

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

TATHIANA MIKAMURA BARCHI

**REDES NEURAIS ARTIFICIAIS E MODELOS DE COMBINAÇÃO PARA
PREVISÃO DE VELOCIDADE DO VENTO**

PONTA GROSSA

2022

TATHIANA MIKAMURA BARCHI

**REDES NEURAIAS ARTIFICIAIS E MODELOS DE COMBINAÇÃO PARA
PREVISÃO DE VELOCIDADE DO VENTO**

**Artificial Neural Networks and Combination Models for Wind Speed
Prediction**

Dissertação apresentada como requisito para obtenção do título de Mestre em Ciência da Computação do Departamento Acadêmico de Informática da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Hugo Valadares Siqueira

Coorientador: Prof^a. Dr^a. Mauren Louise Sguario Coelho de Andrade

PONTA GROSSA

2022



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.



**Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Ponta Grossa**



TATHIANA MIKAMURA BARCHI

REDES NEURAIAS ARTIFICIAIS E MODELOS DE COMBINAÇÃO PARA PREVISÃO DE VELOCIDADE DO VENTO

Trabalho de pesquisa de mestrado apresentado como requisito para obtenção do título de Mestra Em Ciência Da Computação da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Sistemas E Métodos De Computação.

Data de aprovação: 31 de Agosto de 2022

Dr. Hugo Valadares Siqueira, Doutorado - Universidade Tecnológica Federal do Paraná

Dra. Marcella Scoczynski Ribeiro Martins, Doutorado - Universidade Tecnológica Federal do Paraná

Dr. Paulo Salgado Gomes De Mattos Neto, Doutorado - Universidade Federal de Pernambuco (Ufpe)

Dra. Sheila Moraes De Almeida, Doutorado - Universidade Tecnológica Federal do Paraná

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 31/08/2022.

AGRADECIMENTOS

A execução do presente trabalho não seria possível sem o incentivo e apoio daqueles que passaram pela minha vida e se fizeram presentes com palavras de força e coragem. Por isso agradeço:

A Deus;

À minha família de origem;

À minha família em construção;

Ao orientador, aos professores e aos amigos;

Aos parceiros de pesquisa do LICON e do PPGCC;

Gratidão por todos aqueles que não deixaram de acreditar em mim nessa jornada.

*“Todos os modelos estão errados,
mas alguns são úteis.”
Box e Draper (1987)*

RESUMO

A geração de energia por fonte eólica vem recebendo destaque no Brasil devido à necessidade de diversificação da matriz elétrica que depende, em sua maioria, de usinas hidrelétricas. Entretanto, a velocidade do vento é um recurso que apresenta constantes oscilações ao longo do dia, meses e até anos. Neste contexto, a previsão da velocidade do vento é importante pois auxilia, por exemplo, no gerenciamento, despacho e operação do sistema elétrico. Neste sentido, esta pesquisa realizou uma comparação extensiva do desempenho de modelos de previsão, considerando Modelos Lineares (Auto-Regressivo e Auto-Regressivo de Médias Móveis) e Redes Neurais Artificiais (Perceptron de Múltiplas Camadas - MLP, Redes Neurais com Função de Base Radial - RBF, Máquina de Aprendizagem Extremo - ELM e Redes Neurais com Estado de Eco - ESN), *versus* modelos de combinação, *Ensembles* e Sistemas Híbridos de Correção de Erro. A aplicação da metodologia se deu em bases de dados disponibilizadas pelo projeto SONDA (Sistema Nacional de Organização de Dados Ambientais) com as velocidades do vento para as cidades: Brasília, Florianópolis, Natal, Petrolina e São Luís. Foi verificado o erro de cada modelo preditivo através das métricas: Erro Quadrático Médio, Erro Absoluto Médio, Variação Relativa Média e Índice de Concordância. A rede ESN alcançou o 1º lugar geral no ranking de desempenho considerando a sua colocação para cada cidade, sendo ainda o modelo que apresentou o menor erro para Natal e Petrolina. Para os casos específicos de Brasília, Natal e Petrolina, o ARMA se mostrou mais adequado. Todavia, devido ao seu desempenho pouco acurado nos outros casos, sua posição no ranking geral foi degradada. Os *Ensembles* ocuparam o segundo e terceiro lugares com o *Ensemble* de Mediana (Modelos Únicos sem RBF) e *Ensemble* de Mediana (redes neurais sem RBF), respectivamente. No ranking final, os Sistemas Híbridos foram destaque negativo.

Palavras-chaves: velocidade do vento; previsão; modelos lineares (estatística); redes neurais (computação); ensembles; sistema híbrido de correção de erro.

ABSTRACT

Wind power generation has been highlighted in Brazil due to the need to diversify the electrical matrix, which mainly depends on hydroelectric plants. However, wind speed is a resource that presents constant fluctuations throughout the day, months, and even years. In this context, wind speed prediction is essential as it helps in the electrical system's management, dispatch, and operation. In this sense, this research carried out an extensive comparison of traditional forecasting models, being the Linear Models (Auto-Regressive and Auto-Regressive and Moving Average models) and Artificial Neural Networks (Multilayer Perceptron - MLP, Radial Basis Function Networks - RBF, Extreme Learning Machine - ELM and Echo State Neural Networks - ESN), versus combination models: Ensembles and Hybrid Error Correction Systems. The application of the methodology considered databases provided by the SONDA project (National Environmental Data Organization System) with wind speeds for the cities: Brasília, Florianópolis, Natal, Petrolina, and São Luís. The error of each predictive model was verified through the metrics Mean Absolute Error, Mean Squared Error, Average Relative Variance, and Index of Agreement. The ESN network reached 1st place in the performance ranking considering its placement for each city, being the model that presented the smallest error for Natal and Petrolina. The ARMA proved to be more suitable for specific cases of Brasília, Natal, and Petrolina. However, its classification in the general ranking was degraded due to its inaccurate performance in the other cases. The Ensemble occupied second and third places with Median Ensemble (Single Models without RBF) and Median Ensemble (RNAs without RBF), respectively. In the final ranking, Hybrid Systems were a negative highlight.

Key-words: wind speed; forecasting; linear models (statistics); neural networks (computing); ensembles; hybrid system based on error correction.

LISTA DE FIGURAS

Figura 1 – Exemplos de séries não estacionárias	15
Figura 2 – Exemplo de uma série temporal decomposta	18
Figura 3 – Neurônio artificial	23
Figura 4 – MLP	25
Figura 5 – RBF	26
Figura 6 – ELM	28
Figura 7 – ESN	28
Figura 8 – Viés e variância	30
Figura 9 – Viés, variância e o cenário de previsão ideal	31
Figura 10 – Bagging	32
Figura 11 – Boosting	32
Figura 12 – Fluxograma do processo de treinamento do Sistema Híbrido de Correção de Erro	35
Figura 13 – Fluxograma do processo de teste do Sistema Híbrido de Correção de Erro	36
Figura 14 – Reestruturação de uma série temporal para aprendizagem supervisionada	37
Figura 15 – Mapa eólico do Brasil	39
Figura 16 – Pré e Pós-processamento dos dados	41

LISTA DE GRÁFICOS

Gráfico 1 – Brasília - Real x Melhor Previsão	44
Gráfico 2 – Brasília - Real x Melhor Previsão (primeiras 100 amostras)	45
Gráfico 3 – Brasília - Boxplot dos erros de previsão	45
Gráfico 4 – Florianópolis - Real x Melhor Previsão	47
Gráfico 5 – Florianópolis - Real x Melhor Previsão (primeiras 100 amostras)	48
Gráfico 6 – Florianópolis - Boxplot dos erros de previsão	48
Gráfico 7 – Natal - Real x Melhor Previsão	50
Gráfico 8 – Natal - Real x Melhor Previsão (primeiras 100 amostras)	50
Gráfico 9 – Natal - Boxplot dos erros de previsão	51
Gráfico 10 – Petrolina - Real x Melhor Previsão	53
Gráfico 11 – Petrolina - Real x Melhor Previsão (primeiras 100 amostras)	53
Gráfico 12 – Petrolina - Boxplot dos erros de previsão	54
Gráfico 13 – São Luís - Real x Melhor Previsão	56
Gráfico 14 – São Luís - Real x Melhor Previsão (primeiras 100 amostras)	56
Gráfico 15 – São Luís - Boxplot dos erros de previsão	57
Gráfico 16 – Boxplot das classificações dos modelos sobre os 5 estudos de casos e as 4 métricas propostas	58
Gráfico 17 – Visualização da quantidade de neurônios utilizados	63
Gráfico 18 – Uso de cada atraso pelas RNAs	65
Gráfico 19 – Uso de cada atraso pelos Sistemas Híbridos com AR	65
Gráfico 20 – Uso de cada atraso pelos Sistemas Híbridos com ARMA	66

LISTA DE TABELAS

Tabela 1 – Brasília - Análise descritiva dos dados	43
Tabela 2 – Brasília - Erros de previsão	43
Tabela 3 – Florianópolis - Análise descritiva dos dados	46
Tabela 4 – Florianópolis - Erros de previsão	46
Tabela 5 – Natal - Análise descritiva dos dados	49
Tabela 6 – Natal - Erros de previsão	49
Tabela 7 – Petrolina - Análise descritiva dos dados	52
Tabela 8 – Petrolina - Erros de previsão	52
Tabela 9 – São Luís - Análise descritiva dos dados	55
Tabela 10 – São Luís - Erros de previsão	55
Tabela 11 – Ranking de performance	61
Tabela 12 – Quantidade de neurônios por modelo e base de dados	62
Tabela 13 – Atrasos significativos por base de dados e tipo de modelo	64

SUMÁRIO

1	INTRODUÇÃO	10
1.1	Objetivo	12
1.2	Organização do Texto	13
2	SÉRIES TEMPORAIS	14
2.1	Processos estocásticos estacionários	14
2.2	Ferramentas para análise de séries temporais	15
2.3	Ferramentas para tratamento de séries não-estacionárias	17
3	MODELOS DE PREVISÃO	20
3.1	Modelos Lineares de Previsão	20
3.1.1	Modelo Auto-Regressivo (AR)	21
3.1.2	Modelo de Médias Móveis (MA)	22
3.1.3	Modelo Auto-Regressivo de Médias Móveis (ARMA)	22
3.2	Redes Neurais Artificiais	23
3.2.1	Perceptron De Múltiplas Camadas	24
3.2.2	Redes Neurais com Função De Base Radial	25
3.2.3	Máquina De Aprendizado Extremo	27
3.2.4	Redes Neurais Com Estado De Eco	27
3.3	Metodologia <i>Ensemble</i>	30
3.3.1	Amostragem/Seleção de dados	31
3.3.2	<i>Ensemble</i> Homogêneo e Heterogêneo	33
3.3.3	Combinadores	33
3.4	Sistemas Híbridos de correção de erro	34
4	PREVISÃO DE SÉRIES TEMPORAIS COMO UM PROBLEMA DE APRENDIZAGEM SUPERVISIONADA	37
4.1	Seleção de variáveis	37
5	ESTUDOS DE CASOS	39
5.1	Modelos de Previsão Aplicados	41
5.2	Brasília	43
5.3	Florianópolis	46
5.4	Natal	49
5.5	Petrolina	52
5.6	São Luís	55
5.7	Análise Dos Resultados	58
5.8	Quantidade de neurônios	62
5.9	Atrasos	64
6	CONCLUSÃO	67
6.1	Trabalhos futuros	68
	REFERÊNCIAS	70
	ANEXO A PRIMEIRA PÁGINA DO ARTIGO SUBMETIDO	75

1 INTRODUÇÃO

A conversão de energia cinética dos ventos para energia mecânica vem sendo utilizada há mais de 3000 anos. Pode-se considerar como uma das primeiras aplicações os moinhos de grãos. Tais moinhos influenciaram a economia agrícola por séculos, de modo que algumas leis foram criadas para a concessão dos mesmos, como a lei do direito ao vento que proibia a plantação de árvores nos seus arredores (MARTINS; GUARNIERI; PEREIRA, 2008; FLEMING; PROBERT, 1984).

Já o uso do vento para geração de energia elétrica data do final do século XIX, quando a primeira turbina eólica foi usada para alimentar um grande número de lâmpadas incandescentes. Entretanto, tornaram-se populares apenas quando os preços do petróleo aumentaram durante a Segunda Guerra Mundial, levando ao desenvolvimento dos aerogeradores de médio e grande porte, por conta dos esforços em economizar combustíveis fósseis (JONCAS, 2010).

Mais adiante, na década de 1970, durante a crise internacional do petróleo, os Estados Unidos e países da Europa começaram a se interessar e investir intensamente em tecnologias para geração de energia eólica, viabilizando o seu desenvolvimento. A partir da década de 1990 o vento de fato começou a ser utilizado como fonte de energia em escala comercial (ALENCAR, 2018) e em poucos anos passou a ser reconhecido como uma das fontes que mais tem se expandido no mundo (COUNCIL, 2021).

Entre 2019 e 2020 houve um crescimento de 53% no mundo, o recorde da indústria de geração eólica, mesmo havendo interrupções na cadeia de fornecimento global. Tal crescimento agregou 93 GW na capacidade de produção, sendo que destes, 86,9 GW foram instalações terrestres e o restante no mar. O Brasil, em 2020, foi o terceiro país que mais investiu, atrás apenas da China e dos Estados Unidos (COUNCIL, 2021).

Este crescimento fez o Brasil totalizar 212,22 GW em capacidade na geração de energia eólica, a qual está dividida entre as 687 matrizes em operação, com maior concentração no Nordeste (85%) (COUNCIL, 2021).

Um dos principais motivos para tal concentração é o potencial eólico alto durante o período de menor disponibilidade hídrica, possuindo ventos mais fortes entre junho e novembro (SILVA *et al.*, 2017).

Além dos incentivos de não poluição e degradação do meio ambiente, o Brasil também possui vários aspectos necessários para a produção de energia por meio do vento como: altas velocidades médias, pouca variação de direção e baixa frequência de rajada e turbulência. Tais características são facilmente encontradas no Nordeste (SILVA *et al.*, 2017).

Apesar dos grandes investimentos, a principal fonte de energia no Brasil ainda é a usina hidrelétrica, que depende do nível dos reservatórios, fator associado ao volume de precipitações captados.

Outra fonte geradora de energia presente no Brasil são as usinas termelétricas, alimentadas por fontes não renováveis, em sua maioria por combustíveis fósseis, nas quais a queima

gera gases poluentes como o dióxido de carbono (CO_2), considerado principal causador do aquecimento global. Levando-se em consideração o cenário atual do aquecimento global e das mudanças climáticas iminentes, é importante que indústrias de energia encontrem recursos isentos de poluição por dióxido de carbono. Note que os combustíveis fósseis são responsáveis pela produção de 65% da eletricidade global (LETCHER, 2020). Neste sentido, a energia eólica é reconhecida por ser uma importante fonte de energia renovável livre de emissão de dióxido de carbono e gases poluentes durante sua geração.

Outra consideração a ser feita sobre as usinas termelétricas é em relação ao elevado consumo de água, que é utilizada tanto para produção de calor quanto para sistemas de resfriamento (LETCHER, 2020).

Segundo Dupont, Koppelaar e Jeanmart (2018) é esperado para os próximos anos, mais especificamente por volta de 2050, que muitos países considerem o vento como uma importante fonte de energia, pois esta, além de ser uma opção mais sustentável, também é o recurso energético mais econômico. Um dos fatores que a torna mais barata é a redução do custo de transporte da eletricidade para estações de energia distantes, pois o deslocamento em grandes distâncias ocorre em corrente alternada, o que é caro em decorrência do custo de postes e cabos e pelas perdas devido ao efeito Joule. Isso ocorre com parte da energia produzida na Usina Hidrelétrica de Itaipu que é dirigida para o Sudeste, em seguida para o Norte e por fim ao Nordeste, já que não há linha entre Sudeste e Nordeste. Outro fator que influencia no valor é o decréscimo considerável do custo de turbinas com o passar das décadas (LETCHER, 2017).

Entretanto, simultaneamente a expansão das matrizes energéticas, surgem novos desafios tecnológicos e científicos, sendo um dos maiores a previsão do vento, pois a geração eólica depende da sua velocidade, variável esta que não pode ser controlada e tem grande variabilidade (SILVA *et al.*, 2017).

De modo geral, prever a velocidade do vento de maneira realista desempenhará um papel fundamental no estabelecimento de qual, ou quais, metodologias deverão ser utilizadas para ampliar e gerenciar a produção (DUPONT; KOPPELAAR; JEANMART, 2018). Os modelos lineares, introduzidos por Box & Jenkins (BOX *et al.*, 2015), estão presentes nos trabalhos de Milligan, Schwartz e Wan (2003) para previsão de séries de ventos em fazendas dos Estados Unidos e de REN, HUANG e LI (2019) para um comparativo de desempenho. Tais modelos assumem que existe uma relação linear entre as variáveis.

Diferente das abordagens lineares, as Redes Neurais Artificiais (RNAs) constroem um mapeamento não-linear entre as amostras de entrada e saída. Várias arquiteturas já foram utilizadas para previsão de séries de ventos, por exemplo: i) Redes de Elman (XU; MAO, 2016); ii) Perceptron De Múltiplas Camadas (MLP) treinada pelo algoritmo de retropropagação do erro, Redes Neurais com Função de Base Radial (RBF), *Adaptive Neuro-Fuzzy Inference System* (ANFIS) (NOOROLLAHI; JOKAR; KALHOR, 2016), dentre outras.

De acordo com Sharkey (2012) e Júnior (2009) os melhores resultados de previsão são obtidos por meio da combinação de diferentes modelos resultando em um modelo de predi-

ção mais robusto. Os autores destacam que tal combinação leva a uma diminuição do erro de previsão.

A metodologia *Ensemble* é uma das técnicas reconhecidas na literatura que faz junção de informações obtidas à partir de diversos modelos. Nesta é possível ter tanto um conjunto homogêneo quanto um conjunto heterogêneo de modelos para previsão. Além disso, há diversas formas de combinar e organizá-los (KAZIENKO; LUGHOFER; TRAWINSKI, 2015).

É possível encontrar na literatura diversos trabalhos de previsão de séries de ventos aplicando tal metodologia. Em (ZHANG *et al.*, 2016) é construído um modelo utilizando uma MLP e *Numerical Weather Prediction* combinados por voto majoritário. Já em Shao *et al.* (2016) utiliza-se *wavelet transformation* e *AdaBoosting*.

Outra técnica de combinação são os Sistemas Híbridos de Correção de Erro (SHCE) propostos por Zhang (2003), que baseiam-se na hipótese de que os problemas do mundo real representáveis através de séries temporais não são puramente lineares ou não-lineares. Dessa forma, a proposta é um modelo capaz de prever as componentes lineares e não-lineares separadamente e depois uni-las.

O SHCE é apontado como o modelo de maior acurácia nos resultados de predição em séries de ventos, sendo comumente encontrado nos estudos mais recentes sobre previsão de séries temporais. Em Neto *et al.* (2022) é proposto o uso do SHCE que combinam modelos de aprendizado de máquina usando previsão residual como alternativa para melhorar o desempenho da previsão da temperatura da superfície do mar. O trabalho de Saranj e Zolfaghari (2022) apresenta uma abordagem híbrida para prever o consumo de eletricidade em duas etapas: a primeira consiste na decomposição dos dados e em seguida a modelagem dos componentes utilizando *Adaptive Wavelet Transform* (AWT), a rede neural *long short-term memory* (LSTM), modelo Auto-Regressivo de Médias Móveis com variáveis Exógenas (ARIMAX) e modelos Autoregressivos de Heteroscedasticidade Condicional Exponencial (GARCH). O método proposto por Kim e Baek (2022) é aplicável a séries temporais univariadas usando Transformada de Wavelet Discreta de Sobreposição Máxima (MODWT) e bootstrap. Além deste, outros trabalhos também foram promissores na aplicação dos SHCE como em Haider *et al.* (2022), Neto *et al.* (2021), Oliveira, Silva e Neto (2021) e Niu, Wang e Zhang (2022).

1.1 Objetivo

Dado o contexto anterior, este trabalho realizou uma investigação comparativa considerando os modelos tradicionais *versus* novas propostas para previsão de séries de velocidade do vento. Para cumprir tal tarefa, os seguintes objetivos específicos foram estabelecidos:

- Aplicação de métodos estatísticos para análise e pré-processamento dos dados;
- Uso de técnicas para seleção de variáveis para determinação das entradas para a previsão;

- Utilização dos modelos lineares Auto-Regressivo (AR) e Auto-Regressivo de Médias Móveis (ARMA);
- Realização da previsão com as RNAs *feedforward*: Perceptron de Múltiplas Camadas, Redes Neurais de Função de Base Radial e Máquina de Aprendizado Extremo;
- Aplicação da Rede de Estado de Eco;
- Previsão utilizando metodologias *Ensemble* com diferentes combinadores;
- Execução de Sistemas Híbridos de Correção de Erro.

1.2 Organização do Texto

Este trabalho está organizado em seis capítulos, no qual o primeiro é a Introdução e os demais são descritos a seguir:

- Séries Temporais: capítulo que introduz os conceitos sobre séries temporais que são necessários para compreender a aplicação dos métodos de previsão;
- Modelos de Previsão: apresenta (i) os Modelos Lineares introduzidos por Box *et al.* (2015), que são amplamente utilizados para a tarefa de predição de séries temporais; (ii) as Redes Neurais Artificiais (RNAs) no qual são discutidos os conceitos e as topologias aplicadas neste trabalho; (iii) Metodologia *Ensemble*, sua composição e possíveis configurações para predição; e (iv) Sistemas Híbridos de Correção de Erro;
- Previsão de Séries Temporais como um Problema de Aprendizagem Supervisionada: fundamenta como o problema de previsão de séries temporais pode ser modelado como um problema de aprendizagem supervisionado;
- Estudo de Caso: aplica os modelos propostos para dados de 5 cidades diferentes apresentando um comparativo de desempenho entre os mesmos;
- Conclusão: retoma os pontos apresentados no decorrer do documento e propõe tarefas para trabalhos futuros.

2 SÉRIES TEMPORAIS

Diversas áreas de pesquisa trabalham com conjuntos de dados observados e ordenados através do tempo, o que caracteriza uma série temporal. Pode-se citar áreas como: economia, negócios, engenharia e outras, nas quais a tarefa consiste em analisar e principalmente descrever o comportamento de tais observações (BOX *et al.*, 2015).

Uma série temporal, como dito, é uma sequência de observações feitas ao longo do tempo, na qual as observações vizinhas geralmente são dependentes. Essas observações podem ser classificadas entre discretas ou contínuas. No caso discreto, as séries possuem observações feitas em intervalos de tempo fixos. Por outro lado, séries contínuas são aquelas obtidas à partir de observações gravadas continuamente, ao longo de algum intervalo de tempo não determinado (BROCKWELL; DAVIS; CALDER, 2002). Neste trabalho, serão consideradas apenas séries discretas para a tarefa de previsão, o caso mais usual da literatura.

Formalmente, é possível descrever uma série temporal como uma realização de um processo estocástico (SILVA *et al.*, 2017). Um processo estocástico é uma função aleatória que evoluiu no tempo (e/ou no espaço), e é definida sob um espaço de probabilidades.

Assumindo que $T \subset \mathbb{Z}_+$, no qual \mathbb{Z}_+ é o conjunto dos números inteiros positivos, as observações que compõem uma série temporal podem ser explicadas por uma variável aleatória $\{Z_t\}, t \in T$.

(MORETTIN; TOLOI, 2006) Seja T um conjunto arbitrário. Um processo estocástico é uma família $Z = \{Z_t, t \in T\}$ de forma que para cada $t \in T$, Z_t é uma variável aleatória.

2.1 Processos estocásticos estacionários

Ao trabalhar com modelos que buscam descrever séries temporais, é necessário fazer previamente uma análise sobre determinadas classes de processos estocásticos (MORETTIN; TOLOI, 2006). Um conceito relevante neste campo é o de processo estacionário.

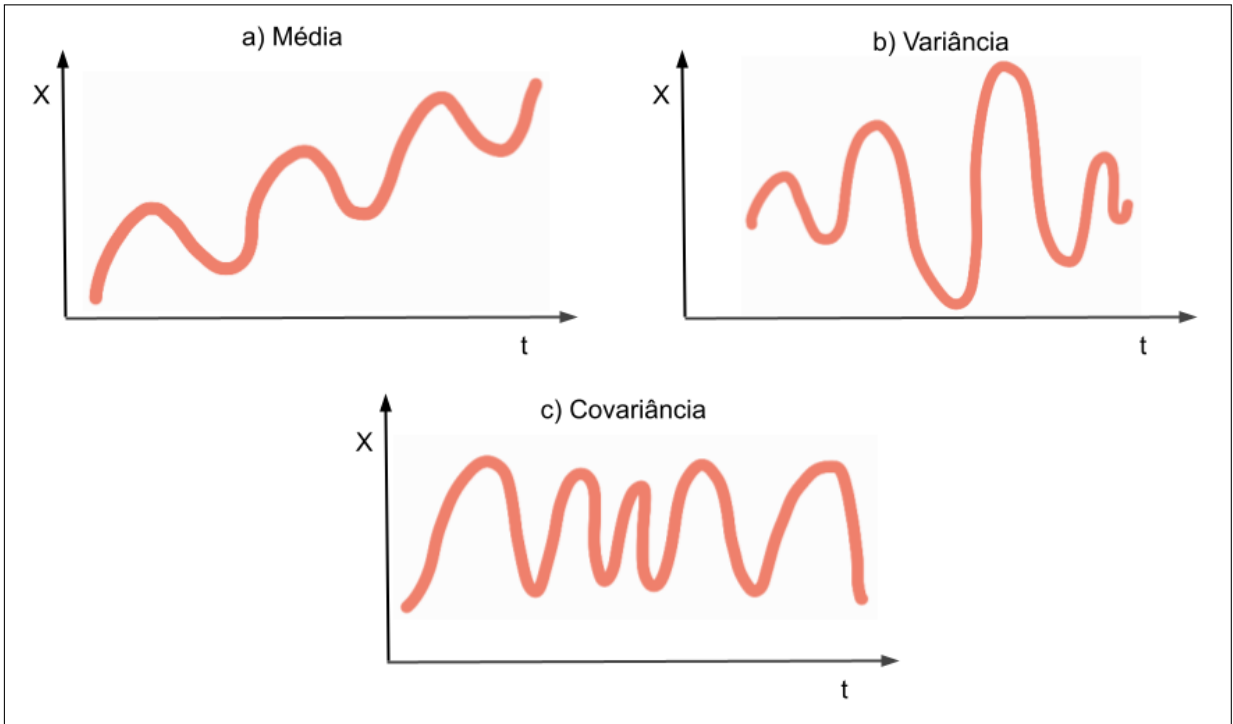
Um processo é dito fracamente estacionário se, e somente se:

- A média é constante: $E[Z_t] = \mu_t = \mu$;
- A variância constante: $Var[Z_t] = \sigma_t^2 = \sigma^2$;
- A covariância $Cov[Z_t, Z_{t+k}] = \gamma(k)$ é uma função exclusiva de k .

Processos fracamente estacionários, também são chamados de estacionários no sentido amplo por definição mantém suas propriedades estatísticas ao longo do tempo.

A Figura 1 exemplifica séries que não são estacionárias. O primeiro caso, a), mostra uma média que não é constante no decorrer do tempo; em b) a dispersão dos valores em relação à média muda no decorrer do tempo; já em c) é difícil identificar um padrão dependente apenas da diferença de tempo que seja satisfeito por quaisquer duas variáveis.

Figura 1 – Exemplos de séries não estacionárias



Fonte: Autoria própria (2022)

2.2 Ferramentas para análise de séries temporais

Para a tarefa de análise de séries temporais são utilizadas ferramentas estatísticas que ajudam a identificar o processo estocástico em questão, mais especificamente a medir o grau de associação entre as observações e com isso identificar a qual classe tal processo pertence (SILVA *et al.*, 2017).

Nessa seção serão apresentados os seguintes conceitos: média, desvio padrão, autocovariância, variância e autocorrelação.

A média (ou esperança matemática - $E[\cdot]$) para uma série temporal é dada conforme a Equação 1.

$$\mu = E[Z_t] = \frac{\sum_{t=1}^N Z_t}{N}, \quad (1)$$

sendo $E[Z_t]$ o valor esperado para Z_t e N o total de observações disponíveis.

O desvio padrão (σ) é responsável por medir a variação, ou dispersão, de um conjunto de dados. Dessa forma, σ indica quão homogênea é uma determinada base de dados, sendo expressa pela Equação 2.

$$\sigma = \sqrt{\frac{\sum_{t=1}^N (Z_t - \mu)^2}{N}}, \quad (2)$$

sendo μ a média.

Também é possível fazer uso da esperança matemática, $E[\cdot]$, para medir a dependência até segunda ordem entre duas observações, t e $t + k$. Com isso mede-se a auto-covariância (γ) de passo k , Equação 3.

$$\gamma(t, t + k) = cov[Z_{t+k}, Z_t] = E[(Z_t - \mu)(Z_{t+k} - \mu)], \quad (3)$$

de modo que o valor k também pode ser chamado de defasagem ou *lag*.

Em outras palavras, a auto-covariância identifica se há um padrão no comportamento das variáveis - variam na mesma direção ou direção oposta. Exemplo: à medida que x_t aumenta, x_{t+k} cai.

Supondo que Z é uma série estacionária, tem-se que a distribuição de probabilidade é a mesma para todo t e $t + k$, sendo k um intervalo. Com isso, a função de auto-covariância é dada em função de k , Equação 4.

$$\gamma(k) = E[(Z_t - \mu)(Z_{t+k} - \mu)] \quad (4)$$

Para k igual a zero, tem-se a variância, medida responsável por definir a amplitude de oscilação da série, conforme Equação 5.

$$\gamma(0) = Var[Z_t] = \sigma^2 \quad (5)$$

A correlação também tem como objetivo analisar a conexão até segunda ordem, com o diferencial de ser uma medida normalizada no qual seus valores variam entre -1 e 1.

O coeficiente de autocorrelação, Equação 6, é dado pela auto-covariância em função de k dividida pela variância.

$$\rho_k = \frac{\gamma_k}{\sigma^2} \quad (6)$$

Seguindo o pressuposto de que Z é estacionária, é importante avaliar a autocorrelação e a auto-covariância para diversos valores para k . Na Equação 7 c_k representa o cálculo de $\gamma(k)$ até o passo K (BOX *et al.*, 2015).

$$c_k = \frac{1}{N - k} \sum_{t=1}^{N-k} (x_t - \hat{\mu})(x_{t+k} - \hat{\mu}), k = 0, 1, \dots, K, \quad (7)$$

no qual N é a quantidade de amostras consideradas e $\hat{\mu}$ a média estimada pela Equação 8

$$\hat{\mu} = \frac{1}{N} \sum_{t=1}^N Z_t \quad (8)$$

Com isso, o coeficiente de autocorrelação pode ser calculado para cada $k = 0, 1, \dots, K$ a depender da quantidade de amostras consideradas em c_k , como mostra a Equação 9. Para $N \cong 100$ recomenda-se $K < \frac{N}{4}$.

$$\rho = \frac{c_k}{c_0} \quad (9)$$

A função de autocorrelação parcial (PACF) tem o papel de estimar a relação entre duas observações nos instantes, exemplo t e $t-4$, de modo que as observações entre estas $t-1, t-2$ e $t-3$ - não sejam contabilizadas. Em outras palavras, a autocorrelação parcial mostra a correlação entre Z_t e Z_{t+k} após a remoção das dependências das observações intermediárias $Z_{t+1}, Z_{t+2}, \dots, Z_{t+k-1}$. É importante ressaltar que tanto a autocorrelação quanto autocorrelação parcial avaliam a dependência linear entre variáveis.

Tais coeficientes são interessantes para estabelecer quais valores de k auxiliarão na tarefa de previsão. Entretanto, avaliar todos os valores de k possíveis e suas combinações, é uma tarefa demorada que pode ser inviável (BELOTTI *et al.*, 2019). Para isso existem algumas técnicas de seleção de variáveis, que serão apresentadas na Seção 4.1.

Vale ressaltar que a auto-covariância, autocorrelação e autocorrelação parcial são ferramentas essenciais para detectar dependência entre dois valores da série e também verificar a condição de estacionaridade, definida na Seção 2.1. Por exemplo, se o teste de autocorrelação apresentar um decaimento vagaroso e autocorrelação parcial com valor significativo no primeiro *lag*, então tem-se um indicativo de não-estacionaridade. Além disso, a correlação entre valores próximos indica dependência de curto-prazo e, para oscilações periódicas, é possível encontrar variações sazonais (SIQUEIRA *et al.*, 2009).

Além destas, também é possível aplicar testes como o *Augmented Dickey-Fuller - ADF* (DICKKEY; FULLER, 1979) que são capazes de apontar a probabilidade de uma série não ser estacionária e com isso a necessidade de tratamento.

2.3 Ferramentas para tratamento de séries não-estacionárias

Uma das suposições mais comuns a respeito das séries é que estas sejam estacionárias. Apesar de tal suposição, a maioria das séries encontradas no mundo real apresenta alguma forma de não estacionaridade (MORETTIN; TOLOI, 2006).

Dado o fato de que a tarefa de previsão em muitos casos supõe que o conjunto de observações seja estacionário, é necessário aplicar técnicas que a tornem estacionária pelo menos no sentido fraco. Nestes casos, a série pode ser decomposta em componentes não-observáveis que estão vinculados ao comportamento não-estacionário. A decomposição em questão pode ocorrer através de um modelo aditivo (Equação 10) ou multiplicativo (Equação 11) (MORETTIN; TOLOI, 2006).

$$Z_t = T_t + C_t + S_t + \varepsilon_t, \quad (10)$$

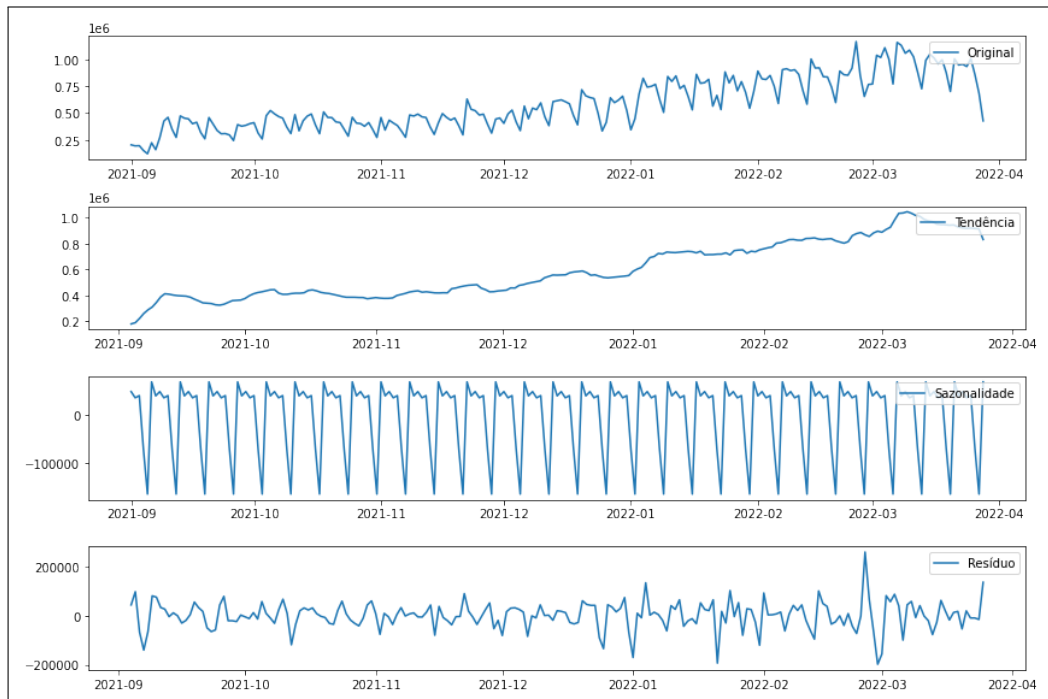
$$Z_t = T_t \times C_t \times S_t + \varepsilon_t, \quad (11)$$

nas quais para uma dada série temporal Z é possível encontrar as componentes:

- Tendência (T): componente de longo prazo, demonstrando comportamento crescente ou decrescente da série;
- Ciclo (C): componente de médio prazo relacionada a mudanças periódicas superiores a um ano;
- Sazonalidade (S): componente de curto prazo associada a variações relacionadas a épocas do ano (meses, estações, ...), ou seja, inferiores a um ano;
- Resíduo (ε): variável aleatória que não pode ser descrita de forma explícita.

A Figura 2 mostra como exemplo de uma dada série Z decomposta em Tendência, Sazonalidade e Resíduo.

Figura 2 – Exemplo de uma série temporal decomposta



Fonte: Autoria própria (2022)

O modelo multiplicativo é mais recomendado para séries com variâncias crescentes, enquanto o aditivo é melhor associado a séries com variâncias estabilizadas no tempo (séries homocedásticas) (ROSSI; NEVES, 2000).

É importante salientar que nem toda série temporal apresentará os quatro componentes. Alguns autores não consideram o Ciclo como uma componente dos modelos aditivos ou multiplicativos, pois este depende de uma análise cuja série temporal deve abranger milhares de amostras para que o fenômeno possa, ou não, ser identificado (REIS, 2020).

Uma vez que as componentes são identificadas, é possível aplicar métodos que tratem a série tornando-a estacionária. A seguir são apresentados alguns métodos para tratamento de Tendência e Sazonalidade em séries aditivas.

Para remoção de tendência é comum realizar sucessivas diferenciações. Geralmente duas diferenciações são suficientes para este fim (MORETTIN; TOLOI, 2006). A Equação 12 apresenta a primeira diferença de Z_t .

$$\Delta Z_t = Z_t - Z_{t-1} \quad (12)$$

Para a retirada da componente sazonal é possível utilizar dessazonalização, na qual o valor padronizado é dado pela Equação 13.

$$Z_{t,s} = \frac{Z_{t,s} - \mu_s}{\sigma_s}, \quad (13)$$

em que μ_s e σ_s são, respectivamente, a média e o desvio padrão da sazonalidade s .

Por exemplo, se uma dada série Z com valores diários apresenta um padrão que se repete à cada mês, então s representará os meses e assim cada mês terá uma média e um desvio padrão. Com isso, aplicar a Equação 13 em Z implicaria em pegar cada amostra Z_t , subtrair a média do seu respectivo mês e por fim dividir também pelo desvio padrão de tal mês. A partir disso uma nova série é gerada com média zero e variância unitária, que pode ser dada como entrada para modelos de previsão (SIQUEIRA *et al.*, 2009).

É importante salientar que estes métodos devem ser inversamente reaplicados na série após os resultados de modelagem e previsão, ou seja, realizar a recomposição da série para cálculo das métricas em escala real. Se esta for aditiva, deve-se somar componentes que antes foram retirados, como tendência e sazonalidade; da mesma maneira, caso a série seja multiplicativa é feita a multiplicação do valor previsto (REIS, 2020).

3 MODELOS DE PREVISÃO

O presente Capítulo apresenta os modelos e técnicas utilizados para previsão de séries de velocidade do vento. Serão apresentados alguns já bastante sedimentados na literatura, como os modelos lineares (Seção 3.1) e Redes Neurais Artificiais (Seção 3.2), e outros que vêm apresentando importantes resultados, como os *Ensembles* (Seção 3.3) e Sistemas Híbridos (Seção 3.4).

3.1 Modelos Lineares de Previsão

Os modelos apresentados neste capítulo foram introduzidos por Box *et al.* (2015) e são caracterizados por criarem métodos cuja relação entre as variáveis de uma série temporal é linear. Assim, seguem a ideia de que amostras sucessivas possuem um certo nível de dependência, requisito indispensável para tarefas desse tipo. Dentre estes serão apresentados o Auto-Regressivo (AR), Médias Móveis (MA) e Auto-Regressivo de Médias Móveis (ARMA).

Para a construção de tais modelos, Box *et al.* (2015) sugerem um ciclo no qual assume-se que o processo de geração das séries temporais pode ser aproximado desde que a série seja estacionária. Tal processo pode ser organizado em quatro etapas iterativas (SILVA *et al.*, 2017):

1. Especificação: etapa em que uma classe geral do modelo é considerada para análise;
2. Identificação: através de análises de dados como autocorrelação, autocorrelação parcial e outros é possível confirmar a necessidade do grau de diferenciação a fim de induzir estacionaridade;
3. Estimação de parâmetros: etapa em que são definidos os coeficientes p e q utilizando métodos como autocorrelação e autocorrelação parcial, sendo p a ordem da componente auto-regressiva (AR) e q a ordem da componente de médias móveis (MA)¹;
4. Validação e aplicação do modelo: etapa em que o modelo é testado de forma que os resíduos, diferença entre valor real e valor previsto, devem caracterizar um ruído branco².

Caso o modelo não seja adequado, o ciclo é repetido a partir da fase de Identificação, na qual é comum que vários modelos diferentes sejam propostos para que posteriormente sejam estimados e verificados.

¹ Os coeficientes p e q serão definidos a seguir.

² Série com média e variância constante.

3.1.1 Modelo Auto-Regressivo (AR)

Dado um valor Z_t de uma série temporal estacionária Z , o Modelo Auto-Regressivo de ordem p é definido como uma combinação linear dos p atrasos de Z_t com a adição de um ruído branco, como mostra a Equação 14.

$$Z_t = \phi_1 Z_{t-1} + \dots + \phi_p Z_{t-p} + \varepsilon_t \quad (14)$$

no qual ε_t é um ruído branco e ϕ_i com $1 \leq i \leq p$ são os coeficientes calculados a fim de minimizar ε_t .

O processo de definição dos coeficientes ϕ_i começa com uma manipulação na Equação 14, no qual ambos os lados são multiplicados por Z_{t-j} , sendo $j \in \{1, 2, \dots, p\}$. Assim temos a Equação 15.

$$Z_t Z_{t-j} = \phi_1 Z_{t-1} Z_{t-j} + \dots + \phi_p Z_{t-p} Z_{t-j} + \varepsilon_t Z_{t-j} \quad (15)$$

Em seguida, aplica-se o operador de Esperança Matemática, gerando a Equação 16.

$$E(Z_t Z_{t-j}) = E(\phi_1 Z_{t-1} Z_{t-j}) + \dots + E(\phi_p Z_{t-p} Z_{t-j}) + E(\varepsilon_t Z_{t-j}) \quad (16)$$

Considerando $j > 0$ tem-se que $E(\varepsilon_t Z_{t-j})$ e a média do processo são iguais a zero. Note que é possível calcular os valores de autocovariância entre os termos da série de acordo com a Sessão 2.2 resultando na Equação 17.

$$\gamma_j = \phi_1 \gamma_{j-1} + \dots + \phi_p \gamma_{j-p}, \forall p \leq j > 0 \quad (17)$$

Conforme mostrado na Sessão 2.2, ao dividir a covariância por $\gamma(0)$ encontra-se a autocorrelação e com isso, ao dividir a Equação 17 por $\gamma(0)$ tem-se como resultado a Equação 18.

$$\rho_j = \phi_1 \rho_{j-1} + \dots + \phi_p \rho_{j-p}, \forall j > 0 \quad (18)$$

Ao expandir essa relação para $j = 1, 2, \dots, p$ tem-se um sistema de equações lineares, apresentado na Equação 19, que define $\phi_1, \phi_2, \dots, \phi_p$ em função de $\rho_1, \rho_2, \dots, \rho_p$.

$$\begin{aligned} \rho_1 &= \phi_1 \rho_0 + \phi_2 \rho_1 + \phi_3 \rho_2 + \dots + \phi_p \rho_{p-1} \\ \rho_2 &= \phi_1 \rho_1 + \phi_2 \rho_0 + \phi_3 \rho_1 + \dots + \phi_p \rho_{p-2} \\ \rho_3 &= \phi_1 \rho_2 + \phi_2 \rho_1 + \phi_3 \rho_0 + \dots + \phi_p \rho_{p-3} \\ &\dots \\ \rho_p &= \phi_0 \rho_{p-1} + \phi_1 \rho_{p-2} + \phi_2 \rho_{p-3} + \dots + \phi_p \rho_0 \end{aligned} \quad (19)$$

Tais relações são chamadas de equações de Yule-Walker e podem ser escritas também na forma matricial, apresentada na Equação 20, na qual tem-se:

$$\mathbf{P}_p = \begin{bmatrix} 1 & \rho_1 & \rho_2 & \dots & \rho_{p-1} \\ \rho_1 & 1 & \rho_1 & \dots & \rho_{p-2} \\ \rho_2 & \rho_1 & 1 & \dots & \rho_{p-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho_{p-1} & \rho_{p-2} & \rho_{p-3} & \dots & 1 \end{bmatrix}, \boldsymbol{\rho}_p = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ \vdots \\ \rho_p \end{bmatrix}, \boldsymbol{\Phi}_p = \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \vdots \\ \phi_p \end{bmatrix} \quad (20)$$

Note que $\rho_0 = 1$ pois $\rho_0 = \frac{\gamma(0)}{\gamma(0)}$.

Por fim, para determinar os coeficientes do modelo AR é suficiente resolver a multiplicação de matrizes dada pela Equação 21:

$$\boldsymbol{\Phi}_p = \mathbf{P}_p^{-1} \boldsymbol{\rho}_p \quad (21)$$

3.1.2 Modelo de Médias Móveis (MA)

Considere Z_t um valor da série temporal estacionária Z o qual será um Modelo de Médias Móveis com q ruídos brancos anteriores denotado por MA(q), representado pela Equação 22 onde $\theta_1, \dots, \theta_q$ são os coeficientes do modelo.

$$Z_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad (22)$$

Dessa forma, qualquer valor Z_t da série é expresso como uma combinação dos q erros aleatórios passados somado a um erro aleatório corrente (BOX *et al.*, 2015). Neste processo, o cálculo dos coeficientes não é trivial pois envolve a resolução de um sistema de equações não-lineares, diferente do caso Auto-Regressivo.

3.1.3 Modelo Auto-Regressivo de Médias Móveis (ARMA)

O modelo Auto-Regressivo de Médias Móveis (ARMA, do inglês *Auto-Regressive and Moving Average*) resulta da combinação do processo Auto-Regressivo (AR) com o de Médias Móveis (MA) (BOX *et al.*, 2015). Assim, para realizar previsões com o modelo ARMA é necessário usar os p sinais anteriores e q sinais de ruído branco. Este é descrito matematicamente pela Equação 23, sendo ϕ_t e θ_t os parâmetros do modelo cuja escolha envolve a resolução de equações não-lineares.

$$Z_t = \sum_{i=1}^p \phi_i Z_{t-i} - \sum_{i=1}^q \theta_i \varepsilon_{t-i} + \varepsilon_t \quad (23)$$

Apesar da dificuldade, a escolha adequada pode resultar em preditor linear ótimo se adequadamente identificado e otimizado (SIQUEIRA; LUNA, 2019).

Existe ainda uma relação entre a quantidade de parâmetros auto-regressivos e médias móveis. Caso o modelo apresente mais parâmetros auto-regressivos do que médias móveis, ou seja $p > q$, a função de autocorrelação apresentará um comportamento exponencial e/ou senoidal amortecido após o lag $q - p$. Por outro lado, se $q = p$, as autocorrelações iniciais não seguirão esse padrão (SILVA *et al.*, 2017). Porém, é importante destacar que somente uma análise gráfica da série temporal não é suficiente para identificar o modelo.

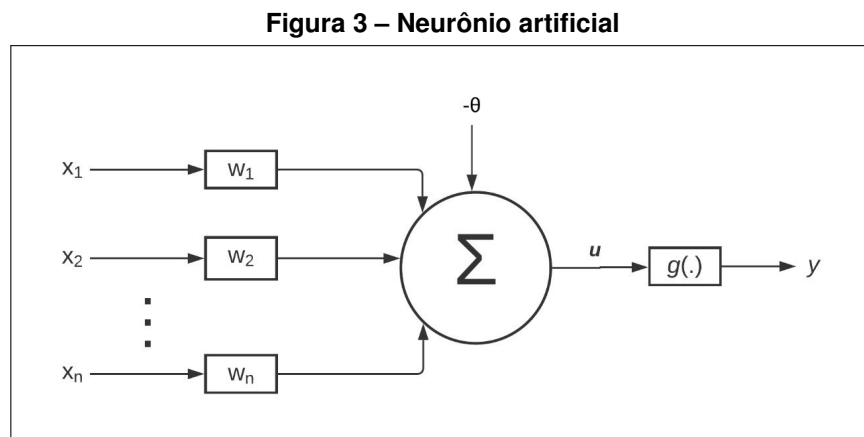
3.2 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNAs) são definidas como sistemas de processamento paralelo e distribuído que apresentam características bastante atrativas como processamento de informação com incerteza e capacidade de aprendizagem (SOUZA, 2014).

A inspiração para as RNAs provém do funcionamento do sistema nervoso dos organismos superiores. Isso as leva a serem compostas por unidades de processamento simples, porém capazes de produzir um elevado grau de interconexão, denominadas de neurônios artificiais (HAYKIN, 2007). No que diz respeito a biologia, os neurônios desempenham os papéis de receber, processar e encaminhar informações através de pulsos elétricos entre dendritos e axônios. Este processo é chamado de sinapse (SIQUEIRA *et al.*, 2013).

McCulloch e Pitts (1943) introduziram o primeiro neurônio artificial que consiste em um modelo matemático com n terminais que representam os dendritos, e um terminal de saída y representando o axônio.

A Figura 3 apresenta um modelo genérico de um neurônio artificial bastante utilizado nos dias atuais. Este neurônio recebe um conjunto de entradas $\mathbf{x} = [x_1, x_2, \dots, x_n]$, sendo n a quantidade de entradas.



Fonte: Autoria própria (2022)

Cada entrada $x_i \in \mathbf{x}(n)$ é ponderada por um peso sináptico artificial w_i , responsável por simular a sinapse natural. Esses pesos podem assumir valores positivos ou negativos, conforme o potencial sináptico excitatório ou inibitório. Isso implica no quanto o neurônio deve considerar os sinais de disparo de tal conexão.

A Junção Aditiva, representada por Σ , é responsável por somar todos os valores recebidos das entradas ponderadas pelos respectivos pesos sinápticos. O neurônio artificial em questão também possui um viés, representado por θ , que pode ser interpretado como um peso, sendo responsável por inserir o viés, ou *offset*, na resposta. Por fim, o valor u , resultado de saída da Junção Aditiva, é submetido a uma função de ativação $g(\cdot)$, responsável por restringir a amplitude do resultado a um valor finito, geralmente entre $[0,1]$ ou $[-1,1]$, e também inserir um grau de não-linearidade à saída do neurônio.

Um parâmetro importante na concepção das RNAs é a arquitetura neural, definida pelo arranjo dos neurônios nas camadas e do padrão de interconexão entre estas. O modo como os neurônios são dispostos está diretamente ligado ao algoritmo utilizado para treinar (SIQUEIRA *et al.*, 2013; ALENCAR, 2018).

Em relação aos padrões de interconexão, as RNAs podem ser classificadas em (SIQUEIRA *et al.*, 2013):

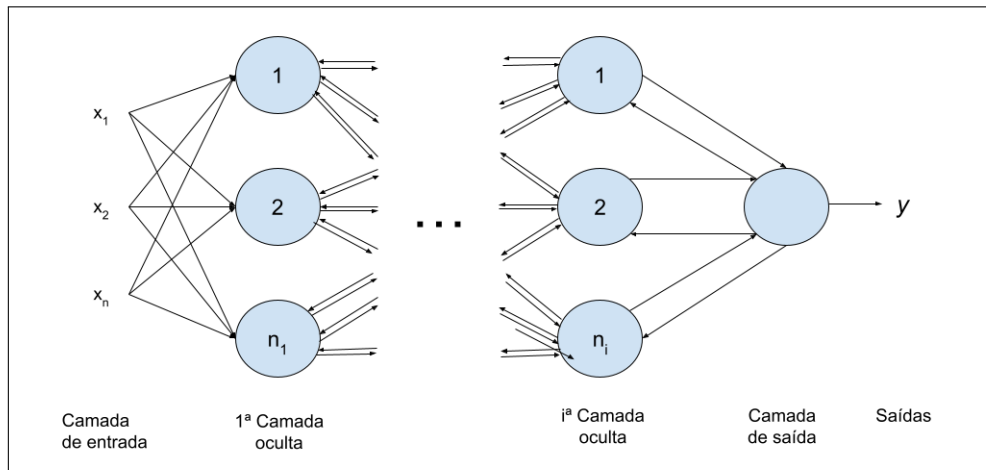
- *Feedforward*: neste caso, o sentido no fluxo de informação na rede flui da entrada para a saída;
- Recorrente: apresenta realimentação, que consiste em transmitir informações para camadas anteriores, ou até mesmo entre neurônios da mesma camada.

3.2.1 Perceptron De Múltiplas Camadas

A rede Perceptron de Múltiplas Camadas (*Multilayer Perceptron* - MLP) é uma das arquiteturas *feedforward* mais conhecidas por conta da sua capacidade de aproximar qualquer tipo de função não-linear contínua, limitada e diferenciável definida num espaço compacto (SIQUEIRA *et al.*, 2013). A MLP em sua versão mais conhecida é composta por:

- Camada de entrada: responsável por receber os sinais de entrada e transmiti-los para a camada oculta;
- Camadas ocultas (ou intermediárias): designadas para mapear os sinais de entrada de forma não linear. É importante destacar que a MLP é flexível em relação à quantidade de camadas ocultadas, sendo necessário no mínimo uma camada;
- Camada de saída: recebe os sinais transformados e através de combinações lineares produz uma saída.

Figura 4 – MLP



Fonte: Autoria própria (2022)

A Figura 4 a seguir apresenta uma generalização de MLP:

O processo de aprendizagem da rede, conhecido como treinamento, consiste em decidir quais características do padrão de entrada devem ser representadas pelos neurônios ocultos. Em outras palavras, a aprendizagem é o processo de ajuste dos pesos sinápticos que têm como objetivo encontrar a melhor combinação para o mapeamento requerido.

Para isso costuma-se utilizar um algoritmo de treinamento bastante difundido chamado *backpropagation* que inclui a otimização linear irrestrita do gradiente ótimo (SIQUEIRA *et al.*, 2013; ALENCAR, 2018). Sua aplicação é dividida em duas fases:

- *Forward*: os dados de entrada são propagados através da rede, camada por camada, até a camada de saída;
- *Backward*: a partir da diferença entre sinal desejado e a resposta da rede é obtido o erro. Este é utilizado para calcular o valor do ajuste dos pesos da rede, os quais são propagados na direção inversa.

A medida de erro mais utilizada para o treinamento é Erro Quadrático Médio (MSE, do inglês *Mean Squared Error*) que é obtido somando o quadrado do erro de todos os neurônios da camada de saída (SOUZA, 2014).

Em síntese, as aplicações sucessivas das fases *forward* e *backward* fazem com que os pesos sinápticos e limiares dos neurônios se ajustem a cada iteração, implicando em uma gradativa diminuição dos erros de saída (ALENCAR, 2018)

3.2.2 Redes Neurais com Função De Base Radial

As Redes Neurais com Função de Base Radial (*Radial Basis Function Network* - RBF) consistem em um modelo com apenas três camadas - entrada, intermediária e saída.

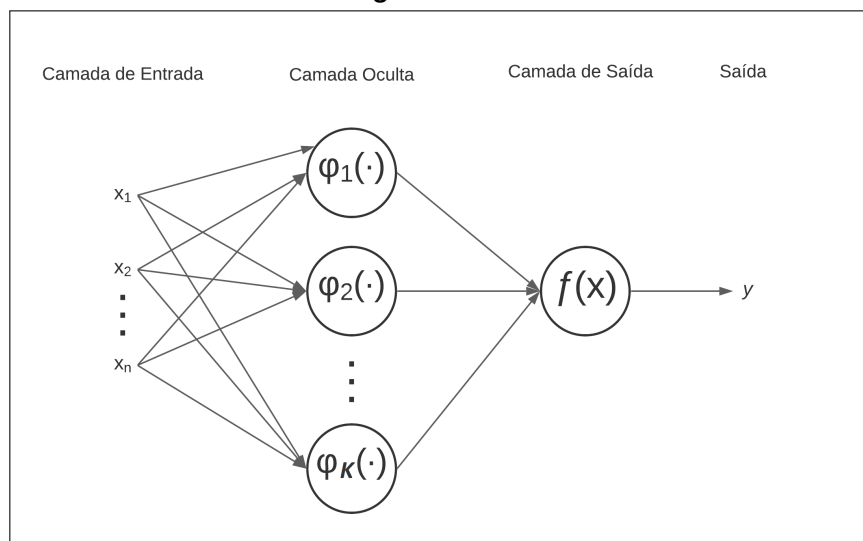
A camada intermediária das RBFs utiliza funções de base radial, tipicamente não-linear. É comum a escolha da função gaussiana, Equação 24, no qual c é o centro da função, σ a variância e \mathbf{x} o conjunto de entradas.

$$\varphi(\mathbf{x}) = e^{-\frac{(\mathbf{x}-c)^2}{2\sigma^2}}, \quad (24)$$

Esta é ajustada utilizando alguma medida de distância para um dado centro (c), por meio de um processo de clusterização (AGGARWAL, 2018).

A Figura 5 apresenta a arquitetura de uma RBF, na qual cada neurônio da camada oculta apresenta uma função gaussiana representada por φ_N .

Figura 5 – RBF



Fonte: Autoria própria (2022)

Nesta representação há apenas uma saída, mas a estrutura admite que sejam múltiplas.

Para definir os valores iniciais de cada centro há diversas técnicas, sendo a geração aleatória a mais simples e menos custosa (HAYKIN, 2007). Em relação à camada de saída, esta é dada pela Equação 25, no qual k é a quantidade de neurônios na camada escondida, w_i o peso entre o neurônio i , \mathbf{x} o conjunto de entradas e φ a função gaussiana.

$$f(\mathbf{x}) = \sum_{i=1}^k w_i \varphi(\mathbf{x}), \quad (25)$$

Um dos principais fundamentos das RBF é baseado no teorema de Cover (1965). Este afirma que os problemas de classificação de padrões são mais prováveis de ser linearmente separáveis quando lançados em um domínio de maior dimensão com uma transformação não-linear, sendo tal tarefa empregada à camada escondida.

O processo de treinamento da RBF consiste em duas etapas. Primeiro é realizado o ajuste dos pesos referentes aos neurônios da camada escondida. Tal ajuste está diretamente ligado à alocação das funções de base radial e é caracterizado por ser não-supervisionado,

ou seja, não se considera a distribuição e localização espacial. Assim, é usual lançar mão de métodos de clusterização como o K-Means.

Somente após a finalização da primeira etapa é feito o ajuste da camada de saída. Diferente da camada oculta, na camada de saída é feito um ajuste supervisionado semelhante ao da MLP, usando, por exemplo o *backpropagation* (NUNES; SILVA, 2017).

3.2.3 Máquina De Aprendizado Extremo

HUANG, ZHU e SIEW (2004) propuseram uma nova metodologia de treinamento para redes SLFNs (*Single Layered Feedforward Neural Network*) com a rede neural Máquina de Aprendizado Extremo (*Extreme Learning Machine-ELM*). Esta apresenta um padrão *feed-forward* e uma arquitetura com apenas uma camada intermediária, semelhante às tradicionais MLPs.

As ELMs escolhem aleatoriamente os pesos da camada intermediária e não utilizam algoritmos iterativos para ajuste dos mesmos. Dessa forma, a tarefa de definir coeficientes ótimos é realizada apenas na camada de saída, utilizando a pseudo inversa de Moore-Penrose, que minimiza o Erro Quadrático Médio entre o sinal desejado e a saída dada (SIQUEIRA *et al.*, 2013).

O processo de treinamento das ELMs pode ser definido em três etapas (SILVESTRE, 2014):

1. $\mathbf{H} = \Psi(\mathbf{x}, Z)$, os dados de entrada \mathbf{x} e os pesos escolhidos aleatoriamente por uma distribuição uniforme Z são mapeados, com o objetivo de linearizar o problema;
2. $\mathbf{W} = \mathbf{H}^g y$, define-se a matriz que será utilizada para calcular a saída através da pseudo inversa de \mathbf{H} , representada por \mathbf{H}^g , e os sinais desejados y .
3. $\hat{y} = \mathbf{H}\mathbf{W}$, por fim, o sinal de saída é calculado.

A Figura 6 representa uma arquitetura genérica de uma ELM.

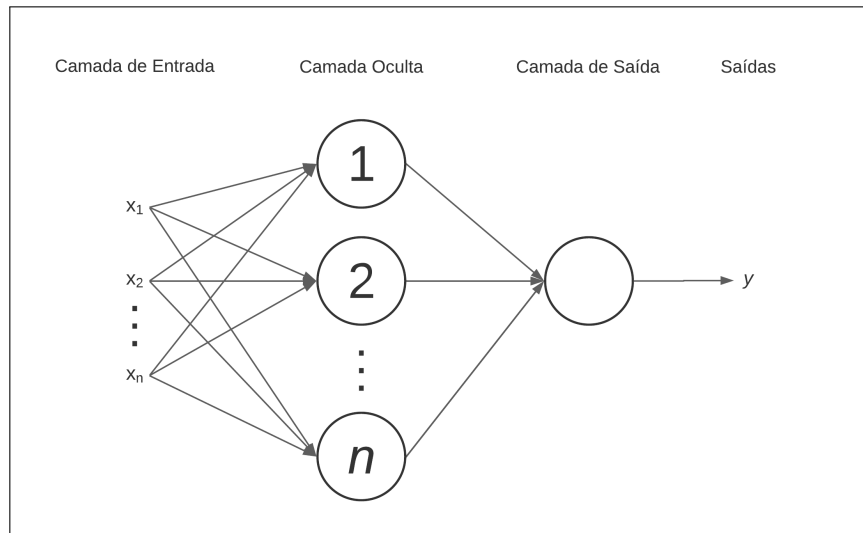
Por não envolver um processo iterativo de ajustes, as ELMs são mais rápidas em seu treinamento quando comparadas às redes convencionais, como MLP.

3.2.4 Redes Neurais Com Estado De Eco

As Redes Neurais com Estado De Eco (*Echo State Networks-ESNs*), introduzidas por Jaeger (2001), são uma simplificação de uma rede neural recorrente clássica pois apresentam interconexão entre os neurônios da camada intermediária. Tal característica é vantajosa para tarefas dinâmicas ou com características temporais (AGGARWAL, 2018).

As redes recorrentes clássicas, apesar do grande poder de processamento, têm um processo de ajuste de pesos custoso. Em geral, os algoritmos apresentam convergência lenta e

Figura 6 – ELM



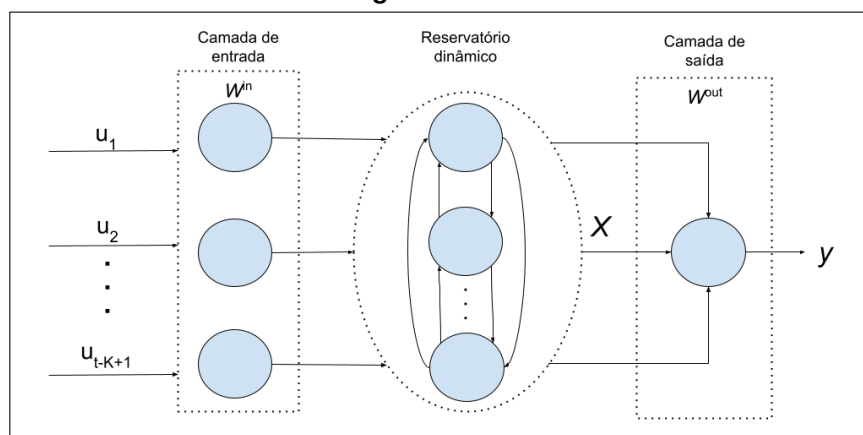
Fonte: Autoria própria (2022)

parâmetros de difícil definição. Contrariamente, tem-se as ESNs que mesmo sendo recorrentes se diferenciam pela construção da rede e processo de treinamento (SIQUEIRA *et al.*, 2013).

Assim como as ELMs, as ESNs também possuem camadas intermediárias não ajustadas, aqui chamadas de reservatório de dinâmicas, em que os pesos são escolhidos aleatoriamente. Nessas redes, cada sinal que sai do reservatório é chamado de estado de eco. Estes sinais são direcionados à camada de saída, que irá produzir sinais por meio de combinação linear. Dessa forma, o treinamento resume-se em encontrar coeficientes ótimos da camada de saída. Para isso, utiliza-se a pseudo-inversa de Moore-Penrose (SIQUEIRA *et al.*, 2013).

O reservatório de dinâmicas, presente na Figura 7, possui neurônios conectados entre si de forma aleatória.

Figura 7 – ESN



Fonte: Autoria própria (2022)

As ESNs utilizadas neste trabalho apresentam três conexões ponderadas (JAEGER, 2001):

- \mathbf{W}^{in} : conexões de entrada - localizadas entre a camada de entrada e o reservatório;
- \mathbf{W} : conexões internas (reservatório de dinâmicas) - ligam neurônios do reservatório;
- \mathbf{W}^{out} : conexões de saída - levam informação do reservatório para a camada de saída.

As matrizes \mathbf{W}^{in} e \mathbf{W} são escolhidas aleatoriamente e não são ajustadas durante o processo de treinamento. Assim, a ativação da saída do reservatório é dada pela Equação 26

$$\mathbf{x}_{t+1} = f(\mathbf{W}^{in}\mathbf{u}_{t+1} + \mathbf{W}\mathbf{x}_t), \quad (26)$$

na qual f é uma função de ativação, \mathbf{u} é o vetor de entrada e \mathbf{x} são os estados de eco presentes na saída do reservatório.

Em relação à saída da rede, esta é calculada de acordo com a Equação 27.

$$\mathbf{y}_{t+1} = \mathbf{W}^{out}\mathbf{x}_{t+1}, \quad (27)$$

Em resumo, o processo de treinamento das ESNs é semelhante ao das ELMs, no que diz respeito a minimizar o Erro Quadrático Médio entre a saída da rede e o sinal desejado. Por fim, para definir a matriz de pesos \mathbf{W}^{out} também é utilizada a pseudo-inversa de Moore-Penrose como mostra a Equação 28.

$$\mathbf{W}^{out} = \mathbf{x}^g\mathbf{y}, \quad (28)$$

sendo \mathbf{x} a matriz de estados de eco e \mathbf{y} a saída desejada.

O processo de treinamento simplificado em uma rede recorrente é possível pois a ativação de cada neurônio do reservatório é, na verdade, uma transformação não-linear de histórico mais recente do sinal de entrada. Destaca-se a presença de uma memória intrínseca que guarda informações sobre o sinal e pode ser utilizada para previsões posteriores (SIQUEIRA *et al.*, 2013).

Jaeger (2001) investigou a dinâmica de uma rede neural recorrente e comprovou a propriedade de estados de eco, constatando que seus estados são assintoticamente independentes. Em outras palavras, se esta rede for inicializada com dois valores diferentes, mas houver a mesma sequência temporal de entradas, então os estados resultantes serão similares.

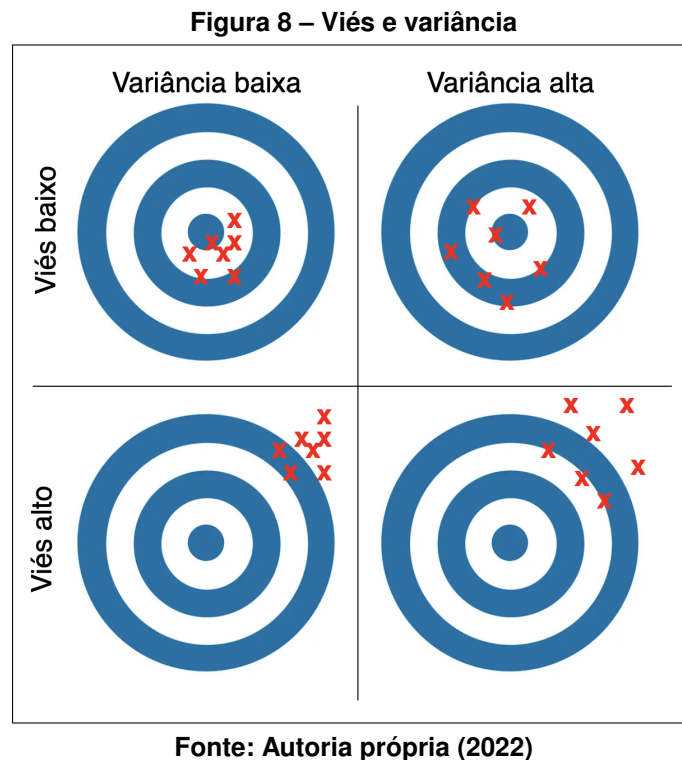
Uma maneira simples de criar um reservatório de dinâmicas diversificado também foi apresentado por Jaeger (2001). Nele é criada uma matriz de pesos com certo grau de esparsidade, favorecendo o desacoplamento de grupos de neurônios e, com isso, induzindo o desenvolvimento de dinâmicas individuais e pouco relacionadas.

A criação dessa matriz pode ser feita conforme mostra a Equação 29, na qual há três valores possíveis.

$$W = \begin{cases} 0,4 & \text{com probabilidade } 0,025, \\ -0,4 & \text{com probabilidade } 0,025 \\ 0 & \text{com probabilidade } 0,95 \end{cases} \quad (29)$$

3.3 Metodologia *Ensemble*

Os *Ensembles* foram propostos para resolver o *trade-off* entre viés e variância (AGGARWAL, 2018). O viés pode ser definido como a diferença entre valor real e predito, medindo a acurácia do estimador, enquanto a variância é medida aponta as flutuações das previsões do modelo, sendo responsável por determinar a precisão de um estimador. A Figura 8 apresenta visualmente as combinações entre viés e variância num exemplo hipotético.

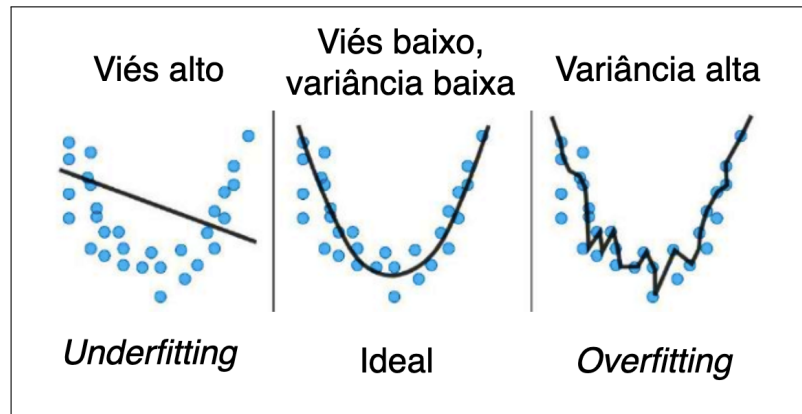


O cenário desejado para modelos de previsão de séries temporais é ter baixo viés e baixa variância. Entretanto, a medida que um diminui o outro aumenta, resultando em *overfitting* ou *underfitting* como mostra a Figura 9.

O *overfitting* está relacionado aos valores elevados de variância, indicando que um estimador considerou os ruídos em sua modelagem. Por outro lado, quando o viés é elevado o estimador não realiza uma boa aproximação entre os valores reais e as saídas previstas - *underfitting*.

Diante disso, a proposta dos *Ensembles* é, a partir de um conjunto de estimadores com baixo viés - e por consequência alta variância, realizar uma combinação de forma que a vari-

Figura 9 – Viés, variância e o cenário de previsão ideal



Fonte: Autoria própria (2022)

ância se torne baixa. Essa metodologia é comumente aplicada utilizando RNAs por conta da sua capacidade de construir modelos complexos com viés relativamente baixos (AGGARWAL, 2018).

Em outros termos, um *Ensemble* é uma estrutura de aprendizado composta por um conjunto finito de modelos treinados para resolver um dado problema e posteriormente combinar suas saídas.

O uso da metodologia *Ensemble* torna-se interessante por receber inúmeras previsões para cada amostra vindas de diferentes estimadores - cada um apresentando uma diferença entre valor real e predito - e ao final produzir uma saída com erro reduzido (ZHANG; MA, 2012).

A combinação de diferentes algoritmos preditores não garante uma performance acima do melhor algoritmo preditivo, entretanto em *Machine Learning* não é possível assegurar em quantidade de tempo polinomial qual o melhor algoritmo de predição e quais os seus melhores parâmetros. Contudo, por meio de um *Ensemble* a probabilidade de escolher um estimador ruim diminui (ZHANG; MA, 2012), ou seja, o *Ensemble* será garantidamente melhor que o pior modelo, mesmo que este faça parte do conjunto a ser combinado. Para a construção de um *Ensemble* três passos devem ser previamente definidos: amostragem/seleção de dados, treinamento, e combinação, que são descritos a seguir.

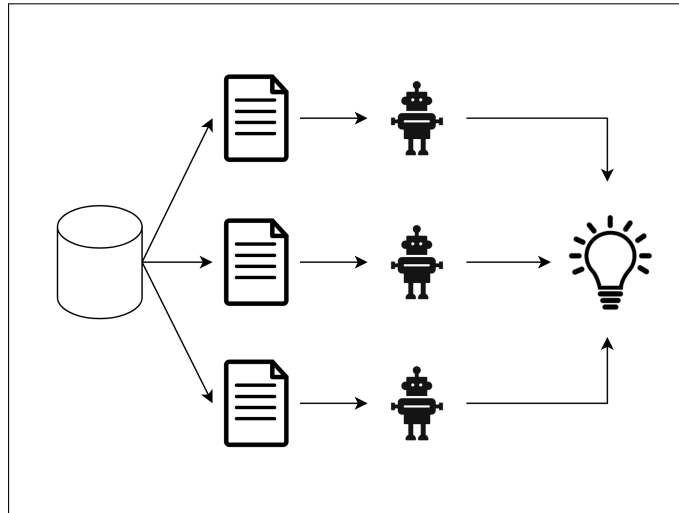
3.3.1 Amostragem/Seleção de dados

Em um *Ensemble* é importante que os estimadores produzam diferentes erros para cada amostra, pois do contrário não há ganho com a formação do sistema. A diversidade é um fator que deve estar presente nos *Ensembles*, e, uma das maneiras de induzir tal diversidade, consiste em criar subconjuntos de amostras. A partir de tal ideia foram propostos dois métodos de amostragem, *Bagging* e *Boosting*.

O *Bagging*, Figura 10, é um método que foi introduzido por Breiman (1996). Nele são criados N subconjuntos de amostras diferentes a partir do conjunto de amostras iniciais, sendo

N a quantidade de estimadores. Dessa forma, cada subconjunto permitirá a construção de um modelo diferente, já que cada estimador é treinado de forma independente.

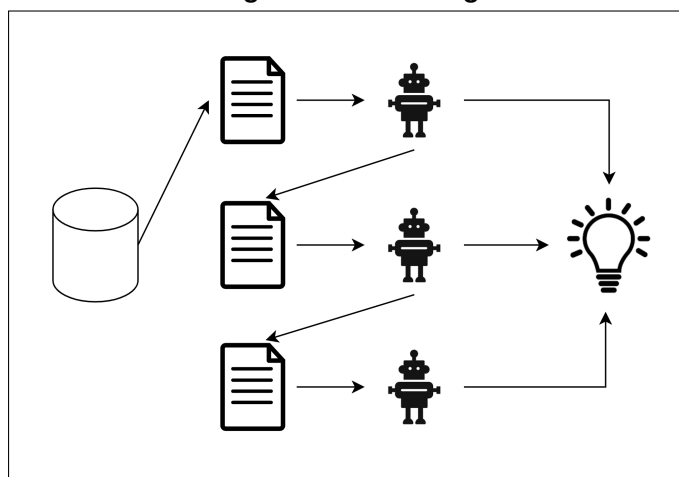
Figura 10 – Bagging



Fonte: Autoria própria (2022)

O *Boosting* (Figura 11), proposto por Schapire (1990), consiste em realizar um treinamento sequencial e com re-amostragem de dados. Nele, o primeiro modelo recebe os dados de treinamento e realiza uma previsão. Em seguida, considerando os erros do modelo anterior, um conjunto de dados é gerado a fim de complementar as previsões anteriores na etapa de combinação.

Figura 11 – Boosting



Fonte: Autoria própria (2022)

Ambas as técnicas são amplamente usadas nas tarefas de classificação e regressão. Em previsão de séries temporais há estudos que mostram um ganho de acurácia na aplicação de tais métodos como em (NGO *et al.*, 2022; GHAEMI *et al.*, 2022; ZHANG *et al.*, 2021; LI *et al.*, 2019).

3.3.2 *Ensemble* Homogêneo e Heterogêneo

No núcleo de todo sistema *Ensemble* encontra-se a estratégia de treinamento. A escolha dessa depende, antes de tudo, da similaridade entre os estimadores selecionados. Estimadores iguais caracterizam um *Ensemble* homogêneo, do contrário tem-se um *Ensemble* heterogêneo.

Os *Ensembles* homogêneos são caracterizadas por conter apenas um tipo de estimador, deixando a tarefa da diversidade, na maioria dos casos, para as estratégias de amostragem que entregam diferentes conjuntos de dados para cada estimador - *Bagging* e *Boosting*. Outra abordagem comum é utilizar o mesmo conjunto de dados para estimadores do mesmo tipo que se diferem nos hiper-parâmetros³.

Se tratando de *Ensembles* heterogêneos, há dois algoritmos de treinamento de maior destaque: *Stacking* e a mistura de especialistas (*Mixture of Experts* - MoE) (ZHANG; MA, 2012). Ambos trazem o conceito de *meta-models* que será discutidos na Sessão 3.3.3.

O *Stacking* - *Stacked Generalization* possui uma arquitetura de dois níveis. No primeiro nível encontram-se os modelos base (*base-models*) que são estimadores responsáveis por receber os dados e gerar uma saída prevista que será usada como entrada para o nível seguinte. Neste, está presente o *meta-model* que é um estimador cuja tarefa é combinar as saídas dos *base-models*.

Geralmente, para treinar os *base-models* o conjunto de dados inicial é separado em subconjuntos de forma que um é usado para teste e os demais para treinamento. Por exemplo, assumindo que há N subconjuntos, os de número 1 à $N - 1$ são usados para ajustar os M estimadores de forma que cada um tenha um subconjunto de treinamento semelhante, mas não idêntico, ao de outro. Após a fase de treinamento, o subconjunto N será dado como entrada para todos os estimadores e as suas saídas - valores previstos - serão dadas como entrada para o *meta-model* ou combinador.

Outro algoritmo famoso é o MoE, que se baseia no princípio de dividir para conquistar. Nele o espaço de problema é dividido entre os estimadores individuais. Tal divisão pode ser feita por meio de função de erro ou então por *clusters*. Com isso, cada estimador se torna especialista em uma região do problema.

Na arquitetura do MoE também encontra-se um *meta-model*, aqui chamado de *gating network*, responsável por ponderar os estimadores. Tais pesos são dinamicamente estimados com os dados de entrada. Por fim, utiliza-se um sistema de combinação para gerar uma saída final.

3.3.3 Combinadores

O último passo a ser definido no *Ensemble* é o combinador. Este é responsável por receber as saídas de cada estimador e gerar uma única saída. A escolha do combinador depende,

³ Parâmetros ajustáveis que permitem controlar o processo de treinamento do modelo

em partes, dos tipos de estimadores usados e os tipos de saída geradas. Para saídas com valores contínuos é possível usar combinadores algébricos não-treináveis ou treináveis (POLIKAR, 2012). Os combinadores algébricos geram uma saída final a partir da aplicação de uma função aritmética simples sobre as saídas dos *base-models*, como exemplificado a seguir:

- Média: a saída final é a média sobre as saídas dos *base-models*;
- Média truncada: a saída final é a média de um subconjunto das saídas dos *base-models*. Uma porcentagem dos valores muito baixos e muito altos é desconsiderada. Essa porcentagem geralmente varia entre 10% e 30%;
- Mínimo/Máximo/Mediana: a saída final consiste no valor mínimo, máximo ou mediana das saídas individuais dos *base-models*;
- Produto: realiza-se o produtório das saídas individuais e posteriormente normaliza-se este valor dividindo-o por T , sendo T a quantidade de *base-models*;

Os combinadores apresentados realizam a junção das saídas de forma que não é possível levar em consideração a especialidade de um *base-model*. Por outro lado, os combinadores treináveis - que precisam de ajuste e/ou treinamento - têm o poder de determinar a probabilidade de sucesso de cada *base-model* sobre uma parte do problema. Nessa categoria encontram-se combinadores como média ponderada e RNAs - *meta-model*.

A média ponderada estabelece um peso para cada *base-model*. Este valor pode ser encontrado, por exemplo, através da quantidade de vezes que tal estimador foi melhor que os outros. Com isso, a saída final é a média do valor das saídas individuais multiplicada pelo peso de seu estimador.

Um *meta-model* é um estimador cuja tarefa é determinar a relevância de cada *base-model* para determinadas partes do problema, e, com isso, compor uma saída final melhor (JACOBS *et al.*, 1991). Para tanto, é possível utilizar RNAs como as arquiteturas apresentadas no Capítulo 3.2, por exemplo MLP, RBF, ELM e ESN.

3.4 Sistemas Híbridos de correção de erro

Assim como os *Ensembles*, os Sistemas Híbridos de Correção de Erro também realizam um tipo de combinação entre estimadores a fim de obter melhores resultados na previsão de séries temporais. Entretanto, há diferenças entre ambos: enquanto os *Ensembles* buscam resolver o *trade-off* ente viés e variância, a metodologia proposta por Zhang (2003) busca melhorar a acurácia.

A inspiração do método parte do pressuposto que a maioria dos problemas do mundo real são complexos e apresentam componentes lineares e não-lineares que podem ser previstos separadamente e posteriormente combinados. Tal combinação pode ser feita por operado-

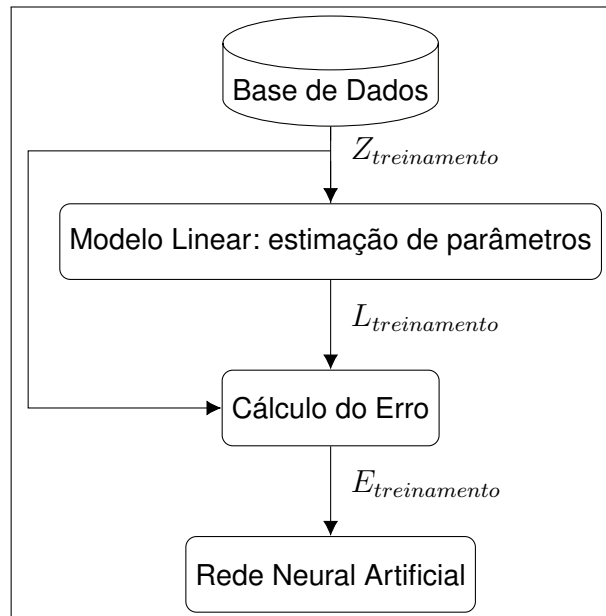
res aritméticos como a soma, RNAs ou *Ensembles*. No trabalho de Saranj e Zolfaghari (2022) seu modelo híbrido utiliza como combinador uma Floresta Aleatória, um *Ensemble* homogêneo composto por Árvores de Decisão.

A previsão final de um Sistema Híbrido é feita conforme Equação 30 no qual \hat{Z}_{t+1} é a saída final do sistema Híbrido, \hat{L}_{t+1} é a previsão feita por um modelo linear e \hat{N}_{t+1} é a previsão do erro por um modelo não-linear, em geral uma RNA.

$$\hat{Z}_{t+1} = \hat{L}_{t+1} + \hat{N}_{t+1}, \quad (30)$$

A Figura 12 apresenta um fluxograma da etapa de treinamento do Sistema Híbrido proposto por Zhang (2003).

Figura 12 – Fluxograma do processo de treinamento do Sistema Híbrido de Correção de Erro



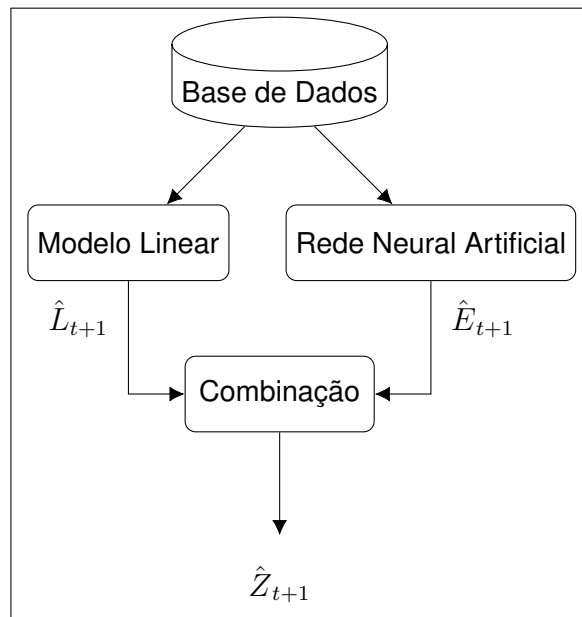
Fonte: Autoria própria (2022)

O processo de treinamento é feito basicamente em três etapas. Primeiro é feita a estimação de parâmetros do modelo que irá prever uma série com padrão linear. A segunda etapa é responsável por receber a série prevista e os valores reais. Com isso, calcula-se o erro através da diferença entre valor real e valor previsto. Este erro forma uma série temporal não-linear que pode ser prevista com RNAs, por exemplo.

Na etapa de teste, mostrada pela Figura 13, é feita a previsão do conjunto de teste pelo modelo linear resultando no conjunto de validação e teste que será utilizado pela RNA. Por fim, com a previsão linear e não-linear em mãos é feita a combinação.

Em suma, a metodologia consiste em dois passos. Primeiro um preditor linear modela o problema e realiza uma previsão com base nos dados de entrada e a partir do erro obtido ocorre o segundo passo, a previsão do erro através de uma RNA.

Figura 13 – Fluxograma do processo de teste do Sistema Híbrido de Correção de Erro



Fonte: Autoria própria (2022)

4 PREVISÃO DE SÉRIES TEMPORAIS COMO UM PROBLEMA DE APRENDIZAGEM SUPERVISIONADA

A aprendizagem supervisionada é uma subcategoria de *Machine Learning* que consiste em utilizar um algoritmo para mapear as variáveis de entrada e saída dadas. Quando o mapeamento envolve uma saída do tipo real então trata-se de um problema de regressão.

Para aplicação das RNAs à previsão de séries temporais é possível modelar uma série como um problema de aprendizagem supervisionada. Dada uma sequência de valores - série temporal - é possível reestruturá-los de modo que obtém-se uma base com variáveis de entrada (X_n tal que n é a quantidade de entradas) e saída (y), como mostra a Figura 14.

Figura 14 – Reestruturação de uma série temporal para aprendizagem supervisionada

t, valor	X1, X2, y
1, 20	?, ?, 20
2, 22	20, ?, 22
3, 25	22, 20, 25
4, 23	25, 22, 23
5, 21	23, 25, 21
	21, 23, ?

Fonte: Autoria própria (2022)

Na Figura 14 a série temporal à esquerda tem seus dados reestruturados, à direita, de forma que os instantes anteriores - comumente chamados de atrasos - são as variáveis dadas como entrada para o modelo, neste caso representadas por X_1 e X_2 . É possível utilizar n atrasos e com isso ter até n variáveis de entrada.

Além disso é possível reestruturar a série de forma que a saída desejada - y - não seja o instante seguinte. Por exemplo, é possível usar os valores dos instantes t_1 e t_2 para prever o valor do instante t_5 . Dessa forma obtém-se a previsão de diferentes horizontes de forma direta.

Na literatura também é possível encontrar a técnica recursiva, em que um valor previsto é inserido na série de forma que este também será usado para prever valores futuros. Neste caso, independente do horizonte, sempre será utilizada a previsão de 1 passo a frente (BELOTTI *et al.*, 2021).

4.1 Seleção de variáveis

Dentre as tarefas a serem executadas num processo de previsão, uma muito importante é a seleção de variáveis, neste caso, de atrasos, para estimar o valor futuro. Se bem executada, além de facilitar a visualização e entendimento dos dados, também reduz o tempo de treinamento e esforço computacional (GUYON; ELISSEEFF, 2003), dado que um dos objetivos da seleção de atributos é remover variáveis preditoras não informativas ou redundantes (KUHN;

JOHNSON *et al.*, 2013). Na literatura é possível encontrar diversas abordagens, dentre elas o *Wrapper*, além da PACF mencionado na Seção 2.2.

O *Wrapper* é um método de seleção que realiza uma busca sistemática pelas melhores variáveis. Esta técnica consiste em avaliar as variáveis preditoras em um dado modelo de previsão já configurado e uma métrica de erro pré-estabelecida (FREITAS *et al.*, 2007). Em relação ao seu funcionamento, existem três propostas básicas (TAN, 2020):

- Seleção progressiva: começa com uma variável preditora e ao longo de cada iteração adiciona a melhor candidata;
- Eliminação regressiva: inicialmente considera todas as variáveis preditoras e ao longo das iterações vai eliminando as variáveis menos importantes;
- Seleção passo a passo: é baseada em uma combinação da seleção progressiva e a eliminação regressiva pois pode adicionar variáveis preditoras que tenham sido removidas.

Vale ressaltar que é necessário definir previamente a quantidade máxima de variáveis preditoras e quais serão consideradas para avaliação. Para isso é possível realizar um teste de 1 até k valores e posteriormente analisar qual quantidade de k apresenta melhor desempenho. No caso de empate é interessante utilizar a menor quantidade de k , já que um maior conjunto de variáveis de entrada não implica diretamente em melhores resultados (SIQUEIRA *et al.*, 2013).

Em relação a PACF, vale ressaltar que esta é uma função que calcula a relação entre duas variáveis assumindo que estão linearmente associadas. A base é uma aplicação sucessiva de modelos AR com diferentes ordens (SIQUEIRA; LUNA, 2019). Entretanto o trabalho de Flores, Engel e Pinto (2012) sugerem que este método é aplicável quando se deseja realizar a previsão de séries temporais não-lineares.

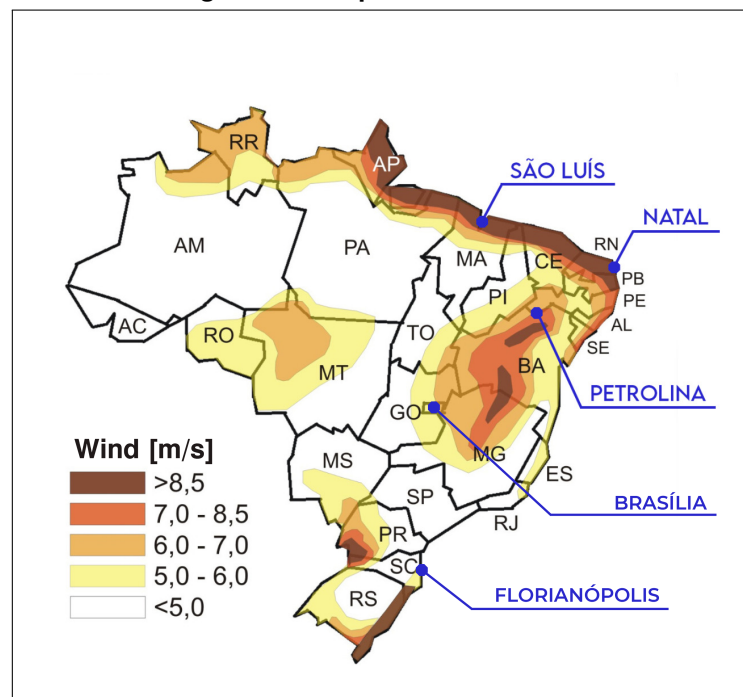
5 ESTUDOS DE CASOS

Este trabalho realizou um estudo extensivo sobre os modelos tradicionais *versus* modelos de combinação para previsão de séries da velocidade do vento, considerando o horizonte de 1 passo a frente.

Os dados históricos utilizados são referentes às cidades de Brasília, Florianópolis, Natal, Petrolina e São Luís, os quais foram obtidos através do Sistema de Organização Nacional de Dados Ambientais (INPE,), que disponibiliza em sua base de dados a velocidade do vento de diversas cidades. As bases selecionadas abrangem o mês de Janeiro e Fevereiro de 2019 e estão dispostas em intervalos de um minuto.

As cidades escolhidas apresentam diferentes padrões de velocidade de vento. Além disso, elas estão localizadas em diferentes estados do Brasil, como mostra a Figura 15. Nesta, ainda é apresentado um mapa do potencial eólico brasileiro com a velocidade média dos ventos nas diversas regiões do país.

Figura 15 – Mapa eólico do Brasil



Fonte: Adaptado de Pires (2016)

Em relação aos modelos aplicados, foram considerados os Modelos Lineares, RNAs, *Ensembles* e Sistemas Híbridos. Para comparar o desempenho de diferentes modelos preditores, é necessário lançar mão de métricas de avaliação. As escolhidas para este trabalho foram o Erro Quadrático Médio (MSE do inglês *Mean Squared Error*), o Erro Absoluto Médio (MAE do inglês *Mean Absolute Error*), a Variação Relativa Média (ARV do inglês *Average Relative Variance*) e o Índice de Concordância (IA do inglês *Index of Agreement*)(NETO *et al.*, 2010).

No primeiro calcula-se a média dos quadrados dos erros, ou seja, a diferença quadrática média entre valor real e valor predito, conforme a Equação 31.

$$MSE = \left(\frac{1}{n}\right) \sum_{i=1}^n e_i^2, \quad (31)$$

sendo e_i a diferença entre valor real e valor predito em relação ao instante i .

Tal métrica está entre as mais utilizadas para avaliação de modelos de previsão junto ao MAE, métrica que calcula a média dos desvios absolutos em relação ao valor real conforme a Equação 32.

$$MAE = \left(\frac{1}{n}\right) \sum_{i=1}^n |e_i|. \quad (32)$$

A métrica ARV, descrita pela Equação 33, mede o desempenho da previsão levando em consideração a previsão média da série temporal de modo que: I) Se $ARV = 1$, então a previsão é a mesma que a previsão da série média, II) Se $ARV > 1$, o modelo é pior que uma previsão média, e, III) Se $ARV < 1$, o modelo tem uma previsão melhor que uma previsão média da série.

$$ARV = \frac{\sum_{t=1}^N (\hat{x}_t - x_t)^2}{\sum_{t=1}^N (\hat{x}_t - \bar{x})^2}, \quad (33)$$

O Índice de Concordância (IA), Equação 34, tem como objetivo medir de forma padronizada o grau de erro de previsão que varia entre 0 e 1 indicando nenhuma ou total correspondência, respectivamente.

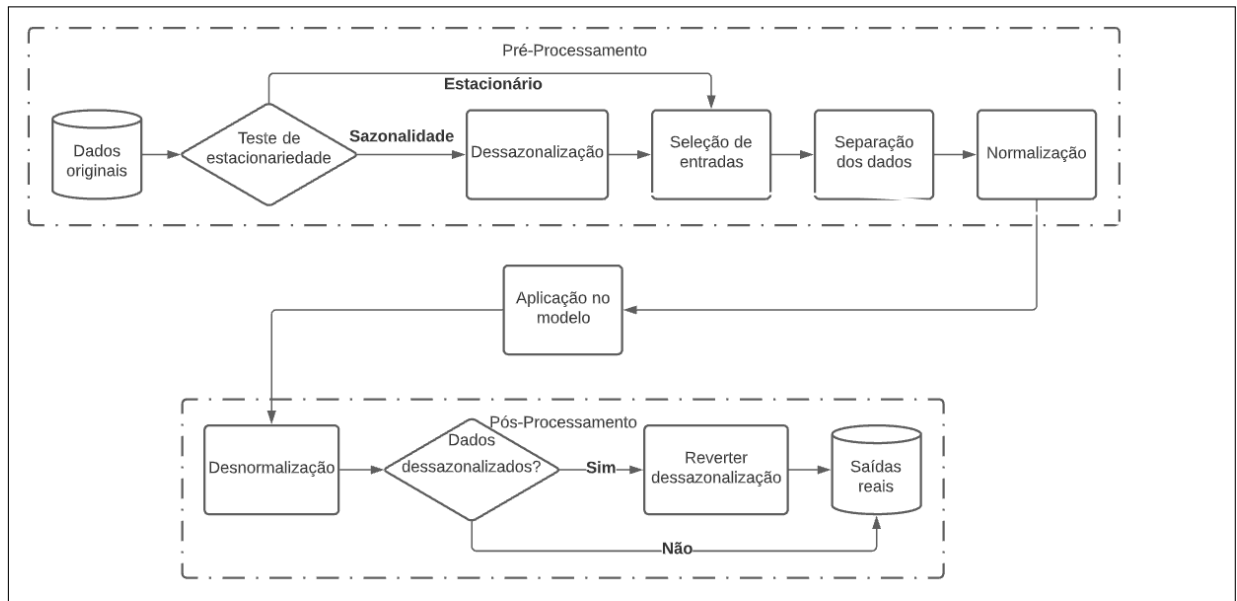
$$IA = 1 - \frac{\sum_{t=1}^N (\hat{x}_t - x_t)^2}{\sum_{t=1}^N (|\hat{x}_t - \bar{x}| + |x_t - \bar{x}|)^2}. \quad (34)$$

As etapas de previsão estão sumarizadas na Figura 16. Note que antes de servirem como entradas para os modelos preditivos, os dados passam por uma etapa de pré-processamento e, após aplicação nos modelos, as saídas passam pelo pós-processamento na sequência inversa.

O primeiro passo após a aquisição dos dados é a verificação de estacionariedade. Caso a série não seja estacionária é feita a dessazonalização, descrita na Seção 2.3. Porém nos estudos de caso abordados neste trabalho o teste ADF, Seção 2.2, rejeitou a hipótese das bases não serem estacionárias.

O segundo passo consiste na separação dos dados que é feita de maneira diferente para os Modelos Lineares e RNAs. No primeiro caso não há a necessidade de validação, portanto os dados são separados em 80% para treinamento e 20% para teste. Caso seja uma Rede Neural

Figura 16 – Pré e Pós-processamento dos dados



Fonte: Autoria própria (2022)

a separação feita é de: 60% para treinamento, 20% para validação e 20% para teste. A etapa de treinamento tem por objetivo ajustar os hiper-parâmetros¹ das RNAs, enquanto a validação evita o *overfitting* e permite a seleção do melhor número de neurônios na camada escondida.

Em seguida, é feita a seleção de entradas, na qual para os Modelos Lineares, a definição dos atrasos é dada através de busca exaustiva considerando até 11 valores anteriores.

Em relação às entradas das RNAs foi utilizada a Função de Auto-Correlação Parcial (PACF) para teste de 30 atrasos no qual a seleção ocorre da seguinte forma:

- Se houver menos de 10 atrasos significativos, todos são usados como entradas;
- Se houver 10 ou mais atrasos significativos, então os 10 mais significativos são usados como entradas.

Por fim, a última etapa do Pré-Processamento é a normalização dos dados, que neste caso consiste em dimensionar os valores dentro de um intervalo pré-definido, neste caso $[-1, 1]$, por conta da função de ativação escolhida para as RNAs, a Tangente Hiperbólica.

5.1 Modelos de Previsão Aplicados

O presente trabalho considerou 26 modelos distintos de previsão: 2 métodos lineares, 4 Redes Neurais Artificiais, 8 Sistemas Híbridos e 12 *Ensembles*. Estes foram aplicados ao problema de previsão da velocidade do vento de diferentes regiões do Brasil.

¹ Parâmetros ajustáveis que permitem controlar o processo de treinamento do modelo

Em relação aos Modelos Lineares, o modelo AR teve seus parâmetros estimados pelas equações de Yule-Walker, enquanto o modelo ARMA usou o método de máxima verossimilhança (BOX *et al.*, 2015).

No que diz respeito às RNAs e aos Sistemas Híbridos a quantidade de neurônios na camada intermediária testada variou de 50 até 150 com incremento de 10 neurônios. Note que foram testadas redes com apenas uma camada intermediária. Para função de ativação foi utilizada a Tangente Hiperbólica.

Para selecionar a melhor quantidade de neurônios, foram realizadas trinta rodadas sobre o conjunto de validação. A partir disto é calculada a média do MSE, e a quantidade de neurônios que apresentou menor MSE médio foi usada para o teste.

Os *Ensembles* aplicados neste estudo são caracterizados por serem não treináveis. Neste caso, os modelos que os compõem são treinados em paralelo conforme apresentado sobre os Modelos Lineares, RNAs e Sistemas Híbridos. Após as previsões de cada modelo é feita a combinação dos valores previstos por Média ou Mediana. Com isso, foram obtidas 6 configurações de *Ensembles*:

- ML: Modelos Lineares - AR e ARMA;
- RNAs: Redes Neurais Artificiais - MLP, RBF, ELM e ESN;
- RNAs sem RBF: Redes Neurais Artificiais desconsiderando RBF;
- MU: Modelos únicos, ou seja, Modelos Lineares e RNAs;
- MU sem RBF: Modelos únicos desconsiderando RBF;
- Todos: Modelos Únicos e Sistemas Híbridos.

Com o propósito de analisar se os erros gerados são estatisticamente diferentes entre si, foi escolhido o teste de hipótese de Wilcoxon. Neste, dado um conjunto de sinais, testa-se os valores a fim de rejeitar a hipótese nula o qual afirma que os dados seguem a mesma distribuição.

A saída do teste de Wilcoxon consiste em um número que indica a probabilidade da hipótese nula ser verdadeira. É comum interpretar os valores iguais ou menores que 0.05 (ou 5%), como hipótese nula rejeitada, ou seja, a distribuição das amostras é diferente. Com isso, garante-se que os valores obtidos pelas métricas de avaliação não são um acaso estatístico, ou seja, ao executar um novo experimento com a mesma configuração provavelmente haverá um desempenho semelhante.

A seguir, são apresentados os resultados obtidos na aplicação dos modelos acima. As bases utilizadas são amostradas por minuto e abrangem as datas entre 1 de Janeiro e 31 de Janeiro de 2019, totalizando 44640 amostras.

5.2 Brasília

A cidade de Brasília, capital federal do Brasil e sede da unidade federativa Distrito Federal, está situada na região Centro-Oeste e é uma das maiores cidades do país. Atualmente toda a energia elétrica consumida pela cidade não é gerada pelo Distrito Federal.

Diferente das demais bases, a série de Brasília continha 270 amostras faltantes que não foram preenchidas. Sua análise descritiva é apresentada na Tabela 1.

Tabela 1 – Brasília - Análise descritiva dos dados

	Quant. de amostras	Média	Desvio padrão	Mínimo	Máximo
Total	44370	1.0271	0.7146	0.0	5.099
Treinamento	26622	0.9724	0.714	0.0	4.724
Validação	8874	0.9726	0.6821	0.0	5.099
Teste	8874	1.2459	0.7064	0.0	4.296

Fonte: Autoria própria (2022)

A Tabela 2 mostra os erros de previsão sobre as simulações realizadas pelos modelos propostos para os dados de Brasília. Tais erros - MSE, MAE, ARV e IA - são calculados sobre os valores previstos após reversão do pré-processamento.

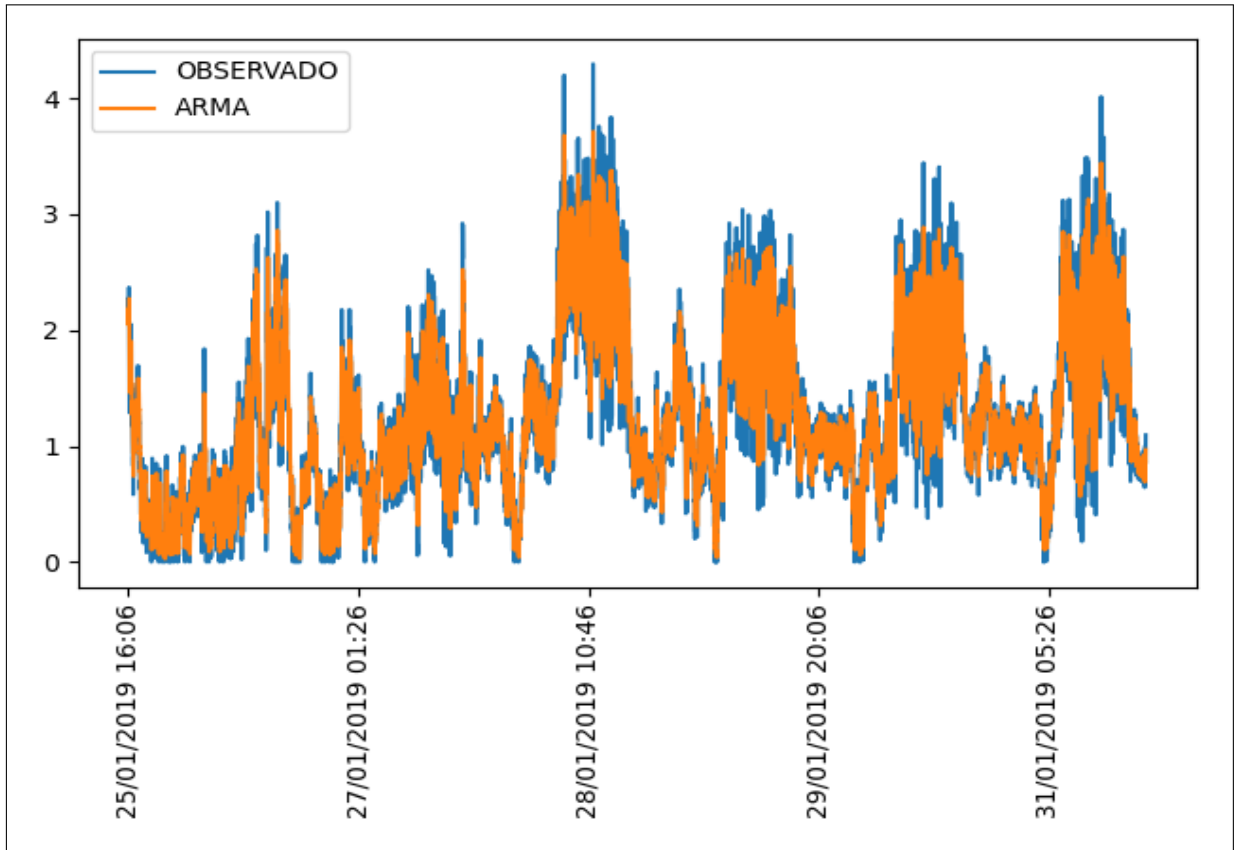
Tabela 2 – Brasília - Erros de previsão

	Modelo	MSE	MAE	ARV	IA
Únicos	AR	0.0761	0.1883	0.1930	0.9538
	ARMA	0.0752	0.1877	0.1901	0.9544
	MLP	0.0956	0.2315	0.3127	0.9373
	RBF	0.2750	0.4286	3.6968	0.6956
	ELM	0.0821	0.1981	0.1891	0.9543
	ESN	0.0810	0.1981	0.1851	0.9551
Híbridos	AR+MLP	0.1789	0.3628	0.3458	0.9056
	AR+RBF	0.1774	0.3589	0.3411	0.9067
	AR+ELM	0.0877	0.2069	0.1908	0.9524
	AR+ESN	0.0842	0.2028	0.1872	0.9539
	ARMA+MLP	0.1153	0.2692	0.2519	0.9366
	ARMA+RBF	0.2305	0.4276	0.4683	0.8734
	ARMA+ELM	0.0853	0.2054	0.1866	0.9537
	ARMA+ESN	0.0834	0.2028	0.1826	0.9547
Ensemble	Mediana (ML)	0.0816	0.1979	0.1913	0.9542
	Mediana (RNAs)	0.0844	0.2085	0.2385	0.9483
	Mediana (RNAs sem RBF)	0.0805	0.1968	0.1925	0.9544
	Mediana (MU)	0.0811	0.1973	0.1984	0.9535
	Mediana (MU sem RBF)	0.0809	0.1972	0.1930	0.9542
	Mediana (todos)	0.0831	0.2033	0.1898	0.9539
	Média (ML)	0.0816	0.1979	0.1913	0.9542
	Média (RNAs)	0.0972	0.2304	0.3511	0.9336
	Média (RNAs sem RBF)	0.0815	0.2018	0.2102	0.9521
	Média (MU)	0.0876	0.2126	0.2717	0.9440
	Média (MU sem RBF)	0.0810	0.1990	0.2009	0.9533
	Média (todos)	0.0922	0.2295	0.2361	0.9458

Fonte: Autoria própria (2022)

Através dos resultados mostrados nota-se que as métricas MSE e MAE apontam para ARMA como o melhor modelo preditivo, enquanto a métrica ARV aponta o ARMA+ESN e a IA o modelo ESN. O Gráfico 1 apresenta a melhor previsão apontada pelo MSE junto ao valor real observado. Devido a quantidade de amostras no conjunto de teste, foi elaborado o Gráfico 2 contendo apenas as 100 primeiras amostras.

Gráfico 1 – Brasília - Real x Melhor Previsão



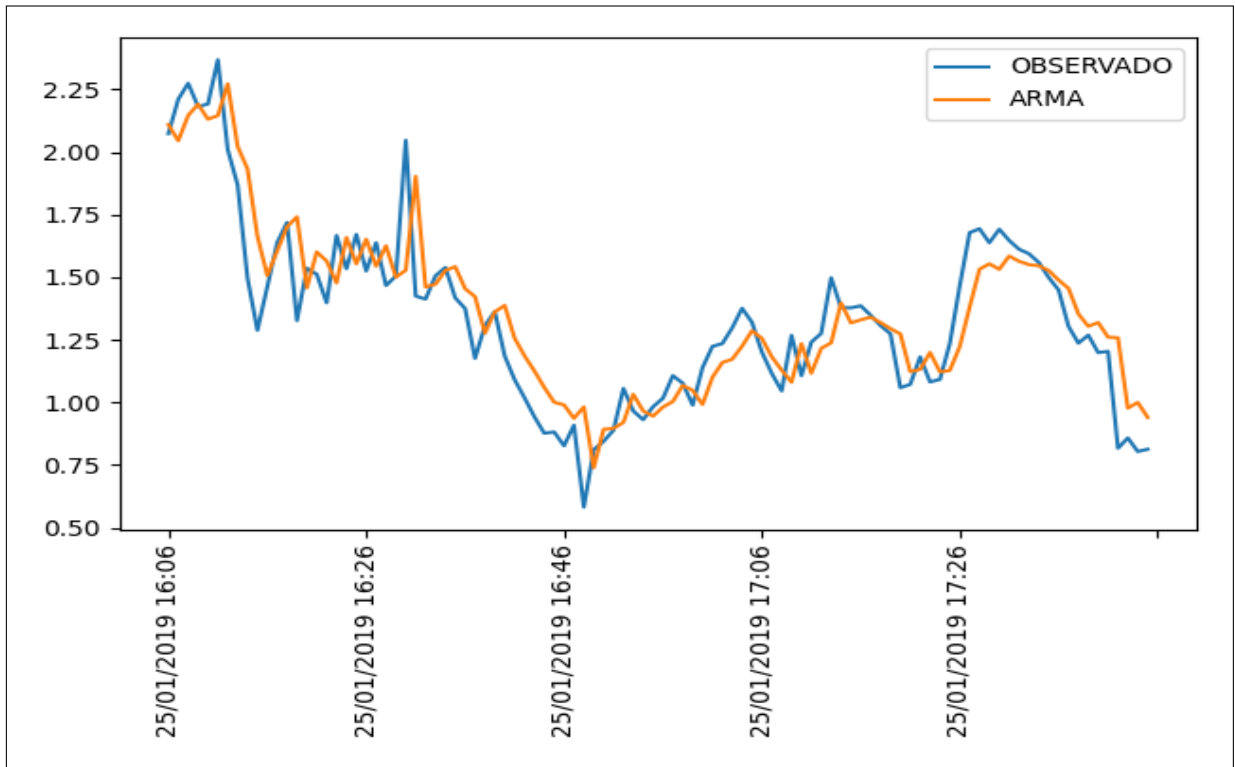
Fonte: Autoria própria (2022)

Ainda sobre a Tabela 2, em termos de MSE o melhor modelo híbrido (ARMA+ESN) e o melhor modelo *Ensemble* (Mediana (RNAs sem RBF)) que têm um erro 10,90% e 7,05% maiores em comparação com o modelo ARMA, respectivamente.

O Gráfico 3 apresenta o Boxplot dos MSEs calculados sobre as 30 rodadas do conjunto de teste e os modelos de previsão aplicados. Nota-se que a rede RBF além de apresentar os erros mais elevados, também apresenta maior dispersão.

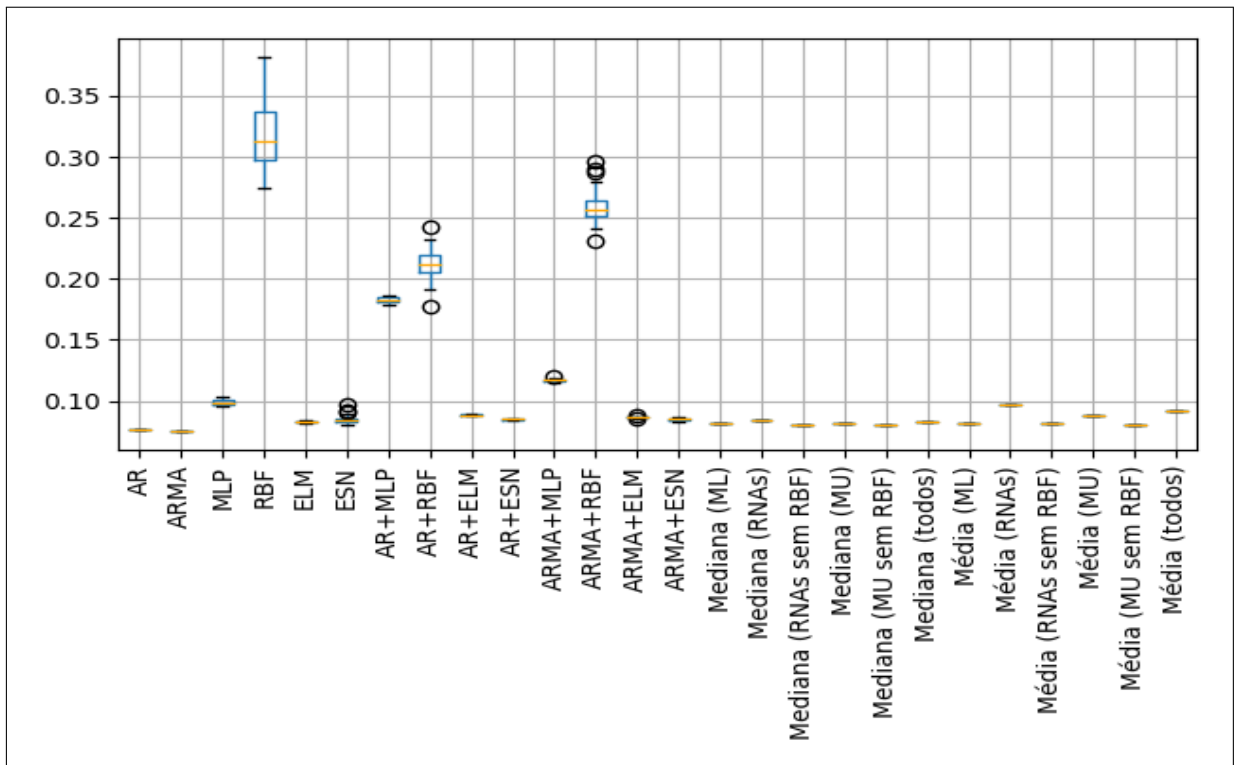
A dispersão de erro ocorre devido as 30 simulações de treinamento dos modelos neurais partirem de pesos inicializados aleatoriamente. Com isso, mesmo utilizando a mesma configuração (número de neurônios, mesma função de ativação) em um determinado modelo, seus experimentos apresentam resultados diferentes.

Gráfico 2 – Brasília - Real x Melhor Previsão (primeiras 100 amostras)



Fonte: Autoria própria (2022)

Gráfico 3 – Brasília - Boxplot dos erros de previsão



Fonte: Autoria própria (2022)

5.3 Florianópolis

Florianópolis encontra-se no Sul do Brasil, onde é a capital de Santa Catarina (SC). Tal estado fez altos investimentos nos anos 2018, 2019 e 2020 resultando em um crescimento de 179,49% na produção de energia eólica. Por outro lado em SC a produção de energia eólica e solar representam menos de 5% do total gerado pelo estado.

A Tabela 3 apresenta uma descrição dos dados eólicos de Florianópolis. Nela é possível notar que o conjunto Teste apresenta os menores valores para média, desvio padrão e valor máximo.

Tabela 3 – Florianópolis - Análise descritiva dos dados

	Quant. de amostras	Média	Desvio padrão	Mínimo	Máximo
Total	44640	2.1154	1.681	0.0	11.63
Treinamento	26784	2.2535	1.6861	0.0	11.63
Validação	8928	2.2759	1.7876	0.0	10.5
Teste	8928	1.5404	1.411	0.0	7.004

Fonte: Autoria própria (2022)

A Tabela 4 mostra os resultados das previsão feitas sobre o conjunto de teste dos dados eólicos de Florianópolis. Nela encontram-se os erros de acordo com as métricas MSE, MAE, ARV e IA. Em destaque, encontram-se os melhores valores indicando os melhores modelos preditivos.

Tabela 4 – Florianópolis - Erros de previsão

	Modelo	MSE	MAE	ARV	IA
Únicos	AR	0.3894	0.4406	0.1690	0.9599
	ARMA	0.3855	0.4385	0.1668	0.9604
	MLP	0.3198	0.4296	0.2366	0.9501
	RBF	0.8012	0.7259	1.8531	0.8048
	ELM	0.2851	0.3712	0.1619	0.9608
	ESN	0.2843	0.3683	0.1601	0.9611
Híbridos	AR+MLP	0.3871	0.4651	0.2191	0.9462
	AR+RBF	0.3212	0.4041	0.2277	0.9508
	AR+ELM	0.5640	0.5356	0.2885	0.9243
	AR+ESN	0.3035	0.3827	0.1609	0.9596
	ARMA+MLP	0.3912	0.4727	0.2177	0.9461
	ARMA+RBF	0.3183	0.3921	0.2144	0.9524
	ARMA+ELM	0.3338	0.4081	0.1705	0.9562
	ARMA+ESN	0.3089	0.3928	0.1602	0.9593
Ensemble	Mediana (ML)	0.2841	0.3754	0.1676	0.9602
	Mediana (RNAs)	0.2913	0.3952	0.1936	0.9568
	Mediana (RNAs sem RBF)	0.2821	0.3691	0.1669	0.9604
	Mediana (MU)	0.2825	0.3737	0.1711	0.9599
	Mediana (MU sem RBF)	0.2822	0.3724	0.1673	0.9604
	Mediana (todos)	0.2894	0.3721	0.1772	0.9587
	Média (ML)	0.2841	0.3754	0.1676	0.9602
	Média (RNAs)	0.3239	0.4317	0.2633	0.9474
	Média (RNAs sem RBF)	0.2844	0.3833	0.1760	0.9592
	Média (MU)	0.3008	0.4069	0.2187	0.9535
	Média (MU sem RBF)	0.2832	0.3794	0.1720	0.9598
	Média (todos)	0.2930	0.3767	0.1881	0.9572

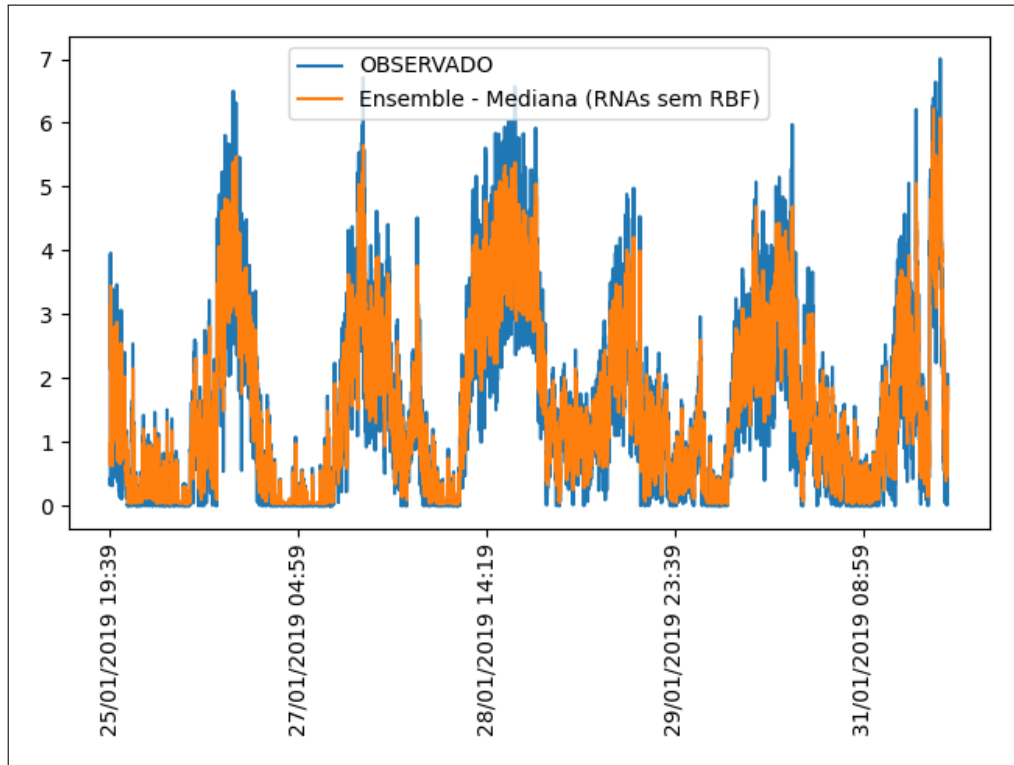
Fonte: Autoria própria (2022)

Ainda sobre a Tabela 4 nota-se que os melhores modelos foram: i) *Ensemble* (RNAs sem RBF), por MSE; ii) ESN de acordo com o MAE, ARV e IA. Na Seção 3.2.4 foi dito que as Redes Neurais com Estado de Eco têm como um dos diferenciais a memória intrínseca, de

modo que os valores previstos são resultados não apenas das entradas atuais, mas também das respostas anteriores. Tal característica pode ser um ponto favorável quando o método é aplicado em problemas de previsão de séries temporais.

Os Gráficos 4 e 5 apresentam uma comparação entre o valor real do conjunto de teste e o valor previsto pelo *Ensemble* (RNAs sem RBF), sendo o segundo Gráfico uma visualização das primeiras 100 amostras previstas.

Gráfico 4 – Florianópolis - Real x Melhor Previsão

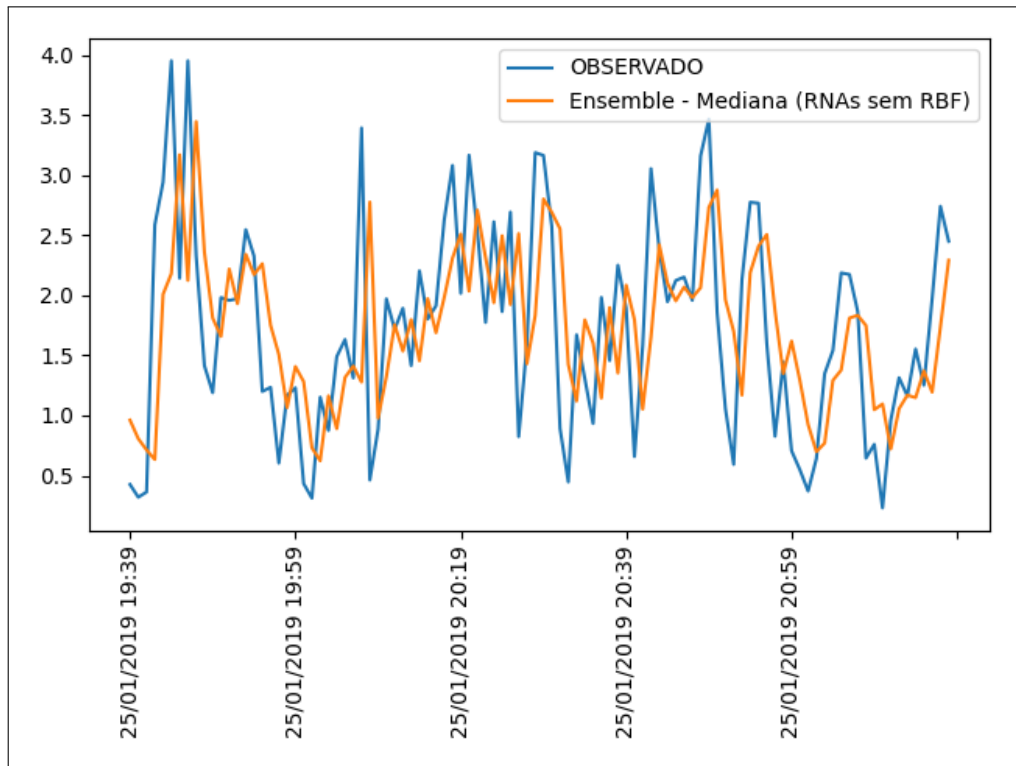


Fonte: Autoria própria (2022)

Ainda sobre a Tabela 4, vale ressaltar que diferente do caso anterior - Brasília - os modelos lineares (AR e ARMA) não se destacaram positivamente, sendo superados por todos os *Ensembles*. É possível que esta série apresente mais padrões não-lineares, que não pode ser modelada pelo AR ou ARMA.

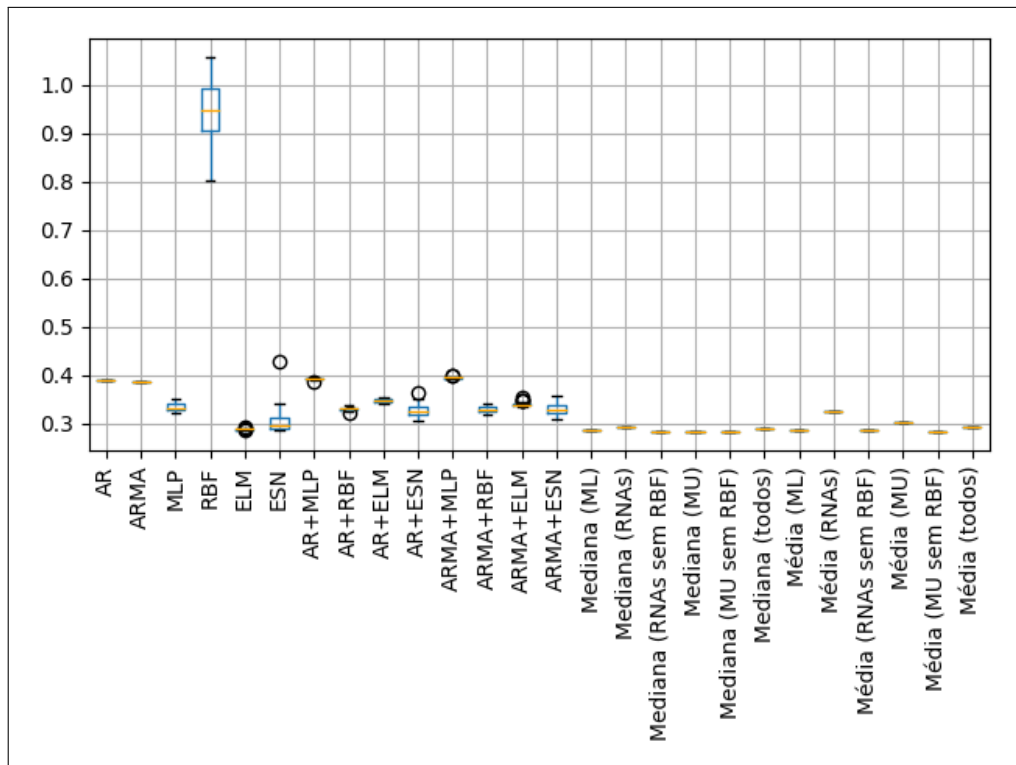
O Gráfico 6 apresenta um Boxplot referente às 30 rodadas de testes para a base de Florianópolis. Nota-se, mais uma vez, que o modelo RBF possui maiores MSEs e maior dispersão, enquanto a menor dispersão por modelo neural é dada pelo Híbrido AR+ELM.

Gráfico 5 – Florianópolis - Real x Melhor Previsão (primeiras 100 amostras)



Fonte: Autoria própria (2022)

Gráfico 6 – Florianópolis - Boxplot dos erros de previsão



Fonte: Autoria própria (2022)

5.4 Natal

A cidade de Natal está localizada no Rio Grande do Norte, estado brasileiro líder em produção de energia eólica. Sua série histórica é descrita estatisticamente na Tabela 5, na qual nota-se a diferença significativa para as demais na média, desvio padrão e valor máximo, considerando o conjunto de Treinamento.

Tabela 5 – Natal - Análise descritiva dos dados

	Quant. de amostras	Média	Desvio padrão	Mínimo	Máximo
Total	44640	5.9776	2.6976	1.274	87.4
Treinamento	26784	6.2882	3.2207	1.524	87.4
Validação	8928	5.3936	1.17	1.511	25.81
Teste	8928	5.6299	1.7735	1.274	32.1

Fonte: Autoria própria (2022)

Na Tabela 6 são mostrados os resultados das aplicações de modelos de previsão para os dados de Natal. É possível observar que o melhor resultado de previsão, tanto em MSE quanto em MAE, foi o ARMA, enquanto ESN foi apontado como melhor pelas métricas ARV e IA. Tal fato valida a afirmativa de que um mesmo modelo pode receber diferentes avaliações de seu desempenho dependendo da métrica aplicada, reforçando que cada métrica tende a destacar e penalizar pontos e características diferentes de cada saída.

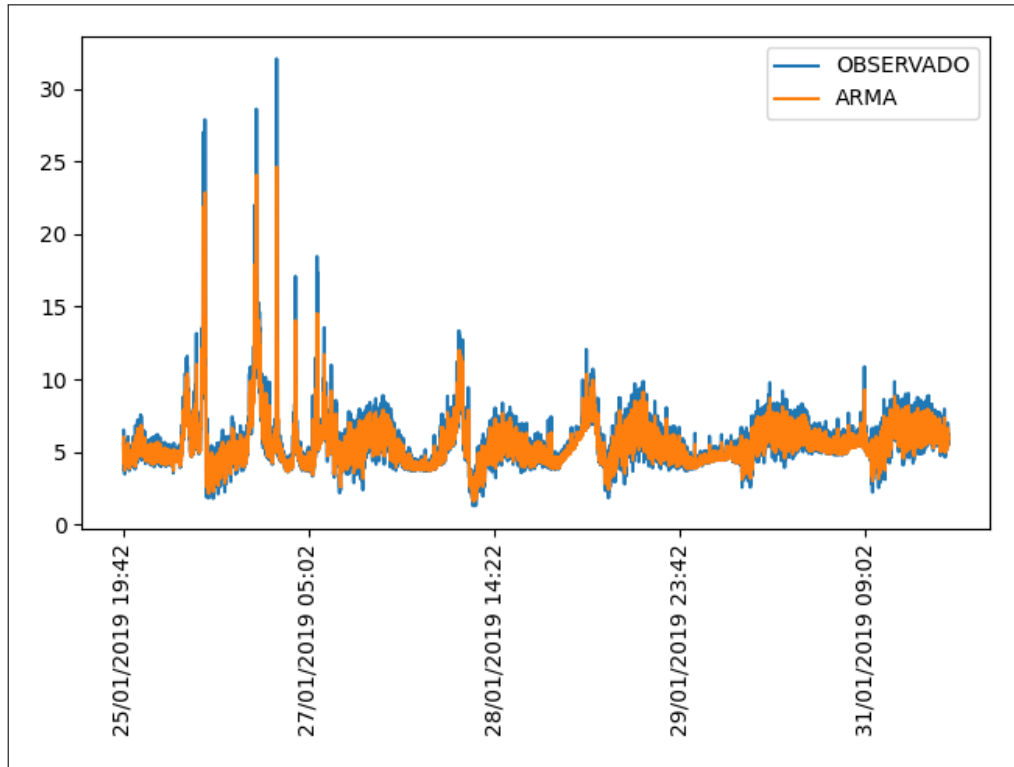
Tabela 6 – Natal - Erros de previsão

	Modelo	MSE	MAE	ARV	IA
Únicos	AR	0.8337	0.6115	0.3807	0.9102
	ARMA	0.8186	0.6059	0.3740	0.9120
	MLP	1.1366	0.7149	0.4887	0.8872
	RBF	1.4199	0.7937	0.5465	0.8637
	ELM	0.9909	0.6239	0.3915	0.9061
	ESN	1.0147	0.6293	0.3276	0.9130
Híbridos	AR+MLP	1.1094	0.7205	0.3917	0.9002
	AR+RBF	1.9326	1.1201	0.5198	0.8435
	AR+ELM	1.273	0.7698	0.4049	0.8905
	AR+ESN	1.1611	0.7137	0.3855	0.8983
	ARMA+MLP	1.2184	0.7845	0.4129	0.8920
	ARMA+RBF	2.1828	1.2218	0.5448	0.8289
	ARMA+ELM	1.2581	0.7208	0.3897	0.8932
	ARMA+ESN	1.1794	0.7281	0.3993	0.8957
Ensemble	Mediana (ML)	0.9316	0.6165	0.3768	0.9113
	Mediana (RNAs)	0.9461	0.6205	0.3957	0.9083
	Mediana (RNAs sem RBF)	0.961	0.6184	0.3916	0.9079
	Mediana (MU)	0.9212	0.6114	0.3815	0.9114
	Mediana (MU sem RBF)	0.9270	0.6139	0.3781	0.9114
	Mediana (todos)	1.0276	0.6752	0.3846	0.9052
	Média (ML)	0.9316	0.6165	0.3768	0.9113
	Média (RNAs)	0.9374	0.6358	0.3848	0.9100
	Média (RNAs sem RBF)	0.9491	0.6262	0.3716	0.9109
	Média (MU)	0.9199	0.6227	0.3782	0.9118
	Média (MU sem RBF)	0.9321	0.6191	0.3711	0.9119
	Média (todos)	1.0511	0.6997	0.3840	0.9040

Fonte: Autoria própria (2022)

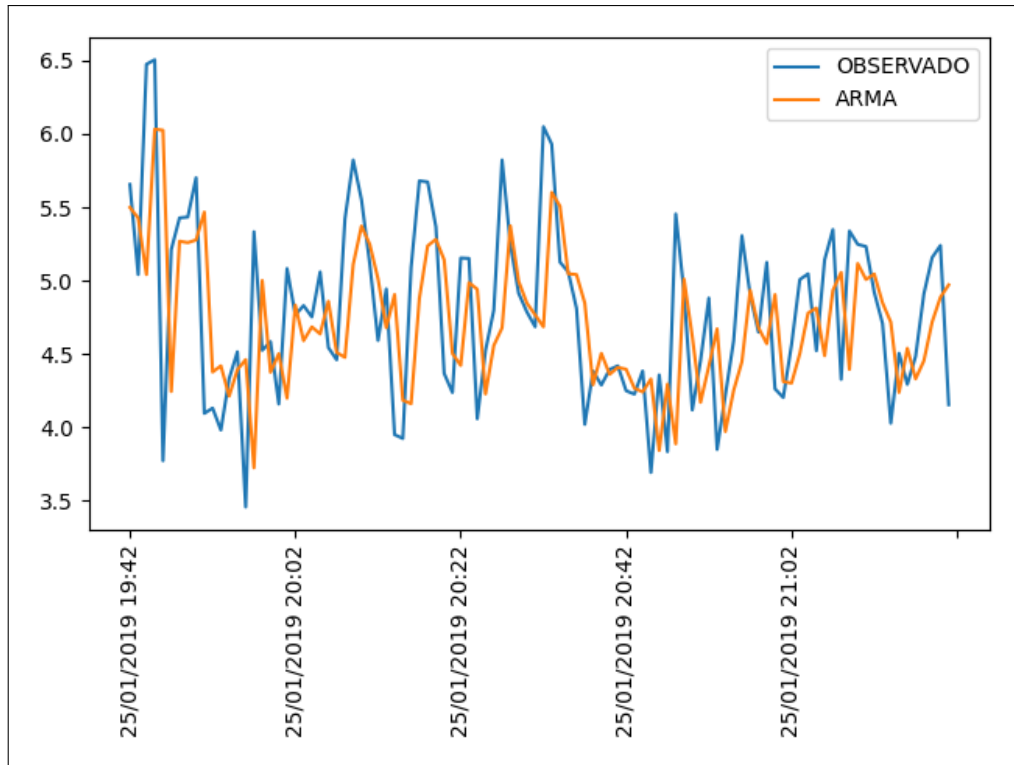
O Gráfico 7 apresenta o valor previsto pelo ARMA em comparação com os valores reais do conjunto de teste, e, no Gráfico 8, é mostrado os valores real e previsto de 100 amostras - as primeiras do conjunto de teste.

Gráfico 7 – Natal - Real x Melhor Previsão



Fonte: Autoria própria (2022)

Gráfico 8 – Natal - Real x Melhor Previsão (primeiras 100 amostras)

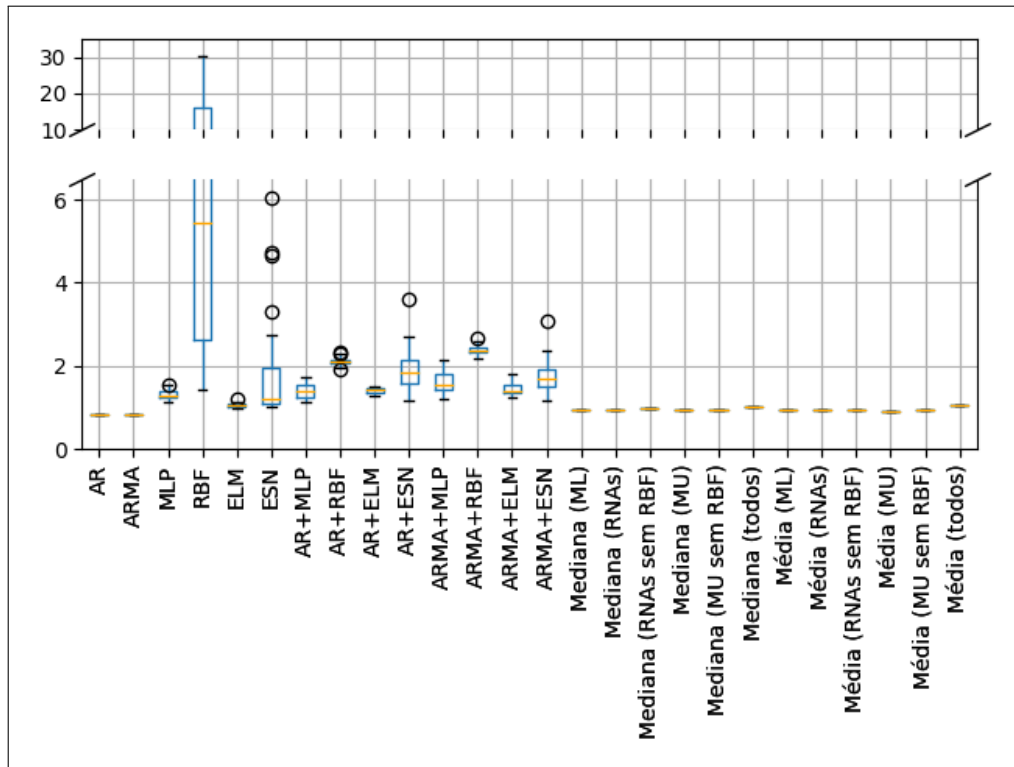


Fonte: Autoria própria (2022)

Através do Boxplot sobre as 30 rodadas de teste, Gráfico 9, nota-se que a ELM apresentou uma dispersão pequena no MSE em comparação com os demais. Observe que modelos

lineares como AR e ARMA são determinísticos em relação aos seus coeficientes e por isso não apresentam variação nas rodadas de testes.

Gráfico 9 – Natal - Boxplot dos erros de previsão



Fonte: Autoria própria (2022)

5.5 Petrolina

A cidade de Petrolina está localizada no estado de Pernambuco e se encontra próxima à divisa com Bahia, como mostra o Gráfico 15. Em relação aos dados históricos, apresenta um valor máximo no conjunto de Teste consideravelmente mais alto em comparação com os máximos dos conjuntos de Treinamento e Validação, como mostra a Tabela 7.

Tabela 7 – Petrolina - Análise descritiva dos dados

	Quant. de amostras	Média	Desvio padrão	Mínimo	Máximo
Total	44640	2.9464	1.3455	0.0	13.55
Treinamento	26784	2.8976	1.3316	0.0	8.38
Validação	8928	2.9142	1.3705	0.0	7.57
Teste	8928	3.125	1.3472	0.0	13.55

Fonte: Autoria própria (2022)

Tabela 8 – Petrolina - Erros de previsão

	Modelo	MSE	MAE	ARV	IA
Únicos	AR	0.5046	0.5355	0.3794	0.9142
	ARMA	0.4928	0.5261	0.3763	0.9155
	MLP	0.6993	0.6965	0.6117	0.8702
	RBF	1.2140	0.9416	1.2501	0.7493
	ELM	0.5079	0.5311	0.3290	0.9198
	ESN	0.5008	0.5286	0.3412	0.9190
Híbridos	AR+MLP	1.3293	1.0068	0.6175	0.8113
	AR+RBF	1.5169	1.0874	0.6616	0.7898
	AR+ELM	1.2609	0.9745	0.6490	0.8109
	AR+ESN	0.6584	0.6472	0.4248	0.8947
	ARMA+MLP	1.6166	1.1268	0.6619	0.7832
	ARMA+RBF	1.5457	1.0973	0.6542	0.7894
	ARMA+ELM	2.4195	1.1838	0.7502	0.7098
	ARMA+ESN	0.6574	0.6363	0.4021	0.8977
Ensemble	Mediana (ML)	0.4988	0.5311	0.3762	0.9154
	Mediana (RNAs)	0.5576	0.5907	0.4454	0.9024
	Mediana (RNAs sem RBF)	0.5071	0.5331	0.3376	0.9189
	Mediana (MU)	0.5035	0.5363	0.3654	0.9161
	Mediana (MU sem RBF)	0.5017	0.5327	0.3568	0.9172
	Mediana (todos)	0.8599	0.7862	0.6074	0.8530
	Média (ML)	0.4988	0.5311	0.3762	0.9154
	Média (RNAs)	0.5990	0.6244	0.5209	0.8906
	Média (RNAs sem RBF)	0.5250	0.5582	0.3917	0.9112
	Média (MU)	0.5472	0.5829	0.4598	0.9021
	Média (MU sem RBF)	0.5100	0.5445	0.3834	0.9135
	Média (todos)	0.8222	0.7581	0.5240	0.8672

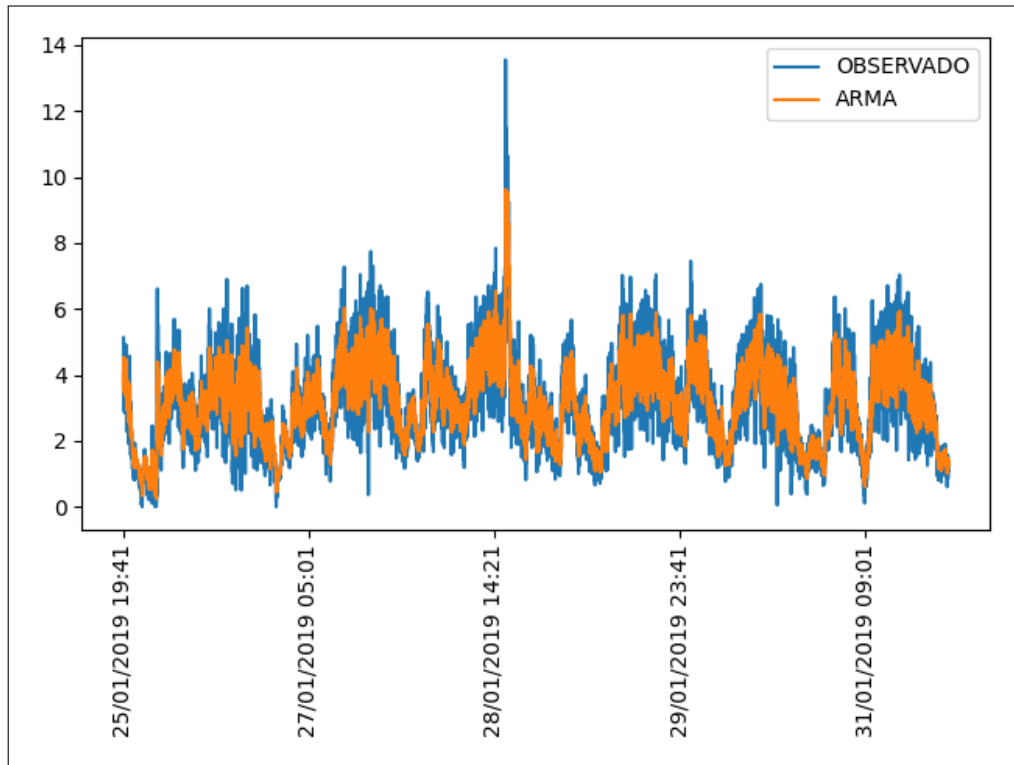
Fonte: Autoria própria (2022)

Na Tabela 8, são apresentados os erros de previsão onde o modelo ARMA apresentou menores valores de MSE e MAE, enquanto a rede neural ELM apresentou melhores valores de ARV e IA. Em segundo lugar, de acordo com MAE e IA tem-se a ESN. Para o MSE destacam-se os *Ensembles* com Mediana (ML) empatado com Média (ML), e, pelo ARV é a Mediana (RNAs sem RBF).

Os Gráficos 10 e 11 mostram as previsões do ARMA junto com os valores reais.

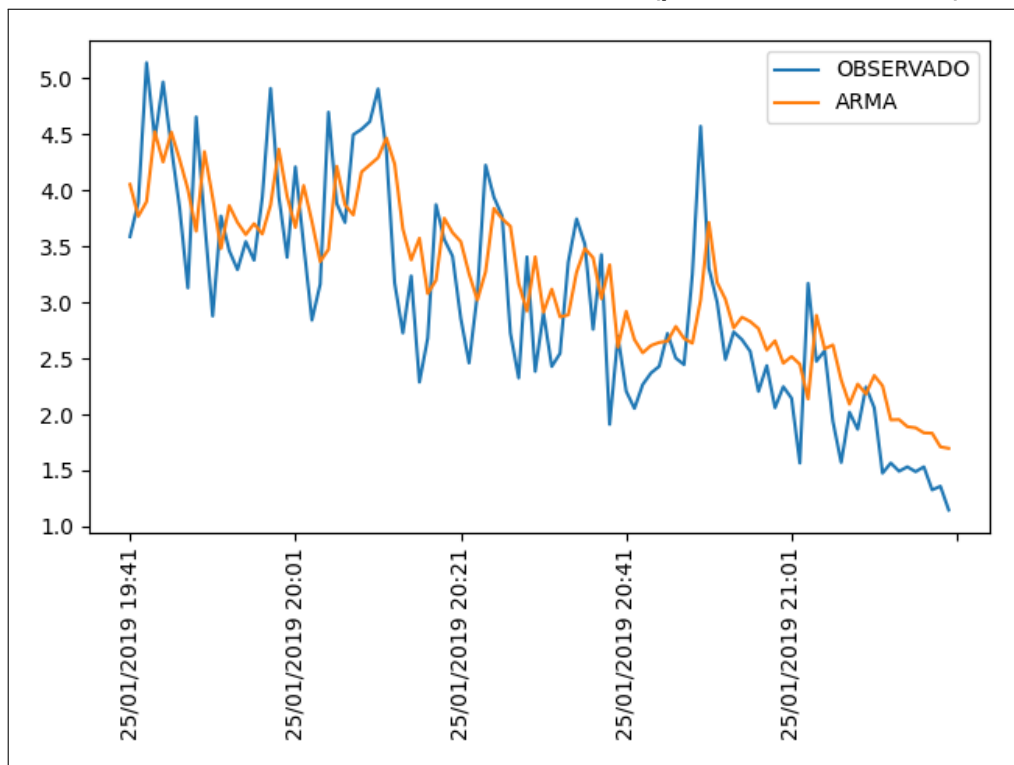
Por meio do Boxplot apresentado no Gráfico 12 nota-se que o modelo ARMA+ELM apresentou o maior valor de MSE bem como maior disparidade em comparação com os demais modelos. Por outro lado, o modelo que apresentou menor dispersão foi o Híbrido ARMA+MLP.

Gráfico 10 – Petrolina - Real x Melhor Previsão



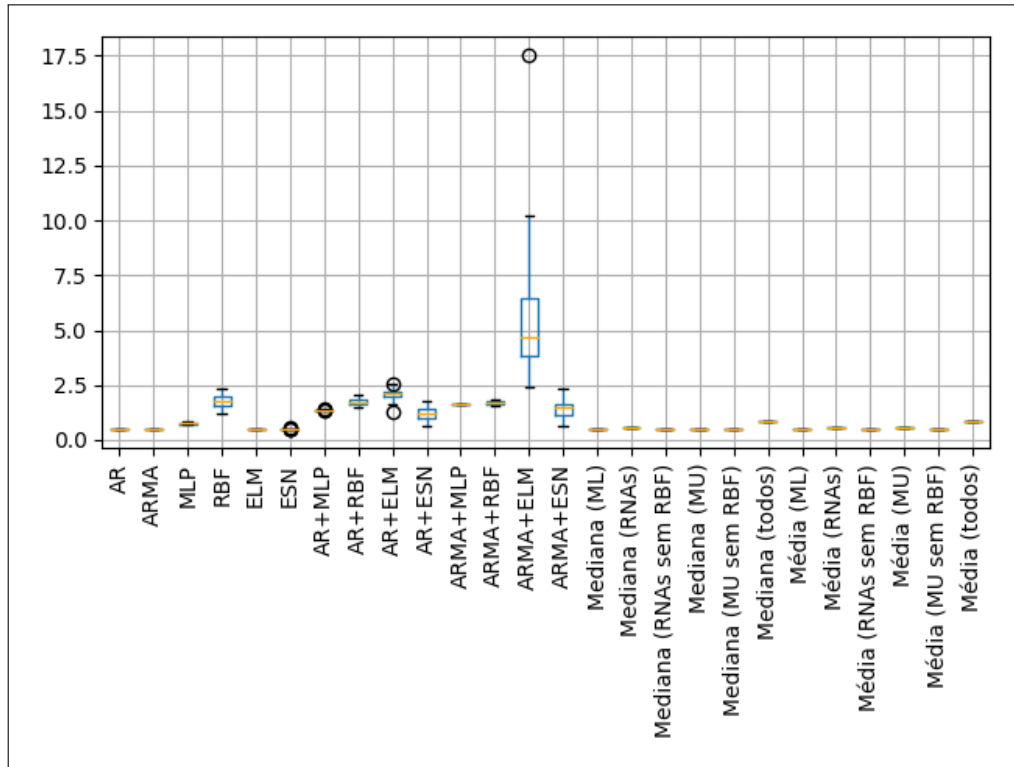
Fonte: Autoria própria (2022)

Gráfico 11 – Petrolina - Real x Melhor Previsão (primeiras 100 amostras)



Fonte: Autoria própria (2022)

Gráfico 12 – Petrolina - Boxplot dos erros de previsão



Fonte: Autoria própria (2022)

5.6 São Luís

São Luís é a capital do estado do Maranhão e está localizada no Nordeste, região brasileira com maior concentração de parques eólicos. As informações sobre quantidade de amostra, média, desvio padrão, valor mínimo e máximo são mostradas na Tabela 9.

Tabela 9 – São Luís - Análise descritiva dos dados

	Quant. de amostras	Média	Desvio padrão	Mínimo	Máximo
Total	44640	2.2264	1.4056	0.022	9.32
Treinamento	26784	2.118	1.4572	0.022	9.32
Validação	8928	2.5276	1.2386	0.034	7.61
Teste	8928	2.2501	1.3595	0.056	8.63

Fonte: Autoria própria (2022)

A Tabela 10 apresenta as avaliações das previsões para os dados de São Luís. Nela percebe-se que o *Ensemble* (RNAs sem RBF) apresentou o melhor resultado em termos de MSE e MAE cuja previsão do conjunto teste pode ser vista nos Gráficos 13 e 14, a qual apresenta as 100 primeiras amostras.

Tabela 10 – São Luís - Erros de previsão

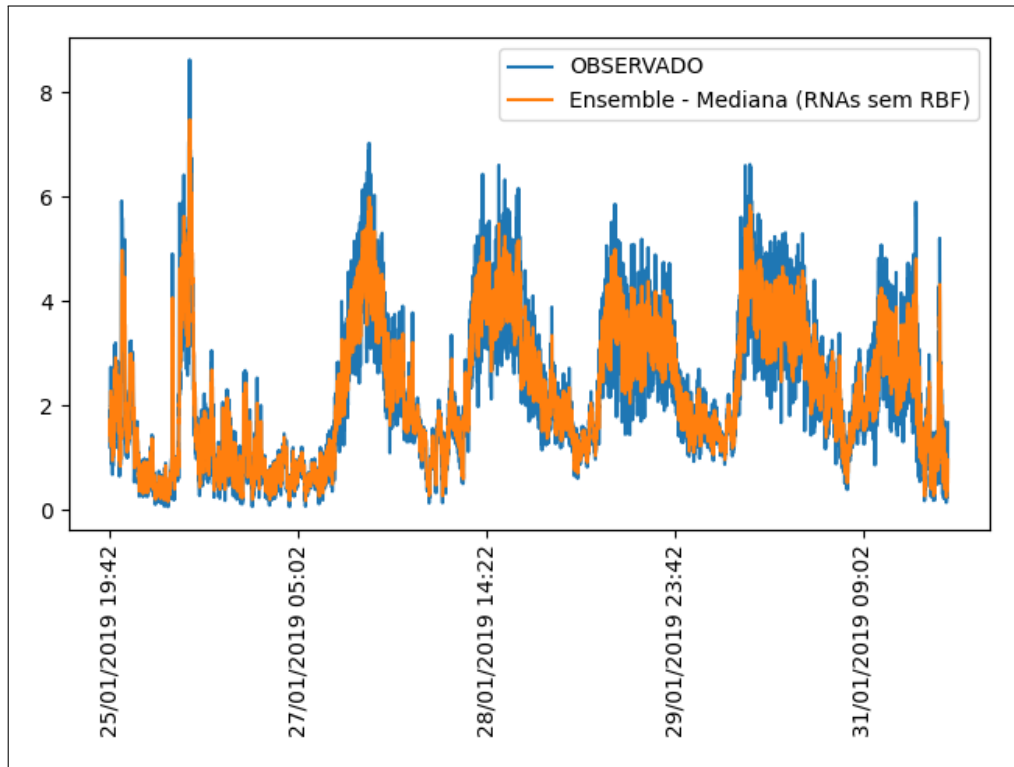
	Modelo	MSE	MAE	ARV	IA
Únicos	AR	0.2546	0.3598	0.1488	0.9641
	ARMA	0.2520	0.3591	0.1479	0.9644
	MLP	0.3279	0.4702	0.2855	0.9425
	RBF	1.3598	1.0418	1.3532	0.7195
	ELM	0.2394	0.3444	0.1432	0.9650
	ESN	0.2375	0.3432	0.1438	0.9651
Híbridos	AR+MLP	0.9181	0.8481	0.4008	0.8780
	AR+RBF	0.7429	0.7494	0.3299	0.9020
	AR+ELM	0.3068	0.3968	0.1661	0.9570
	AR+ESN	0.2795	0.3824	0.1549	0.9604
	ARMA+MLP	1.0505	0.9190	0.4320	0.8636
	ARMA+RBF	0.9885	0.8888	0.3986	0.8739
	ARMA+ELM	0.3153	0.4020	0.1717	0.9556
	ARMA+ESN	0.2695	0.3733	0.1510	0.9617
<i>Ensemble</i>	Mediana (ML)	0.2403	0.3454	0.1480	0.9643
	Mediana (RNAs)	0.2597	0.3900	0.1903	0.9581
	Mediana (RNAs sem RBF)	0.2374	0.3431	0.1471	0.9647
	Mediana (MU)	0.2380	0.3444	0.1496	0.9643
	Mediana (MU sem RBF)	0.2378	0.3438	0.1474	0.9646
	Mediana (todos)	0.2462	0.3503	0.1530	0.9633
	Média (ML)	0.2403	0.3454	0.1480	0.9643
	Média (RNAs)	0.3451	0.4890	0.2942	0.9399
	Média (RNAs sem RBF)	0.2458	0.3660	0.1675	0.9617
	Média (MU)	0.2848	0.4235	0.2195	0.9529
Média (MU sem RBF)	0.2410	0.3536	0.1578	0.9632	
Média (todos)	0.2719	0.3680	0.1751	0.9586	

Fonte: Autoria própria (2022)

Conforme dito anteriormente, cada modelo proposto foi executado 30 vezes a fim de verificar os resultados obtidos. Dito isso, o Gráfico 15 mostra a dispersão dos resultados sobre o conjunto de teste da base de São Luís.

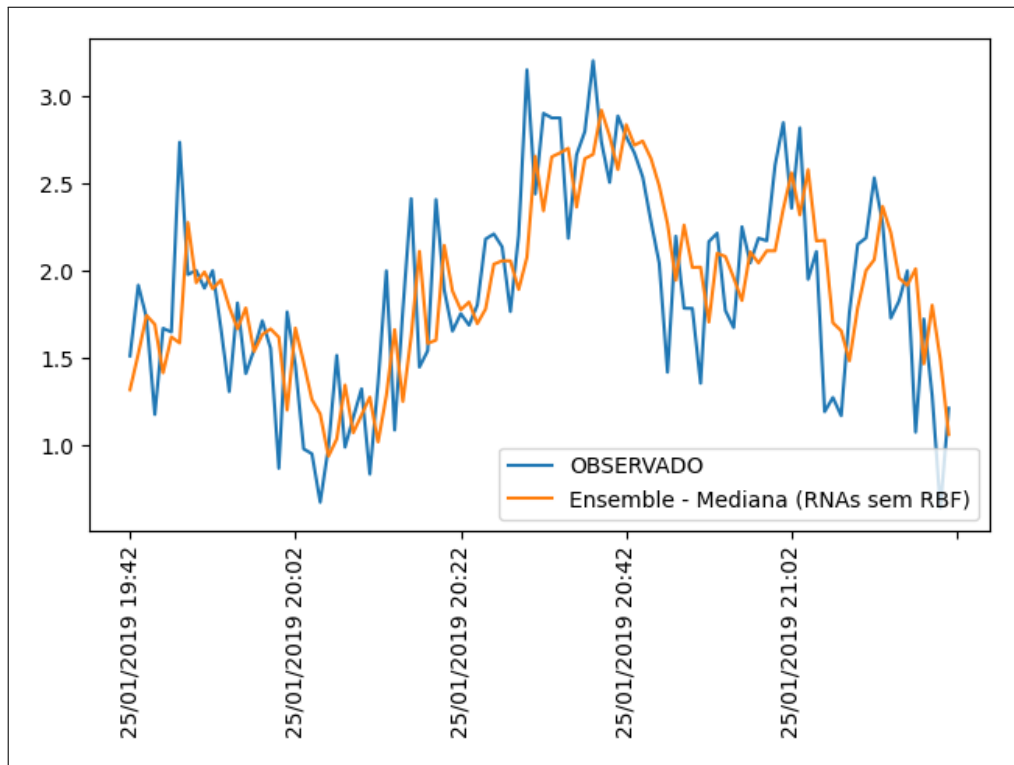
Por meio do Gráfico 15 é possível notar que neste estudo de caso a rede neural RBF e híbridos associados apresentaram as maiores dispersões para a previsão dos dados de São Luís.

Gráfico 13 – São Luís - Real x Melhor Previsão



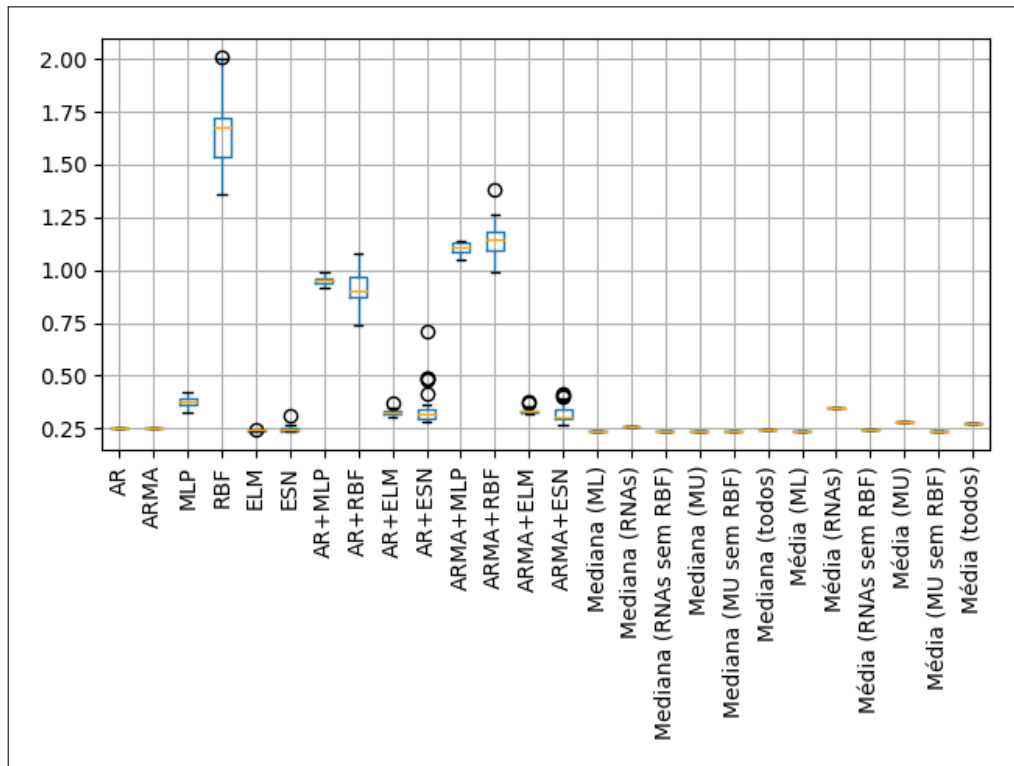
Fonte: Autoria própria (2022)

Gráfico 14 – São Luís - Real x Melhor Previsão (primeiras 100 amostras)



Fonte: Autoria própria (2022)

Gráfico 15 – São Luís - Boxplot dos erros de previsão

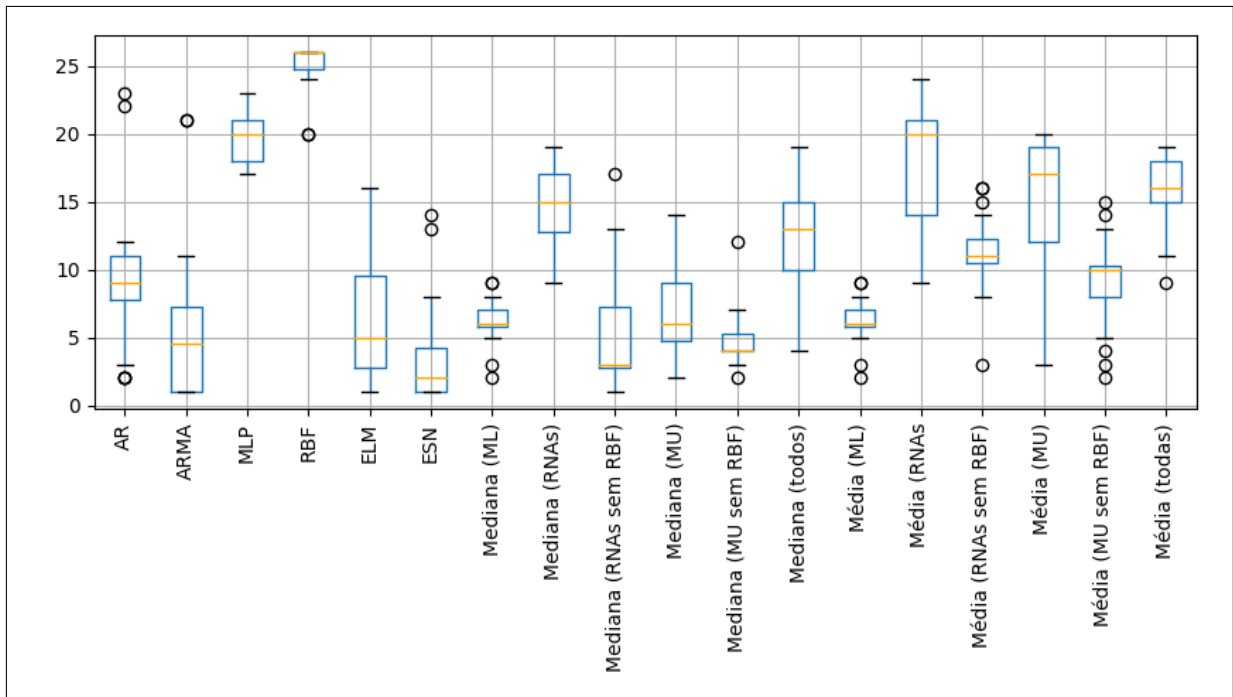


Fonte: Autoria própria (2022)

5.7 Análise Dos Resultados

Esta Seção tem por objetivo analisar os resultados apresentados sobre as bases de dados escolhidas. Para verificar e comparar os modelos de previsão propostos foram elaboradas a Tabela 11 e o Gráfico 16, que apresentam o ranking de classificação que os modelos receberam para cada base de dados, de acordo com as métricas de avaliação propostas - MSE, MAE, ARV e IA.

Gráfico 16 – Boxplot das classificações dos modelos sobre os 5 estudos de casos e as 4 métricas propostas



Fonte: Autoria própria (2022)

Através da Tabela 11 verifica-se que os três melhores modelos preditivos - coluna Total/Rank - foram ESN, *Ensemble* de Mediana (MU sem RBF), *Ensemble* de Mediana (RNAs sem RBF) e ARMA, nesta sequência.

Como dito anteriormente, a ESN apresenta uma memória intrínseca que pode ser uma vantagem para tarefas de previsão de séries temporais. É interessante destacar que para a base Natal esta RNA teve o melhor desempenho de acordo com as métricas ARV e IA, mas ficou em 14º e 13º lugar pelas métricas MSE e MAE. Isso ocorre pelo fato de cada métrica penalizar uma determinada característica. Neste caso, assume-se que a previsão apresentou poucos pontos com grande diferença entre real e previsto. Por outro lado, é possível que as amostras previstas apresentassem um ruído constante ao longo das amostras. Tal variação pode ser observada pelo Gráfico 16.

Em relação aos *Ensembles*, constata-se que o melhor combinador foi o de mediana pois ao combinar as RNAs, RNAs sem RBF, MU, MU sem RBF e todos os modelos percebe-se pelo

Gráfico 16 do Boxplot que os desempenhos são melhores quando comparados às suas versões combinadas por Média.

Os *Ensembles* Mediana (ML) e Média (ML) apresentaram o mesmo resultado. Isso deve ao fato da mediana, neste caso, ser a média dos modelos AR e ARMA, sendo estatisticamente iguais. Em relação aos demais modelos, foi comprovado pelo teste de Wilcoxon que o melhor preditor de cada cidade é estatisticamente diferente dos demais.

Outra observação sobre os *Ensembles* pode ser feita em relação ao modelo RBF já que sua exclusão favoreceu as previsões. Tal informação é confirmada ao comparar o *Ensemble* de Mediana (MU sem RBF) com o *Ensemble* de Mediana (MU), que passou de 2º para 8º lugar, e, o *Ensemble* de Mediana (RNAs sem RBF) com o *Ensemble* de Mediana (RNAs) que foi de 3º para 15º lugar.

Para finalizar a análise dos *Ensembles*, é importante destacar que deste conjunto os seguintes modelos apresentaram pouca variação na avaliação de desempenho: Mediana (ML), Mediana (MU sem RBF), Média (ML) e Média (RNAs sem RBF). Tal fato confirma o poder de generalização da proposta, mesmo quando aplicada em diferentes padrões de dados.

Sobre o modelo ARMA, seu destaque positivo concentrou-se em alguns casos específicos - Brasília, Natal e Petrolina pelas métricas MSE e MAE. Entretanto, nos demais casos e métricas, sua classificação oscilou entre 2º e 21º lugar, como mostra o Gráfico 16. Dessa forma, torna-se compreensível sua classificação final em 4º lugar.

Em relação aos modelos ARMA e ESN, por outro lado, apesar de serem destaque em vários casos, evidencia-se uma dispersão relativamente grande ocupando do 1º ao 21º lugares e do 1ª ao 14º lugares, respectivamente. Os *Ensembles* de Média (ML) e Mediana (ML) variaram entre o 2º até 9º lugares, enquanto a Mediana (RNAs) oscilou entre o 9 e o 19º lugares. Na classificação final, os melhores *Ensembles* detiveram o 2º, 3º, e 5º lugares.

É interessante observar que os SHCE não foram destaque positivo, sendo a melhor classificação geral desse grupo o 13º lugar, pelo ARMA+ESN. Neste caso, não é possível afirmar a melhoria dos resultados pela hibridização, uma vez que os modelos lineares AR e ARMA ora saíram melhor, ora saíram pior que suas versões híbridas.

Além dos *Ensembles*, a RBF apresentou pouca variação de desempenho, mas este não alcançou boa performance. Ao observar o Gráfico 16 fica claro que a melhor avaliação recebida foi de 20º lugar.

Também vale notar que a MLP e a RBF, apesar de serem duas RNAs clássicas, não foram destaque positivo em nenhum momento. Com isso, a classificação final destes modelos foi de 20º e 26º lugar, respectivamente. Esse mau desempenho também aparece nos SHCE relacionados, sendo eles: AR+MLP, AR+RBF, ARMA+RBF e ARMA+MLP, classificados em 22º à 25º lugar, nessa ordem.

Por fim, vale destacar que os desempenhos dos modelos variam conforme o problema abordado, o que implica em conclusões distintas sobre o melhor preditor. Por exemplo, para o caso de São Luís, a ESN tem média no ranking de 2,25, já que foi o melhor para IA e o 2º melhor

para as demais métricas. Já o *Ensembles* (Mediana - MU sem RBF) apresenta média 2,0, tendo ganhado no MSE, MAE e sendo o 3º para ARV e IA. Logo, este último deveria ser adotado para este caso. Dessa maneira, a recomendação para cada série seria como abaixo:

- Brasília: ARMA;
- Florianópolis: ESN;
- Natal: ARMA;
- Petrolina: ESN;
- São Luís: *Ensembles* (Mediana - MU sem RBF).

O caso de Petrolina é interessante, pois o indicativo de usar a ESN considera um modelo que não vence para nenhuma métrica, mas que está entre os primeiros ranqueados para todas. A ELM foi superior para ARV e IA, mas seu comportamento no MSE piorou sua posição. Neste caso, o modelo não conseguiu captar os pontos extremos da série. A mesma observação vale para Natal, na qual a ESN se destaca para ARV e IA, mas tem MSE e MAE longe dos melhores.

Entretanto, a medida que a quantidade de geradores aumenta ou então diferentes regiões são observadas pode ser inviável calibrar um modelo específico para cada parque eólico. Isto ocorre no caso de previsão de vazões a usinas hidrelétricas, em que o um modelo baseado no AR é utilizado para todo o sistema (SIQUEIRA *et al.*, 2013). Neste caso, o uso da ESN se mostra adequado já que o modelo na média apresentou resultados adequados para todas as séries.

Estas observações mostram a dificuldade da tarefa, dado seu comportamento intermitente de forma que nenhum modelo possa ser o melhor em todos os casos.

Tabela 11 – Ranking de performance

Modelo	Brasília				Florianópolis				Natal				Petrolina				São Luís				Total		
	MSE	MAE	ARV	IA	MSE	MAE	ARV	IA	MSE	MAE	ARV	IA	MSE	MAE	ARV	IA	MSE	MAE	ARV	IA	Média	Rank	
Únicos	AR	2	2	12	11	23	22	10	8	2	3	9	10	7	8	9	9	12	11	8	9	9.35	10
	ARMA	1	1	7	3	21	21	5	3	1	1	4	2	1	1	8	6	11	10	5	5	5.85	4
	MLP	20	21	21	20	17	19	23	21	18	18	23	23	17	17	19	17	20	20	20	20	19.7	20
	RBF	26	26	26	26	26	26	26	26	24	24	26	24	20	20	26	25	26	26	26	26	25.05	26
	ELM	11	8	5	5	9	3	4	2	13	11	16	14	9	3	1	1	5	4	1	2	6.35	7
	ESN	5	8	2	1	7	1	1	1	14	13	1	1	4	2	3	2	2	2	2	1	3.65	1
Híbridos	AR+MLP	24	24	23	24	22	23	21	23	17	19	18	17	22	22	20	20	23	23	24	23	21.6	22
	AR+RBF	23	23	22	23	18	16	22	20	25	25	24	25	23	23	23	22	22	22	22	22	22.25	23
	AR+ELM	18	16	8	15	25	25	25	25	23	22	21	22	21	21	21	21	18	17	14	17	19.75	21
	AR+ESN	14	12	4	9	14	11	3	11	19	17	14	18	16	16	13	15	16	15	12	14	13.15	14
	ARMA+MLP	22	22	19	21	24	24	19	24	21	23	22	21	25	25	24	24	25	25	25	25	23	25
	ARMA+RBF	25	25	25	25	16	13	18	19	26	26	25	26	24	24	22	23	24	24	23	24	22.85	24
	ARMA+ELM	16	15	3	12	20	18	11	17	22	20	15	20	26	26	25	26	19	18	16	18	18.15	18
	ARMA+ESN	13	12	1	2	15	14	2	12	20	21	20	19	15	15	12	14	14	14	10	12	12.85	13
Ensemble	Mediana (ML)	9	6	9	6	5	7	8	6	6	5	5	7	2	3	6	7	6	6	6	6	6.05	5
	Mediana (RNAs)	15	17	18	17	11	15	17	16	10	9	19	12	13	13	14	12	13	16	18	16	14.55	15
	Mediana (RNAs sem RBF)	3	3	11	3	1	2	6	3	12	7	17	13	8	7	2	3	1	1	3	3	5.45	3
	Mediana (MU)	7	5	14	13	3	6	12	8	4	2	10	5	6	9	5	5	4	4	9	6	6.85	8
	Mediana (MU sem RBF)	4	4	12	6	2	5	7	3	5	4	7	5	5	6	4	4	3	3	4	4	4.85	2
	Mediana (todos)	12	14	6	9	10	4	15	14	15	15	12	15	19	19	18	19	10	8	11	10	12.75	12
	Média (ML)	9	6	9	6	5	7	8	6	6	5	5	7	2	3	6	7	6	6	6	6	6.05	5
	Média (RNAs)	21	20	24	22	19	20	24	22	9	14	13	11	14	14	16	16	21	21	21	21	18.15	18
	Média (RNAs sem RBF)	8	11	16	16	8	12	14	13	11	12	3	9	11	11	11	11	9	12	15	12	11.25	11
	Média (MU)	17	18	20	19	13	17	20	18	3	10	8	4	12	12	15	13	17	19	19	19	14.65	16
	Média (MU sem RBF)	5	10	15	14	4	10	13	10	8	8	2	3	10	10	10	10	8	9	13	11	9.15	9
Média (todos)	19	19	17	18	12	9	16	15	16	16	11	16	18	18	17	18	15	13	17	15	15.75	17	

Fonte: Autoria própria (2022)

5.8 Quantidade de neurônios

Como dito na Seção 5.1, foram feitos testes em relação à melhor quantidade de neurônios sobre as RNAs e os Sistemas Híbridos, com valores entre 50 e 150, e incremento de 10 neurônios. A Tabela 12 apresenta a quantidade de neurônios utilizados por modelo e base de dados.

Tabela 12 – Quantidade de neurônios por modelo e base de dados

Modelo	Brasília	Florianópolis	Natal	Petrolina	São Luís	Média
ELM	110	70	50	70	50	70
AR+ELM	70	120	70	70	70	80
ARMA+ELM	11	140	60	50	50	62.2
MLP	50	50	140	50	50	68
AR+MLP	50	50	110	130	70	82
ARMA+MLP	50	50	120	140	50	82
ESN	50	60	50	50	60	54
AR+ESN	80	130	90	90	70	92
ARMA+ESN	60	140	60	50	60	74
RBF	50	50	140	50	50	68
AR+RBF	50	50	80	50	60	58
ARMA+RBF	50	50	80	50	50	56
Média	56.75	80	87.5	70.83	57.5	

Fonte: Autoria própria (2022)

Ao analisar as médias de neurônios utilizados para cada base dados (Tabela 12), São Luís e Brasília apresentaram os menos valores, de modo que a primeira base foi a única com quantidade de neurônios com pouca variação, 50 a 70. Por outro lado, Natal foi a base que apresentou maior média. Este fato pode estar relacionado ao desvio padrão do conjunto de treinamento da base Natal (Tabela 5), uma vez que um desvio padrão alto indica que os dados são mais variáveis e mais difíceis de serem previstos.

Em relação aos modelos, a rede ESN apresentou a menor média de neurônios utilizados, lembrando que esta apresentou melhor desempenho geral, conforme mostrado na Seção 5.7.

O Gráfico 17 foi criada a partir dos dados da Tabela 12 com o intuito de gerar uma visualização sobre a frequência do uso de cada quantidade de neurônios. Através dela torna-se mais visível os casos com a quantidade mínima de neurônios - 50, representando 46,7%; seguido de 13,3% com 70 neurônios e 11,7% com 60 neurônios. Esses dados sinalizam que o problema de previsão da velocidade do vento, com as mesmas condições apresentadas, não necessita de um grande número de unidades de processamento.

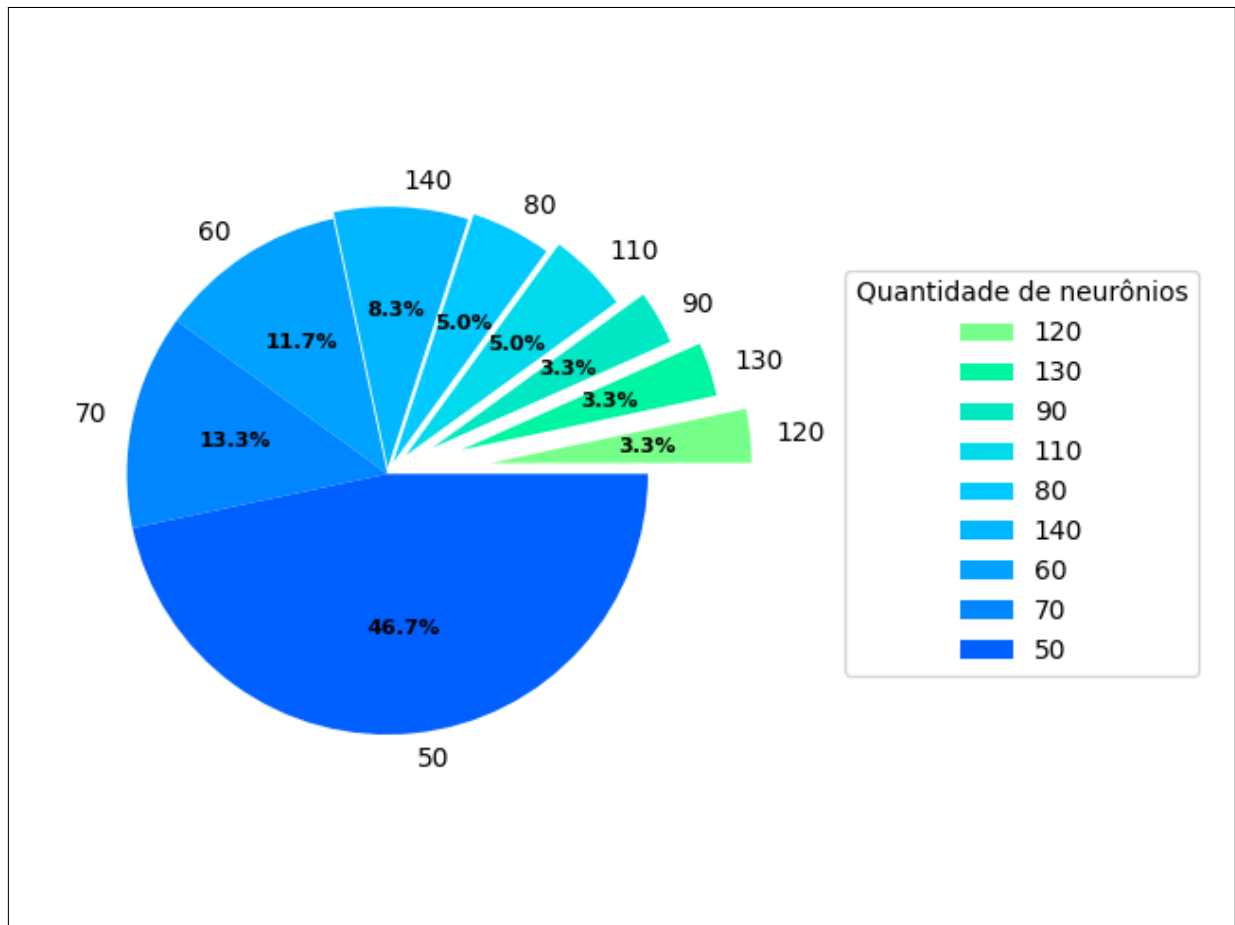
Em 19,9% dos casos foi utilizado 100 ou mais neurônios, sendo eles:

- Brasília e o modelo ELM;
- Florianópolis e os Sistemas Híbridos: AR+ELM, ARMA+ELM, AR+ESN e ARMA+ESN;
- Natal e os modelos: MLP, AR+MLP, ARMA+MLP e RBF;
- Petrolina e os Sistemas Híbridos: AR+MLP e ARMA+MLP.

Uma suposição comumente feita acerca das RNAs é sobre a quantidade de neurônios utilizada ser proporcional ao desempenho, mas observando os Sistemas Híbridos, fica evidente que isso não é uma regra, pois estes apresentaram altas quantidades de neurônios e não foram os melhores (Seção 5.7).

Por fim, é válido afirmar que 70 neurônios é um limite superior suficiente para teste, visto que 71,7% dos casos utilizou uma quantidade menor ou igual esse valor.

Gráfico 17 – Visualização da quantidade de neurônios utilizados



Fonte: Autoria própria (2022)

5.9 Atrasos

Na Seção 4.1 foi dito que para problemas de previsão de séries temporais a tarefa de seleção de entradas é muito importante pois está ligada ao desempenho dos modelos.

Para os Modelos Lineares - AR e ARMA - a seleção de atrasos foi feita por meio de uma busca exaustiva considerando os atrasos de 1 a 11. Em relação às RNAs e aos Sistemas Híbridos a proposta inicial era aplicar o método *Wrapper* a fim de obter as melhores entradas. Entretanto, a dimensão de cada base de dados com aproximadamente 40 mil amostras inviabilizou sua aplicação. Para tanto, a PACF, discutida nas Seções 2.2 e 4.1, foi empregada. Por se tratar de um seletor de entradas do tipo filtro, sua aplicação é mais rápida, sendo adequada para bases grandes. Isto converge com as indicações da literatura (GUYON; ELISSEEFF, 2003; SIQUEIRA *et al.*, 2013).

Vale destacar que neste trabalho não houve seleção de entradas para os *Ensembles* uma vez que as entradas destes modelos são as saídas de RNAs e modelos lineares. Dito isso, a aplicação do método *Wrapper* seria a mais adequada para os *Ensembles*.

É importante lembrar que para seleção de entradas foram testados 30 atrasos no qual a seleção ocorre da seguinte forma:

- Se houver menos de 10 atrasos significativos, todos são usados como entradas;
- Se houver 10 ou mais atrasos significativos, então os 10 mais significativos são usados como entradas.

Na Tabela 13 são apresentados os atrasos selecionados pela PACF para cada base de dados e conjunto de modelo preditivo.

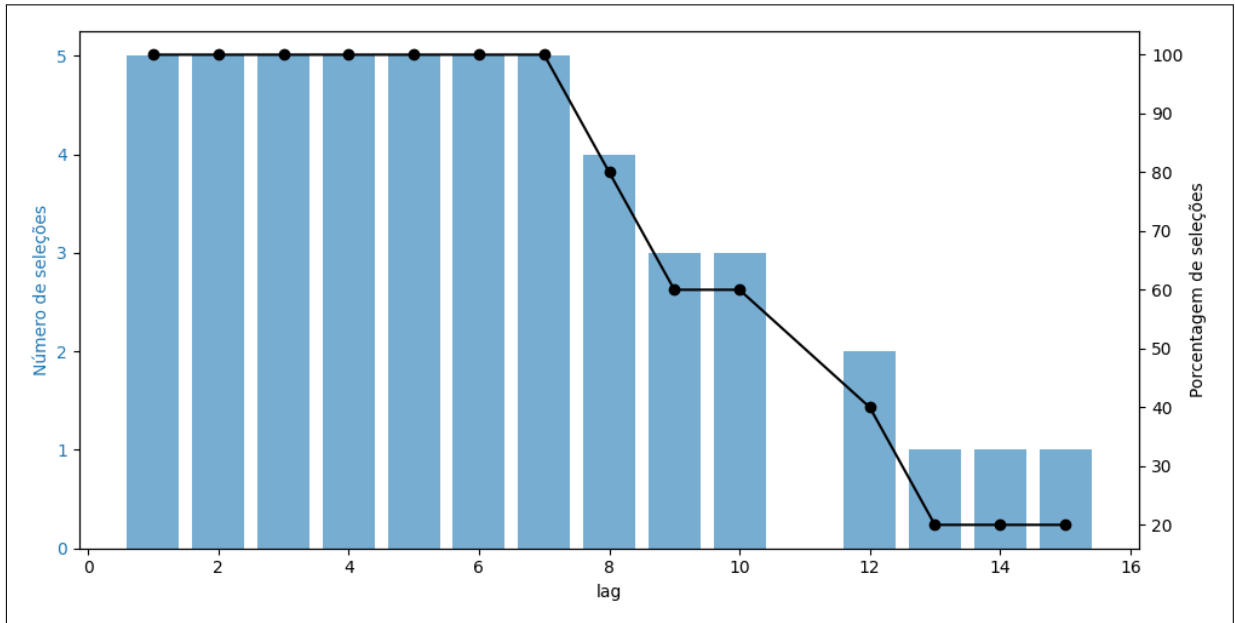
Tabela 13 – Atrasos significativos por base de dados e tipo de modelo

	RNAs	AR+RNAs	ARMA+RNAs
Brasília	[1, 2, 3, 4, 5, 6, 7, 8, 9, 12]	[5, 6, 7, 8, 11, 16, 17, 22, 24, 30]	[2, 3, 5, 7, 10, 16, 17, 22, 27, 30]
Florianópolis	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]	[5, 6, 7, 8, 9, 10, 11, 24]	[3, 9, 10, 11, 14, 19, 21, 22, 24, 27]
Natal	[1, 2, 3, 4, 5, 6, 7, 13, 14, 15]	[8, 9, 14, 15, 16, 21, 22, 23, 24, 30]	[7, 8, 14, 15, 16, 21, 22, 23, 24, 30]
Petrolina	[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]	[3, 4, 5, 12, 15, 17, 21, 23, 27, 29]	[9, 17]
São Luís	[1, 2, 3, 4, 5, 6, 7, 8, 10, 12]	[5, 6, 7, 10, 12, 15, 16, 18, 20, 30]	[3, 9, 11, 12, 15, 16, 18, 20, 28]

Fonte: Autoria própria (2022)

Em relação a quantidade de atrasos utilizados é comum supor que há uma relação entre a quantidade considerada e o desempenho. Entretanto, a Tabela 13 sugere que esta relação não é direta, pois em geral as RNAs apresentaram melhor desempenho que os Sistemas Híbridos mesmo utilizando entradas temporalmente mais próximas da saída desejada. Além disso, sempre que possível, é importante exortar um conjunto de entradas parcimonioso a fim de evitar a Maldição da Dimensionalidade e uso excessivo de esforço computacional.

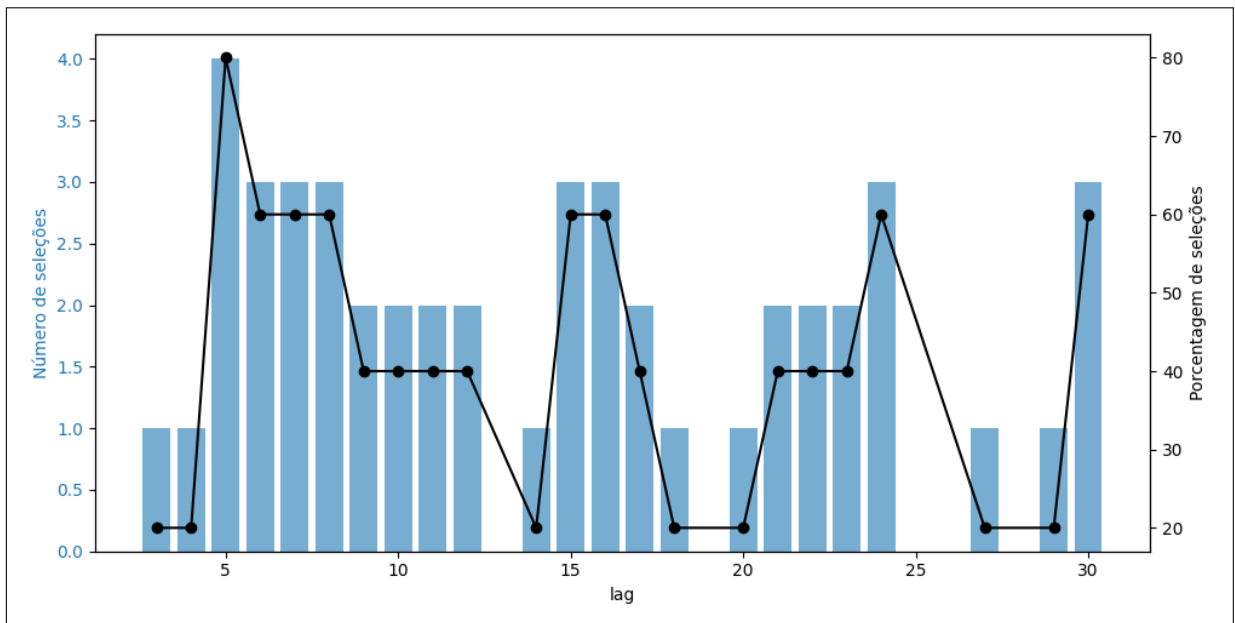
Gráfico 18 – Uso de cada atraso pelas RNAs



Fonte: Autoria própria (2022)

Como mostra o Gráfico 18, sobre as RNAs, é possível notar que os atrasos de 1 a 7 foram usados em todos os casos enquanto os atrasos 13, 14 e 15 foram os menos frequentes. Com isso, observa-se um padrão entre a frequência de uso e a recência dos atrasos, i.e., os atrasos mais próximos foram utilizados mais vezes.

Gráfico 19 – Uso de cada atraso pelos Sistemas Híbridos com AR

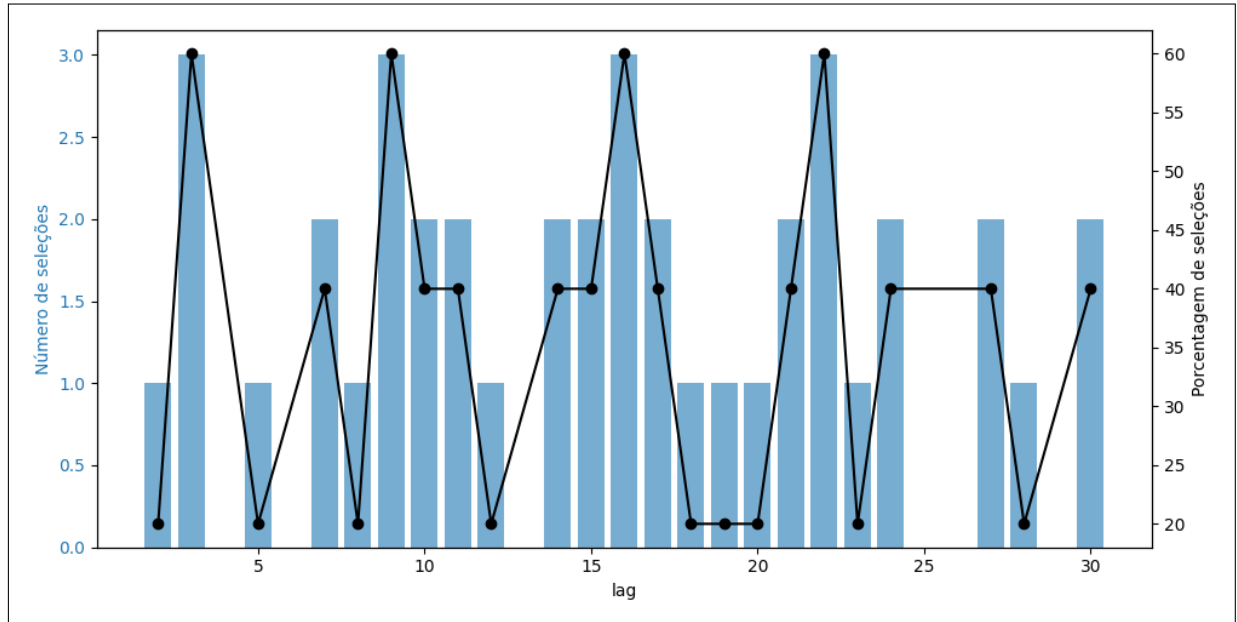


Fonte: Autoria própria (2022)

Nos Sistemas Híbridos com AR, Gráfico 19, não há um padrão semelhante ao visto no Gráfico 18. Ao invés disso, os atrasos 15, 16, 24 e 30 - atrasos temporalmente mais distantes - foram utilizados em 60% dos casos, enquanto os atrasos 3, 4, 6 e 12 foram utilizados em

média por 42% dos casos. Esse comportamento não padronizado também pode ser visto pelos Sistemas Híbridos com ARMA, Gráfico 20.

Gráfico 20 – Uso de cada atraso pelos Sistemas Híbridos com ARMA



Fonte: Autoria própria (2022)

Vale destacar que os modelos AR e ARMA, cujos parâmetros foram decididos por busca exaustiva, utilizaram em todos os casos as ordens 9 e (2, 2), respectivamente.

Por fim, é interessante refletir sobre o uso da PACF nas RNAs e principalmente nos Sistemas Híbridos. Partindo do pressuposto de que as séries temporais são complexas e constituídas por componentes lineares e não-lineares, é válido utilizar este método de seleção de entradas para RNAs. Porém nos Sistemas Híbridos tal método é aplicado na segunda etapa de previsão dos erros, o qual assume-se que a série seja não-linear. O fato da série de erros ser não-linear e o método de seleção de entradas buscar correlações lineares pode implicar negativamente no desempenho das previsões.

6 CONCLUSÃO

O uso do vento para geração de energia elétrica data do final do século XIX, mas foi nos anos 70, quando ocorreu a crise do petróleo, que diversos países voltaram sua atenção para geração de energia eólica. Desde então muitos investimentos têm sido feitos na área, de modo que é esperado para os próximos anos que muitos países vejam a energia eólica com maior importância, pois além de ser um recurso mais econômico também é sustentável.

Em 2020 o Brasil foi o terceiro maior investidor mundial nesse mercado. Alguns dos motivos para tal se dão pelas características do vento no território brasileiro, que são as altas velocidades médias, pouca variação de direção e baixa frequência de rajada e turbulência.

A medida que o mercado de energia eólica se expande surgem novos desafios tecnológicos e científicos, como a previsão da velocidade do vento. Tal tarefa tem um papel importante no planejamento e gerenciamento da produção energética. Na literatura é possível encontrar trabalhos que abordam esse problema com Modelos Lineares e RNAs. Nesta linha, técnicas que combinam vários modelos como os Sistemas Híbridos de correção de erro e *Ensembles* têm sido estudadas como uma alternativa mais robusta, as quais propõem a entrega de resultados mais acurados.

Tendo isso em vista, o presente trabalho utilizou os dados históricos de cinco cidades brasileiras (Brasília, Florianópolis, Natal, Petrolina e São Luís) para realizar uma comparação entre os métodos tradicionais e o estado da arte para previsão de séries temporais, sendo eles:

- Modelos Lineares: AR e ARMA;
- RNAs: MLP, RBF, ELM e ESN;
- Sistemas Híbridos: AR+MLP, AR+RBF, AR+ELM, AR+ESN, ARMA+MLP, ARMA+RBF, ARMA+ELM e ARMA+ESN;
- *Ensemble*: Mediana (ML), Mediana (RNAs), Mediana (RNAs sem RBF), Mediana (MU), Mediana (MU sem RBF), Mediana (todos), Média (ML), Média (RNAs), Média (RNAs sem RBF), Média (MU), Média (MU sem RBF) e Média (todos).

A comparação dos modelos se deu pelas métricas: Erro Quadrático Médio (MSE do inglês *Mean Squared Error*), Erro Absoluto Médio (MAE do inglês *Mean Absolute Error*), Variação Relativa Média (ARV do inglês *Average Relative Variance*) e o Índice de Concordância (IA do inglês *Index of Agreement*).

Através dos testes realizados foi possível observar que o modelo linear ARMA foi o melhor em alguns casos - Brasília, Natal e Petrolina pelas métricas MSE e MAE. Entretanto, ponderando os resultados de todas as métricas - MSE, MAE, ARV e IA - o modelo ESN foi o que se saiu melhor. Tal modelo tem a capacidade de reter informações que são usadas em previsões futuras, o que tende a favorecer a tarefa de previsão.

Por meio de um ranking baseado no método de Borda, modelos ARMA e ESN evidenciam uma dispersão relativamente grande ocupando do 1º ao 21º lugares e do 1ª ao 14º lugares, respectivamente. Por outro lado, os *Ensembles* de Média (ML) e Mediana (ML) variaram entre o 2º até 9º lugares, enquanto a Mediana (RNAs) oscilou entre o 9 e o 19º lugares. Na classificação final, os *Ensembles* detiveram o 2º, 3º, e 5º lugares.

Sobre os combinadores testados para a metodologia *Ensemble*, o melhor foi o de Mediana de acordo com o ranking final, pois as combinações feitas sobre os mesmos modelos com o operador de Média detiveram classificações inferiores.

Os demais modelos como a MLP, RBF e sistemas híbridos não foram destaque positivo, assumindo as piores classificações no ranking final. A ESN e determinados *Ensembles* estiveram entre os 5 melhores modelos. No entanto, o modelo ARMA - 4º lugar - se mostrou competitivo em determinados casos com o menor MSE e MAE.

Em conclusão, os desempenhos dos modelos variam conforme as bases de dados, não sendo possível apontar qual o melhor preditor absoluto, mas para problemas de previsão da velocidade do vento a ESN ou então *Ensembles* se mostraram mais adequadas.

6.1 Trabalhos futuros

Apesar dos inúmeros testes realizados sobre os modelos, este trabalho se restringiu à previsão um passo à frente. Entretanto é possível realizar uma investigação mais aprofundada prevendo vários passos à frente. Tal tarefa pode ser feita ajustando o valor alvo, como mencionado na Seção 4.

Outras abordagens de RNAs podem ser testadas, dentre elas outros *designs* de reservatórios de ESN e saídas não-lineares para ESNs e ELMs (SIQUEIRA *et al.*, 2012). Com relação aos *Ensembles* este trabalho fez a combinação de resultados por Média e Mediana, mas é interessante testar outras formas de combinar as saídas, como um combinador treinável, sendo possível utilizar as RNAs.

Outra forma de aprimorar os *Ensembles* são os modelos de seleção dinâmica. Essa técnica consiste em treinar vários modelos e ao fazer uma previsão para uma amostra em específico, selecionar apenas aquele que apresenta o melhor desempenho.

No que diz respeito ao processo de seleção de parâmetros para modelos lineares, este trabalho realizou busca exaustiva para encontrar os melhores valores, cabendo também os testes de outras metodologias encontradas na literatura, como metaheurísticas bioinspiradas (BELLOTTI *et al.*, 2020; SANTOS *et al.*,).

A princípio o presente estudo tinha o objetivo utilizar o Wrapper para seleção de entradas das RNAs e Sistemas Híbridos de correção de erro, porém a dimensão das bases inviabilizou sua aplicação. Ao invés deste, optou-se pela função de autocorrelação parcial (PACF). Além dos métodos citados neste trabalho cabe testar outros encontrados na literatura a fim de comparar seu impacto no desempenho dos modelos de previsão.

Por fim, é interessante que os modelos apresentados sejam testados considerando outras séries temporais, como aquelas relacionadas às grandes cidades do Brasil.

REFERÊNCIAS

- AGGARWAL, C. C. **Neural networks and deep learning**. [S.l.]: Springer, 2018. v. 10.
- ALENCAR, D. B. D. **Modelo híbrido baseado em séries temporais e redes neurais para previsão da geração de energia eólica**. 2018. Tese (Doutorado) — Universidade Federal do Pará, 2018.
- BELOTTI, J. *et al.* Comparative study of forecasting approaches in monthly streamflow series from brazilian hydroelectric plants using extreme learning machines and box & jenkins models. **Journal of Hydrology and Hydromechanics**, De Gruyter Poland, 2021.
- BELOTTI, J. T. *et al.* Air pollution epidemiology: a simplified generalized linear model approach optimized by bio-inspired metaheuristics. **Environmental Research**, Elsevier, 2020.
- BELOTTI, J. T. *et al.* **Previsão de vazões afluentes utilizando redes neurais artificiais e ensembles**. 2019. Dissertação (Mestrado) — Universidade Tecnológica Federal do Paraná, 2019.
- BOX, G. E.; DRAPER, N. R. **Empirical model-building and response surfaces**. [S.l.]: John Wiley & Sons, 1987. 424 p.
- BOX, G. E. *et al.* **Time series analysis: forecasting and control**. [S.l.]: John Wiley & Sons, 2015.
- BREIMAN, L. Bagging predictors. **Machine learning**, Springer, 1996.
- BROCKWELL, P. J.; DAVIS, R. A.; CALDER, M. V. **Introduction to time series and forecasting**. [S.l.]: Springer, 2002. v. 2.
- COUNCIL, G. W. E. **Global Wind Report 2021**: Global wind energy council. 2021.
- COVER, T. M. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. **IEEE transactions on electronic computers**, IEEE, 1965.
- DICKEY, D. A.; FULLER, W. A. Distribution of the estimators for autoregressive time series with a unit root. **Journal of the American statistical association**, Taylor & Francis, 1979.
- DUPONT, E.; KOPPELAAR, R.; JEANMART, H. Global available wind energy with physical and energy return on investment constraints. **Applied Energy**, Elsevier, 2018.
- FLEMING, P.; PROBERT, S. The evolution of wind-turbines: an historical review. **Applied energy**, Elsevier, 1984.
- FLORES, J. H. F.; ENGEL, P. M.; PINTO, R. C. Autocorrelation and partial autocorrelation functions to improve neural networks models on univariate time series forecasting. **International Joint Conference on Neural Networks**, IEEE, 2012.
- FREITAS, A. A. C. d. *et al.* **Previsão de series temporais via seleção de variáveis, reconstrução dinamica, ARMA-GARCH e redes neurais artificiais**. 2007. Tese (Doutorado) — Universidade Estadual de Campinas, 2007.
- GHAEMI, A. *et al.* Accuracy enhance of fault classification and location in a smart distribution network based on stacked ensemble learning. **Electric Power Systems Research**, Elsevier, 2022.

- GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. **Journal of machine learning research**, 2003.
- HAIDER, S. A. *et al.* Deep learning and statistical methods for short-and long-term solar irradiance forecasting for islamabad. **Renewable Energy**, Elsevier, 2022.
- HAYKIN, S. **Redes neurais: princípios e prática**. São Paulo: Bookman Editora, 2007.
- HUANG, G. B.; ZHU, Q. Y.; SIEW, C. K. Extreme learning machine: a new learning scheme of feedforward neural networks. **International Joint Conference on Neural Networks**, IEEE, 2004.
- INPE.
- JACOBS, R. A. *et al.* Adaptive mixtures of local experts. **Neural Computation**, IEEE, 1991.
- JAEGER, H. The “echo state” approach to analysing and training recurrent neural networks-with an erratum note. **German National Research Center for Information Technology GMD Technical Report**, Bonn, 2001.
- JONCAS, S. **Thermoplastic composite wind turbine blades: an integrated design approach**. 2010. Tese (Doutorado) — Delft University of Technology, 2010.
- JÚNIOR, L. A. T. **Combinação geométrica de métodos preditivos: aplicação à previsão de consumo residencial mensal de energia elétrica**. 2009. Tese (Doutorado) — PUC-Rio, 2009.
- KAZIENKO, P.; LUGHOFER, E.; TRAWINSKI, B. Editorial on the special issue “hybrid and ensemble techniques in soft computing: recent advances and emerging trends”. **Soft Computing**, Springer, 2015.
- KIM, D.; BAEK, J.-G. Bagging ensemble-based novel data generation method for univariate time series forecasting. **Expert Systems with Applications**, Elsevier, 2022.
- KUHN, M.; JOHNSON, K. *et al.* **Applied predictive modeling**. New York: Springer, 2013.
- LETCHER, T. M. **Wind energy engineering: a handbook for onshore and offshore wind turbines**. Cambridge: Academic Press, 2017.
- LETCHER, T. M. **Future energy: improved, sustainable and clean options for our planet**. Cambridge: Elsevier, 2020.
- LI, Y. *et al.* Smart wind speed forecasting approach using various boosting algorithms, big multi-step forecasting strategy. **Renewable energy**, Elsevier, 2019.
- MARTINS, F. R.; GUARNIERI, R. A.; PEREIRA, E. B. O aproveitamento da energia eólica. **Revista Brasileira de Ensino de Física**, SciELO Brasil, 2008.
- MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. **The bulletin of mathematical biophysics**, Springer, 1943.
- MILLIGAN, M.; SCHWARTZ, M.; WAN, Y.-h. **Statistical wind power forecasting models: Results for us wind farms**. United States, 2003.
- MORETTIN, P. A.; TOLOI, C. **Análise de séries temporais**. São Paulo: Blucher, 2006.
- NETO, P. S. de M. *et al.* Hybrid systems using residual modeling for sea surface temperature forecasting. **Scientific reports**, Nature Publishing Group, 2022.
- NETO, P. S. de M. *et al.* An intelligent perturbative approach for the time series forecasting problem. **The 2010 International Joint Conference on Neural Networks**, IEEE, 2010.

- NETO, P. S. de M. *et al.* An adaptive hybrid system using deep learning for wind speed forecasting. **Information Sciences**, Elsevier, 2021.
- NGO, N. *et al.* An ensemble machine learning model for enhancing the prediction accuracy of energy consumption in buildings. **Arabian Journal for Science and Engineering**, Springer, 2022.
- NIU, X.; WANG, J.; ZHANG, L. Carbon price forecasting system based on error correction and divide-conquer strategies. **Applied Soft Computing**, Elsevier, 2022.
- NOOROLLAHI, Y.; JOKAR, M. A.; KALHOR, A. Using artificial neural networks for temporal and spatial wind speed forecasting in iran. **Energy Conversion and Management**, Elsevier, 2016.
- NUNES, I.; SILVA, H. S. D. **Artificial neural networks: a practical course**. Switzerland: Springer, 2017.
- OLIVEIRA, J. F. de; SILVA, E. G.; NETO, P. S. de M. A hybrid system based on dynamic selection for time series forecasting. **Transactions on Neural Networks and Learning Systems**, IEEE, 2021.
- PIRES, J. **Sistema de Informações para Geração de Energia Eólica**. 2016. Disponível em: <https://www.ufrgs.br/sieolica/mapaeolico1.html>.
- POLIKAR, R. **Ensemble machine learning**. Boston: Springer, 2012.
- REIS, M. M. **Estatística para administradores I**. 2020. Disponível em: <https://www.inf.ufsc.br/~marcelo.menezes.reis/INE7001.html>.
- REN, Z.; HUANG, C.; LI, M. Research on wind power prediction. **Conference on Energy Internet and Energy System Integration**, IEEE, 2019.
- ROSSI, J.; NEVES, C. D. **Econometria e séries temporais com aplicações a dados da economia brasileira**. Rio de Janeiro: Grupo Gen-LTC, 2000.
- SANTOS, P. *et al.* Application of pso-based clustering algorithms on educational databases. **Latin American Conference on Computational Intelligence (LA-CCI)**, IEEE.
- SARANJ, A.; ZOLFAGHARI, M. The electricity consumption forecast: Adopting a hybrid approach by deep learning and arimax-garch models. **Energy Reports**, Elsevier, 2022.
- SCHAPIRE, R. E. The strength of weak learnability. **Machine learning**, Springer, 1990.
- SHAO, H. *et al.* Short-term wind speed forecasting using wavelet transformation and adaboosting neural networks in yunnan wind farm. **Renewable Power Generation**, IET, 2016.
- SHARKEY, A. J. **Combining artificial neural nets: ensemble and modular multi-net systems**. [S.l.]: Springer Science & Business Media, 2012.
- SILVA, P. K. d. O. *et al.* **Análise e previsão de curto prazo do vento através de modelagem estatística em áreas de potencial eólico no Nordeste do Brasil**. 2017. Tese (Doutorado) — Universidade Federal de Campina Grande, 2017.
- SILVESTRE, L. J. **Regularização de Extreme Learning Machines: uma abordagem com matrizes de afinidade**. 2014. Tese (Doutorado) — Universidade Federal de Minas Gerais, 2014.
- SIQUEIRA, H. *et al.* Echo state networks for seasonal streamflow series forecasting. **International Conference on Intelligent Data Engineering and Automated Learning**, Springer, 2012.

- SIQUEIRA, H.; LUNA, I. Performance comparison of feedforward neural networks applied to streamflow series forecasting. **Mathematics in Engineering, Science & Aerospace (MESA)**, 2019.
- SIQUEIRA, H. V. *et al.* **Previsão de series de vazões com redes neurais artificiais e modelos lineares ajustados por algoritmos bio-inspirados**. 2009. Dissertação (Mestrado) — Universidade Estadual de Campinas, 2009.
- SIQUEIRA, H. V. *et al.* **Máquinas desorganizadas para previsão de séries de vazões**. 2013. Tese (Doutorado) — Universidade Estadual de Campinas, 2013.
- SOUZA, R. B. d. **Avaliação de modelos de inteligência artificial para previsão da velocidade de vento em curto prazo**. 2014. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2014.
- TAN, J. **Feature Selection for Machine Learning in Python-Wrapper Methods**. Towards Data Science, 2020. Disponível em: <https://towardsdatascience.com/feature-selection-for-machine-learning-in-python-wrapper-methods-2b5e27d2db31>.
- XU, L.; MAO, J. Short-term wind power forecasting based on elman neural network with particle swarm optimization. **Chinese Control and Decision Conference (CCDC)**, IEEE, 2016.
- ZHANG, C.; MA, Y. **Ensemble machine learning: methods and applications**. New York: Springer, 2012.
- ZHANG, C. *et al.* Research on short-term wind power prediction based on combined forecasting models. **MATEC Web of Conferences**, EDP Sciences, 2016.
- ZHANG, G. P. Time series forecasting using a hybrid arima and neural network model. **Neurocomputing**, Elsevier, 2003.
- ZHANG, S. *et al.* A novel ensemble deep learning model with dynamic error correction and multi-objective ensemble pruning for time series forecasting. **Information Sciences**, Elsevier, 2021.

ANEXO A – Primeira página do artigo submetido

An Extensive Comparison of Linear Models, Machine Learning approaches and combination methods for Wind Speed Forecasting

Tathiana Mikamura Barchi^{1†}, Lucas Francesco Piccioni
Costa^{1†}, Mauren L. S. C. Andrade^{1†}, Marcella S. R.
Martins^{2†}, João F. L. de Oliveira^{3†}, Paulo S. G. de Mattos
Neto^{4†} and Hugo V. Siqueira^{1,2*†}

¹Graduate Program in Computer Sciences, Federal University of
Technology - Parana, Ponta Grossa, Parana, Brazil.

²Department of Electronic Engineering, Federal University of
Technology - Parana, Ponta Grossa, Parana, Brazil.

³Escola Politécnica, Universidade de Pernambuco, Recife,
Pernambuco, Brazil.

⁴Centro de Informatica, Universidade Federal de Pernambuco,
Recife, Pernambuco, Brazil.

*Corresponding author(s). E-mail(s): hugosiqueira@utfpr.edu.br;

†These authors contributed equally to this work.

Abstract

In the last decade, the worldwide appeal for the development of new technologies related to green energy has increased. One of the leading representatives of this field is wind energy, a crucial element in the current energy matrices of many countries. In this context, wind speed forecasting is a crucial task since it is related to the energy planning of the operation. However, the volatile and intermittent nature of wind speed and intrinsic nonlinear patterns make the prediction task challenging. In this sense, this work proposes an extensive comparison considering 26 distinct forecasting models among linear approaches from the Box & Jenkins methodology, artificial neural networks, hybrid