Centro de Informática
UFPE

Diogo Medeiros de Almeida

Propostas de Modelos Híbridos para o Mapeamento de Padrões Locais em Séries Temporais de Velocidade do Vento

Universidade Federal de Pernambuco
secpos@cin.ufpe.br
https://portal.cin.ufpe.br/pos-graduacao

Recife
2024

Diogo Medeiros de Almeida

Propostas de Modelos Híbridos para o Mapeamento de Padrões Locais em Séries Temporais de Velocidade do Vento

Trabalho apresentado ao Programa de Pós-graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Doutor em Ciência da Computação.

Área Concentração: Inteligência Computacional.

Orientador: Prof. Dr. Daniel Carvalho da Cunha.
Coorientador: Prof. Dr. Paulo Salgado Gomes de Mattos Neto.

Recife
2024

.Catalogação de Publicação na Fonte. UFPE - Biblioteca Central

Almeida, Diogo Medeiros de.
Propostas de modelos híbridos para o mapeamento de padrões locais em séries temporais de velocidade do vento / Diogo Medeiros de Almeida. - Recife, 2024.
131f.: il.

Tese (Doutorado) - Universidade Federal de Pernambuco, Centro de Informática, Pós-graduação em Ciência da Computação, 2024.
Orientação: Daniel Carvalho da Cunha.
Coorientação: Paulo Salgado Gomes de Mattos Neto.
Inclui referências.

1. velocidade do vento; 2. previsão de séries temporais; 3. modelos híbridos; 4. padrões globais; 5. padrões locais. I. Cunha, Daniel Carvalho da. II. Mattos Neto, Paulo Salgado Gomes de. III. Título.

UFPE-Biblioteca Central

Diogo Medeiros de Almeida

“Propostas de Modelos Híbridos para o Mapeamento de Padrões Locais em Séries Temporais de Velocidade do Vento”

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Ciência da Computação. Área de Concentração: Inteligência Computacional.

Aprovada em: 03/12/2024.

Orientador: Prof. Dr. Daniel Carvalho da Cunha

BANCA EXAMINADORA

Prof. Dr. Ricardo Bastos Cavalcante Prudêncio
Centro de Informática / UFPE

Prof. Dr. Divanilson Rodrigo de Sousa Campelo
Centro de Informática / UFPE

Prof. Dr. Leandro Maciel Almeida
Centro de Informática / UFPB

Prof. Dr. Manoel Henrique da Nóbrega Marinho
Escola Politécnica de Pernambuco / UPE

Prof. Dr. Paulo Renato Alves Firmino
Centro de Ciências e Tecnologia / UFCA

Dedico este trabalho aos meus pais, Cauby Almeida Filho e Maria do Carmo Almeida, por sempre acreditarem em mim e por terem abdicado de suas vidas em prol das realizações e da felicidade de seus filhos. Ao meu irmão, Cauby Almeida Neto, por seu grande incentivo e apoio.

AGRADECIMENTOS

Em primeiro lugar, agradeço a Deus por ter permitido vencer esse desafio, que foi concluir o doutorado no Centro de Informática da UFPE.

Agradeço, muito especialmente, aos meus pais, Cauby Almeida Filho e Maria do Carmo Almeida, pelo interesse em me ver concluir o doutorado, e por serem minha eterna inspiração para continuar na luta por grandes objetivos na vida.

Agradeço ao meu irmão, Cauby Almeida Neto, por sempre me incentivar a crescer profissionalmente.

Agradeço ao meu orientador, professor Dr. Daniel Cunha, e ao meu coorientador, professor Dr. Paulo Mattos. Pois, acreditaram em mim e me conduziram nessa pesquisa, que foi o maior desafio da minha vida.

Aos familiares e amigos por todo apoio e incentivo.

À Universidade Federal de Pernambuco, sobretudo ao Centro de Informática com toda a sua equipe de professores e funcionários. Visto que, proporcionaram um ambiente de excelência para minha formação.

A todos aqueles que acreditam em sonhos.

RESUMO

A energia eólica é considerada uma alternativa promissora na matriz energética global devido à sua disponibilidade abundante e ao baixo impacto ambiental em termos de emissões de gases de efeito estufa. Contudo, sua integração ao sistema elétrico enfrenta desafios técnicos significativos. Especialmente, devido à variabilidade temporal e espacial das velocidades e direções do vento, que resultam em uma geração intermitente e não despachável. Essa característica compromete a estabilidade do fornecimento de energia elétrica, exigindo estratégias avançadas de planejamento, operação e controle em redes interligadas. Para mitigar esses desafios, a previsão adequada da velocidade do vento torna-se um requisito para o dimensionamento apropriado de sistemas eólicos. Isto permite otimizar a operação de turbinas, ajustar o despacho energético e implementar políticas de manutenção preditiva. Apesar das redes neurais serem conhecidas como aproximadores universais de funções, aplicar essa teoria na prática é algo também desafiador. Ademais, a modelagem das séries temporais de velocidade do vento apresenta alta complexidade devido aos padrões não lineares e estocásticos inerentes. Portanto, há uma demanda para o uso de métodos preditivos robustos, como modelos híbridos que integram técnicas estatísticas e aprendizado de máquina. O princípio da "sabedoria das multidões" postula que a resposta agregada de um grupo de indivíduos geralmente supera a de um único especialista. Esse conceito também se aplica a modelos de previsão, em que a combinação de múltiplos preditores frequentemente proporciona resultados superiores ao desempenho do melhor preditor individual. Este cenário tem impulsionado esforços significativos em pesquisa e desenvolvimento tecnológico em escala global. Portanto, este trabalho investiga o uso de diferentes modelos híbridos e suas combinações para alcançar um mapeamento mais adequado de padrões locais em séries temporais complexas de velocidade do vento. Desta forma, foram propostos modelos híbridos, baseando-se na estratégia de dividir para conquistar. Supõe-se que o processo de aprendizado de cada sub-série localmente possa ser mais efetivo do que o aprendizado da série completa de forma global. No geral, os resultados em séries de velocidade do vento com diferentes granularidades e diversos tamanhos evidenciaram a eficácia das propostas. A principal proposta desta pesquisa, denominada LocLN, foi composta por modelos de treinamento rápido. Tais quais, modelos autorregressivos, integrados e de médias móveis (ARIMA, do inglês *Auto Regressive Integrated Moving-Average*) e máquinas de aprendizado extremo (ELM, do inglês *Extreme Learning Machine*). O LocLN conseguiu obter uma diminuição de aproximadamente 30% nos erros de previsão para o horizonte de 3h um passo a frente, quando comparado com modelos individuais de redes neurais recorrentes. Ademais, o teste de hipótese de Diebold-Mariano (DM) sobre os erros quadráticos revelou a relevância do LocLN em relação aos demais 18 métodos concorrentes nas 10 bases de dados. Foram 103 vitórias, 75 empates e 2 derrotas

entre 180 comparações no teste DM. Estes resultados demonstraram que em 98,8% dos casos comparados, o LocLN foi superior ou igual aos métodos concorrentes que possuem abordagens individuais e também híbridas, tais quais *bagging* e *boosting*.

Palavras-chaves: Velocidade do vento, previsão de séries temporais, modelos híbridos, padrões globais, padrões locais.

ABSTRACT

Wind energy is considered a promising alternative in the global energy matrix due to its abundant availability and low environmental impact in terms of greenhouse gas emissions. However, its integration into the electrical system faces significant technical challenges, especially due to the temporal and spatial variability of wind speeds and directions, resulting in intermittent and non-dispatchable generation. This characteristic compromises the stability of electricity supply, requiring advanced planning, operation, and control strategies in interconnected networks. To mitigate these challenges, accurate wind speed forecasting becomes essential for the proper sizing of wind power systems. This enables the optimization of turbine operation, energy dispatch adjustments, and the implementation of predictive maintenance policies. Although neural networks are known as universal function approximators, applying this theory in practice remains challenging. Furthermore, modeling wind speed time series presents high complexity due to inherent nonlinear and stochastic patterns. Therefore, there is a demand for robust predictive methods, such as hybrid models that integrate statistical techniques and machine learning. The "wisdom of crowds" principle states that the aggregated response of a group of individuals often outperforms that of a single expert. This concept also applies to forecasting models, where combining multiple predictors frequently yields superior results compared to the best individual predictor. This scenario has driven significant efforts in research and technological development on a global scale. Thus, this study investigates the use of different hybrid models and their combinations to achieve a more suitable mapping of local patterns in complex wind speed time series. Hybrid models were proposed based on the "divide and conquer" strategy, assuming that the learning process for each local sub-series could be more effective than learning the entire series globally. Overall, the results in wind speed series with different granularities and various sizes highlighted the effectiveness of the proposed approaches. The main proposal of this research, named LocLN, consisted of fast-training models, such as autoregressive integrated moving average models (ARIMA) and extreme learning machines (ELM). The LocLN achieved a reduction of approximately 30% in forecasting errors for the 3-hour one-step-ahead horizon when compared to individual recurrent neural network models. Furthermore, the Diebold-Mariano (DM) hypothesis test on squared errors highlighted the relevance of LocLN in comparison to the other 18 competing methods across 10 datasets. There were 103 wins, 75 ties, and 2 losses out of 180 comparisons in the DM test. These results demonstrated that in 98.8% of the compared cases, LocLN was superior or equal to competing methods that employ individual and hybrid approaches, such as bagging and boosting.

Key-words: Wind speed, time series forecasting, hybrid models, global patterns, local patterns.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ϵ-SVR	*ϵ-Support Vector Regression*
ACF	*Autocorrelation Function*
ADF	*Augmented Dickey Fuller*
AR	*Autoregressive*
ARIMA	*Auto Regressive Integrated Moving-Average*
ARMA	*Autoregressive Moving Average*
BDMEP	Banco de Dados Meteorológicos
CSS	*Conditional Sum of Squares*
DM	Diebold-Mariano
EEMD	*Ensemble Empirical Mode Decomposition*
ELM	*Extreme Learning Machine*
EMD	*Empirical Mode Decomposition*
ES	*Exponential Smoothing*
GRNN	*General Regression Neural Network*
GRU	*Gated Recurrent Units*
IM	Informação Mútua
IMF	*Intrinsic Mode Functions*
INMET	Instituto Nacional de Meteorologia
INPE	Instituto Nacional de Pesquisas Espaciais
KKT	Karush-Kuhn-Tucker
LSTM	*Long Short Term Memory*
MA	*Moving Average*
MAE	*Mean Absolute Error*
MAPE	*Mean Absolute Percentage Error*
MLE	*Maximum Likelihood Estimator*
MLP	*Multilayer Perceptron*
NWP	*Numerical Weather Prediction*
POCID	*Prediction Of Change In Direction*
PQ	Programação Quadrática

RBF	*Radial-Basis Function*
RMSE	*Root Mean Squared Error*
RNA	Rede Neural Artificial
RNN	*Recurrent Neural Network*
RPROP	*Resilient Propagation*
SNNS	*Stuttgart Neural Network Simulator*
SONDA	Sistema de Organização Nacional de Dados Ambientais
SRM	*Structural Risk Minimization*
STL	*Seasonal Trend Decomposition using Loess*
SVD	*Singular Value Decomposition*
SVM	*Support Vector Machine*
SVR	*Support Vector Regression*
VMD	*Variational Mode Decomposition*

LISTA DE SÍMBOLOS

W_t	t-ésimo valor da série temporal diferenciada
∇	Primeira diferença da série temporal
y_t	t-ésimo valorda série temporal
ϕ_i	i-ésimo parâmetro na formulação do modelo AR
θ_j	j-ésimo parâmetro na formulação do modelo MA
ε_t	t-ésimo erro aleatório da série temporal
τ_j	j-ésimo peso de conexão da camada oculta na formulação do modelo MLP
β_{ij}	Peso de conexão entre i-ésimo neurônios da camada de entrada e j-ésimo neurônios da camada oculta na formulação do modelo MLP
e^x	Exponencial natural
$\mathbf{v}$	Vetor de todos os parâmetros na formulação do modelo MLP
$\triangle_0$	Valor inicial de atualização na formulação do RPROP
$\triangle_{max}$	Valor máximo para o tamanho do passo na formulação do RPROP
μ	Expoente de decaimento de peso na formulação do RPROP
$\widehat{y}_t$	t-ésimo valor da previsão da série temporal
π_{ij}	Peso entre i-ésimo e j-ésimo neurônios da função de erro na formulação do RPROP
E	função de erro da MLP
$\boldsymbol{x}_t$	t-ésimo vetor de entrada representando as defasagens de y_t para o SVM
$\langle \cdot, \cdot \rangle$	Produto escalar
$\boldsymbol{\omega}$	Vetor de pesos na formulação do modelos
$\varphi(\cdot)$	Função não linear na formulação do modelo ϵ-SVR
b	*bias* na formulação do modelos
$\lvert \cdot \rvert_\epsilon$	Função de perda ϵ-insensível do modelo ϵ-SVR

ϵ	Limite para a margem de tolerância na formulação do modelo ϵ-SVR
$\boldsymbol{C}_s$	Parâmetro de regularização na formulação de otimização do modelo ϵ-SVR
ξ_t^+	t-ésima penalidade acima do ϵ-tubo na formulação do modelo ϵ-SVR
ξ_t^-	t-ésima penalidade abaixo do ϵ-tubo na formulação do modelo ϵ-SVR
α^+	Multiplicador de Lagrange na formulação do modelo ϵ-SVR
α^-	Multiplicador de Lagrange na formulação do modelo ϵ-SVR
η^+	Multiplicador de Lagrange na formulação do modelo ϵ-SVR
η^-	Multiplicador de Lagrange na formulação do modelo ϵ-SVR
∂	Derivada parcial
σ	Desvio padrão na formulação do *kernel* RBF
λ	Inverso do desvio padrão na formulação do *kernel* RBF
$h^{(t)}$	t-ésimo estado oculto do modelo LSTM
$c^{(t)}$	t-ésimo estado da célula do modelo LSTM
$f\_g^{(t)}$	t-ésimo portão de esquecimento do modelo LSTM
$i\_g^{(t)}$	t-ésimo portão de entrada do modelo LSTM
$o\_g^{(t)}$	t-ésimo portão de saída do modelo LSTM
σ_g	Função de ativação sigmoide
$c\_u^{(t)}$	t-ésima célula de atualização do modelo LSTM
$tanh$	Função de ativação tangente hiperbólica
z_t	t-ésima porta de atualização do modelo GRU
$\boldsymbol{u}_z$	z-ésimo estado da célula do modelo GRU
h_t	t-ésimo estado da célula do modelo GRU
r_t	t-ésimo portão de reinicialização do modelo GRU
$\widetilde{h}_t$	t-ésimo novo conteúdo de memória do modelo GRU
$\odot$	operador produto Hadamard do modelo GRU

L_t	t-ésima componente linear da série temporal
N_t	t-ésima componente não linear da série temporal
$\widehat{L}_t$	t-ésima previsão do modelo linear
$\widehat{N}_t$	t-ésima previsão do modelo não linear
M_L	Modelo linear
G	Técnica de busca para os hyperparâmetros do modelo
M_N	Modelo não linear
M_S	Modelo treinado na série
M_R	Modelo treinado nos resíduos
Y_t	Vetor que representa a série temporal até o tempo t
E_t	Vetor que representa a série temporal de resíduos até o tempo t
M_C	Modelo treinado para combinação das séries
N	Quantidade de modelos
M	Modelo
S_N	N-ésima série reamostrada por *bagging*
P'	*Pool* de preditores
C	Método de combinação de preditores
I	Quantidade inicial de subconjuntos no *loop* do método LocPart
L	Valor de incremento de subconjuntos no *loop* do método LocPart
J	Quantidade máxima de subconjuntos no *loop* do método LocPart
D_I	I-ésimo subconjuntos disjuntos no método LocPart
V	Medida de avaliação
p	preditores especialistas locais selecionados no método LocPart
$LocPart_L$	Método LocPart com modelos base lineares
$LocPart_N$	Método LocPart com modelos base não lineares
C_{SEL}	Método de combinação selecionado no método LocLN

ADD	Método de combinação linear aditivo
T	Tamanho da série temporal
$\overline{Y}$	Média das observações nos conjunto de teste da série temporal
R_i	Relação percentual na métrica i entre proposta e modelo referência
i_{prop}	Valor obtido na métrica i da proposta
i_{ref}	Valor obtido na métrica i do modelo referência
H_0	Hipótese nula
E_s	Valor esperado de uma variável aleatória discreta

SUMÁRIO

1 INTRODUÇÃO

Este capítulo fornece uma contextualização sobre a motivação para o desenvolvimento desta pesquisa. Além disso, os principais objetivos e publicações são apresentados. Por fim, é descrita a estrutura do restante do documento.

1.1 Contextualização e Motivação

Em 2023, a energia eólica atingiu a marca histórica de 1 TW de capacidade instalada, e os próximos anos marcarão um período de transição crucial para a indústria eólica global. Foram necessários 40 anos para chegar nessa marca, e o próximo 1 TW levará menos de uma década (GWEC, 2023).

O Brasil está no 6º lugar no ranking mundial de capacidade instalada em energia eólica *onshore*. Em 2012, este país se situava na 15ª posição (EóLICA, 2024). Já em relação ao ranking mundial anual de novas instalações de eólica *onshore*, o Brasil conseguiu obter a 3ª posição em 2023 (EóLICA, 2023a). Isto demonstra que este país tem seguido a tendência mundial de aumento nos investimentos em energia eólica. De acordo com a Associação Brasileira de Energia Eólica (ABEEólica), em outubro de 2024, o setor de energia eólica atingiu 33 GW em operação comercial com 1.085 parques eólicos e 11.533 aerogeradores em 12 estados do país. Isso representa 16% da matriz elétrica brasileira (EóLICA, 2024), conforme ilustra a Figura 1. Em dias de geração recorde, a energia eólica já chegou a abastecer 100% do consumo da Região Nordeste e exportou 29,41% para as demais regiões do país. Por exemplo, no dia 22/08/2024 houve um recorde de geração com 17.697 MW médios (EóLICA, 2024). O Nordeste brasileiro é uma região que possui um dos melhores ventos do mundo para produção de energia eólica, e por este motivo 90% dos parques eólicos brasileiros estão instalados na Região Nordeste (EóLICA, 2023b).

Mesmo assim, de acordo com (SILVA, 2003), há desafios na exploração de energia eólica no Nordeste, pois nesta região o movimento atmosférico é significativamente influenciado pela fricção do vento com a superfície terrestre. Essa fricção reduz a velocidade do vento à medida que se aproxima do solo. Esse perfil vertical de velocidade pode gerar variações de alta frequência na velocidade do vento, conhecidas como turbulências atmosféricas. Portanto, apesar da energia eólica ser considerada uma opção atraente pelo fato de ser abundante e ecológica, vários desafios são enfrentados em sua exploração: a variabilidade na intensidade e na direção dos ventos causa um fornecimento instável de eletricidade ao sistema de energia (QU et al., 2019). Desta forma, a integração desta energia no sistema elétrico representa um desafio para as operações e práticas de planejamento (JIANG et al., 2019). Uma alternativa é utilizar sistemas preditivos para prever a velocidade do vento (QU

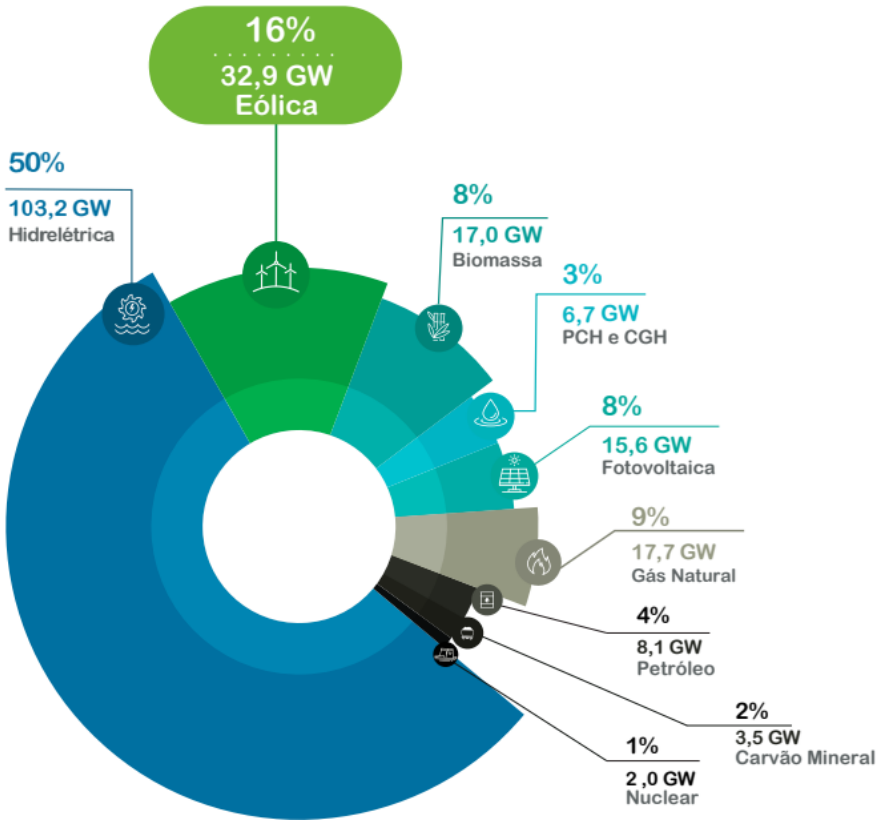

Figura 1 – Composição da Matriz Elétrica Brasileira em outubro de 2024.

Fonte: ANEEL/ABEEólica 2024.

et al., 2019). Portanto, a análise e a avaliação deste tipo de energia têm chamado a atenção de pesquisas em todo o mundo (AHMADI; KHASHEI, 2021).

Com um sistema capaz de prever a velocidade do vento, é possível mitigar operações instáveis quando a eletricidade gerada estiver incorporada na rede elétrica (JIANG et al., 2019). Na literatura, de acordo com o horizonte de previsão, são encontrados dois tipos de previsões para a velocidade do vento: as previsões a curto prazo (escalas de tempo em minutos, horas ou dias) e as previsões a longo prazo (escalas de meses ou anos) (WANG et al., 2014b). Além disso, de acordo com os métodos utilizados, os sistemas preditivos para a velocidade do vento podem ser classificados em quatro categorias: métodos físicos, métodos estatísticos, métodos de aprendizado de máquina e métodos híbridos (WANG et al., 2014a). Os métodos em cada categoria têm suas características específicas (WANG; LI; BAI, 2018), vantagens e desvantagens, que devem ser consideradas antes do desenvolvimento de um sistema preditivo.

Modelos físicos baseados em previsão numérica do tempo (NWP, do inglês *Numerical Weather Prediction*) são frequentemente usados para previsão de velocidade do vento a longo prazo (JIANG et al., 2019), usam dados meteorológicos e geográficos, como informações sobre temperatura, velocidade, densidade e topografia do ar (NIU; WANG, 2019). No entanto, não são confiáveis para previsões a curto prazo, uma vez que a complexidade do

processo de cálculo e os altos custos limitam a aplicação dos modelos NWP (WANG; LI; BAI, 2018).

Métodos estatísticos como os modelos autorregressivos, integrados e de médias móveis (ARIMA, do inglês *Auto Regressive Integrated Moving-Average*) e suas variantes são utilizados há algumas décadas para prever a velocidade do vento (KAMAL; JAFRI, 1997), (TORRES et al., 2005), (CADENAS; RIVERA, 2007), (KAVASSERI; SEETHARAMAN, 2009). Os modelos ARIMA, apesar de demonstrarem boa capacidade preditiva no campo da previsão de velocidade do vento e do baixo custo computacional (JUNG; BROADWATER, 2014), somente são capazes de mapear padrões lineares da série temporal (BOX; JENKINS; REINSEL, 2008). Neste caso, somente são mapeadas as relações temporais que podem ser representadas por funções lineares.

Com o rápido desenvolvimento da inteligência artificial, vários métodos de aprendizado de máquina foram aplicados à previsão da velocidade do vento, por exemplo, Rede Neural Artificial (RNA) (ALEXIADIS et al., 1998), (SFETSOS, 2002), (FLORES; TAPIA; TAPIA, 2005), máquina de vetores de suporte (SVM, do inglês *Support Vector Machine*) (MOHANDES et al., 2004), (SALCEDO-SANZ et al., 2011), (KONG et al., 2015), máquinas de aprendizado extremo (ELM, do inglês *Extreme Learning Machine*) (SAAVEDRA-MORENO et al., 2013), memória de curto e longo prazo (LSTM, do inglês *Long Short Term Memory*) (MEMARZADEH; KEYNIA, 2020; LIU; LIN; FENG, 2021), unidade recorrente fechada (GRU, do inglês *Gated Recurrent Units*) (LIU; LIN; FENG, 2021). Cada um destes modelos tem a habilidade de reconhecer padrões não lineares em uma série. Isto é, reconhecer relações temporais que não conseguem ser representadas por funções lineares. Além disso, cada um destes métodos tem suas particularidades.

No entanto, a principal limitação de modelos individuais, é o mapeamento tradicional global de padrões, em que um único modelo é encarregado de reconhecer todos os padrões da série temporal em todo o conjunto de treinamento. Apesar das redes neurais serem conhecidas como aproximadores universais de funções, aplicar essa teoria na prática é um desafio (AGGARWAL, 2018). Geralmente, é necessário um grande número de unidades ocultas, o que aumenta o número de parâmetros a serem aprendidos, tornando bem complexo o treinamento em dados limitados.

As séries temporais de velocidade do vento exibem padrões complexos e ambíguos, o que constitui um desafio para a modelagem e mapeamento destes comportamentos. Primeiramente, devido a alguns parâmetros complexos que influenciam fortemente o vento. Por exemplo, as propriedades topográficas da Terra, a rotação global, diferenças de temperatura e pressão (HU; WANG; ZENG, 2013). Em segundo lugar, o comportamento do vento está associado à heterogeneidade das regiões, como rugosidade da superfície, variabilidade da vegetação, uso e ocupação do solo (FERREIRA; SANTOS; LUCIO, 2019). Em terceiro lugar, as séries de velocidade do vento apresentam média e variância mensais não constantes, alta volatilidade e irregularidade (HU; WANG; ZENG, 2013). Por fim, algumas

séries de velocidade do vento podem possuir padrões com características caóticas (JIANG et al., 2019).

Portanto, modelos individuais podem não ser a melhor escolha para o mapeamento de padrões de séries temporais de velocidade do vento. De acordo com (GÉRON, 2017), o princípio da "sabedoria das multidões" postula que a resposta agregada de um grupo de indivíduos geralmente supera a de um único especialista. Esse conceito também se aplica a modelos de previsão, em que a combinação de múltiplos preditores frequentemente proporciona resultados superiores ao desempenho do melhor preditor individual. Esse conjunto de preditores é chamado de *ensembles*. A utilização de *ensembles* é bastante comum em previsão de séries temporais, não somente devido à sua maior precisão preditiva, mas também à menor incerteza em comparação com modelos individuais (ALMEIDA, 2024; SANTOS JÚNIOR et al., 2023; de MATTOS NETO et al., 2020a; PETROPOULOS; HYNDMAN; BERGMEIR, 2018; BERGMEIR; HYNDMAN; BENÍTEZ, 2016; SERGIO; LIMA; LUDERMIR, 2016), principalmente no setor de energia eólica (ALMEIDA; de MATTOS NETO; CUNHA, 2024; YAN et al., 2022; AHMADI; KHASHEI, 2021; FERREIRA; SANTOS; LUCIO, 2019; RUIZ-AGUILAR et al., 2021; JIANG; CHE; WANG, 2021; HUANG; WANG; HUANG, 2021a; de MATTOS NETO et al., 2020; ALENCAR et al., 2018a; CAMELO et al., 2018a).

A diversidade entre as previsões dos preditores em um *ensemble* é fundamental para reconhecer diferentes padrões nas séries temporais (SERGIO; LIMA; LUDERMIR, 2016). Em (GÉRON, 2017) é citado que uma maneira de obter um conjunto diversificado de preditores em um *ensemble* homogêneo (modelos individuais do mesmo tipo), é treinar os preditores em diferentes subconjuntos do conjunto de treinamento. Esse método é chamado de *pasting*. Depois que todos os preditores são treinados, o *ensemble* pode fazer uma previsão para uma nova instância agregando as previsões de todos os preditores. Em (AHMADI; KHASHEI, 2021) o *ensemble pasting* é chamado de modelo híbrido paralelo (MHP), pois todos os modelos individuais (também conhecidos como modelos base) usam dados de série temporal como entrada para gerar previsões, possibilitando um treinamento dos preditores em paralelo. Ademais, uma etapa de seleção pode ser utilizada para descartar os preditores com maiores taxas de erro, e assim minimizar a probabilidade de obter resultados distantes do valor desejado (SERGIO, 2017). Outra forma de obter preditores com diversidade é por meio de *ensembles* heterogêneos, ou seja, *ensembles* construídos com modelos base de tipos diferentes. Portanto, uma estratégia comumente utilizada é empregar o método *boosting* com um modelo linear e um modelo não linear. De acordo com (GÉRON, 2017), a ideia geral do *boosting* é treinar preditores sequencialmente, com cada um tentando corrigir seu predecessor. Em (AHMADI; KHASHEI, 2021) esse segundo método de *ensemble* é chamado de modelo híbrido serial (MHS), pois os modelos base são aplicados sequencialmente aos dados. Assim, um modelo usa os dados originais da série temporal como entrada, e o segundo modelo usa os resíduos do primeiro modelo como entrada. Esses resíduos são as diferenças entre os valores reais observados e os valores

previstos pelo primeiro modelo. Posteriormente, as previsões dos modelos individuais são combinadas de alguma maneira para produzir a previsão final (de MATTOS NETO et al., 2020).

Alguns MHPs têm a capacidade de capturar padrões locais complexos ao longo do tempo (SANTOS JÚNIOR et al., 2023; RUIZ-AGUILAR et al., 2021; JIANG; CHE; WANG, 2021; de MATTOS NETO et al., 2020a), baseando-se na estratégia de dividir para conquistar. Supõe-se que o processo de aprendizado de cada sub-série pode ser mais simples do que o aprendizado da série completa (de MATTOS NETO et al., 2020a). Enquanto os MHSs buscam mapear uma mistura de padrões lineares e não lineares (YAN et al., 2022; de MATTOS NETO et al., 2020; HUANG; WANG; HUANG, 2021a; ALENCAR et al., 2018a).

Devido às características distintas destes modelos híbridos, o método LocLinNonPR foi proposto em (ALMEIDA, 2024), que é a primeira contribuição desta tese. O LocLinNonPR presume que a aplicação de um mapeamento linear e não linear de forma global, ou seja, por meio de modelos individuais treinados em todo o conjunto de treinamento, é insuficiente para reconhecer os padrões locais complexos em séries temporais. Por esse motivo, a inovação do LocLinNonPR é a fusão de um MHP com um MHS para mapear conjuntamente padrões lineares locais e não lineares globais em séries temporais. Primeiramente, é aplicado um MHP homogêneo linear em subconjuntos de treinamento em diferentes regiões de igual tamanho no conjunto de treinamento. Este processo possibilita o reconhecimento de padrões lineares locais na série temporal. Depois, um modelo não linear é aplicado para o mapeamento de padrões não lineares de forma global nos resíduos da série. Os resultados de (ALMEIDA, 2024) em dados automotivos com padrões complexos, evidenciaram que o LocLinNonPR superou MHPs e MHSs quando aplicados em separado. Portanto, mais investigações sobre esse tipo de abordagem híbrida são desejáveis, principalmente em séries temporais de velocidade do vento.

Em (ALMEIDA; de MATTOS NETO; CUNHA, 2024), que é outra contribuição desta tese, presume-se que variar o tamanho e a região dos subconjuntos no conjunto de treinamento em um MHP permitirá um reconhecimento de padrões locais mais apropriado do que abordagens que empregam um mapeamento global em séries temporais de velocidade do vento. Portanto, em (ALMEIDA; de MATTOS NETO; CUNHA, 2024), a principal inovação do método LocPart é durante a fase de geração de um MHP homogêneo, pois foi demonstrado que é importante variar a região e o tamanho dos subconjuntos de treinamento em um MHP.

Portanto, esta Tese de Doutorado investiga o uso de MHPs, MHSs e suas combinações para alcançar um mapeamento mais adequado de padrões locais em séries temporais complexas de velocidade do vento. Desta forma, foi proposto mais um método chamado LocLN, que incorporou a inovação da etapa de geração do MHP homogêneo LocPart em uma técnica de MHS. A hipótese é que se for aplicado um reconhecimento de padrões locais lineares na série temporal, e posteriormente um mapeamento de padrões locais não lineares

nos resíduos, esse processo permitirá um reconhecimento mais apropriado dos padrões locais complexos em séries temporais de velocidade do vento. Ao serem estabelecidos mapeamentos de padrões mais adequados, previsões mais confiáveis serão obtidas para as operações e práticas de planejamento no setor de energia eólica.

Neste trabalho as propostas dos métodos LocLinNonPR e LocPart serão avaliadas ainda mais detalhadamente em dez bases de dados de velocidade do vento, juntamente com a proposta do LocLN. Ademais, outros 13 métodos envolvendo abordagens individuais, MHPs e MHSs serão comparados com as propostas. Por meio dos resultados, pretende-se sugerir algumas considerações sobre os seguintes questionamentos:

- Um método que realiza um mapeamento linear e não linear de padrões de forma global é suficiente para reconhecer os padrões locais complexos em séries temporais de velocidade do vento?
- Variar o tamanho e a região dos subconjuntos no conjunto de treinamento em um MHP homogêneo permitirá um reconhecimento de padrões locais mais apropriado do que abordagens que empregam um mapeamento global do respectivo modelo base em séries temporais de velocidade do vento?
- A aplicação de uma etapa de seleção para encontrar os preditores mais acurados faz sentido para MHPs aplicados em séries temporais de velocidade do vento?
- Ao comparar MHPs com MHSs, qual das duas abordagens é a melhor para previsão de séries temporais de velocidade do vento?
- A aplicação de um reconhecimento de padrões locais lineares na série temporal, e posteriormente um mapeamento de padrões locais não lineares nos resíduos, permitirá um reconhecimento mais apropriado dos padrões locais complexos em séries temporais de velocidade do vento?
- O tamanho da série temporal influencia no desempenho preditivo das propostas?

1.2 Objetivos

O objetivo geral deste trabalho é a proposição de métodos que possibilitem o reconhecimento de padrões locais complexos em séries temporais de velocidade do vento para previsões de um passo à frente e a curto prazo (1h e 3h). O primeiro método é o LocLinNonPR, a sua inovação é a fusão de um MHP com um MHS para possibilitar o mapeamento de padrões lineares locais e padrões não lineares globais em séries temporais. O segundo método, chamado LocPart, introduz uma inovação na etapa de geração de um MHP homogêneo. Nesse processo, a região e o tamanho dos subconjuntos de treinamento são variados, permitindo um reconhecimento mais adequado de padrões locais

complexos nas séries temporais. Por fim, o método LocLN combina o MHP homogêneo LocPart com uma técnica de MHS. Inicialmente, aplica-se o MHP LocPart com modelos base lineares à série temporal. Em seguida, utiliza-se o MHP LocPart com modelos base não lineares para modelar os resíduos da série temporal, aprimorando-se o mapeamento dos padrões locais complexos em séries temporais de velocidade do vento. Portanto, o foco das propostas é aumentar a capacidade preditiva de um passo à frente (1h e 3h), ao superar as limitações dos modelos individuais, MHPs e MHSs que possuem o mapeamento tradicional global de padrões.

O objetivo geral anteriormente definido pode ser decomposto nos seguintes objetivos específicos:

- Compreender inicialmente o funcionamento das turbinas eólicas e seus requisitos para a geração de energia eólica;
- Realizar um estudo sobre as séries temporais de velocidade do vento, características e identificação das fontes de dados, como, por exemplo, o Instituto Nacional de Pesquisas Espaciais (INPE) e o Instituto Nacional de Meteorologia (INMET), e suas respectivas bases de dados, Sistema de Organização Nacional de Dados Ambientais (SONDA) e Banco de Dados Meteorológicos (BDMEP);
- Investigar o estado da arte sobre modelos de previsão de séries temporais na área de energia eólica. Para este fim, foi preciso entender o comportamento e, principalmente, as limitações dos modelos individuais, dos MHPs e dos MHSs;
- Investigar o estado da arte dos MHPs e dos MHSs de uma forma geral em diferentes aplicações;
- Proposição e implementação de métodos por meio de MHPs para identificar os padrões locais nas séries temporais de velocidade do vento;
- Proposição e implementação de métodos que empreguem o mapeamento de padrões locais dos MHPs juntamente com a técnica dos MHSs;
- Escolher e implementar métodos atualizados da literatura para servir de *benchmarks* para as propostas;
- Definir as métricas de qualidade e teste estatístico de hipóteses para avaliação dos resultados;
- Aplicar os métodos implementados e as propostas em séries temporais de velocidade do vento;
- Análise dos resultados em séries temporais de velocidade do vento em períodos do ano variados, e com tamanhos e granularidades diferentes.

1.3 Publicações

A pesquisa desenvolvida no doutorado possibilitou as seguintes publicações científicas:

- ALMEIDA, D. M. A proposal for a fusion of an ensemble with error correction model applied to automotive time series. In: IEEE. 2024 International Joint Conference on Neural Networks (IJCNN). S.l.], 2024. p. 1–7.
- ALMEIDA, D. M.; de MATTOS NETO, P. S. d.; CUNHA, D. C. A new ensemble with partition size variation applied to wind speed time series. In: SPRINGER. International Conference on Hybrid Artificial Intelligence Systems (HAIS). [S.l.], 2024. p. 53–65.
- ALMEIDA, D. M.; de MATTOS NETO, P. S. d.; CUNHA, D. C. A data distribution-based ensemble generation applied to wind speed forecasting. In: SPRINGER. Brazilian Conference on Intelligent Systems (BRACIS). [S.l.], 2024.

1.4 Estrutura do Documento

No Capítulo 2, é apresentada a fundamentação teórica e revisão da literatura. No Capítulo 3, são descritos os métodos propostos. Já no Capítulo 4, estão a metodologia e as avaliações utilizadas nesta pesquisa. Os resultados são apresentados no Capítulo 5. Por fim, no Capítulo 6, são apresentadas as conclusões.

2 FUNDAMENTAÇÃO TEÓRICA E REVISÃO DA LITERATURA

Este capítulo envolve a teoria de séries temporais, e os modelos de previsão individuais e híbridos. Os modelos híbridos com seus objetivos e limitações estão descritos em seções de acordo com a abordagem, que pode ser modelos híbridos em paralelo (MHP) ou modelos híbridos em série (MHS).

2.1 Séries Temporais

Uma série temporal é um conjunto de observações ordenadas no tempo, em períodos regulares, que podem apresentar dependência entre instantes de tempo. De uma maneira um pouco mais formal, uma série temporal discreta é um conjunto de observações y_t, cada uma sendo registrada em um momento t específico, e as observações são feitas em intervalos de tempo fixos (BROCKWELL et al., 2016). Muitas séries temporais observadas na prática são mais adequadamente tratadas como componentes de uma série temporal vetorial (multivariada), caracterizadas tanto pela dependência serial em cada série individual quanto pela interdependência entre as diferentes séries componentes (BROCKWELL et al., 2016). Contudo, neste trabalho o foco foi em séries temporais univariadas, em que a única varável analisada foi a velocidade do vento no decorrer do tempo em intervalos de 1h e 3h.

Os principais objetivos do estudo de séries temporais são a análise e a previsão da série temporal, possibilitando encontrar estimativas de valores futuros a partir de modelos que recebem como entrada valores passados e presentes da série. Este tema tem uma ampla aplicabilidade em diversas áreas de pesquisa. Por exemplo, energia sustentável, principalmente, a energia eólica (MA et al., 2020b; AHMADI; KHASHEI, 2021; FERREIRA; SANTOS; LUCIO, 2019; LIU; LIN; FENG, 2021; TANG et al., 2020). Ademais, engenharia automotiva (ALMEIDA; de MATTOS NETO; CUNHA, 2018), informática em saúde (ZHU et al., 2017) e outros ramos da ciência aplicada que envolvem registros ou medições temporais (ver (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017) e referências contidas).

Antes de obter inferências a partir dos dados da série temporal, é necessário estabelecer um modelo de probabilidade hipotético que represente os dados. Assim, de acordo com (BROCKWELL et al., 2016), após a seleção de uma família de modelos apropriada, torna-se viável a estimação dos parâmetros, a avaliação da qualidade do ajuste do modelo aos dados, e a utilização do modelo ajustado para aprimorar a compreensão do mecanismo que gera a série.

O primeiro procedimento a se fazer ao estudar uma série temporal é construir um gráfico para exibir visualmente a evolução da série ao longo do tempo (BARROS, 2004). A Figura 2 ilustra o gráfico de uma série temporal com um pouco mais de 2.000 observações, que representa a evolução da velocidade do vento em intervalos de 1h no decorrer de três meses na cidade de Triunfo-PE em 2006. Este procedimento simples costuma ser bastante esclarecedor e permite identificar o comportamento da série. Por exemplo, a identificação de sazonalidade, tendência e *outliers*. Ainda pode-se utilizar um método de decomposição conhecido como *Seasonal Trend Decomposition using Loess* (STL) (CLEVELAND et al., 1990), que decompõe a série em três componentes, tais quais, sazonalidade, tendência e resto (componentes aleatórios).

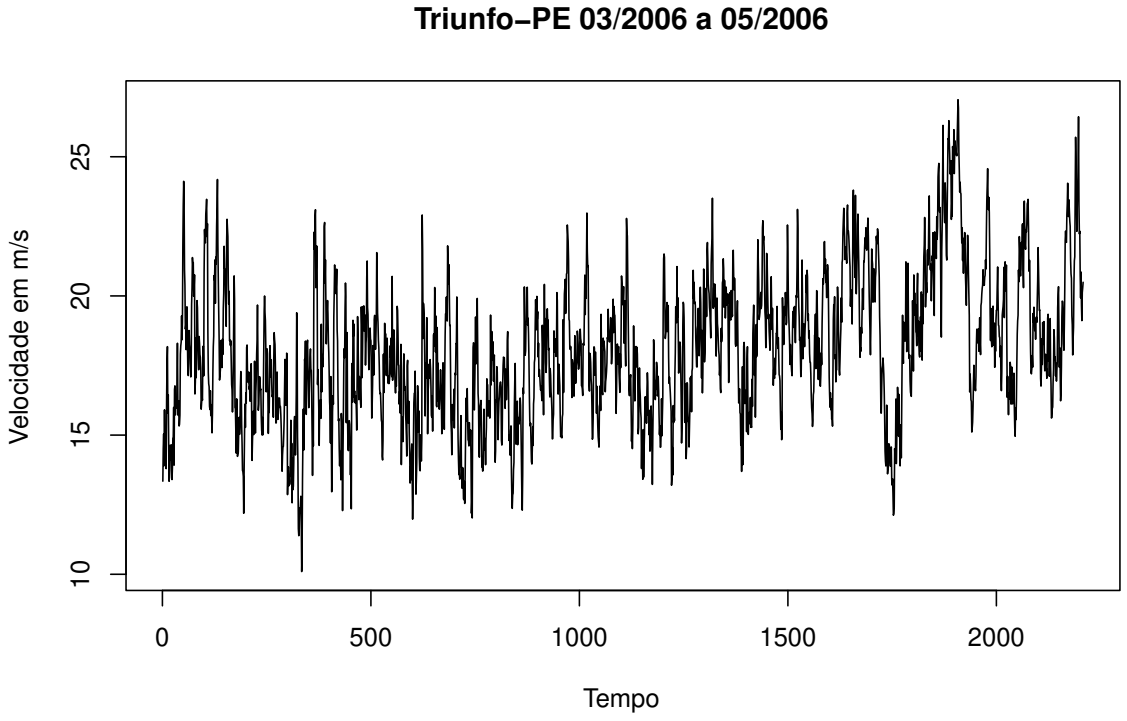

Figura 2 – Evolução da velocidade do vento no decorrer do tempo de três meses em Triunfo-PE.

A sazonalidade é um comportamento das grandezas envolvidas que se repete de forma regular ao longo do tempo em intervalos fixos. Um método para identificar a sazonalidade (padrões de repetição) em uma série temporal é por meio da análise de sua função de autocorrelação, que representa a correlação cruzada do sinal consigo mesmo em diferentes defasagens de tempo. A Figura 3 apresenta o gráfico da função de autocorrelação (ACF, do inglês *Autocorrelation Function*) para uma série temporal. As linhas pontilhadas indicam o intervalo de confiança de 95%. Valores que ultrapassam esse intervalo podem ser interpretados como indícios de padrões repetitivos. Observa-se que à medida que as defasagens aumentam até o valor 15, a amplitude diminui. A partir da defasagem 16, as

amplitudes aumentam, excedendo o limite superior do intervalo de confiança até um pico no valor 24. Esse comportamento aponta para uma evidência de sazonalidade na série temporal, com frequência igual a 24.

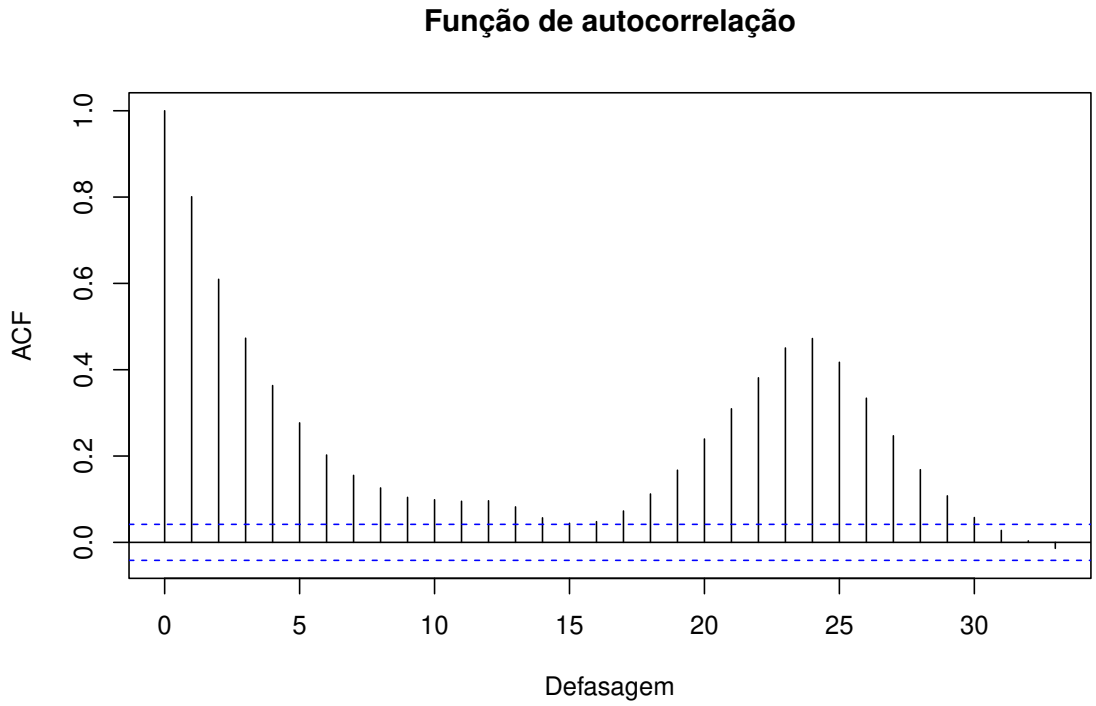

Figura 3 – Função de autocorrelação da série temporal.

A tendência em uma série temporal é definida como um padrão de crescimento ou decrescimento da série em análise em um certo período de tempo. Na Figura 2 é possível notar um padrão de crescimento por volta da observação de número 750 em diante. Logo, é uma série temporal com tendência positiva. *Outliers* são dados que estão fora do padrão comum da série, são valores que fogem da normalidade. Por exemplo, há um momento entre as observações 0 e 500, em que há valores com velocidade próxima de 10m/s. Essa velocidade não se repete mais no gráfico, desta forma, essas observações podem ser consideradas fora do comum.

Ademais, as variações remanescentes após remover os efeitos sazonais e de tendência de uma série são os componentes aleatórios (CLEVELAND et al., 1990). Esses componentes não são precisamente uma realização de um processo aleatório, mas sim uma estimativa dessa realização. É uma estimativa, porque é obtida da série temporal original usando estimativas da tendência e da sazonalidade (COWPERTWAIT; METCALFE, 2009).

2.2 Modelos Individuais de Previsão de Séries Temporais

Nesta seção, serão introduzidos os modelos individuais de previsão de séries temporais relacionados com essa pesquisa.

2.2.1 Modelo ARIMA

O modelo autorregressivo, integrado e de médias móveis (ARIMA, do inglês *Auto Regressive Integrated Moving-Average*) está entre os modelos estatísticos de séries temporais mais conhecidos, e presume que as séries envolvem padrões lineares (AHMADI; KHASHEI, 2021). Ademais, apresenta uma abordagem poderosa para resolver muitos problemas de previsão, pois pode fornecer previsões acuradas de séries temporais (BROCKWELL et al., 2016), principalmente em séries com poucas observações. Inicialmente, sua popularidade surgiu por causa das suas propriedades estatísticas e da conhecida metodologia Box-Jenkins, muito utilizada no processo de construção deste modelo (ZHANG, 2003). Devido às características lineares, o ARIMA é um modelo de fácil interpretação e que requer pouco recurso computacional. Isso o torna, ainda nos dias de hoje, bastante atrativo na modelagem e previsão de séries temporais.

Para descrever a formulação do modelo ARIMA, é necessário antes introduzir dois conceitos importantes: estacionariedade e o ruído branco. De acordo com (DAVID; DAVID, 2015), uma série temporal é considerada fracamente estacionária, se sua média e variância permanecerem constantes ao longo do tempo, e se a autocovariância entre duas amostras depender exclusivamente da defasagem temporal. A autocovariância é definida como a covariância de uma variável aleatória com ela mesma em diferentes defasagens temporais. A covariância, por sua vez, depende tanto das variâncias das variáveis envolvidas quanto da intensidade da relação linear entre elas (DAVID; DAVID, 2015). Uma série temporal é considerada estritamente estacionária se, além de satisfazer os critérios de estacionariedade fraca, sua distribuição de probabilidade for invariante no tempo. A suposição de estacionariedade estrita é considerada extremamente rigorosa, pois requer que todos os aspectos do comportamento estocástico da série sejam constantes ao longo do tempo. O termo 'estacionariedade' é comumente utilizado para se referir à estacionariedade fraca, salvo indicação explícita de que se trata de estacionariedade estrita (BROCKWELL et al., 2016).

Já o ruído branco representa o exemplo mais simples de um processo estacionário (DAVID; DAVID, 2015). Uma série é caracterizada como um processo de ruído branco fraco se apresentar média e variância constantes, e se sua função de autocovariância for igual a zero. A ausência de autocovariância implica que as variáveis são independentes. Assim, o ruído branco fraco constitui um caso particular de um processo fracamente estacionário em que as variáveis são independentes. Quando as variáveis do ruído branco são identicamente distribuídas, o ruído branco é estritamente estacionário. Ademais, se as variáveis identicamente distribuídas seguirem uma distribuição normal, o processo é denominado ruído branco gaussiano. Devido à ausência de dependência, os valores passados de um processo de ruído branco não contêm informações úteis para a previsão de valores futuros (DAVID; DAVID, 2015).

Uma forma de validar os requisitos para a estacionariedade é a partir de testes de hipó-

teses, que consistem em verificar se a função que representa a série contém raiz unitária nos operadores de retardo, pois isto indica um processo com médias que mudam com o tempo, e assinala a não estacionariedade. Testes de hipóteses para a presença de raiz unitária foram introduzidos por Dickey e Fuller em 1979 (BROCKWELL; DAVIS, 2002). Um exemplo é o teste de Dickey-Fuller aumentado (ADF, do inglês *Augmented Dickey Fuller*), que é comumente usado para testar a estacionariedade em séries temporais financeiras (LI et al., 2017). Outros testes também devem ser utilizados juntamente com o ADF, para a análise ficar mais robusta, por exemplo, os testes de Phillips–Perron (PP) (PHILLIPS; PERRON, 1988) e Kwiatkowski-Phillips-Schmidt-Shin (KPSS) (KWIATKOWSKI et al., 1992). Caso os testes indiquem a não estacionariedade, é possível fazer um procedimento para transformar uma série não estacionária em uma série estacionária: obtêm-se as diferenças entre os valores das amostras da série não estacionária. Por exemplo, a Equação (2.1) apresenta uma série diferenciada W_t a partir de uma série y_t.

$$W_t = \nabla y_t = y_t - y_{t-1}. \tag{2.1}$$

A partir dos conceitos definidos anteriormente, é possível introduzir a formulação do modelo autorregressivo de médias móveis (ARMA, do inglês *Autoregressive Moving Average*), muito utilizado para análise de séries temporais e também popularizado por Box e Jenkins (BOX; JENKINS; REINSEL, 2008). Este modelo somente é adequado para séries estacionárias e é obtido a partir da adição de termos autorregressivos (AR, do inglês *Autoregressive*) e médias móveis (MA, do inglês *Moving Average*). Assim, o valor futuro de uma variável é assumido como uma função linear de várias observações passadas, que são os termos AR, e vários erros aleatórios (ruído branco) no passado, que são os termos MA. Por isso, o processo que gera a série temporal em um modelo ARMA tem a forma (COWPERTWAIT; METCALFE, 2009):

$$y_t = \phi_1 y_{t-1} + \phi_2 y_{t-2} + ... + \phi_p y_{t-p} + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + ... + \theta_q \varepsilon_{t-q}, \tag{2.2}$$

em que, y_t e ε_t são o valor real da série e o erro aleatório no tempo t, respectivamente. ϕ_i $(i = 1, 2, ..., p)$ e θ_j $(j = 1, 2, ..., q)$ são os parâmetros do modelo, tal que p e q são inteiros referentes às ordens dos termos AR e MA, respectivamente.Os erros aleatórios ε_t são assumidos como um processo de ruído branco, em que as variáveis são independentes, identicamente distribuídas, com média zero e variância constante.

Outra questão importante é sobre a estacionariedade e invertibilidade do modelo ARMA. Segundo (HYNDMAN; ATHANASOPOULOS, 2021), a estacionariedade de modelos tem a mesma definição de estacionariedade em séries temporais, ou seja, um modelo estacionário é aquele que possui média e variância constantes ao longo do tempo, e sua autocovariância depende apenas da defasagem temporal. A invertibilidade de um modelo ARMA ocorre quando a componente MA pode ser representada como uma combinação infinita de valores no presente e passado da componente AR (BROCKWELL et al., 2016).

Uma consequência da invertibilidade é que as observações mais recentes têm peso maior do que as observações do passado mais distante (HYNDMAN; ATHANASOPOULOS, 2021). Desta forma, modelos ARMA estacionários e invertíveis são considerados modelos estáveis e apropriados para previsão de séries temporais estacionárias (HYNDMAN; ATHANASOPOULOS, 2021).

Determinados os valores de p e q, o objetivo seguinte é a estimação dos parâmetros ϕ_i $(i = 1, 2, ..., p)$ e θ_j $(j = 1, 2, ..., q)$. Geralmente, são utilizados os estimadores de máxima verossimilhança (MLE, do inglês *Maximum Likelihood Estimator*), ou soma condicional de quadrados (CSS, do inglês *Conditional Sum of Squares*). No método MLE, os valores dos parâmetros são estimados de forma que maximizem a probabilidade de que o processo ARMA produza os dados que foram realmente observados. Idealmente, os dados devem seguir uma distribuição normal para uma estimação adequada dos parâmetros. Já o método CCS, ajusta o modelo minimizando a soma condicional dos quadrados e as estimativas são condicionais no pressuposto de que os erros passados não observados são iguais a zero. Enquanto o MLE é considerado um método mais preciso para dados que seguem uma distribuição normal, o CSS é um método alternativo, sem o requisito de normalidade dos dados, e além disso, apresenta um custo computacional menor.

Quando os dados temporais não são estacionários, a recomendação, conforme mencionado anteriormente, é tomar as diferenças até que a estacionariedade seja alcançada e, em seguida, prossegue-se ajustando um modelo ARMA aos dados diferenciados. Um modelo para um processo cuja d-ésima diferença segue um modelo ARMA (p, q) é chamado de processo ARIMA de ordem (p, d, q), ou ARIMA (p, d, q) (PETRIS; PETRONE; CAMPAGNOLI, 2009).

Com relação a séries temporais de velocidade do vento, o modelo ARIMA e suas variantes foram bastante utilizados. Em (KAMAL; JAFRI, 1997), foi demonstrado que os modelos ARMA são úteis para previsões de curto alcance (1-6 horas) da velocidade do vento. Em (ERDEM; SHI, 2011) também foi realizada uma análise da previsão de curto alcance (1 hora) de modelos ARMA para a velocidade do vento, comparou com modelos ARMA vetorial, e concluiu que o modelo ARMA supera o ARMA vetorial. Em (TORRES et al., 2005) foi realizado um comparativo entre o modelo de persistência (*random walk*) e modelos ARMA. O modelo de persistência assume que a previsão no período seguinte será igual ao valor do período mais recente. Os resultados indicaram que os modelos ARMA fornecem melhorias significativas sobre o modelo de persistência, mesmo para curtos períodos de previsão. Para previsões de prazos mais longos, os autores em (CADENAS; RIVERA, 2007) demonstram que o modelo ARIMA sazonal pode ser usado para prever, de forma razoável, a velocidade do vento mensal, e supera o modelo de persistência. Em (KAVASSERI; SEETHARAMAN, 2009) foi examinado o uso do modelo ARIMA fracionário para prever velocidades do vento no dia seguinte (24 horas) e até dois dias seguintes (48 horas). Os resultados indicaram que são obtidas melhorias significativas na acurácia das previsões

do modelo ARIMA fracionário em relação ao modelo de persistência.

Embora modelos ARIMA e suas variantes sejam bastante utilizados em uma ampla gama de séries temporais, e entre elas séries de velocidade do vento, sua principal limitação é a forma linear presumida do modelo. Ou seja, uma estrutura de autocorrelação linear é assumida antes que o modelo seja ajustado aos dados. Portanto, um modelo ARIMA não é capaz de modelar padrões não lineares (ZHANG, 2004). Neste contexto, modelos não lineares são uma alternativa para análise e previsão de séries temporais (ZHU; WEI, 2013).

2.2.2 Modelo MLP

No caso de modelos não lineares, as abordagens baseadas em Rede Neural Artificial (RNA) têm resultados significativos em relação à acurácia (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017). Entre essas abordagens, a perceptron multicamada (MLP, do inglês *Multilayer Perceptron*) com uma única camada oculta é amplamente utilizada em previsões de velocidade do vento (HUANG; WANG; HUANG, 2021b), (LI et al., 2018), (CAMELO et al., 2018b). A MLP é uma RNA *feedforward* que mapeia dados de entrada para um ou mais conjuntos de saídas. No caso particular da previsão de série temporal univariada, em geral, os dados de entrada são as defasagens no tempo e a saída desejada é o valor futuro. O modelo MLP é uma RNA composta por pelo menos três camadas (entrada, oculta e saída) conectadas totalmente e diretamente em apenas uma direção. Embora existam novos e promissores modelos de RNA na literatura, a MLP ainda é uma ferramenta amplamente utilizada no desenvolvimento de sistemas inteligentes para modelagem de séries temporais. Principalmente, séries complexas do mundo real, devido a sua acurácia e simplicidade (GUO et al., 2018; ZHANG, 2003). A relação entre a saída (y_t) e as entradas ($y_{t-1}, y_{t-2}, ..., y_{t-p}$) da MLP com uma única camada oculta tem a seguinte representação matemática (ZHANG, 2003):

$$y_t = \tau_0 + \sum_{j=1}^{q} \tau_j g\left(\beta_{0j} + \sum_{i=1}^{p} \beta_{ij} y_{t-i}\right) + \varepsilon_t, \tag{2.3}$$

em que $\tau_j (j = 0, 1, 2, ..., q)$ e $\beta_{ij} (i = 0, 1, 2; ..., p, j = 1, 2, ..., q)$ são os parâmetros do modelo normalmente chamados de pesos de conexão, p é o número de nós de entrada e q é o número de nós ocultos. A função logística é frequentemente usada como função de transferência da camada oculta e é dada por (ZHANG, 2003):

$$g(x) = \frac{1}{1 + e^{-x}}. \tag{2.4}$$

Desta forma, o modelo RNA definido em (2.3) realiza um mapeamento não linear a partir das observações passadas ($y_{t-1}, y_{t-2}, ..., y_{t-p}$) para obter o valor futuro y_t um passo à frente (ZHANG, 2003):

$$y_t = f(y_{t-1}, y_{t-2}, ..., y_{t-p}; \mathbf{v}) + \varepsilon_t, \tag{2.5}$$

em que $\mathbf{v}$ é um vetor de todos os parâmetros e f é uma função determinada pela estrutura da rede e pesos de conexão. Desta forma, a MLP é equivalente a um modelo autorregressivo não linear.

Uma vez que a estrutura da rede é especificada, o próximo passo é o treinamento. Assim, os pesos de conexão são estimados de forma que um critério geral de acurácia, como erro quadrático médio, seja minimizado (ZHANG, 2003). Um algoritmo muito popular para essa etapa é o *backpropagation*. Neste algoritmo, a minimização da função de erro é realizada usando uma técnica de gradiente descendente. As correções necessárias aos pesos da rede para cada momento t são obtidas calculando-se a derivada parcial da função de erro em relação a cada peso de conexão (BARBALATA; LEUSTEAN, 2004). Contudo, esse método é considerado instável por não possuir uma atualização de peso apropriada, resultando em oscilações e passos desnecessários para alcançar uma solução aceitável (RIEDMILLER; BRAUN, 1993).

Uma alternativa ao *backpropagation* é o algoritmo *Resilient Propagation* (RPROP), que propõe uma convergência mais rápida. O RPROP é um esquema de aprendizado eficiente, que realiza uma adaptação direta da etapa de peso com base em informações do gradiente local. Uma diferença crucial para o *backpropagation*, é que o esforço de adaptação não é comprometido pelo comportamento do gradiente (RIEDMILLER; BRAUN, 1993). O princípio básico do algoritmo RPROP é eliminar a influência prejudicial do tamanho da derivada parcial na etapa de peso. Como consequência, apenas o sinal da derivada é considerado para indicar a direção da atualização de peso (BARBALATA; LEUSTEAN, 2004). Além disso, diferentemente do *backpropagation*, no algoritmo RPROP cada peso de conexão tem seu valor de atualização individual, que determina apenas o tamanho da atualização do peso (RIEDMILLER; BRAUN, 1993). Portanto, o RPROP é um algoritmo com um tratamento mais sofisticado e apropriado para a atualização dos pesos da rede, bem como, mais aprimorado para alcançar uma solução aceitável.

O algoritmo RPROP usa três parâmetros, quais sejam, valor inicial de atualização $\triangle_0$, valor máximo para o tamanho do passo $\triangle_{max}$, e o expoente de decaimento de peso μ (ZELL et al., 2008). Quando o aprendizado é iniciado, todos os valores de atualização são definidos para um valor inicial $\triangle_0$. Para evitar que os pesos se tornem muito grandes, o peso máximo determinado pelo tamanho do valor de atualização é limitado. O limite superior é definido pelo segundo parâmetro do RPROP, $\triangle_{max}$. O parâmetro de decaimento de peso μ determina a relação de dois objetivos: reduzir o erro de saída da rede neural (a meta padrão) e reduzir o tamanho dos pesos (para melhorar a generalização) (ZELL et al., 2008). A formulação da função de erro E é dada por:

$$E = \sum_{t=1}^{n}(y_t - \widehat{y}_t)^2 + 10^{-\mu}\sum_{i,j}\pi_{ij}^2, \tag{2.6}$$

em que y_t e $\widehat{y}_t$ são o valor objetivo e a resposta do t-ésimo valor de saída da MLP, respectivamente; n é o número de instâncias no conjunto de treinamento; π_{ij} é o peso de

conexão entre o i-ésimo e j-ésimo neurônios das diferentes camadas da MLP.

Na área de energia eólica, os modelos MLP são bastante utilizados, principalmente na previsão de velocidade do vento. Em (ALEXIADIS et al., 1998), modelos MLP foram propostos para prever valores de velocidade do vento a curto prazo, 10 minutos e 1 hora. Os autores concluíram que a previsão da velocidade do vento foi satisfatória usando MLP, e que superou o desempenho preditivo de modelos ARMA. Em (SFETSOS, 2002), foi verificado que o modelo MLP superou os modelos ARIMA e previsão ingênua para previsões de velocidade do vento com o prazo de 1 hora. Em (FLORES; TAPIA; TAPIA, 2005), os autores fizeram um estudo do modelo MLP para previsão de velocidade do vento a curto prazo, 1 minuto a 1 hora. O desempenho do modelo foi avaliado como apropriado.

2.2.3 Modelo SVM

O modelo máquina de vetores de suporte (SVM, do inglês *Support Vector Machine*) utiliza a técnica de minimização do risco estrutural (SRM, do inglês *Structural Risk Minimization*), o que significa que minimiza um limite superior do erro de generalização (ZHU; WEI, 2013). Essa minimização pode causar uma resistência ao problema de *overfitting*. Na previsão de séries temporais, a regressão SVM, também chamada de regressão por vetores de suporte (SVR, do inglês *Support Vector Regression*), tem bom desempenho em aplicações do mundo real (SOUALHI; MEDJAHER; ZERHOUNI, 2015; GUO et al., 2018). Contudo, o processo de treinamento do SVM pode levar muito tempo, pois utiliza Programação Quadrática (PQ).

O modelo ϵ-*Support Vector Regression* (ϵ-SVR) está entre os tipos de SVR mais utilizados. O ϵ-SVR utiliza uma margem de tolerância ϵ, que pode ser violada por meio de duas variáveis de folga ξ_t^+ e ξ_t^-. A seguir, é introduzida a formulação.

De acordo com (WANG; HU, 2005), dado o conjunto de dados

$$D = \{(\boldsymbol{x}_1, y_1), ..., (\boldsymbol{x}_t, y_t), ..., (\boldsymbol{x}_n, y_n)\}, \boldsymbol{x}_t \in R^n, y_t \in R. \tag{2.7}$$

em que $\boldsymbol{x}_t$ é a variável de entrada representando as defasagens de y_t, e o y_t é a variável de saída. Objetiva-se aproximá-lo a uma função não linear:

$$f(\boldsymbol{x}) = \langle \boldsymbol{\omega}, \varphi(\boldsymbol{x}) \rangle + b, \tag{2.8}$$

em que $\langle \cdot, \cdot \rangle$ denota o produto escalar; $\boldsymbol{\omega} \in R^{n_h}$ é o vetor peso; $\varphi(\cdot) : R^n \rightarrow R^{n_h}$ é a função não linear que mapeia o espaço de entrada para o chamado espaço característico de alta dimensão onde a regressão linear é executada; b é o *bias*. A dimensão n_h deste espaço é implicitamente definida, o que significa que pode ser de dimensão infinita (WANG; HU, 2005).

O problema de otimização do ϵ-SVR é dado por:

$$min \frac{1}{2}||\boldsymbol{\omega}||^2 + C_s \sum_{t=1}^{n} (\xi_t^+ + \xi_t^-), \tag{2.9}$$

com as restrições:

$$\begin{cases} y_t - \langle \boldsymbol{\omega}, \varphi(\boldsymbol{x}_t) \rangle - b & \leq \epsilon + \xi_t^+; \\ \langle \boldsymbol{\omega}, \varphi(\boldsymbol{x}_t) \rangle + b - y_t & \leq \epsilon + \xi_t^-; \\ \xi_t^+, \xi_t^- & \geq 0. \end{cases} \tag{2.10}$$

e a função de perda ϵ-insensível

$$|y - f(\boldsymbol{x}, \boldsymbol{\omega})|_\epsilon = \begin{cases} 0, & \text{se } |y - f(\boldsymbol{x}, \boldsymbol{\omega})| \leq \epsilon; \\ |y - f(\boldsymbol{x}, \boldsymbol{\omega})| - \epsilon, & \text{caso contrário.} \end{cases} \tag{2.11}$$

em que $|\cdot|_\epsilon$ é a chamada função de perda ϵ-insensível, e ϵ é uma margem de tolerância ajustada para tolerar o desvio da regressão dos valores reais. Segundo (SMOLA; SCHÖLKOPF, 2004), a região limitada por $|y - f(\boldsymbol{x}, \boldsymbol{\omega})| \leq \epsilon$ é chamada de tubo ϵ-insensível. As observações que estão fora do tubo, recebem uma das duas variáveis de penalidade, dependendo de estarem acima ξ_t^+ ou abaixo ξ_t^- do tubo. O parâmetro $C_s > 0$ determina o equilíbrio entre o excesso de ajustes de f e o grau em que os desvios são maiores do que os tolerados (penalidades ξ_t^+ e ξ_t^-) na formulação de otimização. Um valor menor de C_s tolera um desvio maior (WANG; HU, 2005).

De acordo com (WANG; HU, 2005), ao reformular o problema de otimização como uma Lagrangiana L_{svm}, tem-se:

$$\begin{aligned} L_{svm} = \quad & \tfrac{1}{2}||w||^2 + C_s \textstyle\sum_{t=1}^n (\xi_t^+ + \xi_t^-) \\ & - \textstyle\sum_{t=1}^n \alpha_t^+ (\epsilon + \xi_t^+ - y_t + \langle \boldsymbol{\omega}, \varphi(\boldsymbol{x}_t) \rangle + b) \\ & - \textstyle\sum_{t=1}^n \alpha_t^- (\epsilon + \xi_t^- + y_t - \langle \boldsymbol{\omega}, \varphi(\boldsymbol{x}_t) \rangle - b) \\ & - \textstyle\sum_{t=1}^n (\eta_t^+ \xi_t^+ + \eta_t^- \xi_t^-), \end{aligned} \tag{2.12}$$

em que $\alpha^+, \alpha^-, \eta_t^+, \eta_t^- \geq 0$ são multiplicadores de Lagrange. Para encontrar o ponto de sela, obtêm-se as derivadas parciais de L_{svm} com relação às variáveis $(\boldsymbol{\omega}, b, \xi_t^+, \xi_t^-)$ (WANG; HU, 2005):

$$\begin{cases} \frac{\partial L_{svm}}{\partial w} = 0 \rightarrow & w = \sum_{t=1}^n (\alpha_t^+ - \alpha_i^-) \varphi(\boldsymbol{x}_t); \\ \frac{\partial L_{svm}}{\partial b} = 0 \rightarrow & \sum_{i=1}^n (\alpha_t^- - \alpha_t^+) = 0; \\ \frac{\partial L_{svm}}{\partial \xi_t^+} = 0 \rightarrow & C_s - \alpha_t^+ - \eta_t^+ = 0; \\ \frac{\partial L_{svm}}{\partial \xi_t^-} = 0 \rightarrow & C_s - \alpha_t^- - \eta_t^- = 0. \end{cases} \tag{2.13}$$

As condições para otimalidade produzem o seguinte problema na forma dual (WANG; HU, 2005):

$$\max_{\alpha_t^+, \alpha_t^-} Q = \sum_{t=1}^n (\alpha_t^+ - \alpha_t^-) y_t - \epsilon \sum_{t=1}^n (\alpha_t^+ - \alpha_t^-) - \frac{1}{2} \sum_{t,j=1}^n (\alpha_t^+ - \alpha_t^-)(\alpha_j^+ - \alpha_j^-) K(\boldsymbol{x}_t, \boldsymbol{x}_j), \tag{2.14}$$

α_t^+ e α_t^- são obtidos pela aplicação de um PQ *solver* em (2.14) com as restrições:

$$\begin{aligned} 0 \leq \alpha_i^+, \alpha_t^- \leq C_s, t = 1, ..., n. \\ \textstyle\sum_{t=1}^n (\alpha_t^+ - \alpha_t^-) = 0 \forall t. \end{aligned} \tag{2.15}$$

De acordo com (WANG; HU, 2005), no final, ao aplicar a condição de Mercer, o ϵ-*SVR* para estimativa de função não linear toma a forma

$$f(x) = \sum_{t=1}^{n}(\alpha_t^+ - \alpha_t^-)K(\boldsymbol{x}, \boldsymbol{x}_t) + b, \tag{2.16}$$

em que $K(x, x_t)$ é chamada de função *kernel*. Na prática, entre as funções *kernel* mais utilizadas está a função de base radial (RBF, do inglês *Radial-Basis Function*), conhecida também como Gaussiana. A RBF tem a forma de uma função de base radial, ou mais especificamente, uma função gaussiana, e é formulada a seguir:

$$K(\boldsymbol{x}, \boldsymbol{x}_k) = \exp(-\lambda||\boldsymbol{x} - \boldsymbol{x}_t||^2), t = 1, ..., n. \tag{2.17}$$

em que λ é o inverso do desvio padrão σ do *kernel* RBF, tal que

$$\lambda = \frac{1}{2\sigma} \ . \tag{2.18}$$

O parâmetro b em (2.16) pode ser calculado a partir das condições de Karush-Kuhn-Tucker (KKT). Portanto, de (WANG; HU, 2005):

$$\begin{aligned} b &= y_t - \langle \boldsymbol{\omega}, \varphi(\boldsymbol{x}_t) \rangle - \epsilon, \alpha_t^+ \in (0, C_s); \\ b &= y_t - \langle \boldsymbol{\omega}, \varphi(\boldsymbol{x}_t) \rangle + \epsilon, \alpha_t^- \in (0, C_s). \end{aligned} \tag{2.19}$$

Na área de energia eólica, o modelo de regressão SVM é muito encontrado na literatura para prever a velocidade do vento. Em (MOHANDES et al., 2004), o artigo introduz SVM para previsão da velocidade do vento em um curto prazo de 24 horas, e compara seu desempenho com o modelo MLP. Os resultados mostraram que o SVM superou o MLP na capacidade preditiva. Em (SALCEDO-SANZ et al., 2011), os autores concluíram que o SVM tem melhores resultados que a MLP para previsões de velocidade do vento no prazo de 1 hora. Em (KONG et al., 2015), é proposto um novo modelo SVM, o SVM reduzido ou RSVM, para previsão de velocidade do vento no prazo de alguns minutos. O importante a ser destacado neste artigo, é que o modelo SVM superou a acurácia da RNA com função de ativação de base radial.

2.2.4 Modelo ELM

Máquinas de aprendizado extremo (ELM, do inglês *Extreme Learning Machine*) são um método rápido de aprendizado baseado na estrutura da MLP com uma única camada oculta, proposto em (HUANG; ZHU; SIEW, 2006). Um dos problemas da MLP é o *backpropagation*, que tem um custo computacional de treinamento considerado alto. O ELM, em vez de utilizar a iteratividade do *backpropagation* para seleção dos pesos da rede, emprega um método analítico de um único passo, a matriz inversa de Moore-Penrose.

Comumente aplicada para calcular uma solução por meio dos mínimos quadrados em um sistema de equações lineares, a matriz inversa de Moore-Penrose pode ser resolvida

por vários métodos. A decomposição em valores singulares (SVD, do inglês *Singular Value Decomposition*) geralmente consegue resolver na maior parte dos casos (HUANG; ZHU; SIEW, 2006).

O ELM pode ser resumido da seguinte forma (HUANG; ZHU; SIEW, 2006):

1. Atribuir aleatoriamente os pesos dos nós de entrada e o *bias*;
2. Obter a matriz de saída da camada oculta;
3. Calcular os pesos de saída da camada oculta por meio da matriz inversa de Moore-Penrose.

A função logística, assim como no MLP, também é frequentemente usada no ELM como função de transferência da camada oculta, definida na função da Equação 2.4.

A simplicidade do ELM é uma grande vantagem, permite um treinamento extremamente rápido, sem perder a competitividade para a MLP. Em algumas aplicações, apresenta até um melhor resultado preditivo do que o MLP e o SVR (SAAVEDRA-MORENO et al., 2013).

Na área de geração de energia eólica, o trabalho de (SAAVEDRA-MORENO et al., 2013) fez um comparativo entre cinco modelos, entre eles, MLP, SVM e ELM, em oito bases de dados de velocidade do vento. Constatou-se que o tempo de treinamento do ELM é muito menor, e os erros de previsão são similares aos demais modelos de aprendizado de máquina. Em (WU; WANG; CHENG, 2013) e (NIKOLIĆ et al., 2016) foram apresentados métodos baseados em ELM para a estimativa da velocidade do vento utilizando os parâmetros de turbinas eólicas. Os resultados demonstraram que os métodos propostos por meio do ELM foram muito mais rápidos e precisos em relação ao SVR e MLP para estimativa da velocidade do vento. Ademais, em (ABDOOS, 2016), foi realizado um estudo com ELM para prever a geração de energia eólica a curto prazo. O método com ELM também apresentou melhor precisão em comparação com MLP e SVR.

2.2.5 Modelo LSTM

O modelo de memória de curto e longo prazo (LSTM, do inglês *Long Short Term Memory*) foi desenvolvido por (HOCHREITER; SCHMIDHUBER, 1997). Entre os principais objetivos das redes LSTM está a captura e o rastreamento de dependências temporais de curto e longo prazo. Além disso, a mitigação do problema de desaparecimento do gradiente, frequentemente observado durante o processo de treinamento de redes mais tradicionais, por exemplo, redes do tipo MLP. O desaparecimento de gradiente acontece quando as atualizações dos pesos de uma RNA tornam-se muito pequenas e, no pior dos casos, interrompe a aprendizagem. A LSTM possui um algoritmo eficiente baseado em gradiente com uma arquitetura que impõe um fluxo de erro constante por meio de estados internos

de unidades especiais (HOCHREITER; SCHMIDHUBER, 1997). Portanto, diferentemente de uma MLP, o gradiente da LSTM não explode, nem desaparece.

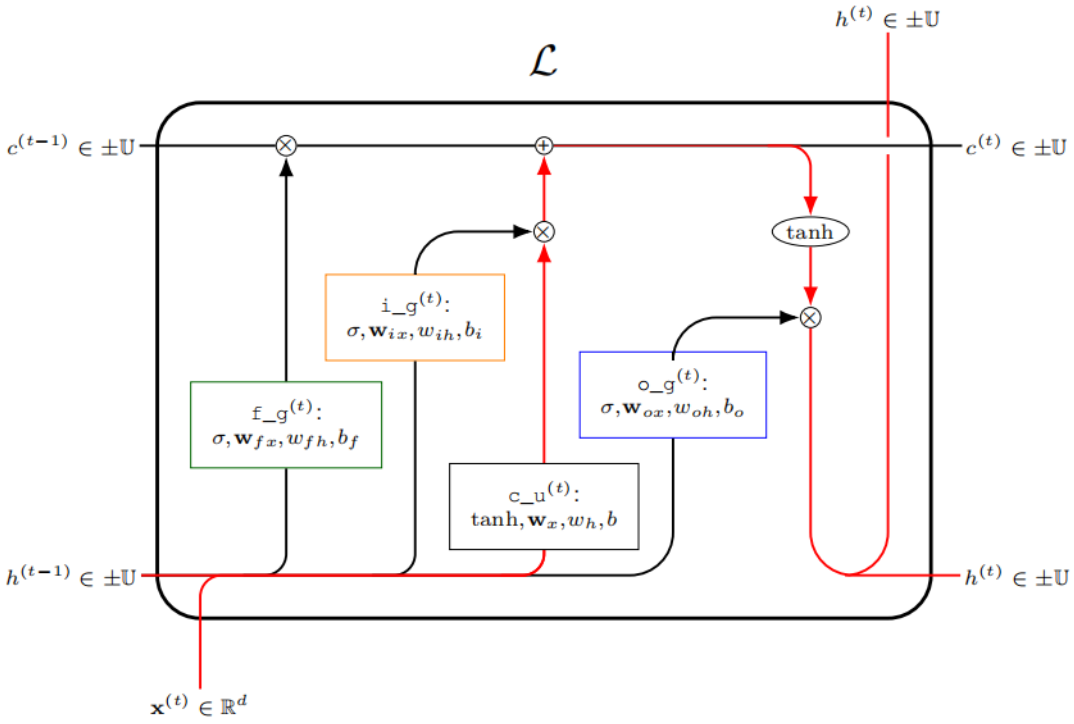

Figura 4 – Ilustração de uma célula LSTM. As setas vermelhas mostram a unidade neural recorrente com duas funções de ativação *tanh*. Os três portões – esquecer (verde), entrada (laranja) e saída (azul) – controlam as interações entre o estado da célula e o estado oculto (ELSWORTH; GÜTTEL, 2020).

Fonte: (ELSWORTH; GÜTTEL, 2020).

Uma célula LSTM tem dois recursos recorrentes, denotados por h e c, chamados de estado oculto e estado da célula, respectivamente. A Figura 4 ilustra a função L de uma célula LSTM. A função matemática L recebe três entradas e produz duas saídas (ELSWORTH; GÜTTEL, 2020), e de acordo com a nomenclatura da Figura 4:

$$(h^{(t)}, c^{(t)}) = L(h^{(t-1)}, c^{(t-1)}, x^{(t)}), \tag{2.20}$$

em que $x^{(t)}$ é uma observação de entrada no tempo t na célula. Ambas as saídas $h^{(t)}$ e $c^{(t)}$ deixam a célula no tempo t e são realimentadas para a mesma célula no tempo $t+1$. As entradas $h^{(t-1)}$ e $c^{(t-1)}$ são as saídas da mesma célula no tempo $t-1$.

Dentro da célula, há três portões (funções) que dependem do vetor de entrada e do estado oculto. Cada um produz valores no intervalo [0,1] com a ajuda da função de ativação sigmoide σ_g (ELSWORTH; GÜTTEL, 2020):

$$f\_g^{(t)}(\boldsymbol{x}^{(t)}, h^{(t-1)}) = \sigma_g(\boldsymbol{w}^T{}_{f,x}\boldsymbol{x}^{(t)} + w_{f,h}h^{(t-1)} + b_f) \in [0,1] \tag{2.21}$$

$$i\_g^{(t)}(\boldsymbol{x}^{(t)}, h^{(t-1)}) = \sigma_g(\boldsymbol{w}^T{}_{i,x}\boldsymbol{x}^{(t)} + w_{i,h}h^{(t-1)} + b_i) \in [0,1] \tag{2.22}$$

$$o\_g^{(t)}(\boldsymbol{x}^{(t)}, h^{(t-1)}) = \sigma_g(\boldsymbol{w}^T{}_{o,x}\boldsymbol{x}^{(t)} + w_{o,h}h^{(t-1)} + b_o) \in [0,1] \tag{2.23}$$

em que $\boldsymbol{w}_{f,x}$, $\boldsymbol{w}_{i,x}$, $\boldsymbol{w}_{o,x} \in R^d$ e $w_{f,x}$, $w_{i,x}$, $w_{o,x}$, b_f, b_i, $b_o \in R$ são os vetores de peso e *biases*. d é a dimensão dos vetores. O portão de esquecimento (f_g) controla quanto do estado atual a célula deve esquecer, o portão de entrada (i_g) controla quanto da atualização da célula deve ser adicionada ao estado da célula e o portão de saída (o_g) controla quanto do estado da célula modificado deve deixar a célula e se tornar o próximo estado oculto. Outra função importante é a célula de atualização, que é definida por (ELSWORTH; GÜTTEL, 2020):

$$c\_u^{(t)}(x^{(t)}, h^{(t-1)}) = tanh(\boldsymbol{w}^T{}_x\boldsymbol{x}^{(t)} + w_h h^{(t-1)} + b) \in [-1,1] \tag{2.24}$$

Logo, a nova célula e os estados ocultos no tempo t serão (ELSWORTH; GÜTTEL, 2020):

$$c^{(t)} = f\_g^{(t)} \cdot c^{(t-1)} + i\_g^{(t)} \cdot c\_u^{(t)} \in [-1,1] \tag{2.25}$$

$$h^{(t)} = o\_g^{(t)} \cdot tanh(c^{(t)}) \in [-1,1] \tag{2.26}$$

em que os argumentos ($x^{(t)}$, $h^{(t-1)}$) foram omitidos para facilitar a leitura. $c^{(t)}$ e $h^{(t)}$ são duas funções de entrada da função L da Equação 2.20.

Para introduzir as redes LSTM *stateless* e *stateful*, é importante entender os conceitos de *batch* e de época. Primeiramente, tamanho do *batch* é a quantidade de observações que devem ser enviadas à LSTM para que ocorra uma atualização dos pesos da rede neural. E, uma época consiste em passar todas as observações do conjunto de treinamento uma vez na rede neural.

Um procedimento de treinamento *stateless* sempre leva os estados iniciais $H^{(0)}$ e $C^{(0)}$ para zero após um *batch*. Em uma rede *stateless*, cada *batch* do conjunto de treinamento é independente de todos os outros *batches* e, portanto, os *batches* podem ser embaralhados.

O procedimento de treinamento *stateful* tenta explorar totalmente a memória da rede (ELSWORTH; GÜTTEL, 2020). Isso é possível enviando os estados $H^{(t)}$ e $C^{(t)}$ do *batch* em n para o próximo *batch* em $n+1$, e assim sucessivamente. Desta forma, o modelo LSTM pode capturar a dependência temporal não apenas dentro dos *batches*, mas também entre os *batches*. Na LSTM *stateful* é recomendável, após uma época, zerar os estados $H^{(t)}$ e $C^{(t)}$ da LSTM. Uma observação interessante, uma rede *stateless* pode simular uma rede *stateful*, para isso, basta definir o tamanho do *batch* igual ao tamanho do conjunto de treinamento.

Os resultados do LSTM em diferentes áreas têm mostrado que o LSTM supera os modelos de redes neurais MLP, principalmente, em problemas que envolvem dependências temporais de longo prazo (SONG et al., 2020). Por exemplo, na área de extração de petróleo, em (SONG et al., 2020) é realizado um comparativo de modelos entre LSTM,

ARIMA, MLP e rede neural recorrente (RNN, do inglês *Recurrent Neural Network*). Os resultados mostraram que o LSTM apresentou os menores erros para prever a taxa diária de extração de petróleo.

Na área de energia eólica, em (MEMARZADEH; KEYNIA, 2020) é proposto um modelo híbrido que envolve decomposição, seleção de recursos e modelos LSTM. É importante destacar que o LSTM apresentou melhores resultados preditivos que o MLP nas bases de dados de velocidade do vento para previsões de 1h. Em (LIU; LIN; FENG, 2021) foi proposto um novo modelo ARIMA sazonal para prever séries temporais de velocidade do vento para o horizonte de 1h. E, apesar do LSTM ser um algoritmo de aprendizado de máquina avançado, o ARIMA sazonal mostrou ter um desempenho preditivo superior em três bases de dados nos resultados de (LIU; LIN; FENG, 2021).

2.2.6 Modelo GRU

O modelo de unidade recorrente fechada (GRU, do inglês *Gated Recurrent Units*) similarmente ao LSTM, também tem o objetivo de rastrear dependências de longo prazo, além de contornar o problema de desaparecimento de gradiente. Porém, diferentemente da célula LSTM, que possui três portões, a célula GRU é mais compacta e simples, possuindo somente dois portões (funções): o portão de atualização e o portão de reinicialização (LIU; LIN; FENG, 2021). Esses dois portões são empregados para resolver o problema do desaparecimento de gradiente, além de manter memórias de longo prazo, sem remover informações relevantes que são significativas para previsões futuras. Especificamente, o portão de atualização determina quanta informação de etapas de tempo anteriores deve ser encaminhada para etapas futuras, enquanto que o portão de reinicialização decide quanta informação passada pode ser esquecida (LIU; LIN; FENG, 2021). A Figura 5 ilustra um diagrama de uma célula GRU.

De acordo com (LIU; LIN; FENG, 2021) , a operação da célula GRU pode ser inicializada a partir do cálculo da porta de atualização z_t no tempo t. Levando em consideração a nomenclatura da Figura 5, temos que:

$$z_t = \sigma_g(\boldsymbol{w}_z x_t + \boldsymbol{u}_z h_{t-1} + b_z) \tag{2.27}$$

em que σ_g é a função de ativação sigmoide, $\boldsymbol{w}_z$ e $\boldsymbol{u}_z$ são os pesos da entrada e o do estado da célula, respectivamente. x_t é uma observação de entrada no tempo t. h_{t-1} é o estado da célula em $t-1$, e b_z é o bias. Depois disso, o portão de reinicialização r_t é calculado pela Equação 2.28:

$$r_t = \sigma_g(\boldsymbol{w}_r x_t + \boldsymbol{u}_r h_{t-1} + b_r) \tag{2.28}$$

em que $\boldsymbol{w}_r$ e $\boldsymbol{u}_r$ são os pesos da entrada e o do estado da célula, respectivamente. x_t é uma observação de entrada no tempo t. h_{t-1} é o estado da célula em $t-1$, e b_r é o bias. Em seguida, o r_t calculado é usado para introduzir um novo conteúdo de memória $\widetilde{h}_t$ na

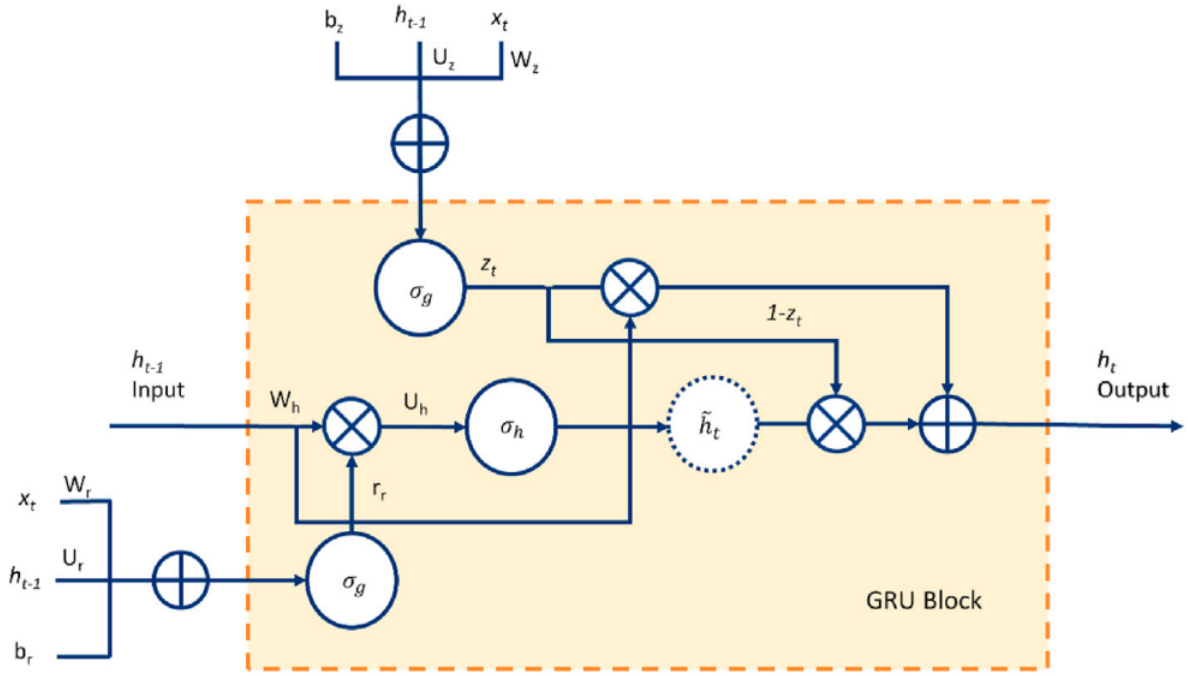

Figura 5 – Ilustração de uma célula GRU. O portões de atualização e reinicialização com a função de ativação sigmoide σ_g controlam as informações que devem ser guardadas ou esquecidas (LIU; LIN; FENG, 2021).

Fonte: (LIU; LIN; FENG, 2021).

Equação 2.29:

$$\widetilde{h}_t = tanh(\boldsymbol{w}_h x_t + (r_t \odot \boldsymbol{u}_h h_{t-1}) + b_h) \tag{2.29}$$

em que $\boldsymbol{w}_h$ e $\boldsymbol{u}_h$ são os pesos da entrada e o do estado da célula, respectivamente. x_t é uma observação de entrada no tempo t. h_{t-1} é o estado da célula em $t-1$, e b_r é o bias. O operador $\odot$ denota o produto Hadamard. Por fim, o vetor de estado atual da célula h_t é calculado para passar as informações guardadas para a próxima unidade, utilizando os valores de z_t e $\widetilde{h}_t$:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \widetilde{h}_t \tag{2.30}$$

em que h_{t-1} é o estado da célula em $t-1$.

Na área de energia eólica, o modelo GRU foi aplicado em (LIU; LIN; FENG, 2021) e comparado com o LSTM e o modelo ARIMA sazonal para prever séries temporais de velocidade do vento para o horizonte de 1h. E, apesar do GRU e LSTM possuírem algoritmos de aprendizado de máquina avançados, o ARIMA sazonal mostrou ter um desempenho preditivo superior em três bases de dados nos resultados.

Aplicar somente um único tipo de modelagem, seja linear ou não linear, pode não ser adequado para modelar efetivamente séries temporais de velocidade do vento. Essa inadequação ocorre porque essas séries temporais demonstram padrões complexos, ambíguos e uma mistura de padrões lineares e não lineares (AHMADI; KHASHEI, 2021; HU; WANG; ZENG, 2013; JIANG et al., 2019; de MATTOS NETO et al., 2020). Portanto, é aconselhável

utilizar modelos híbridos, que incorporam combinações de mais de um preditor. A seguir, serão discutidos os métodos híbridos.

2.3 Modelo Híbrido Serial (MHS)

Modelos híbridos têm sido extensivamente usados em previsão de séries temporais, devido ao aumento da acurácia de previsão e redução da incerteza em relação aos modelos individuais (AHMADI; KHASHEI, 2021). Na prática é difícil determinar se uma série temporal é gerada a partir de um processo linear ou não linear (ZHANG, 2003). Assim, o uso de apenas um tipo de modelagem (linear ou não linear) pode não ser adequado, levando a resultados com baixa acurácia (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017). Em (ZHANG, 2003), um tipo de MHS foi desenvolvido, em que as previsões dos modelos ARIMA e MLP são combinadas linearmente. O ARIMA é usado para analisar o componente linear do problema, enquanto o modelo MLP é aplicado para investigar os resíduos do modelo ARIMA. Já que o ARIMA não pode capturar a estrutura não linear dos dados, os resíduos do ARIMA conterão padrões não lineares, se eles existirem. Assim, com base em (ZHANG, 2003), uma série temporal no tempo t, denotada por y_t, assume-se que é composta por três componentes: a linear, não linear e aleatória. Além disso, a relação entre as componentes linear e não linear pode ser linear (aditiva), tal que

$$y_t = L_t + N_t + \varepsilon_t, \tag{2.31}$$

em que L_t, N_t e ε_t são, respectivamente, componentes linear, não linear e aleatória no tempo t. O modelo ARIMA é responsável por modelar a série temporal e, portanto, seu resíduo no tempo t, denotado por e_t, conterá padrões não lineares e a componente aleatória, ou seja

$$e_t = y_t - \widehat{L}_t = N_t + \varepsilon_t. \tag{2.32}$$

Dessa forma, $\widehat{L}_t$ é a previsão do modelo ARIMA no tempo t. Ao modelar esses resíduos usando uma MLP, relacionamentos não lineares podem ser descobertos. Assim, o modelo MLP para o resíduo e_t é dado por

$$e_t = f(e_{t-1}, e_{t-2}, ..., e_{t-o}) + \varepsilon_t, \tag{2.33}$$

em que f é uma função não linear determinada pelo MLP, e a variável o é o número de nós de entrada. Assumindo que ε_t é um ruído branco imprevisível, e denotando a previsão dada por (2.33) como $\widehat{N}_t$, a previsão do modelo híbrido linearmente combinado no tempo t será

$$\widehat{y}_t = \widehat{L}_t + \widehat{N}_t\,. \tag{2.34}$$

Na Figura 6 é apresentada a fase de treinamento do MHS com combinação linear proposta em (ZHANG, 2003). As entradas são os conjuntos de treinamento e validação, o

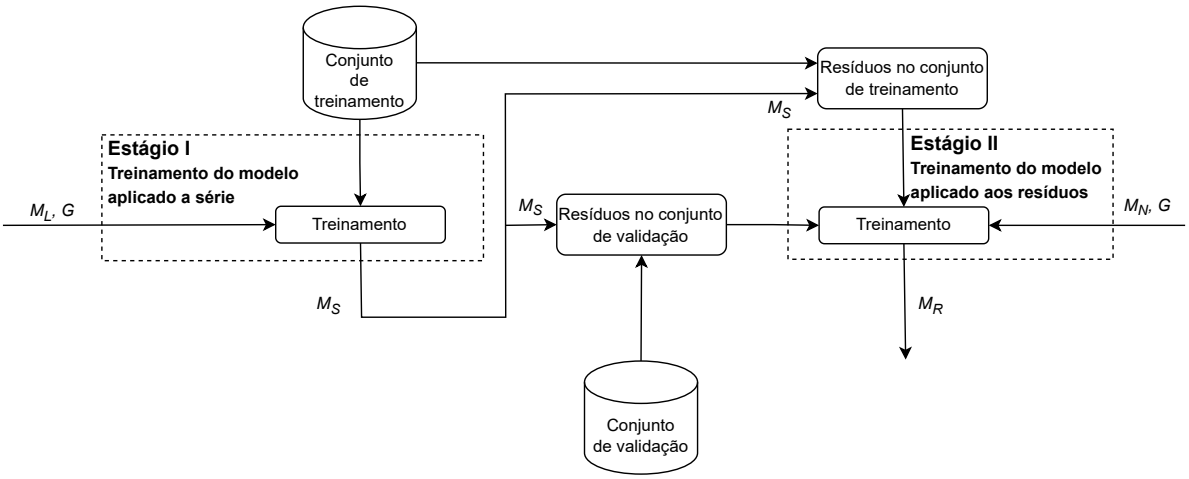

Figura 6 – Fase de treinamento do MHS com combinação linear proposta em (ZHANG, 2003).

modelo linear M_L e sua técnica de busca G pelos hyperparâmetros. Além do modelo não linear M_N e sua técnica de busca G pelos hyperparâmetros. No Estágio I é realizado o treinamento do modelo linear M_L. No Estágio II ocorre o treinamento do modelo não linear M_N, que utiliza um conjunto de validação para evitar o *overfitting*. As saídas envolvem o modelo linear treinado M_S na série e o modelo não linear treinado M_R nos resíduos.

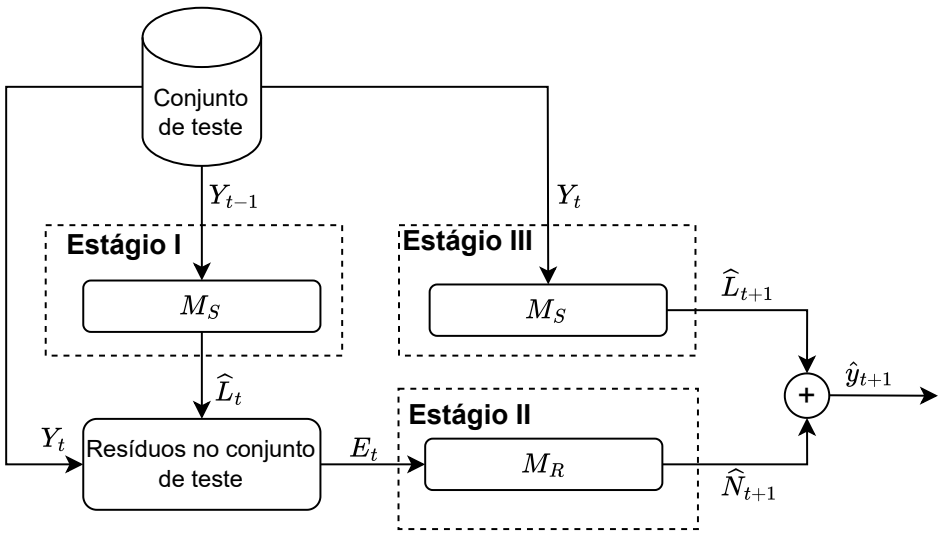

Figura 7 – Fase de teste do MHS com combinação linear proposta em (ZHANG, 2003).

A fase de teste do MHS com combinação linear proposta em (ZHANG, 2003) é mostrada na Figura 7, em que a série temporal $Y_t = [y_t, y_{t-1}, ..., y_{t-k}]$ e a série residual $E_t = [e_t, e_{t-1}, ..., e_{t-s}]$ são representadas com k e s indicando as defasagens determinadas pelos modelos M_S e M_R, respectivamente. No Estágio I ocorre a estimação de $\widehat{L}_t$ no passado até t para obter os resíduos que servirão de entrada para o Estágio II. No Estágio II é realizada a previsão não linear $\widehat{N}_{t+1}$ em $t+1$ dos resíduos pelo modelo M_R. No Estágio III é aplicado o modelo M_S para obter a previsão linear $\widehat{L}_{t+1}$ em $t+1$ da série. E, por fim,

as previsões $\widehat{L}_{t+1}$ e $\widehat{N}_{t+1}$ são combinadas linearmente por meio de uma adição. Assim, a previsão $\widehat{y}_{t+1}$ no tempo $t+1$ para esta abordagem híbrida com três estágios pode ser dada por

$$\widehat{y}_{t+1} = M_S(Y_t) + M_R(E_t)\,. \tag{2.35}$$

MHSs combinados linearmente são encontrados na previsão de velocidade do vento. Em (CADENAS; RIVERA, 2010), o modelo ARIMA foi utilizado para prever a série de velocidade do vento, e posteriormente, nos erros obtidos, foi aplicada uma MLP para capturar os padrões não lineares que o ARIMA não identificou. Os resultados mostraram que o modelo híbrido superou o ARIMA e a MLP individualmente em três bases de dados para previsão a curto prazo (1 hora). Em (CAMELO et al., 2018a) foi proposto um modelo híbrido com variáveis exógenas para prever a velocidade do vento a longo prazo (1 mês). Os autores verificaram que a utilização do modelo exógeno ARIMAX em vez do ARIMA na combinação linear com a MLP, reduziu os erros de previsão do modelo híbrido para previsões a longo prazo. Em (CAMELO et al., 2018b), o mesmo modelo de (CAMELO et al., 2018a), o ARIMAX com MLP, foi aplicado em previsões a curto prazo (1 hora). Notou-se que a diferença em termos de acurácia foi muito pequena com relação ao ARIMA com MLP, e, às vezes, nem existe. Por exemplo, na base de dados Fortaleza, a raiz do erro quadrático médio (RMSE, do inglês *Root Mean Squared Error*) dos modelos híbridos combinados linearmente ARIMA com MLP e ARIMAX com MLP foi idêntica. Em (JIANG et al., 2019), foi proposto um MHS combinado linearmente para previsões de velocidade do vento a curto prazo (10 minutos e 30 minutos). O modelo híbrido integrou as vantagens do método de suavização exponencial (ES, do inglês *Exponential Smoothing*) de Holt para capturar padrões lineares e o SVM para capturar padrões não lineares. Também foi utilizado um pré-processamento nos dados por meio de uma decomposição da série. Os resultados indicaram que o modelo proposto superou a capacidade preditiva dos modelos individuais ARIMA, Holt, rede neural de regressão generalizada (GRNN, do inglês *General Regression Neural Network*), SVM, e dos modelos híbridos combinados linearmente ARIMA com SVM e ARIMA com GRNN. Em (HUANG; WANG; HUANG, 2021b) foram propostos MHSs combinados linearmente para previsão de velocidade do vento a curto prazo (15 minutos a 1 hora) e também foi aplicado um pré-processamento nas séries através de uma decomposição. Neste último artigo, o importante é destacar que a combinação do modelo ES com uma MLP obteve os melhores resultados e superou os modelos ES, ARMA, MLP e ARMA com MLP.

Modelar adequadamente padrões lineares e não lineares em séries temporais é um grande desafio, pois a combinação desses dois componentes nem sempre é linear (KHASHEI; BIJARI, 2011; de MATTOS NETO; CAVALCANTI; MADEIRO, 2017). Com base nisso, uma arquitetura mais geral foi proposta em (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017), em que um método de combinação não linear, intitulado NoLiC, foi usado para mesclar os preditores. A Figura 8 exibe a fase de treinamento desta abordagem de previsão híbrida

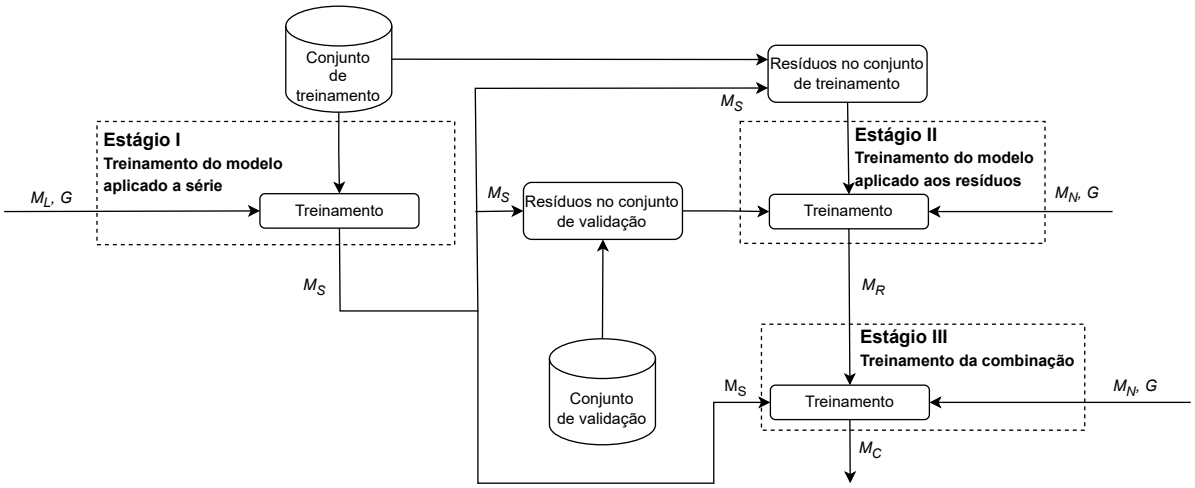

Figura 8 – Fase de treinamento do MHS com combinação não linear proposta em (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017).

com três estágios. A combinação não linear é denotada por M_C no Estágio III, em substituição à adição que ocorre em (ZHANG, 2003). As saídas da fase de treinamento envolvem o modelo linear treinado M_S na série, o modelo não linear treinado M_R nos resíduos, e o modelo não linear treinado M_C para empregar a combinação.

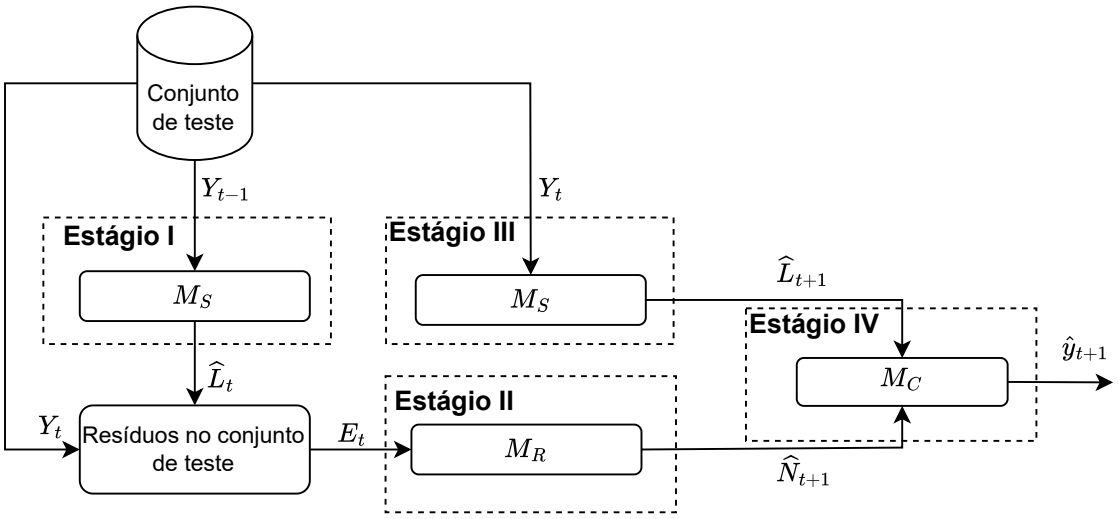

Figura 9 – Fase de teste do MHS com combinação não linear proposta em (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017).

Já na Figura 9 é apresentada a fase de teste do MHS com combinação não linear proposta em (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017). Os Estágios I, II e III ocorrem da mesma forma que o modelo de (ZHANG, 2003). A diferença está no novo Estágio IV, em que as previsões $\widehat{L}_{t+1}$ e $\widehat{N}_{t+1}$ são combinadas não linearmente pelo modelo não linear M_C. Portanto, a saída $\widehat{y}_{t+1}$ no tempo $t+1$ do método NoLiC pode ser representada por

$$\widehat{y}_{t+1} = M_C(M_S(Y_t), M_R(E_t))\,. \tag{2.36}$$

Em (ALENCAR et al., 2018b) foi proposto um MHS combinado não linearmente para previsão de velocidade do vento no Nordeste brasileiro a curto prazo (1 hora). O modelo ARIMA sazonal foi utilizado para a modelagem dos padrões lineares, e a MLP foi empregada para o reconhecimento de padrões não lineares nos resíduos. Posteriormente, foi utilizada uma MLP para empregar uma combinação não linear entre os preditores. As simulações revelaram que o método de previsão híbrido proposto obteve uma maior acurácia que os modelos ARIMA sazonal, MLP e ARIMA sazonal com Wavelet. O modelo Wavelet é um tipo de decomposição para séries temporais. Em (de MATTOS NETO et al., 2020) foi proposto um MHS combinado não linearmente para previsões de velocidade do vento a longo prazo (1 mês). Modelos lineares ARIMA univariado e ARIMAX foram combinados com modelos não lineares MLP e SVR. Foram utilizadas bases de dados do Nordeste brasileiro para uma comparativo entre MHSs combinados linearmente e combinados não linearmente. Os autores verificaram que MHSs combinados não linearmente superam em termos de acurácia os MHSs combinados linearmente na maioria dos casos. A combinação não linear demonstrou ser uma abordagem promissora para a previsão da velocidade do vento.

2.4 Modelo Híbrido Paralelo (MHP)

As características dos MHSs são bem distintas dos MHPs. Na modelagem do MHS, modelos individuais são aplicados sequencialmente, o que significa que a saída de um modelo serve como entrada para o próximo. Em contraste, no que diz respeito aos MHPs, os modelos individuais são aplicados em paralelo. Isso indica que todos os modelos individuais são aplicados diretamente à série temporal de alguma maneira (AHMADI; KHASHEI, 2021). Os autores em (BERGMEIR; HYNDMAN; BENÍTEZ, 2016) desenvolveram um MHP com mapeamento de padrões global usando um método de *bagging* e o aplicaram à base de dados *M3-competition*. A série temporal é reamostrada usando a técnica *moving block bootstrap*. Segundo (BERGMEIR; HYNDMAN; BENÍTEZ, 2016), essa técnica é um procedimento de *bootstrap* que leva a autocorrelação em consideração. Portanto, é um *bagging* direcionado para séries temporais. Primeiramente, aplica-se uma transformação de Box-Cox nos dados. A transformação de Box-Cox é utilizada para estabilizar a variância de séries temporais. Em seguida, a série é decomposta em componentes de tendência, sazonalidade e resíduo. O resíduo pode ser assumido como estacionário, porém pode apresentar autocorrelação. Portanto, a componente de resíduo é submetida ao *bootstrap* utilizando o método *moving block bootstrap*. O tamanho do bloco é definido de forma que possibilite capturar a sazonalidade remanescente nos resíduos. Em seguida, os blocos são amostrados com reposição. E, finalmente, as componentes de tendência e sazonalidade são reincorporados aos resíduos reamostrados, e a transformação de Box-Cox é invertida. Dessa maneira, obtém-se um conjunto de séries temporais geradas por bootstrap. O objetivo deste pro-

cesso é obter preditores que forneçam uma previsão final resiliente à incerteza observada nos dados. Este método envolve o treinamento de 100 modelos base em todo o conjunto de treinamento, resultando em um alto custo computacional.

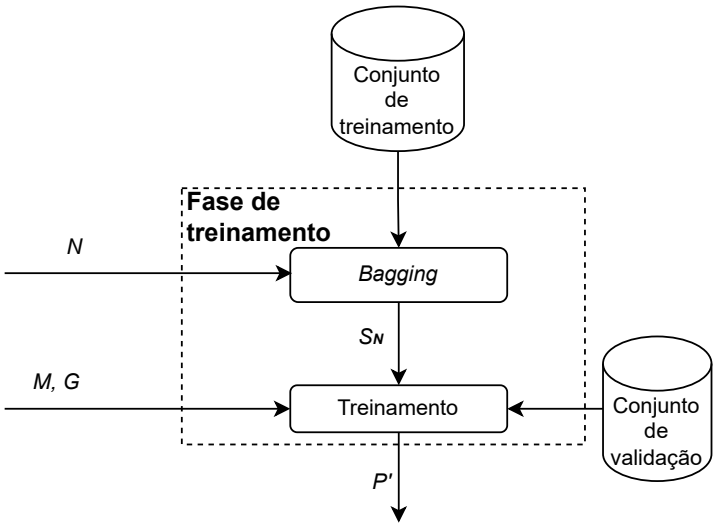

Figura 10 – Fase de treinamento do MHS com combinação não linear proposta em (BERGMEIR; HYNDMAN; BENÍTEZ, 2016).

Na Figura 10 é apresentada a fase de treinamento do MHP por meio do *bagging* proposta em (BERGMEIR; HYNDMAN; BENÍTEZ, 2016). As entradas são os conjuntos de treinamento e validação, a quantidade N de reamostragens da série, o modelo base M e sua técnica de busca G pelos hyperparâmetros. Na fase de treinamento, primeiramente a série temporal do conjunto de treinamento é reamostrada N vezes, e posteriormente cada série reamostrada S_N é treinada pelo modelo M com uma técnica de busca G para os hyperparâmetros. No caso dos modelos base não lineares, um conjunto de validação é utilizado para evitar o *overfitting*. A saída consiste do *pool* de modelos treinados P'.

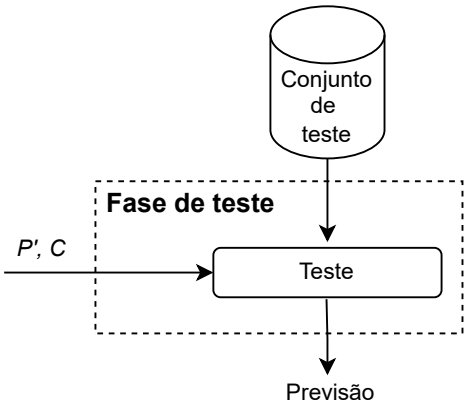

Figura 11 – Fase de treinamento do MHS com combinação não linear proposta em (BERGMEIR; HYNDMAN; BENÍTEZ, 2016).

Ilustrado na Figura 11 está a fase de teste do MHP por meio do *bagging* proposta em (BERGMEIR; HYNDMAN; BENÍTEZ, 2016). A entrada é composta pelo conjunto de

teste da série temporal, o *pool* de preditores treinados P' e o método de combinação C para integração. A saída é a previsão da série temporal.

2.5 Seleção Estática em Modelos Híbridos Paralelos

Nem sempre a utilização de todos os modelos base na composição do MHP resultará nos melhores resultados. Às vezes, é preciso remover alguns modelos base que estão comprometendo o desempenho preditivo do modelo híbrido. Além disso, obtém-se um modelo híbrido mais enxuto com os modelos mais competentes. Isso resulta em uma generalização mais robusta e resposta mais rápida nos dados que serão previstos (VERGARA; ESTÉVEZ, 2014). Desta forma, surge uma nova etapa na construção de MHPs, a chamada fase de seleção. Na literatura é discutido que é um desafio selecionar o subconjunto ideal de modelos individuais entre todos os modelos disponíveis, sem ter que tentar todas as combinações possíveis desses modelos. Por exemplo, existem 502 combinações possíveis, se há nove modelos individuais no total (CANG; YU, 2014).

Para evitar o uso exaustivo de todas as combinações possíveis, a etapa de seleção do subconjunto de modelos mais competentes pode ser realizada de maneira estratégica. Esse processo pode basear-se em propriedades probabilísticas, como a Informação Mútua (IM), e em métricas de erro, como a acurácia, avaliadas entre os modelos preditivos em um conjunto de seleção. Na literatura, o termo 'poda' é frequentemente empregado para descrever esse tipo de seleção, já que ela consiste na remoção dos modelos base que prejudicam o desempenho preditivo do MHP.

A tentativa de selecionar um subconjunto de preditores usando a teoria da informação é bastante usada em reconhecimento de padrões e em áreas que envolvem redes neurais (CANG; YU, 2014). A IM é uma medida da quantidade de informação que uma variável aleatória tem sobre outra variável (VERGARA; ESTÉVEZ, 2014). O valor é positivo se houver alguma dependência entre variáveis, e é zero, se as variáveis são independentes. Ademais, a IM está diretamente ligada à entropia de cada variável. A entropia é uma medida de incerteza de uma variável aleatória. A incerteza está relacionada à probabilidade de ocorrência de um evento. Uma alta entropia significa que cada evento tem aproximadamente a mesma probabilidade de ocorrência, enquanto baixa entropia significa que cada evento tem uma probabilidade diferente de ocorrência (VERGARA; ESTÉVEZ, 2014). Em (CANG; YU, 2014) e (CHE, 2015) são propostos métodos de seleção a partir da IM para problemas de previsão, e os resultados mostram que um subconjunto de modelos pode atingir um desempenho superior em termos de acurácia do que o melhor modelo individual ou a combinação de todos os modelos. Além disso, os autores em (CANG; YU, 2014) concluem que os subconjuntos com os melhores resultados possuem entre dois e cinco modelos.

Uma outra forma de tentar selecionar um subconjunto de preditores é a partir da

acurácia. Neste caso, em vez de utilizar a IM como critério de seleção, é empregada a acurácia. Em (ADHIKARI; VERMA; KHANDELWAL, 2015) foi proposto um método de seleção que aplicou um ranqueamento nos preditores de acordo com o menor erro de previsão. A proposta selecionou os cinco modelos mais acurados entre nove modelos individuais. E os resultados mostraram a superioridade do método de ranqueamento no MHP em termos de acurácia em relação aos modelos individuais e ao MHP com os nove modelos base. No estudo apresentado em (SERGIO, 2017), demonstrou-se que a seleção de preditores base com as menores taxas de erro em um conjunto de seleção resulta em um MHP mais preciso em comparação a um MHP que utiliza todos os modelos base. O autor argumenta que a exclusão de preditores com altas taxas de erro reduz a probabilidade de gerar saídas discrepantes em relação ao valor esperado. Os melhores resultados foram obtidos utilizando um subconjunto de cinco preditores selecionados entre um total de dez disponíveis.

2.6 Modelos Híbridos Paralelos com Mapeamento Local

Todos os métodos mencionados anteriormente (modelos individuais, MHSs e MHPs) usaram todo o conjunto de treinamento para o ajuste dos parâmetros dos modelos base. Isto é, empregaram um reconhecimento de padrões global durante a fase de treinamento. No entanto, essa abordagem não é a melhor escolha devido à fraqueza do mapeamento global, particularmente em sistemas de energia complexos com padrões locais intrincados (ALMEIDA; de MATTOS NETO; CUNHA, 2024). Portanto, na literatura, é possível encontrar algumas abordagens de MHPs que são destinadas a identificar diferentes padrões locais em uma série temporal, permitindo a geração de preditores locais especializados (de MATTOS NETO et al., 2020a). Por exemplo, em (HU; WANG; ZENG, 2013; RUIZ-AGUILAR et al., 2021; JIANG; CHE; WANG, 2021), as séries temporais de velocidade do vento foram decompostas em um número limitado de funções de modo intrínseco (IMF, do inglês *Intrinsic Mode Functions*) usando decomposição de modo empírico (EMD, do inglês *Empirical Mode Decomposition*). Essas IMFs contêm informações sobre tendências e flutuações locais em diferentes escalas do sinal original, permitindo uma análise mais completa do significado físico do sinal. Nesses estudos, os IMFs são utilizados como entradas para MHPs homogêneos, permitindo a identificação de padrões locais e complexos nas séries temporais por meio de IMFs. Já os autores em (de MATTOS NETO et al., 2020a) desenvolveram um método para identificar diferentes padrões locais em séries temporais de várias aplicações. A proposta criou subconjuntos de treinamento de tamanho igual, com a opção de sobreposição entre subconjuntos adjacentes. Essa configuração permitiu que o modelo se adaptasse a mudanças progressivas de padrão. Além disso, os autores investigaram quatro tamanhos de subconjuntos diferentes. Eles concluíram que os valores apropriados para o tamanho do subconjunto de treinamento podem variar de acordo com as características do conjunto de dados. Por exemplo, o comportamento e o tamanho da série temporal.

2.7 Modelos Híbridos Seriais com Mapeamento Local

Em (SANTOS JÚNIOR et al., 2023), é afirmado que a fase de modelagem residual é um dos principais desafios no desenvolvimento dos MHSs, devido à presença de flutuações aleatórias, comportamento heterocedástico e ruído nos resíduos. Adotar uma única abordagem com mapeamento global pode não ser suficiente para capturar padrões não lineares locais complexos em resíduos. Assim, os autores em (SANTOS JÚNIOR et al., 2023) propuseram um MHS que empregou um MHP com mapeamento local na fase de modelagem dos resíduos. O objetivo foi melhorar a capacidade de generalização do sistema, reduzir o risco de selecionar um modelo incorreto e expandir o espaço de funções. O MHP foi construído usando o método de geração *random patches*, e os resultados em séries temporais de várias aplicações mostraram a superioridade preditiva desse sistema híbrido.

No estudo apresentado em (MA et al., 2020a), foi apresentado um outro MHS que também empregou um MHP com mapeamento local na série de resíduos. Uma decomposição de modo variacional (VMD, do inglês *Variational Mode Decomposition*) foi empregada na série de resíduos para obter IMFs. Os resultados obtidos em séries temporais de velocidade do vento demonstraram que o emprego do MHP, aliado à decomposição por VMD nos resíduos, proporcionou maior precisão nas previsões quando comparado a modelos que não empregavam o MHP nos resíduos. Por fim, de forma similar, o trabalho apresentado em (YAN et al., 2022) aplicou um MHP com reconhecimento de padrões locais nos resíduos, porém utilizando o método de decomposição de modo empírico por conjunto (EEMD, do inglês *Ensemble Empirical Mode Decomposition*). Os resultados alcançados para séries temporais de velocidade do vento evidenciaram a superioridade da aplicação do MHP nos resíduos de um MHS.

Estas inovações nos MHSs que empregam métodos de MHPs com reconhecimento de padrões locais destacaram a relevância de combinar MHSs e MHPs. Assim, mais investigações sobre essa abordagem híbrida são desejáveis, especificamente em dados complexos de velocidade do vento.

2.8 Resumo do Capítulo

Neste capítulo, foi apresentada a fundamentação teórica e a literatura utilizada para o desenvolvimento desta pesquisa de doutorado. O capítulo foi introduzido com a teoria de séries temporais, e posteriormente foram discutidos em detalhes os modelos individuais, MHSs e MHPs que servem de referência para esta pesquisa. No capítulo a seguir, serão apresentados os métodos propostos deste trabalho.

3 MÉTODOS PROPOSTOS

Nesse capítulo serão apresentados os métodos propostos desta pesquisa, com a descrição detalhada das fases de treinamento e teste.

3.1 O Método LocPart

O LocPart proposto em (ALMEIDA; de MATTOS NETO; CUNHA, 2024) é uma das contribuições desta pesquisa, pois evidenciou a importância de variar a região e o tamanho dos subconjuntos de treinamento na etapa de geração de um MHP. O LocPart possibilita a geração de preditores especialistas em subconjuntos de tamanhos e regiões variadas para detectar padrões locais complexos, e sua arquitetura é ilustrada na Figura 12.

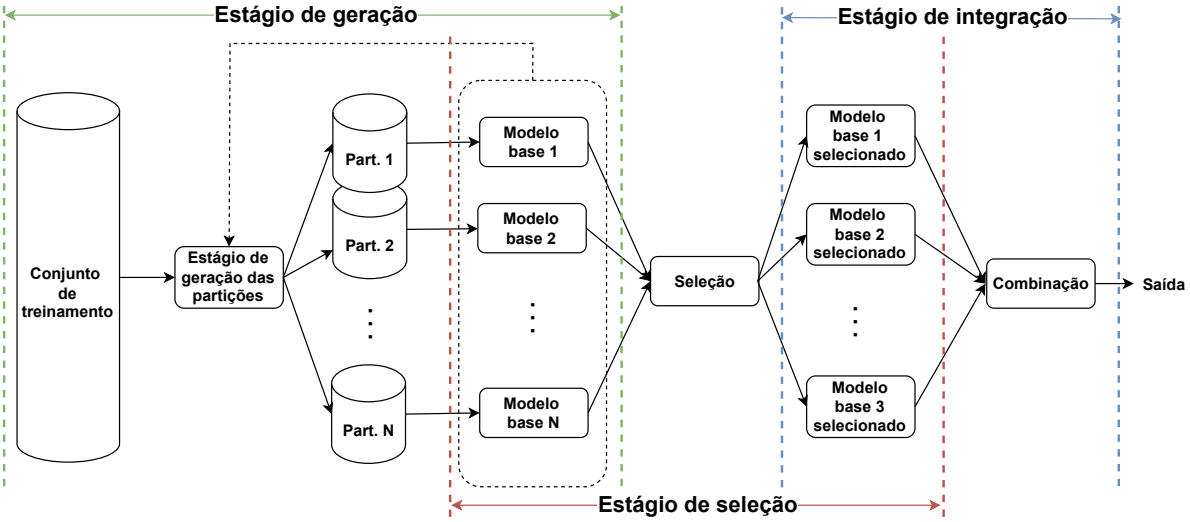

Figura 12 – Arquitetura do MHP com reconhecimento de padrões locais por meio da variação do tamanho e região das subconjuntos (LocPart).

Conforme descrito em (ALMEIDA; de MATTOS NETO; CUNHA, 2024), e de acordo com a Figura 12, o LocPart é um MHP homogêneo que compreende três estágios: geração, seleção e integração. A inovação está no estágio de geração, que envolve o uso de um *loop* para criar subconjuntos de tamanhos variados e em diferentes regiões do conjunto de treinamento, produzindo assim uma coleção de preditores especialistas locais. Posteriormente, um processo de seleção é iniciado para identificar os preditores mais competentes. Finalmente, os preditores selecionados são combinados durante o estágio de integração para produzir a previsão final do MHP. Os resultados de (ALMEIDA; de MATTOS NETO; CUNHA, 2024) foram promissores para séries temporais de velocidade do vento. Demonstrou-se que o LocPart superou métodos que empregam mapeamento de padrões global, tais quais modelos individuais e MHPs gerados por *bagging*.

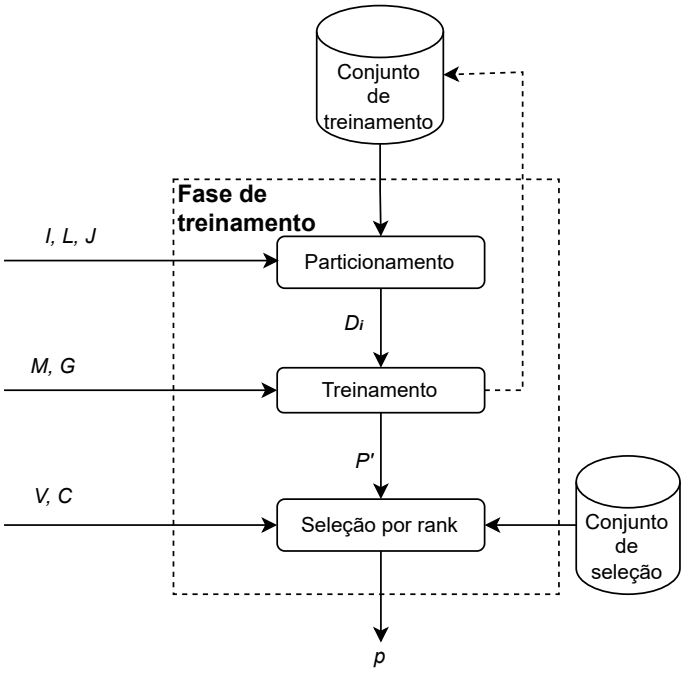

Figura 13 – Fase de treinamento do LocPart.

O treinamento do método LocPart está ilustrado na Figura 13 e pode ser dividido nas seguintes etapas:

(I) **Determinar a Quantidade Inicial de Subconjuntos**: Determine o número inicial I de subconjuntos disjuntos sequenciais;

(II) **Criação dos Subconjuntos**: Divida o comprimento do conjunto de treinamento pelo número de subconjuntos para calcular o tamanho de cada subconjunto. Com base nesse tamanho, crie subconjuntos disjuntos D_i de tamanho igual lado a lado até que todo o conjunto de treinamento seja utilizado;

(III) **Treinar Modelo Base**: Treine um modelo base M usando uma técnica de seleção de hiperparâmetros G nesses subconjuntos disjuntos D_i. Esse processo gera um *subpool* $P_i = \{m_1, m_2, \ldots, m_s\}$ compreendendo m_s preditores especialistas locais, em que s representa o número de subconjuntos disjuntos D_i;

(IV) **Aumentar a Quantidade de Subconjuntos**: De acordo com o valor de incremento L aumente o número de subconjuntos;

(V) ***Loop* Até Máximo de Subconjuntos**: Retorne para a etapa (II) até que o número máximo J de subconjuntos disjuntos seja gerado;

(VI) **Agregar *Subpools***: Combine todos os *subpools* em um *pool* mais abrangente $P' = \{m_1, m_2, \ldots, m_N\}$, em que N é o número total de preditores especialistas produzidos em todos os *loops* na etapa de geração;

(VII) **Seleção de Preditores**: No conjunto de seleção, empregue um sistema de classificação com base em uma medida de avaliação V. A seleção do subconjunto $p = \{m_1, m_2, \ldots, m_n\}$ de preditores é realizada de acordo com o menor erro na medida de avaliação E no conjunto de seleção sobre um método de combinação C dos n preditores selecionados.

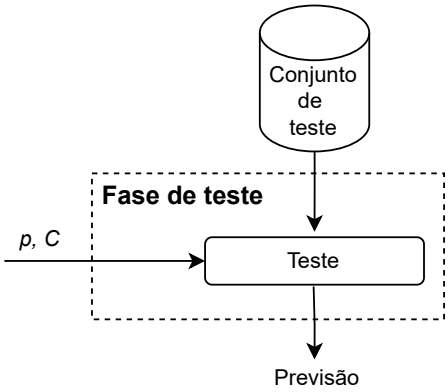

Figura 14 – Fase de teste do LocPart.

A Fig. 14 apresenta a fase de teste do LocPart. A entrada é composta pelo conjunto de teste da série temporal, os n preditores especialistas locais selecionados p e o método de combinação C para integração. A saída é a previsão da série temporal de velocidade do vento.

3.2 O Método LocLinNonPR

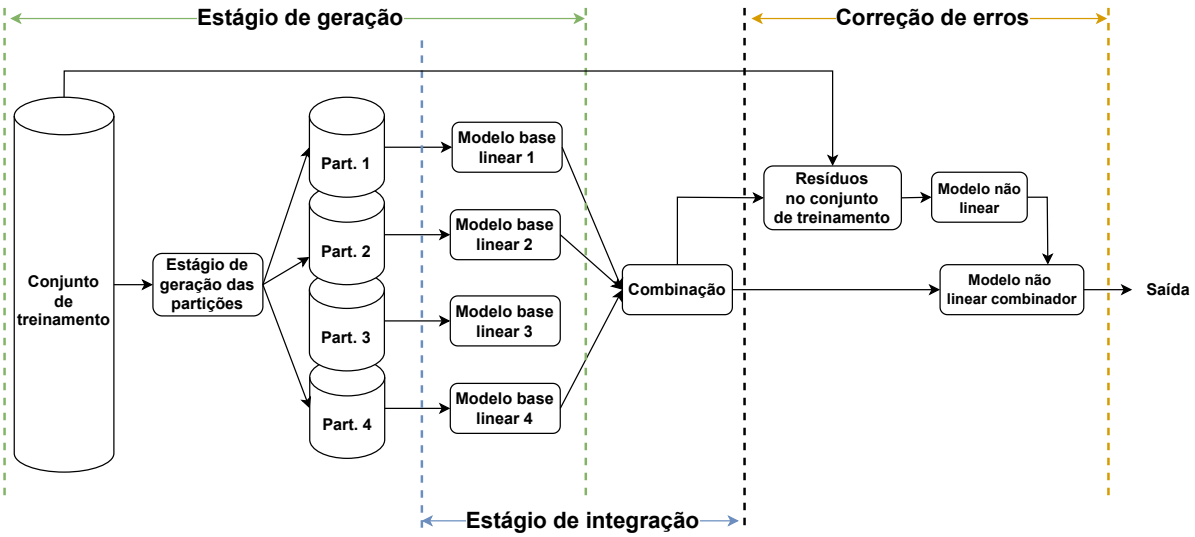

Figura 15 – Arquitetura da fusão do MHP com mapeamento local com um MHS, o método LocLinNonPR.

O LocLinNonPR (ALMEIDA, 2024) é outra contribuição desta pesquisa. A inovação está na fusão de um MHP, que emprega um mapeamento local, com um MHS. A arquite-

tura deste modelo híbrido é exibida na Figura 15. O método LocLinNonPR consiste em um MHP linear com dois estágios, geração e integração. Além disso, uma técnica MHS é aplicada realizando a modelagem não linear nos resíduos do MHP. No estágio de geração do MHP, quatro subconjuntos de tamanho fixo são gerados para a aplicação de quatro modelos base lineares. O objetivo do MHP é realizar o mapeamento de padrões lineares locais na série temporal em diferentes regiões. Posteriormente, uma abordagem de mapeamento de padrões não lineares globais é empregada nos resíduos do MHP linear. Os resultados em séries temporais automotivas evidenciaram que o LocLinNonPR superou MHPs e MHSs quando aplicados separadamente.

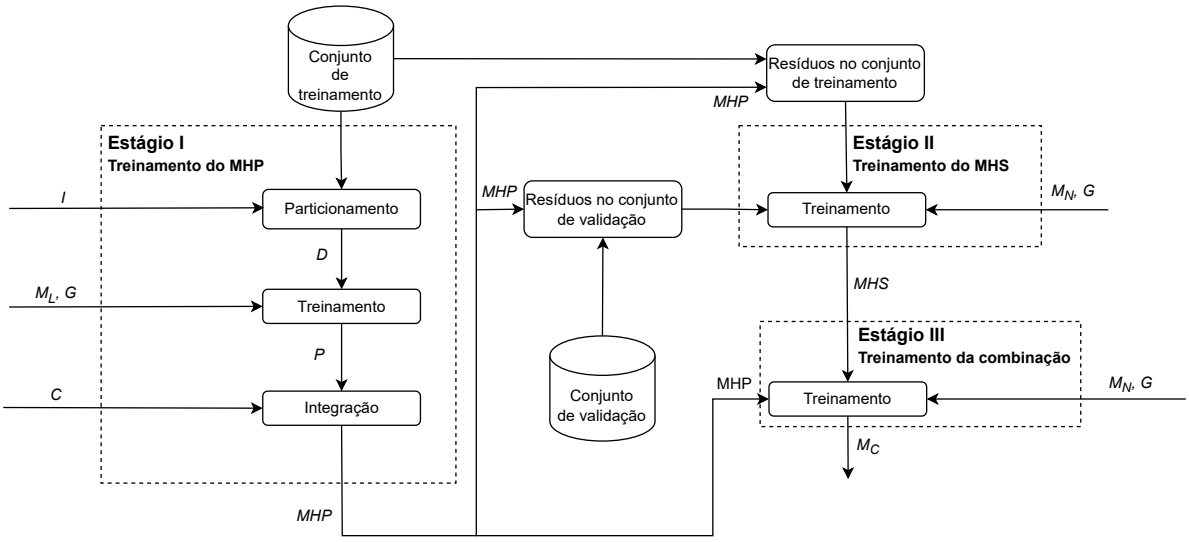

Figura 16 – Fase de treinamento do LocLinNonPR.

O processo de treinamento do método LocLinNonPR está ilustrado na Figura 16 e compreende três estágios:

(I) **O Estágio I é o treinamento do modelo híbrido paralelo *MHP* linear**: As entradas consistem no número I de partições disjuntas em que o conjunto de treinamento será dividido, o modelo linear M_L, uma técnica G para selecionar os hiperparâmetros do modelo M_L, e um método de combinação C para a etapa de integração do MHP. A saída consiste no MHP definido como $P = \{m_1, m_2, ..., m_I\}$, composto por preditores lineares especialistas locais base m_I treinados em partições disjuntas do conjunto de treinamento e integrados por um método de combinação C. Cada preditor é treinado em um único subconjunto disjunto. A fase de treinamento pode ser descrita da seguinte forma:

a) Obtenha o número I de partições disjuntas sequenciais;

b) Divida o tamanho do conjunto de treinamento pelo número I de partições para obter as partições disjuntas $D = \{d_1, d_2, ..., d_I\}$;

c) Treine os modelos base lineares M_L utilizando uma técnica G para seleção de hiperparâmetros nas partições disjuntas D, gerando uma coleção de preditores especialistas locais $P = \{m_1, m_2, ..., m_I\}$;

d) Combine os preditores P com um método de combinação C para obter a saída do MHP.

(II) **No Estágio II é empregada uma técnica MHS, em que um modelo não linear M_N é treinado nos resíduos do *MHP* linear**: As entradas envolvem os resíduos do MHP linear no conjunto de treinamento e no conjunto de validação, o modelo não linear M_N, e uma técnica G para selecionar os hiperparâmetros de M_N. A saída é o MHS treinado nos resíduos do MHP;

(III) **O Estágio III é o treinamento da combinação entre o *MHP* e o *MHS***: As entradas são o MHP treinado na série, o MHS treinado nos resíduos do MHP, o modelo não linear M_N, uma técnica G para selecionar os hiperparâmetros de M_N. A saída é o combinador treinado M_C.

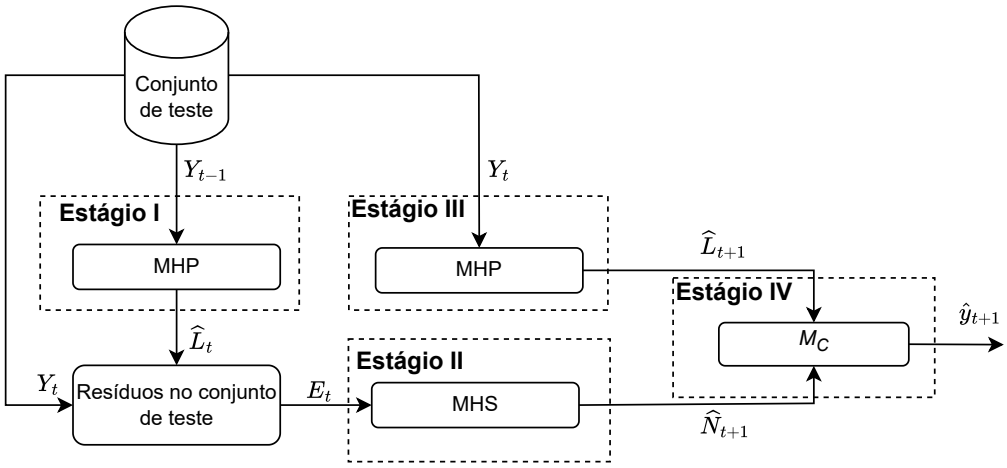

Figura 17 – Fase de teste do LocLinNonPR.

A Fig. 17 exibe a fase de teste do LocLinNonPR, e consiste em quatro estágios:

(I) **Estágio I**: A entrada do Estágio I são observações no passado do conjunto de teste, ou seja, é a série Y_{t-1} até $t-1$, a saída são as previsões lineares $\widehat{L}_t$ até t;

(II) **Estágio II**: A entrada do Estágio II é a série de resíduos E_t de $\widehat{L}_t$ no conjunto de teste até t, a saída é a previsão não linear $\widehat{N}_{t+1}$ em $t+1$;

(III) **Estágio III**: A entrada do Estágio III é a série Y_t no conjunto de teste até t, a saída é a previsão linear $\widehat{L}_{t+1}$ em $t+1$;

(IV) **Estágio IV**: As entradas do Estágio IV são a previsão linear $\widehat{L}_{t+1}$ em $t+1$ e a previsão não linear $\widehat{L}_{t+1}$ em $t+1$, a saída é a previsão da proposta $\widehat{y}_{t+1}$ em $t+1$.

Os objetivos dos quatro estágios da fase de teste são detalhados a seguir. O Estágio I envolve a geração das estimativas $\widehat{L}_t$ no conjunto de teste para o MHP linear no passado até t. Assim, é possível obter os resíduos no conjunto de teste com padrões não lineares remanescentes do mapeamento linear do MHP. Esses resíduos são a entrada do MHS que aplica a modelagem não linear do Estágio II. Portanto, no Estágio II, ocorre a previsão não linear $\widehat{N}_{t+1}$ dos resíduos no conjunto de teste em $t+1$ por meio do MHS com modelagem não linear. Já no Estágio III é realizada a previsão linear $\widehat{L}_{t+1}$ da série no conjunto de teste em $t+1$ por meio do MHP linear. E, finalmente, no Estágio IV, emprega-se por meio do combinador M_C a combinação das saídas dos Estágios II e III, que são a previsão linear $\widehat{L}_{t+1}$ da série e a previsão não linear $\widehat{N}_{t+1}$ dos resíduos em $t+1$ no conjunto de teste. A saída do Estágio IV é a previsão de um passo à frente $\widehat{y}_{t+1}$ do LocLinNonPR.

3.3 O Método LocLN

O LocLN envolve contribuições dos métodos LocPart e LocLinNonPR, e é a principal proposta dessa tese. A inovação está na fusão do MHP LocPart, que possibilita um reconhecimento de padrões locais por meio de subconjuntos de tamanhos e regiões variáveis, com um MHS em que o mapeamento linear é aplicado à série, e o mapeamento não linear é aplicado aos resíduos. Conforme mencionado no Capítulo 1, séries temporais de velocidade do vento possuem padrões locais complexos. Assim, ao empregar o LocLN que combinou as abordagens de MHPs e MHSs com mapeamento de padrões locais, possibilitou-se superar as limitações de MHPs e MHSs que empregam o reconhecimento de padrões tradicional global.

O processo de treinamento do método LocLN está ilustrado na Figura 18 e compreende quatro estágios:

(I) **O Estágio I é o treinamento do MHP linear $\boldsymbol{LocPart_L}$**: As entradas do Estágio I consistem nos conjuntos de treinamento e de seleção da série temporal, o número inicial I de subconjuntos disjuntos, um valor L para os incrementos de subconjuntos em cada loop, e o número máximo J de subconjuntos disjuntos no loop durante a etapa de geração, o modelo linear M_L com uma técnica para seleção de hiperparâmetros G, a métrica de avaliação V para seleção por *rank* e, finalmente, um método de combinação C para etapa de integração do MHP $LocPart_L$. A saída do Estágio I é o subconjunto selecionado $p = \{m_1, m_2, \ldots, m_n\}$ de m_n modelos lineares especialistas locais combinados pelo método C resultando no MHP linear $LocPart_L$ treinado na série temporal;

(II) **No Estágio II é empregada uma técnica MHS, em que o MHP não linear $\boldsymbol{LocPart_N}$ é treinado nos resíduos do MHP linear $\boldsymbol{LocPart_L}$**: As estradas são as mesmas do Estágio I, com exceção do conjunto de dados, que em vez da série

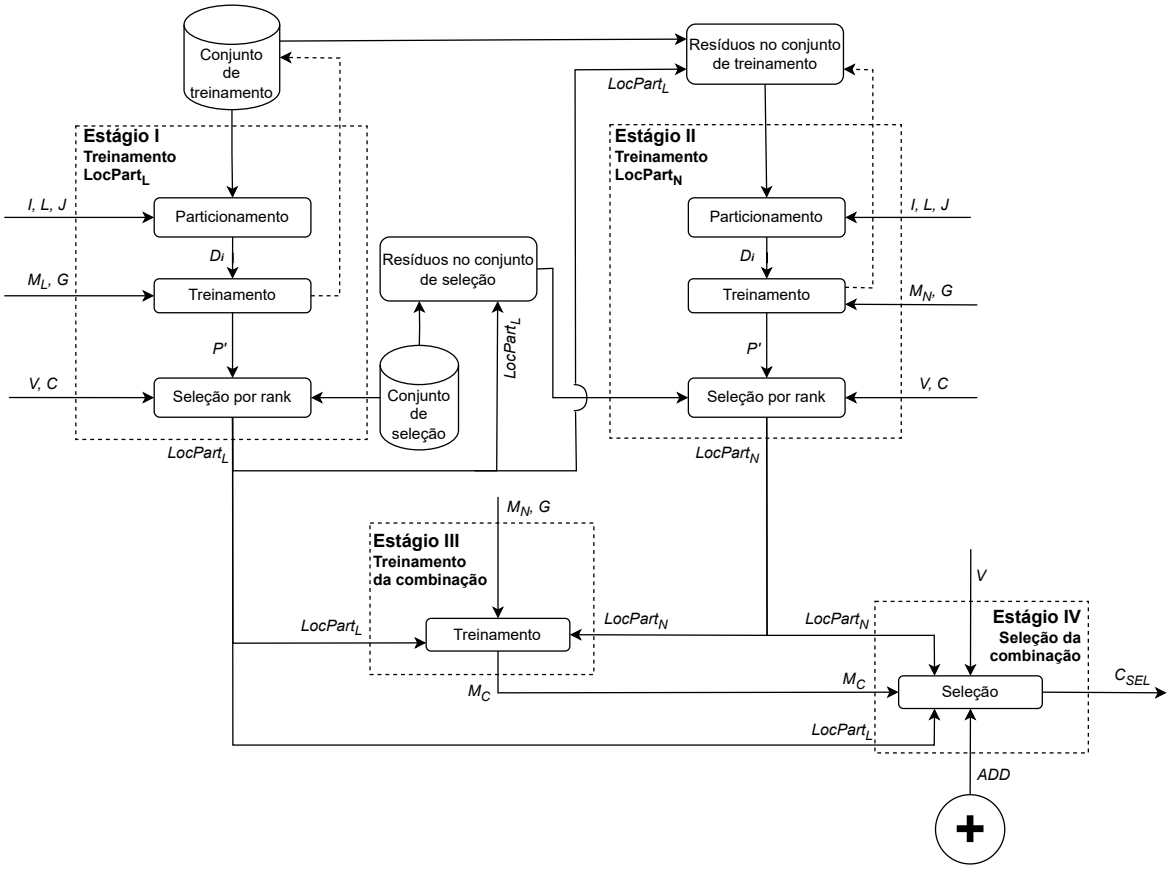

Figura 18 – Fase de treinamento do LocLN.

de treinamento e seleção, são os resíduos do $LocPart_L$ no conjunto de treinamento e seleção, e no lugar do modelo linear M_L é um modelo não linear M_N. A saída é o subconjunto selecionado $p = \{m_1, m_2, \ldots, m_n\}$ de m_n modelos não lineares especialistas locais combinados pelo método C, obtendo o MHP não linear $LocPart_N$ treinado nos resíduos da série temporal;

(III) **O Estágio III é o treinamento da combinação entre o MHP linear $\boldsymbol{LocPart_L}$ treinado na série e o MHP não linear $\boldsymbol{LocPart_L}$ treinado nos resíduos**: As entradas envolvem o MHP linear $LocPart_L$, o MHP não linear $LocPart_L$, o modelo não linear M_N com uma técnica para seleção de hiperparâmetros G. A saída é o modelo combinador M_C treinado para combinar não linearmente o $LocPart_L$ e o $LocPart_N$;

(IV) **O Estágio IV é a seleção do método de combinação dos MHPs $\boldsymbol{LocPart_L}$ e $\boldsymbol{LocPart_N}$**: As entradas são o MHP linear $LocPart_L$ treinado na série, o MHP não linear $LocPart_L$ treinado nos resíduos, o modelo combinador M_C treinado para combinar não linearmente o $LocPart_L$ e o $LocPart_N$, a combinação linear (aditiva) ADD para combinar linearmente o $LocPart_L$ e o $LocPart_N$, e a medida de avaliação V para definir o melhor método de combinação (M_C ou ADD). A saída é o combinador selecionado C_{SEL}.

O processo de treinamento é descrito em detalhes a seguir. Primeiramente, é realizado um mapeamento linear dos padrões da série temporal, baseando-se na proposta de (ZHANG, 2003), que foi o trabalho que introduziu a técnica MHS em séries temporais, e também levando em consideração outros trabalhos que empregam a abordagem MHS (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017), (de MATTOS NETO et al., 2020b), (SANTOS JÚNIOR et al., 2023) e (ALMEIDA, 2024). Posteriormente, é empregado um mapeamento de padrões não lineares nos resíduos. Portanto, o Estágio I do LocLN envolve o reconhecimento de padrões locais lineares na série temporal por meio do $LocPart_L$, e o Estágio II objetiva o mapeamento dos padrões locais não lineares por meio do $LocPart_N$ nos resíduos do MHP $LocPart_L$ após a aplicação do Estágio I. A inovação do LocLN sobre as propostas de (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017), (de MATTOS NETO et al., 2020b), (SANTOS JÚNIOR et al., 2023) e (ALMEIDA, 2024) é a aplicação das modelagens de padrões locais lineares e padrões locais não lineares por meio do método LocPart desenvolvido em (ALMEIDA; de MATTOS NETO; CUNHA, 2024). O treinamento do $LocPart_L$ no Estágio I pode ser dividido nas seguintes etapas:

(I) **Determinar a Quantidade Inicial de subconjuntos**: Determine o número inicial I de subconjuntos disjuntos sequenciais;

(II) **Criação dos Subconjuntos**: Divida o comprimento do conjunto de treinamento pelo número de subconjuntos para calcular o tamanho de cada subconjunto. Com base nesse tamanho, crie subconjuntos disjuntos D_i de tamanho igual lado a lado até que todo o conjunto de treinamento seja utilizado;

(III) **Treinar Modelo Base**: Treine um modelo base linear M_L usando uma técnica de seleção de hiperparâmetros G nesses subconjuntos disjuntas D_i. Esse processo gera um *subpool* $P_i = \{m_1, m_2, \ldots, m_s\}$ compreendendo m_s preditores especialistas locais, em que s representa o número de subconjuntos disjuntos D_i;

(IV) **Aumentar a Quantidade de subconjuntos**: De acordo com o valor de incremento L aumente o número de subconjuntos;

(V) ***Loop* Até Máximo de subconjuntos**: Retorne para a etapa (II) até que o número máximo J de subconjuntos disjuntos seja gerado;

(VI) **Agregar *Subpools***: Combine todos os *subpools* em um *pool* mais abrangente $P' = \{m_1, m_2, \ldots, m_N\}$, em que N é o número total de preditores especialistas produzidos em todos os *loops* na etapa de geração;

(VII) **Seleção de Preditores**: No conjunto de seleção, empregue um sistema de classificação com base em uma medida de avaliação V. A seleção do subconjunto $p = \{m_1, m_2, \ldots, m_n\}$ de preditores é realizada de acordo com o menor erro na

medida de avaliação V no conjunto de seleção sobre um método de combinação C dos n preditores selecionados.

O treinamento do $LocPart_N$ no Estágio II segue o mesmo princípio do Estágio I, mas em vez de utilizar a série de treinamento e seleção como entrada, são usados os resíduos do $LocPart_L$ no conjunto de treinamento e no conjunto de seleção, e no lugar do modelo base linear M_L, é empregado um modelo base não linear M_N. O Estágio III emprega a combinação das saídas dos Estágios I e II, que são os MHPs $LocPart_L$ e $LocPart_N$, respectivamente. A combinação é realizada por meio de um modelo não linear M_N com uma técnica de seleção de hiperparâmetros G, baseado no método NoLiC (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017), e também utilizado em (de MATTOS NETO et al., 2020b) e (ALMEIDA, 2024). O modelo não linear M_N do Estágio III é o responsável por tentar encontrar a melhor função combinadora M_C entre os padrões locais lineares (mapeados pelo MHP $LocPart_L$) e os padrões locais não lineares (mapeados pelo MHP $LocPart_N$) da série temporal e de seus resíduos, respectivamente. Por fim, no Estágio IV, é aplicada uma seleção no conjunto de seleção entre o combinador não linear M_C e o combinador linear (aditivo) ADD. A relação aditiva foi a forma de combinação utilizada na proposta de (ZHANG, 2003) e também em (de MATTOS NETO et al., 2020) e (SANTOS JÚNIOR et al., 2023), que em alguns casos, pode superar a combinação não linear.

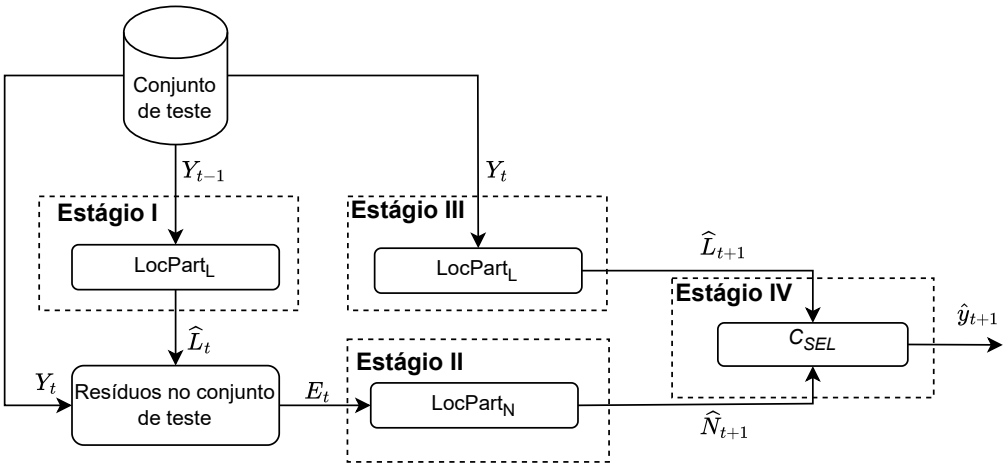

Figura 19 – Fase de teste do LocLN.

A Fig. 19 exibe a fase de teste do LocLN, e consiste em quatro estágios:

(I) **Estágio I**: A entrada do Estágio I são observações no passado do conjunto de teste, ou seja, é a série Y_{t-1} até $t-1$, a saída são as previsões lineares $\widehat{L}_t$ até t;

(II) **Estágio II**: A entrada do Estágio II é a série de resíduos E_t de $\widehat{L}_t$ no conjunto de teste até t, a saída é a previsão não linear $\widehat{N}_{t+1}$ em $t+1$;

(III) **Estágio III**: A entrada do Estágio III é a série Y_t no conjunto de teste até t, a saída é a previsão linear $\widehat{L}_{t+1}$ em $t+1$;

(IV) **Estágio IV**: As entradas do Estágio IV são a previsão linear $\widehat{L}_{t+1}$ em $t+1$ e a previsão não linear $\widehat{L}_{t+1}$ em $t+1$, a saída é a previsão da proposta $\widehat{y}_{t+1}$ em $t+1$.

Os objetivos dos quatro estágios da fase de teste são detalhados a seguir. O Estágio I envolve a geração das estimativas $\widehat{L}_t$ no conjunto de teste para o MHP linear $LocPart_L$ no passado até t. Assim, é possível obter os resíduos no conjunto de teste com padrões não lineares remanescentes do mapeamento linear do $LocPart_L$. Esses resíduos são a entrada do MHP não linear $LocPart_N$ do Estágio II. Portanto, no Estágio II, ocorre a previsão não linear $\widehat{N}_{t+1}$ dos resíduos no conjunto de teste em $t+1$ por meio do MHP não linear $LocPart_N$. Já no Estágio III é realizada a previsão linear $\widehat{L}_{t+1}$ da série no conjunto de teste em $t+1$ por meio do MHP linear $LocPart_L$. E, finalmente, no Estágio IV, emprega-se por meio do combinador C_{SEL} a combinação das saídas dos Estágios II e III, que são a previsão linear $\widehat{L}_{t+1}$ da série e a previsão não linear $\widehat{N}_{t+1}$ dos resíduos em $t+1$ no conjunto de teste. A saída do Estágio IV é a previsão de um passo à frente $\widehat{y}_{t+1}$ da proposta LocLN.

3.4 Resumo do Capítulo

Neste capítulo, as propostas foram descritas em detalhes. Primeiramente, o método LocPart em que a inovação está no estágio de geração, que envolve o uso de um loop para criar subconjuntos de tamanhos variados e em diferentes regiões do conjunto de treinamento. Em seguida, o método LocLinNonPR, a inovação está na fusão de um MHP, que emprega um mapeamento local, com uma técnica de MHS para o mapeamento de padrões lineares e não lineares. E por fim, o método LocLN, em que a inovação está na combinação do MHP LocPart com um MHS. No capítulo a seguir, serão apresentadas a metodologia e avaliações do trabalho.

4 METODOLOGIA E AVALIAÇÕES

Neste capítulo serão apresentadas a metodologia e as avaliações que foram empregadas nesta pesquisa.

4.1 As Bases de Dados de Velocidade do Vento

Figura 20 – Instalação de turbina eólica de eixo horizontal.

Fonte: Enel Green Power 2020.

Na Figura 20 é ilustrada a instalação de uma turbina eólica de eixo horizontal. Este tipo de turbina possui um eixo de rotação horizontal e em paralelo ao chão. É o modelo mais comumente utilizado para geração de energia eólica, está instalado em diversos locais no Brasil e é extremamente eficiente (LIMA, 2021). Em (SILVA, 2003), foi realizado um estudo sobre as características do vento na Região Nordeste. Além disso, foi investigado o comportamento operacional das turbinas eólicas de eixo horizontal. Demonstrou-se que os aerogeradores somente funcionam dentro de uma faixa de velocidade do vento. Em (LIMA, 2021), é relatado que atualmente há uma grande diversidade de modelos de turbinas existentes no mercado, e que a velocidade na qual os aerogeradores começam a girar encontra-se na faixa de velocidade de 3 a 5 m/s. Por esse motivo, foi levada em consideração a velocidade média dos ventos nas séries temporais selecionadas para essa pesquisa.

As bases de dados de velocidade do vento escolhidas são de um instituto brasileiro, o Instituto Nacional de Pesquisas Espaciais (INPE). Por meio de um projeto do INPE, surgiu o Sistema de Organização Nacional de Dados Ambientais (SONDA) para imple-

mentação da infra-estrutura física e de recursos humanos destinada a levantar e melhorar a base de dados dos recursos de energia solar e eólica no Brasil (INPE, 2020).

Devido ao tamanho imenso do território brasileiro, a instalação, operação e manutenção das estações INPE demandam horas de trabalho de vários especialistas e um alto investimento em equipamentos (INPE, 2020). O tratamento, armazenamento e disponibilização dos dados medidos é uma tarefa bem complexa. Ademais, os dados, antes de serem disponibilizados, passam por um processo de validação para garantir a confiabilidade. Os dados brutos em si não são modificados, mas sinalizados em um arquivo separado com informações que indicam o quão confiável são aqueles dados. Devido a relâmpagos, falhas de leitura, ou mesmo acidentes com animais podem alterar as medições tornando-as incorretas (INPE, 2020).

O processo de validação dos dados obtidos pelas estações SONDA baseia-se na estratégia de controle de qualidade de dados adotada pela *BSRN (Baseline Surface Radiation Network)* (INPE, 2020). A BSRN é um projeto do *World Climate Research Programme* e do *Global Energy and Water Cycle Experiment* que visa detectar mudanças importantes de radiação na superfície da Terra. Embora a *BSRN* trate apenas de radiação solar, sua estratégia de controle de dados é também aplicada em dados meteorológicos e anemométricos. Contudo, adotando-se os critérios de análise estabelecidos pela WEBMET (INPE, 2020). A WEBMET é um sistema internacional de registro e gerenciamento de observações meteorológicas adotado pela aeronáutica brasileira (REDEMET, 2020).

O processo de controle de qualidade é composto de 4 etapas sequenciais iniciadas com filtros mais grosseiros e terminadas com filtros mais refinados. Esses filtros sinalizam quando um dado é considerado suspeito de incorreção através da execução de algoritmos que adotam os seguintes critérios (INPE, 2020):

- Algoritmo 1 - Dado suspeito quando fisicamente impossível;
- Algoritmo 2 - Dado suspeito quando o evento é extremamente raro;
- Algoritmo 3 - Dado suspeito quando apresenta uma evolução temporal não condizente com o esperado para a variável;
- Algoritmo 4 - Dado suspeito quando inconsistente com medidas apresentadas por outras variáveis da mesma estação.

A aprovação em cada etapa é requisito para a continuidade do processo. Assim, somente quando um dado for considerado aprovado numa etapa, a etapa seguinte será iniciada. Se não houver aprovação, o processo será interrompido e o dado receberá o código equivalente a suspeito. Se houver aprovação, o dado receberá o código de aprovado (INPE, 2020).

Nesta pesquisa, as bases de dados selecionadas têm menos de 2% de dados suspeitos na 1ª etapa. Os detalhes sobre essas informações podem ser encontrados em (INPE, 2020).

Outra questão relevante, é que cada estação da rede SONDA possui vários sensores, que medem variáveis diferentes. Os dados escolhidos para essa pesquisa são de sensores de velocidade do vento a 50m de altura.

De acordo com (EóLICA, 2023b), o Nordeste brasileiro é uma região que tem um dos melhores ventos do mundo para produção de energia eólica, uma vez que são mais constantes, têm uma velocidade estável e não mudam de direção com frequência. Portanto, 90% dos parques eólicos brasileiros estão instalados na Região Nordeste (EóLICA, 2023b). Por esses motivos, as bases de dados selecionadas são de estações anemométricas no interior do Nordeste, nas cidades de Petrolina-PE e Triunfo-PE, compreendendo diferentes períodos do ano com duração de 3 meses. Assim, foi possível verificar como as propostas dos métodos de previsão se comportam em diferentes períodos do ano. Além disso, foi verificado o coeficiente de variação de cada base de dados, pois regiões de ventos com coeficiente de variação baixo permitem produção eficaz de energia eólica. Isto é, ventos com potencial de geração e mais estáveis despertam interesse das concessionárias de energia eólica (ARAUJO; MARINHO, 2019).

4.1.1 Pré-processamento dos Dados

As bases selecionadas do SONDA possuem um intervalo de $10min$ entre instâncias consecutivas com duração total de três meses. Cinco bases univariadas foram escolhidas, em que quatro bases possuem 13.248 instâncias e uma possui 12.816 instâncias, devido à inclusão do mês de fevereiro que possui menos dias. Não há instâncias ausentes. Posteriormente, dez bases de tamanho menor foram geradas aumentando-se a granularidade dos dados originais a partir de médias. Assim, para as primeiras cinco bases, a cada seis amostras, a média foi calculada, resultando em um total de 2.208 instâncias para as bases 1, 2, 3 e 4, e 2.136 instâncias para a base 5. Esse processo resultou em bases com intervalos de tempo de uma hora entre instâncias consecutivas. Conforme já mencionado no Capítulo 2, séries temporais horárias são bastante comuns em previsão de velocidade do vento. Ademais, séries temporais com intervalos de tempo de três horas entre instâncias consecutivas também foram consideradas, assim a cada 18 amostras das séries originais, a média foi calculada, resultando em um total de 736 instâncias para as bases 6, 7, 8 e 9, e 712 instâncias para a base 10.

Em seguida, cada série temporal foi completamente normalizada no intervalo [0, 1], devido à utilização da função de ativação sigmoide, utilizada nas redes neurais deste trabalho. O codomínio da função de ativação sigmoide é o intervalo [0, 1]. Portanto, de acordo com (DENNIS; ENGELBRECHT; OMBUKI-BERMAN, 2020), o problema de saturação deve ser levado em consideração durante a escolha da função de ativação. A saturação ocorre quando uma função de ativação gera valores próximos aos seus limites superiores ou inferiores. Esse fenômeno compromete a capacidade da rede de representar informações de forma eficaz, podendo dificultar ou até mesmo inviabilizar o treinamento da rede neural.

Logo, normalizar os dados de entrada para intervalos menores ajuda a evitar a saturação da rede.

4.1.2 Bases de Dados SONDA

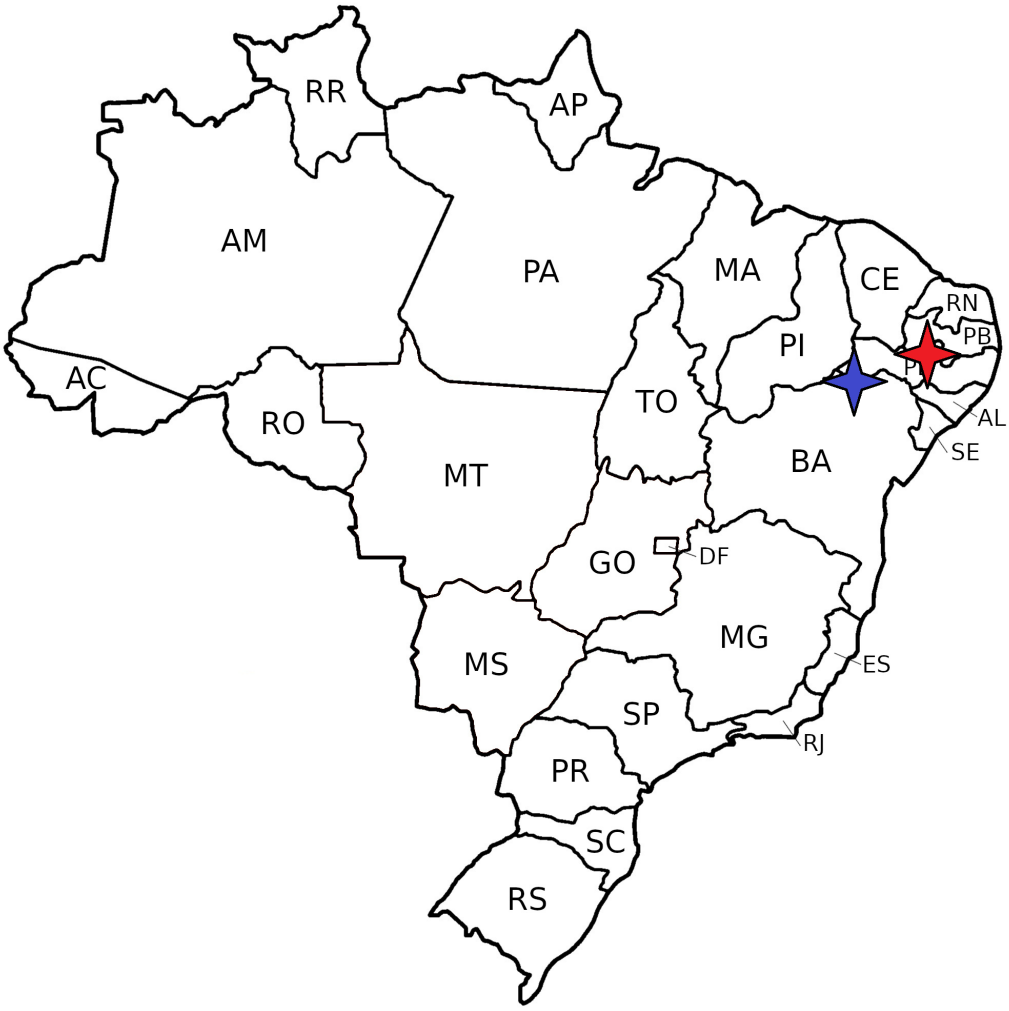

Figura 21 – Mapa do Brasil com a localização das cidades de Petrolina em azul e Triunfo em vermelho no estado de Pernambuco.

Fonte: (INFOESCOLA, 2025).

As bases de dados foram obtidas de duas estações anemométricas, uma na cidade de Petrolina, e outra na cidade de Triunfo, que ficam no estado de Pernambuco. A Figura 21 apresenta o mapa do Brasil com a localização das duas cidades. Petrolina é um município brasileiro do interior do estado de Pernambuco, situa-se a 9º 23' 34" de latitude sul e 40º 30' 28" de longitude oeste. Está na margem norte do Rio São Francisco e na divisa com o estado da Bahia. É considerada a mais importante cidade do sertão pernambucano. Está distante 712 km da capital de Pernambuco, Recife. Possui um clima semiárido, temperatura média de 27,2 ºC, com muito sol e poucas chuvas. Encontra-se a 376m de

altitude e umidade relativa de 55,8%. Triunfo é um município brasileiro também do interior do estado de Pernambuco e na divisa com o estado da Paraíba. Situa-se a 7º 50' 16" de latitude sul e 38º 06' 07" de longitude oeste. Encontra-se a 405km de Recife e a 380km de Petrolina. Está a 1.010m acima do nível do mar, sendo a cidade mais alta do estado de Pernambuco. Temperatura média de 21,4 ºC, e com umidade relativa de 71,4%. O que faz com que seu clima seja tropical semiúmido, ou seja, ameno e chuvoso, bem diferente do clima semiárido de Petrolina. A Tabela 2 resume esses dados.

Tabela 2 – Localização e dados climatológicos para as cidades de Petrolina-PE e Triunfo-PE.

Cidade	**Latitude**	**Longitude**	**Clima**	**Temperatura média**	**Altitude**	**Umidade relativa**
Petrolina-PE	9º 23' 34" S	40º 30' 28" O	Semiárido	27,2 ºC	376m	55,8%
Triunfo-PE	7º 50' 16" S	38º 06' 07" O	Tropical semiúmido	21,4 ºC	1.010m	71,4%

Tabela 3 – Resumo estatístico da velocidade do vento nas dez séries temporais da SONDA.

Série temporal	**Máximo**	**Média**	**Mínimo**	**Desvio Padrão**	**Coeficiente de variação**	**Tamanho**
Base 1 (Petrolina)	10,52 m/s	5,95 m/s	1,59 m/s	1,40 m/s	0,234	2.208
Base 2 (Triunfo)	16,76 m/s	9,35 m/s	3,17 m/s	2,68 m/s	0,287	2.208
Base 3 (Triunfo)	27,05 m/s	18,04 m/s	10,10 m/s	2,61 m/s	0,144	2.208
Base 4 (Triunfo)	23,05 m/s	12,52 m/s	3,29 m/s	3,12 m/s	0,249	2.208
Base 5 (Triunfo)	16,15 m/s	8,78 m/s	2,96 m/s	2,22 m/s	0,253	2.136
Base 6 (Petrolina)	9,81 m/s	5,95 m/s	2,44 m/s	1,25 m/s	0,210	736
Base 7 (Triunfo)	16,35 m/s	9,35 m/s	3,43 m/s	2,53 m/s	0,270	736
Base 8 (Triunfo)	26,36 m/s	18,04 m/s	11,47 m/s	2,50 m/s	0,138	736
Base 9 (Triunfo)	22,63 m/s	12,52 m/s	4,15 m/s	3,02 m/s	0,241	736
Base 10 (Triunfo)	15,30 m/s	8,78 m/s	3,63 m/s	2,08 m/s	0,237	712

Na Tabela 3 é apresentado um resumo estatístico das dez bases de dados univariadas utilizadas nesta pesquisa após a realização do pré-processamento, porém ainda sem a normalização. As bases de 1 a 5 são bases horárias, e as bases de 6 a 10 possuem uma granularidade maior com intervalos de tempo de 3h entre observações consecutivas. Percebe-se que o valor médio das bases horárias se repete nas bases com intervalos de 3h. Por exemplo, a base 1 tem a mesma média da base 6, isto porque a diferença entre essas duas bases é somente a granularidade dos dados. As séries que possuem o maior valor médio são as bases 3 e 8 em Triunfo-PE com o valor 18,04 m/s. Já as séries que possuem o menor valor médio são as bases 1 e 6 com o valor 5,95 m/s. Em relação às demais estatísticas, os valores entre as séries são bem diferentes. A velocidade máxima foi obtida pela base 3 em Trinfo-PE, 27,05 m/s. Enquanto que a velocidade mínima observada foi de 1,59 m/s na base 1 em Petrolina-PE. O maior desvio padrão foi de 3,12 m/s na base 4 em Triunfo-PE. O menor desvio padrão foi de 1,25 m/s na base 6 em Petrolina-PE. O maior coeficiente de variação foi obtido pela base 2 em Triunfo-PE com o valor de 0,287, já o menor coeficiente de variação observado foi de 0,138 na base 8 em Triunfo-PE. O tamanho das séries variam de 2.208 a 712 observações. É importante que as bases apresentem

estatísticas diferentes para a avaliação do comportamento das propostas desta pesquisa em previsão de velocidade do vento.

A seguir são apresentados os gráficos das bases 1 a 5 com a função de autocorrelação. Também serão ilustradas as componentes de sazonalidade e tendência obtidas por meio da decomposição STL.

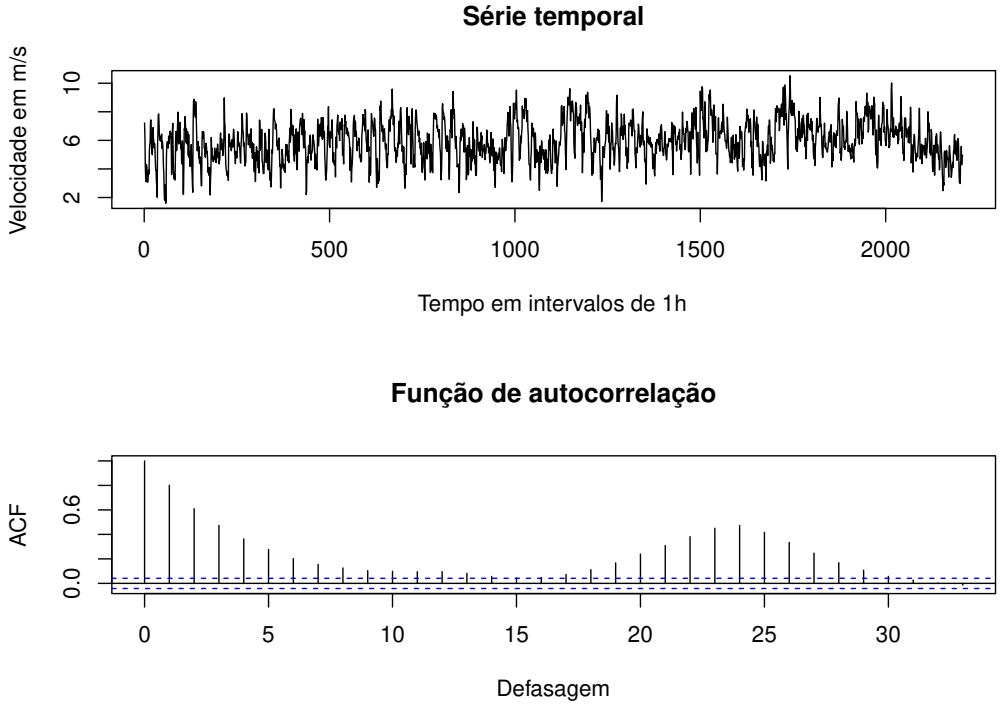

Figura 22 – Base de dados 1 - Evolução da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Petrolina-PE de julho a setembro de 2010, e função de autocorrelação.

A Figura 22 ilustra a velocidade do vento no decorrer do tempo em intervalos de 1h em Petrolina-PE de junho a setembro de 2010, e a função de autocorrelação. Essa é a base de dados 1 com 2.208 observações. Por meio da função de autocorrelação, observa-se que há evidências de uma sazonalidade na defasagem 24.

Na Figura 23 é apresentada a decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Petrolina-PE de julho a setembro de 2010. Observa-se uma sazonalidade não tão bem definida, com variações na amplitude. Também há variações no gráfico da tendência, demonstrando a complexidade nos padrões desta série temporal.

A Figura 24 ilustra a velocidade do vento no decorrer do tempo em intervalos de 1h em Triunfo-PE de outubro a dezembro de 2006, e a função de autocorrelação. Essa é a base de dados 2 com 2.208 observações. Por meio da função de autocorrelação, observa-se que há evidências de uma sazonalidade na defasagem 24.

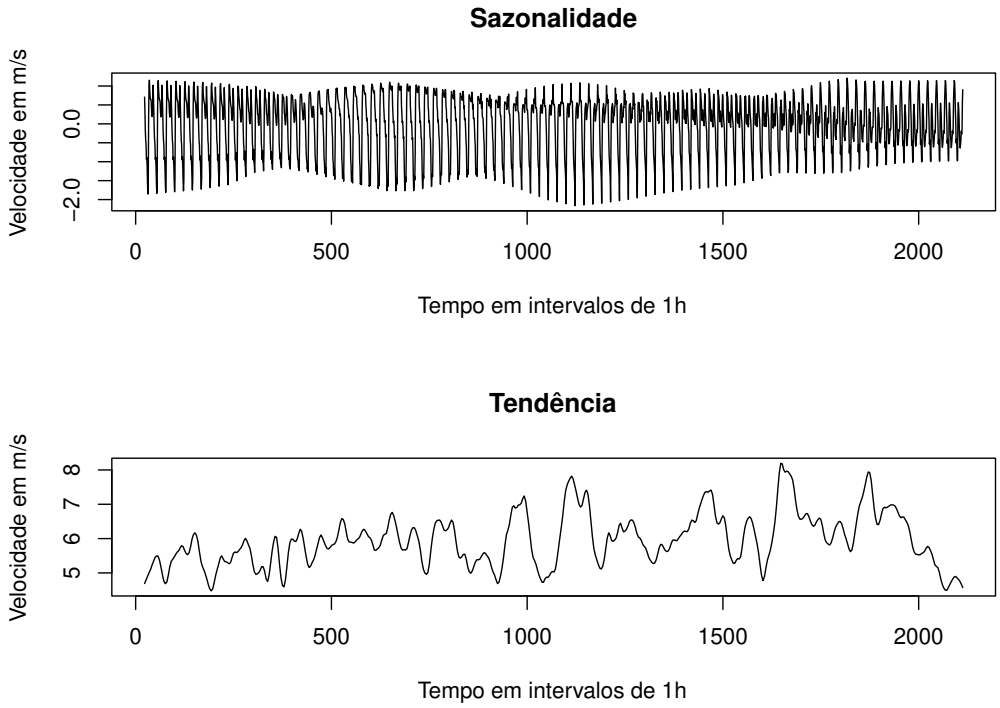

Figura 23 – Base de dados 1 - Decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Petrolina-PE de julho a setembro de 2010.

Na Figura 25 é apresentada a decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de outubro a dezembro de 2006. Nota-se a variação na amplitude da sazonalidade e variações na tendência, demonstrando a complexidade nos padrões desta série temporal.

A Figura 26 ilustra a velocidade do vento no decorrer do tempo em intervalos de 1h em Triunfo-PE de março a maio de 2006, e a função de autocorrelação. Essa é a base de dados 3 com 2.208 observações. Por meio da função de autocorrelação, observa-se que há correlações temporais significativas até a defasagem 24, e diminuem posteriormente. Essa forte dependência temporal entre defasagens adjacentes evidencia a possibilidade desta série possuir características de um processo de passeio aleatório, conhecido por sua forte dependência dos valores observados no passado mais recente.

Na Figura 27 é apresentada a decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de março a maio de 2006. Nota-se a variação na amplitude da sazonalidade e variações na tendência, demonstrando a complexidade nos padrões desta série temporal.

A Figura 28 ilustra a velocidade do vento no decorrer do tempo em intervalos de 1h em Triunfo-PE de julho a setembro de 2006, e a função de autocorrelação. Essa é a base de dados 4 com 2.208 observações. Por meio da função de autocorrelação, observa-se que

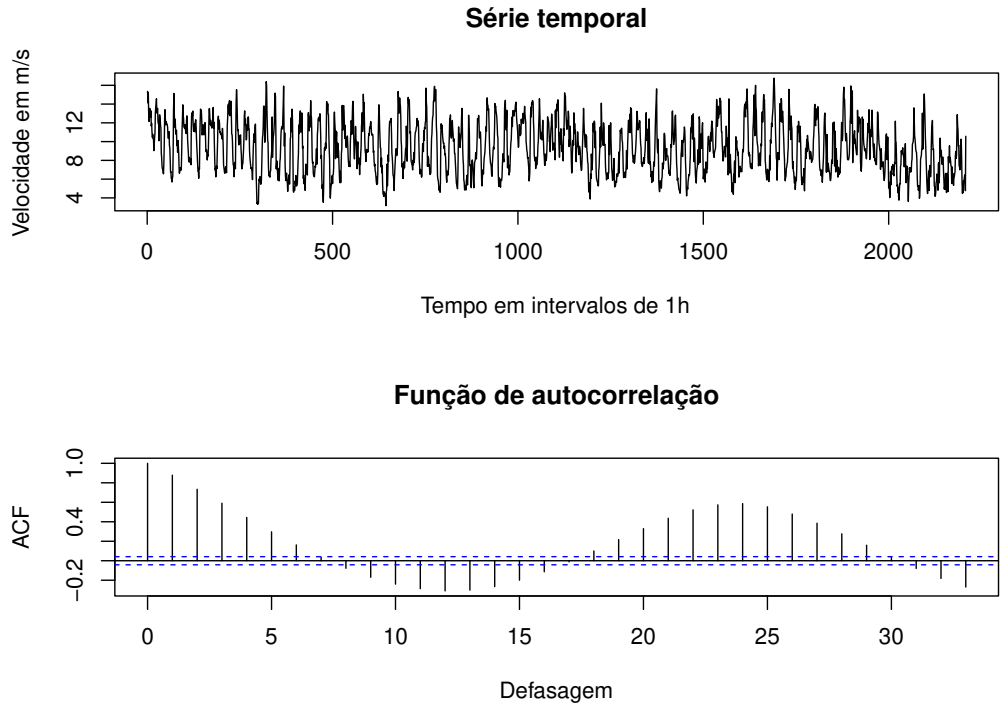

Figura 24 – Base de dados 2 - Evolução da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de outubro a dezembro de 2006., e função de autocorrelação.

há evidências de uma sazonalidade na defasagem 24.

Na Figura 29 é apresentada a decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de julho a setembro de 2006. Nota-se a variação na amplitude da sazonalidade e variações na tendência, demonstrando a complexidade nos padrões desta série temporal.

A Figura 30 ilustra a velocidade do vento no decorrer do tempo em intervalos de 1h em Triunfo-PE de fevereiro a abril de 2007, e a função de autocorrelação. Essa é a base de dados 5 com 2.136 observações. Por meio da função de autocorrelação, observa-se que há evidências de uma sazonalidade na defasagem 24.

Na Figura 31 é apresentada a decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de fevereiro a abril de 2007. Nota-se a variação na amplitude da sazonalidade e variações na tendência, demonstrando a complexidade nos padrões desta série temporal.

As demais cinco bases correspondem às cinco anteriores com uma granularidade três vezes maior. Isso significa que a sazonalidade, em vez de ocorrer na defasagem 24, passa a ocorrer na defasagem 8. Por sua vez, o comportamento da tendência permanece semelhante ao observado na respectiva base de menor granularidade apresentada anteriormente. Por exemplo, a Figura 32 ilustra a velocidade do vento no decorrer do tempo em intervalos de 3h em Petrolina-PE de junho a setembro de 2010, e sua função de autocorre-

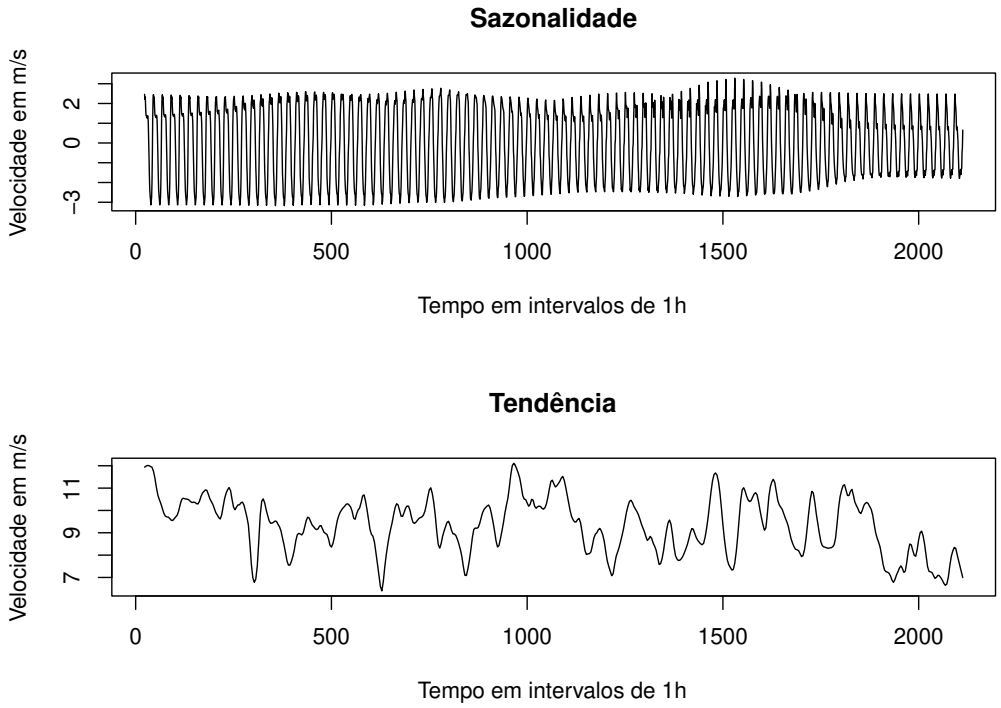

Figura 25 – Base de dados 2 - Decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de outubro a dezembro de 2006.

lação. Essa é a base de dados 6 com 736 observações, e a respectiva base de granularidade menor, é a base de dados 1 com 2.208 observações. Por meio da função de autocorrelação, observa-se que há evidências de uma sazonalidade na defasagem oito.

Na Figura 33 é apresentada a decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 3h em Petrolina-PE de julho a setembro de 2010. Observa-se que o comportamento da tendência é similar à respectiva base de granularidade menor, a base de dados 1. Por esse motivo, não há necessidade de apresentar graficamente a decomposição das demais quatro bases.

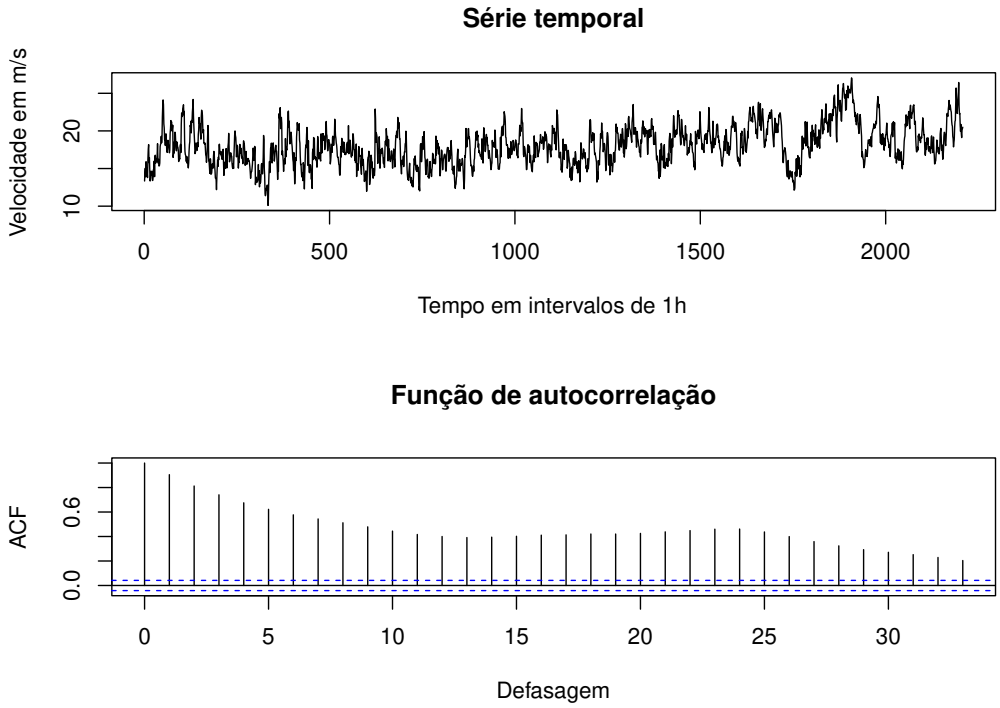

Figura 26 – Base de dados 3 - Evolução da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de março a maio de 2006, e a função de autocorrelação.

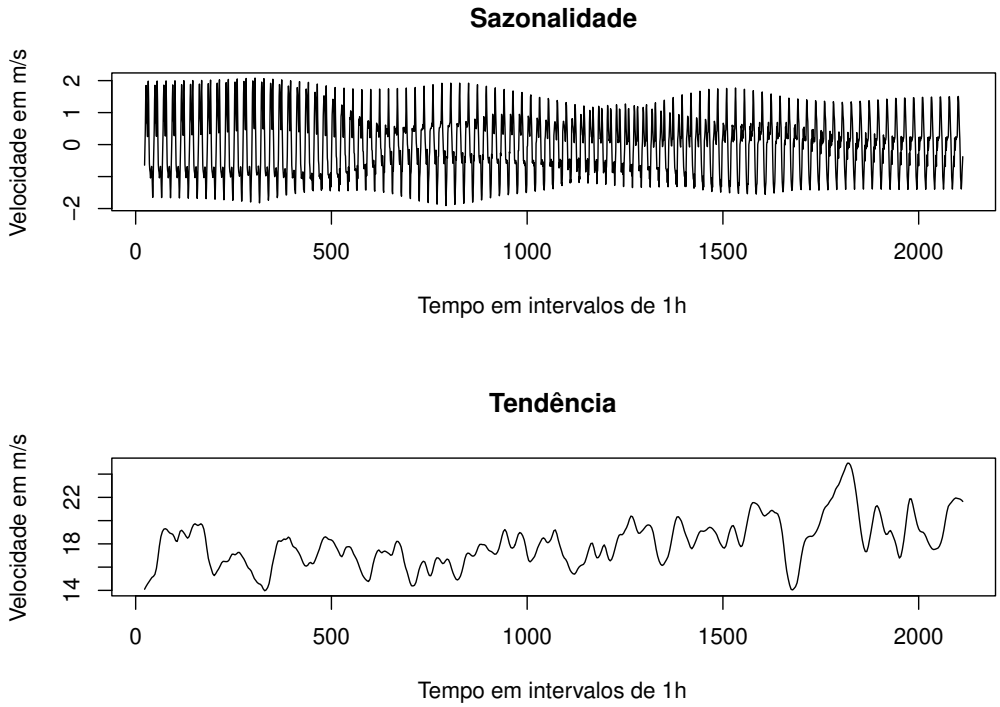

Figura 27 – Base de dados 3 - Decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de março a maio de 2006.

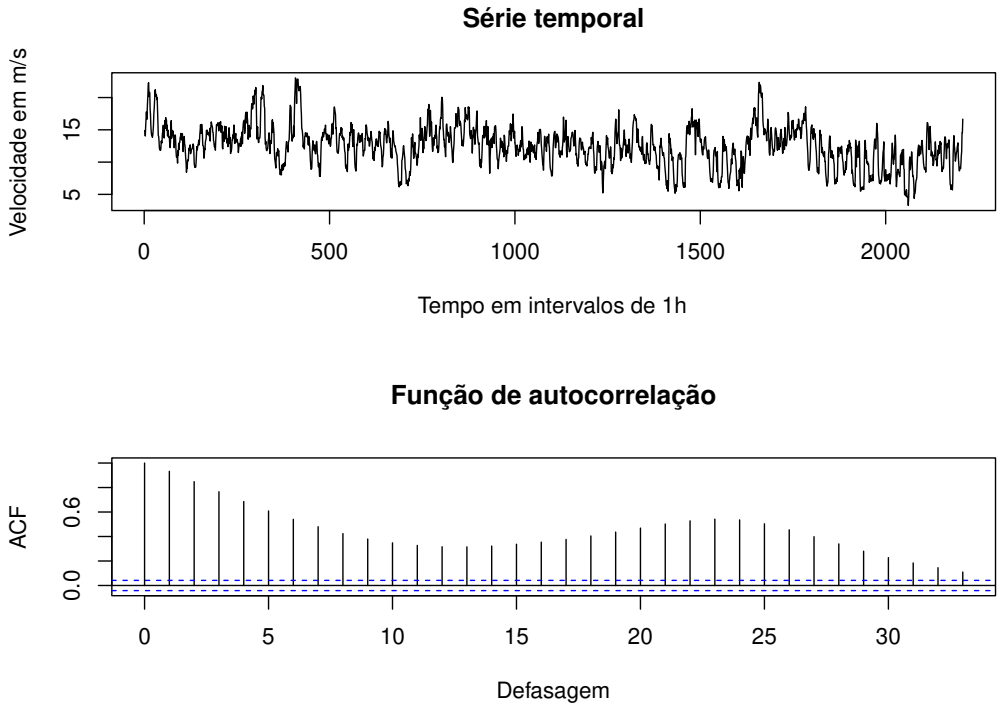

Figura 28 – Base de dados 4 - Evolução da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de julho a setembro de 2006, e função de autocorrelação.

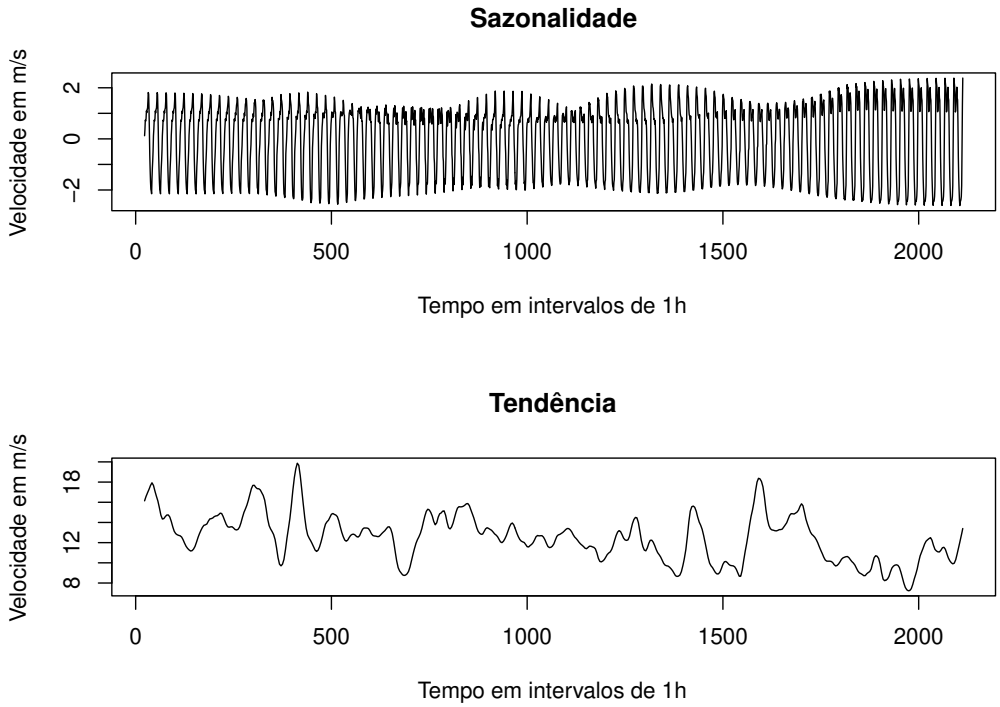

Figura 29 – Base de dados 4 - Decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de julho a setembro de 2006.

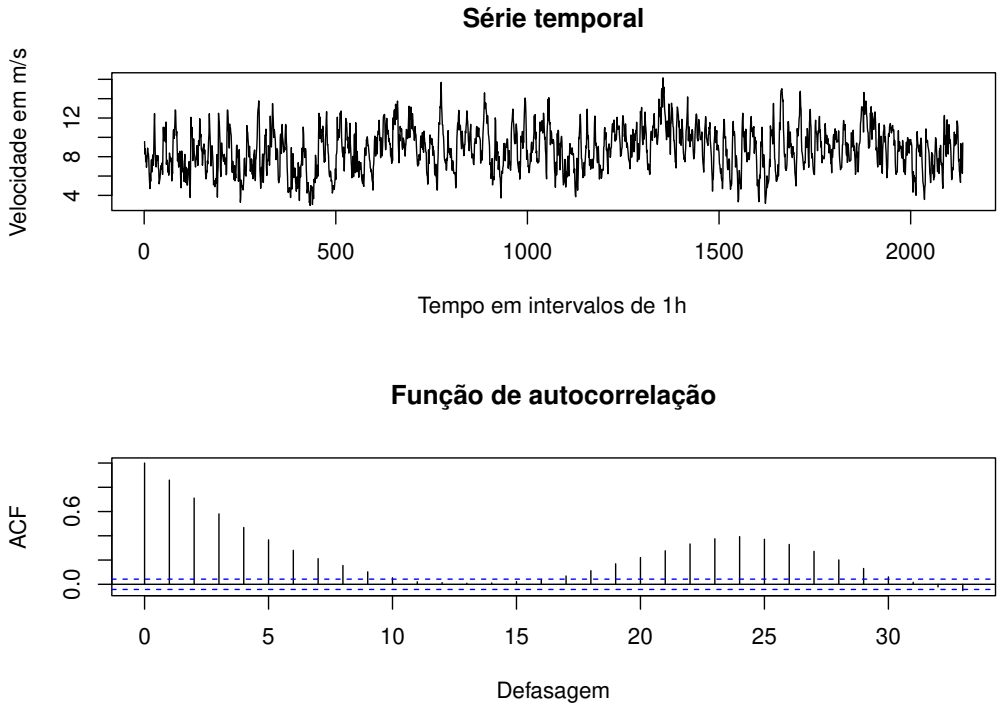

Figura 30 – Base de dados 5 - Evolução da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de fevereiro a abril de 2007, e função de autocorrelação.

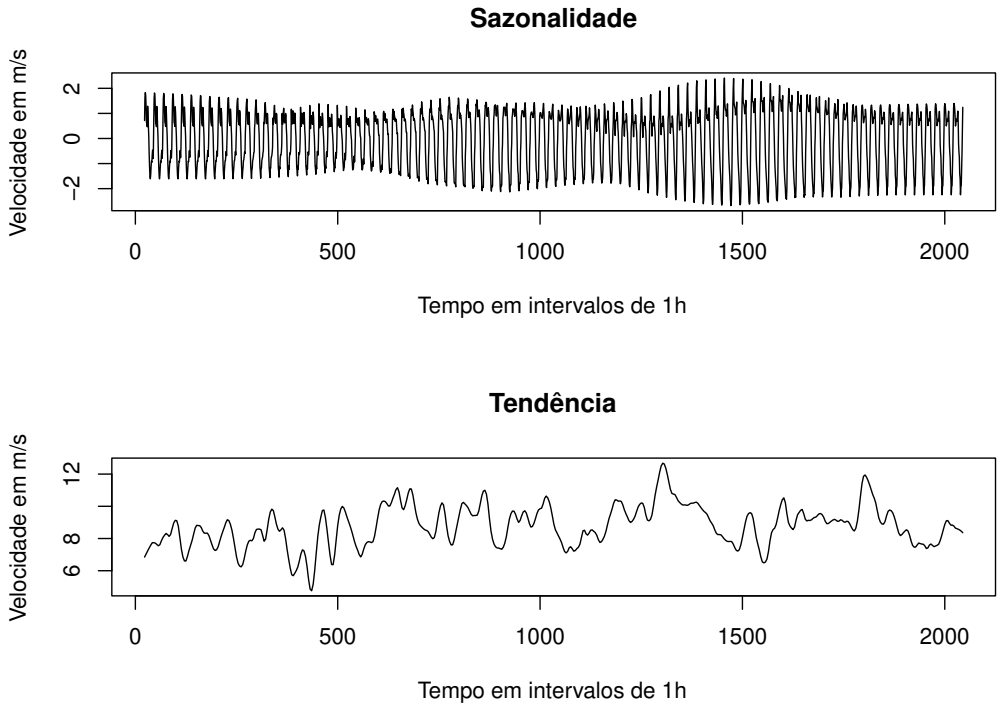

Figura 31 – Base de dados 5 - Decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 1h em Triunfo-PE de fevereiro a abril de 2007.

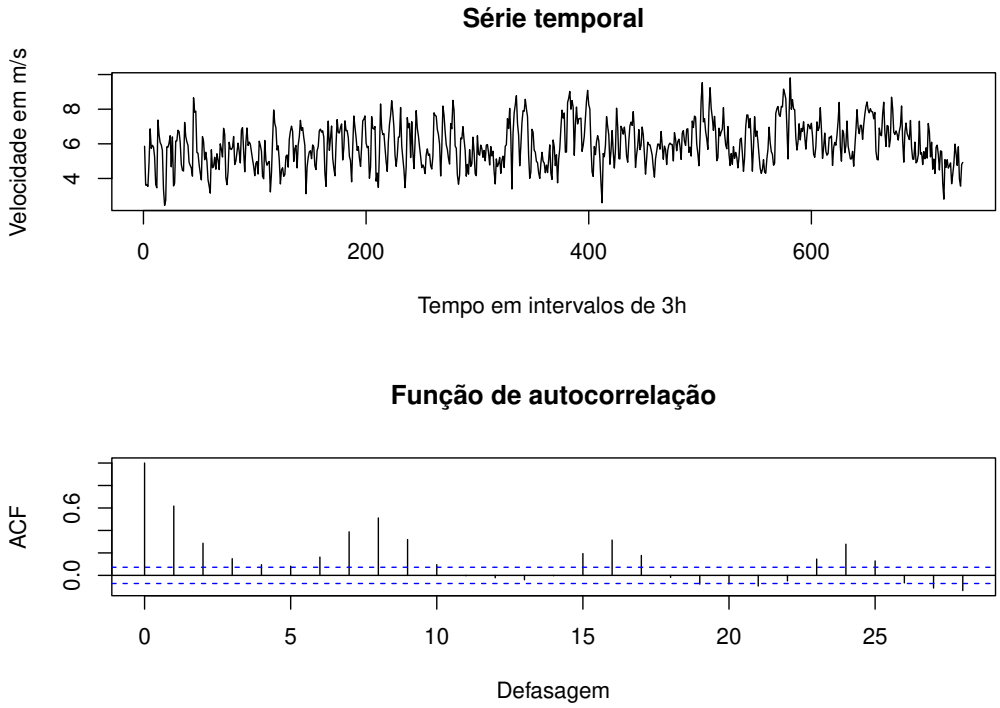

Figura 32 – Base de dados 6 - Evolução da velocidade do vento em m/s no decorrer do tempo em intervalos de 3h em Petrolina-PE de julho a setembro de 2010, e função de autocorrelação.

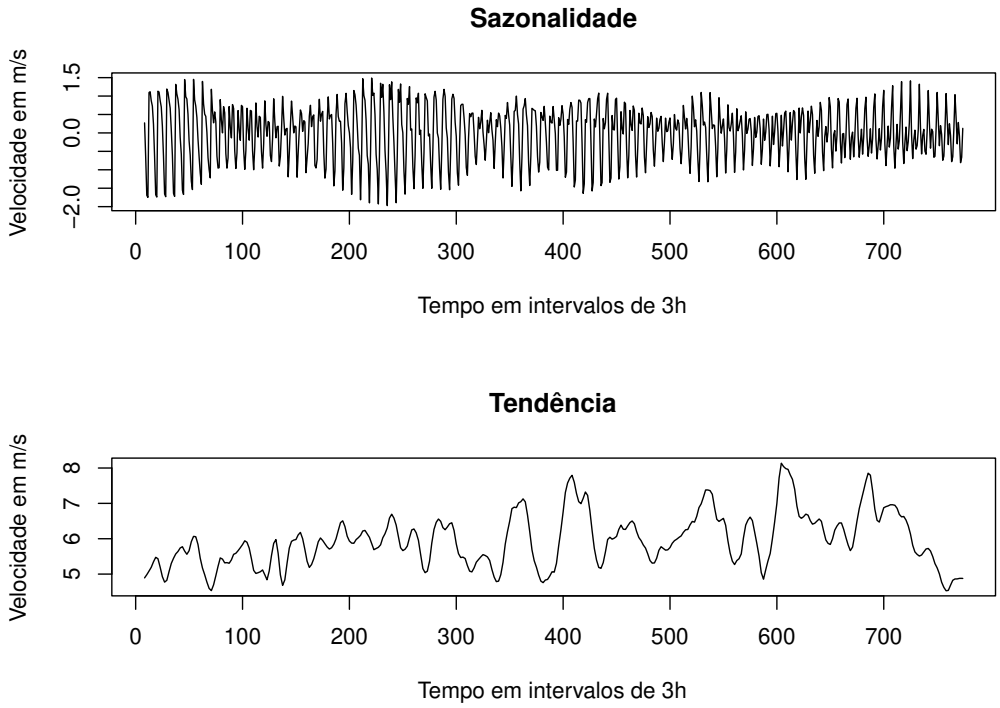

Figura 33 – Base de dados 6 - Decomposição por meio do método STL em sazonalidade e tendência da velocidade do vento em m/s no decorrer do tempo em intervalos de 3h em Petrolina-PE de julho a setembro de 2010.

4.2 Protocolo Experimental

Nessa seção são descritos os procedimentos utilizados para implementação dos métodos utilizados neste trabalho. Desde a linguagem utilizada, bibliotecas, particionamento dos dados e parâmetros empregados. Todas as simulações foram realizadas para previsão de um passo à frente, conforme é comumente utilizado na literatura em trabalhos comparativos de desempenho de modelos de previsão. A linguagem utilizada foi a *R* (R Core Team, 2024) com as seguintes bibliotecas:

- *Forecast* - Fornece métodos e ferramentas para exibir e analisar previsões de séries temporais univariadas, além da modelagem ARIMA (HYNDMAN et al., 2017).
- *RSNNS* - Contém muitas implementações para RNAs a partir de toda a funcionalidade algorítmica e flexibilidade do pacote *Stuttgart Neural Network Simulator* (SNNS) (BERGMEIR; BENíTEZ, 2017).
- *E1071* - Fornece todas as implementações do popular pacote LibSVR, muito utilizado para modelagem do SVR (MEYER et al., 2017).
- *ElmNNRcpp* - É uma reimplementação do *elmNN* usando *RcppArmadillo*. Implementa funções de treinamento e previsão para RNAs de camada única usando o algoritmo Extreme Learning Machine (ELM) (MOUSELIMIS; JONGE, 2022).
- *Tensorflow* (ABADI et al., 2015) - É uma biblioteca de código aberto para aprendizado de máquina desenvolvida pela *Google*. Foi utilizada neste trabalho para a codificação do LSTM e do GRU.
- *Keras* (CHOLLET et al., 2015) - É uma biblioteca de rede neural de código aberto escrita em Python capaz de fornecer uma interface amigável para a utilização do *tensorflow*.

4.2.1 LocPart

Inicialmente, o LocPart usou quatro subconjuntos disjuntos sequenciais $I = 4$, cada um contendo $1.104/4 = 276$ observações para as bases de dados 1, 2, 3 e 4, e $1.068/4 = 267$ observações para a base de dados 5. Um valor de incremento por *loop* de $L = 4$ e um critério de parada de $J = 20$ foram adotados para os modelos ARIMA, ELM, LSTM e GRU. Assim, um total de 60 modelos base foram obtidos na fase de geração para o reconhecimento de padrões locais. Da base de dados 6 até 10, inicialmente, o LocPart usou dois subconjuntos disjuntos sequenciais $I = 2$, cada um contendo $368/2 = 184$ observações para as bases de dados 6, 7, 8 e 9, e $356/2 = 178$ observações para a base de dados 10. Um valor de incremento por *loop* de $L = 2$ e um critério de parada de $J = 8$ foram adotados para os modelos ARIMA, ELM, LSTM e GRU. Para $J = 8$ o tamanho do subconjunto

ficou igual a $368/8 = 46$, o que é considerado uma quantidade pequena de dados para ajuste de parâmetros de um modelo (CERQUEIRA; TORGO; SOARES, 2022). Assim, um total de 20 modelos base foram obtidos na fase de geração para o ARIMA aplicado na série temporal. Os modelos ELM, LSTM e GRU tiveram problemas de convergência em $J = 8$. Desta forma, o critério de parada foi reduzido para $J = 6$. Consequentemente, foi possível gerar um total de 12 modelos ELM, LSTM e GRU. Na fase de seleção do LocPart, a métrica de avaliação V utilizada foi o erro quadrático médio (RMSE, do inglês *Root Mean Squared Error*) no conjunto de seleção, e o método de seleção desenvolvido em (ADHIKARI; VERMA; KHANDELWAL, 2015) foi empregado. Por fim, na etapa de integração, o LocPart utilizou a média como método de combinação C, que apesar de simples, ainda é considerada robusta, e serve como medida mínima de desempenho para combinadores mais sofisticados (SERGIO; LIMA; LUDERMIR, 2016). O RMSE é definido pela Eq. (4.1):

$$RMSE = \sqrt{\left(\frac{\sum_{t=1}^{T}(y_t - \widehat{y}_t)^2}{T}\right)}, \tag{4.1}$$

em que T é o tamanho da série, y_t é o valor real da série temporal no período t e $\widehat{y}_t$ é o valor previsto da série temporal no período t. Quanto menor o RMSE, melhor é a qualidade do previsor.

4.2.1.1 Particionamento das Bases de Dados para o LocPart

Cada série temporal foi dividida em três subconjuntos: os primeiros 50% para treinamento, os 25% seguintes para seleção, e os últimos 25% para teste. O conjunto de treinamento foi usado para geração dos subconjuntos de treinamento do LocPart, o conjunto de seleção para a etapa de seleção do LocPart, e o conjunto de teste para obter o desempenho preditivo realista do modelo em dados que não participaram das etapas de geração nem seleção. Após a criação dos subconjuntos dentro do conjunto de treinamento, cada modelo ARIMA usou o seu respectivo subconjunto para ajuste de parâmetros. Os modelos ELM, LSTM e GRU usaram 50% de seus respectivos subconjuntos para treinamento e os demais 50% para validação, a fim de evitar *overfitting*.

4.2.1.2 Parâmetros dos Modelos Base do LocPart

Para o modelo base ARIMA, o parâmetro inteiro d foi escolhido de acordo com o teste de hipóteses ADF para identificar a estacionariedade. Enquanto p e q foram obtidos no intervalo $[0, 1, ..., 24]$, pois existe uma sazonalidade de 24 lags nas séries temporais horárias. Para o ajuste do modelo foi aplicado o método *Conditional Sum of Squares* (CSS) e o modelo com o menor AICc foi selecionado. No modelo ARIMA foram considerados os requisitos de estabilidade para seleção dos parâmetros em relação à estacionariedade e invertibilidade. Logo, modelos não estacionários e não invertíveis são descartados automaticamente.

A técnica de busca em grade foi aplicada para os modelos base ELM, LSTM e GRU. De acordo com (HSU; CHANG; LIN, 2016), há motivações para justificar a escolha da técnica

busca em grade. Pois, métodos que evitem fazer uma pesquisa exaustiva de parâmetros por aproximações ou heurísticas nem sempre são seguros. Ou seja, algoritmos inteligentes de busca podem falhar em encontrar a solução ótima, porque a população pode ficar presa num ótimo local. Outro ponto importante é que a execução da busca em grade pode ser facilmente paralelizada quando os parâmetros são independentes. Portanto, para o ELM, os parâmetros foram selecionados no intervalo $[1, 2, ..., 24]$ para os nós de entrada e para os nós da camada oculta. As funções de ativação para camadas ocultas e de saída foram a sigmoide e a identidade, respectivamente. Quanto aos modelos LSTM e GRU foi empregada uma rede *stateful* com parâmetros selecionados no intervalo $[1, 6, 12, 18, 24]$ para os nós de entrada, e no intervalo $[6, 12, 18, 24]$ para os nós ocultos. A função de ativação da camada oculta foi a sigmóide. O LSTM e o GRU foram treinados por 100 épocas e tamanho do *batch* igual a 1. O método de otimização para o treinamento foi o Adam.

4.2.2 LocLinNonPR

O MHP linear que faz parte do estágio I do LocLinNonPR foi composto de quatro modelos ARIMA, que são aplicados em quatro subconjuntos de tamanhos iguais e regiões diferentes no conjunto de treinamento. Isto possibilita o reconhecimento de padrões locais lineares na série temporal, e o ARIMA está entre os modelos lineares de séries temporais mais conhecidos (AHMADI; KHASHEI, 2021). Esse MHP é integrado por meio da média. Em seguida, no estágio II do LocLinNonPR, o modelo individual não linear ELM é aplicado nos resíduos do MHP linear, e assim é realizado o mapeamento global não linear dos padrões nos resíduos da série. A combinação da previsão do MHP linear na série com a previsão do modelo não linear ELM nos resíduos foi realizada pelo modelo não linear ELM. O ELM é uma rede neural de treinamento rápido com resultados competitivos em comparação com MLP e SVM (SAAVEDRA-MORENO et al., 2013).

4.2.2.1 Particionamento das Bases de Dados para o LocLinNonPR Cada série temporal foi dividida em três subconjuntos: os primeiros 50% para treinamento, os 25% seguintes para seleção, e os últimos 25% para teste. Para o MHP linear, o conjunto de treinamento foi dividido em quatro subconjuntos iguais.

4.2.2.2 Parâmetros dos Modelos do LocLinNonPR A seleção dos parâmetros dos modelos ocorreu por meio da mesma busca em grade que está descrita para os modelos base ARIMA e ELM no LocPart. Porém, para a etapa de combinação, o modelo combinador ELM teve os parâmetros selecionados no intervalo $[1, 2, ..., 48]$ para os nós da camada oculta. Assim, foi possível obter uma combinação mais apropriada.

4.2.3 LocLN

A proposta do LocLN é um método híbrido com quatro estágios. No Estágio I, o MHP LocPart foi empregado com o modelo M_L linear ARIMA para mapear os padrões locais lineares da série temporal. No Estágio II, os padrões dos resíduos da série temporal foram mapeados localmente não linearmente usando o MHP LocPart com o modelo M_N não linear ELM. No Estágio III, as saídas dos dois estágios anteriores foram combinadas usando uma combinação M_C não linear por meio do modelo ELM. E, finalmente, no Estágio IV, a métrica V que faz a seleção do método de combinação entre o linear ADD e o não linear M_C foi o RMSE. Os parâmetros do LocPart nos estágios I e II do LocLN foram os mesmos descritos anteriormente para o método LocPart.

4.2.3.1 Particionamento das Bases de Dados para o LocLN Cada série temporal foi dividida em três subconjuntos: os primeiros 50% para treinamento, os 25% seguintes para seleção, e os últimos 25% para teste. O conjunto de treinamento foi usado para geração dos subconjuntos de treinamento do LocPart, o conjunto de seleção para a etapa de seleção do LocPart e seleção do combinador C_{SEL} do LocLN. O conjunto de teste para obter o desempenho preditivo realista do modelo em dados que não participaram das etapas de geração nem seleção. Após a criação dos subconjuntos dentro do conjunto de treinamento, cada modelo ARIMA usou o seu respectivo subconjunto para ajuste de parâmetros, enquanto que os modelos ELM usaram 50% de seus respectivos subconjuntos para treinamento e os demais 50% para validação para evitar *overfitting*.

4.2.3.2 Parâmetros dos Modelos do LocLN A seleção dos parâmetros dos modelos ocorreu por meio da mesma busca em grade que está descrita para os modelos base ARIMA e ELM no LocPart. Porém, para a etapa de combinação, o modelo combinador ELM teve os parâmetros selecionados no intervalo $[1, 2, ..., 48]$ para os nós da camada oculta. Assim, foi possível obter uma combinação mais apropriada.

4.2.4 Modelos Individuais

Cinco modelos monolíticos, tais quais, modelo de persistência, ARIMA, ELM, LSTM e GRU foram implementados. Estes modelos individuais foram treinados globalmente em todo o conjunto de treinamento.

4.2.4.1 Particionamento das Bases de Dados nos Modelos Individuais Cada série temporal foi dividida em três subconjuntos: os primeiros 50% para treinamento, 25% para validação, e os últimos 25% para teste. O conjunto de treinamento foi usado para a aprendizagem dos modelos individuais, o conjunto de validação para a seleção de parâmetros dos modelos não lineares, e o conjunto de teste para obter o poder preditivo

realista do modelo em dados que não participaram do treinamento, nem da seleção de parâmetros.

4.2.4.2 Parâmetros dos Modelos Individuais A seleção dos parâmetros do ARIMA, ELM, LSTM e GRU foi a mesma dos modelos base ARIMA, ELM, LSTM e GRU do método LocPart, já descritos anteriormente.

4.2.5 Modelos híbridos Paralelos (MHPs)

Dois MHPs empregaram a técnica de *bagging* desenvolvida em (BERGMEIR; HYNDMAN; BENÍTEZ, 2016). O *bagging* tem o objetivo de diminuir as incertezas de previsão ao utilizar reamostragens do conjunto de treinamento gerando 100 preditores (BERGMEIR; HYNDMAN; BENÍTEZ, 2016). Portanto, o *bagging* foi aplicado nos modelos base ARIMA e ELM, e assim foram obtidos dois MHPs homogêneos, em que os modelos base foram treinados globalmente 100 vezes em séries reamostradas. Esta abordagem tem um alto custo computacional, e por esse motivo o LSTM e GRU não foram incluídos.

4.2.5.1 Particionamento das Bases de Dados nos MHPs As séries temporais foram divididas em três subconjuntos: os primeiros 50% para treinamento, 25% para validação, e os últimos 25% para teste. O conjunto de treinamento foi usado para a reamostragem por *bagging*, o conjunto de validação para seleção dos parâmetros dos modelos não lineares e o teste para a avaliação realista da acurácia.

4.2.5.2 Parâmetros dos Modelos nos MHPs A seleção dos parâmetros do ARIMA, ELM, LSTM e GRU foi a mesma dos modelos base ARIMA, ELM, LSTM e GRU do método LocPart, conforme já descrito anteriormente.

4.2.6 Modelos Híbridos Seriais (MHSs)

Seis MHSs com mapeamento global foram utilizados. O modelo ARIMA, conforme sugerido por (ZHANG, 2003), (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017), (CAMELO et al., 2018a), (de MATTOS NETO et al., 2020b), (SANTOS JÚNIOR et al., 2023) e (ALMEIDA, 2024), foi aplicado à série temporal para o reconhecimento global de padrões lineares, enquanto os modelos não lineares MLP, SVR e ELM foram empregados nos resíduos para o reconhecimento global de padrões não lineares. Três modelos usaram a combinação linear entre a previsão da série temporal e a previsão residual com base em (ZHANG, 2003), (de MATTOS NETO et al., 2020) e (SANTOS JÚNIOR et al., 2023). Ademais, outros três modelos utilizaram a combinação não linear baseada em (de MATTOS NETO; CAVALCANTI; MADEIRO, 2017) por meio dos modelos SVR e ELM. O SVR foi usado com sucesso como um combinador não linear em (de MATTOS NETO et al., 2020), e o ELM se destaca por

seu treinamento rápido, resultados competitivos em relação a MLP e SVR, e foi utilizado como combinador não linear em (ALMEIDA, 2024).

4.2.6.1 Particionamento das Bases de Dados para os MHSs Cada série temporal foi dividida em três subconjuntos: os primeiros 50% para treinamento, 25% para validação, e os últimos 25% para teste. O conjunto de treinamento foi usado para a aprendizagem dos modelos individuais, o conjunto de validação para a seleção de parâmetros dos modelos não lineares, e o conjunto de teste para obter o poder preditivo realista do modelo em dados que não participaram do treinamento, nem da seleção de parâmetros.

4.2.6.2 Parâmetros dos MHSs A seleção dos parâmetros dos modelos ocorreu da mesma forma que está descrito para os modelos base ARIMA e ELM do método LocPart. Porém, o modelo combinador ELM teve os parâmetros selecionados no intervalo $[1, 2, ..., 48]$ para os nós da camada oculta, pois, neste caso, ao aumentar a quantidade de nós, foi possível obter uma combinação mais apropriada. Quanto ao modelo MLP, os parâmetros foram selecionados no intervalo $[1, 6, 12, 18, 24]$ para os nós de entrada, e no intervalo $[3, 6, 9, ..., 24]$ para os nós da camada oculta. As funções de ativação foram a sigmoide e identidade para os nós ocultos e de saída, respectivamente. O método de aprendizagem foi o RPROP com número máximo de iterações (2000), valor de atualização inicial (0), limite máximo para o tamanho do passo (30) e o parâmetro de decaimento de peso α no intervalo $[1, 3, 5, ..., 11]$. O tipo do SVR foi o ϵ-regressão com o *kernel* RBF. A entrada utilizou o método de janela deslizante cuja largura foi selecionada no intervalo $[1, 6, 12, 18, 24]$. O parâmetro C da formulação de otimização foi ajustado no intervalo $[2^0, 2^1, ..., 2^{10}]$, enquanto que a região de tolerância ϵ e o parâmetro λ do *kernel* RBF foram selecionados no intervalo $[2^{-10}, 2^{-9}, ..., 2^0]$.

4.2.7 Tabela com Todos os Métodos Selecionados da Literatura

A Tabela 4 ilustra todos os métodos implementados neste trabalho e seus respectivos acrônimos.

4.2.8 Métricas de Avaliação da Qualidade

Para cada série temporal de velocidade do vento, a seleção dos parâmetros dos modelos é realizada de acordo com o menor RMSE no conjunto de validação para os modelos não lineares, e no treinamento para os modelos lineares.

No cálculo da métrica RMSE, os erros são elevados ao quadrado antes de serem obtidos, portanto, a RMSE fornece um peso maior para erros grandes. Por essa razão, não é recomendável utilizar unicamente a RMSE como uma medida conclusiva para comparação de diferentes modelos de previsão. Dessa forma, para estimar o desempenho real dos

Tabela 4 – Abordagens de previsão utilizadas neste trabalho com os métodos correspondentes e os respectivos acrônimos.

Abordagem	**Método**	**Acrônimo**
Individual	Modelo de persitência	Persitência
	Autoregressive integrated moving average (TORRES et al., 2005)	ARIMA
	Extreme learning machines (SAAVEDRA-MORENO et al., 2013)	ELM
	Long short-term memory (LIU; LIN; FENG, 2021)	LSTM
	Gated recurrent units (LIU; LIN; FENG, 2021)	GRU
MHP	Mean($ARIMA_1$,$ARIMA_2$,...,$ARIMA_{100}$) based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	$BAGG_{ARIMA}$
	Mean(ELM_1,ELM_2,...,ELM_{100}) based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	$BAGG_{ELM}$
MHS	ARIMA+MLP (CADENAS; RIVERA, 2010)	LC_{AM}
	ARIMA+SVR (de MATTOS NETO et al., 2020)	LC_{AS}
	SVR(ARIMA,MLP) (de MATTOS NETO et al., 2020)	$NC - S_{AM}$
	SVR(ARIMA,SVR) (de MATTOS NETO et al., 2020)	$NC - S_{AS}$
	ARIMA+ELM	LC_{AE}
	ELM(ARIMA,ELM) (ALMEIDA, 2024)	$NC - E_{AE}$
Propostas de MHP com mapeamento local	Mean($ARIMA_1$,$ARIMA_2$,...,$ARIMA_n$) (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	$LocPart_{ARIMA}$
	Mean(ELM_1,ELM_2,...,ELM_n) (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	$LocPart_{ELM}$
	Mean($LSTM_1$,$LSTM_2$,...,$LSTM_n$) (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	$LocPart_{LSTM}$
	Mean(GRU_1,GRU_2,...,GRU_n) (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	$LocPart_{GRU}$
Propostas de MHP com MHS e mapeamento local	ELM(Mean($ARIMA_1$,$ARIMA_2$,$ARIMA_3$,$ARIMA_4$), ELM) (ALMEIDA, 2024)	LocNonLinPR
	C_{SEL}(Mean($ARIMA_1$,$ARIMA_2$,...,$ARIMA_n$), Mean(ELM_1,ELM_2,...,ELM_n))	LocLN

modelos no conjunto de teste, além da RMSE, foram adotadas outras três métricas: erro absoluto médio (MAE, do inglês *Mean Absolute Error*), erro absoluto médio percentual (MAPE, do inglês *Mean Absolute Percentage Error*) e previsão de mudança na direção (POCID, do inglês *Prediction Of Change In Direction*). Para as métricas MAE e MAPE, quanto menores seus valores, melhor a acurácia do modelo. No caso da POCID, quanto maior o valor, melhor o desempenho do modelo.

A MAE é definida por:

$$MAE = \frac{\sum_{t=1}^{T} |y_t - \widehat{y}_t|}{T}. \tag{4.2}$$

Na métrica MAE, cada erro contribui para a métrica proporcionalmente ao valor absoluto do erro.

A seguir é apresentada a formulação da métrica MAPE:

$$MAPE = \frac{100}{T} \cdot \sum_{t=1}^{T} \frac{|y_t - \widehat{y}_t|}{y_t}. \tag{4.3}$$

A MAPE é uma das medidas de acurácia de previsão mais amplamente utilizadas, devido as suas vantagens de independência de escalas e interpretabilidade. No entanto, tem sido criticada, porque coloca penalidades mais pesadas em erros positivos do que em erros negativos (HYNDMAN; ATHANASOPOULOS, 2013). Em outras palavras, na métrica MAPE um desvio de $+0,5$ gera um erro maior do que um desvio de $-0,5$.

A métrica POCID é definida por:

$$POCID = 100\frac{\sum_{t=1}^{T} Trend_t}{T}, \tag{4.4}$$

em que

$$Trend_t = \begin{cases} 1, & \text{se } (y_t - y_{t-1})(\widehat{y}_t - \widehat{y}_{t-1}) > 0; \\ 0, & \text{caso contrário.} \end{cases} \tag{4.5}$$

Com a métrica POCID é possível identificar modelos mais acurados com relação à previsão de tendência da série. Um valor $POCID > 50$, indica que o modelo de previsão acertou mais do que errou a tendência da série. Porém, se $POCID < 50$, então o modelo de previsão errou mais do que acertou a tendência da série. O caso ideal é $POCID = 100$, este valor indica que o modelo sempre acertou a tendência da série.

Por fim, uma nova métrica baseada na MAPE foi definida em que somente são consideradas observações acima da média do conjunto de teste. Essa métrica é importante para a área de geração de energia eólica, porque é nos momentos que a velocidade do vento está acima da média que ocorre a maior geração de energia eólica. Essa nova métrica foi intitulada de MAPE_{media}:

$$MAPE_{media} = \frac{100}{T} \cdot \sum_{t=1}^{T} \frac{|y_t - \widehat{y}_t|}{y_t}, \text{ se } y_t > \overline{Y}. \tag{4.6}$$

Em que $\overline{Y}$ é a média das observações no conjunto de teste.

4.2.9 Métrica de Relação Percentual

Para comparar a acurácia da proposta em relação aos outros métodos utilizando uma métrica de avaliação específica, foi definida uma relação percentual R_i tal que:

$$\begin{aligned} R_i &= \left(\tfrac{d_i}{i_{ref}}\right) \cdot 100\%, \\ d_i &= \begin{cases} i_{prop} - i_{ref}, & \text{se } i = \text{POCID}; \\ i_{ref} - i_{prop}, & \text{caso contrário.} \end{cases} \end{aligned} \tag{4.7}$$

em que i_{prop} e i_{ref} representam o valor obtido na métrica i da proposta e do modelo referência, respectivamente. Por exemplo, R_{RMSE} indica a relação percentual dada por (4.7), considerando a proposta e um modelo de referência na métrica RMSE. Quanto maior o valor de R_i, maior é o ganho percentual da proposta com relação ao método de referência.

4.2.10 Teste Estatístico de Hipóteses

Toda previsão está associada a um erro, porém esse erro deve ser o mínimo possível, pois a previsão serve de guia para tomadas de decisões em diversas áreas das ciências. Ademais, a comparação entre acurácia de preditores é muito importante para quem tem interesse em comparar modelos de previsão concorrentes (DIEBOLD; MARIANO, 1995).

Após uma observação nas métricas de desempenho dos modelos, é inevitável que um conjunto de previsões apareça com mais sucesso do que outro, mesmo que apenas por uma pequena quantia. Surge naturalmente a questão de quão provável é que esse resultado seja devido ao acaso (HARVEY; LEYBOURNE; NEWBOLD, 1997). Além disso, existe um consenso de que qualquer tentativa de justificar a superioridade comparativa das previsões de um

determinado modelo é incompleta e inadmissível, se nenhuma consideração tiver sido dada à significância estatística associada à comparação (HASSANI; SILVA, 2015).

Em (HARVEY; LEYBOURNE; NEWBOLD, 1998), é discorrido que previsões podem ser comparadas informalmente ou formalmente de diferentes formas. Porém, é desejável dispor de procedimentos formais de teste de hipóteses para a análise de previsões concorrentes. Há diferentes formas de aplicar testes de hipóteses para comparar previsores. Em (DIEBOLD; MARIANO, 1995), Diebold e Mariano propõem um teste que é diretamente baseado no desempenho preditivo, uma função de perda é aplicada nos erros dos dois modelos concorrentes, e assume-se que esses erros seguem uma distribuição normal. De acordo com (FORSYTH, 2018), por meio do teorema do limite central, pode-se assumir que a distribuição dos dados é normal, se a quantidade de observações na amostra que vai ser aplicado o teste de hipóteses é maior que 30.

De (DIEBOLD; MARIANO, 1995), consideram-se dois previsores $\{\hat{y}_{1t}\}_{t=1}^{n}$ e $\{\hat{y}_{2t}\}_{t=1}^{n}$ da série temporal $\{y_t\}_{t=1}^{n}$. Sejam $\{e_{1t}\}_{t=1}^{n}$ e $\{e_{2t}\}_{t=1}^{n}$ os respectivos erros de previsão e a função de perda é uma função direta dos erros de previsão $g(y_t, \hat{y}_{1t}) = g(e_{1t})$. A hipótese nula de igualdade na acurácia de previsão para dois preditores é: $E_s[g(e_{1t})] = E_s[g(e_{2t})]$, ou $E_s[d_t] = 0$, em que $d_t = g(e_{1t}) - g(e_{2t})$, e E_s é o valor esperado. Portanto, a hipótese nula H_0 afirma que os dois modelos de previsão têm acurácia igual.

Para H_0 ser rejeitada, o valor-p do teste deve ser menor que o nível de significância estatística α. No teste de hipóteses DM, $\alpha = 0,05$ significa que tolera-se atribuir ao acaso medidas de diferença, cuja probabilidade de ocorrência seja ao menos 5%. Tipicamente, são utilizados valores de α iguais a $0,10$, $0,05$ ou, até mesmo, $0,01$ para estudos mais rigorosos.

Os autores em (HARVEY; LEYBOURNE; NEWBOLD, 1998) descrevem que a abordagem de Diebold e Mariano é bem intuitiva e recomendam o teste nos casos em que a previsão é de um passo à frente e a função perda é quadrática. O teste Diebold-Mariano (DM) é altamente citado na literatura e popular para comparar preditores (HASSANI; SILVA, 2015). Além disso, em (HARVEY; LEYBOURNE; NEWBOLD, 1997), é sugerida uma alteração na estatística do teste DM com aplicabilidade em previsões além de um passo e a outras funções de perda além da quadrática. A maioria dos testes de hipóteses baseados em Diebold e Mariano, a função de perda é geralmente quadrática ou absoluta (MCCRACKEN; WEST, 2002).

4.3 Resumo do Capítulo

Neste capítulo, foram discutidas a metodologia e as avaliações desta pesquisa. Foram apresentadas as séries temporais de velocidade do vento e todos os métodos que foram implementados. E, por fim, foram apontadas as métricas de avaliação e o teste estatístico de hipóteses que foram empregados para avaliar os resultados. No capítulo a seguir, serão

apresentados os resultados do trabalho.

5 RESULTADOS

Este capítulo está dividido em quatro seções. As duas primeiras estão separadas de acordo com a granularidade das bases de dados. Primeiramente, serão avaliadas as bases de dados horárias e, posteriormente, as bases com intervalos de tempo de 3h entre instâncias consecutivas. Serão apresentados os resultados no conjunto de teste por meio das medidas de avaliação RMSE, MAE, MAPE e POCID. Ademais, relações percentuais sobre cada medida de avaliação entre a principal proposta (LocLN) e os demais métodos. O teste estatístico de hipóteses de Diebold e Mariano foi aplicado para avaliar se há diferença significativa nos erros das observações entre o LocLN e os outros métodos concorrentes para uma única execução com uma semente. E, finalmente, ilustrações serão exibidas em gráficos para poder observar o comportamento das curvas de previsão dos modelos. A terceira seção contém os resultados após 100 execuções com sementes diferentes para os métodos com o modelo ELM nas cinco bases de dados horárias. Neste caso, foram utilizadas as medidas de avaliação RMSE e POCID_{media}. O teste estatístico de hipóteses de Diebold e Mariano também foi aplicado para verificar se há diferença significativa nos erros médios sobre as 100 execuções entre o LocLN e os outros métodos. E, finalmente, a última seção envolve uma discussão sobre os resultados.

5.1 Base de Dados com Intervalos de Tempo de 1h

Da base de dados 1 até a base de dados 4, as séries têm 2.208 observações. E a base de dados 5 possui 2.136 observações. Essas primeiras cinco bases possuem intervalos de tempo de 1h entre instâncias consecutivas.

5.1.1 Medidas de Avaliação em Relação à Acurácia

A Tabela 5 mostra os resultados das quatro medidas de avaliação no conjunto de teste para a base de dados 1. A primeira coluna identifica a abordagem, tais quais, individual, MHP, MHS, propostas de MHP com mapeamento local, e, por último, propostas de MHP com MHS e mapeamento local. A segunda coluna descreve os métodos desenvolvidos. Por fim, os resultados das quatro medidas de avaliação, RMSE, MAE, MAPE e POCID são apresentados para cada método. O melhor resultado em cada métrica está destacado em negrito. Em relação aos modelos individuais, o ARIMA se destacou por apresentar os menores erros no MAE e no MAPE. Para os MHPs, o $BAGG_{ELM}$ obteve os melhores resultados no RMSE, MAE e MAPE. Em relação aos MHSs, LC_{AM} obteve os menores erros no RMSE, MAE e MAPE. Ao analisar as propostas, o LocLN superou todos os métodos no RMSE, MAE e POCID. Para essa base de dados, a combinação da previsão

Tabela 5 – Medidas de avaliação no conjunto de teste para a base de dados 1.

Abordagem	**Método**	**RMSE(m/s)**	**MAE(m/s)**	**MAPE(%)**	**POCID(%)**
Individual	Persistente	0,8318	0,6305	10,35	47,82
	ARIMA (TORRES et al., 2005)	0,8142	0,6141	10,04	48,81
	ELM (SAAVEDRA-MORENO et al., 2013)	0,8126	0,6227	10,15	48,71
	LSTM (LIU; LIN; FENG, 2021)	0,9105	0,7014	10,85	48,61
	GRU (LIU; LIN; FENG, 2021)	0,8812	0,6796	10,65	49,01
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	0,8273	0,6292	10,30	48,21
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	0,8041	0,6154	10,05	47,82
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	0,7607	0,5807	**9,58**	47,82
	LC_{AS} (de MATTOS NETO et al., 2020)	0,7684	0,5929	9,76	48,21
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	0,7665	0,5903	9,79	47,42
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	0,7703	0,5965	9,85	48,81
	LC_{AE}	0,7734	0,5905	9,71	49,40
	$NC - E_{AE}$ (ALMEIDA, 2024)	0,7738	0,5883	9,68	48,81
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,7949	0,6016	9,89	50,60
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,7984	0,6162	10,25	49,01
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,8167	0,6332	10,40	49,80
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,8065	0,6248	10,39	48,21
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	0,7706	0,5885	9,71	50,20
	LocLN	**0,7549**	**0,5744**	9,65	**50,83**

da série temporal com a previsão dos resíduos foi realizada de forma não linear para o método LocLN.

Tabela 6 – Medidas de avaliação no conjunto de teste para a base de dados 2.

Abordagem	**Método**	**RMSE(m/s)**	**MAE(m/s)**	**MAPE(%)**	**POCID(%)**
Individual	Persistente	1,3950	1,0288	12,30	56,55
	ARIMA (TORRES et al., 2005)	1,4085	1,0249	12,22	55,16
	ELM (SAAVEDRA-MORENO et al., 2013)	1,3765	1,0227	12,78	54,47
	LSTM (LIU; LIN; FENG, 2021)	1,3771	1,0247	12,78	55,95
	GRU (LIU; LIN; FENG, 2021)	1,3929	1,0426	13,11	55,36
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,4032	1,0253	12,14	**56,94**
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,3641	1,0208	12,69	54,96
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,3642	0,9905	12,06	55,36
	LC_{AS} (de MATTOS NETO et al., 2020)	1,3414	0,9695	11,62	55,75
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,3519	0,9946	12,32	55,36
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,3230	0,9801	12,11	55,95
	LC_{AE}	1,3449	0,9901	12,03	55,75
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,3296	0,9804	11,94	55,36
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3461	0,9884	11,85	**56,94**
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3268	0,9711	11,87	55,16
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3298	0,9764	12,00	56,35
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3276	0,9811	12,08	56,15
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,3008	0,9515	11,66	55,95
	LocLN	**1,2976**	**0,9450**	**11,59**	56,04

Os resultados das métricas de avaliação para a base de dados 2 são apresentados na Tabela 6. Esta tabela, bem como as Tabelas 7, 8 e 9, têm um layout de coluna igual ao da Tabela 5. O melhor modelo individual foi o ELM, que obteve os menores erros em RMSE e MAE. Entre os MHPs, o $BAGG_{ELM}$ obteve os menores erros para o RMSE e MAE. Já para MHSs, destacaram-se LC_{AS} e $NC - S_{AS}$, com o primeiro obtendo os melhores resultados no MAE e MAPE, e o último no RMSE e POCID. Ao considerar as propostas, o LocLN superou todos os métodos no RMSE, MAE e MAPE. Para a base de dados 2, a combinação da previsão da série temporal com a previsão de seus resíduos foi realizada de forma não linear para o método LocLN.

A Tabela 7 exibe as medidas de avaliação para o conjunto de teste obtidas na base de dados 3. O ARIMA foi o modelo individual que obteve os menores erros no RMSE e

Tabela 7 – Medidas de avaliação no conjunto de teste para a base de dados 3.

Abordagem	Método	RMSE(m/s)	MAE(m/s)	MAPE(%)	POCID(%)
Individual	Persistente	1,0065	0,7927	4,09	52,18
	ARIMA (TORRES et al., 2005)	1,0055	0,7939	4,08	52,38
	ELM (SAAVEDRA-MORENO et al., 2013)	1,1564	0,9231	4,70	51,79
	LSTM (LIU; LIN; FENG, 2021)	1,1147	0,8838	4,45	51,39
	GRU (LIU; LIN; FENG, 2021)	1,1706	0,9328	4,72	53,17
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,0165	0,8026	4,12	52,18
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,0592	0,8394	4,26	52,18
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,0093	0,8006	4,14	**53,57**
	LC_{AS} (de MATTOS NETO et al., 2020)	1,0353	0,8154	4,22	**53,57**
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,0741	0,8543	4,36	53,17
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,1058	0,8745	4,49	52,98
	LC_{AE}	1,0489	0,8383	4,34	52,18
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,0856	0,8692	4,44	52,58
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	**1,0009**	**0,7889**	**4,06**	51,98
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,0534	0,8351	4,26	51,79
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1059	0,8839	4,50	51,39
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1269	0,9013	4,60	52,38
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,0940	0,8715	4,44	52,18
	LocLN	1,0174	0,8057	4,17	52,08

MAPE. Em relação aos MHPs, o $BAGG_{ARIMA}$ obteve os melhores resultados no RMSE, MAE e MAPE. O LC_{AM} se destacou entre os MHSs, obtendo os melhores resultados em todas as quatro métricas de avaliação. A proposta LocLN foi superada por apenas quatro métodos, o modelo de persistência, ARIMA, o $LocPart_{ARIMA}$ e o LC_{AM}. O desempenho preditivo do modelo de persistência destaca-se, evidenciando que esta série temporal apresenta características marcantes de um processo de passeio aleatório, conhecido por sua forte dependência dos valores observados no passado mais recente. Nesta base de dados, a combinação da previsão da série temporal com a previsão residual foi realizada linearmente para o método LocLN.

Tabela 8 – Medidas de avaliação no conjunto de teste para a base de dados 4.

Abordagem	Método	RMSE(m/s)	MAE(m/s)	MAPE(%)	POCID(%)
Individual	Persistente	1,1938	0,8906	8,67	55,56
	ARIMA (TORRES et al., 2005)	1,2316	0,9346	9,19	56,15
	ELM (SAAVEDRA-MORENO et al., 2013)	1,1538	0,8783	9,01	55,75
	LSTM (LIU; LIN; FENG, 2021)	1,1653	0,8952	9,13	56,55
	GRU (LIU; LIN; FENG, 2021)	1,1842	0,9157	9,34	56,75
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,2822	0,9650	9,54	**58,53**
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,1714	0,8862	8,98	55,16
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,1617	0,8664	8,62	57,94
	LC_{AS} (de MATTOS NETO et al., 2020)	1,1671	0,8648	8,62	57,34
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,1671	0,8716	8,88	57,14
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,1659	0,8665	8,75	57,74
	LC_{AE}	1,1891	0,8783	8,74	56,35
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,2442	0,9146	9,74	56,55
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1369	0,8483	**8,31**	55,95
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1388	0,8525	8,59	55,95
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,2368	0,9642	9,95	56,15
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,2028	0,9142	9,20	55,95
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,1851	0,8864	8,84	58,33
	LocLN	**1,1043**	**0,8307**	**8,31**	56,67

A tabela 8 exibe as medidas de avaliação para a base de dados 4. O ELM foi o modelo individual com os melhores resultados no RMSE, MAE e MAPE. O $BAGG_{ELM}$ foi o melhor entre os MHPs, com os menores erros no RMSE, MAE e MAPE. Em relação aos MHSs, o LC_{AM} obteve os melhores resultados no RMSE, MAPE e POCID. A proposta

LocLN foi o melhor método no geral, alcançando o melhor desempenho no RMSE, MAE e MAPE. Nesta base de dados, a combinação da previsão da série temporal com a previsão residual foi realizada linearmente para o método LocLN.

Tabela 9 – Medidas de avaliação no conjunto de teste para a base de dados 5.

Abordagem	**Método**	**RMSE(m/s)**	**MAE(m/s)**	**MAPE(%)**	**POCID(%)**
Individual	Persistente	1,1453	0,8720	10,05	54,94
	ARIMA (TORRES et al., 2005)	1,1431	0,8690	9,99	54,32
	ELM (SAAVEDRA-MORENO et al., 2013)	1,0946	0,8539	9,92	54,32
	LSTM (LIU; LIN; FENG, 2021)	1,1184	0,9048	11,04	55,35
	GRU (LIU; LIN; FENG, 2021)	1,1341	0,9221	11,33	**56,17**
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,1671	0,8904	10,22	55,14
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,1058	0,8626	10,02	54,94
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,0759	0,8367	9,72	54,73
	LC_{AS} (de MATTOS NETO et al., 2020)	**1,0662**	**0,8318**	9,68	54,94
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,0671	0,8434	9,94	55,97
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,0817	0,8372	**9,59**	55,14
	LC_{AE}	1,0847	0,8464	9,86	53,70
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,0889	0,8548	9,93	54,12
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1304	0,8584	9,88	55,56
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1034	0,8673	10,21	54,73
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1292	0,9058	10,85	54,53
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1126	0,8802	10,36	54,32
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,0901	0,8571	9,95	53,29
	LocLN	1,0804	0,8426	9,92	53,46

Os resultados para a base de dados 5 são ilustrados na Tabela 9. O ELM foi o modelo individual com os menores erros no RMSE, MAE e MAPE. Considerando os MHPs, o $BAGG_{ELM}$ obteve os melhores resultados no MAE, MAPE e POCID. Em relação aos MHSs, o LC_{AS} obteve os melhores resultados no RMSE e MAE. Entre as propostas, verificou-se que o LocLN conseguiu melhores resultados que o LocLinNonPR em todas as quatro métricas. Para a base de dados 5, a combinação da previsão da série temporal com a previsão residual foi realizada linearmente para o método LocLN.

5.1.2 Métrica de Relação Percentual e Teste Estatístico de Hipótese

Tabela 10 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 1.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	9,24	8,90	6,31	6,31	3,50E-06	+
	ARIMA (TORRES et al., 2005)	7,28	6,45	3,42	4,15	7,14E-05	+
	ELM (SAAVEDRA-MORENO et al., 2013)	7,10	7,74	4,46	4,36	2,12E-04	+
	LSTM (LIU; LIN; FENG, 2021)	**17,09**	**18,10**	**10,59**	4,57	7,32E-09	+
	GRU (LIU; LIN; FENG, 2021)	14,33	15,47	8,96	3,72	2,30E-07	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	8,74	8,70	5,83	5,43	1,47E-05	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	6,11	6,65	3,50	6,31	1,28E-03	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	0,76	1,08	-1,20	6,31	0,6898	=
	LC_{AS} (de MATTOS NETO et al., 2020)	1,75	3,12	0,60	5,43	0,8075	=
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,51	2,69	0,98	**7,20**	0,8653	=
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	2,00	3,71	1,50	4,15	0,7442	=
	LC_{AE}	2,38	2,72	0,14	2,89	0,2623	=
	$NC - E_{AE}$ (ALMEIDA, 2024)	2,43	2,36	-0,24	4,15	0,2345	=
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	5,03	4,51	1,96	0,47	5,77E-04	+
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	5,45	6,78	5,38	3,72	2,46E-03	+
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	7,56	9,28	6,74	2,07	5,51E-04	+
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	6,39	8,06	6,64	5,43	1,31E-03	+
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	2,04	2,38	0,17	1,26	0,4559	=

A avaliação também incluiu a relação percentual por métrica e o teste de hipóteses DM nos erros quadráticos entre o LocLN e os outros métodos. Visto que o conjunto de teste possui 25% dos dados das séries, então no caso da menor base horária (base de dados 5) são 534 observações. Assim, por meio do teorema do limite central, assumiu-se que os erros dos métodos no conjunto teste seguem uma distribuição normal. A tabela 10 apresenta os resultados para a base de dados 1. As duas primeiras colunas mostram o tipo de abordagem e o método de previsão. As quatro colunas seguintes exibem a relação percentual por RMSE, MAE, MAPE e POCID. Os maiores ganhos percentuais são destacados em negrito. Finalmente, as duas últimas colunas referem-se ao p-valor e à interpretação do teste de hipóteses DM. As tabelas 11, 12, 13 e 14 seguem a mesma estrutura de layout de colunas da Tabela 10. Observa-se que o método LocLN obteve ganhos percentuais de 17,09% no R_{RMSE}, 18,10% no R_{MAE} e 10,59% no R_{MAPE} em comparação com o modelo LSTM individual, e 7,20% no R_{POCID} em comparação com $NC - S_{AM}$. Em relação ao teste de hipóteses estatístico DM, a hipótese nula de igualdade nos erros quadráticos foi rejeitada para todos os modelos individuais, todos os MHPs, incluindo o LocPart, que possui mapeamento local. A superioridade estatística do LocLN sobre esses métodos foi indicada. Isso corresponde a 61% dos 18 métodos concorrentes. Nos 39% restantes dos casos, o teste DM indicou igualdade no erros quadráticos.

Tabela 11 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 2.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	6,98	8,14	5,80	-0,89	3,89E-05	+
	ARIMA (TORRES et al., 2005)	**7,87**	7,80	5,15	1,60	8,74E-06	+
	ELM (SAAVEDRA-MORENO et al., 2013)	5,73	7,60	9,32	**2,88**	4,10E-05	+
	LSTM (LIU; LIN; FENG, 2021)	5,77	7,78	9,30	0,16	6,06E-06	+
	GRU (LIU; LIN; FENG, 2021)	6,84	**9,36**	**11,61**	1,24	3,56E-07	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	7,53	7,84	4,53	-1,59	1,82E-05	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	4,88	7,43	8,65	1,97	1,50E-04	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	4,89	4,59	3,87	1,24	5,18E-03	+
	LC_{AS} (de MATTOS NETO et al., 2020)	3,27	2,53	0,26	0,52	8,73E-03	+
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	4,02	4,99	5,94	1,24	5,26E-03	+
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,92	3,58	4,32	0,16	7,32E-02	=
	LC_{AE}	3,52	4,56	3,63	0,52	1,52E-03	+
	$NC - E_{AE}$ (ALMEIDA, 2024)	2,41	3,61	2,97	1,24	4,11E-04	+
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	3,60	4,40	2,20	-1,59	1,47E-03	+
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,20	2,69	2,38	1,60	9,04E-03	+
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,42	3,22	3,44	-0,55	2,67E-02	+
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,26	3,69	4,05	-0,19	4,37E-02	+
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	0,25	0,69	0,56	0,16	0,4407	=

A tabela 11 mostra os resultados da relação percentual por métrica e do teste de hipóteses DM para o base de dados 2. Observa-se que o LocLN obteve ganhos percentuais de 9,36% no R_{MAE} e 11,61% no R_{MAPE} em comparação com o modelo GRU individual, 7,87% no R_{RMSE} em comparação com ARIMA e 2,88% no R_{POCID} em comparação com o ELM. O teste DM rejeitou a hipótese nula de igualdade nos erros quadráticos, mostrando que LocLN tem desempenho preditivo superior em 88% entre os 18 métodos concorrentes. As exceções foram os métodos LocNonLinPR e $NC - S_{AS}$, que demonstraram igualdade nos erros.

Tabela 12 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 3.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	-1,08	-1,64	-1,88	-0,19	0,9534	=
	ARIMA (TORRES et al., 2005)	-1,18	-1,49	-2,17	-0,57	0,9082	=
	ELM (SAAVEDRA-MORENO et al., 2013)	8,73	8,83	6,22	**1,35**	1,45E-05	+
	LSTM (LIU; LIN; FENG, 2021)	**13,09**	**13,62**	**11,68**	-2,05	6,48E-08	+
	GRU (LIU; LIN; FENG, 2021)	12,03	12,72	11,34	0,57	2,45E-07	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	-0,08	-0,39	-1,10	-0,19	0,3366	=
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	3,95	4,00	2,19	-0,19	5,13E-03	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	-0,80	-0,65	-0,83	-2,78	0,7997	=
	LC_{AS} (de MATTOS NETO et al., 2020)	1,73	1,18	1,17	-2,78	3,17E-02	+
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	5,29	5,68	4,25	-2,05	1,63E-03	+
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	8,00	7,86	7,13	-1,69	1,45E-05	+
	LC_{AE}	3,00	3,89	4,02	-0,19	7,84E-04	+
	$NC - E_{AE}$ (ALMEIDA, 2024)	6,29	7,30	6,08	-0,94	6,69E-05	+
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	-1,64	-2,14	-2,65	0,19	0,5351	=
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	3,42	3,52	2,11	0,57	1,15E-02	+
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	8,00	8,84	7,35	1,35	5,89E-05	+
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	9,72	10,60	9,43	-0,57	9,30E-06	+
Proposta de MHP com MHS local com MHSP	LocLinNonPR (ALMEIDA, 2024)	7,01	7,54	6,05	-0,19	3,07E-05	+

Os resultados da relação percentual e do teste de hipóteses DM para a base de dados 3 são apresentados na Tabela 12. O LocLN obteve ganhos percentuais de 13,09% no R_{RMSE}, 13,62% no R_{MAE} e 11,68% no R_{MAPE} em comparação com o LSTM. O teste de hipóteses DM forneceu evidências estatísticas de que o LocLN tem erros quadráticos menores em 72% dos 18 métodos concorrentes. Nos 28% dos casos restantes, o teste DM indicou igualdade.

Tabela 13 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 4.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	7,50	6,72	4,18	2,00	1,04E-04	+
	ARIMA (TORRES et al., 2005)	10,34	11,11	9,59	0,92	1,95E-08	+
	ELM (SAAVEDRA-MORENO et al., 2013)	4,29	5,42	7,78	1,64	4,82E-03	+
	LSTM (LIU; LIN; FENG, 2021)	5,24	7,20	8,93	0,21	7,81E-04	+
	GRU (LIU; LIN; FENG, 2021)	6,75	9,28	10,99	-0,14	6,76E-05	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	**13,88**	**13,92**	12,92	-3,19	8,55E-09	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	5,73	6,26	7,44	**2,73**	1,97E-05	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	4,95	4,12	3,64	-2,19	3,33E-02	+
	LC_{AS} (de MATTOS NETO et al., 2020)	5,39	3,94	3,58	-1,18	4,87E-03	+
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	5,38	4,69	6,37	-0,83	1,56E-02	+
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	5,29	4,13	5,02	-1,86	7,69E-03	+
	LC_{AE}	7,13	5,42	4,87	0,56	8,64E-05	+
	$NC - E_{AE}$ (ALMEIDA, 2024)	11,24	9,18	14,69	0,21	1,44E-03	+
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,87	2,08	0,00	1,28	1,77E-04	+
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	3,03	2,56	3,20	1,28	3,80E-02	+
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	10,72	13,84	**16,52**	0,92	6,54E-10	+
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	8,19	9,14	9,70	1,28	8,37E-08	+
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	6,82	6,28	5,95	-2,86	2,40E-04	+

A Tabela 13 apresenta os resultados das relações percentuais e do teste DM para a base de dados 4. Observa-se que o LocLN obteve ganhos percentuais de 13,88% no R_{RMSE} e 13,92% no R_{MAE} em comparação com $BAGG_{ARIMA}$, e 16,52% no R_{MAPE} em comparação com $LocPart_{LSTM}$. O teste de hipóteses DM indicou que o LocLN tem desempenho preditivo superior nos erros quadráticos em 100% das comparações, considerando os 18 métodos concorrentes.

Tabela 14 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 5.

Abordagem	Método	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	p-valor	Teste DM
Individual	Persistência	5,66	3,37	1,36	-2,68	3,95E-03	+
	ARIMA (TORRES et al., 2005)	5,48	3,04	0,76	-1,58	1,72E-02	+
	ELM (SAAVEDRA-MORENO et al., 2013)	1,29	1,32	0,06	-1,58	0,8232	=
	LSTM (LIU; LIN; FENG, 2021)	3,39	6,88	10,14	-3,41	0,0665	=
	GRU (LIU; LIN; FENG, 2021)	4,73	**8,62**	**12,44**	-4,82	1,58E-02	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	**7,43**	5,37	2,92	-3,05	1,01E-03	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	2,30	2,32	1,06	-2,68	0,3721	=
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	-0,42	-0,70	-2,08	-2,32	0,7013	=
	LC_{AS} (de MATTOS NETO et al., 2020)	-1,34	-1,30	-2,46	-2,68	0,1810	=
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	-1,25	0,09	0,18	-4,47	0,2546	=
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	0,11	-0,64	-3,37	-3,05	0,2906	=
	LC_{AE}	0,39	0,45	-0,54	-0,45	0,8956	=
	$NC - E_{AE}$ (ALMEIDA, 2024)	0,78	1,43	0,16	-1,20	0,8124	=
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	4,42	1,84	-0,36	-3,77	3,16E-02	+
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,08	2,85	2,90	-2,32	0,3739	=
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	4,32	6,98	8,63	-1,95	0,0796	=
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,89	4,28	4,28	-1,58	0,2402	=
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	0,89	1,69	0,36	**0,32**	0,8910	=

Os resultados da relação percentual e do teste de hipóteses DM para a base de dados 5 são apresentados na Tabela 14. Pode-se ver que o LocLN obteve ganhos percentuais de 8,62% no R_{MAE} e 12,44% no R_{MAPE} em comparação com GRU, e 7,43% no R_{RMSE} em comparação com $BAGG_{ARIMA}$. A superioridade do LocLN foi observada principalmente entre modelos individuais e MHPs. Os resultados do teste de hipóteses DM indicaram que o LocLN tem erros quadráticos menores em 27% dos casos entre os 18 métodos concorrentes. Nos 73% dos casos restantes, o teste DM indicou igualdade nos erros quadráticos.

5.1.3 Gráficos de Previsão

Foram selecionados alguns métodos juntamente com o LocLN para comparar visualmente as previsões das séries temporais no conjunto de teste. Primeiro, o método $LocPart_{ARIMA}$, que faz parte do Estágio I do LocLN. Além disso, um modelo individual foi escolhido com base nos valores da relação percentual R_i. Para a base de dados 1, o LSTM foi selecionado porque foi o modelo individual em que LocLN obteve o maior ganho percentual. A Figura 34 ilustra as previsões um passo à frente considerando 65 observações sequenciais no conjunto de teste para a base de dados 1. Verifica-se a superioridade do LocLN, principalmente em relação ao LSTM. A diferença de erros entre $LocPart_{ARIMA}$ e LocLN é mais sutil, no entanto, é notável que entre as observações 10 e 30, $LocPart_{ARIMA}$ mostra erros aparentes, principalmente nos vales da série temporal.

O modelo individual ARIMA foi selecionado para a avaliação gráfica da base de dados 2. A Figura 35 mostra as previsões de um passo à frente considerando 65 observações sequenciais do conjunto de teste para a base de dados 2. A superioridade do LocLN sobre o ARIMA e $LocPart_{ARIMA}$ é notável nos picos e vales da série temporal.

A Figura 36 mostra as previsões de um passo à frente considerando 65 observações sequenciais do conjunto de teste para a base de dados 3. O modelo individual seleci-

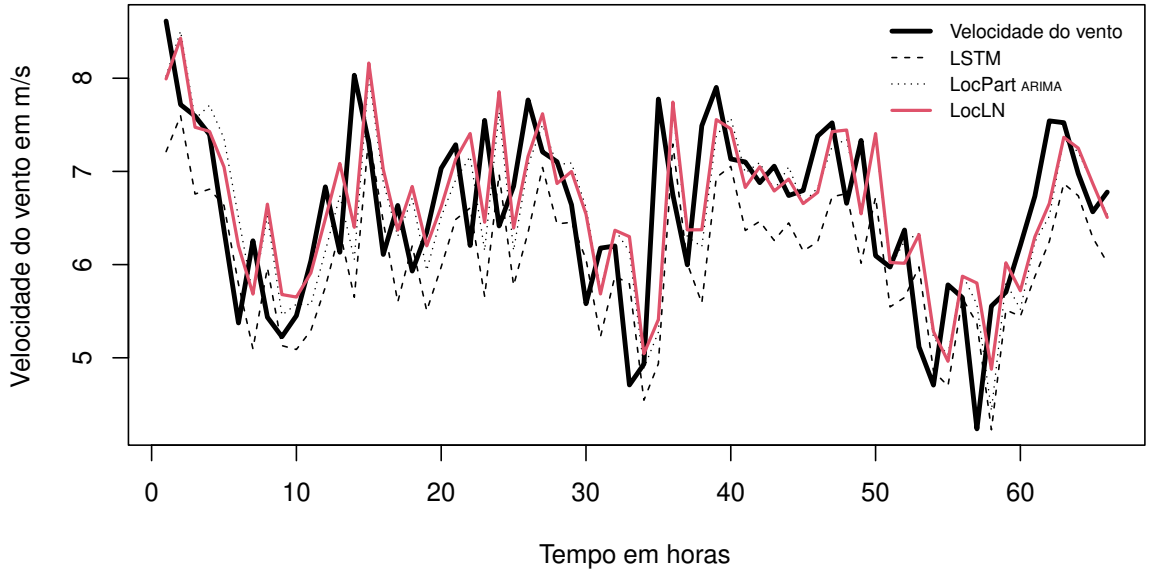

Figura 34 – Resultados da previsão de um passo à frente para LocLN em comparação com o LSTM individual e o $LocPart_{ARIMA}$ para a base de dados 1, considerando uma sequência de 65 observações do conjunto de teste.

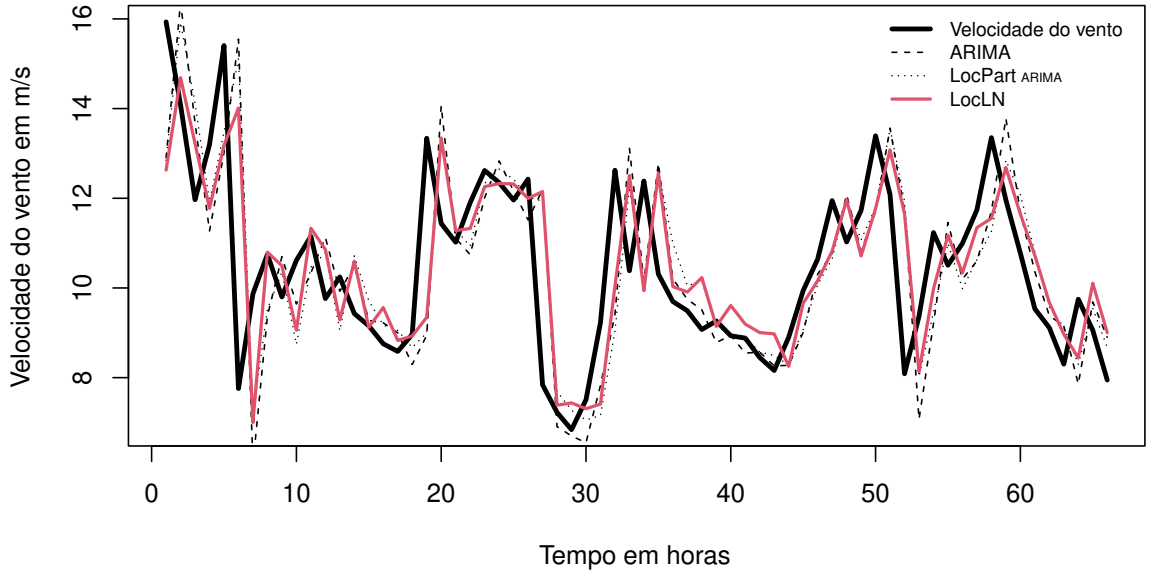

Figura 35 – Resultados da previsão de um passo à frente para LocLN em comparação com o ARIMA individual e o $LocPart_{ARIMA}$ para a base de dados 2, considerando uma sequência de 65 observações do conjunto de teste.

onado foi LSTM. A superioridade do LocLN sobre o LSTM é evidente. Em contraste,

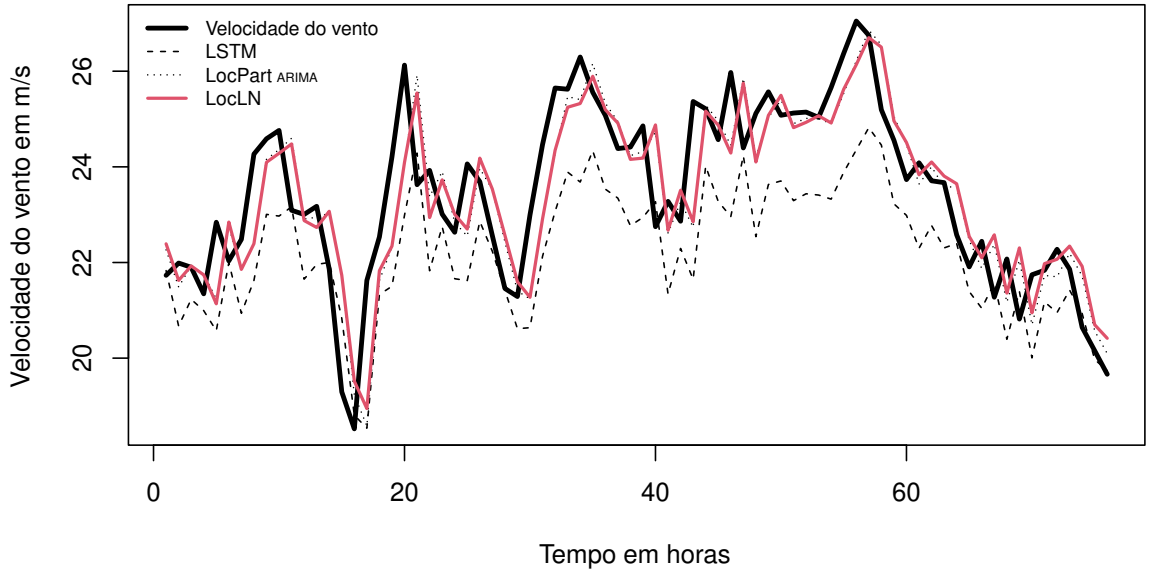

Figura 36 – Resultados da previsão de um passo à frente para LocLN em comparação com o LSTM individual e o $LocPart_{ARIMA}$ para a base de dados 3, considerando uma sequência de 65 observações do conjunto de teste.

para $LocPart_{ARIMA}$, as curvas de previsão são muito semelhantes, dificultando discernir diferenças nos erros de previsão.

O modelo individual ARIMA foi selecionado para a avaliação gráfica para a base de dados 4, e a Figura 37 mostra as previsões de um passo à frente considerando 65 observações sequenciais do conjunto de teste para a base de dados 4. LocLN supera ARIMA, principalmente nos vales da série temporal. E, comparado a $LocPart_{ARIMA}$, não há diferenças substanciais.

E, finalmente, a Figura 38 mostra as previsões de um passo à frente considerando 65 observações sequenciais do conjunto de teste para a base de dados 5. O modelo individual selecionado foi GRU. Observa-se que os erros do GRU são maiores do que os erros do LocLN ao longo da maior parte da série temporal mostrada na Figura 38. $LocPart_{ARIMA}$ exibe erros maiores do que o método LocLN, particularmente nos vales da série temporal.

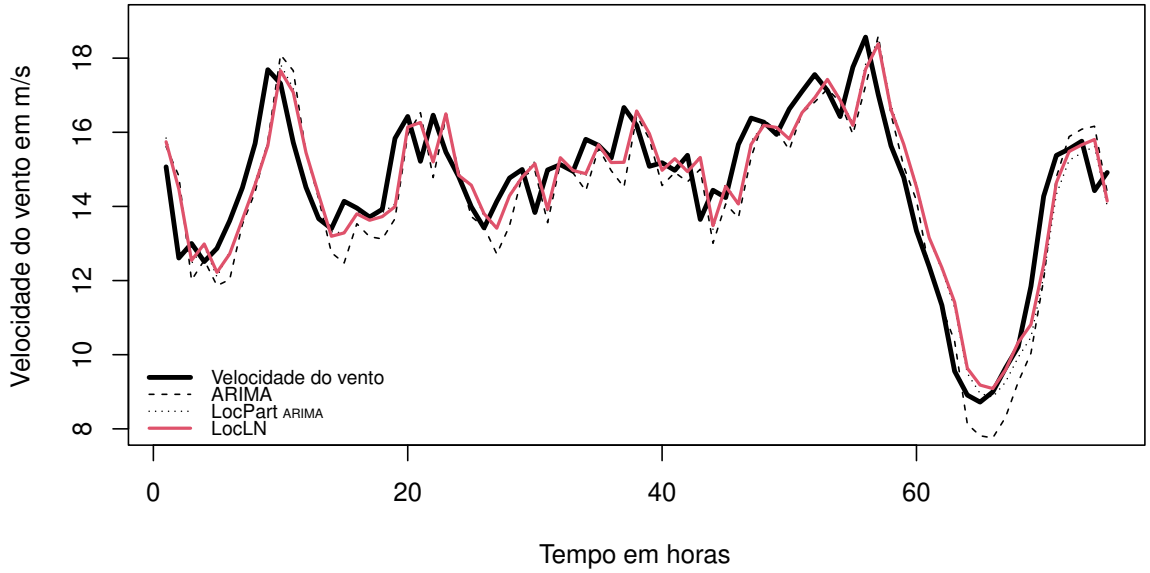

Figura 37 – Resultados da previsão de um passo à frente para LocLN em comparação com o LSTM individual e o $LocPart_{ARIMA}$ para a base de dados 4, considerando uma sequência de 65 observações do conjunto de teste.

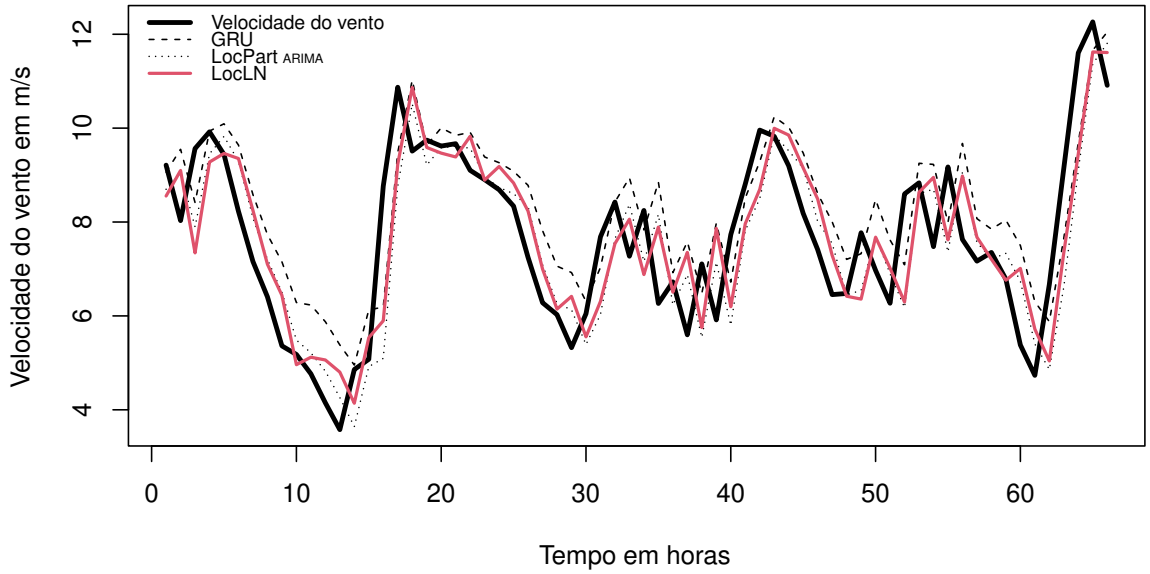

Figura 38 – Resultados da previsão de um passo à frente para LocLN em comparação com o GRU individual e o $LocPart_{ARIMA}$ para a base de dados 5, considerando uma sequência de 65 observações do conjunto de teste.

5.2 Base de Dados com Intervalos de Tempo de 3h

Da base de dados 6 até a base de dados 9, as séries têm 736 observações. E a base de dados 10 possui 712 observações. Essas cinco bases possuem intervalos de tempo de 3h entre instâncias consecutivas.

5.2.1 Medidas de Avaliação em Relação à Acurácia

Tabela 15 – Medidas de avaliação no conjunto de teste para a base de dados 6.

Abordagem	Método	RMSE(m/s)	MAE(m/s)	MAPE(%)	POCID(%)
Individual	Persistente	0,9892	0,7900	13,62	42,65
	ARIMA (TORRES et al., 2005)	0,9087	0,7297	12,17	45,59
	ELM (SAAVEDRA-MORENO et al., 2013)	0,8421	0,6879	12,02	47,06
	LSTM (LIU; LIN; FENG, 2021)	0,9080	0,7378	12,59	47,06
	GRU (LIU; LIN; FENG, 2021)	0,9346	0,7571	12,82	47,06
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	0,9286	0,7404	12,75	42,65
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	0,8586	0,7022	12,13	38,24
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	0,7834	0,6390	10,90	50,74
	LC_{AS} (de MATTOS NETO et al., 2020)	0,8361	0,6698	11,39	50,00
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	0,7873	0,6457	11,03	50,74
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	0,8605	0,6896	11,65	**54,41**
	LC_{AE}	0,8304	0,6753	11,56	49,26
	$NC - E_{AE}$ (ALMEIDA, 2024)	0,8393	0,6780	11,67	48,53
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,8490	0,6867	11,71	48,53
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,9089	0,7411	12,89	42,65
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,9327	0,7593	13,36	46,32
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,9146	0,7454	12,88	44,12
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	**0,7752**	**0,6228**	**10,79**	48,53
	LocLN	0,8149	0,6719	11,93	47,32

Os resultados das métricas de avaliação para a base de dados 6 são apresentados na Tabela 15. Esta tabela, bem como as Tabelas 16, 17, 18 e 19, têm um layout de coluna igual ao da Tabela 5. O melhor resultado em cada métrica está destacado em negrito. O melhor modelo individual foi o ELM, que obteve os menores erros em RMSE, MAE e MAPE. O melhor MHP foi o $BAGG_{ELM}$, com os melhores resultados para o RMSE, MAE e MAPE. Entre os MHSs o LC_{AM} obteve os menores erros no RMSE, MAE e MAPE. Ao analisar as propostas, o LocNonLinPR superou todos os métodos no RMSE, MAE e MAPE. A abordagem LocLN superou os modelos individuais e MHPs em todas as métricas. Para a base de dados 6, a combinação da previsão da série temporal com a previsão de seus resíduos foi realizada de forma não linear para o método LocLN.

A Tabela 16 exibe as medidas de avaliação para o conjunto de teste obtidas na base de dados 7. ARIMA foi o modelo individual que obteve os melhores resultados em todas as métricas. Em relação aos MHPs, o $BAGG_{ELM}$ obteve os melhores resultados em todas as métricas. O $NC - S_{AM}$ se destacou entre os MHSs, obtendo os melhores resultados no RMSE, MAE e POCID. Entre as propostas, o LocNonLinPR apresentou desempenho preditivo superior a todos os métodos no RMSE e MAE. O LocLN superou a maioria dos modelos no RMSE e MAE, a exceção foi o LocNonLinPR. Nesta base de dados, a combinação da previsão da série temporal com a previsão residual foi realizada não linearmente para o método LocLN.

Tabela 16 – Medidas de avaliação no conjunto de teste para a base de dados 7.

Abordagem	Método	RMSE(m/s)	MAE(m/s)	MAPE(%)	POCID(%)
Individual	Persistente	1,9278	1,5410	18,76	59,56
	ARIMA (TORRES et al., 2005)	1,7766	1,3821	16,70	65,44
	ELM (SAAVEDRA-MORENO et al., 2013)	1,8032	1,4260	18,82	64,71
	LSTM (LIU; LIN; FENG, 2021)	1,9042	1,5650	21,03	55,15
	GRU (LIU; LIN; FENG, 2021)	1,8704	1,5401	20,69	56,62
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,6888	1,3685	16,66	53,68
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,5994	1,2944	16,43	**66,91**
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,6126	1,2445	15,41	64,71
	LC_{AS} (de MATTOS NETO et al., 2020)	1,6742	1,2830	16,05	61,03
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,5850	1,2365	15,83	65,44
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,7787	1,3499	17,03	63,97
	LC_{AE}	1,6490	1,3002	16,11	64,71
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,6450	1,2988	16,22	64,71
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,5421	1,2119	**14,48**	64,71
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,6360	1,3217	16,54	65,44
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,7434	1,4198	17,95	62,50
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,7999	1,4822	19,47	58,82
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	**1,5181**	**1,1935**	14,73	63,97
	LocLN	1,5250	1,2027	14,93	63,39

Tabela 17 – Medidas de avaliação no conjunto de teste para a base de dados 8.

Abordagem	Método	RMSE(m/s)	MAE(m/s)	MAPE(%)	POCID(%)
Individual	Persistente	1,3877	**1,0971**	**5,48**	46,32
	ARIMA (TORRES et al., 2005)	1,5341	1,2284	6,08	47,79
	ELM (SAAVEDRA-MORENO et al., 2013)	1,7429	1,4089	6,74	50,74
	LSTM (LIU; LIN; FENG, 2021)	1,6748	1,3330	6,32	46,32
	GRU (LIU; LIN; FENG, 2021)	1,7446	1,3877	6,55	47,06
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,4438	1,1551	5,74	45,59
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,8693	1,5010	7,08	45,59
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,4453	1,1794	5,85	50,74
	LC_{AS} (de MATTOS NETO et al., 2020)	1,4602	1,2032	6,01	49,26
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,6239	1,3365	6,45	50,74
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,5487	1,2880	6,29	49,26
	LC_{AE}	1,4888	1,2169	5,96	50,00
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,6175	1,3241	6,38	49,26
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	**1,3778**	1,1154	5,58	51,47
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,1609	1,7102	8,03	46,32
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,5313	1,2502	6,13	48,53
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,8807	1,5081	7,09	47,06
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,5375	1,2560	6,10	54,41
	LocLN	1,3939	1,1152	5,68	**55,36**

A tabela 17 exibe as medidas de avaliação para a base de dados 8. O modelo de persistência foi o modelo individual com os melhores resultados no RMSE, MAE e MAPE. O $BAGG_{ARIMA}$ foi o melhor entre os MHPs no RMSE, MAE e MAPE. Em relação aos MHSs, o LC_{AM} obteve os melhores resultados também no RMSE, MAE e MAPE. Ao analisar as propostas, o $LocPart_{ARIMA}$ obteve o melhor RMSE entre todos os modelos. O LocLN alcançou o melhor resultado no POCID entre todos os métodos, e obteve o segundo melhor valor no MAE, a exceção foi o modelo de persistência. A combinação da previsão da série temporal com a previsão residual foi realizada linearmente para o método LocLN. Nesta base de dados, o desempenho preditivo do modelo de persistência se destaca, evidenciando que essa série temporal exibe fortes características de um processo de passeio aleatório.

Os resultados para a base de dados 9 são ilustrados na Tabela 18. O ARIMA foi o modelo individual com os menores erros no RMSE, MAE e MAPE. Considerando os MHPs, o $BAGG_{ELM}$ obteve os melhores resultados no RMSE e no POCID. Em relação aos MHSs,

Tabela 18 – Medidas de avaliação no conjunto de teste para a base de dados 9.

Abordagem	Método	RMSE(m/s)	MAE(m/s)	MAPE(%)	POCID(%)
Individual	Persistente	2,1367	1,5731	16,43	55,88
	ARIMA (TORRES et al., 2005)	1,8054	1,3376	14,07	61,76
	ELM (SAAVEDRA-MORENO et al., 2013)	1,8677	1,3870	14,78	66,91
	LSTM (LIU; LIN; FENG, 2021)	1,9561	1,5094	16,78	60,29
	GRU (LIU; LIN; FENG, 2021)	2,0676	1,6055	18,18	58,82
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,8756	1,4095	14,78	59,56
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,8686	1,4155	15,28	63,97
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,6499	**1,2299**	**13,26**	61,03
	LC_{AS} (de MATTOS NETO et al., 2020)	1,6788	1,2851	13,56	64,71
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,7035	1,2555	13,44	61,03
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,6820	1,2994	13,76	**66,91**
	LC_{AE}	1,7898	1,3294	14,12	63,97
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,8474	1,3817	14,87	59,56
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,8189	1,3774	14,77	65,44
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,9478	1,5211	16,84	63,24
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,0760	1,6313	18,78	54,41
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	2,1492	1,6941	19,74	55,15
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	**1,6334**	1,2336	13,62	63,24
	LocLN	1,8460	1,3942	15,07	61,61

o LC_{AM} obteve os menores erros no RMSE, MAE e MAPE. A proposta LocLinNonPR superou todos os métodos no RMSE. O LocLN conseguiu melhores resultados no RMSE que a maioria dos modelos individuais, MHPs e MHPs com mapeamento local, as exceções foram o ARIMA e o $LocPart_{ARIMA}$. Para a base de dados 9, a combinação da previsão da série temporal com a previsão residual foi realizada linearmente para o método LocLN.

Tabela 19 – Medidas de avaliação no conjunto de teste para a base de dados 10.

Abordagem	Método	RMSE(m/s)	MAE(m/s)	MAPE(%)	POCID(%)
Individual	Persistente	1,5620	1,2265	14,58	45,38
	ARIMA (TORRES et al., 2005)	1,4733	1,1950	14,02	47,69
	ELM (SAAVEDRA-MORENO et al., 2013)	1,2949	1,0168	12,06	56,15
	LSTM (LIU; LIN; FENG, 2021)	1,3704	1,0940	13,51	49,23
	GRU (LIU; LIN; FENG, 2021)	1,3829	1,1033	13,68	47,69
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,4387	1,1476	13,63	46,15
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,3536	1,0864	12,93	53,85
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	1,2219	0,9768	11,67	57,69
	LC_{AS} (de MATTOS NETO et al., 2020)	1,2117	0,9516	11,57	60,00
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	1,2113	0,9608	**11,51**	55,38
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	1,2102	**0,9488**	11,57	58,46
	LC_{AE}	**1,2051**	0,9807	11,72	**60,77**
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,2285	0,9674	11,65	**60,77**
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,2869	1,0364	12,15	54,62
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3388	1,0735	13,04	56,15
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3845	1,1079	13,63	47,69
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3941	1,1164	13,71	47,69
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,2435	0,9927	11,92	56,15
	LocLN	1,3179	1,0678	12,91	52,83

A tabela 19 exibe as medidas de avaliação para a base de dados 10. Em relação aos modelos individuais, o ELM se destacou por apresentar os melhores resultados em todas as medidas de avaliação. Para os MHPs, $BAGG_{ELM}$ obteve os menores erros em todas as métricas. Em relação aos MHSs, LC_{AE} obteve os melhores resultados no RMSE e POCID. A proposta LocNonLinPR superou todos os modelos individuais, MHPs e MHPs com mapeamento local no RMSE, MAE e MAPE. A proposta LocLN superou no RMSE, MAE e MAPE a maioria dos modelos individuais, MHPs e MHPs com mapeamento local, as exceções foram o ELM e o $LocPart_{ARIMA}$. Para essa base de dados, a combinação da

previsão da séries temporal com a previsão dos resíduos foi realizada de forma linear para o método LocLN.

5.2.2 Métrica de Relação Percentual e Teste Estatístico de Hipótese

Tabela 20 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 6.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	**17,63**	**14,94**	**12,40**	10,96	7,92E-04	+
	ARIMA (TORRES et al., 2005)	10,32	7,93	1,98	3,80	0,0934	=
	ELM (SAAVEDRA-MORENO et al., 2013)	3,24	2,32	0,73	0,56	0,1021	=
	LSTM (LIU; LIN; FENG, 2021)	10,25	8,93	5,23	0,56	9,79E-04	+
	GRU (LIU; LIN; FENG, 2021)	12,81	11,26	6,96	0,56	4,21E-04	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	12,25	9,25	6,45	10,96	3,69E-03	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	5,10	4,31	1,68	**23,76**	2,84E-02	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	-4,02	-5,16	-9,40	-6,73	0,6583	=
	LC_{AS} (de MATTOS NETO et al., 2020)	2,54	-0,32	-4,69	-5,36	0,3531	=
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	-3,50	-4,05	-8,16	-6,73	0,8023	=
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	5,31	2,56	-2,37	-13,03	0,1350	=
	LC_{AE}	1,87	0,50	-3,23	-3,94	0,5413	=
	$NC - E_{AE}$ (ALMEIDA, 2024)	2,91	0,90	-2,22	-2,49	0,4295	=
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	4,02	2,16	-1,87	-2,49	0,1059	=
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	10,34	9,34	7,43	10,96	1,67E-03	+
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	12,64	11,51	10,74	2,15	1,37E-04	+
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	10,90	9,86	7,38	7,26	8,82E-04	+
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	-5,12	-7,89	-10,61	-2,49	0,7187	=

Em seguida estão os resultados da relação percentual por métrica e o teste de hipóteses DM nos erros quadráticos entre o LocLN e os outros métodos. Visto que o conjunto de teste possui 25% dos dados das séries, então no caso da menor base com intervalo de 3h (base de dados 10) são 178 observações. Assim, por meio do teorema do limite central, assumiu-se que os erros dos métodos no conjunto teste seguem uma distribuição normal. As tabelas 20, 21, 22, 23 e 24 seguem a mesma estrutura de layout de colunas da Tabela 10. Observa-se que o método LocLN obteve ganhos percentuais de 17,63% no R_{RMSE}, 14,94% no R_{MAE} e 12,40% no R_{MAPE} em comparação com o modelo de persistência, e 23,76% no R_{POCID} em comparação com o $BAGG_{ELM}$. Em relação ao teste de hipóteses estatístico DM, a hipótese nula de igualdade nos erros quadráticos foi rejeitada em 44% dos 18 métodos concorrentes, indicando superioridade para o LocLN. Nos 56% restantes dos casos, o teste DM indicou igualdade nos erros quadráticos.

A tabela 21 mostra os resultados da relação percentual por métrica e do teste de hipóteses DM para a base de dados 7. Observa-se que o LocLN obteve ganhos percentuais de 23,15% no R_{MAE} e 28,99% no R_{MAPE} em comparação com o modelo LSTM individual, 20,90% no R_{RMSE} em comparação com o modelo de persistência e 18,10% no R_{POCID} em comparação com o $BAGG_{ELM}$. O teste DM rejeitou a hipótese nula de igualdade nos erros quadráticos, mostrando que LocLN tem desempenho preditivo superior em 78% dos 18 métodos de comparação. Nos 22% dos casos restantes, o teste DM indicou igualdade.

Os resultados da relação percentual e do teste de hipóteses DM para a base de dados 8 são apresentados na Tabela 22. O LocLN obteve ganhos percentuais de 35,49% no R_{RMSE},

Tabela 21 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 7.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	**20,90**	21,96	20,42	6,44	3,62E-05	+
	ARIMA (TORRES et al., 2005)	14,16	12,98	10,57	-3,13	2,66E-03	+
	ELM (SAAVEDRA-MORENO et al., 2013)	15,43	15,66	20,63	-2,03	3,91E-04	+
	LSTM (LIU; LIN; FENG, 2021)	19,91	**23,15**	**28,99**	14,95	3,92E-05	+
	GRU (LIU; LIN; FENG, 2021)	18,47	21,91	27,81	11,97	6,40E-05	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	9,70	12,12	10,39	**18,10**	2,33E-03	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	4,65	7,09	9,09	-5,26	0,0645	=
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	5,43	3,36	3,12	-2,03	3,98E-02	+
	LC_{AS} (de MATTOS NETO et al., 2020)	8,91	6,26	6,99	3,87	8,99E-03	+
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	3,79	2,73	5,69	-3,13	0,1079	=
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	14,27	10,91	12,30	-0,90	1,64E-03	+
	LC_{AE}	7,52	7,50	7,28	-2,03	2,80E-02	+
	$NC - E_{AE}$ (ALMEIDA, 2024)	7,30	7,40	7,95	-2,03	3,25E-02	+
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,11	0,76	-3,11	-2,03	0,1482	=
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	6,79	9,01	9,71	-3,13	3,09E-02	+
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	12,53	15,29	16,79	1,43	1,04E-03	+
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	15,28	18,86	23,29	7,77	1,67E-04	+
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	-0,45	-0,77	-1,40	-0,90	0,2715	=

Tabela 22 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 8.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	-0,45	-1,65	-3,65	19,50	0,5585	=
	ARIMA (TORRES et al., 2005)	9,14	9,21	6,63	15,82	0,4031	=
	ELM (SAAVEDRA-MORENO et al., 2013)	20,03	20,84	15,66	9,11	4,64E-04	+
	LSTM (LIU; LIN; FENG, 2021)	16,77	16,34	10,09	19,50	2,80E-02	+
	GRU (LIU; LIN; FENG, 2021)	20,10	19,64	13,22	17,63	8,78E-03	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	3,45	3,45	1,06	**21,43**	0,5659	=
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	25,43	25,70	19,81	**21,43**	2,71E-04	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	3,56	5,44	2,86	9,11	0,9748	=
	LC_{AS} (de MATTOS NETO et al., 2020)	4,54	7,31	5,44	12,37	0,9125	=
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	14,16	16,56	11,91	9,11	4,32E-02	+
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	10,00	13,41	9,67	12,37	0,2513	=
	LC_{AE}	6,38	8,36	4,67	10,71	0,7688	=
	$NC - E_{AE}$ (ALMEIDA, 2024)	13,82	15,78	10,95	12,37	0,0773	=
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	-1,17	0,02	-1,78	7,55	0,5009	=
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	**35,49**	**34,79**	**29,29**	19,50	5,63E-06	+
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	8,97	10,80	7,30	14,07	0,0659	=
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	25,88	26,05	19,88	17,63	8,18E-04	+
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	9,34	11,21	6,81	1,74	0,1963	=

34,79% no R_{MAE} e 29,29% no R_{MAPE} em comparação com o $LocPart_{ELM}$. O teste de hipóteses DM forneceu evidências estatísticas de que o LocLN tem erros quadráticos menores em 39% dos casos entre os 18 métodos concorrentes. Nos 61% dos casos restantes, o teste DM indicou igualdade.

A Tabela 23 apresenta os resultados das relações percentuais e do teste DM para a base de dados 9. Observa-se que o LocLN obteve ganhos percentuais de 14,10% no R_{RMSE} e 17,10% no R_{MAE} em comparação com $BAGG_{ARIMA}$, e 23,66% no R_{MAPE} em comparação com $LocPart_{GRU}$. O teste de hipóteses DM indicou que o LocLN tem desempenho preditivo superior nos erros quadráticos em 39% das comparações considerando os 18 métodos concorrentes. Em 50% dos casos, o teste DM indicou igualdade. E, nos 11% dos casos restantes, o teste DM apresentou diferença significativa nos erros quadráticos, e indicou que os métodos $LocPart_{ARIMA}$ e LC_{AM} superaram o LocLN.

Os resultados da relação percentual e do teste de hipóteses DM para a base de dados

Tabela 23 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 9.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	13,60	11,37	8,28	10,24	1,09E-02	+
	ARIMA (TORRES et al., 2005)	-2,25	-4,23	-7,13	-0,26	0,4638	=
	ELM (SAAVEDRA-MORENO et al., 2013)	1,16	-0,52	-1,96	-7,93	0,2661	=
	LSTM (LIU; LIN; FENG, 2021)	5,63	7,63	10,17	2,18	4,98E-02	+
	GRU (LIU; LIN; FENG, 2021)	10,72	13,16	17,08	4,73	4,15E-03	+
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,57	1,09	-1,96	3,44	2,93E-01	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,21	1,51	1,33	-3,69	2,18E-01	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	-11,89	-13,35	-13,66	0,95	1,05E-02	-
	LC_{AS} (de MATTOS NETO et al., 2020)	-9,96	-8,49	-11,20	-4,79	0,2661	=
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	-8,37	-11,05	-12,18	0,95	0,2833	=
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	-9,76	-7,30	-9,52	-7,93	0,5223	=
	LC_{AE}	-3,14	-4,87	-6,74	-3,69	0,7616	=
	$NC - E_{AE}$ (ALMEIDA, 2024)	0,08	-0,90	-1,34	3,44	0,2394	=
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	-1,49	-1,22	-2,07	-5,86	2,23E-03	-
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	5,23	8,34	10,47	-2,57	0,08014	=
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	11,08	14,54	19,72	**13,22**	2,75E-02	+
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	**14,10**	**17,70**	**23,66**	11,71	7,73E-03	+
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	-13,02	-13,02	-10,69	-2,57	0,1365	=

Tabela 24 – Relação percentual por métrica e teste de hipóteses de Diebold-Mariano do método LocLN em relação aos demais métodos concorrentes no conjunto de teste para a base de dados 10.

Abordagem	**Método**	R_{RMSE}(%)	R_{MAE}(%)	R_{MAPE}(%)	R_{POCID}(%)	**p-valor**	**Teste DM**
Individual	Persistência	**15,63**	**12,94**	**11,48**	**16,41**	1,63E-03	+
	ARIMA (TORRES et al., 2005)	10,55	10,64	7,95	10,77	1,86E-02	+
	ELM (SAAVEDRA-MORENO et al., 2013)	-1,78	-5,01	-7,01	-5,92	0,3996	=
	LSTM (LIU; LIN; FENG, 2021)	3,83	2,40	4,45	7,31	0,1370	=
	GRU (LIU; LIN; FENG, 2021)	4,70	3,22	5,64	10,77	0,0904	=
MHP	$BAGG_{ARIMA}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	8,40	6,96	5,34	14,47	1,04E-02	+
	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	2,64	1,72	0,19	-1,89	3,71E-02	+
MHS	LC_{AM} (CADENAS; RIVERA, 2010)	-7,85	-9,31	-10,63	-8,43	0,3099	=
	LC_{AS} (de MATTOS NETO et al., 2020)	-8,76	-12,21	-11,54	-11,95	0,4385	=
	$NC - S_{AM}$ (de MATTOS NETO et al., 2020)	-8,80	-11,13	-12,10	-4,61	0,2399	=
	$NC - S_{AS}$ (de MATTOS NETO et al., 2020)	-8,90	-12,53	-11,57	-9,63	0,4902	=
	LC_{AE}	-9,36	-8,88	-10,07	-13,06	0,2680	=
	$NC - E_{AE}$ (ALMEIDA, 2024)	-7,28	-10,37	-10,79	-13,06	0,4874	=
Propostas de MHP com mapeamento local	$LocPart_{ARIMA}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	-2,41	-3,03	-6,24	-3,27	0,2079	=
	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,56	0,54	1,04	-5,92	0,2206	=
	$LocPart_{LSTM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	4,81	3,63	5,30	10,77	0,0690	=
	$LocPart_{GRU}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	5,47	4,35	5,85	10,77	0,0478	=
Proposta de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	-5,98	-7,56	-8,24	-5,92	0,7262	=

10 são apresentados na Tabela 24. Observa-se que o LocLN obteve ganhos percentuais de 15,63% no R_{RMSE}, 12,94% no R_{MAE}, 11,48% no R_{MAPE} e 16,41% no R_{POCID} em comparação com o modelo de persistência. Os resultados do teste de hipóteses DM indicaram que o LocLN tem diferença significativa nos erros quadráticos em 22% dos casos, evidenciando a superioridade do LocLN contra os 18 métodos concorrentes. Nos demais 78% dos casos, o teste DM indicou igualdade.

5.2.3 Gráficos de Previsão

Do mesmo modo que ocorreu com as bases de dados horárias, foram selecionados alguns métodos juntamente com o LocLN para comparar visualmente as previsões das séries temporais com intervalos temporais de 3h no conjunto de teste. Primeiro, o método $LocPart_{ARIMA}$, que faz parte do Estágio I do LocLN. Além disso, um modelo indivi-

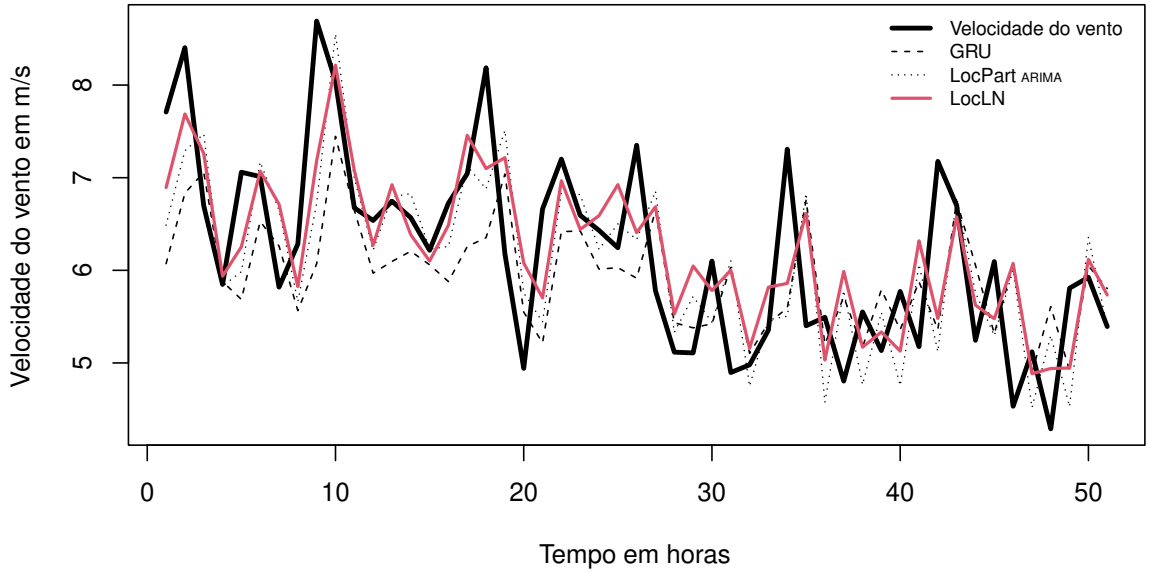

Figura 39 – Resultados da previsão de um passo à frente para LocLN em comparação com o GRU individual e o $LocPart_{ARIMA}$ para a base de dados 6, considerando uma sequência de 50 observações do conjunto de teste.

dual foi escolhido com base nos valores da relação percentual R_i. Para a base de dados 6, o GRU foi selecionado porque foi o modelo individual em que LocLN obteve o maior ganho percentual. A Figura 39 ilustra as previsões um passo à frente considerando 50 observações sequenciais no conjunto de teste para a base de dados 6. Verifica-se a superioridade do LocLN, particularmente em comparação ao GRU. A diferença de erros entre $LocPart_{ARIMA}$ e LocLN é mais sutil, no entanto, é notável que após a observação de número 30, $LocPart_{ARIMA}$ mostra erros aparentes nos picos e vales da série temporal.

O modelo individual LSTM foi selecionado para a avaliação gráfica da base de dados 7. A Figura 40 mostra as previsões de um passo à frente considerando 50 observações sequenciais do conjunto de teste para a base de dados 7. A superioridade do LocLN sobre o LSTM é notável. Já para o $LocPart_{ARIMA}$ as diferenças são sutis.

A Figura 41 mostra as previsões de um passo à frente considerando 50 observações sequenciais do conjunto de teste para a base de dados 8. O modelo individual selecionado foi ELM. A superioridade do LocLN sobre o ELM é evidente. Em contraste, para $LocPart_{ARIMA}$, há uma dificuldade em discernir qual das curvas tem o menor erro.

O modelo individual GRU foi selecionado para a avaliação gráfica para a base de dados 9, e a Figura 42 mostra as previsões de um passo à frente considerando 50 observações sequenciais do conjunto de teste para a base de dados 9. LocLN supera o GRU, principalmente nos picos e vales da série temporal. E, comparado a $LocPart_{ARIMA}$, não há diferenças substanciais.

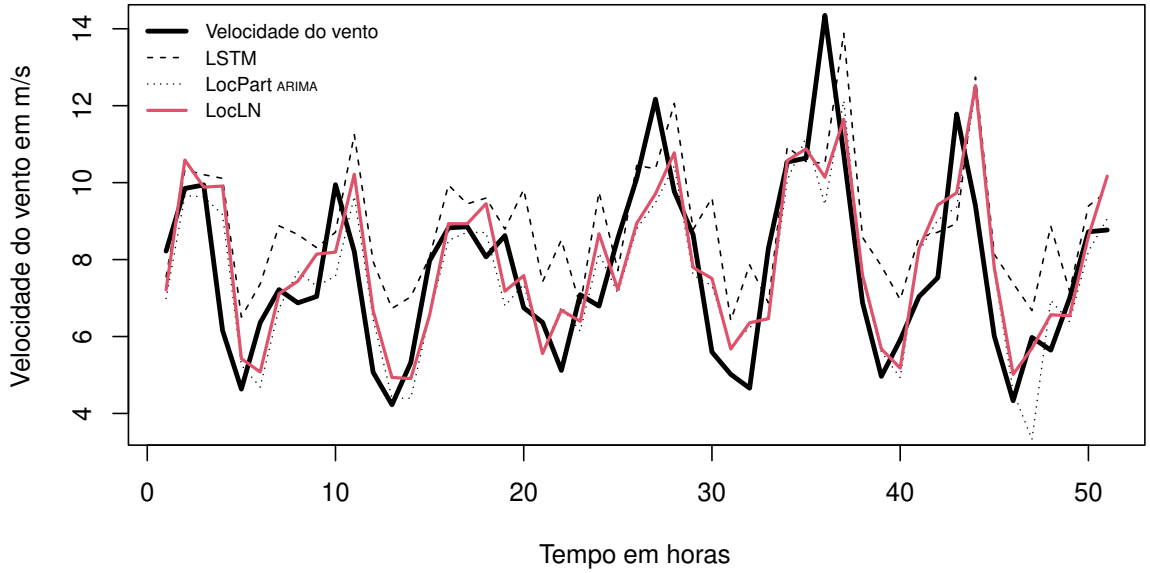

Figura 40 – Resultados da previsão de um passo à frente para LocLN em comparação com o LSTM individual e o $LocPart_{ARIMA}$ para a base de dados 7, considerando uma sequência de 50 observações do conjunto de teste.

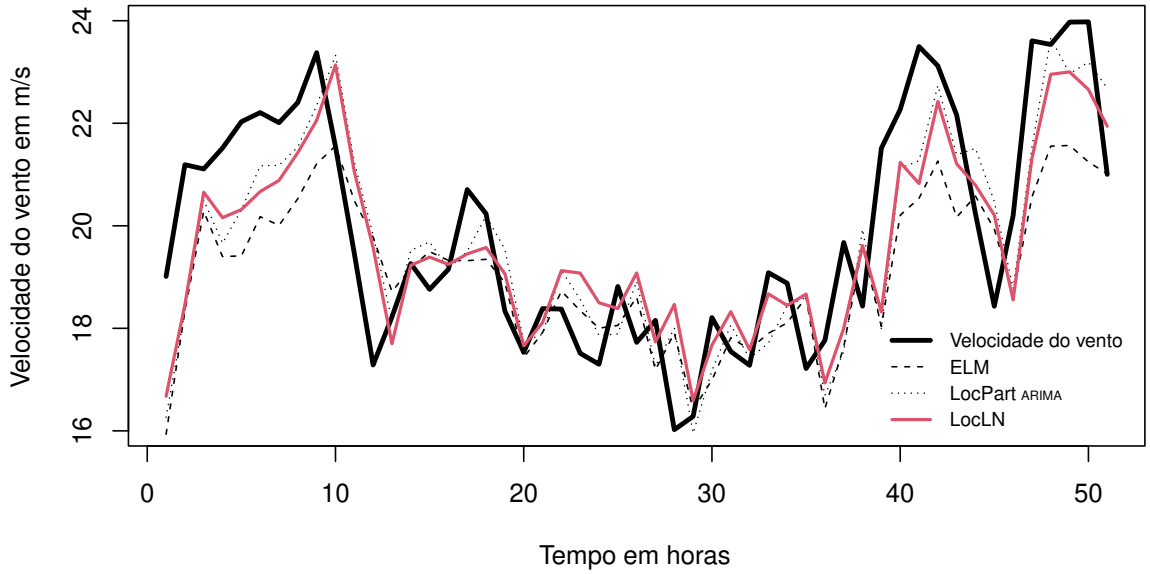

Figura 41 – Resultados da previsão de um passo à frente para LocLN em comparação com o ELM individual e o $LocPart_{ARIMA}$ para a base de dados 8, considerando uma sequência de 50 observações do conjunto de teste.

E, finalmente, a Figura 43 mostra as previsões de um passo à frente considerando 50

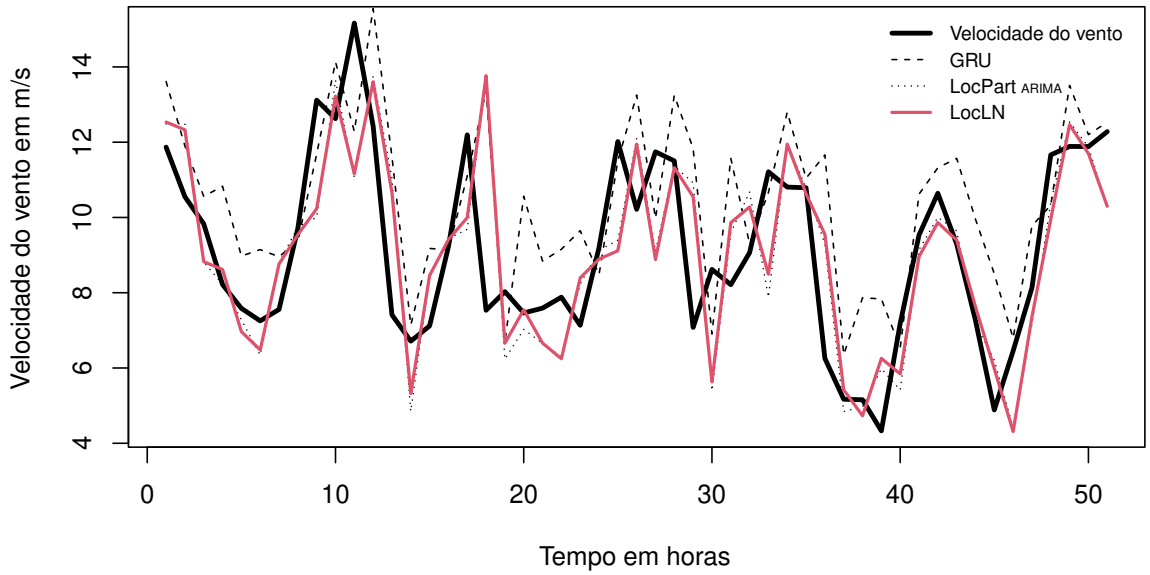

Figura 42 – Resultados da previsão de um passo à frente para LocLN em comparação com o GRU individual e o $LocPart_{ARIMA}$ para a base de dados 9, considerando uma sequência de 50 observações do conjunto de teste.

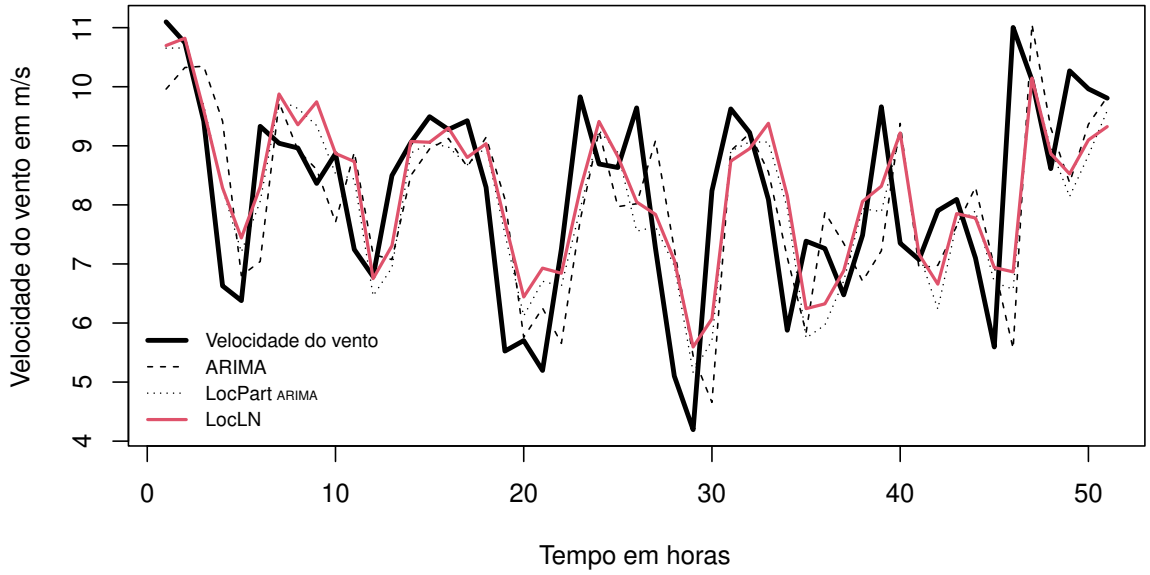

Figura 43 – Resultados da previsão de um passo à frente para LocLN em comparação com o ARIMA individual e o $LocPart_{ARIMA}$ para a base de dados 10, considerando uma sequência de 50 observações do conjunto de teste.

observações sequenciais do conjunto de teste para a base de dados 10. O modelo individual

selecionado foi o ARIMA. Observa-se que o LocLN supera o ARIMA em diferentes partes da série temporal. Já o $LocPart_{ARIMA}$ exibe um comportamento mais semelhante ao método LocLN.

5.3 Resultados sobre 100 execuções dos métodos com o modelo ELM

Nesta seção estão os resultados da média e desvio padrão após 100 execuções com diferentes sementes para os métodos que empregaram o modelo ELM. O modelo ELM é o que tem o menor tempo computacional para treinar, além de ser o único modelo base não linear empregado nos métodos propostos LocLinNonPR e LocLN. Por esses motivos, foram selecionados o ELM individual, o MHP $BAGG_{ELM}$, os MHSs LC_{AE} e $NC - E_{AE}$, a proposta de MHP com mapeamento local $LocPart_{ELM}$, e as propostas de MHP com MHS e mapeamento local: LocLinNonPR e LocLN.

As cinco bases horárias foram utilizadas neste experimento juntamente com as medidas de avaliação RMSE e MAPE_{media}. O RMSE é a principal métrica de avaliação, pois é a métrica utilizada para selecionar os hyperparâmetros dos modelos. Já o MAPE_{media} é uma métrica importante que também deve ser considerada. A relevância MAPE_{media} é devido a métrica computar os erros nos momentos que a velocidade do vento está acima da média da série. É nestes momentos que ocorre a maior geração de energia eólica.

Também foi considerado o tempo de treinamento de cada método para uma execução. Cada método foi treinado individualmente em uma mesma máquina com o seguinte *hardware*: processador Intel Core i7-1165G7, memória 8GB de RAM e armazenamento SSD NVME 256GB. Por fim, o teste estatístico de hipóteses de Diebold e Mariano foi aplicado para verificar se há diferença significativa nos erros médios sobre as 100 execuções entre a principal proposta LocLN e a abordagem individual ELM. E assim, por meio do teorema do limite central, assumiu-se que os erros médios dos métodos nestas 100 execuções seguem uma distribuição normal.

Tabela 25 – Resultados para base de dados 1. Média e desvio padrão no conjunto de teste para 100 execuções. E, tempo de treinamento no conjunto de treinamento para uma execução.

Abordagem	**Método**	**RMSE(m/s)**	**MAPE_{media}(%)**	**Tempo(minutos)**
Individual	ELM (SAAVEDRA-MORENO et al., 2013)	0,8123 (0,0010)	9,46 (0,05)	0,04
MHP	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	0,8040 (0,0004)	9,16 (0,01)	3,21
MHS	LC_{AE}	0,7678 (0,0046)	8,25 (0,08)	2,08
	$NC - E_{AE}$ (ALMEIDA, 2024)	0,7672 (0,0042)	8,37 (0,10)	2,09
Propostas de MHP com mapeamento local	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	0,7996 (0,0044)	8,76 (0,20)	1,17
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	0,7677 (0,0039)	8,43 (0,10)	3,28
	LocLN	**0,7581** (0,0044)	**8,23** (0,14)	22,70

A Tabela 25 exibe a média e o desvio padrão no conjunto de teste para 100 execuções na base de dados 1. A primeira coluna identifica a abordagem, tais quais, individual,

MHP, MHS, propostas de MHP com mapeamento local, e, por último, propostas de MHP com MHS e mapeamento local. A segunda coluna descreve os métodos desenvolvidos. Posteriormente, os resultados das duas medidas de avaliação, RMSE e MAPE_{media} são apresentados para cada método. O melhor resultado para a média em cada métrica está destacado em negrito. Por fim, na última coluna é apresentado o tempo em minutos do treinamento no conjunto de treinamento para uma única execução dos métodos. Observa-se que o valor médio da proposta LocLN superou todos os métodos no RMSE e no MAPE_{media}. Ademais, o valor médio do LocLN obteve ganhos percentuais de 6,67% no R_{RMSE} e 13,10% no $R_{MAPE_{media}}$ em comparação com o modelo ELM individual.

Com relação ao tempo computacional, o LocLN é o mais custoso, com um tempo de treinamento de 22,70 minutos. O LocLN envolve o treinamento de 60 modelos ARIMA para o mapeamento de padrões locais na série, e 60 modelos ELM para o reconhecimento de padrões locais nos resíduos. São 120 modelos que precisam ser treinados. Contudo, o ARIMA e o ELM são considerados modelos com baixo custo computacional, e esse foi um dos motivos de serem selecionados para compor o LocLN. Por exemplo, os modelos LSTM e GRU podem ter um treinamento com um tempo centenas de vezes maior que o ELM. E, assumindo uma função de perda quadrática, ao aplicar o teste de hipóteses DM sobre o valores do RMSE nas 100 execuções entre o LocLN e do modelo individual ELM, o p-valor obtido foi de 2,2E-16. Isso indica que há significância estatística entre os erros dos dois métodos.

Tabela 26 – Resultados para base de dados 2. Média e desvio padrão no conjunto de teste para 100 execuções. E, tempo de treinamento no conjunto de treinamento para uma execução.

Abordagem	**Método**	**RMSE(m/s)**	**MAPE_{media}(%)**	**Tempo(minutos)**
Individual	ELM (SAAVEDRA-MORENO et al., 2013)	1,3680 (0,0123)	**9,46** (0,14)	0,05
MHP	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,3411 (0,0066)	9,55 (0,06)	3,33
MHS	LC_{AE}	1,3380 (0,0131)	9,90 (0,14)	2,67
	$NC - E_{AE}$ (ALMEIDA, 2024)	1,3169 (0,0110)	9,61 (0,12)	2,68
Propostas de MHP com mapeamento local	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,3315 (0,0048)	9,59 (0,13)	1,15
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,3205 (0,0123)	9,60 (0,15)	2,86
	LocLN	**1,3151** (0,0108)	9,74 (0,17)	23,10

A média e o desvio padrão das métricas RMSE e MAPE_{media} para 100 execuções, e o tempo de treinamento para uma execução na base de dados 2 são apresentados na Tabela 26. Esta tabela, bem como as Tabelas 27, 28 e 29, têm um layout de coluna igual ao da Tabela 5. Verifica-se que o valor médio da proposta LocLN superou todos os métodos no RMSE, e o ELM individual obteve o menor erro MAPE_{media}. Ademais, o valor médio do LocLN obteve ganho percentual de 3,87% no R_{RMSE} em comparação com o modelo ELM individual. O maior custo computacional foi do método LocLN com 23,10 minutos. Assumindo uma função de perda quadrática, ao aplicar o teste de hipóteses DM sobre o RMSE nas 100 execuções entre o LocLN e do modelo individual ELM, o p-valor obtido foi de 2,2E-16, indicando que há significância estatística entre os erros dos dois métodos.

Tabela 27 – Resultados para base de dados 3. Média e desvio padrão no conjunto de teste para 100 execuções. E, tempo de treinamento no conjunto de treinamento para uma execução.

Abordagem	**Método**	**RMSE(m/s)**	**MAPE$_{media}$(%)**	**Tempo(minutos)**
Individual	ELM (SAAVEDRA-MORENO et al., 2013)	1,1202 (0,0241)	4,76 (0,16)	0,04
MHP	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,0584 (0,0038)	4,32 (0,03)	3,54
	LC_{AE}	1,0340 (0,0128)	3,92 (0,08)	2,69
MHS	$NC - E_{AE}$ (ALMEIDA, 2024)	1,0687 (0,0243)	4,30 (0,16)	2,70
Propostas de MHP com mapeamento local	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,0543 (0,0284)	4,15 (0,22)	1,15
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,0923 (0,0229)	4,53 (0,16)	2,96
	LocLN	**1,0216** (0,0078)	**3,81** (0,04)	22,50

Na Tabela 27 estão a média e o desvio padrão das métricas RMSE e MAPE$_{media}$ para 100 execuções, e o tempo de treinamento para uma execução na base de dados 3. Mais uma vez, conforme ocorreu na base de dados 1, o valor médio da proposta LocLN superou todos os métodos no RMSE e no MAPE$_{media}$. Além disso, o valor médio do LocLN obteve ganhos percentuais de 8,80% no R_{RMSE} e 19,96% no $R_{MAPE_{media}}$ em comparação com o modelo ELM individual. O maior custo computacional foi do método LocLN com 22,50 minutos. Assumindo uma função de perda quadrática, ao aplicar o teste de hipóteses DM sobre o RMSE nas 100 execuções entre o LocLN e do modelo individual ELM, o p-valor obtido foi de 2,2E-16. Isto demonstra que há significância estatística entre os erros dos dois métodos.

Tabela 28 – Resultados para base de dados 4. Média e desvio padrão no conjunto de teste para 100 execuções. E, tempo de treinamento no conjunto de treinamento para uma execução.

Abordagem	**Método**	**RMSE(m/s)**	**MAPE$_{media}$(%)**	**Tempo(minutos)**
Individual	ELM (SAAVEDRA-MORENO et al., 2013)	1,1532 (0,0051)	6,49 (0,06)	0,04
MHP	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,1670 (0,0083)	6,67 (0,07)	3,27
	LC_{AE}	1,1804 (0,0125)	6,70 (0,11)	2,33
MHS	$NC - E_{AE}$ (ALMEIDA, 2024)	1,2513 (0,0557)	**6,42** (0,19)	2,35
Propostas de MHP com mapeamento local	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,1345 (0,0041)	6,53 (0,05)	1,15
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,2419 (0,0403)	6,58 (0,21)	2,99
	LocLN	**1,1047** (0,0061)	6,72 (0,05)	21,90

A média e o desvio padrão das métricas RMSE e MAPE$_{media}$ para 100 execuções, e o tempo de treinamento para uma execução na base de dados 4 são ilustrados na Tabela 28. Observa-se que o valor médio da proposta LocLN superou todos os métodos no RMSE, e $NC - E_{AE}$ obteve o menor erro MAPE$_{media}$. E, o valor médio do LocLN obteve ganho percentual de 4,21% no R_{RMSE} em comparação com o modelo ELM individual. O maior custo computacional foi do método LocLN com 21,90 minutos. Assumindo uma função de perda quadrática, ao aplicar o teste de hipóteses DM sobre o RMSE nas 100 execuções entre o LocLN e do modelo individual ELM, o p-valor obtido foi de 2,2E-16. Isto revela que há significância estatística entre os erros dos dois métodos.

Na Tabela 29 estão A média e o desvio padrão das métricas RMSE e MAPE$_{media}$ para 100 execuções, e o tempo de treinamento para uma execução na base de dados 5. O valor médio da proposta LocLN superou todos os métodos no RMSE. O $LocPart_{ELM}$ obteve o

Tabela 29 – Resultados para base de dados 5. Média e desvio padrão no conjunto de teste para 100 execuções. E, tempo de treinamento no conjunto de treinamento para uma execução.

Abordagem	**Método**	**RMSE(m/s)**	**MAPE$_{media}$(%)**	**Tempo(minutos)**
Individual	ELM (SAAVEDRA-MORENO et al., 2013)	1,0993 (0,0108)	8,15 (0,08)	0,04
MHP	$BAGG_{ELM}$ based on (BERGMEIR; HYNDMAN; BENÍTEZ, 2016)	1,1041 (0,0039)	8,11 (0,04)	3,48
	LC_{AE}	1,0782 (0,0067)	7,87 (0,10)	2,88
MHS	$NC - E_{AE}$ (ALMEIDA, 2024)	1,0806 (0,0070)	7,87 (0,10)	2,89
Propostas de MHP com mapeamento local	$LocPart_{ELM}$ (ALMEIDA; de MATTOS NETO; CUNHA, 2024)	1,0923 (0,0046)	**7,76** (0,11)	1,17
Propostas de MHP com MHS e mapeamento local	LocLinNonPR (ALMEIDA, 2024)	1,0837 (0,0054)	7,89 (0,09)	2,77
	LocLN	**1,0768** (0,0090)	7,85 (0,13)	22,20

menor erro MAPE$_{media}$. Ademais, o valor médio do LocLN obteve ganhos percentuais de 2,05% no R_{RMSE} e 3,68% no $R_{MAPE_{media}}$ em comparação com o modelo ELM individual. O maior custo computacional foi do método LocLN com 22,20 minutos. Assumindo uma função de perda quadrática, ao aplicar o teste de hipóteses DM sobre o RMSE nas 100 execuções entre o LocLN e do modelo individual ELM, o p-valor obtido foi de 2,2E-16, demostrando que há significância estatística entre os erros dos dois métodos.

5.4 Discussão

Na Seção 5.1 foi possível observar os resultados para as cinco bases horárias. O mapeamento de padrões locais foi relevante, principalmente para a proposta LocLN, que empregou um mapeamento de padrões locais na série e também nos seus resíduos. Na base de dados 1, o método LocLN conseguiu obter ganhos percentuais de 17,09% no R_{RMSE}, 18,10% no R_{MAE}, 10,59% no R_{MAPE} e 4,57% no R_{POCID} em comparação com o modelo LSTM individual.

Já na Seção 5.2, foram apresentados os resultados para as cinco bases de granularidade maior, que possuem intervalos de tempo de 3h. O mapeamento de padrões locais também foi promissor, com destaque para a proposta LocLinNonPR, que aplicou um mapeamento local somente na série, e nos resíduos foi empregado um reconhecimento de padrões globais. No entanto, o LocLN também mostrou sua relevância. Por exemplo, na base de dados 7, observou-se que o LocLN conseguiu ganhos percentuais de 19,91% no R_{RMSE}, 23,15% no R_{MAE}, 28,99% no R_{MAPE} e 14,95% no R_{POCID} em comparação com o modelo LSTM individual.

Por meio dos resultados em séries com granularidades diferentes, verificou-se algumas limitações nas propostas. Pois, nem sempre o LocLN foi o melhor método. As bases de granularidade maior possuem menos observações no mesmo intervalo de tempo, quando comparadas com as bases de granularidade menor. Isso implica em menos padrões locais, pois as observações das séries de granularidade maior são suavizadas por meio de médias. Além disso, as séries com intervalos de tempo de 3h, ao considerar o LocLN, foi possível gerar 20 modelos ARIMA para o mapeamento local das séries e 12 modelos ELM para o

mapeamento local dos seus resíduos. Enquanto nas bases horárias, conseguiu-se gerar 60 modelos ARIMA para o reconhecimento dos padrões locais das séries e 60 modelos ELM para o mapeamento dos padrões locais dos seus resíduos. O fato do LocLN gerar menos modelos nas séries de granularidade maior limitou a capacidade de busca por padrões locais nas séries. Isso implicou diretamente no desempenho final do LocLN.

Na Seção 5.3, foi verificado o comportamento dos métodos ao incluir a aleatoriedade da inicialização da rede neural ELM. Por esse motivo, os métodos foram executados 100 vezes com sementes diferentes nas bases de dados horárias. Os resultados da proposta LocLN foram promissores, e indicaram significância estatística nos erros quando comparado com o ELM individual. Em relação ao tempo computacional das abordagens com o ELM, o LocLN foi o mais custoso. Visto que, envolve o treinamento de 60 modelos ARIMA para o mapeamento de padrões locais lineares na série, e 60 modelos ELM para o mapeamento de padrões locais não lineares nos resíduos. No total, são treinados 120 modelos. Porém, o tempo de treinamento próximo de 20 minutos em um processador simples, modelo Core i7, pode ser considerado bom neste caso. O problema envolve o estabelecimento de previsões de um passo à frente para horizontes de 1h. Logo, há tempo suficiente para obter os resultados das previsões.

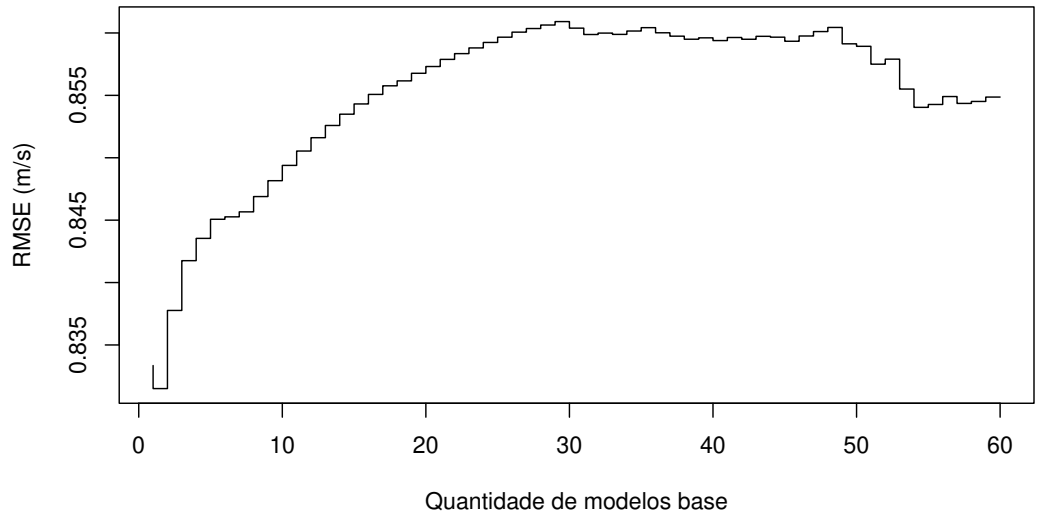

Figura 44 – Gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de modelos combinados para a base de dados 1.

Outra questão que deve ser considerada é a etapa de seleção do LocPart, que também é utilizada no LocLN. Pois, o LocPart foi empregado com o modelo ARIMA para o reconhecimento de padrões locais lineares na Etapa I do LocLN. Ao considerar os resultados da Seção 5.1 com o $LocPart_{ARIMA}$, a quantidade de modelos selecionados na etapa de seleção é variável e depende da base de dados. Por exemplo, na Figura 44 está ilustrado um gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de

acordo com a quantidade de preditores combinados para a base de dados 1. Primeiramente, conforme mencionado na Seção 4.2.1, os preditores foram ranqueados de acordo com o menor RMSE. Posteriormente, os preditores foram combinados para obter o RMSE da combinação. O primeiro degrau, representado pela primeira reta vertical à esquerda na Figura 44, é o RMSE do modelo com o menor erro. No segundo degrau está o RMSE da combinação dos dois modelos com o menor erro, e assim segue até obter o RMSE de todos os 60 modelos combinados. Ao observar todo o gráfico, há uma variação no RMSE de acordo com a quantidade de modelos selecionados. O menor RMSE está no segundo degrau, indicando que o conjunto com os dois primeiros preditores é a melhor seleção para o $LocPart_{ARIMA}$ na base de dados 1.

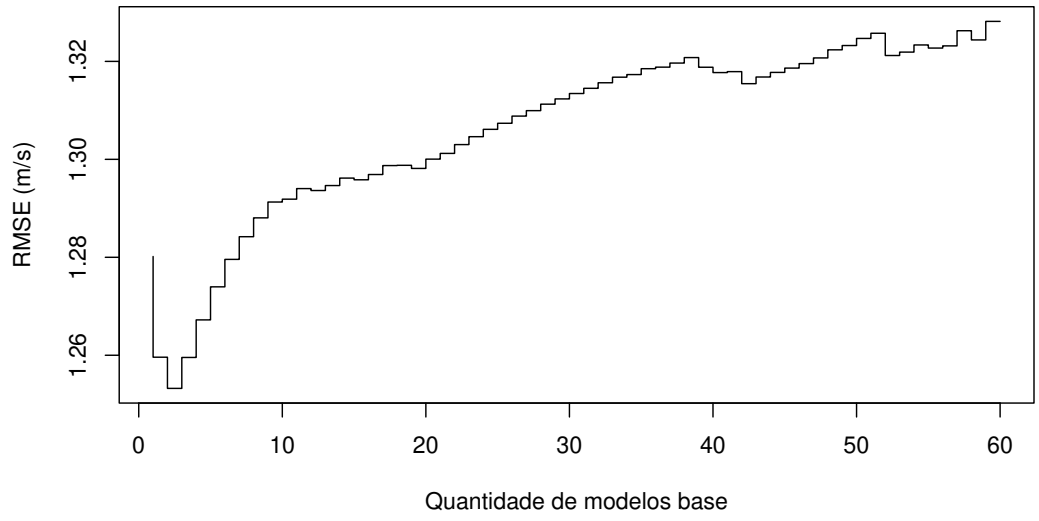

Figura 45 – Gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de modelos combinados para a base de dados 2.

Já na Figura 45 está exibido o gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de preditores combinados para a base de dados 2. Verifica-se que também há uma variação no RMSE ao variar a quantidade de modelos selecionados. O menor RMSE está no terceiro degrau, indicando que a melhor seleção é o conjunto com os três primeiros preditores na base de dados 2.

O gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de preditores combinados para a base de dados 3 é mostrado na Figura 46. Observa-se que variações no RMSE de acordo com a quantidade de modelos selecionados. O menor RMSE está no 52º degrau. Isso indica que a melhor seleção é o conjunto com os 52 primeiros preditores para a base de dados 3.

Na Figura 47 está exibido o gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de preditores combinados para a base de dados 4. Também observa-se uma variação no RMSE ao variar a quantidade de modelos

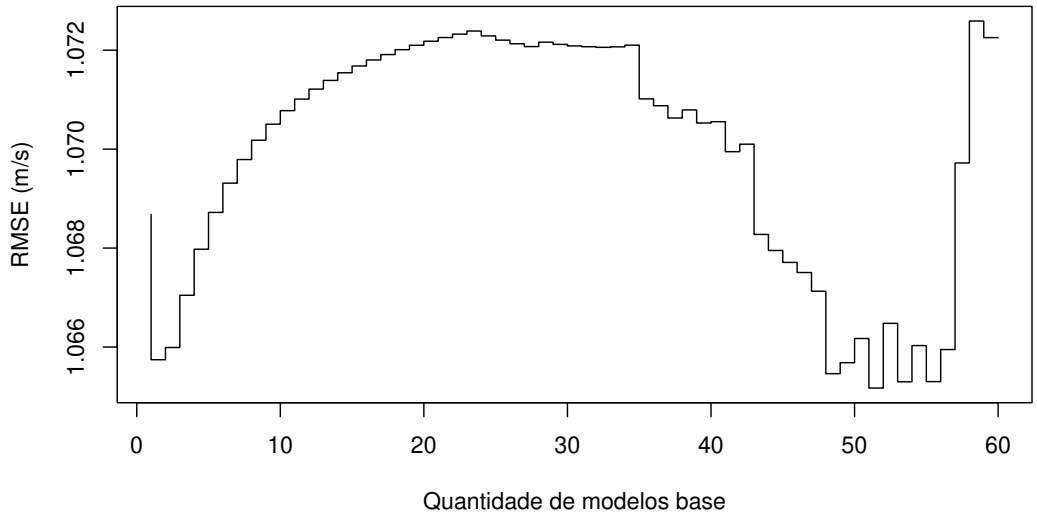

Figura 46 – Gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de modelos combinados para a base de dados 3.

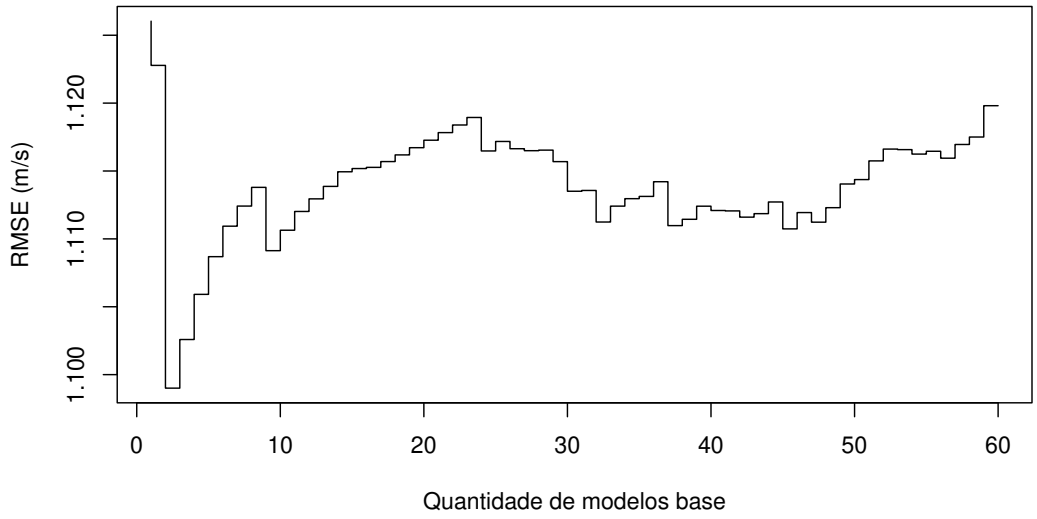

Figura 47 – Gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de modelos combinados para a base de dados 4.

selecionados. O menor RMSE está no terceiro degrau, indicando que a melhor seleção é o conjunto com os três primeiros preditores na base de dados 4.

Por fim, o gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de preditores combinados para a base de dados 5 é mostrado na Figura 48. Observa-se que há variações no RMSE de acordo com a quantidade de modelos selecionados. O menor RMSE nesta base está no último degrau. Isso indica que a melhor seleção é o conjunto com os todos os preditores na base de dados 5. No geral, os resultados

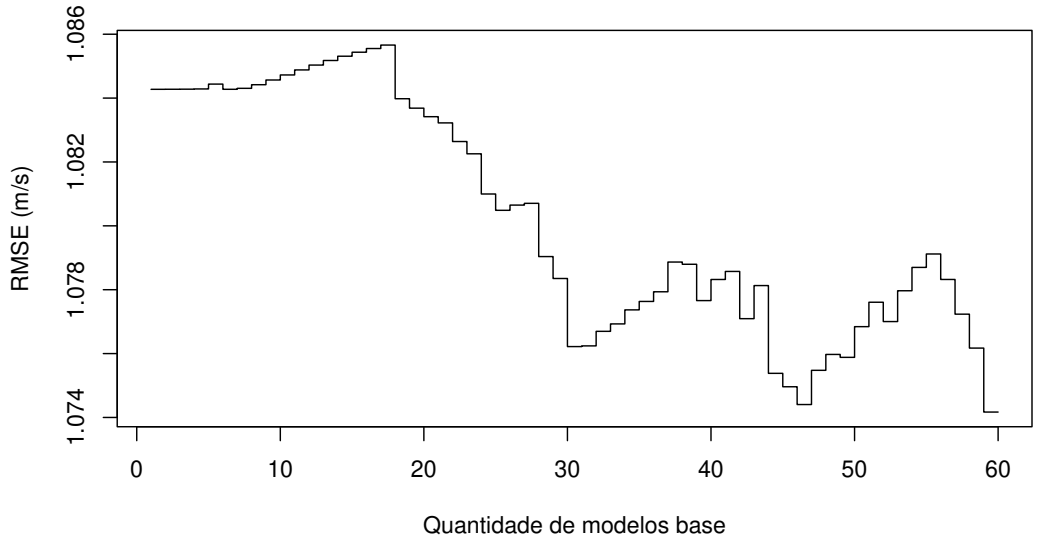

Figura 48 – Gráfico escada com o RMSE do $LocPart_{ARIMA}$ no conjunto de seleção de acordo com a quantidade de modelos combinados para a base de dados 5.

apresentados nos gráficos escada nessas cinco bases de dados evidenciaram a importância da etapa de seleção. Pois, na maioria das vezes, é preciso selecionar os preditores mais capacitados para poder obter uma combinação mais eficientemente e competente para o reconhecimento dos padrões locais.

É importante destacar também as contribuições das propostas para a área de energia eólica. Pois, os ganhos percentuais nas métricas de avaliação em relação aos métodos tradicionais com mapeamento global evidenciaram a maior precisão das propostas, principalmente a relevância do LocLN. Por exemplo, na Seção 5.2, a diminuição percentual nos erros do LocLN para o modelo GRU individual na base de dados 7 foi de 18,47% no R_{RMSE}, 21,91% no R_{MAE}, 27,81% no R_{MAPE} e 11,97% no R_{POCID}. Além disso, na base de dados 8, o LocLN conseguiu uma diminuição percentual nos erros de 20,03% no R_{RMSE}, 20,84% no R_{MAE}, 15,66% no R_{MAPE} e 9,11% no R_{POCID} em comparação com o ELM individual. Os mercados de energia operam com contratos baseados em previsões de geração. Logo, erros nas previsões de geração podem resultar na necessidade de compra de energia para compensação. Portanto, um modelo mais preciso e confiável, possibilita que a geradora reduza os riscos de ajustes de última hora. Isto evita a compra emergencial de energia a preços desfavoráveis. Ademais, previsões mais precisas também permitem uma gestão mais eficiente nos aerogeradores dos parques eólicos. Pois, proporcionam uma manutenção preditiva nos momentos mais adequados. E assim, reduz-se os custos com reparos, viabilizando um aumento da vida útil das turbinas. Por fim, empresas com previsões mais precisas e com maior estabilidade nos planejamentos de geração de energia, tendem a atrair mais investimentos.

5.5 Resumo do Capítulo

Neste capítulo, foram expostos e discutidos os resultados obtidos dos modelos propostos em séries temporais de velocidade do vento com granularidades diferentes para previsões de um passo. Foram utilizadas cinco medidas de qualidade em relação à acurácia, métricas de relação percentual, gráficos das previsões, e teste de hipóteses de Diebold-Mariano. Ficou evidenciado que, no geral, as propostas conseguem obter resultados melhores que os modelos individuais, MHPs e MHS encontrados na literatura. Esses resultados colaboram com a principal hipótese dessa Tese de Doutorado, que presume que a aplicação de um MHP com reconhecimento local de padrões, juntamente com uma técnica de MHS irá empregar um mapeamento de padrões mais adequado em séries temporais de velocidade do vento. No capítulo a seguir, serão apresentadas as conclusões do trabalho.

6 CONCLUSÕES

Este capítulo compreende considerações finais, algumas questões levantadas na pesquisa, contribuições, dificuldades encontradas e possíveis trabalhos futuros.

6.1 Considerações Finais

Este trabalho reuniu diferentes contribuições da pesquisa do doutorado. O objetivo foi desenvolver modelos híbridos para reconhecer os padrões locais complexos em séries temporais de velocidade do vento de uma forma mais apropriada que os métodos encontrados na literatura. A proposta LocPart possibilitou a geração de subconjuntos variáveis em relação ao tamanho e à região no conjunto de treinamento para, posteriormente, selecionar no conjunto de validação os preditores especialistas locais mais capacitados em relação à acurácia.

A proposta LocLinNonPR é um método híbrido que aplicou um modelo híbrido em paralelo (MHP) para o reconhecimento de padrões locais lineares por meio do ARIMA em subconjuntos de diferentes regiões e tamanhos iguais no conjunto de treinamento. Depois, foi empregada uma técnica de modelos híbridos seriais (MHSs) com mapeamento global não linear através do ELM individual. Por fim, um método de combinação não linear (modelo ELM) foi utilizado para combinar a previsão da série com a previsão dos resíduos.

Já a proposta LocLN também é um método híbrido que combinou um MHP com um MHS. Neste caso, o reconhecimento de padrões locais lineares e não lineares foi realizado por um empilhamento do MHP LocPart. Desta forma, um LocPart com modelos lineares ARIMA foi aplicado à série temporal, e um LocPart com modelos não lineares ELM foi empregado nos resíduos da série. E, finalmente, um método de combinação entre o linear (adição) e o não linear (modelo ELM) foi utilizado para combinar a previsão da série com a previsão dos resíduos.

Dez bases de dados univariadas de velocidades de vento em cidades no Nordeste brasileiro foram utilizadas. Em que, quatro bases são séries horárias no decorrer de três meses obtidas por meio do Sistema de Organização Nacional de Dados Ambientais (SONDA) do Brasil, e outras cinco bases são as séries anteriores com uma granularidade maior em intervalos temporais de 3h. No geral, os resultados nas cinco medidas de avaliação (RMSE, MAE, MAPE, POCID e MAPE_{media}) demonstram a eficácia das propostas.

O destaque foi o LocLN, principalmente nas bases de dados de menor granularidade, as bases horárias. Pois, foi possível gerar uma maior quantidade de modelos base para a busca de padrões locais em relação às séries de maior granularidade, as bases com

intervalos temporais de 3h. Já o método LocLinNonPR teve maior relevância nas cinco bases de maior granularidade, que são séries com uma maior suavização obtida por médias das observações das séries horárias. A suavização elimina flutuações de curto prazo e diminui a quantidade de observações da série. O LocLinNonPR é um método menos sensível ao reconhecimento de padrões locais em relação ao LocLN. Pois, não emprega uma variação no tamanho dos subconjuntos da etapa de geração do MHP, nem aplica um mapeamento local nos resíduos. Assim, identificou-se uma limitação do método LocLN em relação à granularidade e ao tamanho das séries temporais de velocidade do vento. Isso evidenciou que o desempenho do LocLN em relação à acurácia é mais promissor em séries de granularidade menor e com uma quantidade maior de observações.

A respeito dos resultados sobre as relações percentuais nas medidas de qualidade para as bases horárias na Seção 5.1, o LocLN conseguiu obter ganhos percentuais de 17,09% no R_{RMSE}, 18,10% no R_{MAE}, 10,59% no R_{MAPE} e 4,57% no R_{POCID} em comparação com o modelo individual LSTM para a base de dados 1. E quanto às bases com intervalos temporais de 3h na Seção 5.2, o LocLN alcançou ganhos percentuais de 19,91% no R_{RMSE}, 23,15% no R_{MAE}, 28,99% no R_{MAPE} e 14,95% no R_{POCID} em comparação também com o modelo individual LSTM para a base de dados 7. Ademais, o teste de hipótese de Diebold-Mariano sobre os erros quadráticos revelou a relevância do LocLN em relação aos demais 18 métodos concorrentes nas 10 bases de dados, ao indicar 103 vitórias, 75 empates e 2 derrotas entre 180 comparações. Estes resultados demonstraram que em 98,8% dos casos comparados, o LocLN é superior ou igual aos métodos concorrentes que possuem abordagens individuais, de MHPs e de MHSs encontradas na literatura.

Além disso, os métodos com o modelo ELM foram executados 100 vezes com sementes diferentes nas bases de dados horárias. Os resultados da proposta LocLN foram promissores, e indicaram significância estatística no teste de hipóteses de Diebold-Mariano, quando comparado com a abordagem ELM individual. Com relação ao tempo computacional, o LocLN exige o treinamento de 120 modelos (60 ARIMA e 60 ELM). Apesar disso, o tempo de treinamento de cerca de 20 minutos em um processador Core i7 foi considerado adequado, pois há tempo suficiente para gerar previsões de um passo à frente para horizontes de 1h. E esse tempo é menor que o treinamento de uma única rede neural recorrente, por exemplo, os modelos individuais LSTM e GRU.

Para as geradoras de energia eólica esses resultados trazem benefícios, pois previsões mais precisas para geração de energia reduzem riscos de ajustes de última hora, evitando compras emergenciais a preços elevados. Também possibilita uma gestão mais eficiente dos aerogeradores para uma manutenção preditiva mais adequada, e um prolongamento da vida útil das turbinas. Por fim, empresas com planejamentos mais seguros e estáveis de previsões atraem mais investimentos.

6.2 Questões da Pesquisa

Nesta seção são apresentadas algumas considerações sobre as questões levantadas nesta pesquisa de acordo com os resultados descritos no capítulo anterior.

6.2.1 Um método que realiza um mapeamento linear e não linear de forma global é suficiente para reconhecer os padrões locais complexos em séries temporais de velocidade do vento?

Não. Os resultados mostram que o mapeamento local de alguns MHPs superou o mapeamento global de alguns MHSs nas séries temporais de velocidade do vento. Por exemplo, nas bases de dados 3 e 4, o $LocPart_{ARIMA}$ superou todos os métodos com correção de erros no RMSE, MAE e MAPE. Isso demonstrou que a aplicação puramente da abordagem de MHSs de forma global não é suficiente para reconhecer os padrões locais complexos em séries temporais de velocidade do vento.

6.2.2 Variar o tamanho e a região dos subconjuntos no conjunto de treinamento em um MHP permitirá um reconhecimento de padrões locais mais apropriado do que abordagens que empregam um mapeamento global do respectivo modelo base em séries temporais de velocidade do vento?

Sim. Os resultados do RMSE nas bases de dados 1, 2, 3 e 6 mostram a superioridade do mapeamento local do método LocPart com os modelos base ARIMA, ELM, LSTM e GRU em relação ao mapeamento global dos respectivos modelos individuais e do método *bagging*.

6.2.3 A aplicação de uma etapa de seleção para encontrar os preditores mais acurados faz sentido para MHPs aplicados em séries temporais de velocidade do vento?

Sim. Foi verificado que a etapa de seleção tem um papel importante, pois selecionou um subconjunto de preditores mais capacitados. Isso possibilitou obter resultados com previsões mais eficientes e com menores erros em relação ao conjunto completo de preditores gerados.

6.2.4 Ao comparar MHPs com MHSs, qual das duas abordagens é a melhor para previsão de séries temporais de velocidade do vento?

Não há uma abordagem melhor para todos os casos. Por exemplo, nas bases de dados 3 e 4, o MHP $LocPart_{ARIMA}$ superou todos os MHSs no RMSE, MAE e MAPE. E nas bases de dados 1 e 5, o MHS LC_{AM} superou todos os MHPs no RMSE, MAE e MAPE. Estes resultados indicam que o desempenho preditivo de cada abordagem depende da série temporal. Há algumas séries de velocidade do vento que aceitam melhor o mapeamento de padrões dos MHSs, já outras, o reconhecimento de padrões locais dos MHPs se destaca.

6.2.5 A aplicação de um reconhecimento local linear na série temporal, juntamente com um mapeamento local não linear nos resíduos permitirá um reconhecimento mais apropriado dos padrões locais complexos em séries temporais de velocidade do vento?

Sim. Os resultados do teste de hipótese de Diebold-Mariano sobre o RMSE demonstraram a relevância do LocLN em relação aos 18 métodos concorrentes nas 10 bases de dados de velocidade do vento. Em 98,8% dos casos comparados (180 casos), o LocLN é superior ou igual aos métodos concorrentes que possuem abordagens individuais, de MHPs e de MHSs encontradas na literatura.

6.2.6 O tamanho da série temporal influencia no desempenho preditivo da proposta LocLN?

Sim. Os resultados do método LocLN foram mais promissores nas bases de dados horárias, que são maiores em relação às bases de dados com intervalos temporais de 3h. Pois, nas séries horárias foi possível gerar 120 modelos base (60 ARIMA e 60 ELM) para o reconhecimento dos padrões locais. Já nas séries com intervalos temporais de 3h, foram gerados no máximo 32 modelos base (20 ARIMA e 12 ELM). Isso possibilitou uma maior busca por padrões locais nas séries horárias (menor granularidade) em relação às séries com intervalos de 3h (maior granularidade e mais suavizadas).

6.3 Contribuições

Os resultados obtidos por meio da proposta tornam válida a proposição de que a aplicação de um reconhecimento local linear na série temporal, em conjunto com um mapeamento local não linear nos resíduos permitirá um mapeamento mais apropriado dos padrões locais complexos em séries temporais de velocidade do vento. Ademais, pode-se especificar as seguintes contribuições desta Tese de Doutorado:

- Desenvolvimento do MHP LocPart (ALMEIDA; de MATTOS NETO; CUNHA, 2024) para o reconhecimento de padrões locais em séries temporais de velocidade do vento. O LocPart é um MHP homogêneo com uma inovação na etapa de geração, em que são criados subconjuntos variáveis em relação ao tamanho e a região no conjunto de treinamento;

- O método LocPart gera automaticamente os preditores mais competentes e os combina de acordo com sete entradas informadas pelo usuário: a quantidade inicial I de subconjuntos, o incremento L na quantidade de subconjuntos em cada *loop*, a quantidade máxima J de subconjuntos no *loop*, o modelo base M e sua técnica G de busca para os hiperparâmetros, uma medida de avaliação V e o método de combinação C para os modelos base;

- Desenvolvimento do método híbrido LocLinNonPR (ALMEIDA, 2024) para combinar um MHP com um MHS. O LocLinNonPR é um método híbrido, e sua inovação é a combinação entre um MHP, que possui um mapeamento local linear em subconjuntos de diferentes regiões de tamanhos iguais, e um MHS com mapeamento global não linear;

- O método híbrido LocLinNonPR gera automaticamente os preditores lineares em subconjuntos de tamanhos iguais distribuídos sequencialmente no conjunto de treinamento, os combina e emprega uma técnica MHS por meio de um modelo não linear. São necessárias sete entradas informadas pelo usuário: a quantidade I de subconjuntos, o modelo base linear M_L e sua técnica G de busca para os hiperparâmetros, o modelo base não linear M_N e sua técnica G de busca para os hiperparâmetros, o modelo base não linear combinador M_N e sua técnica G de busca para os hiperparâmetros;

- Desenvolvimento do método híbrido LocLN para empilhar o MHP homogêneo LocPart em uma técnica de MHS. O LocLN é um método híbrido, e a sua inovação é o emprego do mapeamento local do método LocPart na série de forma linear, além da aplicação do LocPart nos resíduos de forma não linear;

- O método híbrido LocLN gera automaticamente os preditores lineares mais competentes e os combina por meio de um MHP LocPart linear, também gera automaticamente os preditores não lineares mais competentes e os combina por meio de um MHP LocPart não linear, e por fim realiza a combinação do LocPart linear com o LocPart não linear selecionando o método de combinação entre dois tipos (linear ADD ou não linear M_C). Todo esse processo envolve os parâmetros de entrada do LocPart linear, LocPart não linear, o modelo base não linear combinador M_N e sua técnica G de busca para os hiperparâmetros, o combinador linear ADD, e a métrica V para seleção do método de combinação.

- As propostas são genéricas e podem ser utilizadas com qualquer modelo base linear para o reconhecimento de padrões locais lineares. Também pode ser utilizado qualquer modelo não linear para o mapeamento de padrões locais não lineares.

6.4 Dificuldades Encontradas

Uma das grandes dificuldades encontradas neste trabalho foi a ausência de banco de dados brasileiros recentes de velocidade do vento apropriados para geração de energia eólica. Gravações de dados das empresas responsáveis pelos parques eólicos brasileiros são altamente confidenciais, e por esse motivo, não há um compartilhamento com a comunidade científica. Contudo, o INPE fornece algumas bases de dados adequadas para pesquisas em energia eólica, o ponto negativo é que são antigas. Já o INMET fornece bases de dados atuais, todavia, não são convenientes para pesquisas em energia eólica, visto que grande parte das bases possui ventos com velocidades baixas. Isto torna qualquer trabalho que envolva dados de velocidade do vento no Brasil um grande desafio, visto que os autores da pesquisa devem utilizar banco de dados não recentes do INPE, ou fazer uma vasta seleção nas bases de dados do INMET, em busca de estações de medição com velocidades do vento aceitáveis para a geração de energia eólica.

6.5 Trabalhos Futuros

Seria interessante investigar a escalabilidade das propostas para séries temporais ainda maiores, pois a quantidade de modelos base gerados no LocPart e no LocLN aumenta com o tamanho do conjunto de treinamento da série temporal. Durante o desenvolvimento da pesquisa, a seleção dinâmica foi avaliada, mas não apresentou bons resultados nas séries temporais de velocidade do vento utilizadas neste trabalho. Pois, os resultados da seleção estática foram melhores. Contudo, é preciso fazer um estudo mais profundo sobre a possibilidade de se aplicar métodos de seleção dinâmica nos MHPs da proposta. Ademais, o desenvolvimento de um ajuste automático dos parâmetros de entrada para a criação dos subconjuntos do LocPart, em vez de ser definido pelo usuário. Também a aplicação de métodos inteligentes de estimação de hiperparâmetros para os modelos base em vez da técnica de busca em grade utilizada neste trabalho. Além disso, empregar outros tipos de modelos base nas propostas, inclusive modelos mais modernos. Por fim, as propostas apresentaram resultados promissores para séries temporais de velocidade do vento, que são consideradas séries com padrões complexos. Por esse motivo, as propostas poderiam ser avaliadas em outras aplicações que possuam séries complexas, inclusive em outras energias renováveis, por exemplo, séries temporais de irradiação solar.

REFERÊNCIAS

ABADI, M.; AGARWAL, A.; BARHAM, P.; BREVDO, E.; CHEN, Z.; CITRO, C.; CORRADO, G. S.; DAVIS, A.; DEAN, J.; DEVIN, M.; GHEMAWAT, S.; GOODFELLOW, I.; HARP, A.; IRVING, G.; ISARD, M.; JIA, Y.; JOZEFOWICZ, R.; KAISER, L.; KUDLUR, M.; LEVENBERG, J.; MANé, D.; MONGA, R.; MOORE, S.; MURRAY, D.; OLAH, C.; SCHUSTER, M.; SHLENS, J.; STEINER, B.; SUTSKEVER, I.; TALWAR, K.; TUCKER, P.; VANHOUCKE, V.; VASUDEVAN, V.; VIéGAS, F.; VINYALS, O.; WARDEN, P.; WATTENBERG, M.; WICKE, M.; YU, Y.; ZHENG, X. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. 2015. Software available from tensorflow.org. Disponível em: <https://www.tensorflow.org/>.

ABDOOS, A. A. A new intelligent method based on combination of vmd and elm for short term wind power forecasting. *Neurocomputing*, Elsevier, v. 203, p. 111–120, 2016.

ADHIKARI, R.; VERMA, G.; KHANDELWAL, I. A model ranking based selective ensemble approach for time series forecasting. *Procedia Computer Science*, Elsevier, v. 48, p. 14–21, 2015.

AGGARWAL, C. C. *Neural networks and deep learning: a textbook*. [S.l.]: Springer, 2018.

AHMADI, M.; KHASHEI, M. Current status of hybrid structures in wind forecasting. *Engineering Applications of Artificial Intelligence*, Elsevier, v. 99, p. 104133, 2021.

ALENCAR, D. B.; AFFONSO, C. M.; OLIVEIRA, R. C.; FILHO, C. J. Hybrid approach combining sarima and neural networks for multi-step ahead wind speed forecasting in brazil. *IEEE Access*, IEEE, v. 6, p. 55986–55994, 2018.

ALENCAR, D. B.; AFFONSO, C. M.; OLIVEIRA, R. C.; FILHO, C. J. Hybrid approach combining sarima and neural networks for multi-step ahead wind speed forecasting in brazil. *IEEE Access*, IEEE, v. 6, p. 55986–55994, 2018.

ALEXIADIS, M.; DOKOPOULOS, P.; SAHSAMANOGLOU, H.; MANOUSARIDIS, I. Short-term forecasting of wind speed and related electrical power. *Solar Energy*, Elsevier, v. 63, n. 1, p. 61–68, 1998.

ALMEIDA, D. M. A proposal for a fusion of an ensemble with error correction model applied to automotive time series. In: IEEE. *2024 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2024. p. 1–7.

ALMEIDA, D. M.; de MATTOS NETO, P. S. d.; CUNHA, D. C. A new ensemble with partition size variation applied to wind speed time series. In: SPRINGER. *International Conference on Hybrid Artificial Intelligence Systems*. [S.l.], 2024. p. 53–65.

ALMEIDA, D. M.; de MATTOS NETO, P. S. G.; CUNHA, D. C. Hybrid time series forecasting models applied to automotive on-board diagnostics systems. In: *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.: s.n.], 2018. p. 1–8.

ARAUJO, P.; MARINHO, M. Analysis of hydro-wind complementarity in state of pernambuco, brazil by means of weibull parameters. *IEEE Latin America Transactions*, IEEE, v. 17, n. 04, p. 556–563, 2019.

BARBALATA, C.; LEUSTEAN, L. *Average monthly liquid flow forecasting using neural networks*. Bucharest, Romania, 2004.

BARROS, M. *Processos Estocásticos*. 1. ed. Rio de Janeiro: Papel Virtual, 2004.

BERGMEIR, C.; BENíTEZ, J. M. *Neural Networks using the Stuttgart Neural Network Simulator (SNNS)*. [S.l.], 2017. User Manual Package 'RSNNS' for R.

BERGMEIR, C.; HYNDMAN, R. J.; BENÍTEZ, J. M. Bagging exponential smoothing methods using stl decomposition and box–cox transformation. *International journal of forecasting*, Elsevier, v. 32, n. 2, p. 303–312, 2016.

BOX, G. E. P.; JENKINS, G. M.; REINSEL, G. C. *Time series analysis, forecasting and control*. 4. ed. San Francisco: John Wiley & Sons, 2008.

BROCKWELL, P. J.; BROCKWELL, P. J.; DAVIS, R. A.; DAVIS, R. A. *Introduction to time series and forecasting*. [S.l.]: Springer, 2016.

BROCKWELL, P. J.; DAVIS, R. A. *Introduction to time series and forecasting*. New York: Springer, 2002. v. 2.

CADENAS, E.; RIVERA, W. Wind speed forecasting in the south coast of oaxaca, mexico. *Renewable energy*, Elsevier, v. 32, n. 12, p. 2116–2128, 2007.

CADENAS, E.; RIVERA, W. Wind speed forecasting in three different regions of mexico, using a hybrid arima–ann model. *Renewable Energy*, Elsevier, v. 35, n. 12, p. 2732–2738, 2010.

CAMELO, H. do N.; LUCIO, P. S.; JUNIOR, J. B. V. L.; CARVALHO, P. C. M. de. A hybrid model based on time series models and neural network for forecasting wind speed in the brazilian northeast region. *Sustainable Energy Technologies and Assessments*, Elsevier, v. 28, p. 65–72, 2018.

CAMELO, H. do N.; LUCIO, P. S.; JUNIOR, J. B. V. L.; CARVALHO, P. C. M. de; SANTOS, D. v. G. dos. Innovative hybrid models for forecasting time series applied in wind generation based on the combination of time series models with artificial neural networks. *Energy*, Elsevier, v. 151, p. 347–357, 2018.

CANG, S.; YU, H. A combination selection algorithm on forecasting. *European Journal of Operational Research*, Elsevier, v. 234, n. 1, p. 127–139, 2014.

CERQUEIRA, V.; TORGO, L.; SOARES, C. A case study comparing machine learning with statistical methods for time series forecasting: size matters. *Journal of Intelligent Information Systems*, Springer, v. 59, n. 2, p. 415–433, 2022.

CHE, J. Optimal sub-models selection algorithm for combination forecasting model. *Neurocomputing*, Elsevier, v. 151, p. 364–375, 2015.

CHOLLET, F. et al. *Keras*. 2015. <https://keras.io>.

CLEVELAND, R. B.; CLEVELAND, W. S.; MCRAE, J. E.; TERPENNING, I. Stl: A seasonal-trend decomposition procedure based on loess. *Journal of Official Statistics*, v. 6, n. 1, p. 3–73, 1990.

COWPERTWAIT, P. S. P.; METCALFE, A. V. *Introductory Time Series with R*. 1. ed. New York: Springer, 2009.

DAVID, R.; DAVID, S. M. *Statistics and data analysis for financial engineering: with R examples*. [S.l.]: Spinger, 2015.

de MATTOS NETO, P. S.; CAVALCANTI, G. D.; FIRMINO, P. R.; SILVA, E. G.; FILHO, S. R. V. N. A temporal-window framework for modelling and forecasting time series. *Knowledge-Based Systems*, Elsevier, v. 193, p. 105476, 2020.

de MATTOS NETO, P. S.; OLIVEIRA, J. F. L. de; JÚNIOR, D. S. d. O. S.; SIQUEIRA, H. V.; MARINHO, M. H. D. N.; MADEIRO, F. A hybrid nonlinear combination system for monthly wind speed forecasting. *IEEE Access*, IEEE, v. 8, p. 191365–191377, 2020.

de MATTOS NETO, P. S. G.; CAVALCANTI, G. D. C.; MADEIRO, F. Nonlinear combination method of forecasters applied to pm time series. *Pattern Recognit. Lett.*, v. 95, p. 65–72, 2017.

de MATTOS NETO, P. S. G.; OLIVEIRA, J. F. L. de; JúNIOR, D. S. de O. S.; SIQUEIRA, H. V.; MARINHO, M. H. D. N.; MADEIRO, F. A hybrid nonlinear combination system for monthly wind speed forecasting. *IEEE Access*, v. 8, p. 191365–191377, 2020.

DENNIS, C.; ENGELBRECHT, A. P.; OMBUKI-BERMAN, B. M. An analysis of activation function saturation in particle swarm optimization trained neural networks. *Neural Processing Letters*, Springer, v. 52, p. 1123–1153, 2020.

DIEBOLD, F. X.; MARIANO, R. S. Comparing predictive accuracy. *Journal of Business & Economic Statistics*, v. 13, n. 3, 1995.

ELSWORTH, S.; GÜTTEL, S. Time series forecasting using lstm networks: A symbolic approach. *arXiv preprint arXiv:2003.05672*, 2020.

ERDEM, E.; SHI, J. Arma based approaches for forecasting the tuple of wind speed and direction. *Applied Energy*, Elsevier, v. 88, n. 4, p. 1405–1414, 2011.

EóLICA, A. B. de E. Abeeólica | boletim anual de dados 2023. 2023.

EóLICA, A. B. de E. Abeeólica | infovento. *INFOVENTO 31, 15 de junho de 2023*, 2023.

EóLICA, A. B. de E. Abeeólica | infovento. *INFOVENTO 35, 02 de outubro de 2024*, 2024.

FERREIRA, M.; SANTOS, A.; LUCIO, P. Short-term forecast of wind speed through mathematical models. *Energy Reports*, Elsevier, v. 5, p. 1172–1184, 2019.

FLORES, P.; TAPIA, A.; TAPIA, G. Application of a control algorithm for wind speed prediction and active power generation. *Renewable Energy*, Elsevier, v. 30, n. 4, p. 523–536, 2005.

FORSYTH, D. *Probability and statistics for computer science*. [S.l.]: Springer, 2018. v. 13.

GÉRON, A. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. [S.l.]: "O'Reilly Media, Inc.", 2017.

GUO, S. H. Y.; SHEN, C.; LI, X. Y. Y.; BAI, Y. An adaptive svr for high-frequency stock price forecasting. *IEEE Access*, v. 6, p. 11397–11404, 2018.

GWEC, G. W. E. C. Gwec| global wind report 2023. *Global Wind Energy Council: Brussels, Belgium*, 2023.

HARVEY, D.; LEYBOURNE, S.; NEWBOLD, P. Testing the equality of prediction mean squared errors. *International Journal of forecasting*, Elsevier, v. 13, n. 2, p. 281–291, 1997.

HARVEY, D. I.; LEYBOURNE, S. J.; NEWBOLD, P. Tests for forecast encompassing. *Journal of Business & Economic Statistics*, Taylor & Francis, v. 16, n. 2, p. 254–259, 1998.

HASSANI, H.; SILVA, E. S. A kolmogorov-smirnov based test for comparing the predictive accuracy of two sets of forecasts. *Econometrics*, Multidisciplinary Digital Publishing Institute, v. 3, n. 3, p. 590–609, 2015.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural computation*, MIT press, v. 9, n. 8, p. 1735–1780, 1997.

HSU, C.; CHANG, C.; LIN, C. *A practical guide to support vector classification*. Taiwan, 2016.

HU, J.; WANG, J.; ZENG, G. A hybrid forecasting approach applied to wind speed time series. *Renewable Energy*, Elsevier, v. 60, p. 185–194, 2013.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: theory and applications. *Neurocomputing*, Elsevier, v. 70, n. 1-3, p. 489–501, 2006.

HUANG, X.; WANG, J.; HUANG, B. Two novel hybrid linear and nonlinear models for wind speed forecasting. *Energy Conversion and Management*, Elsevier, v. 238, p. 114162, 2021.

HUANG, X.; WANG, J.; HUANG, B. Two novel hybrid linear and nonlinear models for wind speed forecasting. *Energy Conversion and Management*, Elsevier, v. 238, p. 114162, 2021.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: principles and practice*. Melbourne, Australia: OTexts, 2013.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: principles and practice*. 3. ed. [S.l.]: OTexts, 2021.

HYNDMAN, R. J.; O'HARA-WILD, M.; BERGMEIR, C.; RAZBASH, S.; WANG, E. *Forecasting Functions for Time Series and Linear Models*. [S.l.], 2017. User Manual Package 'forecast' for R.

INFOESCOLA. *Mapa do Brasil*. 2025. Acessado em: 27-01-2025. Disponível em: <https://www.infoescola.com/geografia/mapa-do-brasil/>.

INPE. *Rede do Sistema de Organização Nacional de Dados Ambientais*. 2020. Acessado em: 27-07-2023. Disponível em: <http://sonda.ccst.inpe.br/index.html>.

JIANG, P.; WANG, B.; LI, H.; LU, H. Modeling for chaotic time series based on linear and nonlinear framework: Application to wind speed forecasting. *Energy*, Elsevier, v. 173, p. 468–482, 2019.

JIANG, Z.; CHE, J.; WANG, L. Ultra-short-term wind speed forecasting based on emd-var model and spatial correlation. *Energy Conversion and Management*, Elsevier, v. 250, p. 114919, 2021.

JUNG, J.; BROADWATER, R. P. Current status and future advances for wind speed and power forecasting. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 31, p. 762–777, 2014.

KAMAL, L.; JAFRI, Y. Z. Time series models to simulate and forecast hourly averaged wind speed in quetta, pakistan. *Solar Energy*, Elsevier, v. 61, n. 1, p. 23–32, 1997.

KAVASSERI, R. G.; SEETHARAMAN, K. Day-ahead wind speed forecasting using f-arima models. *Renewable Energy*, Elsevier, v. 34, n. 5, p. 1388–1393, 2009.

KHASHEI, M.; BIJARI, M. A novel hybridization of artificial neural networks and arima models for time series forecasting. *Applied Soft Computing*, v. 11, p. 2664–2675, 2011.

KONG, X.; LIU, X.; SHI, R.; LEE, K. Y. Wind speed prediction using reduced support vector machines with feature selection. *Neurocomputing*, Elsevier, v. 169, p. 449–456, 2015.

KWIATKOWSKI, D.; PHILLIPS, P.; SCHMIDT, P.; SHIN, Y. How sure are we that economic time series have a unit root. *Journal of Econometrics*, v. 54, n. 1992, p. 159–178, 1992.

LI, B.; ZHANG, J.; HE, Y.; WANG, Y. Short-term load-forecasting method based on wavelet decomposition with second-order gray neural network model combined with adf test. *IEEE Access*, IEEE, v. 5, p. 16324–16331, 2017.

LI, H.; WANG, J.; LU, H.; GUO, Z. Research and application of a combined model based on variable weight for short term wind speed forecasting. *Renewable Energy*, Elsevier, v. 116, p. 669–684, 2018.

LIMA, D. R. S. *Turbinas Eólicas ou Aerogeradores*. 2021. Acessado em: 10-09-2021. Disponível em: <https://oakenergia.com.br/blog/turbinas-eolicas/>.

LIU, X.; LIN, Z.; FENG, Z. Short-term offshore wind speed forecast by seasonal arima-a comparison against gru and lstm. *Energy*, Elsevier, v. 227, p. 120492, 2021.

MA, Z.; CHEN, H.; WANG, J.; YANG, X.; YAN, R.; JIA, J.; XU, W. Application of hybrid model based on double decomposition, error correction and deep learning in short-term wind speed prediction. *Energy Conversion and Management*, Elsevier, v. 205, p. 112345, 2020.

MA, Z.; GUO, S.; XU, G.; AZIZ, S. Meta learning-based hybrid ensemble approach for short-term wind speed forecasting. *IEEE Access*, IEEE, v. 8, p. 172859–172868, 2020.

MCCRACKEN, M. W.; WEST, K. D. Inference about predictive ability. *A companion to economic forecasting*, Basil Blackwell Oxford, p. 299–321, 2002.

MEMARZADEH, G.; KEYNIA, F. A new short-term wind speed forecasting method based on fine-tuned lstm neural network and optimal input sets. *Energy Conversion and Management*, Elsevier, v. 213, p. 112824, 2020.

MEYER, D.; DIMITRIADOU, E.; HORNIK, K.; WEINGESSEL, A.; LEISCH, F. *Misc Functions of the Department of Statistics, Probability Theory Group*. [S.l.], 2017. User Manual Package 'e1071' for R.

MOHANDES, M. A.; HALAWANI, T. O.; REHMAN, S.; HUSSAIN, A. A. Support vector machines for wind speed prediction. *Renewable energy*, Elsevier, v. 29, n. 6, p. 939–947, 2004.

MOUSELIMIS, A. G. L.; JONGE, E. de. *elmNNRcpp: The Extreme Learning Machine Algorithm*. 2022. Acessado em: 08-08-2023. Disponível em: <https://CRAN.R-project.org/package=elmNNRcpp>.

NIKOLIĆ, V.; MOTAMEDI, S.; SHAMSHIRBAND, S.; PETKOVIĆ, D.; CH, S.; ARIF, M. Extreme learning machine approach for sensorless wind speed estimation. *Mechatronics*, Elsevier, v. 34, p. 78–83, 2016.

NIU, X.; WANG, J. A combined model based on data preprocessing strategy and multi-objective optimization algorithm for short-term wind speed forecasting. *Applied Energy*, Elsevier, v. 241, p. 519–539, 2019.

PETRIS, G.; PETRONE, S.; CAMPAGNOLI, P. *Dynamic linear models with R*. New York: Springer Science & Business Media, 2009.

PETROPOULOS, F.; HYNDMAN, R. J.; BERGMEIR, C. Exploring the sources of uncertainty: Why does bagging for time series forecasting work? *European Journal of Operational Research*, Elsevier, v. 268, n. 2, p. 545–554, 2018.

PHILLIPS, P. C.; PERRON, P. Testing for a unit root in time series regression. *Biometrika*, Oxford University Press, v. 75, n. 2, p. 335–346, 1988.

QU, Z.; MAO, W.; ZHANG, K.; ZHANG, W.; LI, Z. Multi-step wind speed forecasting based on a hybrid decomposition technique and an improved back-propagation neural network. *Renewable Energy*, Elsevier, v. 133, p. 919–929, 2019.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2024. Disponível em: <https://www.R-project.org/>.

REDEMET. *Rede de Meteorologia do Comando da Aeronáutica*. 2020. Acessado em: 29-02-2024. Disponível em: <https://www.redemet.aer.mil.br/old/index.php?i=sistemas&p=webmet>.

RIEDMILLER, M.; BRAUN, H. *A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm*. Karlsruhe, Germany, 1993.

RUIZ-AGUILAR, J. J.; TURIAS, I.; GONZÁLEZ-ENRIQUE, J.; URDA, D.; ELIZONDO, D. A permutation entropy-based emd–ann forecasting ensemble approach for wind speed prediction. *Neural Computing and Applications*, Springer, v. 33, n. 7, p. 2369–2391, 2021.

SAAVEDRA-MORENO, B.; SALCEDO-SANZ, S.; CARRO-CALVO, L.; GASCÓN-MORENO, J.; JIMÉNEZ-FERNÁNDEZ, S.; PRIETO, L. Very fast training neural-computation techniques for real measure-correlate-predict wind operations in wind farms. *Journal of Wind Engineering and Industrial Aerodynamics*, Elsevier, v. 116, p. 49–60, 2013.

SALCEDO-SANZ, S.; ORTIZ-GARCI, E. G.; PÉREZ-BELLIDO, Á. M.; PORTILLA-FIGUERAS, A.; PRIETO, L. et al. Short term wind speed prediction based on evolutionary support vector regression algorithms. *Expert Systems with Applications*, Elsevier, v. 38, n. 4, p. 4052–4057, 2011.

SANTOS JÚNIOR, D. S. d. O.; de MATTOS NETO, P. S.; OLIVEIRA, J. F. de; CAVALCANTI, G. D. A hybrid system based on ensemble learning to model residuals for time series forecasting. *Information Sciences*, Elsevier, v. 649, p. 119614, 2023.

SERGIO, A. T. Seleção dinâmica de combinadores de previsão de séries temporais. *Tese de Doutorado - Ciência da computação*, Universidade Federal de Pernambuco, 2017.

SERGIO, A. T.; LIMA, T. P. de; LUDERMIR, T. B. Dynamic selection of forecast combiners. *Neurocomputing*, Elsevier, v. 218, p. 37–50, 2016.

SFETSOS, A. A novel approach for the forecasting of mean hourly wind speed time series. *Renewable energy*, Elsevier, v. 27, n. 2, p. 163–174, 2002.

SILVA, G. R. *Características de Vento da Região Nordeste: análise, modelagem e aplicações para projetos de centrais eólicas*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2003.

SILVA, G. R. Características de vento na região nordeste. *Dissertação de Mestrado - Engenharia mecâmica*, Universidade Federal de Pernambuco, 2003.

SMOLA, A. J.; SCHÖLKOPF, B. A tutorial on support vector regression. *Statistics and computing*, Springer, v. 14, n. 3, p. 199–222, 2004.

SONG, X.; LIU, Y.; XUE, L.; WANG, J.; ZHANG, J.; WANG, J.; JIANG, L.; CHENG, Z. Time-series well performance prediction based on long short-term memory (lstm) neural network model. *Journal of Petroleum Science and Engineering*, Elsevier, v. 186, p. 106682, 2020.

SOUALHI, A.; MEDJAHER, K.; ZERHOUNI, N. Bearing health monitoring based on hilbert–huang transform, support vector machine, and regression. *IEEE Transactions on Instrumentation and Measurement*, v. 64, n. 1, p. 52–62, 2015.

TANG, Z.; ZHAO, G.; WANG, G.; OUYANG, T. Hybrid ensemble framework for short-term wind speed forecasting. *IEEE Access*, IEEE, v. 8, p. 45271–45291, 2020.

TORRES, J. L.; GARCIA, A.; BLAS, M. D.; FRANCISCO, A. D. Forecast of hourly average wind speed with arma models in navarre (spain). *Solar energy*, Elsevier, v. 79, n. 1, p. 65–77, 2005.

VERGARA, J. R.; ESTÉVEZ, P. A. A review of feature selection methods based on mutual information. *Neural computing and applications*, Springer, v. 24, n. 1, p. 175–186, 2014.

WANG, H.; HU, D. *Comparison of SVM and LS-SVM for Regression*. Shanghai, China, 2005.

WANG, J.; ZHANG, W.; LI, Y.; WANG, J.; DANG, Z. Forecasting wind speed using empirical mode decomposition and elman neural network. *Applied soft computing*, Elsevier, v. 23, p. 452–459, 2014.

WANG, J.; ZHANG, W.; WANG, J.; HAN, T.; KONG, L. A novel hybrid approach for wind speed prediction. *Information Sciences*, Elsevier, v. 273, p. 304–318, 2014.

WANG, L.; LI, X.; BAI, Y. Short-term wind speed prediction using an extreme learning machine model with error correction. *Energy Conversion and Management*, Elsevier, v. 162, p. 239–250, 2018.

WU, S.; WANG, Y.; CHENG, S. Extreme learning machine based wind speed estimation and sensorless control for wind turbine power generation system. *Neurocomputing*, Elsevier, v. 102, p. 163–175, 2013.

YAN, Y.; WANG, X.; REN, F.; SHAO, Z.; TIAN, C. Wind speed prediction using a hybrid model of eemd and lstm considering seasonal features. *Energy Reports*, Elsevier, v. 8, p. 8965–8980, 2022.

ZELL, A.; MACHE, N.; HUBNER, R.; MAMIER, G.; VOGT, M.; SCHMALZL, M.; HERRMANN, K. *Stuttgart Neural Network Simulator (SNNS)*. Germany, 2008. v. 4. User Manual SNNS.

ZHANG, G. P. Time series forecasting using a hybrid arima and neural network model. *Neurocomputing*, v. 50, p. 159–175, 2003.

ZHANG, G. P. *Neural Networks in Business Forecasting*. Hershey: IRM Press, 2004.

ZHU, B.; WEI, Y. Carbon price forecasting with a novel hybrid arima and least squares support vector machines methodology. *Omega*, v. 41, p. 517–524, 2013.

ZHU, T.; LUO, L.; ZHANG, X.; SHI, Y.; SHEN, W. Time-series approaches for forecasting the number of hospital daily discharged inpatients. *IEEE J. Biomed. Health Inform.*, v. 21, n. 2, p. 515–526, 2017.