a corrente associada a cada canal  $j$  do circuito. Um canal é definido como uma associação em série dos componentes eletrônicos, como um resistor e uma fonte.

A equação da diferença de potencial do neurônio é:

$$C \frac{\partial U(t)}{\partial t} = - \sum_j I_j(t) \quad (2.24)$$

Os canais são controlados por parâmetros empíricos tais que  $I(t)$  obedeça:

$$\sum_j I_j(t) = g_{Na} m^3 h (U - E_{Na}) + g_K n^4 (U - E_K) + g_L (U - E_L) \quad (2.25)$$

Onde as funções  $g$  representam a condutância (inverso da resistência) de cada canal.

As variáveis  $m$ ,  $n$  e  $h$  obedecem as equações diferenciais:

$$\frac{\partial m}{\partial t} = \alpha_m(1 - m) - \beta_m m \quad (2.26)$$

$$\frac{\partial n}{\partial t} = \alpha_n(1 - n) - \beta_n n \quad (2.27)$$

$$\frac{\partial h}{\partial t} = \alpha_h(1 - h) - \beta_h h \quad (2.28)$$

As funções  $\alpha$  e  $\beta$  são funções empíricas que dependem da diferença de potencial e foram obtidas através de observações experimentais.

Embora realístico, este modelo possui elevado custo computacional devido a complexidade das equações diferenciais. Desde que foi publicado, novos modelos inspirados neste foram desenvolvidos, com objetivo de simplificá-lo para reduzir o custo computacional. Uma versão popular é o modelo de Izhikevich.

- Modelo Izhikevich (IZHIKEVICH, 2003):

Este modelo reduz as quatro equações diferenciais em apenas duas equações de voltagem usando teoria de bifurcação:

$$\frac{\partial V}{\partial t} = 0,04V^2 + 5V + 140 - U + I \quad (2.29)$$

$$\frac{\partial U}{\partial t} = a(bV - U) \quad (2.30)$$

As voltagens  $U$  e  $V$  estão sujeitas à condição:

$$se \quad V \geq 30 \quad \text{então} \quad \begin{cases} V = c \\ U = U + d \end{cases} \quad (2.31)$$

Com  $a, b, c$  e  $d$  parâmetros fixos que determinam o comportamento dos pulsos, como frequência de pulso e receptividade de sinal. Os números presentes nas equações são consequência da aproximação do modelo e tem esta forma devido a razões biológicas (por exemplo, um neurônio biológico possui, em média, potencial máximo de 30 mV, justificando a condição  $V \geq 30$ ). A vantagem desse modelo consiste em seu baixo custo computacional (comparado com o modelo de Hodgkin-Huxley) e preservação do realismo biológico.

## 2.7.2 Aprendizado em neurônios de terceira geração

Embora existam vários modelos de aprendizado em neurônios de terceira geração, um dos que possui maior realismo biológico é a Plasticidade dependente de pulsos temporais (STDP)

que considera diferenças temporais de pulsos para atualização do peso.

STDP considera dois processos, depressão de longo prazo (LTD) e potenciação de longo prazo (LTP). LTD deprime a frequência em que um neurônio dispara pulsos, ao contrário da LTP, que a estimula.

O processo de aprendizagem da STDP atualiza o pesos da seguinte maneira: se um nerônio recebe um pulso de outro neurônio anterior ou um conjunto de neurônios anteriores, e se este sinal for forte suficiente, o neurônio irá emitir um pulso. A diferença de tempo entre o pulso de chegada e o pulso de saída determina o processo da STDP. Se esta diferença de tempo for positivo o processo será LTP, caso contrário será LTD. A equação 2.32 descreve estes processos:

$$\Delta w = \begin{cases} \eta e^{-|t_j - t_i|} & \text{se } LTP \\ -\eta \gamma e^{-|t_j - t_i|} & \text{se } LTD \end{cases} \quad (2.32)$$

Sendo  $t_i$  e  $t_j$  os neurônios que emitem e recebem o pulso respectivamente,  $\eta$  a taxa de aprendizagem e  $\gamma$  a taxa em quanto um neurônio deve ser depremido em relação a potenciação.

## 2.8 ALGORITMO GERAL PARA SIMULAR REDES NEURAIIS

Esta sessão apresenta como uma rede neural pode ser simulada de forma geral, dividida em etapas:

- 1) Escolha uma arquitetura para a rede neural;
- 2) Determine a geração e a função ativação dos neurônios, é possível misturar vários tipos;
- 3) Opte por uma técnica de aprendizado que condiz com o tipo de neurônio e dados disponíveis para treino (por exemplo, se há valores esperados então recomenda-se aprendizado supervisionado);
- 4) Determine os valores dos hiperparâmetros (se a técnica escolhida for o algoritmo genético, então os hiperparâmetros podem ser inicializados de forma aleatória);
- 5) Distribua aos pesos valores aleatórios;
- 6) Inicie o processo de treinamento: os pesos devem ser atualizados de acordo com a técnica escolhida, fundamentalmente a função erro deve ser minimizada ou a função objetivo maximizada;
- 7) Continue atualizando os pesos até que alguma condição de parada seja obedecida (por exemplo, o erro deve diminuir até um certo valor);
- 8) Após o aprendizado estar completo, a rede neural pode ser usada para resolver qualquer problema da categoria em que foi treinada (os pesos não são mais atualizados).

Estes passos representam apenas orientações de como uma rede neural pode ser usada. Para cada problema vai haver redes neurais melhores que outras, determinar a estrutura e técnica ideal depende da experiência do programador.

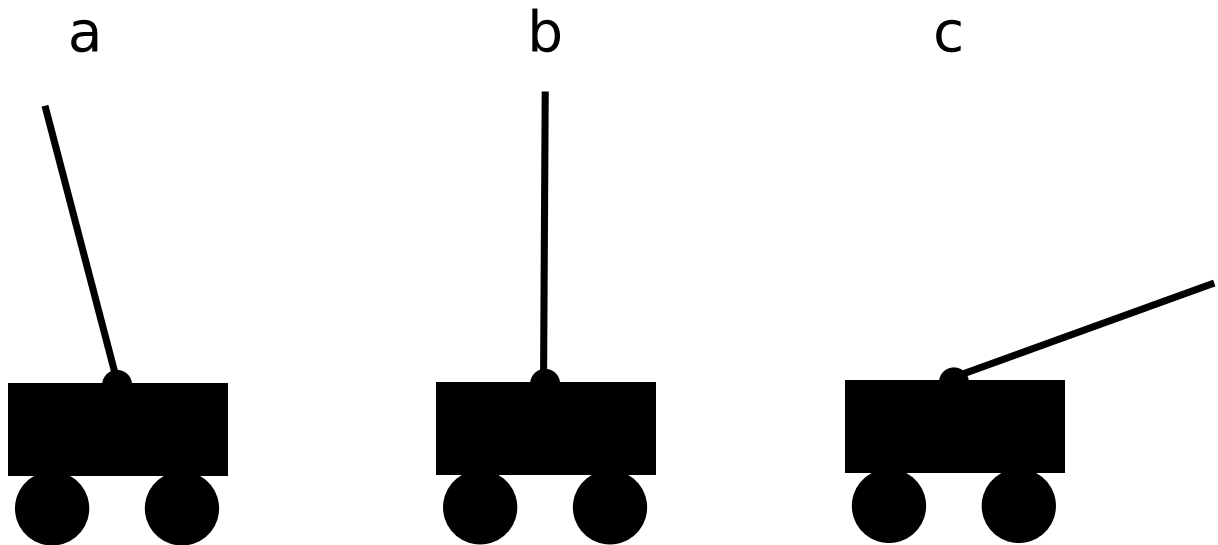
### 3 METODOLOGIA

Três sistemas foram resolvidos por inteligência artificial.

O primeiro problema consiste em aproximar uma função analítica, para evidenciar o teorema do conexionismo. A função analítica  $\sin^2(x)$  foi escolhida por ser uma função simples e ajudar melhor na compreensão humana de redes neurais. Para a segunda geração foi aplicado o método do gradiente com aprendizado supervisionado.

O segundo problema consiste em encontrar o ponto de equilíbrio de um pêndulo inverso (ANDERSON, 1989). O pêndulo inverso é um sistema físico que consiste em pêndulo, livre para girar fixo em uma plataforma horizontal móvel, sobre a presença de um campo gravitacional vertical (Figura 3.1). Nas condições iniciais, o pêndulo é posto na vertical em uma situação de equilíbrio instável. Para manter o pêndulo em equilíbrio, é necessário aplicar uma força na plataforma a fim de descolá-la horizontalmente. O problema é resolvido se for possível manter o pêndulo em equilíbrio sobre um prolongado período de tempo.

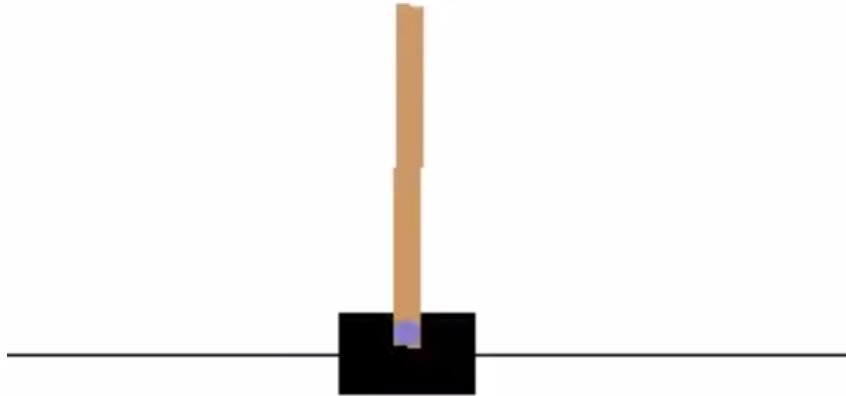
Figura 3.1 – Figura representando três condições do pêndulo invertido. (a) e (c) fora do equilíbrio. (b) no equilíbrio instável



Para simular o pêndulo inverso, foi usado a plataforma digital OpeanAI Gym (OPENAI, 2016), que consiste em um ambiente virtual para simulação de cenários físicos e jogos, chamados de ambiente, para treinar e testar agentes artificiais inteligentes. O ambiente pêndulo físico (Figura 3.2) simula o problema do pêndulo sobre condições similares as condições físicas na superfície da Terra.

A rede neural de segunda geração usou o método do gradiente com aprendizado reforçado, onde o estado ótimo é o estado de equilíbrio instável. A rede neural de terceira geração usou o modelo Izhikevich, com parâmetros fixos  $a=0,02$ ,  $b=0,2$ ,  $c=-65$  e  $d=8$ .

Figura 3.2 – Captural de tela do ambiente virtual para o equilíbrio instável do pêndulo inverso



O terceiro problema consiste em previsão do tempo. Este sistema foi escolhido devido a sua complexidade elevada e quantidade de dados disponíveis, além do interesse socioeconômico que representa.

Os dados de entrada são contituídos pelos parâmetros físicos obseráveis: temperatura, humidade, radiação solar, precipitação e intensidade do vento. O objetivo principal é desenvolver uma rede neural que consiga fazer previsão do tempo no futuro a partir dos dados históricos.

Os dados usados na rede neural são da região de Santa Maria com base em sensores meteorológicos presentes no aeroporto de Santa Maria, cujos dados estão publicamente disponíveis (WWO, 2016). Estes dados foram separados em duas categorias: dados de entrada, que constituem os parâmetros usados de acesso a rede neural e dados de teste, que constituem os parâmetros que serão comparados com os resultados das redes neurais para assim testar sua qualidade de previsão.

Todos os problemas foram resolvidos usando linguagem Python.

## 4 RESULTADOS

Nesta sessão são apresentados os resultados para os problemas discutidos na metodologia.

### 4.1 APROXIMAÇÃO DE FUNÇÃO

Um problema simples que redes neurais podem resolver é aproximar funções. De acordo o teorema do conexionismo, uma rede neural podem aproximar qualquer função analítica arbitrariamente bem, isto é, quanto maior o tempo de treinamento e número de neurônios, melhor será a qualidade da aproximação. Esta técnica é análoga a aproximações em série, como série de Taylor ou de Fourier.

A rede neural usada nesta aproximação consiste de duas camadas de propagação direta, cada camada é constituída de 40 neurônios, sendo cada neurônio da camada entrada associado a um ponto da função a ser aproximada. Portanto, devido a sua arquitetura, há  $40^2$  conexões entre os neurônios.

#### 4.1.1 Segunda geração

A figura 4.1 mostra a aproximação da função em relação ao período de treinamento. Quanto maior o período de treinamento melhor é a qualidade da aproximação.

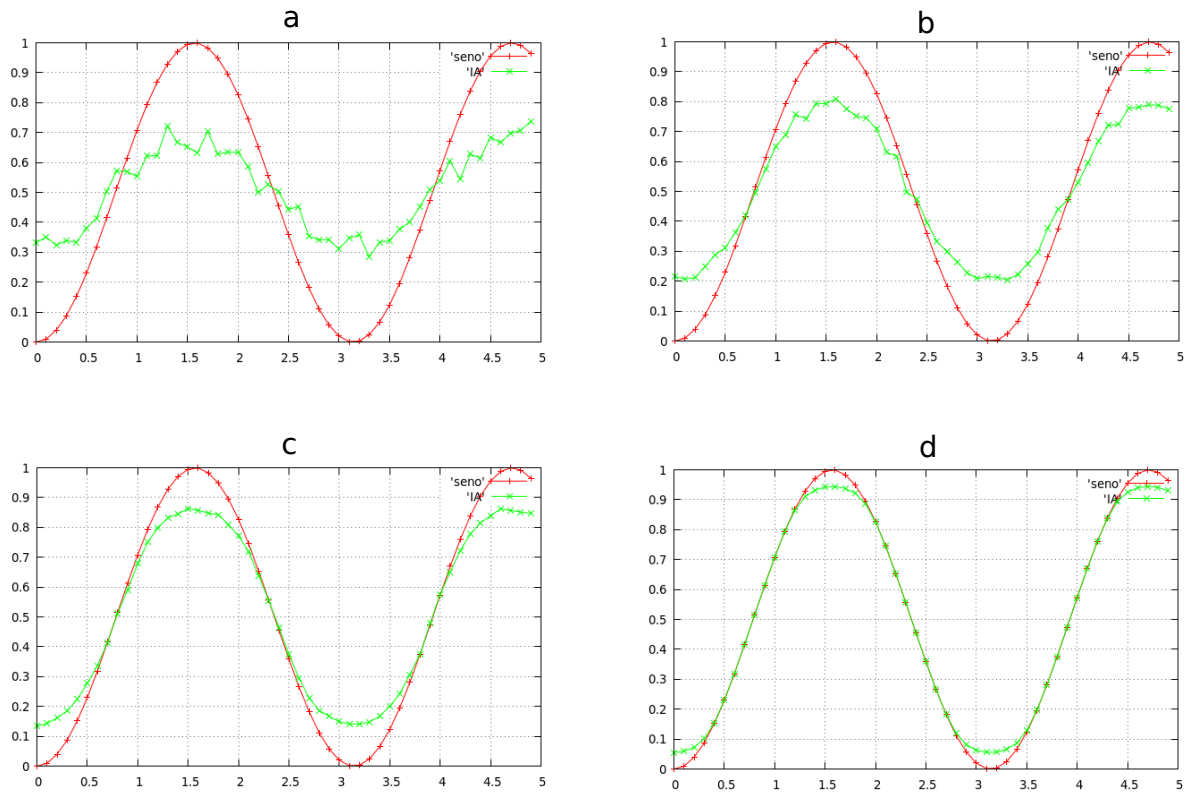
A qualidade da aproximação também pode ser contabilizada pelo função erro: soma da diferença absoluta ponto a ponto entre o valor esperado e o valor produzido pela rede neural, como mostrado pela figura 4.2.

#### 4.1.2 Terceira geração

A função  $y = \sin^2(x)$  foi transformada em um sinal S da forma  $S = 100y$ , isto é, S é fornecido para cada neurônio como corrente. Esse sinal foi alimentado a rede neural. A figura 4.3 mostra a relação entre os neurônios e o tempo ao qual emitiram um pulso, sendo que a cor dos pontos indica a frequência de pulsos, fundamentalmente esta figura mostra a codificação da função fornecida.

Para a aproximação da função, é necessário decodificar (transformar o mapa de informações de volta em sinal) a forma em como a rede neural armazena informações. Existem duas maneiras de decodificação: por tempo ou por frequência.

Figura 4.1 – Aproximação de  $\sin^2(x)$  para vários períodos de treinamento. (a) 5 iterações, (b) 10 iterações, (c) 20 iterações, (d) 100 iterações



Na decodificação por tempo, a frequência dos sinais é irrelevante, o que importa é o tempo em que ocorre os pulsos. Por isto, consideramos os pulsos emitidos em uma frequência específica, por exemplo a mais baixa, obtemos a figura 4.4.

Quanto maior a intensidade do sinal, menor é o tempo em que um neurônio irá emitir um pulso, portanto o sinal é inversamente proporcional ao tempo do pulso. Ao dividir o período dos neurônios para o período da função, inverter e normalizar o tempo de cada pulso, obtém-se a figura 4.5.

Na decodificação por frequência, o tempo em que ocorreu um pulso é irrelevante, o importante é quantos pulsos os neurônios emitiram. A figura 4.6-a mostra a relação entre os neurônios e suas frequências de pulsos. Ao normalizar a frequência e ajustar o período de neurônios, ocorre a decodificação do sinal, como mostrado na figura 4.6-b

## 4.2 PÊNDULO INVERSO

Para o pêndulo inverso, foi construída uma rede neural de três camadas de propagação direta. A camada de entrada consiste de 4 quatro neurônios, cada parâmetro associado a um

Figura 4.2 – Função erro em relação ao período de treinamento

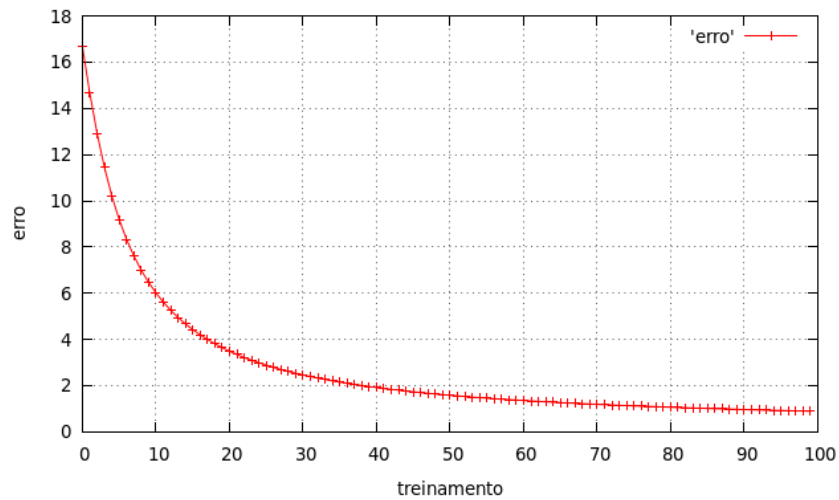
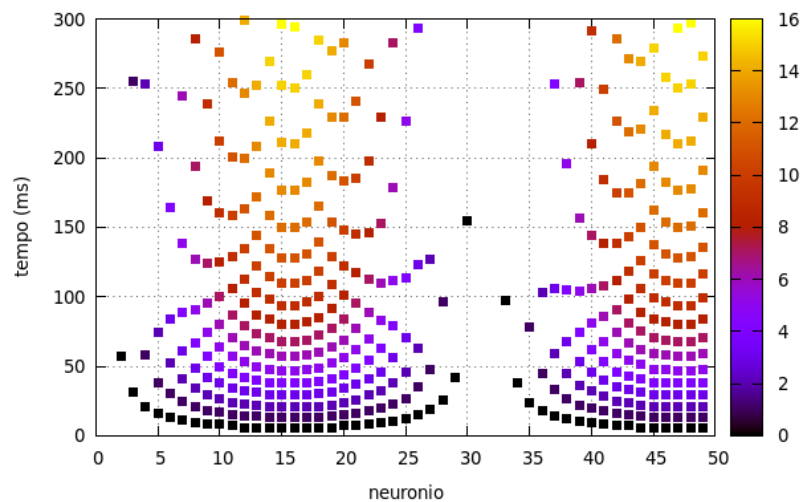


Figura 4.3 – Codificação do função  $y = \text{sen}^2(x)$  pela rede neural



parâmetro observável: posição da plataforma, velocidade da plataforma, ângulo do pêndulo e velocidade do pêndulo. A segunda camada consiste de 100 neurônios e a terceira camada consiste de dois neurônios, cada neurônio associado a força que a I.A. aplica sobre a plataforma para deslocá-la: força pela esquerda ou pela direita.

O sistema foi treinado sobre fases. Uma fase consiste na observação do pêndulo e interação na plataforma móvel, se o pêndulo sair de uma certa faixa de ângulos (maior que 10 graus) ou a plataforma for movida para distâncias grandes, então o sistema é reiniciado e começa uma nova fase. Cada fase corresponde uma iteração de treinamento.



Figura 4.4 – Tempo de pulso para cada neurônio na frequência mais baixa

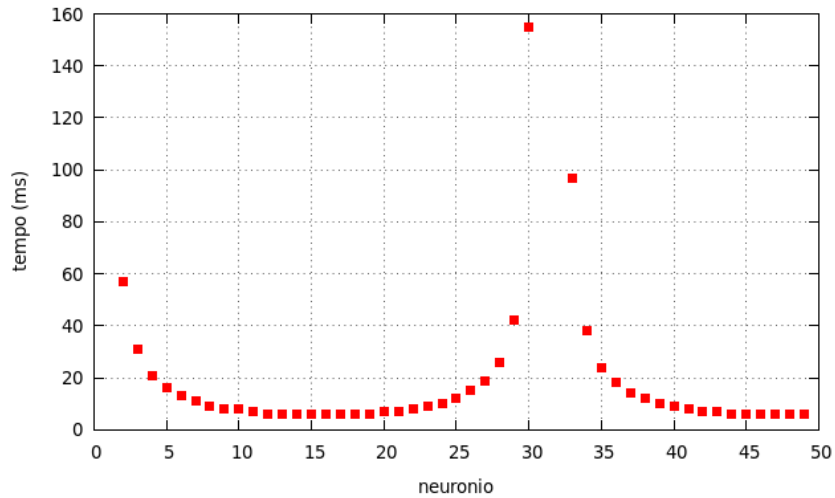
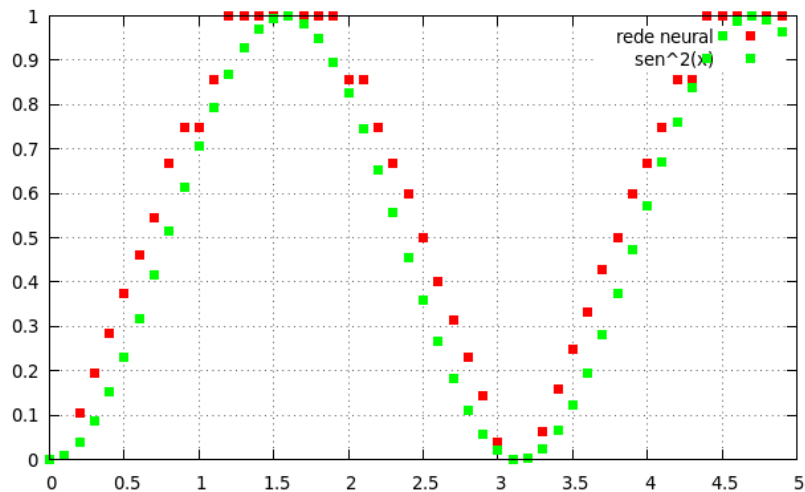


Figura 4.5 – Decodificação por tempo da rede neural



### 4.2.1 Segunda geração

Para o modelo de segunda geração, a figura 4.7 mostra o tempo que a rede neural consegue manter o pêndulo no equilíbrio instável, a medida que o período de treinamento foi aumentando, maior foi o tempo de equilíbrio. Perto de 100 iterações de treinamento, a rede neural consegue elevar o período de equilíbrio consideravelmente, neste ponto pode ser considerado que a rede neural aprendeu a resolver o problema.

Figura 4.6 – (a) Frequência de pulsos para cada neurônio. (b) Decodificação por frequência da rede neural

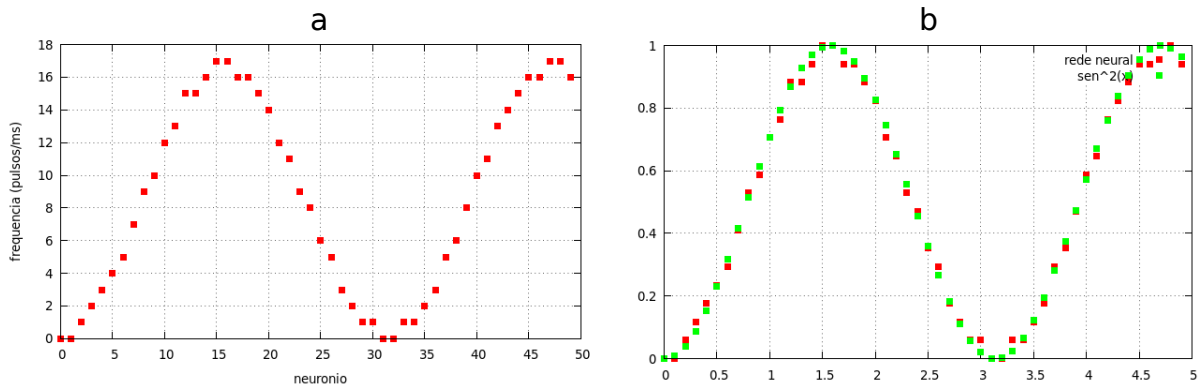
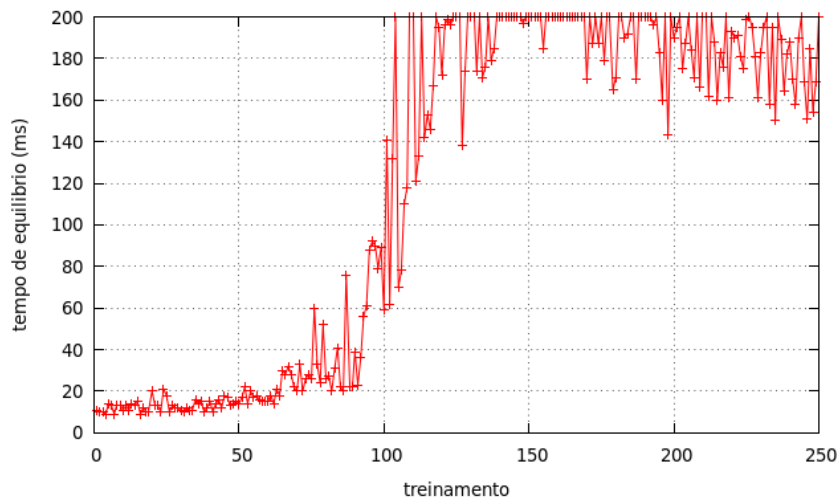


Figura 4.7 – Tempo de equilíbrio instável em relação ao período de treinamento



## 4.2.2 Terceira geração

A figura 4.8 mostra o tempo de equilíbrio para uma rede neural de terceira geração de mesmo tamanho e arquitetura da segunda geração usada.

O resultado inferior, comparado com a segunda geração, sugere uma complexidade maior, devido a seu número de parâmetros elevados. Um período maior de treinamento foi considerado (Figura 4.9).

Figura 4.8 – Tempo de equilíbrio instável em relação ao período de treinamento

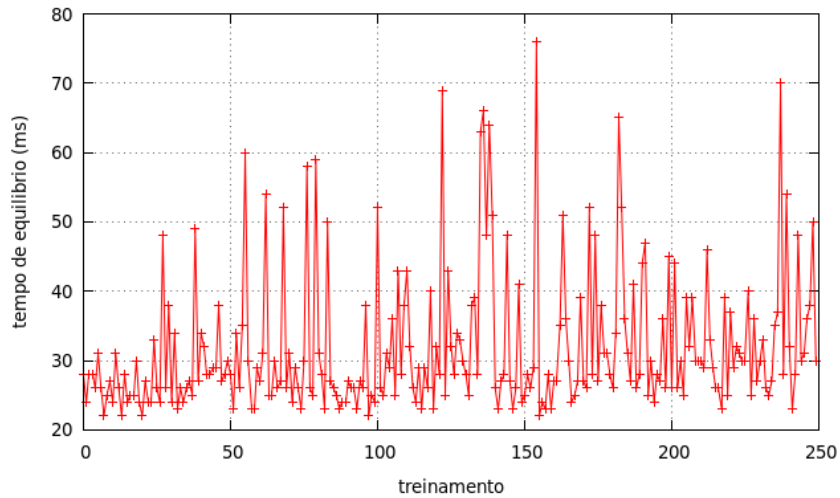
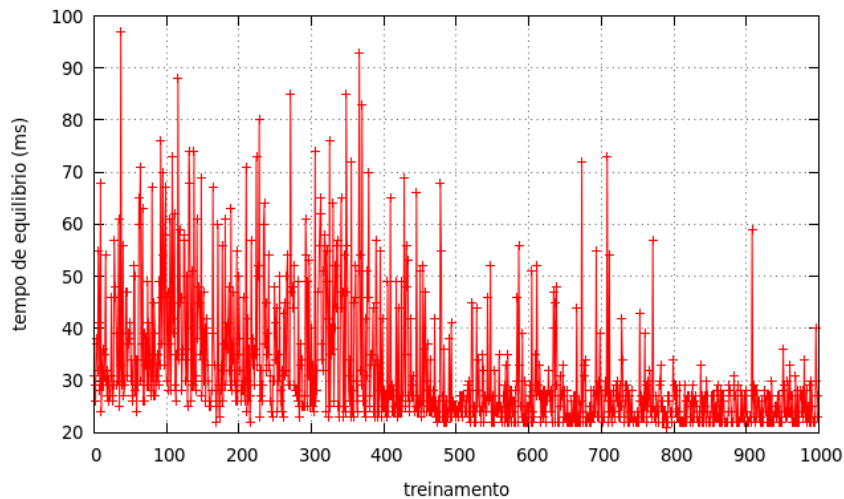


Figura 4.9 – Tempo de equilíbrio instável em relação ao período de treinamento



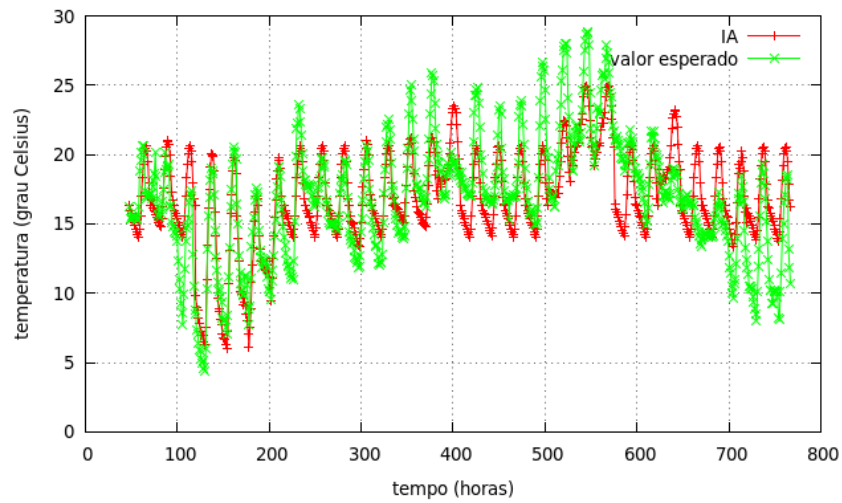
### 4.3 PREVISÃO DO TEMPO

#### 4.3.1 Segunda geração

A figura 4.10 mostra a previsão do tempo da temperatura - mês de maio/2016 Santa Maria. A figura 4.11 mostra a previsão para os outros parâmetros físicos (humidade, radiação solar, intensidade do vento e precipitação) para o mesmo período.

No objetivo deste trabalho, a rede neural deve ser capaz de fazer previsões de longa duração também, para estender suas capacidades de abstração, i.e. melhorar a busca por correlações entre os parâmetros físicos, a rede neural foi treinada para períodos maiores de meses. A figura 4.12 mostra a previsão da temperatura - mês de maio/2016 Santa Maria - sobre treinamento com

Figura 4.10 – Relação temperatura x tempo para a previsão da rede neural com um mês de dados de treinamento



dados de três meses (maio, abril e março).

A figura 4.13 mostra a previsão da temperatura - mês de maio/2016 Santa Maria - sobre treinamento com dados de seis meses (dezembro/2015 - maio/2016).

Somente com os dados de Santa Maria sobre treinamento do mês de maio, também foi feita previsão de um mês anterior (Figura 4.14) e do mês de maio na cidade de Porto alegre (Figura 4.15). Previsões de outros parâmetros físicos (com dados extras para Porto Alegre), podem ser encontrados no apêndice.

Figura 4.11 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural com um mês de dados de treinamento

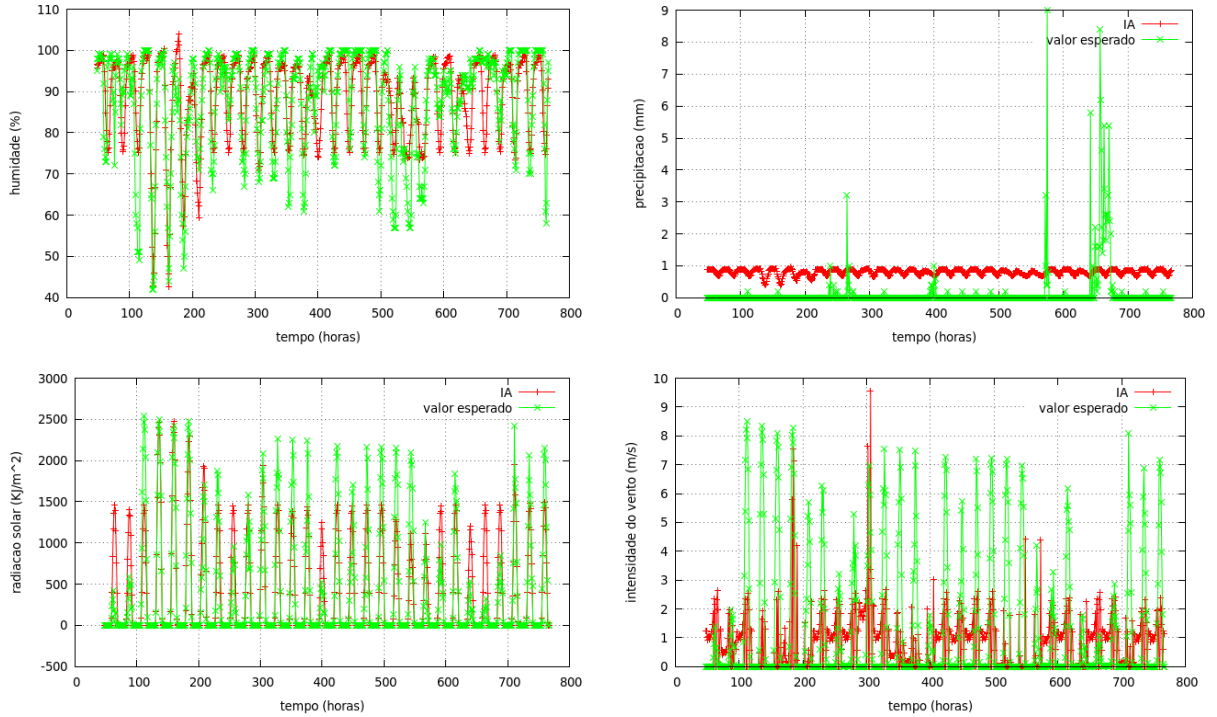


Figura 4.12 – Relação temperatura x tempo para a previsão da rede neural com três meses de dados de treinamento

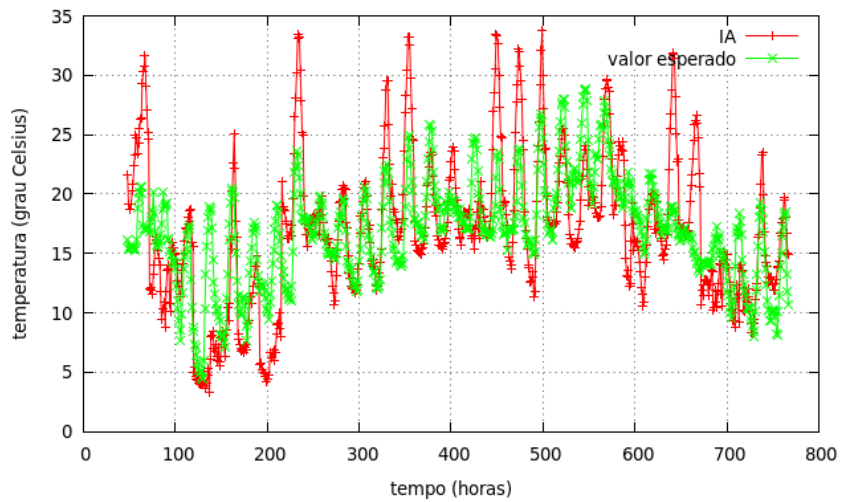


Figura 4.13 – Relação temperatura x tempo para a previsão da rede neural, 6 meses de dados

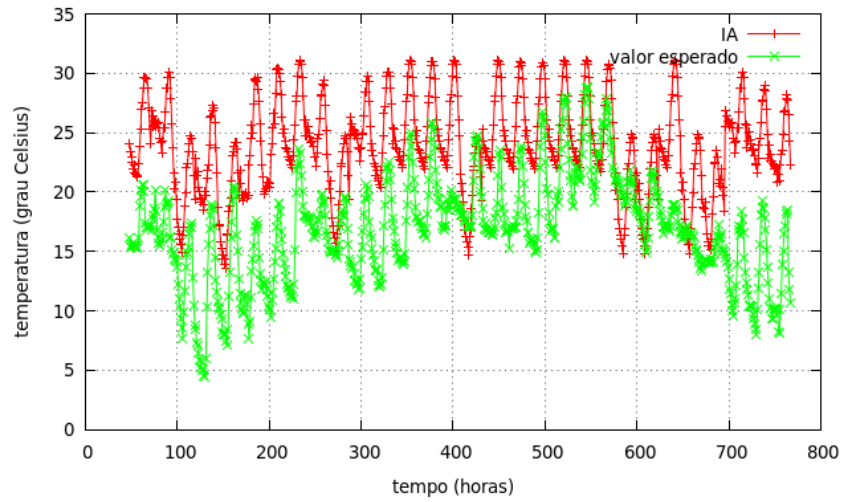


Figura 4.14 – Relação temperatura x tempo para a previsão da rede neural do mês de abril/2016

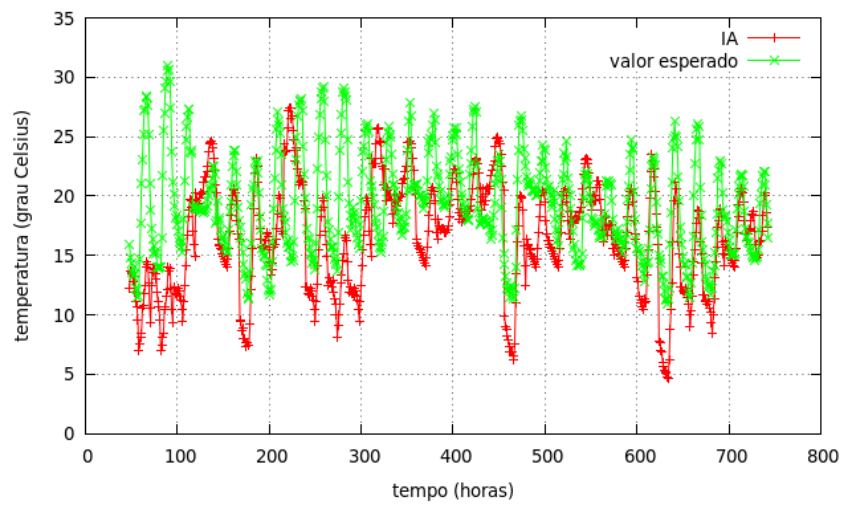
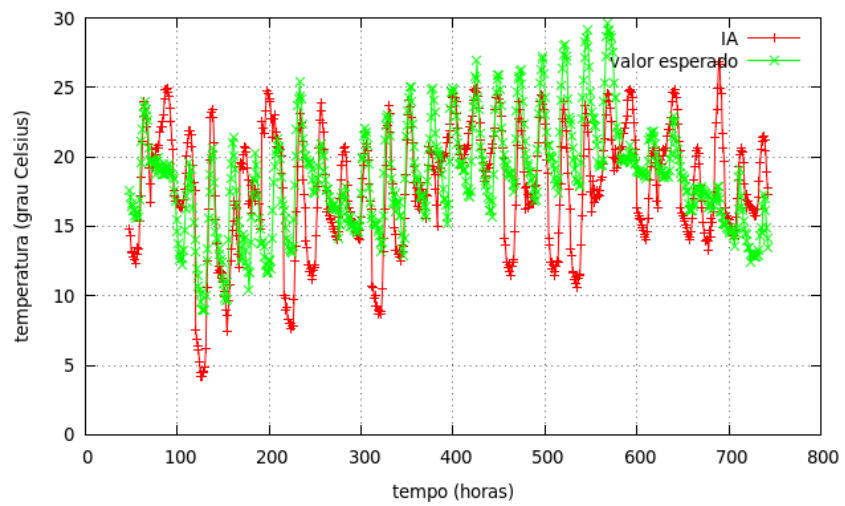


Figura 4.15 – Relação temperatura x tempo para a previsão da rede neural do mês de maio/2016 em Porto Alegre



## 5 CONCLUSÃO

O primeiro problema estudado, a aproximação de uma função analítica, mostrou que redes neurais, de ambas gerações, conseguem aproximar bem uma função analítica, como indicado pelo teorema do conexionismo. Enquanto na segunda geração de neurônios a aproximação da função é de forma direta, a terceira geração requer compreensão em como uma rede neural codifica e decodifica informação, fundamentalmente importante para resolver problemas neste tipo de neurônio. Duas formas de decodificação foram apresentadas: por tempo e por frequência, sendo ambas equivalentes para aproximação de função.

O segundo problema concebido, o pêndulo inverso, foi resolvido com sucesso pela rede neural de segunda geração. Mostramos que a I.A. consegue manter o equilíbrio instável por tempo suficiente.

O terceiro problema que foi considerado, a previsão do tempo, demonstrou ser um grande desafio a rede neural. Apesar disso, o comportamento oscilatório para a maioria dos parâmetros físicos (exceto precipitação) foi bem capturado. Os resultados podem, possivelmente, serem melhorados se vários fatores forem aperfeiçoados, listamos:

- Número de neurônios e número de camadas: adicionar mais neurônios, além de mais camadas, pode reduzir o erro devido a melhoria que a rede neural ganha em aproximar funções;

- Arquitetura da rede neural: outros formatos de rede podem ser considerados, como rede convolucional que é excepcionalmente boa em reconhecer padrões de imagens;

- Função ativação: embora a função sigmóide seja popular por seu desempenho, não necessariamente significa ser a melhor função para resolver qualquer problema;

- Parâmetros físicos: para o caso específico da previsão do tempo, considerar outros fatores físicos, como direção do vento, pressão atmosférica e imagem de satélite podem ser benéficos.

A terceira geração de neurônios, para o caso do pêndulo inverso, demonstrou ser um desafio ainda maior. A baixa qualidade nos resultados pode ser atribuído a quatro causas:

- 1) Hiperparâmetros: a quantidade de parâmetros que devem ser determinados antes da rede neural ser simulada é superior ao número presente na segunda geração;

- 2) Codificação: as informações que são fornecidas para os neurônios receptores devem ser transformadas em sinais que possam ser interpretados pela rede neural, determinar uma função codificadora adequada é um problema em aberto;

- 3) Decodificação: traduzir os sinais emitidos pelos neurônios da camada de saída em informação útil, como por exemplo as ações de manter o equilíbrio do pêndulo inverso, também permanece um problema em aberto.

- 4) Aprendizado: a forma em como os neurônios aprendem, STDP, possui realismo biológico, entretanto isto não implica na melhor eficiência em como os neurônios podem aprender a resolver um problema. Além disso, a própria técnica STDP apresenta dificuldades técnicas,



como sua independência de voltagem e da frequência de pulsos.

O ramo de redes neurais vem crescendo aceleradamente. Cada vez mais pesquisadores do mundo todo voltam a sua atenção a inteligência artificial. Cada vez mais reconhecido como utilidade ao desenvolvimento de novas tecnologias e, mais recentemente, ao progresso do conhecimento científico, I.A. representa a nova ferramenta da humanidade, cujos limites ainda não são conhecidos. A quantidade de informações e dados disponíveis são tão vastos que uma pessoa sozinha já não é mais capaz de lidar, devido a isto e sua extensa aplicabilidade, é necessário um investimento de recursos e pessoas cada vez maior a esta área. Nesta nova era da informação, I.A. se tornará o símbolo do próximo salto tecnológico.

## 6 REFERÊNCIAS

2001, A Space Odyssey. Disponível em <<http://www.imdb.com/title/tt0062622/>>. Acesso em 4 dez. 2016.

ABBOTT, L., Lapique's introduction of the integrate-and-fire model neuron (1907), *Brain Research Bulletin* 50 (5/6): 303–304. doi:10.1016/S0361-9230(99)00161-6. PMID 10643408, 1999.

ABHISHEK, K. et al., Weather Forecasting Model using Artificial Neural Network. 2nd International Conference on Computer, Communication, Control and Information Technology (C3IT-2012) on February 25 - 26, 2012. DOI:10.1016/j.protcy.2012.05.047.

ANDERSON, C. W., Learning to Control an Inverted Pendulum using Neural Networks, *Control Systems Magazine, IEEE*, vol. 9, no. 3, pp. 31-37, 1989.

AZAR, G. et al., Speedy Q-Learning". *Advances in Neural Information Processing Systems* 24: 2411–2419, 2011.

BAYMANI, M., KERAYECHAN, A., EFFATI, S., Artificial Neural Networks Approach for Solving Stokes Problem. *Applied Mathematics*, 2010, 1, 288-292. DOI:10.4236/am.2010.14037.

CAMPBELL, M., HOANE, A., HSU, F., Deep Blue, Elsevier, *Artificial Intelligence* 134, 57–83, 2002.

GALATEIA. Disponível em <<http://www.theoi.com/Pontios/NereisGalateia.html>>. Acesso em 4 dez. 2016.

GRAVES, A. et al., A Novel Connectionist System for Improved Unconstrained Handwriting Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, 2009.

HODGKIN, A. L., HUXLEY, A. F., A quantitative description of membrane current and its application to conduction and excitation in nerve". *The Journal of Physiology* 117 (4): 500–544. doi:10.1113/jphysiol.1952.sp004764. PMC 1392413. PMID 12991237, 1952.

HORNBY, G. et al., Automated antenna design with evolutionary algorithms. American Institute of Aeronautics and Astronautics, 2006

HORNIK, K., Approximation Capabilities of Multilayer Feedforward Networks, *Neural Networks*, 4(2), 251–257. doi:10.1016/0893-6080(91)90009-T, 1991.

IZHIKEVICH, E. M., Simple Model of Spiking Neurons, *IEEE TRANSACTIONS ON NEURAL NETWORKS*, VOL. 14, NO. 6, NOVEMBER, 2003.

KOHONEN, T., Self-Organized Formation of Topologically Correct Feature Maps". *Biological Cybernetics* 43 (1): 59–69. doi:10.1007/bf00337288, 1982.

LECUN, Y. et al., Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278-2324, 1998.

LECUN, Y. et al., Efficient BackProp, in Orr, G. and Muller K. (Eds), *Neural Networks: Tricks of the trade*, Springer, 1998.

LONG, L. N., Evolving Curricula for an Exponential World, *Educause Review*, Vol. 51, No. 2, Apr. 4, 2016.

MAHMUD, F., Zuhori S.T, Intelligent Autonomous Vehicle Navigated by Using Artificial Neural Network, pp. 105-108, *Electrical and Computer Engineering (ICECE)*.

MATUSUGU, M. et al., Subject independent facial expression recognition with robust face detection using a convolutional neural network. *Neural Networks* 16 (5): 555–559. doi:10.1016/S0893-6080(03)00115-1, 2003.

MOOR, J., The Dartmouth College Artificial Intelligence Conference: The Next Fifty Years, *AI Magazine Volume 27 Number 4*, 87-91, 2006.

NOSENGO, N., Can artificial intelligence create the next wonder material?, *Nature* 533, 22–25, doi:10.1038/533022a, 2016.

OBEL, L. et al., Brain glycogen-new perspectives on its metabolic function and regulation at the subcellular level. *Frontiers in neuroenergetics* 4: 3. doi:10.3389/fnene.2012.00003. PMC 3291878. PMID 22403540, 2012.

OPENAI, Gym. Disponível em <<https://gym.openai.com/>>. Acesso em 4 dez. 2016.

SABERIAN, H. et al., Modelling and Prediction of Photovoltaic Power Output Using Artificial Neural Networks, *International Journal of Photoenergy*, vol. 2014, Article ID 469701,

doi:10.1155/2014/469701, 2014.

SILVER, D. et al. Mastering the game of Go with deep neural networks and tree search, *Nature* 529, 484–489, doi:10.1038/nature169601, 2016.

STANLEY, K. O., MIIKKULAINEN, R., Evolving Neural Networks Through Augmenting Topologies. *Evolutionary Computation*, 10(2):99-127, 2002.

STAR WARS, A NEW HOPE, Disponível em <<http://www.imdb.com/title/tt0076759/>>. Acesso em 4 dez. 2016.

STORK, D. G., Is Backpropagation Biologically Plausible?, *International Joint Conference on Neural Networks*, Vol. II, IEEE/89CH2765-6, 241-246, 1989.

TANAKA, A., TOMIYA, A., Detection of phase transition via convolutional neural network. arXiv: 1609.09087, 2016.

TURING, A., Computing Machinery and Intelligence, *Mind* LIX (236): 433–460, 1950.

WWO, WORLD WEATHER ONLINE. Disponível em <<http://www.worldweatheronline.com/v2/historical-weather.aspx?q=RIA>>. Acesso em: 21 jun. 2016.

YU, H., Neural network iterative diagonalization method to solve eigenvalue problems in quantum mechanics, *Phys. Chem. Chem. Phys.*, 2015,17, 14071-14082, DOI: 10.1039/C5CP01438G.

YOSINSKI, J. et al., Understanding Neural Networks Through Deep Visualization, arXiv:1506.06579, 2015.

## 7 APÊNDICE

Previsão do tempo para outros parâmetros sobre diferentes períodos de treinamento, mês e local. A rede neural usou apenas dados climáticos de Santa Maria.

Figura 7.1 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural para maio/2016 - Santa Maria, 3 meses de dados

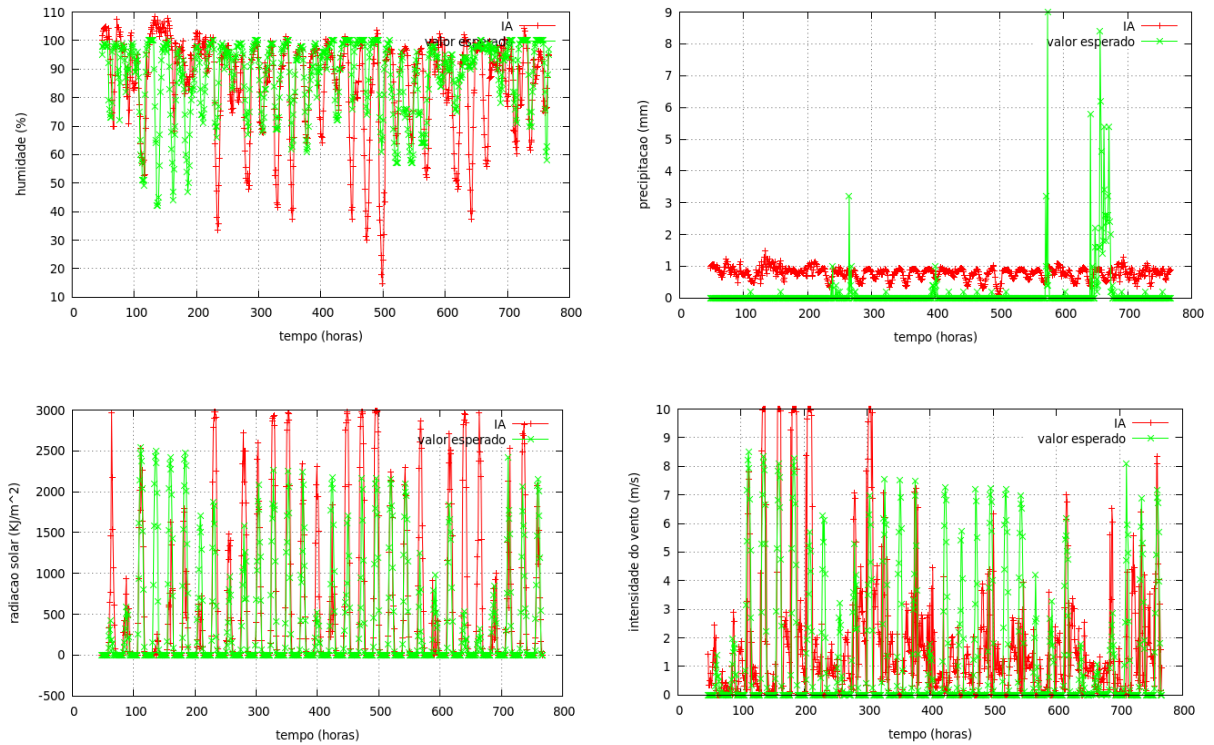


Figura 7.2 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural para maio/2016 - Santa Maria, 6 meses de dados

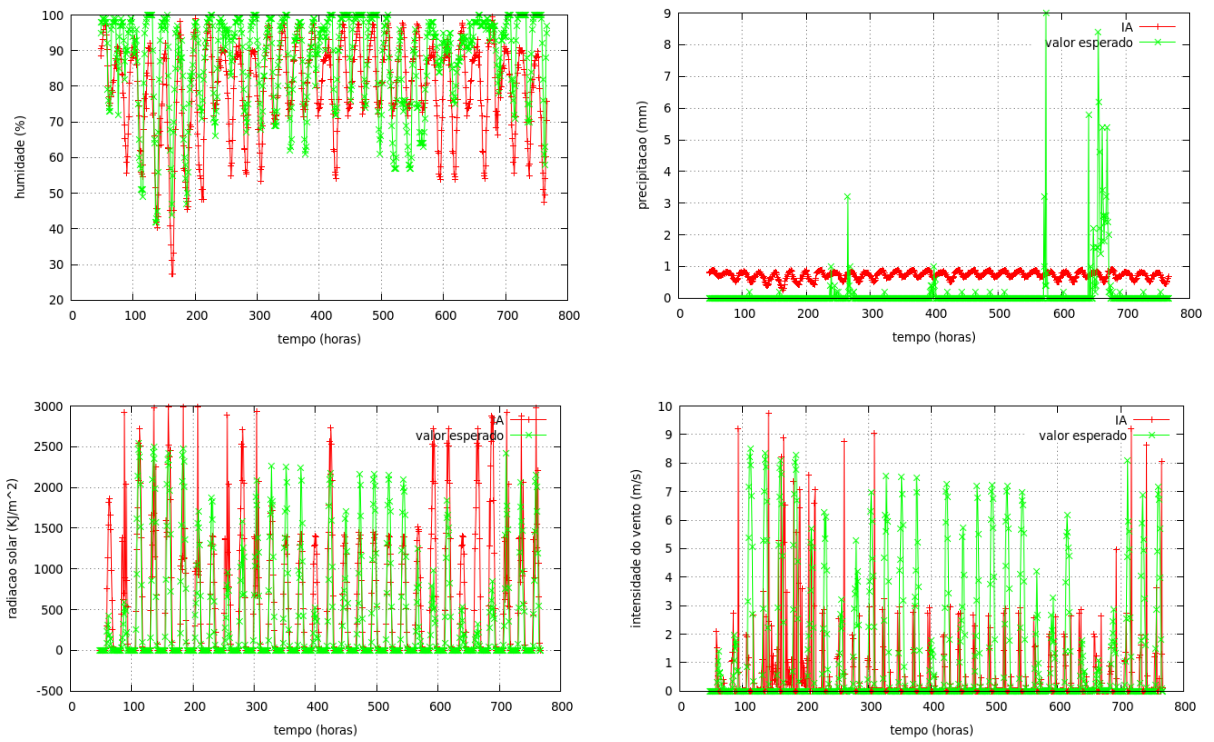


Figura 7.3 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural para maio/2016 - Porto Alegre, 1 mês de dados

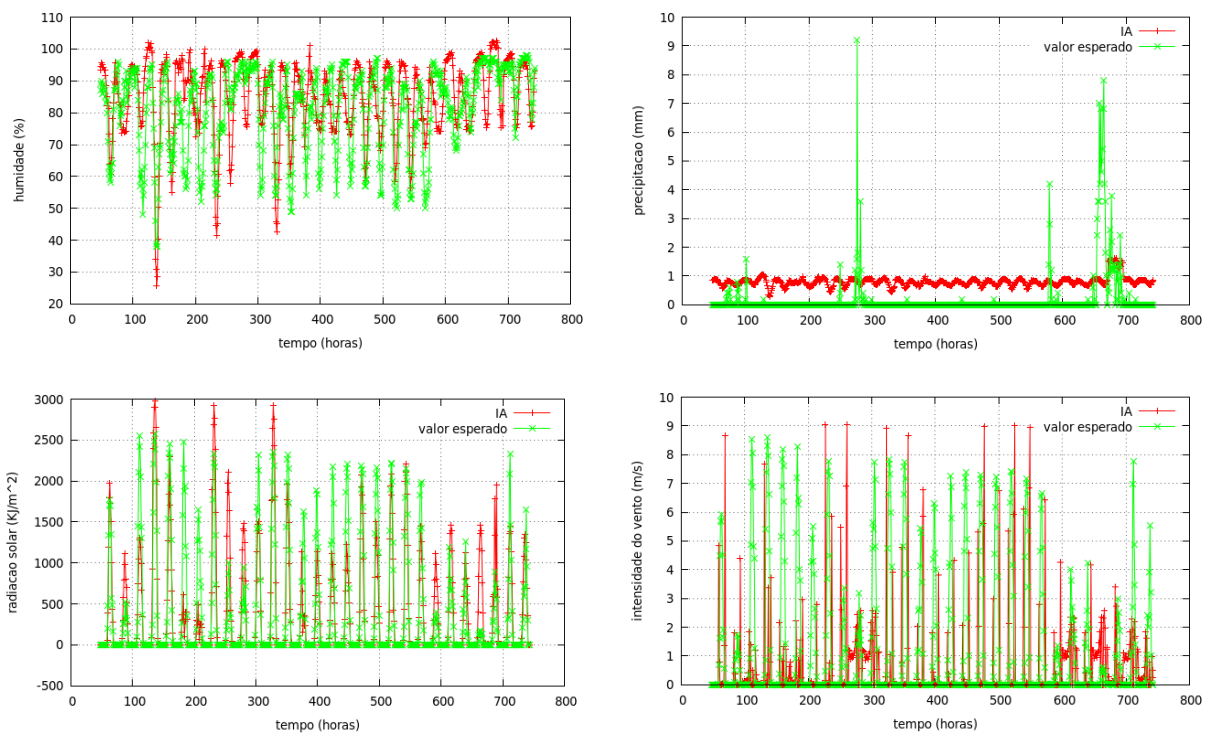


Figura 7.4 – Relação de outros parâmetros físicos x tempo para a previsão da rede neural para abril/2016 - Santa Maria, 1 mês de dados (maio)

