



UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE ENERGIAS ALTERNATIVAS E RENOVÁVEIS
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

Ricardo Cavalcanti Costa

**Modelos Preditivos de Velocidade de Vento para Sistemas Eólicos
Baseados em Redes Neurais Artificiais**

João Pessoa – PB
Junho de 2016

RICARDO CAVALCANTI COSTA

**MODELOS PREDITIVOS DE VELOCIDADE DE VENTO PARA
SISTEMAS EÓLICOS BASEADOS EM REDES NEURAS ARTIFICIAIS**

*Trabalho de Conclusão de Curso apresentado
ao Departamento de Engenharia Elétrica
como um dos pré-requisitos para a obtenção
do título de Engenheiro Eletricista sob
orientação do Prof. Rogério Gaspar de
Almeida*

João Pessoa - PB
Junho de 2016

COSTA, Ricardo Cavalcanti

Modelos Preditivos de Velocidade de Vento para Sistemas
Eólicos Baseados em Redes Neurais Artificiais.

66 f. il.

Orientador: Prof. Dr. Rogério Gaspar de Almeida

Trabalho de Conclusão de Curso, 2016

1. Modelos Preditivos. 2. Redes Neurais Artificiais. 3.
Velocidade de Vento.

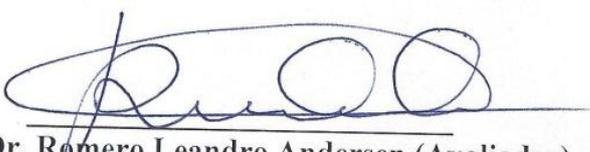
RICARDO CAVALCANTI COSTA

**MODELOS PREDITIVOS DE VELOCIDADE DE VENTO
PARA SISTEMAS EÓLICOS BASEADOS EM REDES NEURAIAS
ARTIFICIAIS**

*Trabalho de Conclusão de Curso submetido
ao Departamento de Engenharia Elétrica da
Universidade Federal da Paraíba como parte dos
requisitos necessários para a obtenção do título de
Engenheiro Eletricista.*

Aprovado em: 17 / 06 / 2016


Prof. Dr. Rogério Gaspar de Almeida (Orientador)
Universidade Federal da Paraíba


Prof. Dr. Romero Leandro Andersen (Avaliador)
Universidade Federal da Paraíba


Prof. Dr. Yuri Percy Molina Rodriguez (Avaliador)
Universidade Federal da Paraíba

João Pessoa, Paraíba
Junho de 2016

*Dedico este trabalho aos meus pais,
Silvio Romero Santos da Costa e
Girlene dos Santos Cavalcanti da
Costa, por todo cuidado, proteção e
carinho que tiveram comigo durante
toda a minha vida.*

AGRADECIMENTOS

Eu agradeço primeiramente a Deus, por ele ser o centro da minha vida, o sentido da minha alegria, por ele ter me dado força e coragem durante os anos de graduação e por ele ser a grande prioridade da minha vida.

Agradeço a meus pais e meus irmãos por toda força e confiança que tiveram comigo neste tempo da vida acadêmica.

Agradeço aos meus irmãos da Comunidade Católica Shalom, a qual sou membro, por suas orações e cuidado para comigo.

Agradeço ao meu professor e orientador Rogério Almeida, pela sua dedicação e empenho como professor e orientador, o qual pude aprender bastante e me ajudou na realização deste trabalho.

Agradeço aos meus colegas de classe que dividiram os conhecimentos comigo durante todo o tempo de graduação

Agradeço também aos todos os professores, que me ensinaram e me ajudaram a consolidar o conhecimento durante a graduação

*“Buscai em primeiro lugar o Reino de Deus e a sua justiça e todas estas coisas vos serão dadas em acréscimo.”
Mateus 6-33.*

RESUMO

Este trabalho tem por objetivo o desenvolvimento de modelos de predição de velocidade de vento baseados em redes neurais artificiais do tipo *Perceptron* Multicamadas (*Multi-Layer Perceptron*) utilizando o algoritmo *Back-Propagation* através da *toolbox* de redes neurais que faz parte do software MATLAB/SIMULINK[®]. Foram criados onze modelos de predição, cujos dados utilizados para o treinamento, a validação e o teste dos modelos neurais envolveram combinações entre o horário de medição, a temperatura, a umidade e a pressão atmosférica coletados da estação meteorológica do Centro de Energias Alternativas e Renováveis (CEAR) do campus de João Pessoa da UFPB entre o período de 2009 a 2013. Através de índices quantitativos, foi possível qualificar o modelo que apresentou o melhor desempenho, tendo em conta, o conjunto específico de grandezas medidas que são utilizadas para a predição da velocidade média de vento mensal. Tal modelo pode ser utilizado na predição da potência elétrica média que um dado sistema eólico pode produzir.

Palavras-Chaves: Redes Neurais Artificiais. Predição de velocidade média mensal de vento. MATLAB/SIMULINK[®].

ABSTRACT

This work has for its object the development of wind speed prediction models based on artificial neural networks type Multilayer Perceptron (Multi-Layer Perceptron) using the back-propagation algorithm through the toolbox neural network which is part of MATLAB / SIMULINK®. Were created eleven prediction models whose data used for training, validation and testing of neural models involved combinations of the measuring time, temperature, humidity and atmospheric pressure collected from the meteorological station of Alternative Energy and Renewable Center (CEAR) campus of Joao Pessoa UFPB between the time of 2009 to 2013. Through quantitative indices, it was possible to qualify the model that showed the best performance, taking into account the specific set of measured quantities that are used for prediction average monthly wind speed. This model can be used to predict the average electrical power to a given wind system can produce.

Keywords: Artificial Neural Networks. Predicting wind monthly average speed. MATLAB / SIMULINK®.

LISTA DE FIGURAS

Figura 1 - Estrutura do neurônio biológico	17
Figura 2 – Modelo Matemático de um neurônio biológico.....	18
Figura 3 – Esquema de uma rede multi-layer perceptron (MLP).....	24
Figura 4 – Esquema de uma rede que mostra a direção da propagação dos sinais funcionais e da retropropagação dos sinais de erro	25
Figura 5 – Esquema que mostra o fluxo de processamento do algoritmo <i>backpropagation</i>	26
Figura 6 – Fluxograma da estrutura dos modelos preditivos desenvolvida pelo autor	33
Figura 7 – Dados de entrada para o treinamento e validação dos modelos preditivos de velocidade de vento entre os anos de 2009 a 2011.....	35
Figura 8 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 1 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	41
Figura 9 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 5 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	43
Figura 10 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 8 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	44
Figura 11 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 9 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	46
Figura 12 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 11 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	47
Figura 13 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 2 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	58
Figura 14 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 3 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	59
Figura 15 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 4 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	61
Figura 16 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 6 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	62
Figura 17 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 7 da rede neural entre os meses de Janeiro a Dezembro de 2012.....	64
Figura 18 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 10 da rede neural para os meses de Janeiro a Dezembro de 2012.....	65

LISTA DE TABELAS

Tabela 1 - Modelos da Rede Neural para predição de velocidade de vento.....	31
Tabela 2 - Parâmetros da rede MLP do tipo <i>backpropagation</i> para diferentes modelos da Rede Neural. A linha destacada corresponde ao melhor modelo para a rede neural	36
Tabela 3 - Média dos dados reais e estimados de vento para o modelo 1 da Rede Neural Artificial....	42
Tabela 4 – Dados de entrada utilizados para o modelo 1	42
Tabela 5 - Média dos dados reais e estimados de vento para o modelo 5 da Rede Neural Artificial....	43
Tabela 6 - Dados de entrada utilizados para o modelo 5.....	44
Tabela 7 - Média dos dados reais e estimados de vento para o modelo 8 da Rede Neural Artificial....	45
Tabela 8 - Dados de entrada utilizados para o modelo 8.....	45
Tabela 9 - Média dos dados reais e estimados de vento para o modelo 9 da Rede Neural Artificial....	46
Tabela 10 - Dados de entrada utilizados para o modelo 9.....	47
Tabela 11 – Média dos dados reais e estimados de vento para o modelo 11 da Rede Neural Artificial....	48
Tabela 12 - Dados de entrada utilizados para o modelo 11.....	48
Tabela 13 - Média dos dados reais e estimados de vento para o modelo 2 da Rede Neural Artificial..	58
Tabela 14 - Dados de entrada utilizados para o modelo 2.....	59
Tabela 15 - Média dos dados reais e estimados de vento para o modelo 3 da Rede Neural Artificial..	60
Tabela 16 - Dados de entrada utilizados para o modelo 3.....	60
Tabela 17 - Média dos dados reais e estimados de vento para o modelo 4 da Rede Neural Artificial..	61
Tabela 18 - Dados de entrada utilizados para o modelo 4.....	62
Tabela 19 - Média dos dados reais e estimados de vento para o modelo 6 da Rede Neural Artificial..	63
Tabela 20 - Dados de entrada utilizados para o modelo 6.....	63
Tabela 21 - Média dos dados reais e estimados de vento para o modelo 7 da Rede Neural Artificial..	64
Tabela 22 - Dados de entrada utilizados para o modelo 7.....	65
Tabela 23 - Média dos dados reais e estimados de vento para o modelo 10 da Rede Neural Artificial....	66
Tabela 24 - Dados de entrada utilizados para o modelo 10.....	66

SUMÁRIO

1	INTRODUÇÃO	13
1.1	MOTIVAÇÃO	14
1.2	OBJETIVO GERAL	14
1.2.1	Objetivos Específicos.....	14
1.3	ORGANIZAÇÃO DO TRABALHO	15
2	FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA.....	16
2.1	INTRODUÇÃO	16
2.2	REDES NEURAIS ARTIFICIAIS	16
2.3	REVISÃO BIBLIOGRÁFICA.....	19
2.3.1	Diferentes modelos e meios para Predição do vento.....	20
2.3.2	Redes Neurais Artificiais: Tipos e Aplicações.....	21
3	ARQUITETURA DA REDE NEURAL PARA OS MODELOS PREDITIVOS.....	23
3.1	INTRODUÇÃO	23
3.2	REDE NEURAL MULTICAMADA COM <i>BACKPROPAGATION</i>	23
3.2.1	Algoritmo <i>BackPropagation</i>	25
3.3	ESTRUTURA DOS MODELOS PREDITIVOS	31
4	ANÁLISE DE RESULTADOS	34
4.1	INTRODUÇÃO	34
4.2	ANÁLISE QUALITATIVA E QUANTITATIVA	34
5	CONCLUSÕES E PERSPECTIVAS PARA TRABALHOS FUTUROS	49
6	REFERÊNCIAS BIBLIOGRÁFICAS	50
	APÊNDICE.....	52

1 INTRODUÇÃO

Nos dias atuais devido ao aumento da demanda de energia elétrica, e tendo como tendências a escassez de combustíveis fósseis e as duras restrições para a construção de novas hidrelétricas, a busca por formas de energia ambientalmente limpas com mínima produção de resíduos durante a sua utilização na produção de eletricidade tem vindo a ser cada vez mais intensificada. Dentre estas formas alternativas de energia, a geração de energia através do vento é uma das que vem sendo mais utilizadas no Brasil e no mundo.

Entretanto, o uso cada vez maior da energia eólica na produção de energia elétrica é um desafio, pois existe a necessidade de se conhecer os impactos de operação, manutenção e interligação dessa tecnologia com o sistema elétrico existente. Com base nesse pressuposto, uma questão a se considerar consiste na predição da geração de energia elétrica pelos geradores eólicos que visa diminuir esse desafio, uma vez que a velocidade de vento não depende da intervenção humana.

Uma das alternativas de minimizar os impactos de sistemas eólicos nas redes elétricas, sobretudo nas suas capacidades de geração em determinados períodos, consiste no conhecimento intenso dos regimes de vento. Dada a importância desses regimes, a coleta de dados de vento obtida de estações de medição deve estar ausente de erro. Dessa forma, torna-se imprescindível realizar o tratamento desses dados de modo a evitar erros de predição de potência produzidos pelos parques eólicos.

Uma maneira de se estimar a velocidade do vento de modo a minimizar os erros de medição decorrentes de falhas humanas, de equipamentos e sensores consiste no uso da inteligência computacional. Nesta área do conhecimento, destacam-se as redes neurais artificiais, cujo emprego na predição de velocidades de vento para curto, médio ou longos períodos de tempo é uma opção cada vez mais consolidada (VERAS, 2013).

Com base nesse argumento, este trabalho de conclusão do curso visa desenvolver modelos de predição de velocidade média de vento mensal baseados em redes neurais artificiais, tal que, possam ser utilizados na predição de potência elétrica média produzida por geradores eólicos.

1.1 Motivação

A motivação deste trabalho se baseia no desenvolvimento de modelos preditivos de velocidade média de vento utilizando redes neurais artificiais. Para isto foram utilizados neste trabalho para o aprendizado dos modelos neurais dados reais como: horário de medição, temperatura, umidade e pressão. Uma vez treinados, estes modelos de predição fariam o papel, por exemplo, do anemômetro¹ da estação meteorológica do Centro de Energias Alternativas e Renováveis (CEAR) da Universidade Federal da Paraíba (UFPB) do Campus de João Pessoa, numa situação em que o aparelho estivesse temporariamente em manutenção, não comprometendo os estudos que necessitam do conhecimento da velocidade de vento. Por outro lado, servem também para suprir erros de medição que ocasionalmente podem ocorrer na coleta e armazenamento de dados.

1.2 Objetivo Geral

Desenvolvimento de modelos baseados em redes neurais artificiais para predição de velocidade média de vento mensal.

1.2.1 Objetivos Específicos

Para se alcançar o objetivo geral descrito anteriormente, o trabalho envolverá:

- Uso de dados de vento reais coletados a partir da estação meteorológica localizada no campus de João Pessoa, próximo ao novo prédio do Centro de Energias Alternativas e Renováveis (CEAR);
- Tratamento dos dados coletados para a definição dos conjuntos de dados de treinamento, validação e de teste a serem utilizados no treinamento e avaliação de desempenho dos modelos de predição baseados em redes neurais;

¹ Aparelho que mede a velocidade de vento

- Definições das arquiteturas das redes neurais para cada modelo de predição adotado e implementação das mesmas através de programas computacionais que explorarão a *toolbox* de redes neurais associada ao software MATLAB/SIMULINK®;
- Uma vez implementadas as diversas arquiteturas de redes neurais de cada modelo de predição através de programas computacionais em MATLAB®, o passo seguinte consiste em testá-las e alterá-las, de modo a averiguar a estrutura relativa a um dado modelo que apresente os melhores resultados de predição de velocidade média de vento mensal, qualificados a partir de índices quantitativos (Erro Médio Quadrático, Erro Percentual Médio Absoluto e Coeficiente de Regressão Linear);
- Comparação de desempenho entre os demais modelos de predição.

1.3 Organização do trabalho

O trabalho organiza-se em 4 capítulos, descritos conforme a sequência abaixo:

- No Capítulo 2 serão apresentadas a Fundamentação teórica e a revisão bibliográfica, em que serão apresentados os principais trabalhos da ênfase que contextualizaram o assunto desenvolvido neste Trabalho de Conclusão de Curso.
- O Capítulo 3 descreve o tipo da rede neural utilizada neste trabalho, bem como o desenvolvimento das arquiteturas das redes neurais para os diferentes modelos de previsão definidos para a previsão de velocidade média de vento mensal;
- No Capítulo 4 serão apresentados os resultados e análises acerca dos diferentes modelos neurais de previsão, destacando-se a melhor configuração dentre os demais modelos.
- Por fim, o Capítulo 5 apresenta as conclusões referentes ao trabalho desenvolvido e as perspectivas para tarefas futuras.

2 FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA

2.1 Introdução

Este capítulo abordará os principais conceitos referentes a redes neurais artificiais, assim como, os trabalhos da literatura da especialidade que serviram de base para a construção deste trabalho.

2.2 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNAs) são definidas como técnicas computacionais constituídas por neurônios matemáticos, em que cada neurônio artificial tem uma estrutura que se assemelha a de um neurônio biológico. Para propósitos de comparação, uma rede neural artificial pode chegar a ter centenas ou milhares de unidades de processamento enquanto que o cérebro de um mamífero, por exemplo, pode ter bilhões de neurônios. (CARVALHO, 2010)

A estrutura neural de um cérebro biológico é basicamente formada por neurônios, cuja composição é ilustrada na Figura 1. Um neurônio biológico é basicamente constituído por dendritos que são finos prolongamentos e representam os terminais de entrada de sinais recebidos de outros neurônios; pelo corpo central que é responsável por coletar e combinar as informações recebidas a partir dos dendritos; e pelo axônio, que é constituído de uma fibra tubular e representa o terminal de saída. Eles se comunicam através de sinapses, que consistem numa região onde dois neurônios entram em contato e através da qual os impulsos nervosos são transmitidos entre eles. Os impulsos recebidos por um neurônio não são imediatamente retransmitidos, mas sim processados e, caso atinja um dado limiar de ação (*threshold*) o neurônio produz uma substância neurotransmissora que flui do corpo central para o axônio, que pode estar conectado a um dendrito de um outro neurônio.

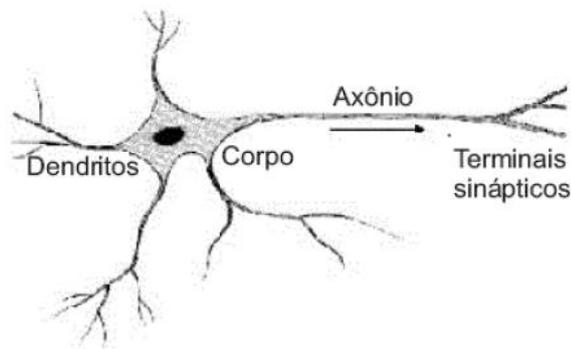


Figura 1 - Estrutura do neurônio biológico

Fonte: (CARVALHO, 2010)

A passagem de um sinal de um neurônio a outro pode ser amplificado ou atenuado, pois depende das características dos dendritos de origem. Sendo assim, cada dendrito está associado a um peso pelo qual o sinal recebido pelo neurônio é multiplicado. Os pesos representam justamente a memória. Portanto, a memorização tem relação com os valores dos pesos que são estabelecidos por meio de treinamento que o cérebro recebe durante toda a sua existência.

É importante destacar que o cérebro é construído por várias regiões especializadas a uma dada função, como por exemplo, a processamento de sinais auditivos, a elaboração de pensamentos, entre outros. Esse processamento se realiza através de redes específicas interligadas entre si, realizando processamento paralelo. Cada região do cérebro possui uma arquitetura de rede diferente, pois varia o número de neurônios, de sinapses por neurônio, de pesos, entre outros (HAYKIN, 2000).

A partir das características funcionais de um neurônio biológico, pode-se definir um modelo matemático bem simples, formado por um conjunto de entradas, uma unidade de processamento e uma ou mais unidades de saídas, que comparando com a estrutura do neurônio biológico, correspondem, respectivamente, aos dendritos, corpo celular e os axônios. A representação do neurônio matemático (neurônio artificial) é mostrada na Figura 2.

Em regra, uma rede neural artificial é composta por camadas, onde se define uma camada de entrada na qual estão os neurônios de entrada; a camada de saída, no qual estão os neurônios de saída; e uma camada oculta onde encontram-se neurônios internos ou ocultos que recebem os sinais de saída dos neurônios da camada de entrada e após processá-los transmite-os através de suas saídas para a entrada dos neurônios da camada de saída. Porém, é muito comum a rede neural possuir mais de uma camada oculta.

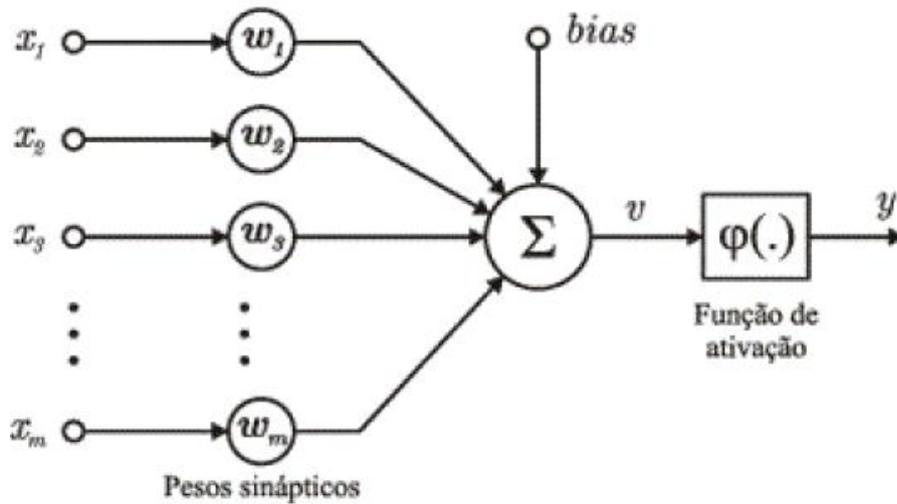


Figura 2 – Modelo Matemático de um neurônio biológico

Fonte: (VERAS, 2013)

Por outro lado, matematicamente, a equação que representa o neurônio artificial se dá pelo somatório do produto entre as entradas $x_1, x_2, x_3, \dots, x_n$ e os pesos sinápticos $w_1, w_2, w_3, \dots, w_n$, o qual passa para a função de ativação até chegar a saída. Esta equação está descrita a seguir (VERAS, 2013):

$$net_i = \sum_{i=0}^n x_i \cdot w_i \quad (1)$$

A saída y do neurônio é representada como uma função da equação em (1), conforme segue a expressão a seguir (VERAS, 2013):

$$y = f(net_i) \quad (2)$$

As RNAs começaram a surgir por volta do ano de 1943, quando Warren McCulloch e Walter Pitts produziram pela primeira vez o primeiro modelo artificial do neurônio biológico. A tentativa começou com McCulloch, psiquiatra e neuroanatomista, que passou 20 anos tentando representar um evento no sistema nervoso. Seu colega Pitts, matemático recém-graduado, só se juntou a ele em 1942 (BRAGA, LUDERMIR & CARVALHO, 2000).

No seu trabalho pioneiro em 1943, cujo título era “*A Logical Calculus of the Ideas Immmanent in Nervous Activity*”, foram discutidas novas idéias sobre redes lógicas de nodos e máquinas de estados finitos, com diversas representações lógicas envolvendo vários tipos de

comportamento e memória. A discussão referente às Redes Neurais Artificiais (RNAs) era geralmente com relação aos métodos de aprendizado, no qual o estudo envolvia basicamente os nodos, com o intuito que pudessem ser capazes de executar uma determinada função. O trabalho de Pitts e McCulloch era mais voltado em caracterizar um modelo artificial para um neurônio, apresentando seus resultados computacionais, do que em mostrar técnicas de aprendizado. (BRAGA, LUDERMIR & CARVALHO, 2000).

As técnicas de aprendizado de redes biológicas e artificiais passou a ser o foco das discussões alguns anos após os trabalhos de Pitts e McCulloch. O primeiro trabalho, foi o de Donald Hebb, em 1949, no qual teve ligação direta com as técnicas de aprendizado. Hebb pode apresentar que através da variação de pesos de entradas dos nodos é atingida a plasticidade de aprendizagem de redes neurais (BRAGA, LUDERMIR & CARVALHO, 2000).

O *Perceptron*, só veio ser discutido em 1958, por Frank Rosenblatt, no qual as redes neurais artificiais com modelo MCP (McCulloch e Pitts) que fossem acrescidas de sinapses ajustáveis classificariam certos tipos de padrões através de seu treinamento. Onze anos depois, em 1969, Minsky e Papert perceberam que o *perceptron* não conseguia executar determinadas tarefas, devido ao fato deste só resolver problemas linearmente separáveis, através de uma reta. Os tipos de conexão para as RNAs ficaram adormecidas na década de 70 e só se voltou a discutir sobre o assunto em 1982, quando John Hopfield escreveu um artigo sobre as propriedades de conexão das RNAs o que fez renascer o interesse por tais algoritmos (BRAGA, LUDERMIR & CARVALHO, 2000). Desde então, muitas estruturas de redes neurais foram desenvolvidas bem como técnicas de aprendizado o que fez com que o uso de redes neurais artificiais abrangesse várias áreas do conhecimento.

2.3 Revisão Bibliográfica

Neste trabalho o foco principal consiste no desenvolvimento de redes neurais para a predição de velocidade média de vento. Para a realização do trabalho foi tomado como referência artigos publicados em revistas da especialidade, bem como monografias e dissertações de mestrado que abrangem tema semelhante. Para propósitos de simplificação, essa seção foi subdividida em duas subseções, na qual a primeira subseção enfatiza os

diferentes modelos e meios para previsão de vento e sua importância como já comentado no capítulo 1; e a segunda descreve sobre as redes neurais artificiais, os tipos e principais aplicações.

2.3.1 Diferentes modelos e meios para Predição do vento

Conforme já anunciado no capítulo 1, a energia eólica é uma das mais utilizadas na produção de potência elétrica por ser renovável, limpa, e praticamente por não produzir resíduos.

Segundo Bitencourt (2015), o mundo vive um momento de adoção de fontes de energias renováveis e o Brasil não está fora desse contexto. O autor também afirma que uma dessas novas fontes é a energia eólica que é obtida pela energia cinética gerada pela migração das massas de ar, provocada pela diferença de temperatura entre regiões da superfície da terra. O autor afirma ainda que os ventos do Brasil estão entre os melhores do mundo, compostos pelos ventos elegantes e bem comportados. Uma vez que a geração eólica depende da velocidade dos ventos, cujo fenômeno independe da intervenção do homem, torna-se um desafio a sua produção, distribuição e o fornecimento de eletricidade em um nível permanente. Por este motivo, a previsão confiável deste fato é importante para garantir o fornecimento de energia num futuro próximo.

O autor afirma também que a geração de energia numa turbina eólica depende principalmente da área do rotor, da densidade do ar e da velocidade do vento, e que a área do rotor e a densidade do ar não variam expressamente com a localização geográfica do aerogerador, e dentre estes três fatores, o que caracteriza a volatilidade da geração eólica é a velocidade dos ventos, que depende da hora, temperatura, umidade e pressão e do terreno.

Por outro lado, Souza (2011) afirma que os modelos de predição podem ser classificados em três categorias: Modelos físicos, modelos estatísticos/inteligência artificial e modelos híbridos. Os modelos físicos fornecem previsões de velocidade e direção do vento com base em informações meteorológicas, como a pressão, temperatura, e informações físicas como a orografia do terreno e a altura das turbinas. Estes modelos empregam métodos numéricos para a previsão climática e são adequados para horizontes de previsão superiores a 6 horas a frente. Todavia, os modelos estatísticos empregam algoritmos estáticos, e neste

cenário destacam-se as seguintes técnicas: ARIMA (Modelo Auto-Regressivo Integrado de Média Móvel) que é uma técnica muito utilizado na modelagem e previsões de séries temporais (GREENE, 2008) ; ARFIMA, modelo que generaliza a técnica ARIMA; Filtro de Kalman, regressão harmônica, *wavelets* e modelos híbridos, inteligência computacional, como: redes neurais artificiais, desenvolvido pelo autor deste trabalho de conclusão de curso, lógica fuzzy, máquinas de vetor de suporte e modelos híbridos, por exemplo, as redes neuro-fuzzy e os híbridos.

Já o estudo apresentado por Freire (2014), afirma que os modelos de previsão mais usados são os modelos físicos e os modelos estatísticos/inteligência artificial. Os modelos físicos realizam as previsões com base nos dados meteorológicos e utilizam métodos numéricos para a previsão. Os modelos estatísticos e de inteligência artificial são mais simples e mais rápidos na realização das previsões e são mais recomendados para previsões de curto prazo. A técnica mais usada em modelos de inteligência artificial é a rede neural artificial. Por outro lado, o autor afirma também que um outro modelo que apresenta bons resultados em previsões de velocidade do vento de curto prazo é o modelo de persistência, no qual consiste em assumir que as condições que influenciam a previsão não mudarão em um instante de tempo futuro, fazendo com que a velocidade neste instante de tempo seja igual a velocidade atual.

Conforme se atesta nas literaturas acima consultadas, as redes neurais artificiais consistem na ferramenta de inteligência computacional mais utilizada na predição de vento, razão pela qual incide no assunto explorado pelo autor deste trabalho de conclusão de curso.

2.3.2 Redes Neurais Artificiais: Tipos e Aplicações

Bitencourt (2015) classifica a rede neural artificial de *perceptron* como uma rede direta constituída de unidades binárias que aprende a classificar padrões através de aprendizado supervisionado. Ele classifica ainda o agrupamento de *perceptrons* em camadas como Multicamada de *Perceptron* (*Multilayer Perceptron - MLP*, na língua inglesa), para resolver problemas mais complexos que um *perceptron* não seria capaz de resolver.

Por outro lado, Henrique (2014) afirma que nos anos de 50 e 60 foi quando surgiram as redes neurais *Perceptron* (1957) e a *Adaline* em 1960. A *Perceptron* foi uma das redes neurais a utilizar a regra de aprendizado para reajustar os pesos, enquanto a *Adaline* permitiu

trabalhar com a saída da rede no espaço contínuo. Essas redes foram fundamentais para o estudo das RNAs futuras, como a MLP (*Multi-Layer Perceptron*) e a RBNN (*Radial Basis Neural Network*).

Segundo Tablada (2010), um dos algoritmos mais sucedidos e utilizados em redes MLP é o algoritmo *BackPropagation*, que significa propagação recursiva e com aprendizado supervisionado. Ele corrige os pesos da rede na ordem inversa da propagação do sinal, através da taxa de aprendizado e do momento.

Em Zuben e Attux (2004), pertencente ao aprendizado não-supervisionado, tem-se a rede neural de *Kohonen*, que diferentemente da rede neural do tipo *BackPropagation*, não se acompanha o processo de aprendizado, o treinamento.

Em Freire (2014), a taxa de aprendizado corresponde ao tamanho do passo em direção a correção do erro e o momento é utilizado para acelerar a convergência da rede e diminuir a incidência de mínimos locais.

Com base nas afirmações apontadas pelas referências acima, optou-se nesse trabalho de conclusão de curso pelo o uso de redes neurais do tipo *perceptron* multicamadas com a técnica de propagação recursiva do erro (*BackPropagation*) para a predição da velocidade média de vento. Além disso, adotou-se também como base para o desenvolvimento dos modelos de predição neurais neste trabalho, o artigo elaborado por Maitha et al (2011), no qual, os autores propõem 11 (onze) modelos diferentes de redes neurais para a predição de irradiação média tendo como dados para o treinamento das redes neurais, a temperatura média, a hora de isolamento diária, a umidade e a pressão. A arquitetura das redes neurais MLP definidas neste trabalho se descreve a seguir no capítulo 3.

3 ARQUITETURA DA REDE NEURAL PARA OS MODELOS

PREDITIVOS

3.1 Introdução

Neste capítulo serão apresentados os principais conceitos acerca da arquitetura da rede neural do tipo *perceptron* multicamada com algoritmos de propagação recursiva do erro (*backpropagation*) adotada neste trabalho. Serão também abordados a definição dos modelos neurais preditivos de velocidade de vento bem como os índices quantitativos utilizados para qualificar o desempenho dos modelos adotados.

3.2 Rede Neural Multicamada com *Backpropagation*

Como visto no capítulo 2, a rede neural do tipo *perceptron* com multicamadas ou MLP, é bastante adequada para a predição de velocidade de vento, e por esta razão, foi adotada para os modelos preditivos que serão descritos mais adiante.

Em regra, as redes neurais MLP são as redes constituídas de uma ou mais camadas intermediárias ou escondidas, além da camada de entrada e da camada de saída. As camadas intermediárias fornecem à rede MLP a capacidade de resolver problemas não lineares, uma vez que os primeiros tipos de RNA, como as redes *Perceptron* e *Adaline*, por exemplo, não possuem esse poder de generalização. Tal diferença acontece devido ao fato das camadas ocultas funcionarem como extratoras de características, no qual, os pesos apresentam uma codificação de tais características que são apresentadas nos padrões de entrada, e essas características funcionam de modo a permitir que a rede neural crie sua própria representação, mais complexa e rica. Com efeito, a camada de saída tem como finalidade construir o padrão de saída a partir dos dados fornecidos pelas camadas intermediárias, escondidas ou ocultas (HENRIQUE, 2014). Um esquema simplificado de uma rede neural MLP é mostrado na Figura 3.

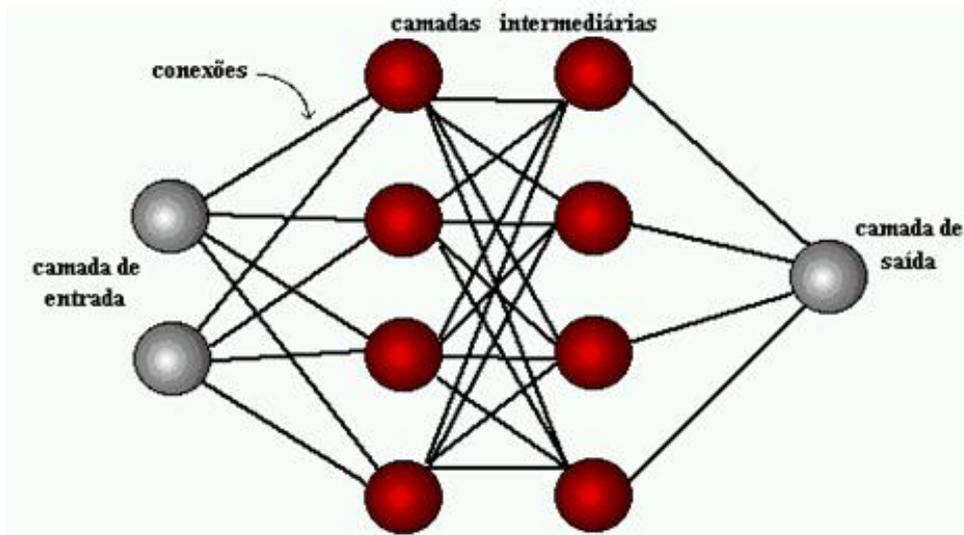


Figura 3 – Esquema de uma rede multi-layer perceptron (MLP)

Fonte: (CARVALHO, 2010)

Uma rede neural MLP, possui três características principais:

- Os neurônios das camadas ocultas, possuem uma função de ativação não-linear do tipo sigmoideal, função logística ou tangente hiperbólica e os neurônios da camada de saída possui uma função de ativação linear.
- A rede possui uma ou mais camadas escondidas
- A rede possui um alto grau de conexão

Na rede neural MLP são identificados dois tipos de sinais:

- Sinal funcional: Ele é um sinal de entrada, um estímulo, que sai a partir do terminal de entrada da rede, propaga-se para frente, neurônio por neurônio, através da rede neural, e chega ao terminal de saída da rede neural como um sinal de saída. O sinal funcional também é chamado de sinal de entrada.
- Sinal de erro: Ele se origina em um neurônio de saída da rede e seu sentido de propagação é o inverso do sentido do sinal funcional, camada por camada, através da rede. Ele é assim chamado devido a sua computação por cada neurônio da rede neural envolver uma função dependente do erro. (HAYKIN, 2000)

Logo abaixo, segue um esquema de uma rede neural MLP com os sentidos dos sinais funcionais e de erro:

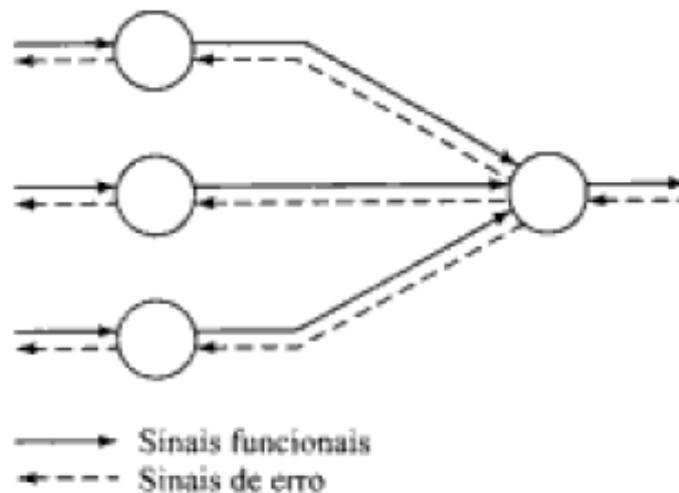


Figura 4 – Esquema de uma rede que mostra a direção da propagação dos sinais funcionais e da retropropagação dos sinais de erro

Fonte: (HAYKIN, 2000)

3.2.1 Algoritmo *BackPropagation*

Atualmente, existem diversos algoritmos computacionais de treinamento de redes neurais MLP. Eles são geralmente do tipo supervisionado, que é o método mais comum de aprendizado de treinamento para redes neurais artificiais. Eles são assim chamados porque a entrada e a saída desejadas para a rede neural são dadas por um supervisor externo, cujo objetivo é ajustar os parâmetros para encontrar uma relação entre os pares de entrada e saída fornecidos. Os algoritmos para treinamento das redes MLP ainda podem ser classificados, através dos parâmetros que eles atualizam, em estáticos e dinâmicos.

Os algoritmos estáticos não mudam a estrutura da rede, como o próprio nome já indica, embora varie os valores de seus pesos. Os algoritmos dinâmicos podem alterar a estrutura da rede, aumentando ou diminuindo o tamanho da rede, através dos números de camadas, números de neurônios nas camadas ocultas e número de conexões.

O algoritmo de aprendizado mais conhecido e utilizado para treinar redes neurais artificiais é o algoritmo *backpropagation*. Seu método de aprendizado é do tipo supervisionado, que utiliza pares de entrada e saída desejados para ajustar os pesos da rede através de um mecanismo de correção de erros.

O treinamento da rede neural MLP utilizando o algoritmo *backpropagation* ocorre em duas fases: A fase *forward* e a fase *backward*. A primeira é utilizada para definir a saída a

partir de um dado padrão de entrada. A segunda atualiza os pesos de suas conexões a partir da saída desejada e fornecida pela rede neural.

Este algoritmo funciona através dos seguintes passos:

- a) Inicialização dos parâmetros e pesos
- b) Repetição do algoritmo até o erro ser o menor desejado ou até a realização de um determinado número de ciclos
 - b.1) Para cada tipo de treinamento X
 - b.1.1) definir a saída da rede por meio da fase *forward*
 - b.1.2) comparar as saídas produzidas com as saídas desejadas
 - b.1.3) atualizar pesos de nodos através da fase *backward* (BRAGA, LUDERMIR & CARVALHO, 2000).

Na Figura 5 é ilustrado o esquema que mostra o funcionamento da rede neural artificial através do algoritmo *backpropagation*:

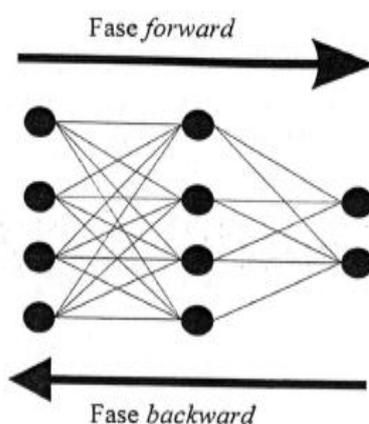


Figura 5 – Esquema que mostra o fluxo de processamento do algoritmo *backpropagation*.

Fonte: (BRAGA, LUDERMIR & CARVALHO, 2000).

As redes neurais que utilizam o algoritmo *backpropagation* trabalham com a regra delta generalizada, que é caracterizada por uma variação apropriada para as redes com multicamadas. O treinamento das redes neurais com multicamadas que utiliza o algoritmo *backpropagation* pode resultar em muitos passos, que demanda um tempo muito grande de treinamento. Para que o treinamento seja mais rápido, o ideal seria aumentar a taxa de aprendizado, definida pela variável α .

A finalidade do algoritmo *backpropagation* é de minimizar o erro obtido pela rede neural, ajustando os pesos e limiares, para que fiquem nos pontos mais inferiores das coordenadas da superfície de erro, utilizando o método de gradiente descendente, em que a

função de ativação utilizada precisa ser contínua, diferenciável e não-decrescente e a mais utilizada é a sigmoideal. Porém, quanto maior o gradiente descendente, maior será também a mudança dos pesos, o que faz aumentar a velocidade de aprendizado, podendo causar uma oscilação do modelo (BITENCOURT, 2015).

A função que calcula o erro também precisa ser diferenciável. Os pesos que minimizam o erro são encontrados a partir da derivada da função de erro e o cálculo de delta (sensibilidade) utiliza a derivada das funções de ativação.

A partir da sensibilidade de cada neurônio obtém-se os novos valores de pesos. A sensibilidade δ para a camada de saída é dada pela seguinte equação (VERAS, 2013):

$$\delta_j = (d_j - y_j)f'(net_j) \quad (3)$$

Em que:

- d_j corresponde à saída desejada para as entradas fornecidas;
- y_j representa a saída obtida pelo mesmo neurônio com as mesmas entradas;
- $f'(net_j)$ é a derivada da função de ativação do neurônio de saída.

Para as outras camadas, a sensibilidade para cada neurônio j da respectiva camada é calculado com a seguinte expressão (VERAS, 2013):

$$\delta_j = f'(net_j) \sum_i \delta_i w_{i,j} \quad (4)$$

Em que:

- $f'(net_j)$ é a derivada da função de ativação do neurônio da camada;
- δ_i corresponde a sensibilidade propagada pelo i -ésimo neurônio da camada a frente; $w_{i,j}$ representa o peso j do neurônio i .

Para aumentar a taxa de aprendizado, aumentando também o treinamento da rede e a sua velocidade conseqüentemente, sem oscilar o sistema, é necessário alterar a regra delta generalizada incluindo o termo momentum (β), que é uma constante que determina o efeito das alterações passadas dos pesos com relação aos pesos atuais.

No processo de aprendizado, os pesos de suas conexões, na sua regra de treinamento, são ajustados através de iterações. Este treinamento é realizado através dos dados realizados, no qual a rede aprende de seu ambiente, melhorando o seu desempenho.

O ajuste de pesos a cada iteração t é dada pela seguinte expressão (VERAS, 2013):

$$\Delta w_{i,j}(t + 1) = w_{i,j}(t) + \alpha \delta_j x_i + \beta (w_{i,j}(t) - w_{i,j}(t - 1)) \quad (5)$$

Em que:

- $\Delta w_{i,j}(t + 1)$ refere-se ao novo valor dado ao i-ésimo peso do neurônio j;
- $w_{i,j}(t)$ refere-se ao valor do i-ésimo peso do neurônio j no instante t da iteração;
- α é a taxa de aprendizado;
- δ_j é a sensibilidade;
- x_i é o valor do sinal de neurônio;
- β é o *momentum*;
- $w_{i,j}(t - 1)$ é o valor do peso de índice i do neurônio j no instante $t - 1$.

O treinamento da rede neural é a propriedade mais importante no qual a rede neural adquire a forma adequada para executar o processo dos dados fornecidos. É este processo de treinamento que faz com que a rede neural seja diferente da computação programada, onde este utiliza um conjunto de regras rígidas pré-fixadas.

O algoritmo de aprendizado pode ser definido como um conjunto de regras para a solução de um problema. Para a aplicação às redes neurais artificiais, é preciso passar por alguns passos. Os dois primeiros passos são: a coleta dos dados e a sua separação em dados de treinamento e dados de teste, para verificar o desempenho da rede. Depois é criado um conjunto de validação que verifica a eficiência da rede quanto a sua capacidade de generalização durante o treinamento, empregado como critério de parada de treinamento, segundo o qual eles são separados de forma aleatória de modo a evitar erros associados à ordem dos dados.

O terceiro passo é a configuração da rede, que segue três etapas:

- Seleção do modelo neural apropriado à aplicação;
- Determinação do tipo da rede a ser utilizada, ou seja, colocar o número de camadas, o número de neurônios em cada camada, etc.;
- Determinação dos parâmetros do algoritmo de treinamento e das funções de ativação.

O quarto passo é o treinamento da rede neural, no qual são ajustados os pesos através do algoritmo escolhido. É considerado nesta etapa algumas características, como:

- A inicialização da rede: Deve-se escolher valores iniciais para os pesos da rede, que diminuem ou aumentam o tempo de treinamento da rede. Se os

valores dos pesos forem escolhidos de forma errada, pode acontecer uma saturação prematura da rede. Existem diversos métodos para escolha dos valores iniciais dos pesos, cada um dependendo da aplicação da rede neural.

- O modo de treinamento: Existem dois modos de treinamento, o modo padrão e o modo *batch*. No modo padrão, há várias correções dos pesos a cada ciclo da rede e o erro é apresentado naquela iteração. Este é o modo mais utilizado devido ao menor armazenamento de dados e por ser menos suscetível ao problema de mínimos locais. Porém, com o modo *batch*, a cada ciclo é feita apenas uma correção, com um treinamento mais estável e uma melhor estimativa do vetor gradiente. Assim, pode-se escolher um dos dois modelos para treinamento da rede, dependendo do problema a ser tratado;
- Tempo de treinamento: Deve-se escolher o critério de parada do treinamento, considerando a taxa de erro médio por ciclo e a capacidade de generalização da rede. A interrupção do treinamento da rede para o algoritmo *backpropagation* irá ocorrer quando a rede tiver uma boa capacidade de generalização e quando o erro for bem pequeno e dentro dos padrões. Logo, deve-se encontrar um ponto ótimo de parada com erro mínimo e capacidade de generalização máxima.

O quinto passo é o teste da rede, no qual se determina o desempenho da rede utilizando um conjunto de dados de teste.

O sexto e último passo é a integração da rede com um sistema do ambiente operacional da aplicação com o objetivo de melhorar o desempenho da rede, fazendo a manutenção da rede, podendo refazer o treinamento para obter um melhor resultado.

Os índices de desempenho mais utilizados em redes neurais que utilizam a previsão são calculados em função do erro de previsão (BITENCOURT, 2015):

$$e(t) = y(t) - \hat{y}(t) \quad (6)$$

No qual :

$y(t)$: valor real de vento para o período de tempo t

$\hat{y}(t)$: valor de previsão de vento para o mesmo período

$e(t)$: erro de previsão de vento

Os índices de desempenho utilizados neste trabalho para qualificar o desempenho das redes neurais são apresentados a seguir:

- Erro Quadrático Médio, MSE, do inglês *Mean Squared Error* (BITENCOURT, 2015)

$$MSE = \frac{1}{n} \sum_{t=1}^n (e(t))^2 \quad (7)$$

- Erro Percentual Absoluto Médio, MAPE, do inglês *Mean Absolute Percentage Error* (BITENCOURT, 2015)

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{e(t)}{y(t)} \right| * 100 \quad (8)$$

- Coeficiente de Correlação de Pearson (ALVES & VECCHIA, 2011)

$$R = \sqrt{\frac{\sum_{i=1}^n (\hat{y}(t) - \bar{y}(t))^2}{\sum_{i=1}^n (y(t) - \bar{y}(t))^2}} \quad (9)$$

Em que $\bar{y}(t)$ é o valor médio de vento e n é o número de períodos de tempo.

O *MSE* é o erro quadrático médio, ou seja, a média do quadrado do desvio entre o valor de previsão e o valor desejado. Ele é considerado uma medida importante para escolher um modelo para a rede, pois um *MSE* grande resulta em um peso grande para a média e um *MSE* pequeno resulta em um erro menor. Logo, o melhor modelo é o que tiver o menor *MSE*, será o mais preciso.

O *MAPE* é o erro percentual absoluto médio, que é a média de todos os erros absolutos percentuais, o qual fornece o valor do tamanho médio do erro, expresso como porcentagem do valor indicado, independente do erro ser positivo ou negativo. Por exemplo, se o *MAPE* for 7, em média, a sua previsão está incorreta em 7%. Portanto, quanto maior for o *MAPE*, mais incorreta é a previsão e quanto menor for o *MAPE*, mais correta será.

O *R* é o coeficiente de correlação de Pearson, que mede o grau de correlação e a sua direção. Ele corresponde a valores entre 0 e 1. O valor 1 significa correlação positiva perfeita entre as duas variáveis e o valor 0 determina a ausência de correlação. Para os valores que são próximos de zero, significa que a correlação é muito pequena, há um baixo relacionamento entre as variáveis e para valores próximos de 1, significa uma boa correlação.

3.3 Estrutura dos Modelos Preditivos

Os dados de entrada para as redes neurais MLP com *backpropagation* utilizados neste trabalho foram: hora, temperatura, umidade e pressão. Estes dados reais foram coletados da estação meteorológica do CEAR relativos aos anos de 2009, 2010, 2011 e 2012. As horas de medição adotadas por dia para essas demais grandezas durante esses anos corresponderam a 0:00 h, 12:00 h e 18:00 h. Juntamente com essas grandezas, foi também coletada a velocidade de vento para cada hora medida ao longo desses anos. Essa grandeza, por sua vez, foi adotada como dado de saída para a rede neural.

Os dados coletados estão disponíveis em planilha do Excel e foram copiados para arquivo de texto em extensão.txt para serem implementados no código do software MATLAB®.

A normalização dos dados de treinamento da rede neural artificial foi feita com o objetivo de obter um melhor desempenho de previsão para a velocidade do vento, no qual foi utilizada a função *mapminmax* do MATLAB® que coloca as variáveis de entrada e as variáveis de saída no intervalo entre um negativo e um positivo. Esta estratégia garante que as variáveis utilizadas na entrada da rede neural e no treinamento tenham o mesmo grau de importância.

Neste trabalho foram desenvolvidos 11 modelos de previsão de velocidade de vento baseados em redes neurais MLP. Os dados utilizados como entrada e saída para esses modelos estão resumidos na Tabela 1.

Tabela 1 - Modelos da Rede Neural para previsão de velocidade de vento

Modelos Preditivos das Redes Neurais	Dados de Entrada	Dado de Saída
Modelo 1	Hora, Temperatura, Umidade, Pressão	Velocidade de Vento
Modelo 2	Hora, Temperatura, Umidade	
Modelo 3	Hora, Temperatura, Pressão	
Modelo 4	Hora, Umidade, Pressão	
Modelo 5	Temperatura, Umidade, Pressão	
Modelo 6	Hora, Temperatura	
Modelo 7	Hora, Umidade	
Modelo 8	Hora, Pressão	
Modelo 9	Temperatura, Umidade	
Modelo 10	Temperatura, Pressão	

Modelo 11	Umidade, Pressão	
-----------	------------------	--

Fonte: Do autor

Para todos os modelos preditivos de velocidade de vento foram adotados os dados relativos aos anos de 2009, 2010 e 2011 para seus treinamentos e validação, enquanto que nos testes de desempenho dos referentes modelos foram utilizados os dados referentes ao ano de 2012. Por outro lado, utilizou-se o algoritmo *Levenberg – Marquardt* para o treinamento da rede em todos os casos, cuja função correspondente no MATLAB[®] consiste na função *trainlm*. A sequência de estruturação dos modelos preditivos é resumida no fluxograma da Figura 6.

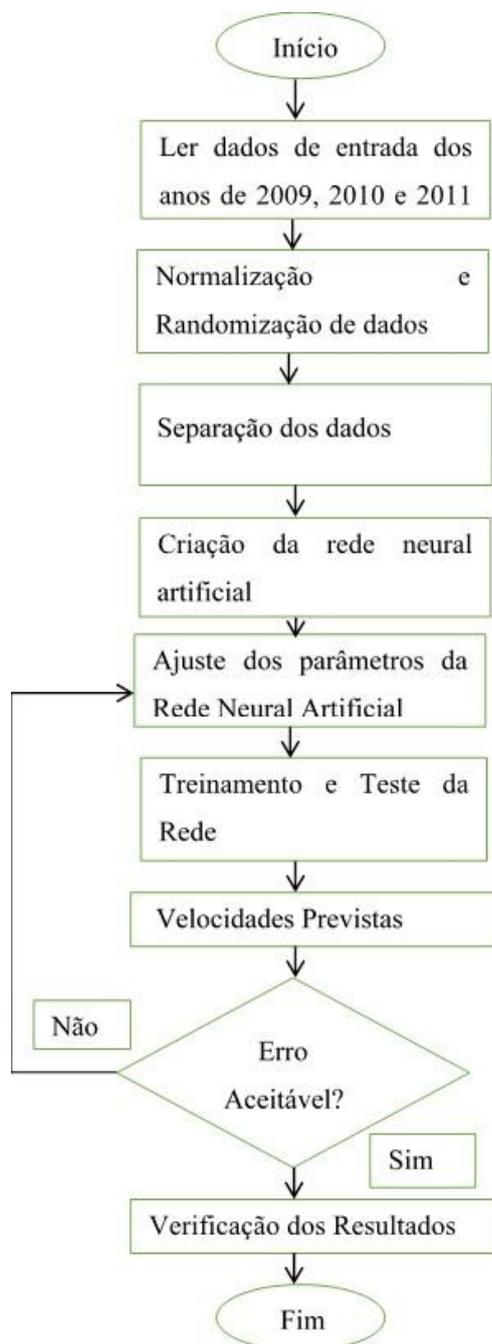


Figura 6 – Fluxograma da estrutura dos modelos preditivos desenvolvida pelo autor

Fonte: Do Autor

O fluxograma da Figura 6 apresenta sucintamente os passos realizados no desenvolvimento de programas em código MATLAB[®] para o treinamento, validação e teste dos modelos preditivos. Parte desses códigos são apresentados no Anexo I. Os resultados dos desempenhos desses modelos são apresentados no capítulo 4 a seguir.

4 ANÁLISE DE RESULTADOS

4.1 Introdução

Neste capítulo serão apresentados os resultados de desempenho dos modelos preditivos de velocidade de vento abordados no capítulo anterior. Para todos os modelos foram determinados os índices quantitativos, pelos quais, pode-se destacar o modelo de melhor desempenho.

4.2 Análise Qualitativa e Quantitativa

Conforme apresentado no capítulo 3, o autor desse trabalho utilizou como estrutura de rede neural para os modelos preditivos de velocidade de vento, a rede neural artificial do tipo MLP (*Multilayer Perceptron*) com algoritmo *backpropagation*, no qual foi utilizado o algoritmo de treinamento *Levenberg – Marquardt*. Os dados de entrada para o treinamento e validação dos modelos de predição são apresentados na Figura 7. Vale ressaltar que das medições de temperatura, umidade relativa do ar e pressão foram coletados em intervalos de medição diários de 0:00 h, 12:00 hs e 18:00 hs.

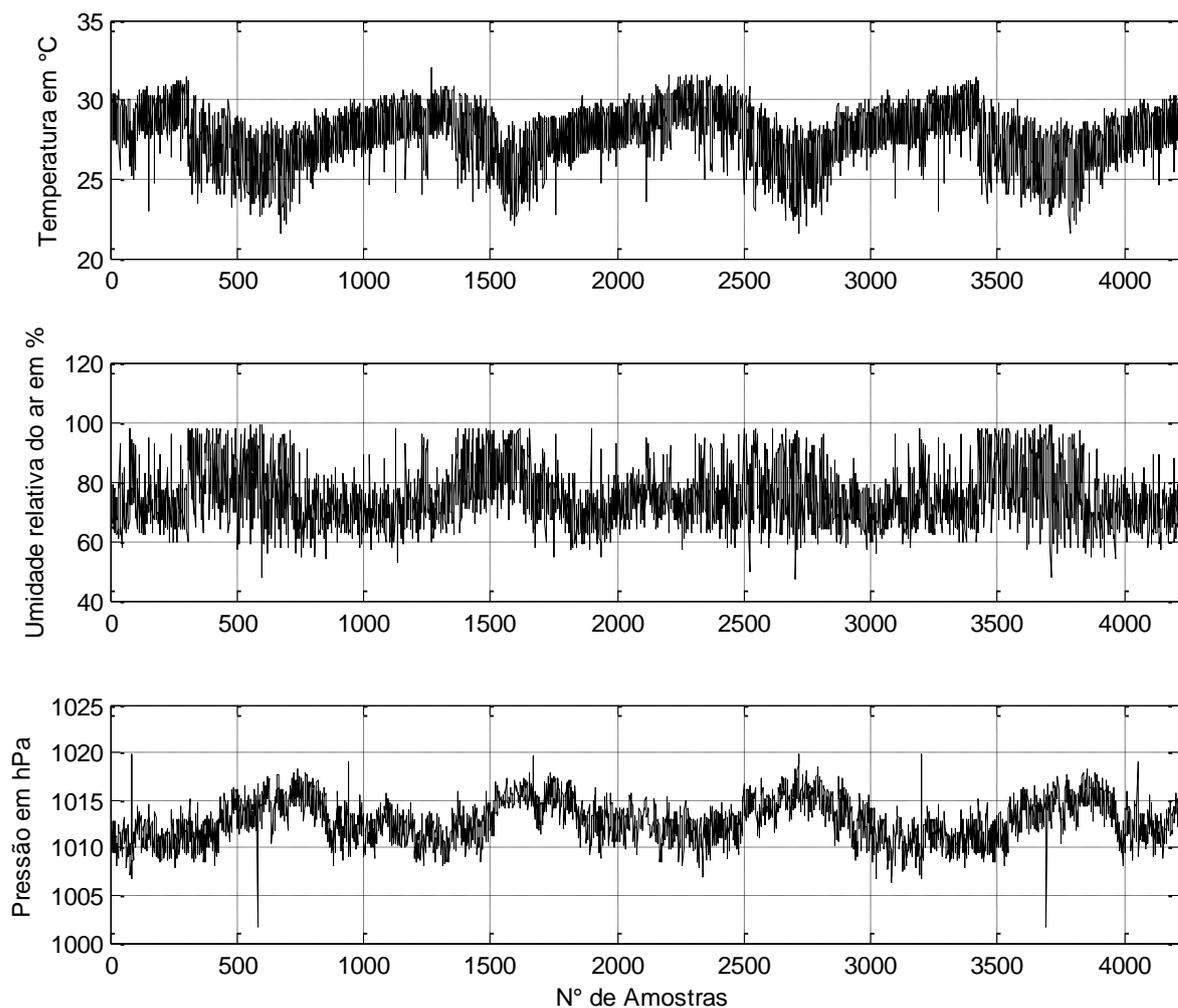


Figura 7 – Dados de entrada para o treinamento e validação dos modelos preditivos de velocidade de vento entre os anos de 2009 a 2011.

Fonte: Do autor

O autor do trabalho também organizou uma tabela contendo os 11 diferentes modelos preditivos de redes neurais, com a estrutura de cada modelo, o valor do coeficiente de correlação R para cada um, o valor do erro quadrático médio MSE por modelo, o erro percentual absoluto médio MAPE, a função de ativação nas camadas ocultas e o destaque para o melhor modelo de rede neural esperado. Esta tabela segue logo abaixo:

Tabela 2 - Parâmetros da rede MLP do tipo *backpropagation* para diferentes modelos da Rede Neural. A linha destacada corresponde ao melhor modelo para a rede neural

Modelo	Estrutura Neural	R	MSE	MAPE	Função de Ativação
1	4 - 25 - 35 - 25 - 1	0,8329	0,0579	7,1074	Tangente Hiperbólico
2	3- 25 - 25 - 1	0,8551	0,0843	8,5413	Tangente Hiperbólico
3	3- 100 - 1	0,7944	0,0723	7,6271	Tangente Hiperbólico
4	3- 35 - 40 - 35 - 1	0,7545	0,0644	7,7446	Tangente Hiperbólico
5	3- 35 - 35 - 1	0,7916	0,0482	6,6847	Tangente Hiperbólico
6	2 - 100 - 1	0,8665	0,0860	8,8173	Tangente Hiperbólico
7	2 - 25 - 35 - 1	0,7985	0,0668	7,8750	Tangente Hiperbólico
8	2 - 30 - 35 - 1	0,8510	0,0584	7,0220	Tangente Hiperbólico
9	2 - 45 - 50 - 45 - 1	0,8355	0,0629	7,7451	Tangente Hiperbólico
10	2- 35 - 40 - 35 - 1	0,7984	0,0879	8,1528	Tangente Hiperbólico
11	2- 30 - 35 - 30 - 1	0,8449	0,0512	6,9834	Tangente Hiperbólico

Fonte: Do Autor

Pela análise dos resultados na Tabela 2, pode-se afirmar para cada um dos modelos o seguinte:

Modelo 1:

Neste modelo os dados de entrada para a rede neural MLP envolveram a hora de medição, a temperatura, a umidade e a pressão. Foram treinadas oito redes, com diferentes camadas escondidas e número de neurônios. Contudo, a que apresentou melhor resultado foi a rede neural (4-25-35-21-1), ou seja, 4 neurônios na camada de entrada, 25, 35 e 21 neurônios na primeira, segunda e terceira camadas escondidas, respectivamente, e 1 neurônio na camada de saída. A rede apresentou como erro de validação o valor de 0,11208 e erro de treino de 0,0844, e um total de 12 iterações. Para as camadas de entrada e escondidas foram utilizadas como função de ativação a função tangente hiperbólica (função *tansig*, no MATLAB®) e para a camada de saída foi utilizada a função linear *purelin*. Esta configuração de rede foi a que apresentou o menor erro MSE encontrado (MSE = 0,0579), as demais redes treinadas resultaram em um valor maior para MSE. O coeficiente de correlação de R resultou em um valor próximo de 1 (R = 0,8329), o que indica uma boa correlação e o Erro Percentual Absoluto Médio, MAPE teve como valor 7,1074.

Modelo 2:

Para este modelo foram adotadas como dados de entrada para o modelo neural a hora de medição, a temperatura e a umidade. Foram treinadas 13 redes, com diferentes números de camadas ocultas, diferentes quantidades de neurônios relativos a essas camadas. A rede deste modelo que teve a melhor configuração foi a rede neural (3-25-25-1), ou seja, com 3 neurônios na camada de entrada, 25 e 25 neurônios na primeira e segunda camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,063857 e o erro de treino foi de 0,0594. Um total de 12 iterações, com as mesmas funções de ativação que o modelo 1 foi realizado, em que se obteve um $MSE = 0,0843$, sendo o menor erro encontrado. O valor de $R = 0,8551$ foi o mais próximo de 1 para esta configuração de rede e o valor do Erro Percentual Absoluto Médio foi $MAPE = 8,5413$.

Modelo 3:

Neste modelo foram utilizadas como dados de entrada para a RNA a hora de medição, a temperatura e a pressão. Foram treinadas 13 estruturas de redes, com diferentes números de camadas ocultas e de quantidades de neurônios dessas camadas. A rede deste modelo que teve a melhor configuração foi a rede neural (3- 100 -1), ou seja, com 3 neurônios na camada de entrada, 100 neurônios na oculta e 1 neurônio na camada de saída. O erro de validação foi de 0,10428 e o erro de treino de 0,0573. Um total de 11 iterações, com as mesmas funções de ativação do modelo 1 foram realizadas obtendo-se um $MSE = 0,0723$, um $R = 0,7944$ e valor do Erro Percentual Absoluto Médio foi $MAPE = 7,6271$.

Modelo 4:

Utilizou-se como dados de entrada para a rede neural a hora de medição, a umidade e pressão. Foram treinadas 8 redes, com diferentes números de camadas ocultas e diferentes quantidades de neurônios relativos a essas camadas. O modelo que teve a melhor configuração foi a rede neural (3-35-40-35-1), ou seja, com 3 neurônios na camada de entrada, 35, 40 e 35 neurônios na primeira, segunda e terceira camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,075382 e o erro de treino foi de 0,0687 após um total de 14 iterações do processo de treinamento. As funções de ativação para as camadas de entrada, oculta e de saída são iguais aquelas do modelo 1. O valor de MSE correspondeu a 0,0644, sendo o menor erro encontrado dentre as demais redes treinadas para este modelo. O valor de R foi de 0,7545 e o MAPE correspondeu a 7,7446.

Modelo 5:

Para este modelo foram utilizados como dados de entrada para o modelo neural a temperatura, a umidade e a pressão. Foram treinadas 13 redes, com diferentes números de camadas ocultas, diferentes quantidades de neurônios relativos a essas camadas. A rede deste modelo que teve a melhor configuração foi a rede neural (3-35-35-1), ou seja, com 3 neurônios na camada de entrada, 35 e 35 neurônios na primeira e segunda camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,069036 e o erro de treino foi de 0,0606. Um total de 13 iterações, com as mesmas funções de ativação que o modelo 1 foi realizado, em que se obteve um $MSE = 0,0482$, o menor erro encontrado não só neste modelo, como também comparado com todos os outros modelos. O valor de $R = 0,7916$ foi o mais próximo de 1 para esta configuração de rede e o valor do Erro Percentual Absoluto Médio foi $MAPE = 6,6847$. Este foi o melhor modelo para a rede neural artificial.

Modelo 6:

Utilizou-se como dados de entrada para a rede neural a hora de medição e a temperatura. Foram treinadas 12 redes, com diferentes números de camadas ocultas e diferentes quantidades de neurônios relativos a essas camadas. O modelo que teve a melhor configuração foi a rede neural (2-100-1), ou seja, com 2 neurônios na camada de entrada, 100 neurônios na camadas oculta, e 1 neurônio na camada de saída. O erro de validação foi de 0,087484 e o erro de treino foi de 0,1052 após um total de 10 iterações do processo de treinamento. As funções de ativação para as camadas de entrada, oculta e de saída são iguais aquelas do modelo 1. O valor de MSE correspondeu a 0,0860, sendo o menor erro encontrado dentre as demais redes treinadas para este modelo. O valor de R foi de 0,8665 e o MAPE correspondeu a 8,8173.

Modelo 7:

Neste modelo foram utilizadas como dados de entrada para a RNA a hora de medição e a umidade. Foram treinadas 8 estruturas de redes, com diferentes números de camadas ocultas e de quantidades de neurônios dessas camadas. A rede deste modelo que teve a melhor configuração foi a rede neural (2-25-35-1), ou seja, com 2 neurônios na camada de entrada, 25 e 35 neurônios na primeira e segunda camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,071598 e o erro de treino de 0,0704. Um total de 13 iterações, com as mesmas funções de ativação do modelo 1 foram

realizadas obtendo-se um $MSE = 0,0668$, um $R = 0,7985$ e valor do Erro Percentual Absoluto Médio foi $MAPE = 7,8750$.

Modelo 8:

Utilizou-se como dados de entrada para a rede neural a hora de medição e a pressão. Foram treinadas 8 redes, com diferentes números de camadas ocultas e diferentes quantidades de neurônios relativos a essas camadas. O modelo que teve a melhor configuração foi a rede neural (2-30-35-1), ou seja, com 2 neurônios na camada de entrada, 30 e 35 neurônios na primeira e segunda camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,076868 e o erro de treino foi de 0,0734 após um total de 11 iterações do processo de treinamento. As funções de ativação para as camadas de entrada, oculta e de saída são iguais aquelas do modelo 1. O valor de MSE correspondeu a 0,0584, sendo o menor erro encontrado dentre as demais redes treinadas para este modelo. O valor de R foi de 0,8510 e o MAPE correspondeu a 7,0220.

Modelo 9:

Para este modelo foram adotadas como dados de entrada para o modelo neural a temperatura e a umidade. Foram treinadas 8 redes, com diferentes números de camadas ocultas, diferentes quantidades de neurônios relativos a essas camadas. A rede deste modelo que teve a melhor configuração foi a rede neural (2-45-50-45-1), ou seja, com 2 neurônios na camada de entrada, 45, 50 e 45 neurônios na primeira, segunda e terceira camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,068667 e o erro de treino foi de 0,0645. Um total de 16 iterações, com as mesmas funções de ativação que o modelo 1 foi realizado, em que se obteve um $MSE = 0,0629$, sendo o menor erro encontrado. O valor de $R = 0,8355$ foi o mais próximo de 1 para esta configuração de rede e o valor do Erro Percentual Absoluto Médio foi $MAPE = 7,7451$.

Modelo 10:

Neste modelo foram utilizadas como dados de entrada para a RNA a temperatura e a pressão. Foram treinadas 12 estruturas de redes, com diferentes números de camadas ocultas e

de quantidades de neurônios dessas camadas. A rede deste modelo que teve a melhor configuração foi a rede neural (2-35-40-35-1), ou seja, com 2 neurônios na camada de entrada, 35, 40 e 35 neurônios na primeira, segunda e terceira camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,07458 e o erro de treino de 0,0760. Um total de 18 iterações, com as mesmas funções de ativação do modelo 1 foram realizadas obtendo-se um MSE = 0,0879, um R = 0,7984 e valor do Erro Percentual Absoluto Médio foi MAPE = 8,1528.

Modelo 11:

Por fim, utilizou-se como dados de entrada para esta rede neural a umidade e a pressão. Foram treinadas 8 redes, com diferentes números de camadas ocultas e diferentes quantidades de neurônios relativos a essas camadas. O modelo que teve a melhor configuração foi a rede neural (2-30-35-30-1), ou seja, com 2 neurônios na camada de entrada, 30, 35 e 30 neurônios na primeira, segunda e terceira camadas ocultas, respectivamente, e 1 neurônio na camada de saída. O erro de validação foi de 0,083751 e o erro de treino foi de 0,0689 após um total de 10 iterações do processo de treinamento. As funções de ativação para as camadas de entrada, oculta e de saída são iguais aquelas do modelo 1. O valor de MSE correspondeu a 0,0512, sendo o menor erro encontrado dentre as demais redes treinadas para este modelo. O valor de R foi de 0,8449 e o MAPE correspondeu a 6,9834.

Com base no resultado da tabela 2, os 5 melhores modelos obtidos foram: em 5º lugar, o modelo 9 com MSE = 0,0629; em 4º lugar ficou o modelo 8, com um MSE = 0,0584; em 3º lugar, com MSE = 0,0579 foi o modelo 1; em 2º lugar ficou o modelo 11, com MSE = 0,0512 e por fim, em 1º lugar, o melhor modelo foi o modelo 5, com MSE = 0,0482.

Uma vez definidos os 11 modelos preditivos, o passo seguinte foi o de avaliá-los para dados, pelos quais, a rede não foi treinada. Os dados utilizados foram os dados relativos ao ano 2012. Para isto, foram calculados os valores médios de velocidade de vento reais de cada mês do ano de 2012 e comparados com os valores de velocidades de vento médios mensais estimados pelos modelos de predição. Os cinco melhores desempenhos são mostrados nas Figuras 8 a 12, cujos dados numéricos de entrada (para o 1º dia de Janeiro de 2012) e de saída (velocidade de vento em m/s) podem ser melhor visualizados nas tabelas 3 a 12.

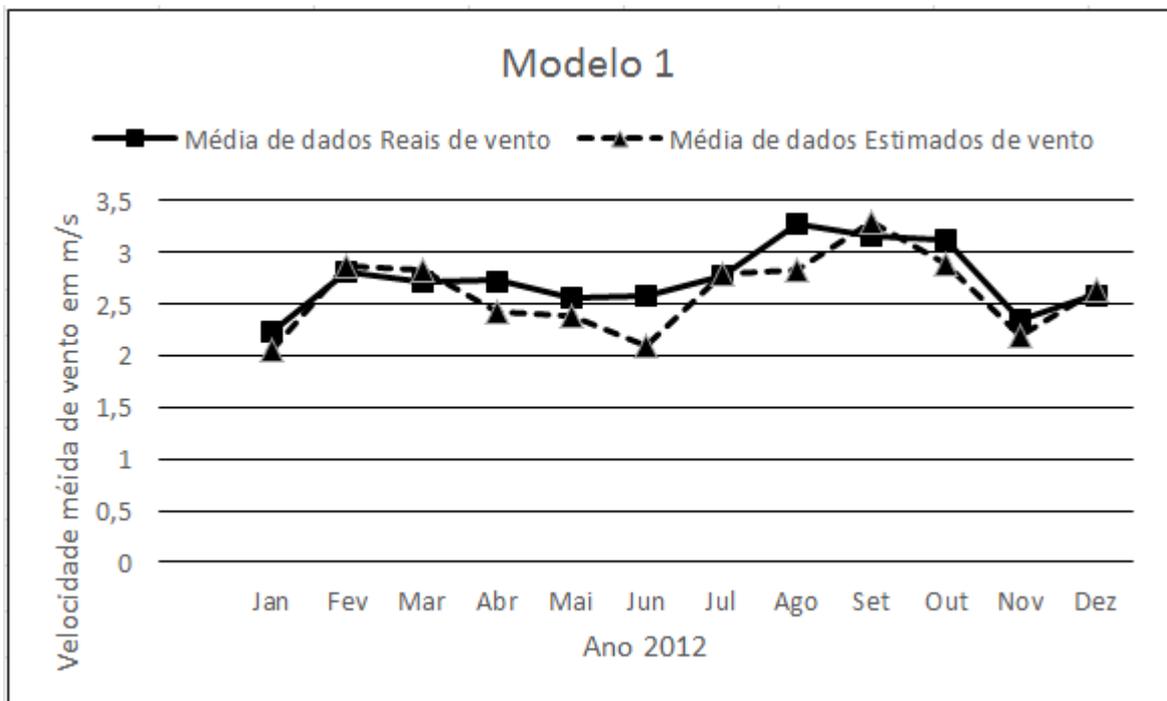


Figura 8 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 1 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 3 - Média dos dados reais e estimados de vento para o modelo 1 da Rede Neural Artificial

Velocidade Média de Vento		
Meses de 2012	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,0577
Fev	2,81	2,8709
Mar	2,72	2,8309
Abr	2,73	2,4224
Mai	2,55	2,3878
Jun	2,58	2,0953
Jul	2,78	2,7914
Ago	3,27	2,8337
Set	3,15	3,3048
Out	3,12	2,894
Nov	2,35	2,1877
Dez	2,59	2,6404

Fonte: Do Autor

Tabela 4 – Dados de entrada utilizados para o modelo 1

Hora (h)	Temperatura (°C)	Umidade (%)	Pressão (hPa)
0	27,2	76	1012,5
12	29,4	70	1014,3
18	28,8	69	1012,2

Fonte: Do Autor

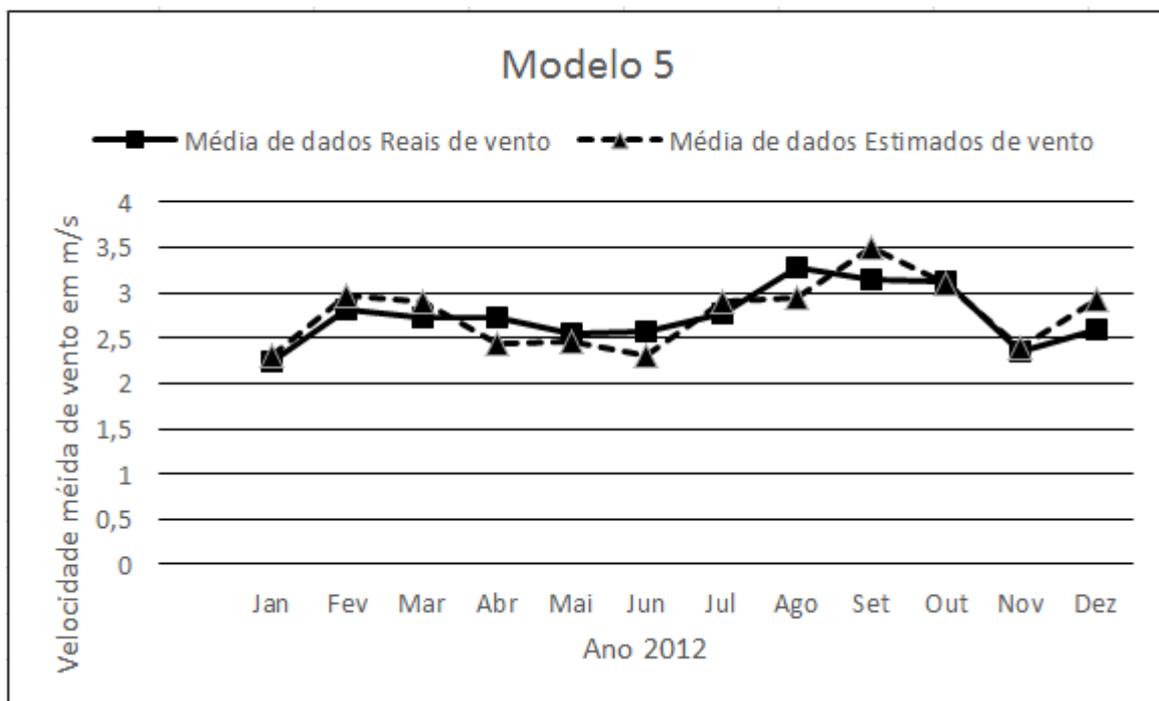


Figura 9 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 5 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 5 - Média dos dados reais e estimados de vento para o modelo 5 da Rede Neural Artificial

Velocidade Media de Vento		
Meses de 2012	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,3065
Fev	2,81	2,9709
Mar	2,72	2,9064
Abr	2,73	2,4333
Mai	2,55	2,4697
Jun	2,58	2,3165
Jul	2,78	2,9002
Ago	3,27	2,9409
Set	3,15	3,4911
Out	3,12	3,0951
Nov	2,35	2,389
Dez	2,59	2,9252

Fonte: Do Autor

Tabela 6 – Dados de entrada utilizados para o modelo 5

Temperatura (°C)	Umidade (%)	Pressão (hPa)
27,2	76	1012,5
29,4	70	1014,3
28,8	69	1012,2

Fonte: Do Autor

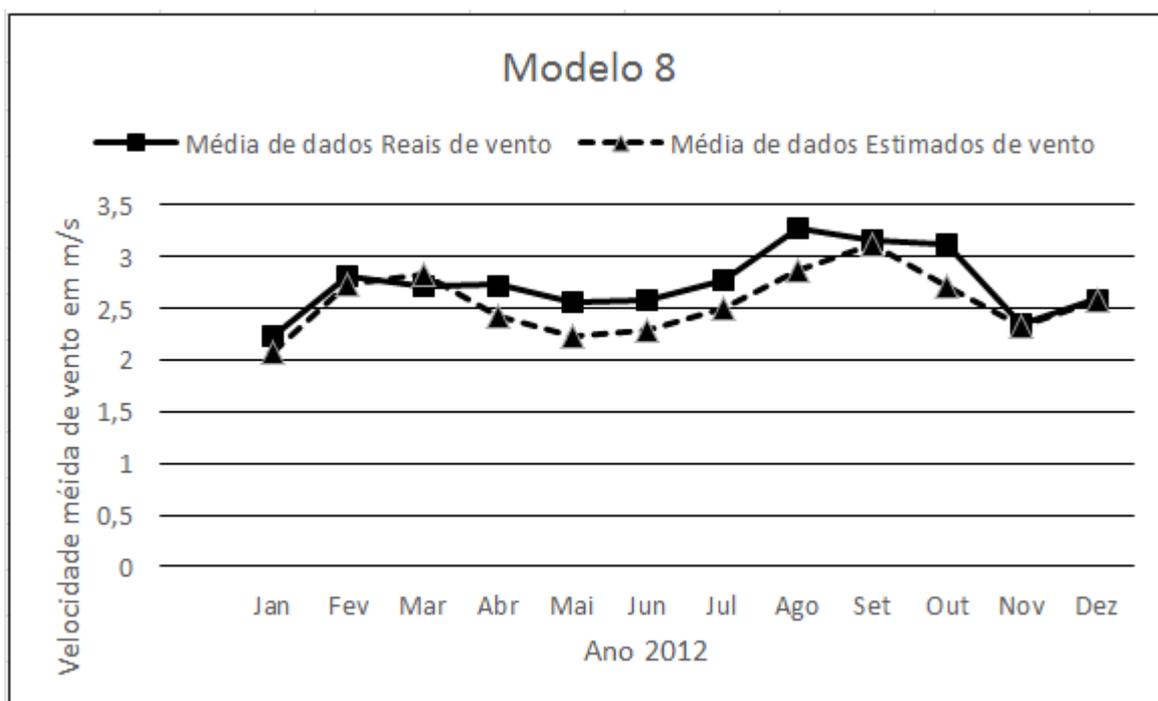


Figura 10 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 8 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 7 - Média dos dados reais e estimados de vento para o modelo 8 da Rede Neural Artificial

Velocidade Média de Vento		
Meses de 2012	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,0813
Fev	2,81	2,7393
Mar	2,72	2,8238
Abr	2,73	2,4284
Mai	2,55	2,2418
Jun	2,58	2,2968
Jul	2,78	2,4994
Ago	3,27	2,8776
Set	3,15	3,1209
Out	3,12	2,7111
Nov	2,35	2,333
Dez	2,59	2,5809

Fonte: Do Autor

Tabela 8 – Dados de entrada utilizados para o modelo 8

Hora (h)	Pressão (hPa)
0	1012,5
12	1014,3
18	1012,2

Fonte: Do Autor

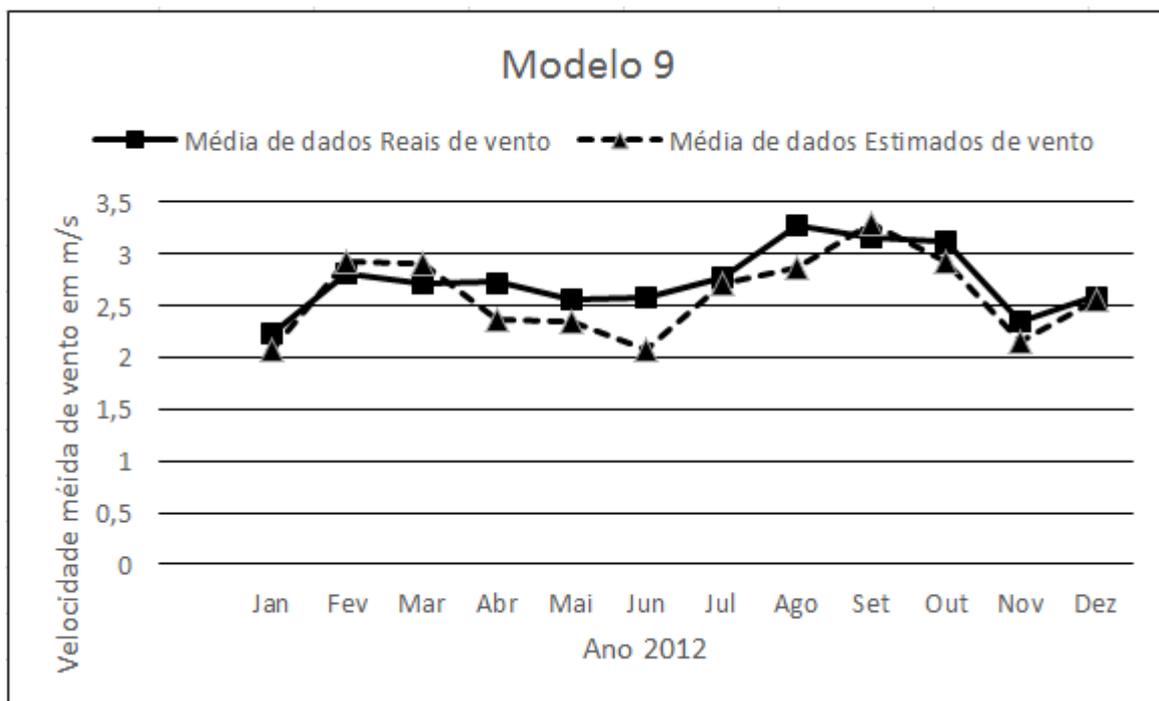


Figura 11 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 9 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 9 – Média dos dados reais e estimados de vento para o modelo 9 da Rede Neural Artificial

Meses de 2012	Velocidade Média de Vento	
	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,0824
Fev	2,81	2,9208
Mar	2,72	2,9158
Abr	2,73	2,3763
Mai	2,55	2,3417
Jun	2,58	2,0786
Jul	2,78	2,7169
Ago	3,27	2,8645
Set	3,15	3,2991
Out	3,12	2,9342
Nov	2,35	2,1514
Dez	2,59	2,5651

Fonte: Do Autor

Tabela 10 – Dados de entrada utilizados para o modelo 9

Temperatura (°C)	Umidade (%)
27,2	76
29,4	70
28,8	69

Fonte: Do Autor

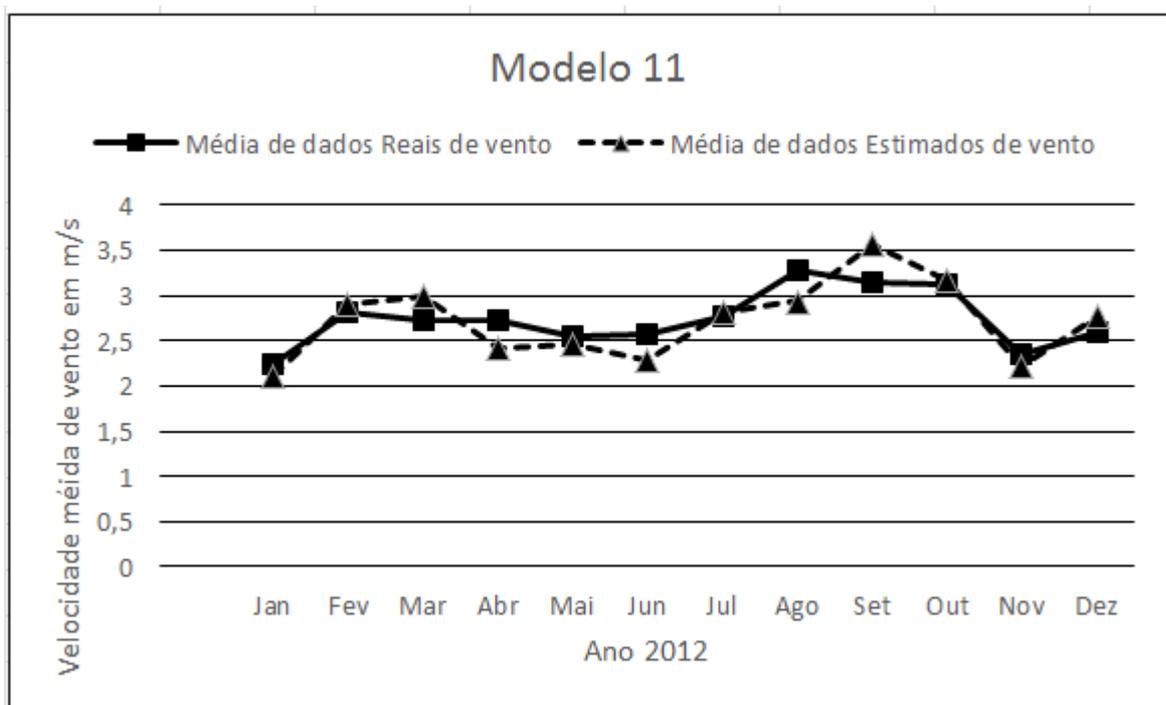


Figura 12 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 11 da rede neural para os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 11 - Média dos dados reais e estimados de vento para o modelo 11 da Rede Neural Artificial

Velocidade Média de Vento		
Meses de 2012	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,1189
Fev	2,81	2,8963
Mar	2,72	2,9855
Abr	2,73	2,4245
Mai	2,55	2,4617
Jun	2,58	2,2839
Jul	2,78	2,8193
Ago	3,27	2,9381
Set	3,15	3,563
Out	3,12	3,1774
Nov	2,35	2,2164
Dez	2,59	2,7788

Fonte: Do Autor

Tabela 12 – Dados de entrada utilizados para o modelo 11

Umidade (%)	Pressão (hPa)
76	1012,5
70	1014,3
69	1012,2

Fonte: Do Autor

Pode-se concluir, de acordo com os resultados encontrados acima, comparando todos os gráficos, que os dois melhores modelos para a velocidade média de vento foram o modelo 5 e o modelo 11, cujos dados reais e estimados de velocidade média de vento foram muito próximos. Ambos os modelos não dependem do horário de medição, apenas da temperatura, umidade e pressão, no caso do modelo 5, e da umidade e pressão, no caso do modelo 11. Sendo assim, ambos os modelos podem ser utilizados para a previsão de potencial eólico médio mensal, bem como da potência elétrica média mensal produzido por um determinado aerogerador ou conjunto de geradores eólicos.

5 CONCLUSÕES E PERSPECTIVAS PARA TRABALHOS FUTUROS

Neste trabalho de conclusão de curso foi feito o desenvolvimento, através do MATLAB/SIMULINK[®], de modelos preditivos de velocidade de vento baseados em redes neurais artificiais.

Foram treinadas diversas configurações de redes para obter um melhor resultado, pelo método da tentativa e erro, no qual a que tivesse o menor erro quadrático (MSE) teria a melhor configuração. Todas as redes foram do tipo MLP (Multilayer Perceptron) no qual foi utilizado o algoritmo *Backpropagation*. Para todas as redes e modelos, a função de ativação da camada escondida foi Tangente Hiperbólica (*Tansig* no Matlab[®]) e para a camada de saída foi a função de ativação linear *Purelin*.

O objetivo foi alcançado pelo autor deste trabalho, no qual encontrou-se para a rede neural do modelo 5, o menor erro quadrático médio esperado, com valor de 0,0482, em que as médias das velocidades estimadas pela rede foram bem próximas às médias das velocidades reais coletadas.

O uso do MATLAB/SIMULINK[®], em particular, da *toolbox* de redes neurais, permitiu o autor deste trabalho ampliar seu conhecimento nesta área do conhecimento, e sobretudo, na importância que as redes neurais representam nas soluções de problemas de Engenharia.

A partir do que foi desenvolvido neste trabalho de conclusão de curso pode-se sugerir como propostas de trabalhos futuros o seguinte:

- Desenvolvimento de outros tipos de redes neurais para previsão de vento, com outras denominações, outras funções de ativação, diferentes camadas de entrada, escondida e de saída;
- Comparação dos modelos apresentados neste trabalho com modelos relativos a outros programas de simulação que envolvam outros métodos para a previsão de vento;
- Estudos de predição de velocidades de ventos por redes neurais para curtos períodos de tempo. Este tipo de estudo seria interessante para prever substituir dados com erros de medição.

6 REFERÊNCIAS BIBLIOGRÁFICAS

CARVALHO, ANDRÉ. **Redes Neurais Artificiais**. Artigo. Instituto de Ciências Médicas e de Computação, Universidade de São Paulo. 2010

Departamento de Informática, **Uma Introdução às Redes Neurais Artificiais**. Disponível em: <<http://www.din.uem.br/ia/neurais/#artificial>>. Acesso em 12 de Maio de 2016

BITENCOURT, Diego. **Estratégias de Previsão de Geração de Energia Eólica Utilizando Redes Neurais Artificiais**. Dissertação de Mestrado. Universidade Federal de Sergipe. São Cristóvão - Sergipe. Fevereiro de 2015.

HENRIQUE, PEDRO. **Previsão da Geração de Potência Eólica Utilizando Redes Neurais Artificiais**. Trabalho de Conclusão do Curso. Universidade de Pernambuco. Junho de 2014.

BRAGA, ANTÔNIO; CARVALHO, ANDRÉ; LUDERMIR, TERESA. **Redes Neurais Artificiais, Teorias e Aplicações**. Livros Técnicos e Científicos Editora S.A. Rio de Janeiro. 2000

HAYKIN, S. **Redes Neurais: Princípios e Prática**, 2. ed. Editora Bookman. 2000

DEMUTH, H., BEALE, M., HAGAN, M., **Neural Network Toolbox™ 6 – User's Guide**. The MathWorks, Inc. 2010

TABLADA, LORENA. **Utilizando Redes Neurais para Previsão de Velocidade do Vento**. Trabalho de Conclusão de Curso. Universidade de Pernambuco. Maio de 2010

VERAS, CARLOS. **Estudo Comparativo de Técnicas de Redes Neurais Artificiais na Previsão da Velocidade do Vento em Curto Prazo**. Trabalho de Conclusão de Curso. Universidade de Pernambuco. Novembro de 2013.

FREIRE, MARCELO. **Utilização de Redes Neurais na Previsão da Velocidade do Vento**. Trabalho de Conclusão de Curso. Universidade de Pernambuco. Julho de 2014.

CARNEIRO, TATIANE; SANTOS, HERON; BRAGA, ARTHUR; CARVALHO, PAULO. **Redes Neurais Artificiais para Previsão da Velocidade do Vento: Estudo de Caso para Maracanaú, CE**, Anais do XX Congresso Brasileiro de Automática. Belo Horizonte-MG. 20 a 24 de Setembro de 2014.

ROZZA, GIOVANN; SILVA, RUY; MILLER, SONIA. **Estudo Comparativo do Uso de Redes Neurais Artificiais e Regressão Linear Múltipla para a Previsão da Concentração Cáustica em uma Etapa do Processo de Fabricação de Alumina**, Artigo. Universidade Federal do Paraná, Pontifícia Universidade Católica, Curitiba, 2015

SHAMISI, MAITHA; ASSI, ALI; HEJASI, HASSAN.. **Using MATLAB to develop Artificial Neural Network Models for Predicting Global Solar Radiation in Al Ain City – UAE**, United Arab Emirates University, 2011

RODRIGUES, GUILHERME. **Utilização de Redes Neurais para Previsão de Ventos no Horizonte de 24 horas**, Dissertação de Mestrado, Rio de Janeiro, 2007

ALVES, ELIS; VECCHIA, FRANCISCO. **Análise de diferentes métodos de interpolação para a precipitação pluvial do Estado de Goiás**, Artigo. Maringá, 2011

GREENE, WILLIAM H. **Econometric Analysis**, (6th ed.) New Jersey: Pearson Prentice Hall, 2008.

GUJARATI, DAMODAR. **Econometria Básica**, 3ª edição, Elsevier, 2006.

SOUZA, REINALDO C. **Previsão de Geração Eólica**, Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica, 2011.

ZUBEN, FERNANDO; ATTUX, ROMIS. **Rede Neural de Kohonen e Aprendizado Não-Supervisionado**, DCA/FEEC/Unicamp & DECOM/FEEC/Unicamp, 2004.

APÊNDICE

Abaixo são listados os principais códigos desenvolvidos em MATLAB[®] para o Modelo 1 de predição de velocidade de vento. Para os demais modelos são apenas alterados os dados de entrada. Aqui também são apresentados os demais gráficos e tabelas dos outros 6 modelos preditivos que não tiveram os melhores desempenhos.

Arquivo *modelo1_neural.m*

(Arquivo de construção e treinamento da rede neural para o modelo 1):

```
clear all; clc;

% carregamento dos dados de entrada:
dadosentrada=load('dadostreino200920102011.txt')

% randomização dos dados de entrada:
dados_rand=dadosentrada(randperm(length(dadosentrada)),1:5);

% Valor da quantidade total de dados:
qtdeTreino = ceil((2/3)*length(dados_rand));

%2/3 (dois terços) de dados para treino:
dadosTreino = dados_rand(1:qtdeTreino,1:4);

%1/3 (um terço) % dados para validação:
dadosValidacao = dados_rand(qtdeTreino+1:length(dados_rand),1:4);

% dados medidos (reais) de saída para treino:
saidaTreino = dados_rand(1:qtdeTreino,5);

% dados reais (target) de saída para validação
saidaValidacao = dados_rand(qtdeTreino+1:length(dados_rand),5);

%Normalizando:
%A função mapminmax normaliza os dados entre -1 e 1

% normalização dos dados % de treino:
[dadosTreinoNormal, psTreino] = mapminmax(dadosTreino');

% normalização dos dados de validação:
[dadosValidacaoNormal, psValidacao] = mapminmax(dadosValidacao');
```

```

% normalização do      % vetor de dados reais (target) para treino:
[saidaTreinoNormal, psSaidaTreino] = mapminmax(saidaTreino');

% normalização do vetor dados reais (target) para validação:
[saidaValidacaoNormal, psSaidaValidacao] = mapminmax(saidaValidacao');

%Definição da RNA:

%Valores para Validação a serem usadas no treinamento
VV.P = dadosValidacaoNormal; % contém os dados de entrada para validação
VV.T = saidaValidacaoNormal; % contém dos dados de saída (target) para validação

rede = newff(minmax(dadosTreinoNormal),[25 35 25 1],{'tansig' 'tansig' 'tansig'
'purelin'});

% definição de parâmetros de simulação da RNA:
rede.trainParam.epochs = 50; % numero de épocas
rede.trainParam.show = 5; % intervalo de amostragem via gráfico de performance
rede.trainParam.lr = 0.5; % taxa de aprendizagem
net.trainparam.min_grad = 1e-6; % desempenho minimo do gradiente
rede.trainParam.max_fail = 6; % numero máximo de interações para overfitting

%Treinamento da Rede com Dados de Treino e Validação:
[redetr,tr] = train(rede,dadosTreinoNormal,saidaTreinoNormal,[],[],VV);
plotperform(tr);

%Simulação da Rede com Dados de Validação:
resultadoSimulacaoValidacaoNormal = sim(redetr, dadosValidacaoNormal);
resultadoSimulacaoValidacao =
mapminmax('reverse',resultadoSimulacaoValidacaoNormal,psSaidaValidacao);
saidaValidacao = mapminmax('reverse',saidaValidacaoNormal,psSaidaValidacao);
plot(saidaValidacao, 'b');
hold on
plot(resultadoSimulacaoValidacao, 'r');

% salvando os dados da rede treinada
save('RNA_modelo1.mat','redetr','tr');

```

Arquivo *avaliacao_RNA_modelo1.m*

(Arquivo de construção e treinamento da rede neural para o modelo 1):

```

clear all; clc;

load('RNA_modelo1.mat','redetr','tr');

%---- Avaliação dos dados de vento de janeiro de 2012 (modelo1)

load('jan2012modelo1.txt');

entrada = jan2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=jan2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);

```

```

% calculo da média dos valores estimados e reais
n = length(tgt);
mediaJanDadosReaisMod1 = sum(tgt)/n;
mediaJanDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Fevereiro de 2012 (modelo1)

load('fev2012modelo1.txt');

entrada = fev2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=fev2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaFevDadosReaisMod1 = sum(tgt)/n;
mediaFevDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Março de 2012 (modelo1)

load('mar2012modelo1.txt');

entrada = mar2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=mar2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaMarDadosReaisMod1 = sum(tgt)/n;
mediaMarDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Abril de 2012 (modelo1)

load('abr2012modelo1.txt');

entrada = abr2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=abr2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaAbrDadosReaisMod1 = sum(tgt)/n;
mediaAbrDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Maio de 2012 (modelo1)

load('mai2012modelo1.txt');

entrada = mai2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=mai2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);

```

```

% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaMaiDadosReaisMod1 = sum(tgt)/n;
mediaMaiDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Junho de 2012 (modelo1)

load('jun2012modelo1.txt');

entrada = jun2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=jun2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaJunDadosReaisMod1 = sum(tgt)/n;
mediaJunDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Julho de 2012 (modelo1)

load('jul2012modelo1.txt');

entrada = jul2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=jul2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaJulDadosReaisMod1 = sum(tgt)/n;
mediaJulDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Agosto de 2012 (modelo1)

load('ago2012modelo1.txt');

entrada = ago2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=ago2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaAgoDadosReaisMod1 = sum(tgt)/n;
mediaAgoDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Setembro de 2012 (modelo1)

load('set2012modelo1.txt');

entrada = set2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=set2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro

```

```

n = length(tgt);
mediaSetDadosReaisMod1 = sum(tgt)/n;
mediaSetDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Outubro de 2012 (modelo1)

load('out2012modelo1.txt');

entrada = out2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=out2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaOutDadosReaisMod1 = sum(tgt)/n;
mediaOutDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Novembro de 2012 (modelo1)

load('nov2012modelo1.txt');

entrada = nov2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=nov2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaNovDadosReaisMod1 = sum(tgt)/n;
mediaNovDadosEstMod1 = sum(ver)/n;

% ---- Avaliação dos dados de vento para Dezembro de 2012 (modelo1)

load('dez2012modelo1.txt');

entrada = dez2012modelo1(:,1:4);
[pn, ps] = mapminmax(entrada');
saida=dez2012modelo1(:,5);
[tn, ts] = mapminmax(saida');
vern=sim(redetr,pn);
ver=mapminmax('reverse',vern,ts);
tgt = mapminmax('reverse',tn,ts);
% calculo da média dos valores estimados e reais de fevereiro
n = length(tgt);
mediaDezDadosReaisMod1 = sum(tgt)/n;
mediaDezDadosEstMod1 = sum(ver)/n;

mediaRealMod12012 = [mediaJanDadosReaisMod1, mediaFevDadosReaisMod1,...
    mediaMarDadosReaisMod1, mediaAbrDadosReaisMod1, mediaMaiDadosReaisMod1,...
    mediaJunDadosReaisMod1, mediaJulDadosReaisMod1, mediaAgoDadosReaisMod1,...
    mediaSetDadosReaisMod1, mediaOutDadosReaisMod1, mediaNovDadosReaisMod1,...
    mediaDezDadosReaisMod1];

mediaEstMod12012 = [mediaJanDadosEstMod1, mediaFevDadosEstMod1,...
    mediaMarDadosEstMod1, mediaAbrDadosEstMod1, mediaMaiDadosEstMod1,...
    mediaJunDadosEstMod1, mediaJulDadosEstMod1, mediaAgoDadosEstMod1,...
    mediaSetDadosEstMod1, mediaOutDadosEstMod1, mediaNovDadosEstMod1,...
    mediaDezDadosEstMod1];

hold off

```

```

n = 1:1:length(mediaRealMod12012);
plot(n,mediaRealMod12012,'-k^','LineWidth',2)
hold on
plot(n,mediaEstMod12012,'--ks','LineWidth',2)

xlabel('Numero de amostras');
ylabel('Velocidade Média de Vento (m/s)');
title('MODELO 1 DE PREVISÃO');
legend('Dados Reais', 'Dados Previsto pela RNA');

% Coeficientes de avaliação da RNA
x = mediaRealMod12012; % variável real
y = mediaEstMod12012; % variavel estimada
n = length(x);

% determinação do erro médio quadrático
MSE = (sum((x-y).^2)/n)

% Determinação do Erro Percentual Absoluto Médio (MAPE - Mean Absolut Percentual
Error)
MAPE = ((sum(abs((x-y)./x)))/n)*100

% determinação da função de regressão linear, do coeficiente de correlação
% e de R2
m_est=(mean(x)*mean(y)-mean(x.*y))/(mean(x).^2-mean(x.^2));
b_est=mean(y)-m_est*mean(x);
a=(mean(x.*y)- mean(x)* mean(y)).^2;
b=(mean(x.^2)-(mean(x)).^2)*(mean(y.^2)-(mean(y)).^2));

% Determinação do coeficiente de regressão linear R
R=sqrt(a/b)
% Determinação do coeficiente de determinação R2
R2=(a/b)

```

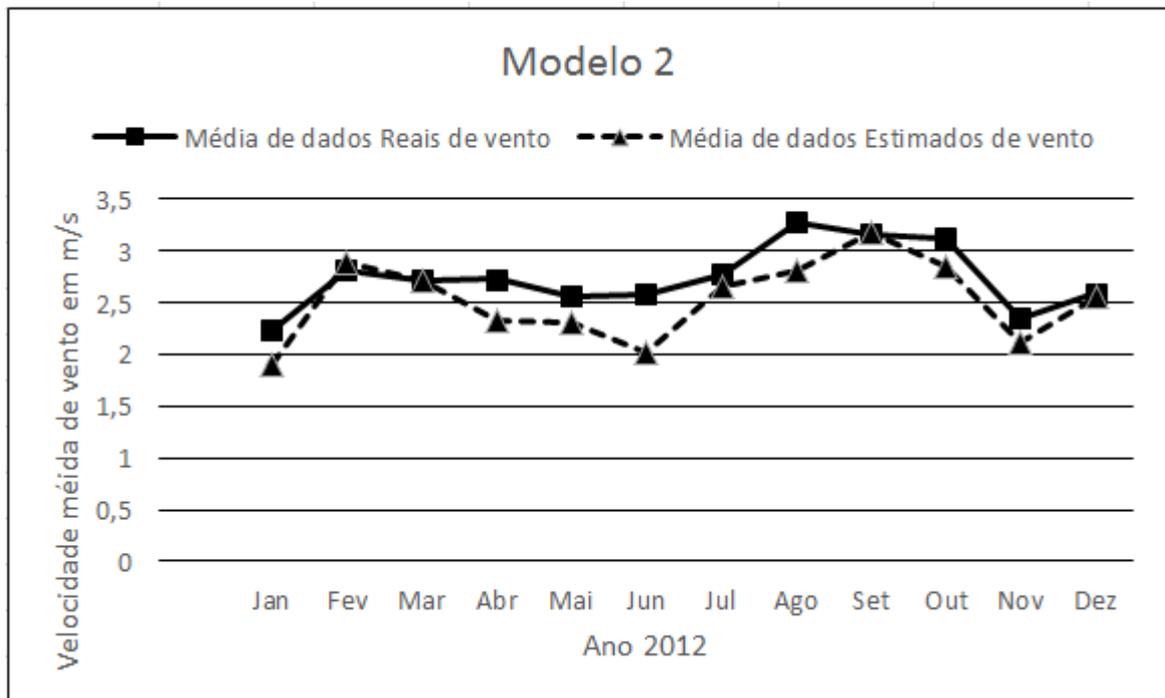


Figura 13 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 2 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 13 - Média dos dados reais e estimados de vento para o modelo 2 da Rede Neural Artificial

Meses de 2012	Velocidade Média de Vento	
	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	1,901
Fev	2,81	2,8815
Mar	2,72	2,7246
Abr	2,73	2,3267
Mai	2,55	2,3084
Jun	2,58	2,011
Jul	2,78	2,6552
Ago	3,27	2,8157
Set	3,15	3,1715
Out	3,12	2,8491
Nov	2,35	2,113
Dez	2,59	2,5633

Fonte: Do Autor

Tabela 14 – Dados de entrada utilizados para o modelo 2

Hora (h)	Temperatura (°C)	Umidade (%)
0	27,2	76
12	29,4	70
18	28,8	69

Fonte: Do Autor

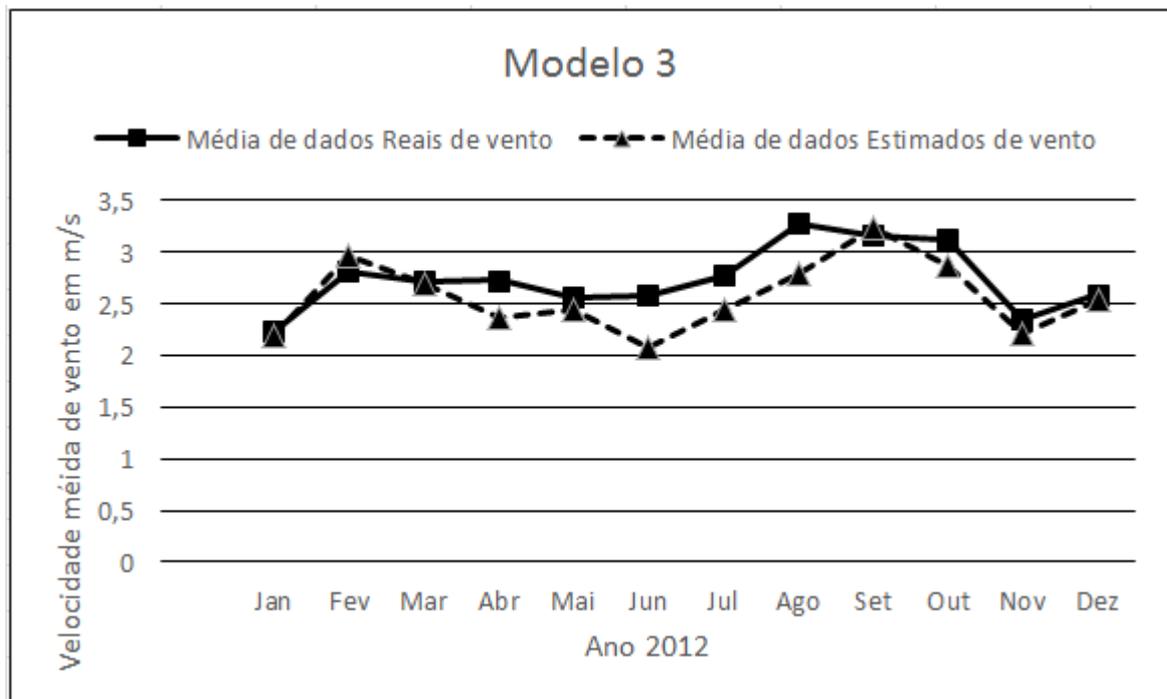


Figura 14 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 3 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 15 - Média dos dados reais e estimados de vento para o modelo 3 da Rede Neural Artificial

Velocidade Média de Vento		
Meses de 2012	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,2022
Fev	2,81	2,9726
Mar	2,72	2,6917
Abr	2,73	2,3597
Mai	2,55	2,4396
Jun	2,58	2,0764
Jul	2,78	2,4369
Ago	3,27	2,791
Set	3,15	3,2328
Out	3,12	2,8646
Nov	2,35	2,2116
Dez	2,59	2,5358

Fonte: Do Autor

Tabela 16 – Dados de entrada utilizados para o modelo 3

Hora (h)	Temperatura (°C)	Pressão (hPa)
0	27,2	1012,5
12	29,4	1014,3
18	28,8	1012,2

Fonte: Do Autor

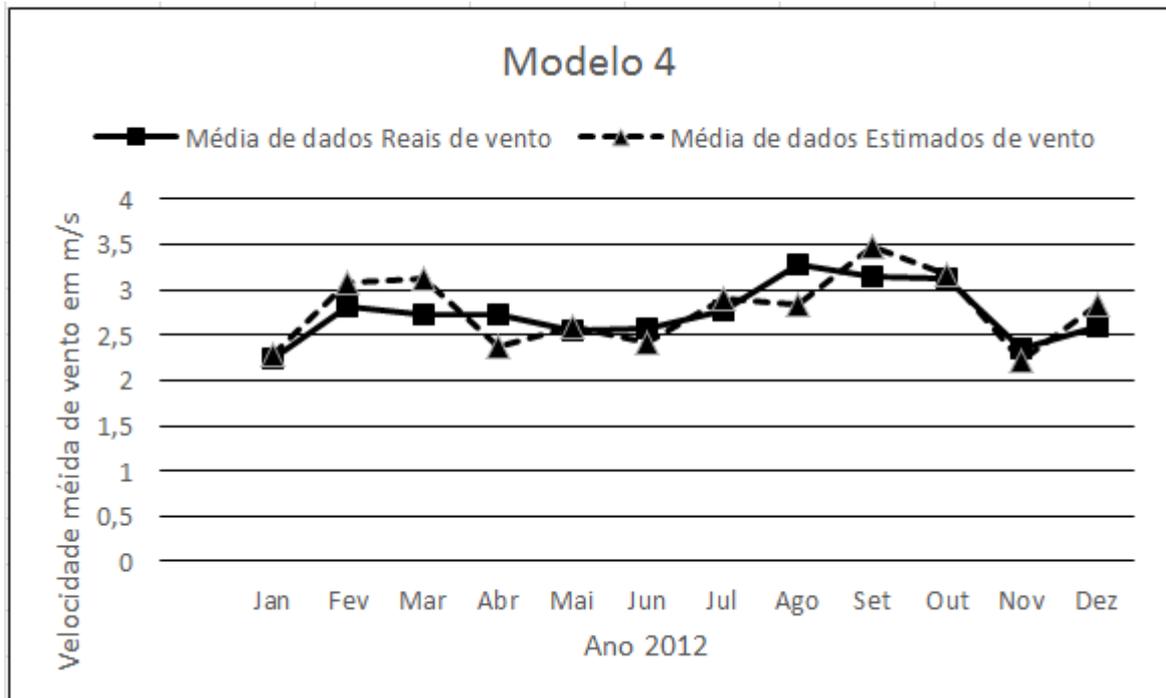


Figura 15 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 4 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 17 - Média dos dados reais e estimados de vento para o modelo 4 da Rede Neural Artificial

Meses de 2012	Velocidade Media de Vento	
	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,2939
Fev	2,81	3,0815
Mar	2,72	3,1228
Abr	2,73	2,3713
Mai	2,55	2,6022
Jun	2,58	2,4235
Jul	2,78	2,9158
Ago	3,27	2,8494
Set	3,15	3,4765
Out	3,12	3,1693
Nov	2,35	2,2223
Dez	2,59	2,8373

Fonte: Do Autor

Tabela 18 – Dados de entrada utilizados para o modelo 4

Hora (h)	Umidade (%)	Pressão (hPa)
0	76	1012,5
12	70	1014,3
18	69	1012,2

Fonte: Do Autor

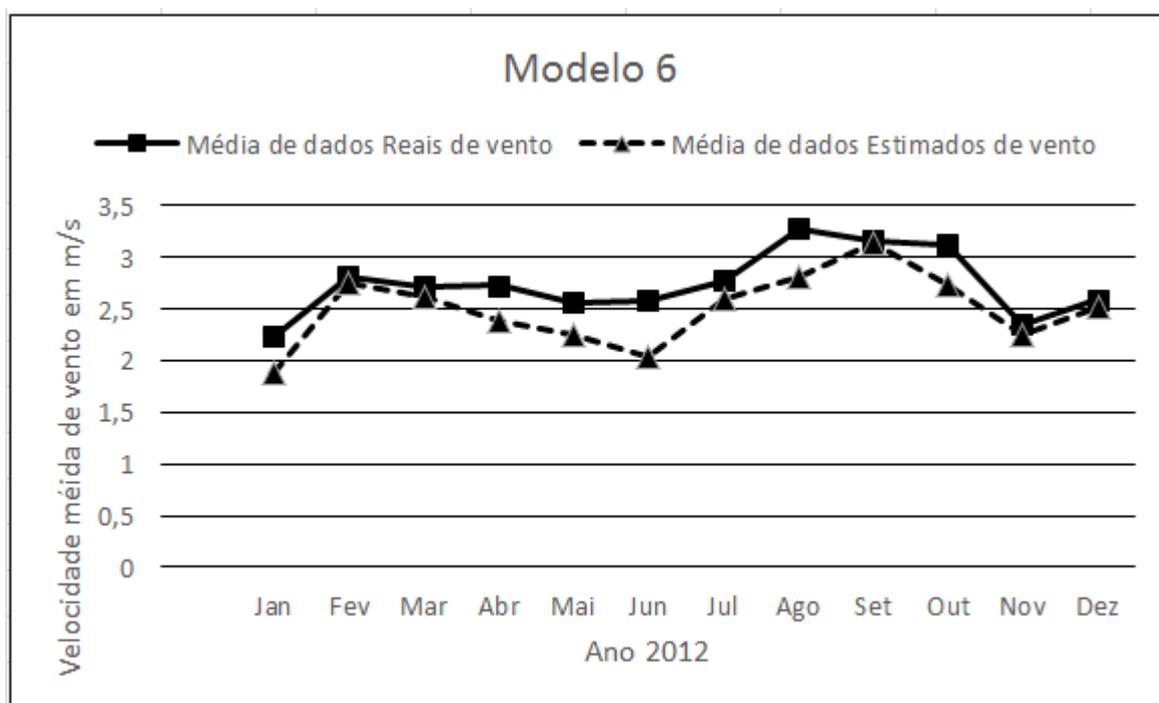


Figura 16 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 6 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 19 - Média dos dados reais e estimados de vento para o modelo 6 da Rede Neural Artificial

Velocidade Média de Vento		
Meses de 2012	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	1,886
Fev	2,81	2,7527
Mar	2,72	2,6263
Abr	2,73	2,3961
Mai	2,55	2,2581
Jun	2,58	2,032
Jul	2,78	2,6036
Ago	3,27	2,8161
Set	3,15	3,1448
Out	3,12	2,7293
Nov	2,35	2,2448
Dez	2,59	2,53

Fonte: Do Autor

Tabela 20 – Dados de entrada utilizados para o modelo 6

Hora (h)	Temperatura (°C)
0	27,2
12	29,4
18	28,8

Fonte: Do Autor

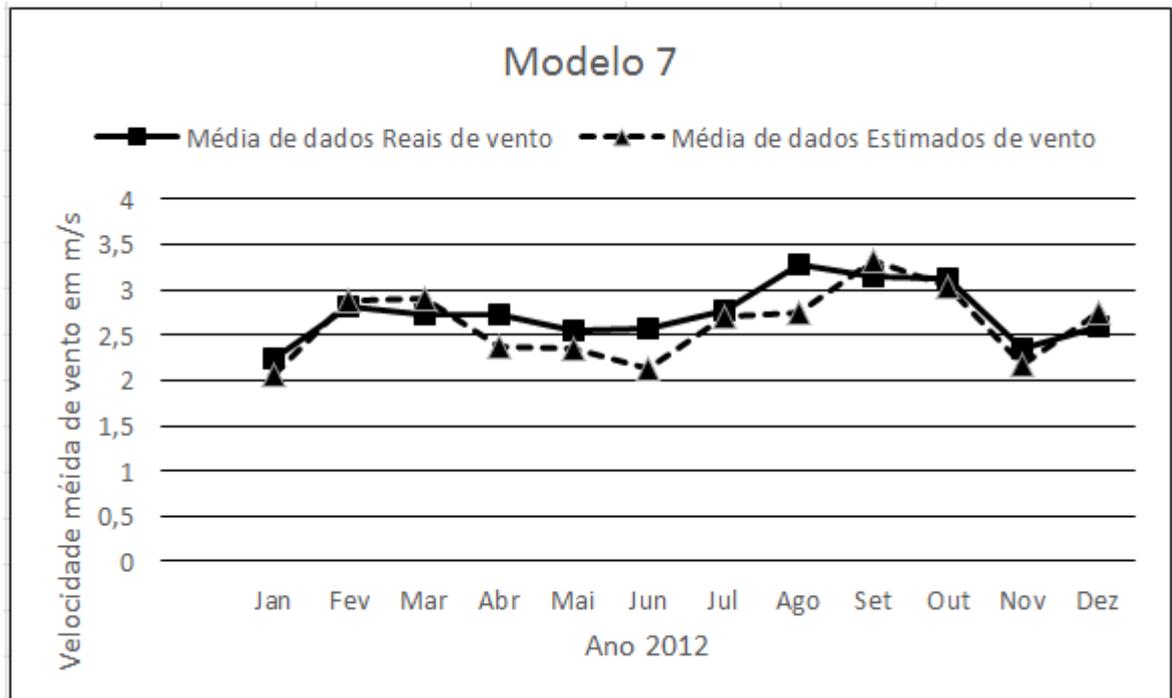


Figura 17 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 7 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 21 - Média dos dados reais e estimados de vento para o modelo 7 da Rede Neural Artificial

Meses de 2012	Velocidade Média de Vento	
	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,0682
Fev	2,81	2,883
Mar	2,72	2,9045
Abr	2,73	2,3693
Mai	2,55	2,3561
Jun	2,58	2,1244
Jul	2,78	2,7065
Ago	3,27	2,7529
Set	3,15	3,3357
Out	3,12	3,0467
Nov	2,35	2,1865
Dez	2,59	2,7428

Fonte: Do Autor

Tabela 22 – Dados de entrada utilizados para o modelo 7

Hora (h)	Umidade (%)
0	76
12	70
18	69

Fonte: Do Autor

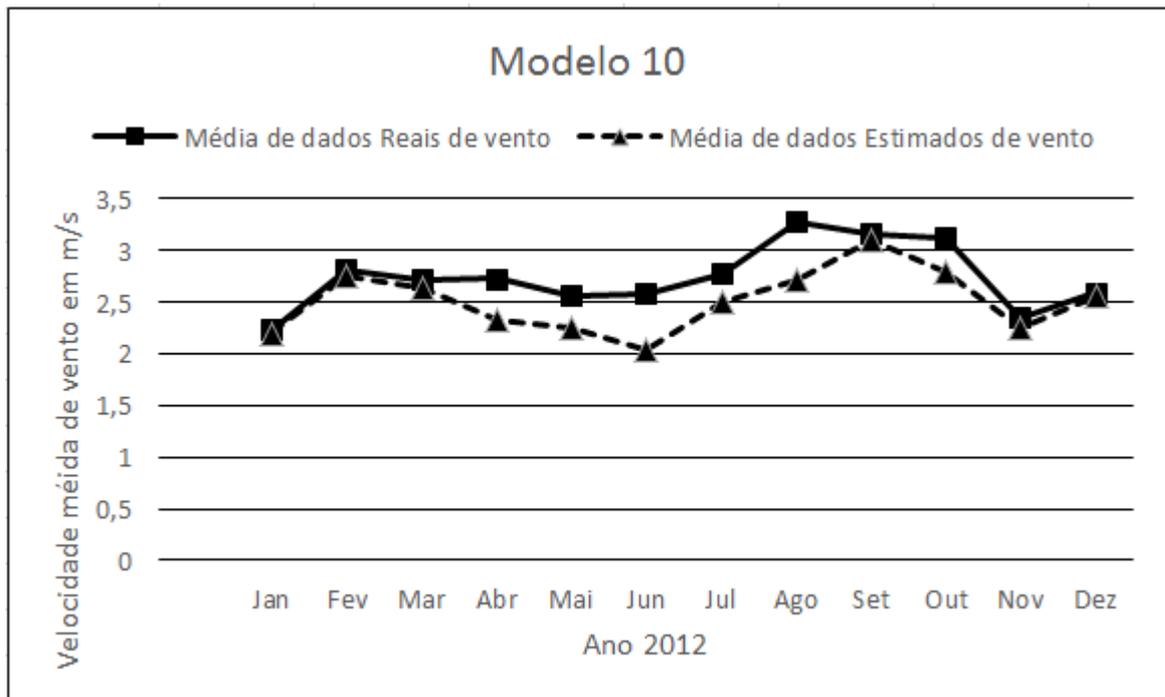


Figura 18 – Comparação da velocidade média real do vento com a velocidade média estimada do vento para o modelo 10 da rede neural entre os meses de Janeiro a Dezembro de 2012

Fonte: Do Autor

Tabela 23 - Média dos dados reais e estimados de vento para o modelo 10 da Rede Neural Artificial

Velocidade Média de Vento		
Meses de 2012	Média de dados Reais de vento	Média de dados Estimados de vento
Jan	2,24	2,2035
Fev	2,81	2,7545
Mar	2,72	2,6391
Abr	2,73	2,3386
Mai	2,55	2,2465
Jun	2,58	2,0353
Jul	2,78	2,5004
Ago	3,27	2,7096
Set	3,15	3,1054
Out	3,12	2,8008
Nov	2,35	2,2566
Dez	2,59	2,5585

Fonte: Do Autor

Tabela 24 – Dados de entrada utilizados para o modelo 10

Temperatura (°C)	Pressão (hPa)
27,2	1012,5
29,4	1014,3
28,8	1012,2

Fonte: Do Autor