

Universidade Federal de Juiz de Fora  
Faculdade de Engenharia Elétrica  
Departamento de Energia Elétrica

**Thales Tozatto Filgueiras**

**Predição da Velocidade dos Ventos Utilizando Redes Neurais Artificiais**

Juiz de Fora

2019

**Thales Tozatto Filgueiras**

**Predição da Velocidade dos Ventos Utilizando Redes Neurais Artificiais**

Monografia apresentada à Coordenação do Curso de Graduação em Engenharia Elétrica, modalidade Energia, da Faculdade de Engenharia da Universidade Federal de Juiz de Fora, como parte dos requisitos para obtenção do grau de Engenheiro Eletricista.

Orientador: Leonardo Rocha Olivi

Juiz de Fora

2019

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF  
com os dados fornecidos pelo(a) autor(a)

Tozatto, Thales.

Predição da Velocidade dos Ventos Utilizando Redes Neurais Artificiais /  
Thales Tozatto Filgueiras. – 2019.

45 f. : il.

Orientador: Leonardo Rocha Olivi

Monografia (Graduação) – Universidade Federal de Juiz de Fora, Faculdade de Engenharia Elétrica. Departamento de Energia Elétrica, 2019.

1. Energia Eólica. 2. Predição. 3. Inteligência Artificial. I. R. Olivi, Leonardo, orient. II. Título.

Thales Tozatto Filgueiras

Predição da Velocidade dos Ventos Utilizando Redes Neurais Artificiais

Monografia apresentada à Coordenação do Curso de Graduação em Engenharia Elétrica, modalidade Energia, da Faculdade de Engenharia da Universidade Federal de Juiz de Fora, como parte dos requisitos para obtenção do grau de Engenheiro Eletricista.

Aprovada em:

BANCA EXAMINADORA



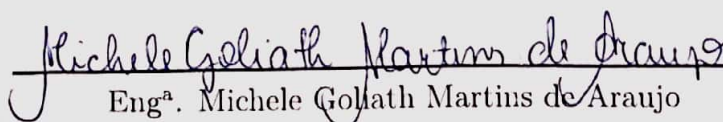
Prof. Dr. Leonardo Rocha Olivi - Orientador  
Universidade Federal de Juiz de Fora



Professor Dr. Exuperry Barros Costa  
Universidade Federal de Juiz de Fora



Professor Dr. Leonardo Willer de Oliveira  
Universidade Federal de Juiz de Fora



Eng<sup>a</sup>. Michele Goliath Martins de Araujo

*Dedico este trabalho a minha avó Maria das Graças Sporch Tozatto, que por muito pouco não pôde ver a esperada conclusão desta etapa.*

## AGRADECIMENTOS

Agradeço primeiramente aos meus pais, Carlos José e Giovana Giarola, por todo o suporte e pela constante motivação nos momentos em que fraquejei. Essa graduação não seria possível sem eles.

Agradeço a toda a minha grande família, a qual amo, e por isso sempre me motivou a pensar nas escolhas com resultados de longo prazo.

Agradeço também aos amigos pelo apoio e pelos momentos de descontração que tornaram a jornada mais leve.

Agradeço a minha companheira, Nanda, pelo companheirismo incondicional, pela compreensão e pelo amor compartilhados.

Agradeço ao meu orientador, o professor Leonardo Rocha Olivi, pelo direcionamento, pelos ensinamentos construtivos e entusiasmo com o projeto.

Agradeço a todos os professores que se dedicam ao ensino nessa instituição, pelo conhecimento dividido e pelas muitas vezes que, sem saber, me lembraram do motivo da minha escolha.

“O maior atributo da inteligência artificial não é a malícia, mas a competência. Uma inteligência artificial super inteligente será muito boa em alcançar seus objetivos, e se estes não estiverem alinhados com os nossos, teremos problemas. Você provavelmente não é o tipo de pessoa que odeia formigas e que pisa nelas por maldade, mas se você for responsável por uma hidroelétrica e tiver um formigueiro na região que será inundada, problema das formigas. Não vamos colocar a humanidade na posição dessas formigas. É importante pensar não só na criação da inteligência artificial, mas como fazer um uso benéfico dela.”

(Stephen Hawking, 2015)

## RESUMO

O presente trabalho estuda a previsão de velocidades do vento de curto prazo, um tema que vem sendo amplamente explorado devido ao grande aumento da participação da energia eólica nas matrizes energéticas em todo o mundo. O avanço das técnicas de previsão contribui para que fontes energéticas dependentes de comportamentos climáticos possam ser inseridas no Sistema Elétrico com maior confiabilidade. O estudo explora os dados meteorológicos disponíveis na cidade de Petrolina, PE. São descritas, analisadas e aplicadas técnicas de inteligência artificial muito exploradas na atualidade, as redes *Multilayer Perceptron* e a camada *Long-Short Term Memory*. São implementados quatro modelos diferentes, utilizando as duas técnicas e executando previsões com horizontes de 1 a 6 horas a frente, comparando também as estratégias *single-step* e *multi-step*. Os resultados mostram que as técnicas conseguem prever mudanças de tendências em muitos casos e têm potencial de serem melhor exploradas, estudando combinações com outras técnicas complementares.

Palavras-chave: Energia eólica. Redes Neurais Artificiais. Previsão de velocidade do vento.



## ABSTRACT

The present work studies the short-term forecast of wind speed, a subject that has been widely explored due to the large increase of wind energy participation in energy matrices worldwide. The advance of forecasting techniques contributes so that energy sources dependent on climate behavior can be inserted into the Electric System with greater reliability. The study explores the meteorological data available in the city of Petrolina, PE, Brazil. Artificial Intelligence techniques widely explored today are described, analyzed and applied, the Multilayer Perceptron Networks and the Long-Short Term Memory layer. Four different models are implemented using both techniques and running forecasts with horizons from 1 to 6 hours ahead, also comparing the single-step and multi-step strategies. The results show that the techniques can predict trend changes in many cases and have the potential to be further explored by studying combinations with other complementary techniques.

Key-words: Wind Energy. Artificial Neural Networks. Wind Speed Forecast.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Torre com anemômetros a 10m, 25m e 50m de altura. Petrolina. Fonte: INPE. . . . .	14
Figura 2 – Médias móveis com janelas 5 e 20. . . . .	15
Figura 3 – Média móvel com desvio padrão. . . . .	16
Figura 4 – Aprendizado supervisionado. . . . .	18
Figura 5 – Perceptron. . . . .	19
Figura 6 – Variável $ws_{10m}$ no período 2007-2011. . . . .	22
Figura 7 – Matriz de correlação. . . . .	23
Figura 8 – Histograma de velocidades do vento. . . . .	24
Figura 9 – MLP com duas camadas escondida. . . . .	26
Figura 10 – ReLU. . . . .	27
Figura 11 – Modelos A-1 e A-2. . . . .	28
Figura 12 – Arranjo de células LSTM. . . . .	28
Figura 13 – Célula LSTM. . . . .	29
Figura 14 – Função sigmoide. . . . .	30
Figura 15 – Modelos B-1 e B-2. . . . .	31
Figura 16 – <i>Previsão Multi-Step</i> . . . . .	32
Figura 17 – <i>Previsão Single-Step</i> . . . . .	32
Figura 18 – Conjuntos de Treino, Validação e Teste. . . . .	33
Figura 19 – Treinamento modelo A-2. . . . .	35
Figura 20 – Treinamento modelo B-2. . . . .	35
Figura 21 – MAE. . . . .	37
Figura 22 – MSE. . . . .	37
Figura 23 – Erro de energia. . . . .	38
Figura 24 – Erro de potência. . . . .	38
Figura 25 – RMSE. . . . .	38
Figura 26 – Comparação de previsões 1. . . . .	39
Figura 27 – Comparação de previsões 2. . . . .	39
Figura 28 – Comparação de previsões 3. . . . .	40
Figura 29 – Previsão modelo MLP-single. . . . .	40
Figura 30 – Previsão modelo MLP-multi. . . . .	41
Figura 31 – Previsão modelo LSTM-single. . . . .	41
Figura 32 – Previsão modelo LSTM-multi. . . . .	42

## LISTA DE ABREVIATURAS E SIGLAS

SIN	Sistema Interligado Nacional
PDE	Plano Decenal de Expansão de Energia
INPE	Instituto Nacional de Pesquisas Espaciais
SONDA	Sistema de Organização Nacional de Dados Ambientais
IA	Inteligência Artificial
ARMA	<i>Autoregressive Moving Average</i>
ARIMA	<i>Autoregressive Integrated Moving Average</i>
SARIMA	<i>Seasonal Autoregressive Integrated Moving Average</i>
RNA	Redes Neurais Artificiais
SVM	<i>Support Vector Machines</i>
PCA	<i>Principal Component Analysis</i>
FRS	<i>Fuzzy Rough Set</i>
LSTM	<i>Long Short-Term Memory</i>
BP	<i>Back Propagation</i>
ML	<i>Machine Learning</i>
SVR	<i>Support Vector Regression</i>
FIS	<i>Fuzzy Inference System</i>
ANFIS	<i>Adaptative Fuzzy Inference System</i>
AWNN	<i>Adaptative Wavelet Neural Network</i>
BP	<i>Backpropagation</i>
MSE	<i>Mean Squared Error</i>
RMSE	<i>Root Mean Squared Error</i>
MAE	<i>Mean Absolute Error</i>
Adam	<i>Adaptive Moment Estimation</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>11</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TÉCNICA . . . . .</b>	<b>12</b>
2.1	Geração Eólica . . . . .	12
2.1.1	Potência extraída do vento . . . . .	12
2.1.2	Ajuste da velocidade do vento pela altura . . . . .	13
2.1.3	Horizontes de previsão . . . . .	14
2.2	Séries temporais . . . . .	14
2.3	Trabalhos anteriores . . . . .	16
2.4	Redes Neurais Artificiais . . . . .	17
<b>3</b>	<b>DESENVOLVIMENTO . . . . .</b>	<b>21</b>
3.1	Apresentação dos dados . . . . .	21
3.2	Tratamento dos dados . . . . .	22
3.3	Definição dos Modelos Preditivos . . . . .	24
3.3.1	Arquiteturas . . . . .	24
3.3.1.1	<i>Multilayer Perceptron</i> . . . . .	26
3.3.1.2	<i>Long-Short Term Memory</i> . . . . .	27
3.3.2	Divisão dos dados . . . . .	30
3.3.3	Treinamento . . . . .	32
<b>4</b>	<b>RESULTADOS . . . . .</b>	<b>36</b>
<b>5</b>	<b>CONCLUSÃO . . . . .</b>	<b>43</b>
5.1	PROPOSTAS PARA TRABALHOS FUTUROS . . . . .	43
	<b>REFERÊNCIAS . . . . .</b>	<b>44</b>

## 1 INTRODUÇÃO

Segundo o Atlas de Energia Elétrica do Brasil (2005, p. 93):

Denomina-se energia eólica a energia cinética contida nas massas de ar em movimento (vento). Seu aproveitamento ocorre por meio da conversão da energia cinética de translação em energia cinética de rotação, com o emprego de turbinas eólicas, também denominadas aerogeradores, para a geração de eletricidade, ou cataventos (e moinhos), para trabalhos mecânicos como bombeamento d'água [1].

O amplo aproveitamento de energia eólica é uma realidade no Brasil e no mundo. Segundo o Plano Decenal de Energia (PDE) 2027 [2], o Brasil possui uma capacidade instalada no Sistema Interligado Nacional (SIN), oriunda da fonte eólica, de 12.674 MW, o que representa 8% da capacidade total instalada no país. Com os leilões realizados até abril de 2018, existe a contratação 3.998 MW de capacidade instalada, de fonte eólica, para entrar em operação até 2023, resultando num crescimento de 31,5% no período. O PDE 2027 também prevê que a geração de energia eólica tenha participação de 12.7% no SIN em 2027.

A participação brasileira na geração de energia eólica mundial é de 2,9% [3]. Tais dados demonstram a importância da fonte, assim como seu potencial de crescimento. Porém, como todo tipo de geração de energia elétrica, a geração eólica apresenta suas vantagens e desvantagens, sendo a incerteza e inconstância dos ventos partícipes dessas desvantagens [4].

Quando inserida em larga escala no sistema elétrico, a energia eólica gera desafios tanto para a operação confiável do sistema, dado que a mesma sofre bruscas variações, quanto para o planejamento da geração, pela sua imprevisibilidade [4]. Em virtude desse revés, o trabalho aqui apresentado tem como objetivo propor uma ferramenta de previsão de velocidade dos ventos para auxiliar no planejamento da geração.

O avanço de ferramentas de previsão impulsiona a indústria de energias renováveis, pois são fundamentais para que planejamentos mais confiáveis sejam feitos e o setor ganhe cada vez mais credibilidade. É importante destacar também que esse não é apenas um estudo voltado ao setor eólico. Os métodos aqui investigados são aplicáveis, com as devidas adaptações, a diversos tipos de séries temporais, podendo servir de base para o desenvolvimento de ferramentas de previsão também para o setor de geração solar ou do mercado de energia, dentre outros. Ademais, o trabalho explora o grande potencial do aprendizado de máquina, tecnologia que vem ganhando cada vez mais destaque e aplicação em diversas áreas do conhecimento.

## 2 FUNDAMENTAÇÃO TÉCNICA

### 2.1 Geração Eólica

Para desenvolver os métodos de previsão é importante ter conhecimento sobre a natureza do fenômeno estudado. Nesta seção serão abordadas algumas equações que regem o universo da geração eólica, como a relação entre velocidade do vento e potência e a relação entre velocidade do vento e altura. A partir de tais relações é possível obter extrair novas informações que não existem no conjunto de dados disponível.

#### 2.1.1 Potência extraída do vento

A energia cinética de uma parcela de ar de massa  $m$ , se movendo a uma velocidade  $u$ , na direção  $x$  é dada por [5]:

$$U = \frac{1}{2}mu^2 = \frac{1}{2}(\rho Ax)u^2 \quad (2.1)$$

em que  $A$  é área transversal atravessada pela massa de ar  $m$  e  $\rho$  é a densidade do ar em  $\frac{kg}{m^3}$ . A potência de tal massa de ar é a derivada da energia em relação ao tempo:

$$P_w = \frac{dU}{dt} = \frac{1}{2}\rho Au^2 \frac{dx}{dt} = \frac{1}{2}\rho Au^3 \quad (2.2)$$

Para condições normais de temperatura e pressão, 273 K e 101.3 kPa, a expressão se reduz a:

$$P_w = 0.647Au^3 \quad (2.3)$$

Há, porém, um limite teórico para a potência que o rotor de uma turbina ideal consegue extrair do vento, chamado limite de *Betz*. Dado que a potência extraída pelo rotor é encontrada pela equação 2.4 [6],

$$P_{rotor} = \frac{1}{2}\rho Au^3 4a(1-a)^2 \quad (2.4)$$

$$a = \frac{U_1 - U_2}{U_1} \quad (2.5)$$

em que  $U_1$  é a velocidade do vento antes das pás da turbina e  $U_2$  a velocidade do vento após as pás da turbina. Um coeficiente de potência adimensional  $C_p$  representa a performance do rotor e é definido como [6]:

$$C_p = P_{rotor}/P_w = 4a(1-a)^2 \quad (2.6)$$

O máximo valor teórico de  $C_p$  é o limite de *Betz* e pode ser encontrado quando a derivada de  $C_p$  em relação a  $a$  é igual a zero, resultando num valor de  $a$  igual a  $1/3$  [6]:

$$C_{p,max} = 16/27 = 0.5926 \quad (2.7)$$

Finalmente, para encontrar a potência elétrica  $P_{ele}$  que o aerogerador pode fornecer, é preciso ainda incluir a eficiência  $\eta$  da turbina, que irá incluir perdas mecânicas e elétricas, existentes em qualquer turbina real [6]:

$$P_{ele} = P_w C_p \eta = 0.647 A u^3 C_p \eta \quad W \quad (2.8)$$

A equação 2.3 nos mostra que a potência extraída do vento depende linearmente da área considerada, que no problema estudado é representada pela área varrida pelas pás da turbina eólica, e com um fator cúbico da velocidade do vento, sendo o último de grande relevância para um bom aproveitamento energético e, por isso, o objeto de muitos estudos.

### 2.1.2 Ajuste da velocidade do vento pela altura

O conhecimento do comportamento do vento ao longo do ano é fundamental para o projeto de um parque eólico em um dado local. Para a medição da velocidade do vento são usados, geralmente, anemômetros em alturas variando de 10 a 50 *m* de altura, como o que pode ser visto na Figura 1. Conhecendo o perfil vertical da velocidade do vento e o comprimento de rugosidade do terreno, que representa a altura na qual a velocidade do vento possui valor nulo, e que varia em função do tipo de terreno, é possível estimar a velocidade do vento em uma altura desejada a partir de uma velocidade conhecida em uma altura de referência.

Na literatura, os métodos mais utilizados para a correção da velocidade em relação a altitude são a lei logarítmica e a lei da potência [7]. A lei logarítmica é dada pela seguinte equação:

$$v(h) = v_{ref} \frac{\ln(h/z_0)}{\ln(h_{ref}/z_0)} \quad (2.9)$$

em que  $h_{ref}$  representa a altura de referência da medida da velocidade,  $v_{ref}$  a velocidade nesta altura de referência e  $z_0$  é o comprimento de rugosidade.

A lei da potência é dada pela seguinte equação[7]

$$v(h) = v_{ref} \left( \frac{h}{h_{ref}} \right)^\alpha \quad (2.10)$$

em que  $\alpha$  é o expoente de camada limite e varia em função da rugosidade da superfície.



Figura 1 – Torre com anemômetros a 10m, 25m e 50m de altura. Petrolina. Fonte: INPE.

### 2.1.3 Horizontes de previsão

A previsão da velocidade do vento pode ter diferentes horizontes de tempo, dependendo de sua aplicação. A literatura apresenta principalmente 4 categorias [10] [11]:

- Ultra curto prazo (alguns segundos a minutos): Analisa turbulências e variações em períodos de tempo muito curtos a fim estudar aspectos operacionais.
- Curto prazo (até 72 horas): O fluxo do vento é composto por movimentos atmosféricos de grande escala e efeitos microclimáticos. Previsões com potencial para uso no despacho econômico.
- Médio prazo (até 7 dias): Modelos usados para planejamento de recursos.
- Mensal e Sazonal: Previsões para identificar mudanças nos padrões de velocidade do vento em períodos específicos. Também usada no planejamento e alocação de recursos.

## 2.2 Séries temporais

Uma série temporal é um conjunto de dados medidos sequencialmente ao longo do tempo. Nesse tipo de série não importam apenas os valores, mas também a ordem em que



eles se encontram. Isso significa que o valor futuro da variável em questão depende de seus valores passados. Essa dependência é o foco da análise de séries temporais [8].

Para a análise dos dados estudados, será feito o uso de algumas medidas estatísticas. A primeira delas é a média móvel, aqui usada para transformar o intervalo de tempo entre cada dado nas variáveis aleatórias. Para o cálculo da média móvel é definida uma janela, que será o tamanho do intervalo a serem tomadas médias. Essa janela então se desloca por toda a série e cada média extraída da janela se torna um elemento da nova série. Tal processo tem o efeito de suavizar séries que apresentam muitas flutuações em períodos curtos. Uma série de médias móveis  $S_{mm}$  pode ser extraída de uma série  $S$ , com elementos  $x_i$ , seguindo a seguinte relação:

$$S_{mm} = \left\{ \frac{x_1 + x_2 + \dots + x_n}{n}, \frac{x_2 + x_3 + \dots + x_{n+1}}{n}, \dots, \frac{x_{m-n+1} + x_{m-n+2} + \dots + x_m}{n} \right\} \quad (2.11)$$

em que  $n$  é o tamanho da janela e  $m$  o número de elementos da série.

Outra medida usada será o desvio padrão, que informará a dispersão de valores em um dado intervalo, fornecendo uma descrição adicional da série. O desvio padrão, comumente representado pela letra  $\sigma$ , pode ser calculado da seguinte forma [9]:

$$\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}} \quad (2.12)$$

Exemplos de aplicação da média móvel e do desvio padrão, respectivamente, na Figura 2 e na Figura 3. Nos exemplos citados as médias móveis e desvio padrão estão deslocados, de modo que sejam centralizados em relação aos dados que os geraram.

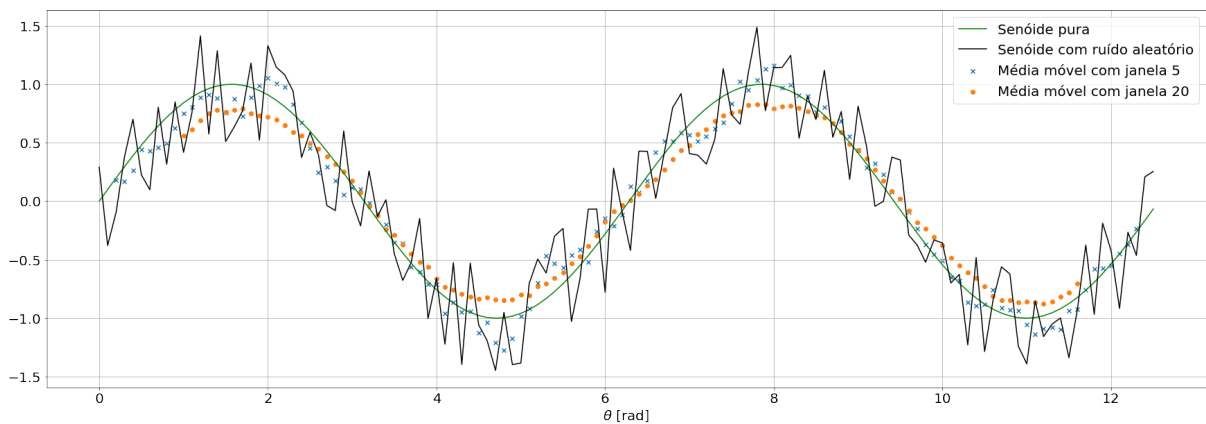


Figura 2 – Médias móveis com janelas 5 e 20.

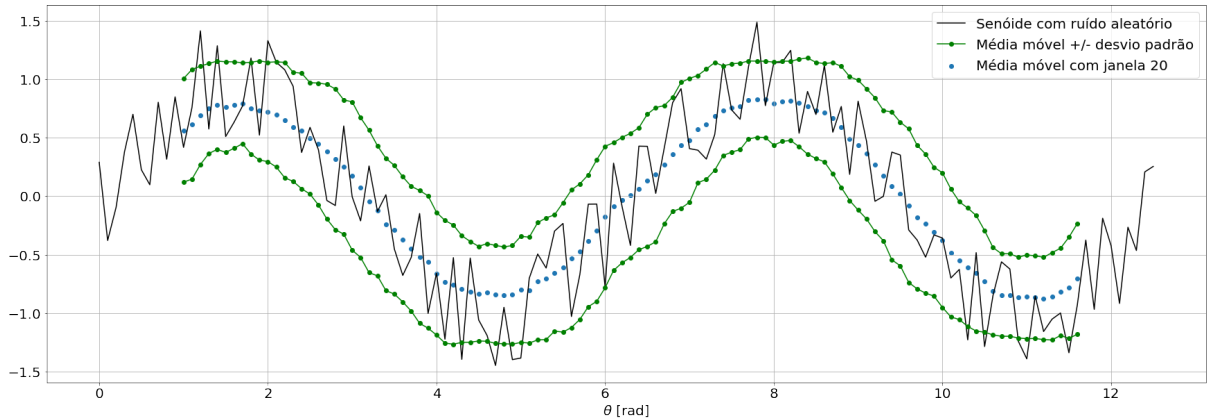


Figura 3 – Média móvel com desvio padrão.

### 2.3 Trabalhos anteriores

Diversos trabalhos vêm sendo publicados com objetivo de prever a velocidade do vento em diferentes horizontes de tempo. Nas publicações que investigam previsões de curto prazo têm se destacado os modelos que combinam métodos estatísticos tradicionais, ex., o modelo Auto-Regressivo de Médias Móveis (ARMA) e o Auto-Regressivo Integrado de Médias Móveis (ARIMA), com métodos baseados em Inteligência Artificial (IA), ex., Redes Neurais Artificiais (RNA) e Máquinas de Vetores de Suporte, mais conhecido como *Support-Vector Machines* (SVM) [12].

A referência [10] apresenta uma destas abordagens híbridas, combinando o método estatístico *Seasonal Autoregressive Integrated Moving Average* (SARIMA) com Redes Neurais Artificiais (RNA). Na primeira parte do método, as variáveis Temperatura, Umidade, Pressão e Direção do Vento são previstas pelo modelo SARIMA, em seguida é aplicada técnica *Principal Component Analysis* (PCA) para a redução de dimensionalidade, então as variáveis passam pela primeira RNA e mais uma etapa de PCA. Posteriormente são usadas como entrada em uma segunda RNA, que também recebe um vetor de Velocidades do Vento. Finalmente, essa segunda RNA exibe como saída a previsão de velocidades do vento.

O referido modelo utiliza dados meteorológicos de Junho de 2016 a Maio de 2017, das cidades de Petrolina-PE e Macau-RN. No artigo, o método é comparado com outros comumente usados (SARIMA, SARIMA+Wavelet e RNA) e alcança resultados melhores em todos os horizontes testados (6, 12, 18 e 24 horas).

A lógica difusa, ou lógica *fuzzy*, é outro método de inteligência computacional que vem sendo amplamente usado em conjunto com RNAs em modelos híbridos. A referência [13] propõe tal abordagem. Os autores de [13] apresentam um modelo que usa *Fuzzy Rough Sets* (FRS) para a redução de dimensionalidade das variáveis. Tais variáveis são então processadas por uma RNA do tipo *Long Short-Term Memory* (LSTM), que exibe

em sua saída a previsão de velocidade do vento.

Nesse artigo são usados, numa primeira análise, velocidade e direção do vento, temperatura, umidade e densidade do ar como variáveis de entrada. O modelo acaba por usar apenas velocidade do vento e temperatura, visto que as demais variáveis pouco influenciaram nas previsões e, desta forma, o processo de treinamento foi acelerado. O trabalho ainda mostra que o modelo FRS-LSTM prevê a velocidade do vento com menor erro do que os modelos LSTM e *Back Propagation* (BP).

Os autores de [14] fazem uma comparação de entre diversas técnicas de *Machine Learning* (ML), usadas para prever a velocidade do vento na região de Osório-RS, onde existe um parque eólico com 148 turbinas e uma capacidade instalada de 317.9 MW.

O trabalho analisa a performance na previsão dos horizontes de 5, 10, 15 e 30 minutos, testando os modelos *multi-layer feed-forward neural network* (MLFFNN), *support vector regression* (SVR), *fuzzy inference system* (FIS), *adaptive neuro-fuzzy inference system* (ANFIS), a RNA do tipo *group method of data handling* (GMDH), ANFIS otimizado com Enxame de Partículas (ANFIS-PSO) e ANFIS otimizado com Algoritmo Genético (ANFIS-GA). O modelo MLFFN obteve melhor performance nas métricas RMSE, R e MSE nos horizontes de 5, 10 e 15 minutos. No horizonte de 30 minutos o modelo SVR teve a melhor performance [14].

A transformada *wavelet* vem sendo amplamente explorada na análise e predição de séries temporais. Os autores de [15] aplicam *adptative wavelet neural network* (AWNN) à predição da velocidade do vento da estação meteorológica de *Trois-Rivieres*, Canadá. Na AWNN, a primeira camada decompõe o sinal em diferentes *wavelets*, em que cada *wavelet* é obtida por meio de uma transformação com parâmetros treináveis. Tais parâmetros serão ajustados através da derivada parcial do erro de previsão em relação ao próprio parâmetro. Após essa primeira camada o modelo tem a forma de uma RNA comum.

Esse artigo ainda analisa o coeficiente de *Hurst* para cada componente da transformada, o que torna possível detectar e eliminar componentes que não colaboram para a redução do erro de previsão, reduzindo assim a quantidade de processamento. O trabalho mostra que a AWNN prevê a velocidade do vento em horizontes de 1, 2, 3, 4 e 5 horas com menor erro do que um modelo que repete os valores do dia anterior, além de conseguir eliminar 2 componentes do sinal decomposto, reduzindo em 29% o processamento [15].

## 2.4 Redes Neurais Artificiais

A pesquisa de redes neurais artificiais, comumente chamadas apenas de redes neurais, foi motivada pela percepção de que o cérebro humano tem o funcionamento completamente diferente de um computador convencional. O cérebro é um sistema de processamento altamente complexo, não-linear e paralelo. Ele tem a capacidade de

organizar seus constituintes estruturais, conhecidos como neurônios, para executar tarefas como reconhecimento de padrões e controle de motores muito mais rapidamente do que os mais poderosos computadores atuais [16].

Isso é possível graças à sua capacidade de se adaptar ao ambiente que o cerca, se desenvolvendo à medida que obtém experiência. Essa característica é fundamental para o funcionamento dos neurônios, que são as unidades básicas de processamento de informações do cérebro humano. O mesmo acontece com uma rede neural composta de neurônios artificiais [16].

A ideia de redes neurais enquanto máquinas computacionais surgiu em 1943 com McCulloch and Pitts, mas foi Rosenblatt, em 1958, quem propôs o Perceptron, o primeiro modelo que era capaz de aprender através de um professor, o que é chamado de aprendizado supervisionado [16].

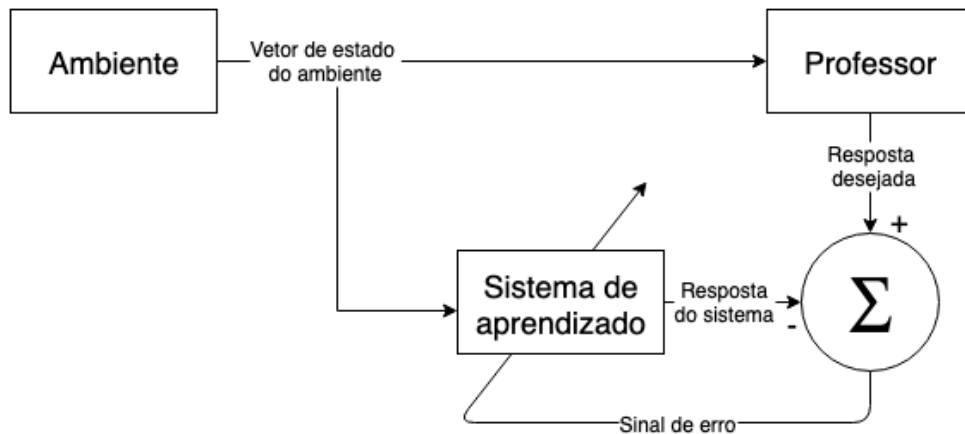


Figura 4 – Aprendizado supervisionado.

O aprendizado supervisionado, representado na Figura 4, pode ser facilmente entendido quando comparado a uma situação em que existe um ambiente, um professor e um sistema com capacidade de aprendizado, que aqui é representado pela RNA. Nesse contexto, o sistema não possui nenhum conhecimento inicial sobre o ambiente, enquanto o professor entende completamente o funcionamento do ambiente, que pode ser representado por pares de dados de entrada e saída. Dado um certo estado do ambiente (ou um vetor de dados de entrada) o professor é capaz de determinar qual será a saída, que é exatamente a resposta ótima que o sistema tem como objetivo exibir. Dessa forma, o sistema de aprendizado irá exibir respostas aleatórias inicialmente, porém, a cada resposta, um sinal de erro será gerado pela comparação com a resposta fornecida pelo professor, desta forma o sistema pode atualizar seu estado interno com o objetivo de minimizar o erro entre sua própria resposta e a fornecida pelo professor [16].

O Perceptron, ilustrado na Figura 5, é a forma mais simples de RNA e é usado na classificação de padrões linearmente separáveis. Basicamente, o Perceptron consiste em

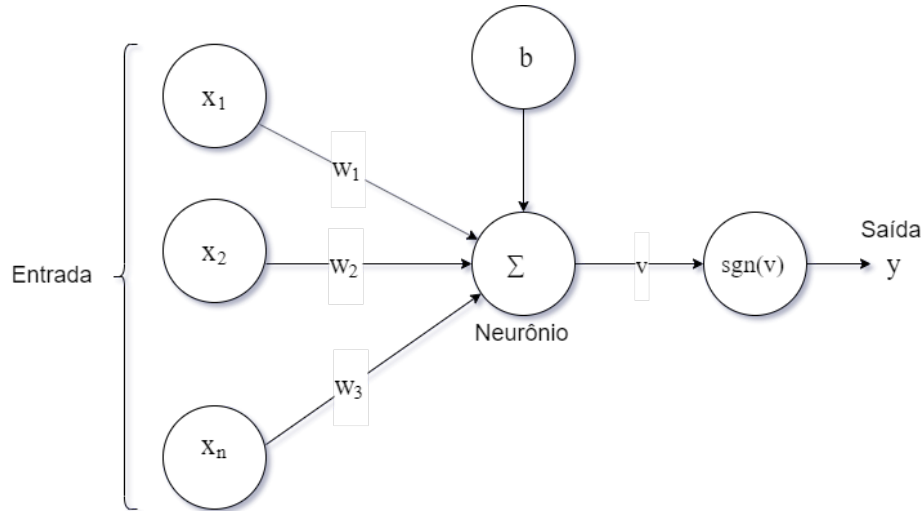


Figura 5 – Perceptron.

um único neurônio com pesos  $w_i$  e um viés  $b$  ajustáveis. A saída  $v$  do neurônio é calculada pela combinação linear da entrada com os pesos, com o viés externamente adicionado.

$$v = \sum_{i=1}^n x_i w_i + b \quad (2.13)$$

Essa saída passa então por um limitador, chamado por função de ativação, que pode ser representado pela função *sign*. Após o limitador temos a saída  $y$  do Perceptron.

$$y = \begin{cases} -1, & \text{para } v < 0 \\ 0, & \text{para } v = 0 \\ 1, & \text{para } v > 0 \end{cases} \quad (2.14)$$

Como mencionado anteriormente, o ajuste dos pesos  $w_i$  e viés  $b$  são realizados através da comparação entre a resposta do Perceptron e a resposta desejada, num sistema que posteriormente seria chamado *Backpropagation* (BP), que será melhor explicado na seção 3.3.3. O algoritmo para atualização dos pesos do Perceptron pode ser resumido pelo Algoritmo 1:

Em que:

- $\mathbf{x}(m) = [+1, x_1(m), x_2(m), \dots, x_n(m)]^T$ .
- $\mathbf{w}(m) = [b, w_1(m), w_2(m), \dots, w_n(m)]^T$ .
- $\eta$ : Taxa de aprendizado. Valor entre 0 e 1.
- $d = \begin{cases} -1, & \text{se } \mathbf{x}(n) \text{ pertence à classe 1} \\ 1, & \text{se } \mathbf{x}(n) \text{ pertence à classe 2} \end{cases}$

---

**Algoritmo 1:** Atualização dos pesos do Perceptron

---

1. Inicializar o vetor de pesos:  $\mathbf{w}(0) = 0$ .
  2. A cada passo  $m$  aplicar a entrada  $\mathbf{x}(m)$  ao Perceptron e fornecer a resposta desejada  $d(m)$ .
  3. Calcular resposta do Perceptron:  $y(m) = \text{sgn}[\mathbf{w}^T(m)\mathbf{x}(m)]$ .
  4. Atualizar o vetor de pesos:  $\mathbf{w}(m+1) = \mathbf{w}(m) + \eta[d(m) - y(m)]\mathbf{x}(m)$ .
  5. Incrementar  $m$  e voltar ao passo 2.
- 

O Perceptron, porém, é capaz apenas de decidir entre duas classes linearmente separáveis. Sua estrutura, hoje, é apenas a base para redes com muitos neurônios, camadas com diversas funções que são capazes de aperfeiçoar o processo de treinamento e uma infinidade de arquiteturas, o que possibilita que problemas extremamente complexos possam ser atacados. As técnicas que complementam o Perceptron para a realização deste trabalho serão abordadas no próximo capítulo.

### 3 DESENVOLVIMENTO

#### 3.1 Apresentação dos dados

Os dados para treinamento e teste do modelo de previsão de velocidades do vento foram obtidos através da plataforma SONDA, mantida pelo INMET. A estação de Petrolina, situada na latitude  $9^{\circ} 04' 08''$  Sul (-9,0689) e longitude  $40^{\circ} 19' 11''$  Oeste (-40,3197), foi escolhida para o estudo devido à grande disponibilidade de dados meteorológicos.

As variáveis disponíveis são divididas da seguinte forma:

- Variáveis ambientais - Periodicidade de 1 minuto:
  - Temperatura do Ar na Superfície ( $tp_{sfc}$ ): Temperatura do ar superficial em  $^{\circ}\text{C}$ .
  - Umidade do Ar ( $humid$ ): Umidade relativa do ar em %.
  - Pressão Atmosférica ( $press$ ): Pressão atmosférica em milibares.
  - Precipitação de Chuva ( $rain$ ): Precipitação de chuva em milímetros.
  - Velocidade do Vento a 10m ( $ws_{10m}$ ): Velocidade média do vento a 10m em  $m/s$ .
  - Direção do Vento a 10m ( $wd_{10m}$ ): Direção média do vento a 10m de  $0^{\circ}$  a  $360^{\circ}$  (sentido horário).
  - Radiação Global Horizontal ( $glo_{avg}$ ): Média da radiação global horizontal em  $W/m^2$ .
  - Radiação Direta Normal ( $dir_{avg}$ ): Média da radiação direta normal em  $W/m^2$ .
  - Radiação Difusa ( $diff_{avg}$ ): Média da radiação difusa em  $W/m^2$ .
  - Radiação de Onda Longa ( $lw_{avg}$ ): Média da radiação de Onda Longa descendente em  $W/m^2$  com correção.
  - Radiação PAR ( $par_{avg}$ ): Média da radiação fotossinteticamente ativa em  $\mu\text{mols s}^{-1}\text{m}^{-2}$ .
  - Iluminância ( $lux_{avg}$ ): Média da iluminância em  $kLux$ .
- Variáveis anemométricas - Periodicidade de 10 minutos:
  - Velocidade do Vento a 25m ( $ws_{25m}$ ): Velocidade média do vento a 25m em  $m/s$ .
  - Direção do Vento a 10m ( $wd_{25m}$ ): Direção média do vento a 25m de  $0^{\circ}$  a  $360^{\circ}$  (sentido horário).
  - Temperatura a 25m ( $tp_{25m}$ ): Temperatura do ar a 25m em  $^{\circ}\text{C}$ .

- Velocidade do Vento a 50m ( $ws_{50m}$ ): Velocidade média do vento a 50m em  $m/s$ .
- Direção do Vento a 50m ( $wd_{50m}$ ): Direção média do vento a 50m de  $0^\circ$  a  $360^\circ$  (sentido horário).
- Temperatura a 50m ( $tp_{50m}$ ): Temperatura do ar a 50m em  $^\circ C$ .

### 3.2 Tratamento dos dados

A estação possui dados anemométricos disponíveis no intervalo de Julho de 2004 a Abril de 2011; e dados ambientais no intervalo de Julho de 2004 a Fevereiro de 2011. Porém muitos meses possuem dados faltantes, dificultando o uso dos dados do período. Em vista disso, foi escolhido um período em que não houvesse muitos dados faltantes (Primeira Seleção).

Uma primeira análise foi feita a partir das informações de confiabilidade dos dados disponíveis na plataforma SONDA. Em seguida, dentro o período selecionado, foi feita uma nova seleção via inspeção visual, com o objetivo de identificar novos períodos de dados faltantes (Segunda Seleção). Os dados faltantes foram preenchidos com a repetição dos dados do instante de tempo imediatamente anterior. Esse método causa pouco impacto no treinamento quando ocorre em períodos curtos, porém, o uso foi evitado em períodos maiores. Os períodos maiores de dados faltantes são facilmente identificados na Figura 6. Os mesmos aparecem como linhas constantes após passarem pela substituição mencionada. Tal análise resultou no período escolhido de Julho de 2009 a Abril de 2011, um total de 1581 dias.

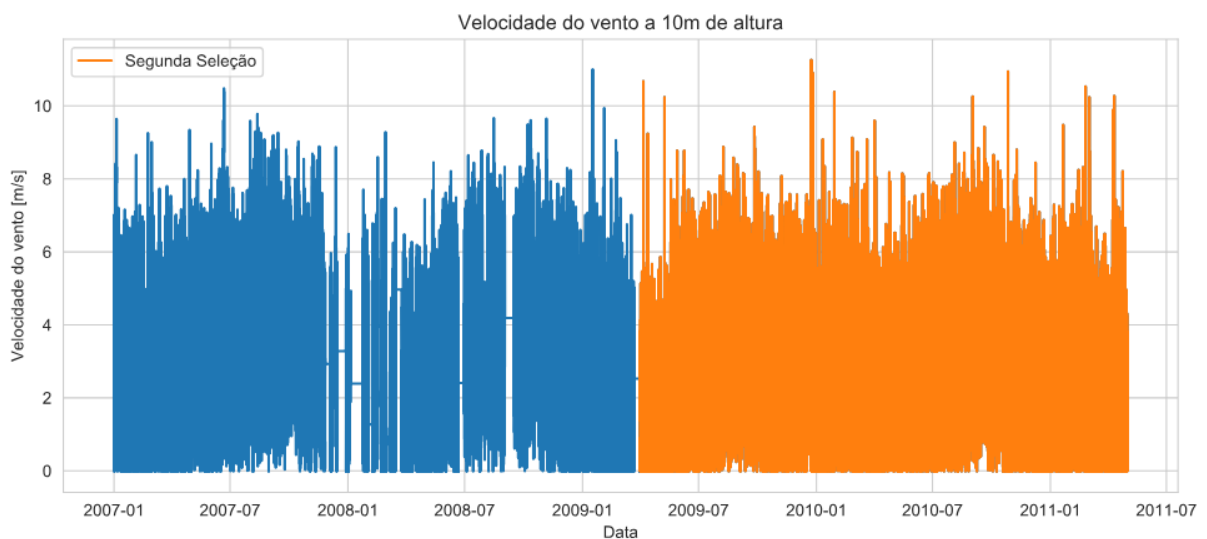


Figura 6 – Variável  $ws_{10m}$  no período 2007-2011.

Para a escolha das variáveis a serem usadas na entrada do modelo analisou-se a matriz de correlação, exibida na Figura 7. A correlação entre duas variáveis aleatórias



quaisquer  $X$  e  $Y$  é definida segundo a Equação 3.1 [17]. Quanto mais próximo de 1 o módulo da correlação, maior o grau de relacionamento entre as variáveis. É fácil notar visualmente que a variável *rain* teve correlação próxima de 0 com todas as demais variáveis, por essa razão a mesma não foi utilizada como variável de entrada do modelo. O restante das variáveis foi mantido, visto que, apesar de algumas não apresentarem boa correlação com as variáveis de velocidade do vento, possuem correlação significativa com outras variáveis. Isso é importante para o segundo passo de previsão em diante, pois um dos métodos utilizado para esses casos foi o *Multi-Step*, que será melhor explicado posteriormente.

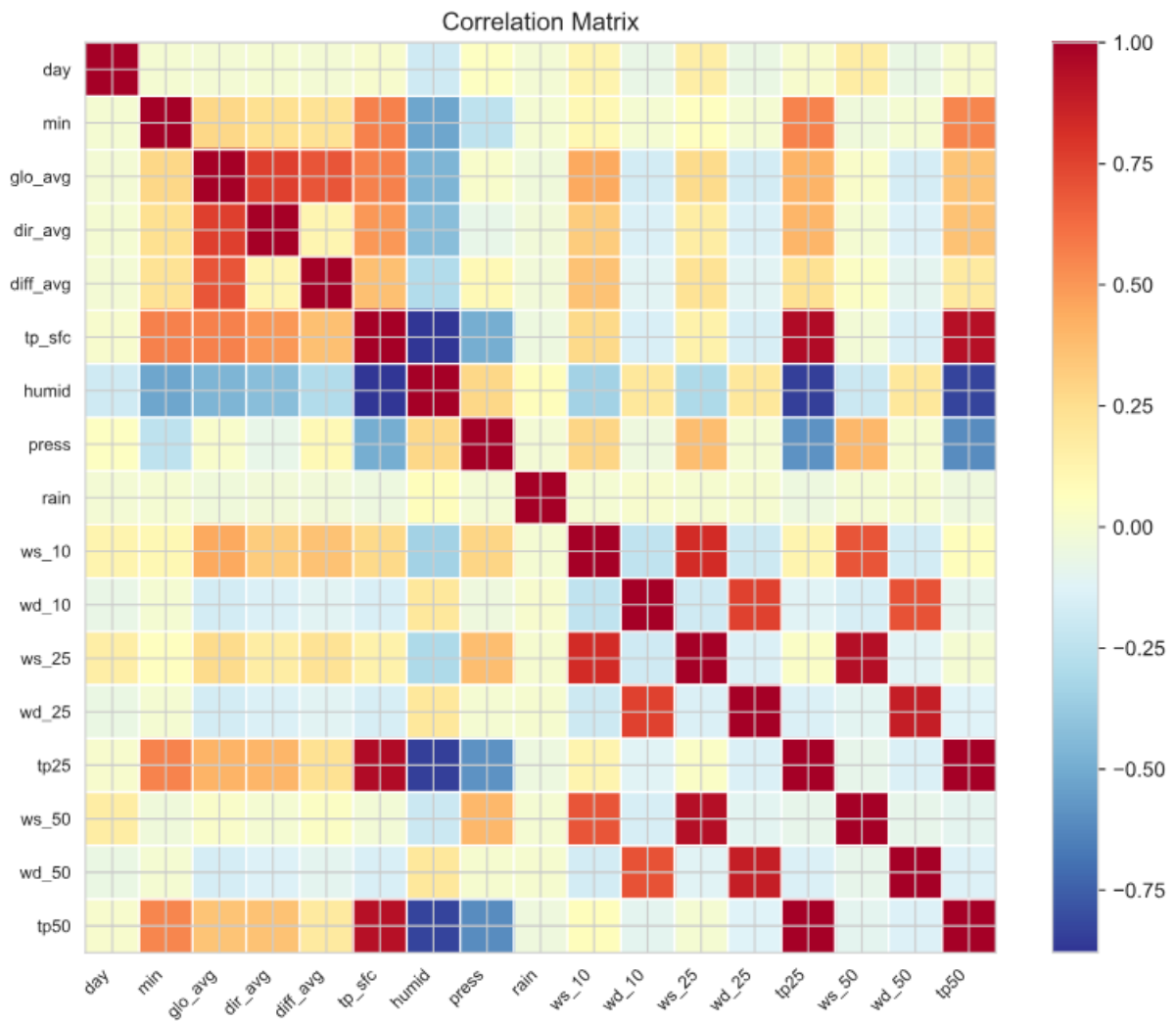


Figura 7 – Matriz de correlação.

$$\text{corr}(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (3.1)$$

Dentre as variáveis de velocidade do vento,  $ws_{10m}$ ,  $ws_{25m}$  e  $ws_{50m}$ , decidiu-se que o foco do estudo seria a variável  $ws_{10m}$ , referente à velocidade do vento a 10m de altura. A decisão se deu pelo fato da variável apresentar maior correlação com as demais variáveis

e, por isso, possuir maior previsibilidade. O ajuste da velocidade para outras alturas desejadas pode ser feito por meio das equações 2.9 e 2.10. Os histogramas das 3 variáveis são exibidos na Figura 8.

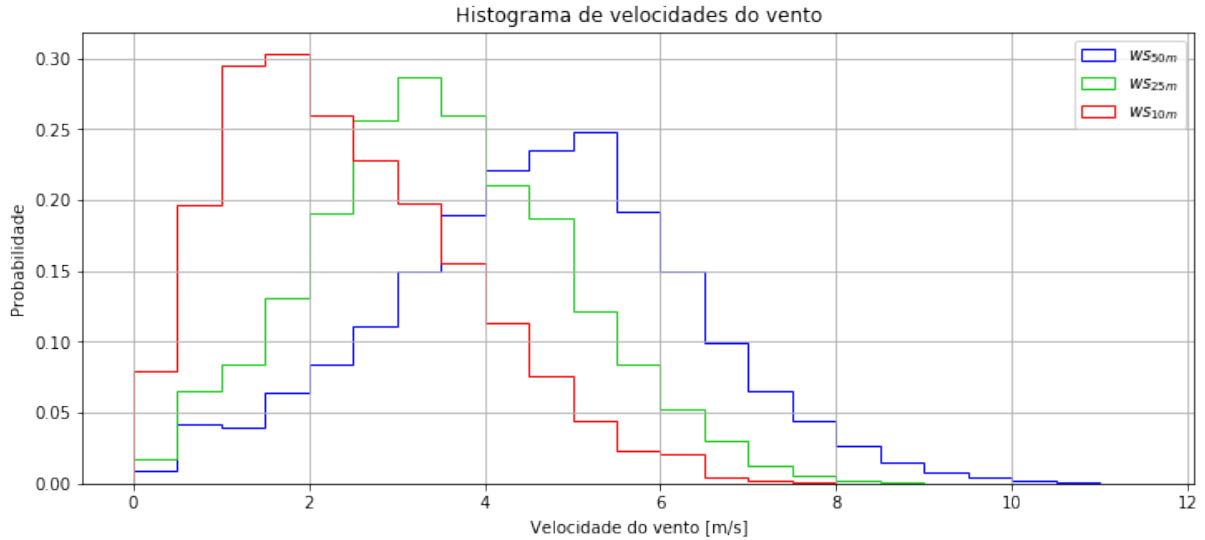


Figura 8 – Histograma de velocidades do vento.

A Tabela 1 fornece uma descrição das variáveis usadas. É possível perceber que algumas variáveis apresentam valores negativos, o que não tem sentido para nenhuma delas. Tais valores negativos representam erros de medição e aparecem em medições em que deveriam ser iguais a zero. Todos os valores negativos foram substituídos por 0. O valor extremo de -6999.00 para a variável  $diff_{avg}$  foi uma exceção que apareceu em um único ponto, essa medição também foi substituída por 0. Por análise visual também foram detectados outros erros de medição, com valores extremamente altos ou extremamente baixos em relação a média em poucos pontos específicos, nas variáveis  $ws_{25}$ ,  $tp_{sfc}$ ,  $humid$  e  $press$ . Esses valores foram substituídos pelo valor médio de cada variável.

Para o melhorar o treinamento de redes neurais é recomendável que as variáveis estejam na mesma faixa de valores. Todas as variáveis aleatórias foram então transformadas de modo a terem 0 e 1 como valores mínimo e máximo respectivamente. Para que essa transformação seja bem sucedida é importante fazer o tratamento descrito no parágrafo anterior. A nova descrição das variáveis é fornecida na Tabela 2.

### 3.3 Definição dos Modelos Preditivos

#### 3.3.1 Arquiteturas

Para fins de comparação, será estudada a performance de dois modelos distintos, que serão individualmente descritos e analisados a seguir. Ambos os modelos têm como base de seu funcionamento as ideias do perceptron e do aprendizado supervisionado, abordados na seção 2.4.

	<b>contagem</b>	<b>média</b>	<b>desvio padrão</b>	<b>mínimo</b>	<b>máximo</b>
<i>dia</i>	1.09E+06	179.84	104.51	1.00	365.00
<i>min</i>	1.09E+06	719.32	415.64	0.00	1439.00
<i>glo<sub>avg</sub></i>	1.09E+06	225.89	321.74	-10.49	1488.00
<i>dir<sub>avg</sub></i>	1.09E+06	154.22	294.60	-45.93	986.00
<i>diff<sub>avg</sub></i>	1.09E+06	122.59	199.33	-6999.00	1557.00
<i>tp<sub>sfc</sub></i>	1.09E+06	26.45	4.06	-54.74	38.48
<i>humid</i>	1.09E+06	61.14	19.25	-5.81	96.20
<i>press</i>	1.09E+06	969.49	3.63	0.00	978.83
<i>ws<sub>10m</sub></i>	1.09E+06	2.40	1.39	-0.02	11.26
<i>wd<sub>10m</sub></i>	1.09E+06	139.78	53.44	0.00	360.00
<i>ws<sub>25</sub></i>	1.09E+06	3.54	2.15	-0.07	92.50
<i>wd<sub>25m</sub></i>	1.09E+06	142.92	48.60	0.01	360.00
<i>tp<sub>25</sub></i>	1.09E+06	26.39	3.71	14.00	36.69
<i>ws<sub>50m</sub></i>	1.09E+06	4.63	1.76	-0.03	15.22
<i>wd<sub>50m</sub></i>	1.09E+06	137.01	47.43	0.03	360.00
<i>tp<sub>50</sub></i>	1.09E+06	25.99	3.65	14.41	35.94

Tabela 1 – Descrição dos dados.

	<b>contagem</b>	<b>média</b>	<b>desvio padrão</b>	<b>mínimo</b>	<b>máximo</b>
<i>dia</i>	1.09E+06	0.4913	0.2871	0	1
<i>min</i>	1.09E+06	0.4999	0.2888	0	1
<i>glo<sub>avg</sub></i>	1.09E+06	0.1524	0.2158	0	1
<i>dir<sub>avg</sub></i>	1.09E+06	0.1568	0.2986	0	1
<i>diff<sub>avg</sub></i>	1.09E+06	0.0790	0.1278	0	1
<i>tp<sub>sfc</sub></i>	1.09E+06	0.4876	0.1729	0	1
<i>humid</i>	1.09E+06	0.5826	0.2292	0	1
<i>press</i>	1.09E+06	0.4887	0.1547	0	1
<i>ws<sub>10m</sub></i>	1.09E+06	0.2130	0.1235	0	1
<i>wd<sub>10m</sub></i>	1.09E+06	0.3883	0.1485	0	1
<i>ws<sub>25</sub></i>	1.09E+06	0.2799	0.1166	0	1
<i>wd<sub>25m</sub></i>	1.09E+06	0.3970	0.1350	0	1
<i>tp<sub>25</sub></i>	1.09E+06	0.5461	0.1633	0	1
<i>ws<sub>50m</sub></i>	1.09E+06	0.3041	0.1158	0	1
<i>wd<sub>50m</sub></i>	1.09E+06	0.3805	0.1318	0	1
<i>tp<sub>50</sub></i>	1.09E+06	0.5377	0.1694	0	1

Tabela 2 – Descrição dos dados após tratamento.

### 3.3.1.1 *Multilayer Perceptron*

Para contornar as limitações do Perceptron, foram desenvolvidas as redes *Multilayer Perceptron* (MLP), que, ao invés de possuírem apenas um neurônio, são compostas por camadas de neurônios. As camadas localizadas entre a entrada e a saída da rede são chamadas camadas escondidas, ou camadas internas [16]. A Figura 9 ilustra uma rede com duas camadas escondidas, a primeira com 4 neurônios e a segunda com 5 neurônios. O funcionamento dos neurônios é muito parecido ao Perceptron de Rosenblatt, com a única diferença de que a função de ativação pode ser qualquer função não linear diferenciável [16].

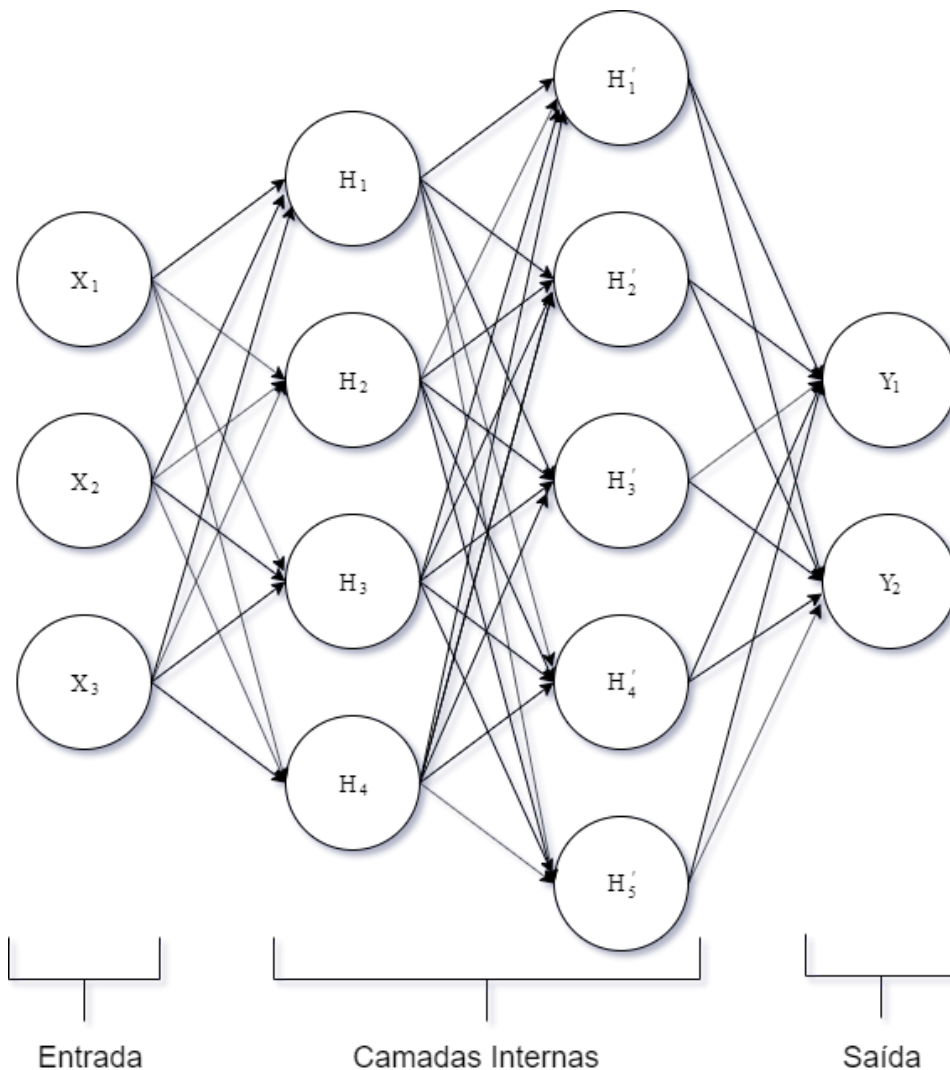


Figura 9 – MLP com duas camadas escondida.

A função de ativação usada nas camadas escondidas MLP do presente trabalho foi a *Rectified Linear Unit*, ou simplesmente ReLU. A função ReLU tem a saída igual a entrada para valores de entrada maiores que 0, e tem a saída igual a 0 para entradas menores que 0. A função está representada na Figura 10. Já a saída da rede possui ativação linear, ou seja, exibe na saída o próprio valor de entrada.

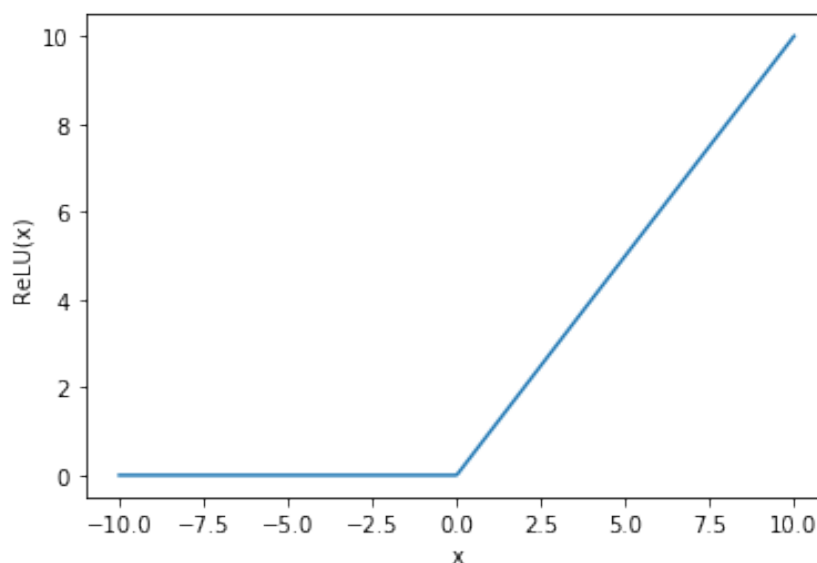


Figura 10 – ReLU.

O Teorema da Aproximação Universal determina que uma MLP com uma única camada escondida é capaz de aproximar qualquer função não linear [18]. Porém, o teorema não determina o número de neurônios que essa camada precisa ter, podendo ser esse um número inviável para implementação. Redes com mais de uma camada escondida têm se mostrado muito úteis na detecção de padrões mais complexos. Tais redes possuem a capacidade de detectar padrões em diferentes níveis de profundidade em cada camada escondida [16]. As camadas da MLP proposta possuem conexões entre todos os neurônios das camadas adjacentes e são chamadas de totalmente conectadas, ou camadas densas.

O modelo de MLP proposto está ilustrado na Figura 11. A rede é composta de 3 camadas densas, a primeira com 256 neurônios, a segunda com 64 e a terceira com 16. As dimensões da entrada e da saída da rede e a diferença entre os modelos A-1 e A-2 serão explicadas na seção 3.3.2.

### 3.3.1.2 Long-Short Term Memory

*Long-Short Term Memory* (LSTM) é um tipo de RNA recorrente que foi proposto por Sepp Hochreiter e Jürgen Schmidhuber em 1997 [19]. Redes Neurais Recorrentes são redes que permitem a modelagem de dependência temporal nos dados, utilizando-se de realimentação com atrasos temporais, como pode ser visto na Figura 12. O surgimento da LSTM permitiu grande avanço no processamento de linguagem natural e hoje vem sendo amplamente aplicada em diversos campos, incluindo a predição de séries temporais.

O diferencial desta técnica é a possibilidade de detectar dependências temporais mais distantes, em contraste às redes recorrentes anteriores que sofriam do problema do *vanishing gradient*. Nas redes anteriores, o sinal de erro se tornava desprezível na atualização dos pesos das conexões mais remotas, sendo possível apenas a detecção de

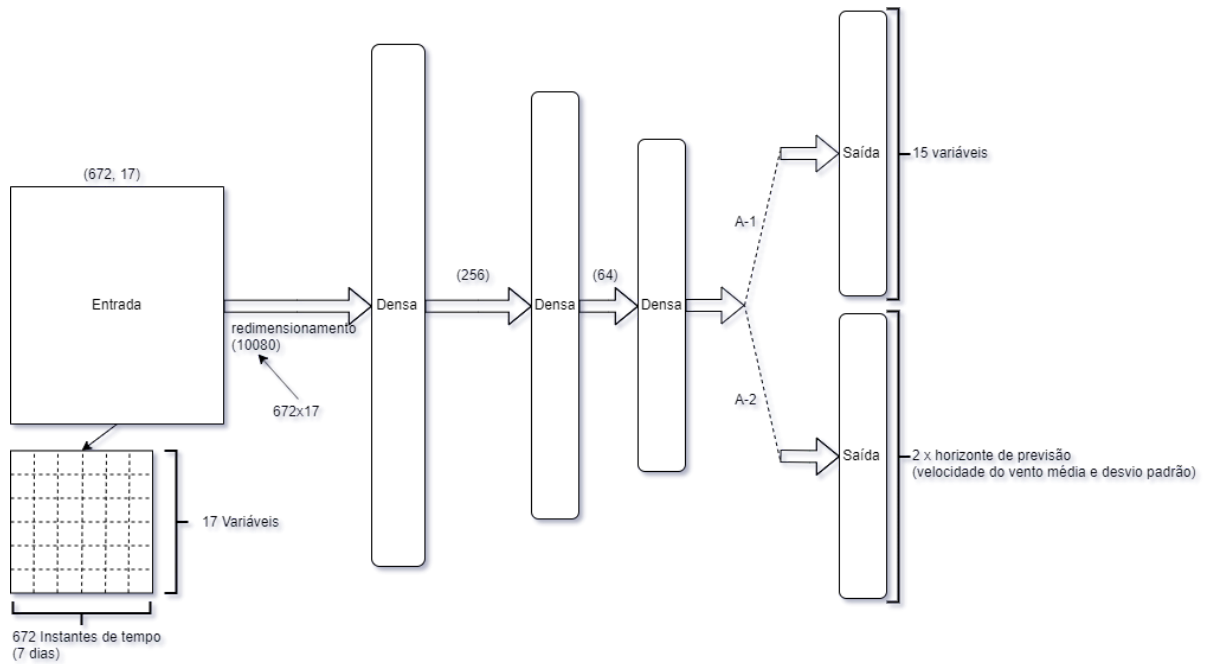


Figura 11 – Modelos A-1 e A-2.

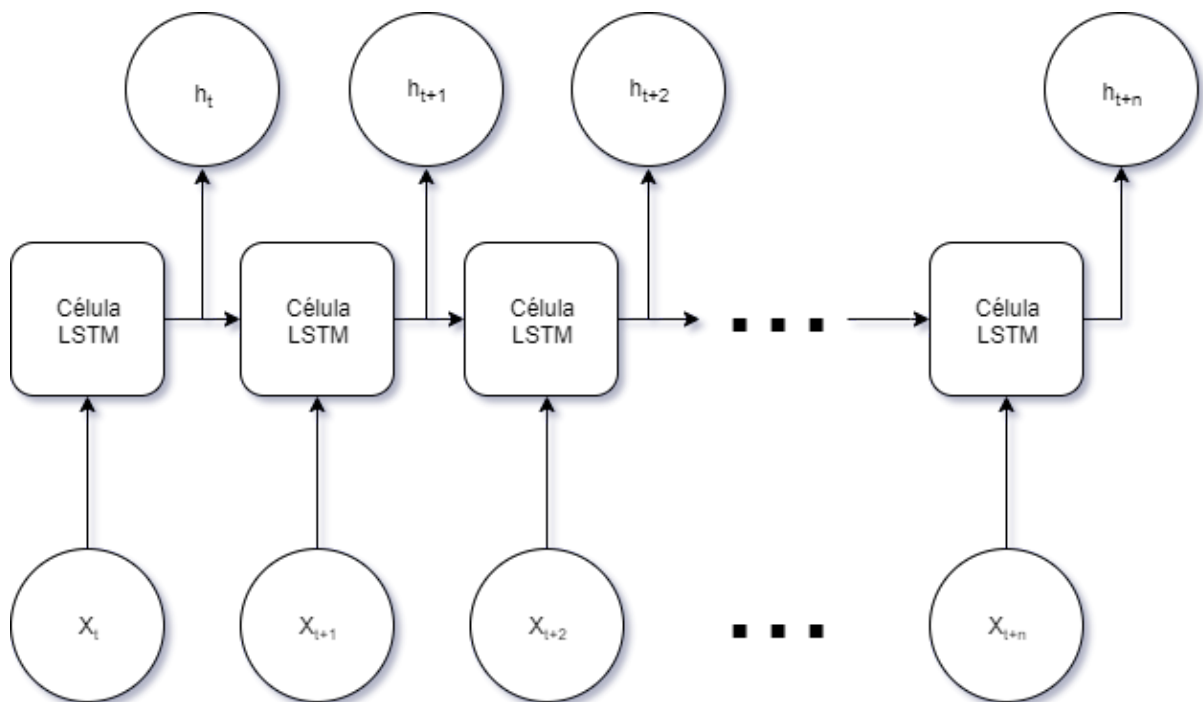


Figura 12 – Arranjo de células LSTM.

padrões em um número muito reduzido de atrasos temporais [19].

Para isso, cada célula LSTM, representada na Figura 13, utiliza-se de 3 *gates* que controlam o fluxo de informações, determinando quais devem ser guardadas e quais devem ser descartadas. A ideia central que possibilita a preservação do sinal através um grande número de atrasos temporais é a forma como os estados são atualizados, sofrendo pequenas alterações, sendo elas apenas um multiplicação por um valor entre 0 e 1, e a adição de um

valor entre -1 e 1.

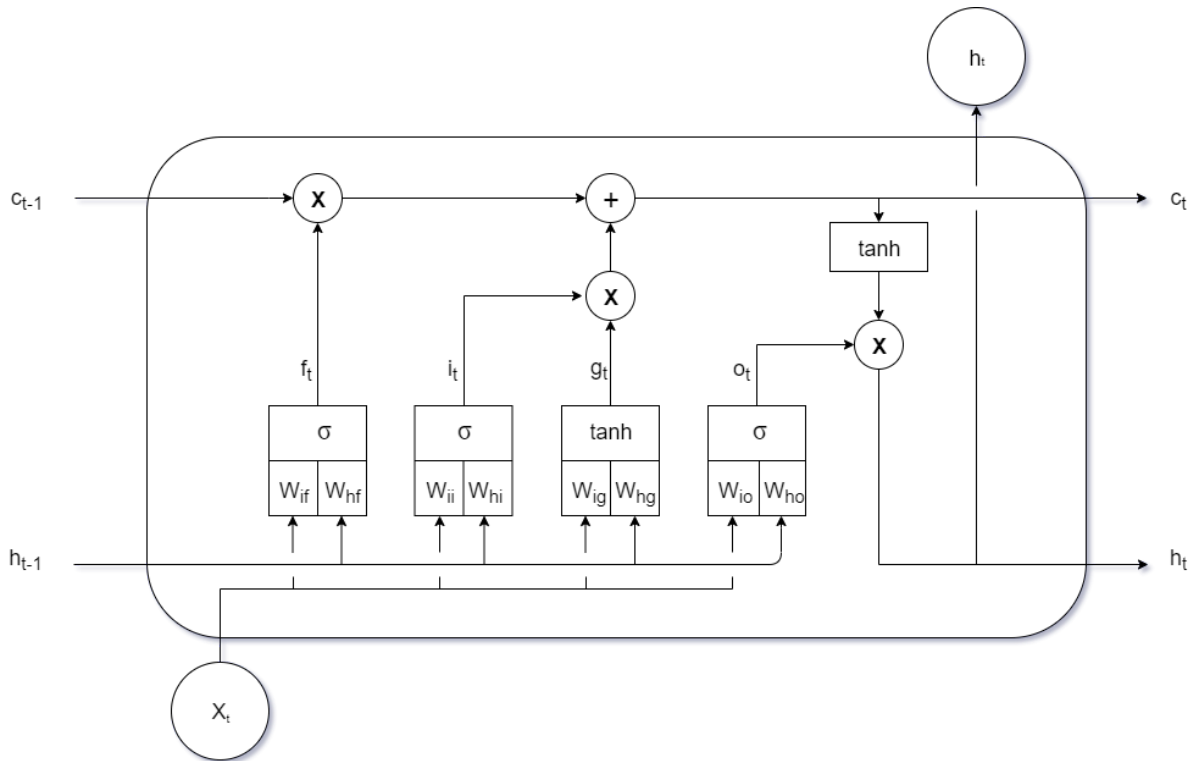


Figura 13 – Célula LSTM.

O primeiro *gate* é o *forget gate*, denotado por  $f_t$ , que olha para a entrada atual  $x_t$  e a saída da célula anterior  $h_{t-1}$  e aplica a função sigmoide, denotada pela letra grega  $\sigma$  e ilustrada na Figura 14, à combinação linear dos dois. Desse modo, o valor 0 no *forget gate* significa que a informação passada deve ser totalmente descartada, enquanto com o valor 1 a informação é totalmente preservada. O valor de  $f_t$  é dado por:

$$f_t = \sigma(x_t W_{if} + h_{t-1} W_{hf} + b_f) \quad (3.2)$$

em que  $W_{if}$ ,  $W_{hf}$  e  $b_f$  são os pesos e o *bias* do *forget gate*, respectivamente.

O gate seguinte, chamado *input gate layer*, é composto por  $i_t$  e  $g_t$  e é responsável por determinar qual será a nova informação adicionada, atualizando o estado anterior  $c_{t-1}$  para  $c_t$ . O valor de  $c_t$  pode ser determinado através das relações seguintes:

$$i_t = \sigma(x_t W_{ii} + h_{t-1} W_{hi} + b_i) \quad (3.3)$$

$$g_t = \sigma(x_t W_{ig} + h_{t-1} W_{hg} + b_g) \quad (3.4)$$

$$c_t = c_{t-1} + i_t g_t \quad (3.5)$$

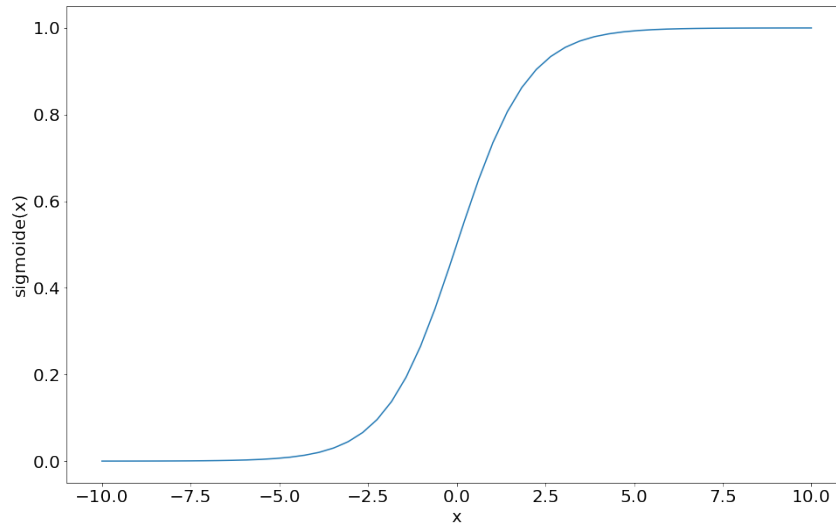


Figura 14 – Função sigmoide.

Finalmente, o último gate, chamado *output gate* e denotado por  $o_t$ , determina o que a célula exibirá como saída. A saída da célula  $h_t$  é determinada pelas seguintes relações:

$$o_t = \sigma(x_t W_{io} + h_{t-1} W_{ho} + b_o) \quad (3.6)$$

$$h_t = o_t \tanh(c_t) \quad (3.7)$$

Os modelos B-1 e B-2 propostos no presente trabalho, compostos por 1 camada LSTM e 2 camadas Densas é ilustrado na Figura 15. O número de neurônios numa camada LSTM representa a dimensionalidade do vetor  $h_t$ , que nos modelos propostos é igual a 64. Ademais, cada saída  $h_t$  não é necessariamente usada no restante da rede. Nos modelos aqui propostos apenas a última saída  $h_{t+n}$ , que possui dimensionalidade 64, é usada como entrada da primeira camada Densa.

### 3.3.2 Divisão dos dados

As definições do período de atraso e do número de atrasos temporais foram feitas de forma empírica. Para que os testes com o período de atraso pudessem ser feitos, as variáveis com periodicidade de 10 minutos foram transformadas de forma que sua periodicidade passasse a ser de 1 minuto. Isso foi feito simplesmente repetindo cada ponto do vetor por mais 9 posições. Posteriormente foi extraída das variáveis a média móvel com a janela igual ao período de atraso desejado. No vetor final foram usados apenas os pontos de média móvel cujos centros se distanciam em 15 minutos. Ademais, foi extraído o desvio padrão das janelas, que foi usado como entrada e saída dos modelos, de modo que as redes façam a previsão não só da velocidade do vento média para o horizonte desejado, mas também do desvio padrão.



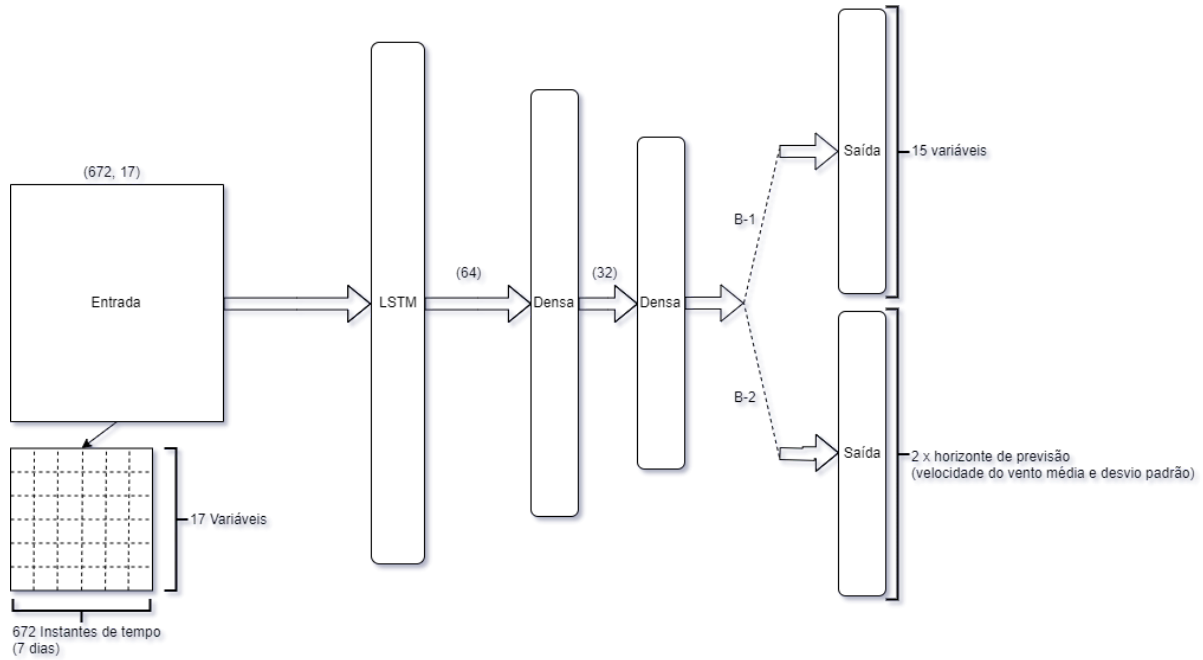


Figura 15 – Modelos B-1 e B-2.

Os valores finais foram atrasos de 15 minutos e 672 pontos de dados para cada uma das 17 variáveis, resultando num histórico de 7 dias.

$$15 \times 672 = 10080 \text{ minutos} = 7 \text{ dias}$$

Dessa forma, o vetor de entrada compartilhado pelos 4 modelos analisados é uma matriz de dimensão (672, 17).

A saída dos modelos foi estudada de duas formas. Na primeira, usada nos modelos A-1 e B-1, a rede faz a previsão de 15 variáveis (as 17 de entrada menos *dia* e *min*) apenas para os 15 minutos à frente. Então os instantes de tempo mais antigos do vetor de entrada são excluídos e a previsão resultante é adicionada, mantendo a dimensão original do vetor. Dessa forma, novas previsões de 15 minutos podem ser feitas quantas vezes forem desejadas. Esse método de previsão é chamado *multi-step* e está ilustrado na Figura 16.

A segunda forma foi o método *single-step*, ilustrado na Figura 17, que consiste em fazer a previsão do horizonte total desejado em um único passo. Esse método foi usado nos modelos A-2 e B-2, sendo que a dimensão de suas saídas dependem do horizonte de previsão desejado e, portanto, para cada horizonte existe um modelo único. Nesses modelos a saída tem a dimensão de duas vezes o horizonte desejado, pois são previstos, para cada instante de tempo, a velocidade média do vento a 10 metros de altura e o desvio padrão no intervalo.

Após a transformação dos dados para a periodicidade desejada, obteve-se um conjunto final de 72000 pontos de dados por variável. Desses, 70% foram destinados ao treinamento dos modelos, 15% para validação e 15% para teste. A divisão dos conjuntos

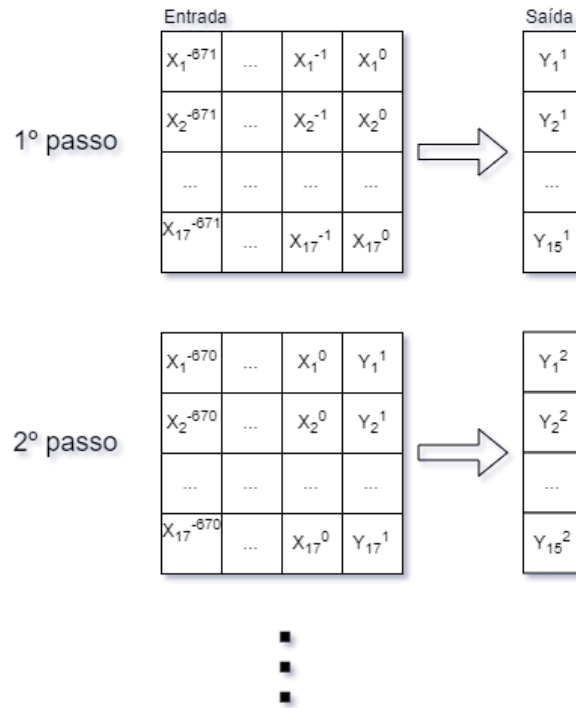


Figura 16 – *Previsão Multi-Step*.

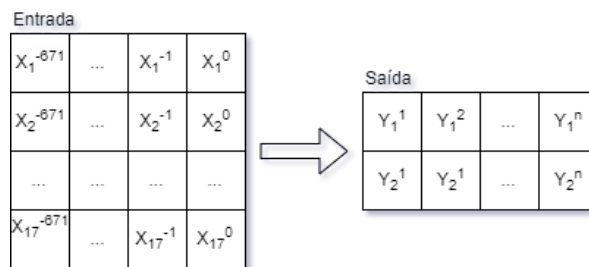


Figura 17 – *Previsão Single-Step*.

pode ser visualizada na Figura 18. Essa divisão entre treino, validação e teste é importante para que os resultados tenham significado real, uma vez que todas as métricas serão extraídas da aplicação do modelo em dados que não foram usados para o treinamento do mesmo. O conjunto de validação permite que se observe a capacidade de generalização do modelo, servindo como uma referência de qualidade durante o treinamento, e assim auxiliando na decisão sobre a parada do treinamento. Os detalhes do treinamento serão apresentados na seção seguinte.

### 3.3.3 Treinamento

Denomina-se treinamento o processo de adaptação do modelo aos dados. Inicialmente, cria-se o modelo com pesos aleatórios, então o processo de treinamento os ajusta para que o modelo exiba, para cada entrada, a saída desejada. Nos modelos aqui propostos a atualização dos pesos foi feita através do algoritmo *Backpropagation*, que, como o nome

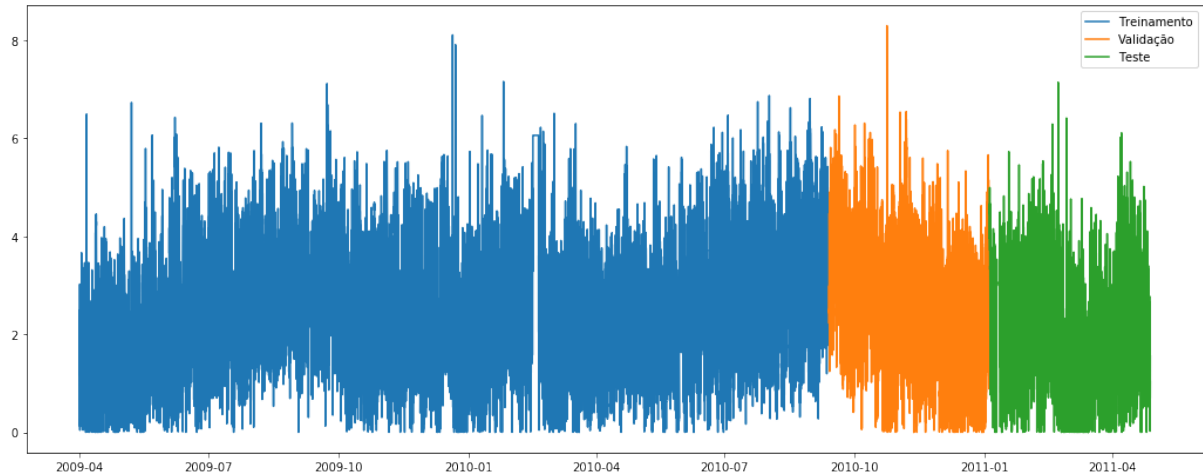


Figura 18 – Conjuntos de Treino, Validação e Teste.

sugere, consiste em propagar o sinal de erro, obtido pela comparação entre a saída exibida pela rede e a saída desejada, em direção à entrada da rede, atualizando cada peso de acordo com sua contribuição para o erro. A contribuição de cada peso para o erro é determinada através da derivada parcial da função de erro em relação à cada peso, utilizando-se da regra da cadeia. Percebe-se então a necessidade das funções de ativação e da função de erro serem diferenciáveis [16].

A função de erro usada no treinamento dos modelos propostos foi a Mean Squared Error (MSE), definida pela seguinte equação:

$$MSE = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (3.8)$$

em que  $\hat{Y}_i$  é cada saída prevista pelo modelo e  $Y_i$  a respectiva saída desejada.

O objetivo do treinamento é encontrar valores para os pesos que minimizem a função de erro, o que pode ser encarado como um problema de otimização. Em problemas de Deep Learning, a função de erro é normalmente não-linear e possui muitos mínimos locais. Por isso, diversos otimizadores são usados atualmente, sendo cada um mais apropriado para cada tipo de problema [16], [20].

O otimizador usado para o treinamento dos modelos foi o *Adaptive Moment Estimation*, ou Adam, como é mais conhecido [21]. Esse otimizador tem a característica de convergir facilmente para diversos tipos de problemas complexos, apesar de não apresentar a melhor precisão quando comparado a outros otimizadores. A escolha se deu pelo fato de ser um estudo inicial do tema, de modo que a possibilidade de testar diversas variações dos modelos foi considerada mais importante do que a precisão dos resultados. Ademais, segundo os autores do otimizador, o mesmo é recomendado para problemas com conjuntos de dados muito grandes, ruidosos e não estacionários, e modelos com muitos parâmetros

(pesos), que são características do atual problema.

O otimizador Adam é baseado no cálculo do gradiente local da função de erro, buscando atualizar os pesos na direção em que o gradiente é negativo, de modo a encontrar pontos de mínimo. Adam possui três parâmetros. Um deles é a taxa de aprendizado, que consiste em determinar em que intensidade a atualização será aplicada nos pesos, sendo que uma taxa muito baixa tende a resultar num processo de convergência lenta, enquanto uma taxa muito alta tende a resultar num treinamento muito instável, em que o erro oscila indefinidamente. A taxa de aprendizado usada em todos os treinamentos foi de 0,0001, sendo esse um valor de referência, pois nesse otimizador a taxa de aprendizado é calculada individualmente para cada peso e a cada passo. Os outros dois parâmetros do otimizador são relativos à sua funcionalidade de não estagnar em mínimos locais, conhecida como *momentum*, que também é adaptativa. Esses dois parâmetros  $\beta_1$  e  $\beta_2$  foram usados no valor padrão do artigo original 0,9 e 0,999, respectivamente.

A atualização dos pesos pode ainda ser executada com diferentes períodos, sendo que em relação a isso o método adotado foi o *Mini-Batch*, que consiste em aplicar uma determinada quantidade de amostras ao modelo, computar o erro e então fazer uma atualização baseada em todo esse conjunto. O número de amostras usadas para cada atualização foi 64. Esse é um fator que também deve ser cuidadosamente selecionado, pois está relacionado ao uso de memória e à velocidade do treinamento, além de poder impactar na precisão do modelo final [16].

Por fim, o critério de parada do treinamento foi um número fixo de épocas. Uma época é finalizada quando todas as amostras do conjunto de treinamento são submetidas ao modelo, tendo seu erro calculado e sendo os pesos atualizados à cada *batch* (64 amostras) submetida. Um critério de parada do treinamento comumente usado é o ponto de mínimo da função de erro do conjunto de validação em relação à época. Essa é uma curva que tende a cair rapidamente nas épocas iniciais e a partir de um certo ponto subir indefinidamente, indicando que o modelo está se tornando enviesado em relação ao conjunto de treinamento, e assim perdendo a capacidade de generalização, o que dá-se o nome de *overfitting*.

Porém, o problema presente é computacionalmente muito custoso, por isso foram determinados números de épocas em que foi possível perceber que não haveria queda significativa do erro nas épocas seguintes. Tais valores foram encontrados executando-se uma vez o algoritmo por um número de épocas maior. O número de épocas para os treinamentos foi de 50 épocas para os modelos A-1 e A-2, e 20 épocas para os modelos B-1 e B-2. Gráficos da função de erro do conjunto de treinamento e de teste são fornecidos nas Figuras 19 e 20. A curva em azul representa a função de erro para o conjunto de teste e a laranja a função de erro para o conjunto de validação

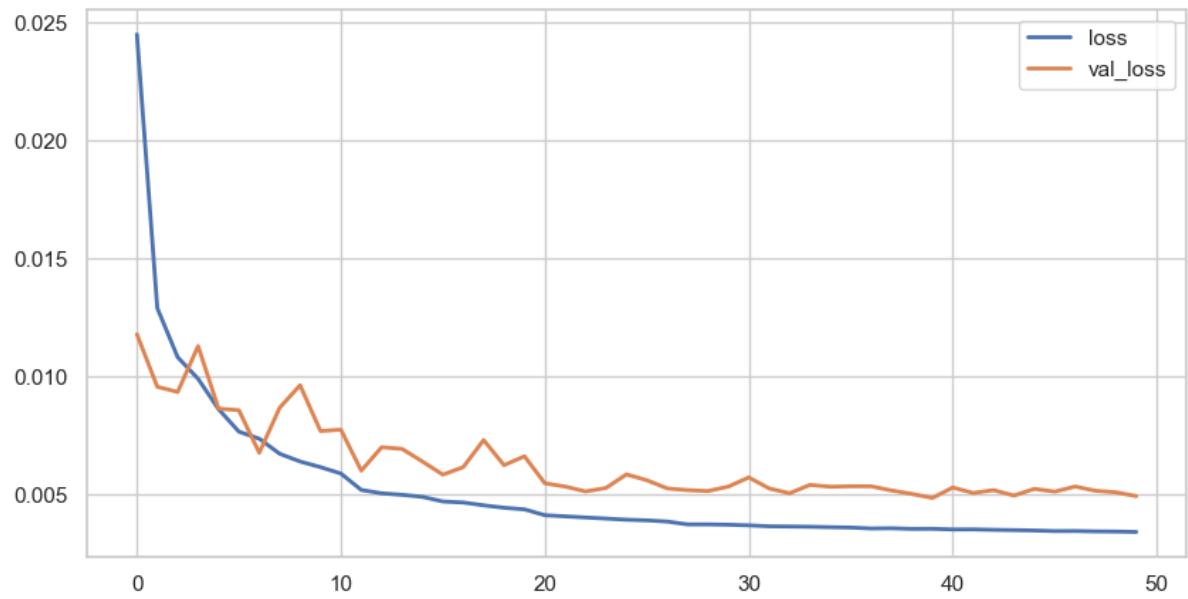


Figura 19 – Treinamento modelo A-2.

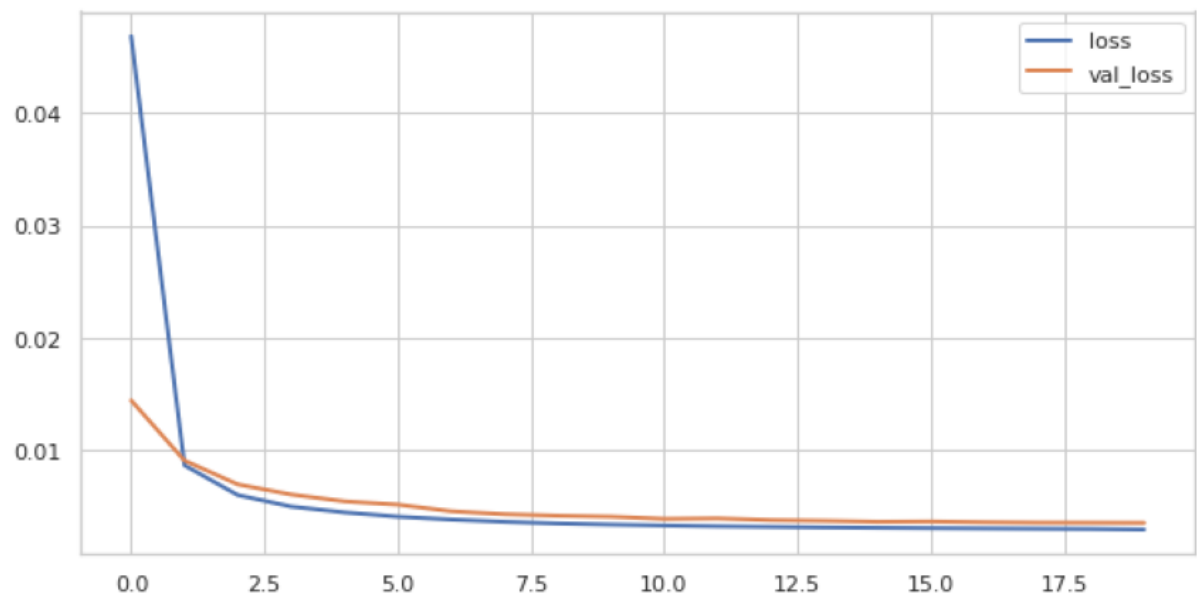


Figura 20 – Treinamento modelo B-2.

## 4 RESULTADOS

Os modelos foram implementados na linguagem Python, utilizando a biblioteca *open-source* Keras. A biblioteca Keras dá suporte a toda a modelagem matemática das RNAs e dos otimizadores, tornando possível a codificação dos modelos com mais agilidade e eficiência. A execução do treinamento foi feita na plataforma gratuita Google Colaboratory, que conta com uma TPU (*Tensor Processing Unit*) de 35.35 Gb de memória RAM, o que acelerou consideravelmente os treinamentos. Os modelos A-2 e B-2 levaram em média 13 minutos para treinar cada versão, treinadas separadamente para cada horizonte de previsão. Os modelos A-1 e B-1 levaram em média 190 minutos no treinamento.

Para a avaliação e comparação dos modelos foram usadas, além da MSE, definida na equação 3.8, outras 2 métricas que são comumente utilizadas em problemas de regressão:

*Mean Absolute Error*

$$MAE = \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \quad (4.1)$$

*Root Mean Squared Error*

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2} \quad (4.2)$$

Ademais, foi calculado o erro de potência média e energia por metro quadrado disponíveis no vento, através da equação 2.3, obtendo-se as relações a seguir:

$$Erro_{potência} = \frac{1}{n} \sum_{i=1}^n |0.647(Y_i^3 - \hat{Y}_i^3)| \quad (4.3)$$

$$Energia_{observada} = \sum_{i=1}^n 0.647Y_i^3 t \quad (4.4)$$

$$Energia_{prevista} = \sum_{i=1}^n 0.647\hat{Y}_i^3 t \quad (4.5)$$

$$Erro_{energia} = |Energia_{observada} - Energia_{prevista}| \quad (4.6)$$

em que  $t$  é a duração de cada velocidade, que nos modelos propostos é igual a 15 minutos.

As 5 métricas citadas foram extraídas da execução dos modelos no conjunto de teste e são exibidas nas tabelas 4, 3, 6 e 5.

É possível perceber que para o horizonte de previsão de 1 hora os 4 modelos tiveram performances próximas, porém, para os horizontes maiores os modelos A-2 e B-2

Horizonte [h]	MAE		MSE		RMSE		Erro de Energia [Wh/m <sup>2</sup> ]		Erro de Potência [W/m <sup>2</sup> ]	
	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão
1	0.5662	0.4042	0.5804	1.0565	0.6310	0.4268	4.51	7.12	4.51	7.12
2	0.6131	0.3789	0.6727	0.9970	0.7070	0.4157	9.27	12.92	4.64	6.46
3	0.6451	0.3632	0.7389	0.9595	0.7569	0.4073	14.19	18.28	4.73	6.09
4	0.6687	0.3501	0.7890	0.9353	0.7937	0.3988	19.14	23.21	4.79	5.80
5	0.6872	0.3380	0.8292	0.9080	0.8228	0.3901	24.08	27.74	4.82	5.55
6	0.7022	0.3279	0.8629	0.8829	0.8467	0.3820	29.00	31.90	4.83	5.32

Tabela 3 – Resultados modelo A-1.

Horizonte [h]	MAE		MSE		RMSE		Erro de Energia [Wh/m <sup>2</sup> ]		Erro de Potência [W/m <sup>2</sup> ]	
	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão
1	0.5607	0.3648	0.5397	0.8791	0.6256	0.3851	4.43	6.89	4.43	6.89
2	0.5983	0.3283	0.6033	0.7736	0.6891	0.3584	8.89	12.28	4.44	6.14
3	0.6168	0.3020	0.6345	0.6972	0.7216	0.3372	13.20	17.37	4.40	5.79
4	0.6297	0.2860	0.6628	0.6681	0.7464	0.3251	17.55	22.19	4.39	5.55
5	0.6500	0.2682	0.6956	0.6169	0.7752	0.3077	21.59	25.05	4.32	5.01
6	0.6558	0.2539	0.7073	0.5863	0.7874	0.2954	25.49	28.82	4.25	4.80

Tabela 4 – Resultados modelo A-2.

Horizonte [h]	MAE		MSE		RMSE		Erro de Energia [Wh/m <sup>2</sup> ]		Erro de Potência [W/m <sup>2</sup> ]	
	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão
1	0.5313	0.3305	0.4859	0.7390	0.5988	0.3568	4.28	5.99	4.28	5.99
2	0.6452	0.3333	0.6841	0.7557	0.7435	0.3623	9.92	11.50	4.96	5.75
3	0.7359	0.3407	0.8657	0.7844	0.8540	0.3692	16.65	16.68	5.55	5.56
4	0.8095	0.3539	1.0340	0.8477	0.9424	0.3818	23.99	21.23	6.00	5.31
5	0.8708	0.3701	1.1902	0.9300	1.0157	0.3982	31.76	25.26	6.35	5.05
6	0.9231	0.3858	1.3338	1.0148	1.0779	0.4145	39.79	29.29	6.63	4.88

Tabela 5 – Resultados modelo B-1.

Horizonte [h]	MAE		MSE		RMSE		Erro de Energia [Wh/m <sup>2</sup> ]		Erro de Potência [W/m <sup>2</sup> ]	
	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão	Média	Desvio Padrão
1	0.4821	0.3258	0.4280	0.7625	0.5485	0.3566	3.71	5.91	3.71	5.91
2	0.5476	0.3171	0.5376	0.7563	0.6402	0.3575	8.08	11.28	4.04	5.64
3	0.5876	0.3043	0.6039	0.7278	0.6949	0.3479	12.45	16.07	4.15	5.36
4	0.6176	0.2862	0.6522	0.6812	0.7371	0.3299	17.08	20.45	4.27	5.11
5	0.6392	0.2685	0.6864	0.6314	0.7672	0.3126	20.99	23.88	4.20	4.78
6	0.6475	0.2659	0.7104	0.6368	0.7825	0.3132	25.48	27.90	4.25	4.65

Tabela 6 – Resultados modelo B-2.

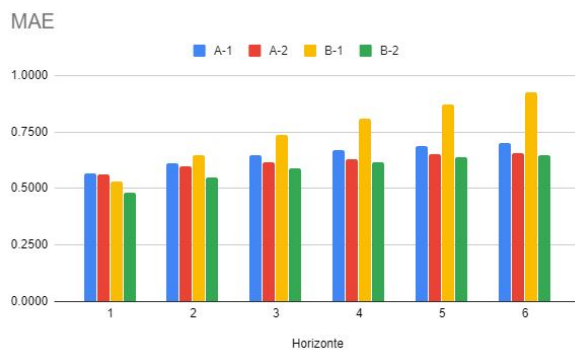


Figura 21 – MAE.

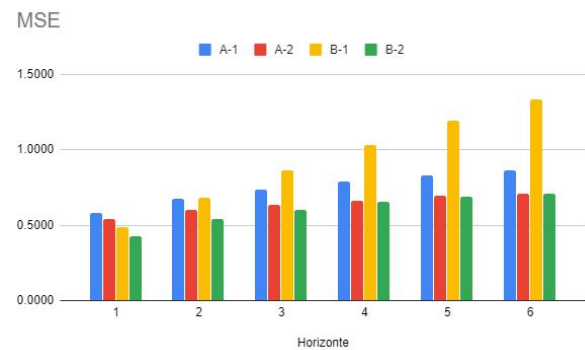


Figura 22 – MSE.

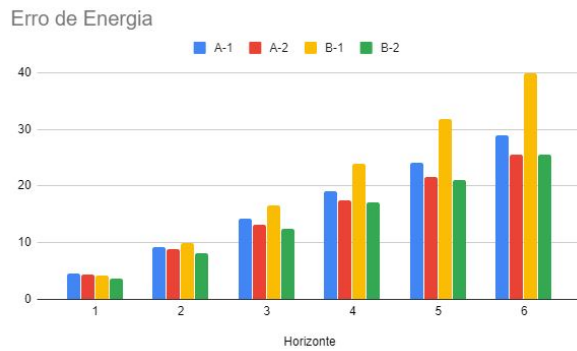


Figura 23 – Erro de energia.

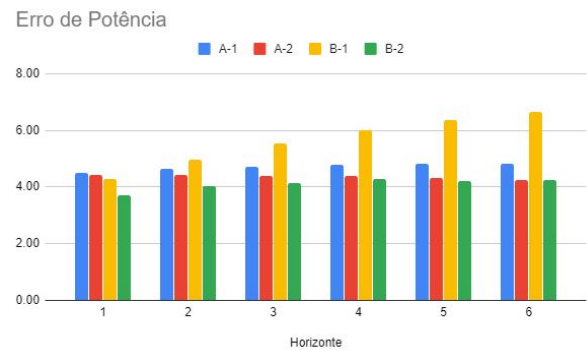


Figura 24 – Erro de potência.

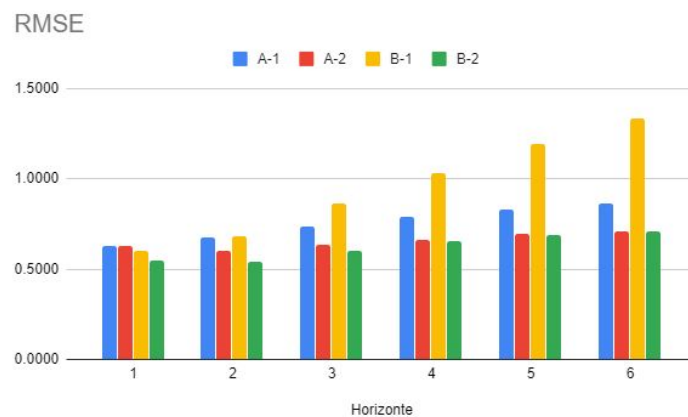


Figura 25 – RMSE.

se mostraram superiores, demonstrando uma melhor performance do método *single-step*. Essa comparação fica mais clara com os gráficos das Figuras 21 a 25.

Analisando algumas previsões individualmente é possível compreender melhor as particularidades de cada modelo. As Figuras 26, 27 e 28 ilustram a comparação dos 4 modelos em amostras individuais. Nos gráficos a seguir são mostrados apenas parte dos pontos anteriores usados na entrada dos modelos, para dar mais clareza à visualização.

Percebe-se que nenhum dos modelos consegue prever com precisão as variações mais rápidas, porém, demonstram um bom potencial para prever tendências.

Outro recurso explorado foi a previsão de intervalos de confiança, que aqui foi definido como 1.96 vezes o desvio padrão previsto pela rede. O intervalo se mostrou adequado num grande número de amostras. Alguns exemplos são ilustrados nas Figuras 29, 30, 31 e 32.



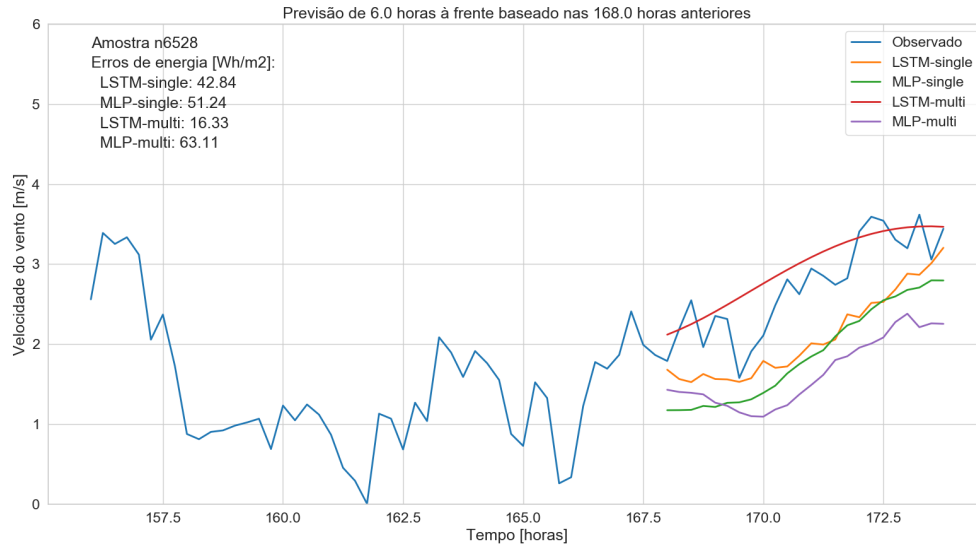


Figura 26 – Comparação de previsões 1.

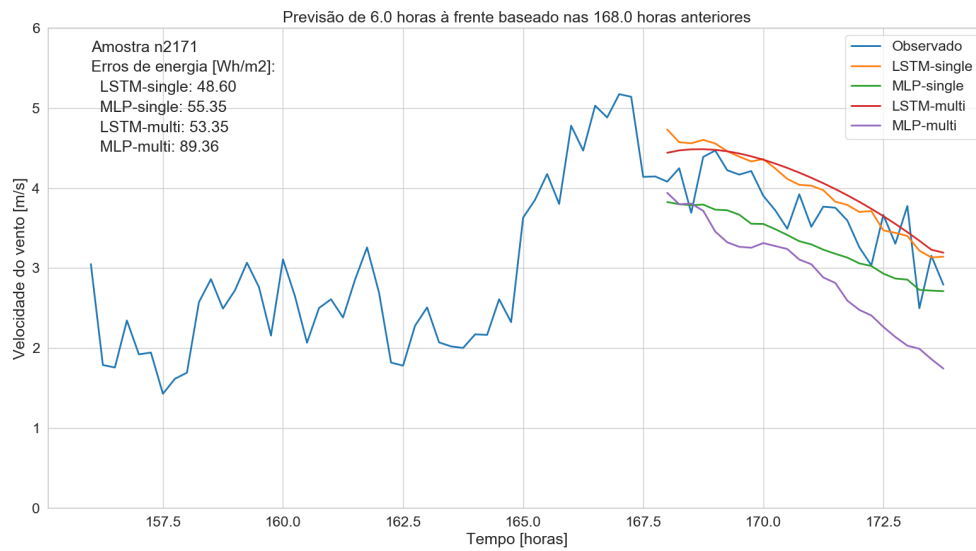


Figura 27 – Comparação de previsões 2.

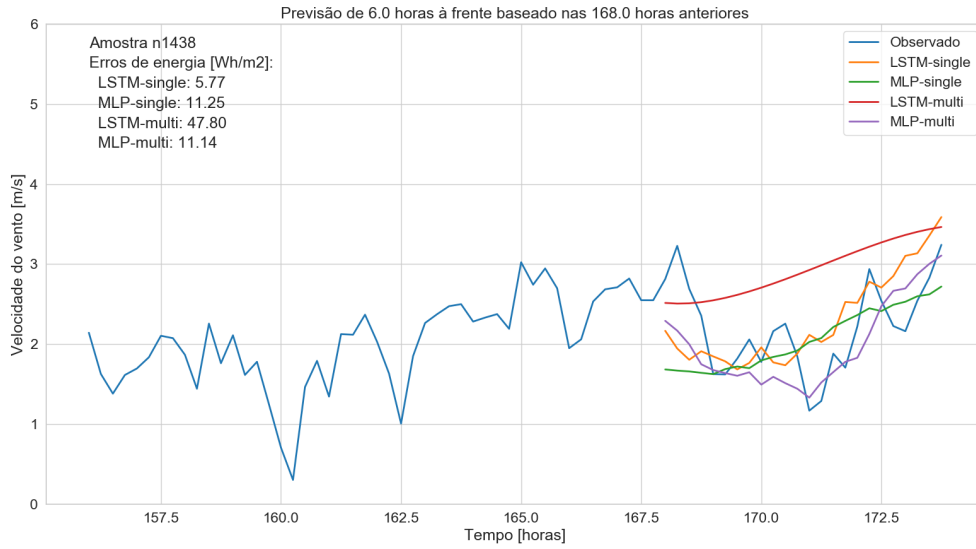


Figura 28 – Comparação de previsões 3.

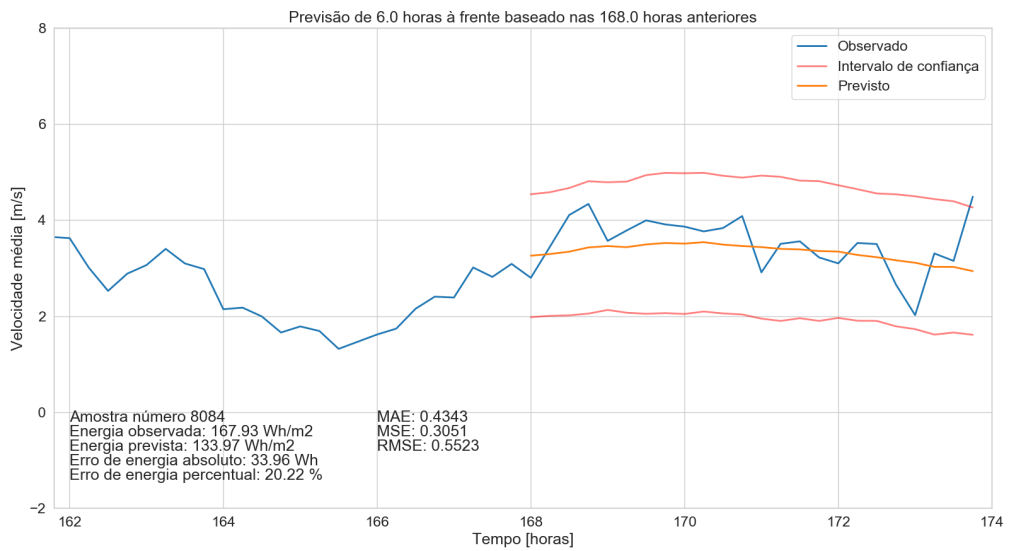


Figura 29 – Previsão modelo MLP-single.

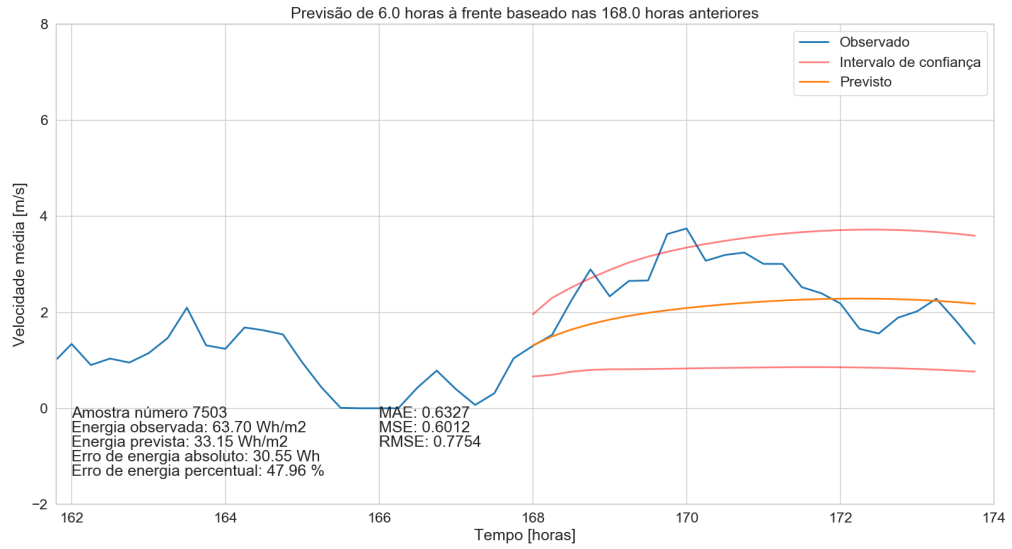


Figura 30 – Previsão modelo MLP-multi.

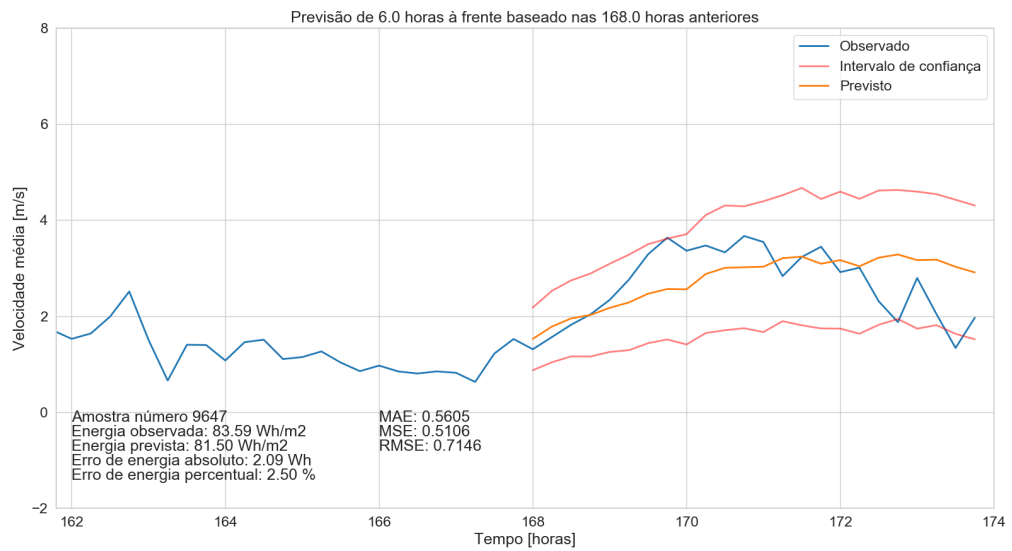


Figura 31 – Previsão modelo LSTM-single.

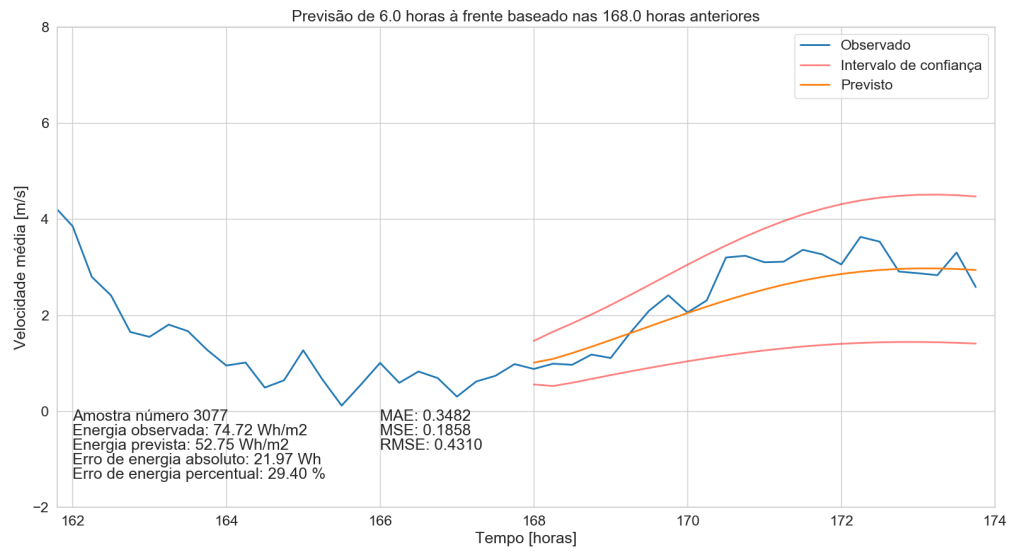


Figura 32 – Previsão modelo LSTM-multi.

## 5 CONCLUSÃO

O uso de técnicas de Inteligência Artificial tem sido cada vez maior em previsões meteorológicas, porém, tais técnicas não foram suficientes no presente estudo. Embora muitas tendências consigam ser previstas com razoável precisão, as variações rápidas da velocidade do vento ainda são um desafio muito grande para o planejamento e operação do Sistema Elétrico, pois são de difícil detecção.

Dentre os modelos propostos, foi possível notar o aumento de precisão com a inserção da camada LSTM em comparação aos modelos baseados somente em MLP. Tal conclusão é corroborada pelas mais atuais publicações no campo de previsão de séries temporais.

Apesar da estratégia *single-step* ter se mostrado numericamente superior à *multi-step*, a segunda se saiu melhor em certos casos e demonstra ter potencial. Tal estratégia tem a vantagem de possuir um modelo único para prever qualquer horizonte de tempo, em contraste à *single-step*, que passa por um treinamento para cada horizonte.

O estudo dos intervalos de confiança é muitas vezes tão importante quanto o valor previsto, visto que as margens de erro são essenciais para tomadas de decisão. Modelos que consigam prever margens para cada previsão individualmente serão de grande contribuição para o progresso da área.

### 5.1 PROPOSTAS PARA TRABALHOS FUTUROS

Uma dificuldade de se trabalhar com RNAs é a quantidade de parâmetros e arquiteturas que podem ser combinados. O estudo da busca automática de parâmetros e arquiteturas também é um campo de pesquisa muito forte atualmente e pode contribuir para o avanço dos modelos propostos.

É recomendado que mais estudos nesse tópico sejam feitos, testando combinações de novas técnicas de inteligência computacional com já consolidados métodos estatísticos, como foi exemplificado na seção 2.3. A redução da dimensionalidade e outros tratamentos a montante das RNAs são exemplos que se mostram eficazes em diversos estudos e devem ser explorados.

Um fator que ainda é deficiente para o avanço de tais estudos é a aquisição de dados meteorológicos, que ainda contém muitos erros que dificultam os estudos. Também é recomendado que se teste modelos de previsão que integrem mais de uma localidade, o que suaviza o impacto dos erros de medição e dá mais recursos para que as redes façam previsões melhores, visto que todo o planeta é conectado climaticamente.

## REFERÊNCIAS

- [1] ANEEL. Atlas de Energia Eólica do Brasil. 2 ed, 2005
- [2] EPE. Plano Decenal de Expansão de Energia 2027, 2018
- [3] GWEC. Global Wind Report 2018, 2019
- [4] SUWANNARAT, A., BAK-JENSEN, B., CHEN, Z., NIELSEN, H. HJERRILD, J., HANSEN, A.D. Power System Operation with Large Scale Wind Power Integration. *IEEE Lausanne Power Tech*, 2007
- [5] JOHNSON, G. L. Wind Energy Systems. Englewood Cliffs: Prentice Hall, 1985.
- [6] MANWELL, J. F.; MCGOWAN, J. G.; ROGERS, A. L. Wind energy explained. New York: Wiley, 2002.
- [7] CAMELO, H. N.; MARIA, P.H.S.; CARVALHO, P.C.M.; PEREIRA, T.B. Métodos de Extrapolação de Velocidade do Vento para Regiões Litorâneas do Nordeste Brasileiro. XVI CBMET, 2010
- [8] EHLERS, R. S. Análise de Séries Temporais. Laboratório de Estatística e Geoinformação. Universidade Federal do Paraná, 2007
- [9] FERNANDES, M. Statistics for Business and Economics, 2009
- [10] ALENCAR, B. D.; AFFONSO, C. M.; OLIVEIRA, R. C. L.; FILHO, J.C.R. Hybrid Approach Combining SARIMA and Neural Networks for Multi-Step Ahead Wind Speed Forecasting in Brazil. *IEEEAccess*, 2018
- [11] SFETSOS, A. A Comprehensive Overview of Short Term Wind Forecasting Models Based on Time Series Analysis. National Centre for Scientific Research Demokritos, Greece, 2011
- [12] MAO, M; LING, J.; CHANG, L.; HATZIARGYRIOU, N. D.; ZHANG, J.; DING, Y. A novel short-term wind speed prediction based on MFEC., *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2016.
- [13] CHU, J.; YUAN, L.; WANG, W.; PAN, L.; WEI, J. Short-Term Wind Speed Prediction based on Deep Learning. *IOP Conf. Series: Earth and Environmental Science 233*, 2019
- [14] KHOSRAVI, A.; MACHADO, L.; NUNES, R. O. Time-series prediction of wind speed using machine learning algorithms: A case study Osorio wind farm, Brazil. Applied Energy - Elsevier, 2018.
- [15] DOUCOURE, B.; AGBOSSOU, K.; CARDENAS, A. Time series prediction using artificial wavelet neural network and multi-resolution analysis: Application to wind speed data. Renewable Energy - Elsevier, 2016.
- [16] HAYKIN, S. Neural Networks and Learning Machines. 3 ed. Canada: Pearson, 2009.
- [17] JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. An Introduction to Statistical Learning: with Applications in R. New York: Springer, 2013.

- [18] HORNIK, K. Approximation capabilities of multilayer feedforward networks. Neural Networks - Elsevier, 1991.
- [19] HOCHREITER, S.; SCHMIDHUBER, J. Long Short-Term Memory. Neural Computation, 1997.
- [20] RUDER, S. An overview of gradient descent optimization algorithms. Insight Centre for Data Analytics, NUI Galway, 2016.
- [21] KINGMA, D. P.; BA, J. L. ADAM: A method for stochastic optimization. ICLR 2015.