

Universidade Federal Fluminense
Escola de Engenharia
Curso de Graduação em Engenharia Elétrica

Bernardo Albuquerque Domingues da Silva

**Estimação da Geração Hidrotermoeólica Utilizando
Redes Neurais e Variáveis do Fenômeno El Niño**

Niterói

2025

Bernardo Albuquerque Domingues da Silva

Estimação da Geração Hidrotermoeólica Utilizando Redes Neurais e Variáveis do Fenômeno El Niño

Projeto de Conclusão de Curso apresentado ao Corpo Docente do Departamento de Engenharia Elétrica da Escola de Engenharia da Universidade Federal Fluminense, como parte dos requisitos necessários à obtenção do título de Engenheiro Eletricista.

Orientador:
Prof. André da Costa Pinho, D.Sc.

Niterói

2025

Ficha Catalográfica elaborada pelo Aluno, acesse o site:

<https://bibliotecas.uff.br/bcg/fichacatalografica/>

Gere o pdf da ficha e substitua o arquivo **ficha.pdf** com a sua ficha.

As informações abaixo são ilustrativas. Os alunos devem inserir a Ficha Catalográfica nesta página.

M514 Tal, Fulano de

Título do Trabalho / Fulano de Tal. -Rio de Janeiro, RJ: [s.n.],
2022.

xx f.

Trabalho de Conclusão de Curso de Graduação em
Engenharia Civil – Universidade Federal Fluminense, 2022.

1. Construção Civil. 2. Lean Construction. 3. Gerenciamento
de Obra

Bernardo Albuquerque Domingues da Silva

Estimação da Geração Hidrotermoeólica Utilizando Redes Neurais e Variáveis do Fenômeno El Niño

Projeto de Conclusão de Curso apresentado
ao Corpo Docente do Departamento de
Engenharia Elétrica da Escola de Engenharia
da Universidade Federal Fluminense, como
parte dos requisitos necessários à obtenção
do título de Engenheiro Eletricista.

Aprovado em: _____ de _____ de _____

BANCA EXAMINADORA

Prof. André da Costa Pinho, D.Sc. - UFF

Prof. Bruno Soares Moreira Cesar Borba, D.Sc. - UFF

Prof. Marcio Andre Ribeiro Guimaraens, D.Sc. - UFF

Em memória de minha querida avó Célia, por toda
a sua paciência, carinho e amor incondicionais.

Agradecimentos

Agradeço aos meus pais, Débora e Gilvan, por sempre me apoiarem nas minhas escolhas, terem me ajudado a superar os obstáculos que surgiram no caminho e sempre estarem presentes nos momentos mais importantes. Obrigado por acreditarem em mim, por terem abdicado de tantas coisas para me proporcionar uma educação de qualidade e me ensinado a importância da ética, esforço, estudo e trabalho. Serei eternamente grato pelos esforços e sacrifícios que fizeram por mim. Vocês sempre serão os meus maiores exemplos na vida.

À minha irmã, Letícia, por ser meu alívio cômico por todos esses anos. À minha família, por compreender minhas ausências e por todos os aprendizados que me proporcionaram.

À minha companheira, Juliana, por ter me apoiado, incentivado e compreendido durante essa trajetória. Por ter me ajudado a manter a calma e acreditar que eu era capaz de superar qualquer obstáculo. Obrigado por ser a minha maior incentivadora e por nunca me deixar desistir nos momentos de dificuldade. Seu amor e paciência foram essenciais para que eu pudesse concluir este trabalho.

Aos amigos que fiz durante a graduação, que sem dúvidas espero levar para a vida toda. Sem vocês o caminho teria sido muito mais difícil. Agradeço por todos os momentos de descontração, pelas risadas, pelos estudos em grupo e, principalmente, pelo revezamento de faltas nas disciplinas mais chatas. À Faraday E-Racing, que representou um marco na minha trajetória acadêmica e me proporcionou aprendizados e oportunidades essenciais para a minha formação.

Ao professor André Pinho, pela sua excelência, ética, maestria em ensinar e por ter me orientado de maneira exemplar durante o desenvolvimento deste projeto. Agradeço também aos professores Flávio Martins, Felipe Sass e Marcio Guimaraens, por me lembrarem em cada aula do motivo pelo qual escolhi a Engenharia Elétrica.

Resumo

A matriz energética brasileira é caracterizada por uma dependência significativa de fontes renováveis, especialmente a geração hidrelétrica. Essa dependência torna o sistema elétrico vulnerável a variações climáticas, como secas prolongadas que podem ser intensificadas por fenômenos como o El Niño e La Niña, afetando a disponibilidade de água nos reservatórios e, consequentemente, a geração de energia. Com o crescimento da fonte eólica, também vulnerável a variações climáticas, é necessário investigar o impacto dessas variáveis na operação do Sistema Interligado Nacional (SIN), com ênfase nas fontes hidráulica, térmica e eólica. Este trabalho analisa séries históricas disponibilizadas pelo Operador Nacional do Sistema Elétrico (ONS) e variáveis que definem o fenômeno El Niño, obtidas do ERA5, para avaliar os efeitos desses fenômenos na geração de energia elétrica. Modelos de regressão e aprendizado de máquina são aplicados para analisar o impacto dessas variáveis na geração de energia.

Palavras-chave: Geração de energia. Clima. Planejamento energético. Machine learning.

Abstract

The Brazilian electrical system is characterized by a significant dependence on renewable sources, especially hydropower generation. This dependence makes the electric system vulnerable to climatic variations, such as prolonged droughts and phenomena such as El Niño (EN) and La Niña (LN), which can affect the availability of water in reservoirs and, consequently, energy generation. With the adoption of wind power, also vulnerable to climatic variations, it is necessary to investigate the impact of climatic variables on the operation of the National Interconnected System (SIN), with an emphasis on hydraulic, wind and thermal sources. This work analyzes historical series provided by the National Electric System Operator (ONS) and variables that define the El Niño phenomenon, obtained from the European Centre for Medium-Range Weather Forecasts (ECMWF) Reanalysis project (ERA5), to evaluate the effects of these phenomena on electricity generation. Regression and machine learning models are applied to analyze the impact of these variables on energy generation.

Key-words: Energy generation. Climate. Risk mitigation. Machine learning.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

ACR	Ambiente de Contratação Regulada
API	Aplication Programming Interface (Interface de Programação de Aplicações)
CDS	Climate Data Store
CEPEL	Centro de Pesquisas de Energia Elétrica
CPAMP	Comissão Permanente para Análise de Metodologias e Programas Computacionais do Setor Elétrico
CSV	Comma Separated Values (Valores Separados por Vírgula)
EN	El-Niño
ENSO	El Niño-Oscilação Sul
EPE	Empresa de Pesquisa Energética
GPU	Graphics Processing Unit (Unidade de Processamento Gráfico)
LN	La-Niña
LOESS	Local Polynomial Regression Fitting (Ajuste de Regressão Polinomial Local)
MLP	Multi Layer Perceptrons
MSE	Erro Quadrático Médio
ONI	Oceanic Niño Index
ONS	Operador Nacional do Sistema Elétrico
PDDE	Programação Dinâmica Dual Estocástica
PDE	Plano Decenal de Expansão de Energia
PEE	Parques Eólicos Equivalentes
PEN	Plano de Operação Energética
PLD	Preço de Liquidação das Diferenças

PMO	Programa Mensal de Operação
R ²	Coeficiente de Determinação
REE	Reservatórios Equivalentes de Energia
SIN	Sistema Interligado Nacional
SST	Sea Surface Temperature (Temperatura da Superfície do Mar)
TTM	Tiny Time Mixer

Sumário

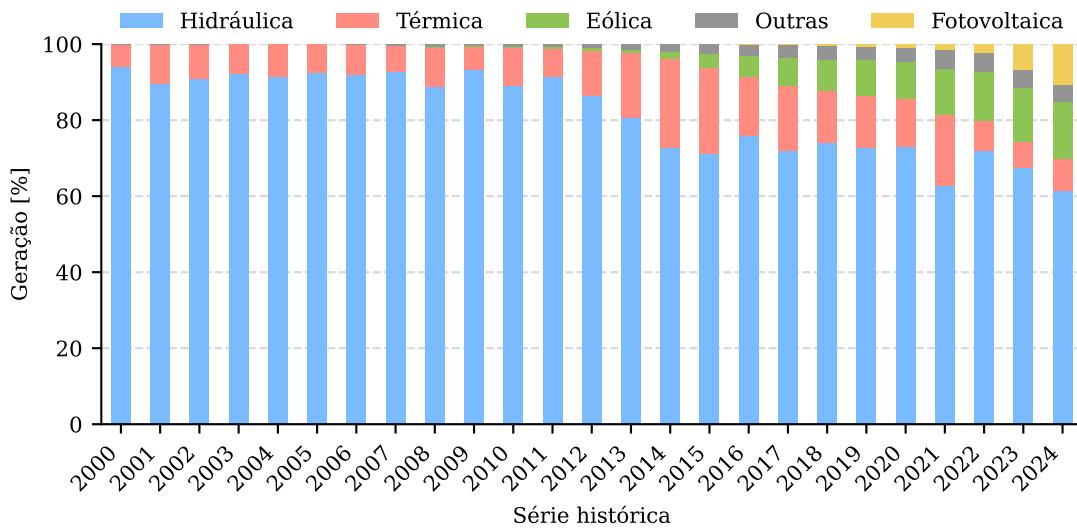
Capítulo 1

Introdução

1.1 Contexto

Historicamente, a matriz elétrica brasileira é considerada uma das mais limpas do mundo, com destaque para a fonte hidráulica, que é responsável pela maior parte da geração de energia elétrica no país. Nos últimos anos, outras fontes de geração vêm sendo incorporadas ao sistema, das quais destacam-se a eólica e solar fotovoltaica, conforme observado na Figura ??, elaborada a partir de dados brutos de geração centralizada obtidos do Operador Nacional do Sistema Elétrico (ONS), sem considerar a geração distribuída.

Figura 1 - Geração centralizada anual por fonte



Fonte: o autor.

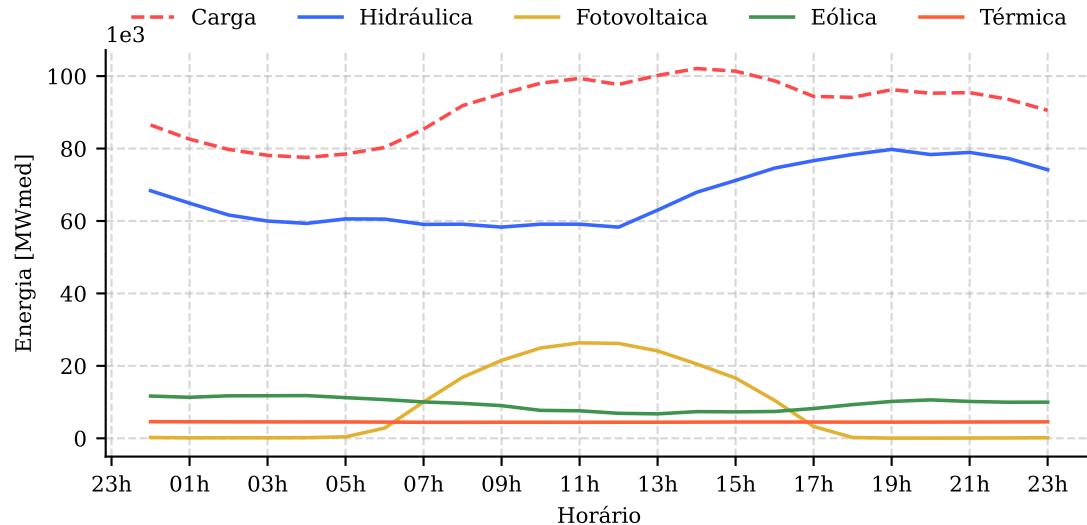
Nota-se, em especial, um crescimento significativo da geração eólica, observado a partir de 2015, e uma diminuição significativa da contribuição de geração térmica média no panorama geral nos anos seguintes. Em 2023, a fonte eólica foi responsável por 48% da expansão da capacidade instalada total de 10,19 GW (??). Essa expansão se dá em função do maior número de empreendimentos participantes nos Leilões de Energia Elétrica do Ambiente de Contratação Regulada (ACR) realizados pela Empresa de Pesquisa Energética (EPE). Isso ocorre, dentre outros fatores, devido à queda nos custos de aerogeradores e

painéis fotovoltaicos, além do fator "combustível zero" dessas fontes, o que torna novos empreendimentos mais atrativos economicamente para os agentes.

Embora essa expansão seja positiva, poupando recursos hídricos, contribuindo para a diversificação da matriz elétrica e reduzindo o acionamento de usinas térmicas, essas fontes possuem características intrínsecas que as tornam intermitentes, como a incidência solar e a velocidade do vento. Sendo assim, uma alta dependência dessas fontes tem o potencial de tornar o sistema como um todo mais vulnerável.

Além disso, ao analisar a curva de carga do SIN, observa-se que, embora o seu pico ocorra no início da tarde, momento no qual a geração solar fotovoltaica apresenta significativa contribuição, o período noturno também apresenta carga considerável, conforme a Figura ??, que mostra a curva de carga do SIN para o dia 15 de março de 2024, dia em que registrou-se um recorde de demanda máxima instantânea de 102.478 MW, segundo o ONS.

Figura 2 - Curva de carga diária do SIN em base horária



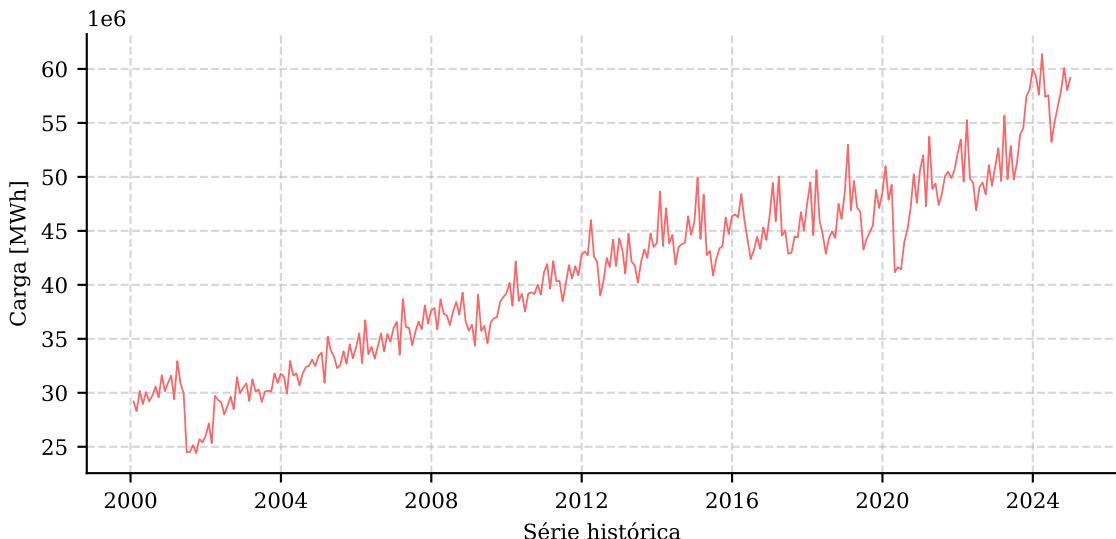
Fonte: o autor.

1.2 Motivação

Em um contexto no qual a implementação de sistemas de armazenamento de energia elétrica ainda é incipiente, a matriz segue bastante dependente da fonte hidráulica e, de maneira complementar, das térmicas. A dependência da fonte hidráulica, por sua vez, torna o sistema elétrico vulnerável a eventos climáticos extremos ocasionados pelas mudanças

climáticas. Por exemplo, em 2021, verificou-se um acionamento recorde de usinas térmicas e uma geração hidráulica percentual mínima. Isso se deve em razão da forte crise hídrica enfrentada pelo Brasil no período, a pior dos últimos 91 anos até então. (??)

Figura 3 – Curva de carga do SIN em base mensal



Fonte: o autor.

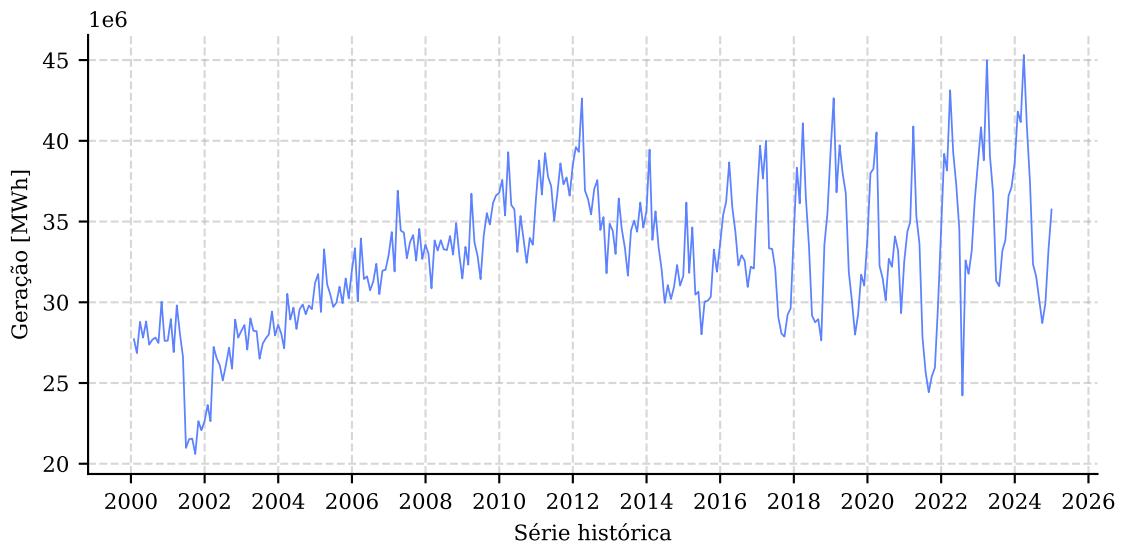
Portanto, o estudo do sistema elétrico brasileiro, no contexto de cenários de eventos climatológicos extremos é altamente relevante para a segurança energética do país, considerando a tendência crescente na curva de carga do SIN, ilustrada na Figura ??, e uma projeção de crescimento médio anual de 3,2%. (??)

Ao analisar a geração hidráulica bruta na Figura ??, evidenciam-se pontos nos quais a geração é reduzida. Isso ocorre devido à sazonalidade das vazões nas bacias hidrográficas, responsáveis pelo abastecimento dos reservatórios. Considerando a amostragem em base mensal, observa-se que a geração é reduzida nos meses de inverno, período caracterizado por menor ocorrência de precipitação e, consequentemente, menor vazão nos rios. Por outro lado, nos meses de verão, a geração atinge seus maiores valores.

Esse comportamento é esperado, uma vez que a geração hidráulica é diretamente influenciada pelas condições que afetam a vazão dos rios. No entanto, a ocorrência de eventos climáticos como o El Niño-Oscilação Sul (ENSO) pode favorecer condições que impactam diretamente no potencial de geração hidráulica. (??)

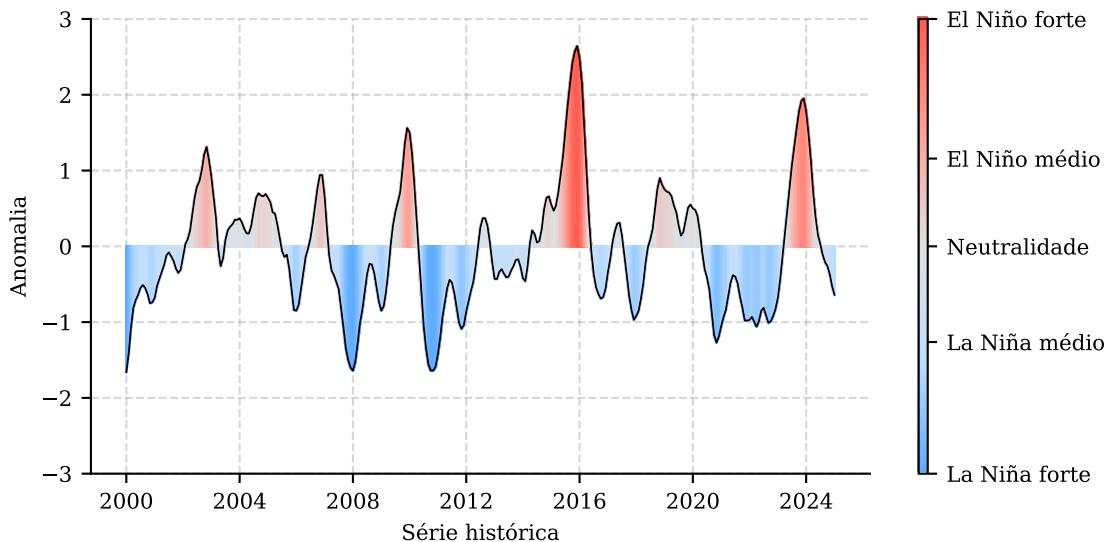
Fenômenos como o ENSO são monitoradas por meio de índices como o ONI (Oceanic Niño Index), que classifica os eventos em três categorias: *El-Niño* (EN), *La-Niña* (LN) e neutro. A Figura ?? mostra a classificação dos eventos de EN e LN ocorridos entre 2000 e

Figura 4 – Geração hidráulica total em base mensal



Fonte: o autor.

Figura 5 – Índice ONI (Oceanic Niño Index)



Fonte: o autor.

2024, na qual a escala de cores indica a intensidade do evento, representado pela anomalia, que é a diferença do valor observado para um período em relação à sua média histórica.

Fundamentalmente, no sistema elétrico brasileiro, cuja fonte hidráulica constitui a base da matriz, é essencial, para um planejamento energético eficiente, otimizar o sistema de modo a considerar a incerteza associada à disponibilidade de recursos hidrológicos futuros. Dessa forma, estima-se o valor da geração hidrelétrica que poderia substituir a geração térmica a curto ou longo prazo, de modo a reduzir os custos do sistema e o risco de utilizar reservatórios de maneira desnecessária, garantindo assim o atendimento à demanda

futura, principalmente em casos de escassez hídrica.

Com a introdução de fontes como a eólica, a incerteza referente aos perfis de velocidade do vento também deve ser considerada para o planejamento energético. Os estudos de planejamento são realizados por meio de modelos computacionais como o NEWAVE, DECOMP e DESSEM, do Centro de Pesquisas de Energia Elétrica (CEPEL), que consideram diferentes horizontes temporais: longo, médio e curto prazos, respectivamente. Também há outras soluções disponíveis no mercado, como o PSR SDDP, que engloba todos os horizontes temporais. Considerando o escopo deste trabalho, o modelo NEWAVE será brevemente apresentado no capítulo seguinte.

Embora esses modelos sejam amplamente utilizados pelo setor elétrico brasileiro, sendo consolidados como ferramentas confiáveis e essenciais para o planejamento energético, eles não consideram variáveis externas, como fenômenos climáticos como o EN e LN, que podem impactar a geração de energia elétrica. Esses modelos são baseados em dados históricos de vazões e velocidade do vento, que são obtidos a partir de medições locais, além de dados individuais de cada usina.

1.3 Objetivo

Este trabalho tem como objetivo investigar o impacto de variáveis climáticas externas na geração de energia elétrica no Brasil, com foco nas fontes hidráulica, térmica e eólica. Para tanto, foram empregadas técnicas computacionais para relacionar as séries históricas de geração com as séries de variáveis climáticas, através de modelos lineares, não lineares e neurais.

Além disso, foram utilizadas séries históricas de geração, carga e vazões disponibilizados pelo ONS, bem como séries históricas de variáveis climáticas, como temperatura da superfície do mar, obtidas a partir de dados do ERA5, um projeto de reanálise atmosférica proposto por ??), que combina resultados de modelos computacionais com observações de diferentes fontes, como satélites e estações meteorológicas, resultando em um *dataset* global de alta resolução espacial e temporal.

A partir dessa investigação, espera-se poder avaliar o impacto dessas variáveis na geração de energia elétrica, o que pode contribuir para o planejamento energético do país, especialmente em cenários de eventos climáticos extremos e tendências climáticas.

É importante salientar que outras variáveis externas poderiam ser incorporadas

ao estudo, ou até mesmo uma combinação entre variáveis locais e externas. Também poderiam ser considerados indicadores econômicos e outros dados relevantes. No entanto, este trabalho considera apenas as variáveis relacionadas ao fenômeno EN e LN, uma vez que estudos indicam uma alta correlação entre esses fenômenos e o regime de chuvas no Brasil (????), sendo a incorporação de outras variáveis reservada para trabalhos futuros.

1.4 Estrutura do Trabalho

No capítulo 1, é feita uma breve introdução apresentando o contexto, motivação, objetivo e a estrutura do trabalho. Uma breve análise da matriz elétrica é apresentada, com foco no histórico recente e no crescimento da geração eólica. Também são apresentadas as curvas de carga do SIN e de geração hidráulica. Uma breve contextualização acerca dos fenômenos do ENSO e a importância do modelo NEWAVE para o planejamento energético do país são apresentadas. Por fim, é apresentado o objetivo do trabalho e a estrutura do documento. O capítulo 2 apresenta um panorama acerca de trabalhos relacionados ao tema.

No capítulo 3, a fundamentação teórica necessária para a compreensão do projeto é apresentada. São abordados os fatores que fazem com que os fenômenos do EN e LN impactem o regime de chuvas no Brasil e a sua relação com a geração de energia elétrica. Também será feita uma breve introdução ao modelo NEWAVE, que é amplamente utilizado pelo setor elétrico brasileiro para planejamento energético. Por fim, serão apresentados os modelos de previsão de séries temporais implementados, com foco no modelo neural, uma implementação baseada na arquitetura *TSMixer* desenvolvida pela Google.

O capítulo 4 mostra a metodologia utilizada para a realização do projeto. Serão apresentados os conjuntos de dados considerados e suas respectivas etapas de obtenção, tratamento e análise. Além disso, será apresentada a metodologia utilizada para implementação dos modelos de previsão, incluindo seus parâmetros e métricas de avaliação. *Snippets* de códigos serão apresentados para facilitar a compreensão.

O capítulo 5 apresenta os resultados do projeto. Para cada modelo, serão apresentados os resultados de previsão, métricas de avaliação e uma análise crítica dos resultados obtidos. No capítulo 6, serão apresentadas as considerações finais.

Capítulo 2

Trabalhos Relacionados

Os impactos do fenômeno ENSO vêm sendo estudados extensamente em diversas áreas. Segundo ??), eventos de El Niño e La Niña influenciam significativamente a agricultura brasileira, especialmente nas regiões Sul e Nordeste.

Um estudo de ??) demonstrou que a ocorrência do fenômeno El Niño está associada a uma maior taxa de casos de dengue no estado de São Paulo, devido a um aumento na temperatura e precipitação, condições favoráveis para a proliferação do mosquito Aedes aegypti.

Segundo ??), o fenômeno ENSO tem uma relação direta sobre a disponibilidade de recursos hídricos nas bacias hidrográficas brasileiras, o que sugere um impacto direto sobre a geração hidrelétrica no Brasil. Além disso, os resultados são coerentes com a literatura, indicando impactos distintos nas diferentes regiões do país.

Naturalmente, o fenômeno ENSO também vem sendo estudado no contexto do setor elétrico brasileiro. Segundo um estudo da ??), a ocorrência do fenômeno La Niña em 2021 foi um fator determinante para a crise hídrica que afetou o Brasil nesse período, em decorrência da redução das afluências, ou seja, a quantidade de água que chega aos reservatórios das usinas hidrelétricas.

De acordo com ??), o recurso associado à fonte eólica é diretamente afetado por variáveis climáticas, como densidade local do ar, precipitação, temperatura e cobertura de nuvens. Nesse sentido, o trabalho apresenta um sistema de previsão estatística para a produção de energia eólica de curto prazo, destacando a necessidade de adaptabilidade para lidar com as relações não lineares entre as variáveis climáticas e a geração eólica.

Segundo o relatório da Comissão Permanente para Análise de Metodologias e Programas Computacionais do Setor Elétrico (CPAMP), constituída por instituições do setor elétrico como EPE, ONS e CEPEL, a incorporação de dados de variáveis climáticas, como o fenômeno ENSO, aos modelos computacionais é ativamente discutida devido a relação entre as séries históricas de vazões e dos ciclos de índices climáticos. ??)

De acordo com ??), o uso de modelos de aprendizado de máquina para previsão de carga do SIN tem o potencial de aprimorar o resultado das previsões, reduzindo os

desvios de previsão de carga e, consequentemente, uma redução significativa dos custos de operação do sistema elétrico. Infere-se, portanto, que essa abordagem também poderia ser aplicada para estimar outros parâmetros, como a geração futura.

Para isso, é essencial selecionar modelos que sejam capazes de capturar as relações potencialmente complexas entre os dados de geração e os fenômenos climáticos. Considerando que a literatura sugere que essa relação seja altamente não-linear, modelos neurais seriam uma escolha natural, mas não necessariamente os modelos neurais mais avançados seriam os mais adequados.

Segundo ??), modelos de previsão de séries temporais baseados na arquitetura *Transformer*, introduzida por ??), podem produzir resultados inferiores quando comparados a modelos mais simples. Nesse contexto, surge então a arquitetura *TSMixer*, proposta por ??). Essa arquitetura, embora mais simples, produz resultados consideravelmente superiores com uma fração do custo computacional.

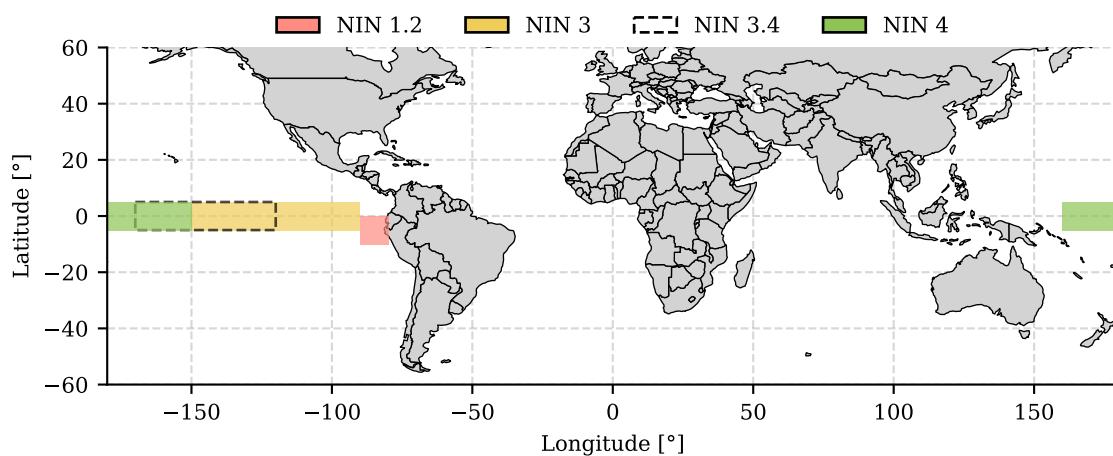
Capítulo 3

Fundamentação Teórica

3.1 Impacto do ENSO na Geração de Energia Elétrica

O ENSO é um fenômeno que ocorre no Oceano Pacífico Equatorial, caracterizado por variações na temperatura da superfície do mar (SST) em regiões específicas, como ilustrado na Figura ???. O fenômeno é um dos principais fatores que influenciam os padrões de vento e precipitação em diversas regiões da América do Sul e seus efeitos se estendem por todas as regiões do Brasil. (??)

Figura 6 - Regiões do fenômeno El Niño-Oscilação Sul (ENSO)



Fonte: o autor.

Os impactos em cada região estão resumidos a seguir, de acordo com ???:

- **Sul:** a região Sul é uma das mais consistentemente afetadas. Eventos de El Niño tendem a causar precipitação acima da média, particularmente durante a primavera e o verão, enquanto eventos de La Niña estão associados à condições de seca.
- **Sudeste:** a região Sudeste apresenta uma resposta mais complexa, sendo considerada uma zona de transição. A bacia do Rio Paraná, em especial, apresenta sensibilidade aos fenômenos do ENSO, tendo apresentado tendência de aumento de vazão durante alguns eventos de El Niño.
- **Norte/Nordeste:** para as regiões Norte e Nordeste, eventos de El Niño estão associados a períodos de seca, enquanto eventos de La Niña tendem a trazer

chuvas acima da média. No entanto, é importante destacar que outros fenômenos atmosféricos podem interferir com esses padrões, modulando os efeitos do ENSO.

Sendo assim, verifica-se que as variações induzidas pelos fenômenos do ENSO traduzem-se diretamente em variações nas vazões dos rios que alimentam as bacias, que por sua vez impactam o potencial de geração da fonte hidráulica.

3.2 O Modelo NEWAVE

Desenvolvido e mantido pelo Centro de Pesquisas de Energia Elétrica (CEPEL) e amplamente utilizado pelo setor elétrico brasileiro para definição de estratégias e tomada de decisão, o NEWAVE é um modelo de otimização que busca minimizar os custos de operação do sistema, considerando a incerteza das afluências futuras e a operação de um sistema hidro-térmico-eólico interligado. O modelo é utilizado para estudos como:

- Elaboração do Plano Decenal de Expansão de Energia (PDE), pela EPE;
- Elaboração do Programa Mensal de Operação (PMO) e Plano de Operação Energética (PEN), pelo ONS;
- Formação de preços, como no cálculo do Preço de Liquidação das Diferenças (PLD) pelo CCEE;
- Cálculo de Garantia Física e da Energia Assegurada para empreendimentos de geração participantes nos leilões de energia elétrica, pela EPE;
- Elaboração de diretrizes para os leilões de energia, pela EPE.

Em resumo, o modelo emprega a Programação Dinâmica Dual Estocástica (PDDE), uma técnica de otimização que permite lidar com as incertezas ligadas às afluências futuras sem que o modelo se torne computacionalmente impraticável, considerando múltiplos reservatórios, interconexões e o horizonte temporal de médio e longo prazos.

3.2.1 Representação das Usinas

O NEWAVE modela o sistema de geração hidrelétrico em Reservatórios Equivalentes de Energia (REEs), que são grupos de usinas associadas a um subsistema ou submercado de energia. Cada subsistema pode conter mais de um REE, possibilitando diferenciar bacias hidrográficas com regimes distintos, ainda que pertençam a um mesmo subsistema.

Além disso, cada REE é definido por um conjunto de parâmetros que são calculados a partir das características individuais de cada usina. Nas versões mais recentes do modelo, também é possível considerar todas as usinas individualmente ou operar de maneira híbrida, ou seja, considerando alguns REEs e outras usinas individualmente.

As usinas termelétricas são representadas no modelo através de classes térmicas. Cada classe agrupa usinas com custos semelhantes e está associada a um subsistema. Cada classe também é definida por um conjunto de parâmetros calculados a partir das características individuais de cada usina.

Nas versões mais recentes do modelo, a fonte eólica também é modelada. De maneira similar, os parques eólicos são agrupados em Parques Eólicos Equivalentes (PEE). O agrupamento é feito a partir de dados de cadastro de cada parque eólico, estado, submercado, função de produção (curva relacionando a velocidade do vento com a potência gerada), dados sobre torres de medição e séries históricas de velocidade do vento.

3.2.2 Dados de Entrada

O modelo requer um conjunto de dados de entrada que inclui as características das usinas, dados dos subsistemas, demanda, séries históricas de vazões e ventos, cronogramas de expansão, restrições operativas, dentre outros. Observa-se que todos os dados de entrada são locais e, portanto, o modelo não considera variáveis externas, como fenômenos climáticos como o EN e LN, que podem impactar a geração de energia elétrica.

Ainda que as últimas versões do modelo apresentem campos previstos para a entrada de dados do ENSO, esses campos estão marcados como "não implementados". Dessa forma, entende-se que o modelo não considera diretamente o impacto dessas variáveis. No entanto, vale destacar que essas variáveis externas podem ser utilizadas para elaborar as séries históricas de vazões e velocidade de ventos utilizadas como dados de entrada.

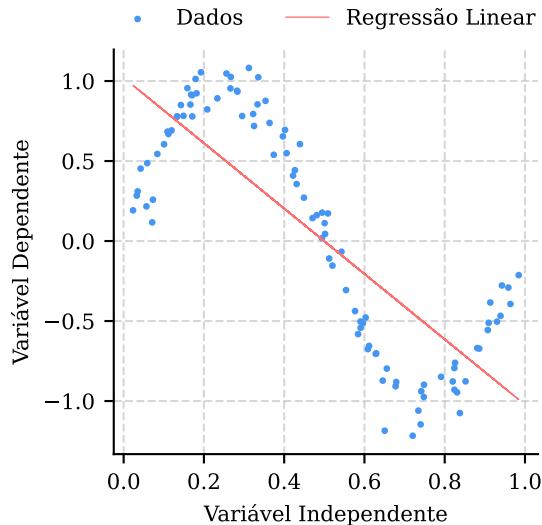
3.3 Modelos Linear e Não-linear

Neste trabalho, foram utilizados modelos de regressão linear e não linear a fim de se obter uma *baseline* para comparação com o modelo neural. Para ambos os casos, o `scikit-learn`, uma biblioteca Python que contém diversos algoritmos para análise de dados, incluindo modelos de classificação e regressão, foi utilizada. Esta biblioteca é construída utilizando as bibliotecas `NumPy` e `SciPy`, que fornecem suporte para operações matemáticas e científicas, amplamente utilizadas na comunidade de ciência de dados e aprendizado de máquina.

3.3.1 Modelo Linear

O modelo linear utilizado foi o `LinearRegression` da biblioteca `scikit-learn`. Ele consiste na aplicação do método dos mínimos quadrados para determinar os coeficientes da equação linear que melhor se ajusta aos dados, conforme ilustrado na Figura ??.

Figura 7 – Regressão linear de uma senoide



Fonte: o autor.

Naturalmente, esse tipo de modelo é utilizado quando se espera que a variável de interesse (dependente) seja uma combinação linear das variáveis de entrada (independentes), conforme a equação ??,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n \quad (3.1)$$

em que y é a variável de interesse, β_0 é o intercepto, $\beta_1, \beta_2, \dots, \beta_n$ são os coeficientes e x_1, x_2, \dots, x_n são as variáveis de entrada.

O método de mínimos quadrados busca minimizar a soma dos quadrados dos resíduos, ou seja, a diferença entre os valores observados e os valores previstos pelo modelo. Matematicamente, isso é representado pela equação ??,

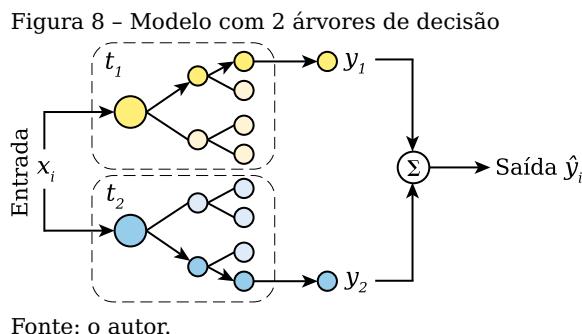
$$\min_{\beta} \sum_{i=1}^m (y_i - (\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_n x_{in}))^2 \quad (3.2)$$

em que m é o número de observações, y_i é o valor observado da variável de interesse para a i -ésima observação, x_{ij} é o valor da j -ésima variável de entrada para a i -ésima observação e β é o vetor de coeficientes do modelo.

Ainda que se espere que as relações entre os dados seja altamente não linear, o método é capaz de capturar algumas das relações lineares entre as variáveis, além de servir como ponto de partida para as próximas análises.

3.3.2 Modelo Não-linear

O modelo de regressão não linear utilizado foi o `RandomForestRegressor` da biblioteca `scikit-learn`. O modelo funciona a partir de árvores de decisão construídas a partir dos dados de entrada. Cada árvore é treinada em um *subset* aleatório dos dados e então a árvore cresce por meio da bifurcação. A estrutura do modelo é ilustrada na Figura ??.



Durante a bifurcação, o modelo estabelece o melhor caminho através da combinação de subconjuntos das variáveis de entrada, de forma a minimizar o erro na previsão da variável de interesse. A previsão final é obtida através da média das previsões de todas as árvores, o que ajuda a reduzir o *overfitting* e melhora a generalização do modelo. A saída do modelo é dada por:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^T y_t(x) \quad (3.3)$$

em que T é o número de árvores, $y_t(x)$ é a previsão da t -ésima árvore e \hat{y} é a previsão final do modelo.

3.4 Modelo Neural

Antes de apresentar o modelo neural, será feita uma breve introdução a respeito de alguns conceitos importantes, como o *Perceptron* e os *Multi Layer Perceptrons* (MLPs). Esses conceitos são fundamentais para compreender a arquitetura do modelo utilizado.

3.4.1 O Perceptron

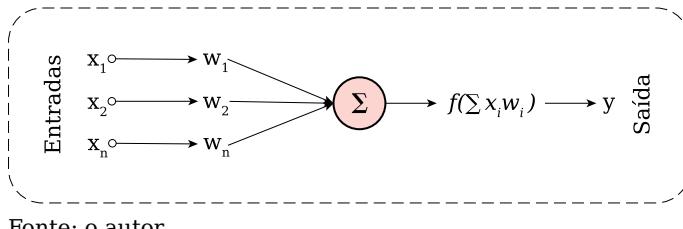
O *Perceptron* é um modelo de rede neural artificial proposto por ??), inspirado no funcionamento de neurônios biológicos. A figura ?? ilustra a estrutura básica de um Perceptron.

Um perceptron recebe um conjunto $X = [x_1, x_2, \dots, x_n]$ de entradas, com cada x_i associada a um peso aleatório w_i . O Perceptron então calcula a soma ponderada das entradas e aplica uma função de ativação para produzir a saída. Matematicamente, isso é dado por:

$$\hat{y}(X) = f\left(\sum_{i=1}^n w_i x_i + b\right) \quad (3.4)$$

em que \hat{y} é a saída do Perceptron e f a função de ativação que, para esse caso, usa-se a função degrau $u(t)$.

Figura 9 – Estrutura do Perceptron



Fonte: o autor.

O Perceptron é limitado a resolver problemas de classificação linearmente separáveis. Ou seja, problemas nos quais é possível traçar uma linha (ou hiperplano) que separe as classes de forma clara. Para essas aplicações, o Perceptron atualiza os pesos de modo iterativo durante o treinamento, através da taxa de aprendizado r , conforme a equação ??, que demonstra a atualização dos pesos no tempo:

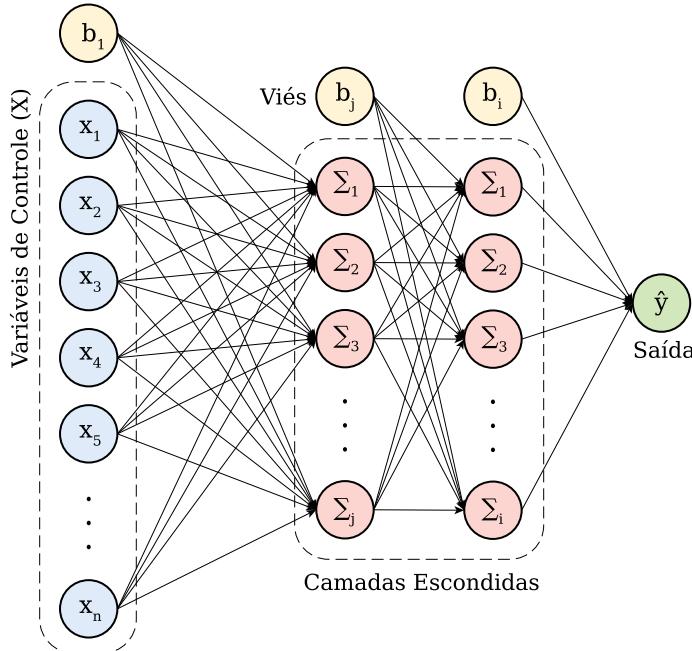
$$w_i(t+1) \leftarrow w_i(t) + r(y(t) - \hat{y}(t))x_i \quad (3.5)$$

em que y é o valor real e \hat{y} a saída do Perceptron. A taxa de aprendizado representa o quanto rápido os pesos são atualizados durante o treinamento. A atualização é feita de forma a minimizar o erro entre a saída prevista e a saída real.

3.4.2 Multi Layer Perceptrons (MLPs)

MLPs surgiram como uma evolução dos *Perceptrons* simples, com a finalidade de permitir a modelagem de relações não lineares, e são a base para o *deep learning*, metodologia que utiliza redes neurais com inúmeras camadas para resolução de tarefas complexas. A figura ?? ilustra a estrutura básica de um MLP de dupla camada.

Figura 10 – MLP de dupla camada



Fonte: o autor.

Essas estruturas são compostas por neurônios (perceptrons) interconectados e organizados em camadas. Cada neurônio de uma camada está conectado a todos os outros das camadas adjacentes (*fully connected*). A primeira e última camadas são chamadas de camada de entrada e camada de saída, respectivamente, enquanto as camadas intermediárias são chamadas de camadas ocultas.

MLPs são treinadas utilizando o algoritmo de *backpropagation*, ou retropropagação, que ajusta iterativamente os pesos e vieses da rede de forma a minimizar o erro entre a saída prevista e a saída real. O algoritmo funciona em um ciclo de quatro etapas, descritas abaixo.

1. **Forward Pass:** os dados de entrada são propagados pelos neurônios até a camada de saída. A saída de cada neurônio é calculada em duas fases. Primeiro, a entrada ponderada $z_i^{(l)}$ para o neurônio i da camada l é a soma das saídas da camada anterior, $v_j^{(l-1)}$, multiplicadas pelos seus respectivos pesos $w_{ij}^{(l)}$, mais um viés $b_i^{(l)}$:

$$z_i^{(l)} = \sum_j (w_{ij}^{(l)} v_j^{(l-1)}) + b_i^{(l)} \quad (3.6)$$

Em seguida, a saída ativada $v_i^{(l)}$ é obtida aplicando-se a função de ativação σ :

$$v_i^{(l)} = \sigma(z_i^{(l)}) \quad (3.7)$$

2. **Cálculo do Erro:** uma função de perda E é usada para quantificar o erro entre a saída \hat{y} da rede e o valor real y . Para tarefas de regressão, costuma-se usar o Erro Quadrático Médio (MSE):

$$E = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (3.8)$$

em que n é o número de neurônios na camada de saída.

3. **Backward Pass:** o erro E é propagado de volta pela rede até a primeira camada. Para isso, a regra da cadeia é utilizada para determinar o gradiente da função de perda em relação a cada peso e viés da rede. Para a camada L de saída, o erro $\delta_i^{(L)}$ é calculado diretamente para cada neurônio i :

$$\delta_i^{(L)} = \frac{\partial E}{\partial v_i^{(L)}} \cdot \sigma'(z_i^{(L)}) \quad (3.9)$$

em que $\frac{\partial E}{\partial v_i^{(L)}}$ é a derivada da perda em relação à saída do neurônio i e σ' é a derivada da função de ativação. Caso seja utilizado o MSE e sabendo que $v_i^{(L)} = \hat{y}_i$, a equação ?? torna-se:

$$\delta_i^{(L)} = (\hat{y}_i - y_i) \cdot \sigma'(z_i^{(L)}) \quad (3.10)$$

Para as camadas ocultas, o erro $\delta_i^{(l)}$ para um neurônio i na camada l é calculado recursivamente, com base nos erros da camada $l+1$ seguinte:

$$\delta_i^{(l)} = \left(\sum_j \delta_j^{(l+1)} w_{ji}^{(l+1)} \right) \cdot \sigma'(z_i^{(l)}) \quad (3.11)$$

em que a soma percorre todos os neurônios j da camada seguinte, ponderando seus erros $\delta_j^{(l+1)}$ pelos pesos $w_{ji}^{(l+1)}$ que os conectam ao neurônio i da camada atual.

Com o termo $\delta_i^{(l)}$ para cada neurônio, os gradientes dos pesos e vieses são encontrados:

$$\frac{\partial E}{\partial w_{ij}^{(l)}} = \delta_i^{(l)} v_j^{(l-1)} \quad (3.12)$$

$$\frac{\partial E}{\partial b_i^{(l)}} = \delta_i^{(l)} \quad (3.13)$$

em que $v_j^{(l-1)}$ é a saída do neurônio j da camada anterior $l - 1$.

4. Atualização dos Pesos: finalmente, cada peso e viés da rede é atualizado na direção oposta à do seu gradiente, de modo a reduzir o erro na próxima iteração:

$$w_{ij}^{(l)}(t + 1) \leftarrow w_{ij}^{(l)}(t) - r \frac{\partial E}{\partial w_{ij}^{(l)}} \quad (3.14)$$

$$b_i^{(l)}(t + 1) \leftarrow b_i^{(l)}(t) - r \frac{\partial E}{\partial b_i^{(l)}} \quad (3.15)$$

em que t representa a iteração atual e r é a taxa de aprendizado, que controla a magnitude da atualização.

Em resumo, o processo de aprendizado de um MLP envolve a propagação dos dados de entrada pela rede, o cálculo do erro entre a saída prevista e a real, a retropropagação do erro para calcular os gradientes e a atualização dos pesos e vieses para minimizar o erro. Esse processo é repetido até que a rede atinja um nível satisfatório de desempenho ou até que um critério de parada seja atingido, como um número máximo de iterações ou uma tolerância de erro.

3.4.3 Arquitetura TSMixer

A arquitetura proposta por ??) é uma abordagem inovadora para previsão de séries temporais. Ela utiliza MLPs em cascata, denominadas de *Mixing Layers*, para capturar as relações na dimensão temporal, bem como na dimensão das características (variáveis).

A mistura no domínio do tempo permite ao modelo capturar os padrões temporais da série. Essa abordagem se mostrou eficaz para aprender padrões temporais como sazonais e tendências, sem a necessidade de mecanismos de atenção, como os utilizados na arquitetura *Transformer*.

A mistura no domínio das características permite ao modelo capturar as relações entre as variáveis em cada instante de tempo. Ou seja, o modelo é capaz de aprender como

as variáveis interagem entre si ao longo da série temporal. Essa abordagem é eficaz para capturar correlações e dependências, sem a necessidade de mecanismos de atenção.

A arquitetura emprega as técnicas de *dropout* e resíduo durante o treinamento. O objetivo é evitar *overfitting* e melhorar a generalização do modelo. A cada passo de treinamento (época), o *dropout* desliga aleatoriamente um número de neurônios em uma camada. Dessa forma, a rede é "forçada" a não depender de um conjunto pequeno de neurônios, fazendo com que o aprendizado seja distribuído entre os demais neurônios da camada. Essa abordagem diminui a chance do modelo apenas memorizar os dados de treinamento (*overfitting*).

A técnica de resíduo é usada para permitir que a entrada de uma camada seja somada à sua saída. O objetivo é manter o processo de aprendizado eficiente, evitando o efeito de *vanishing gradient*, no qual o gradiente do erro se torna muito pequeno ao ser propagado às camadas iniciais pelo algoritmo de *backpropagation*. Esse efeito pode ocorrer em redes com muitas camadas quando as derivadas parciais da função de erro podem ser muito pequenas para as camadas iniciais.

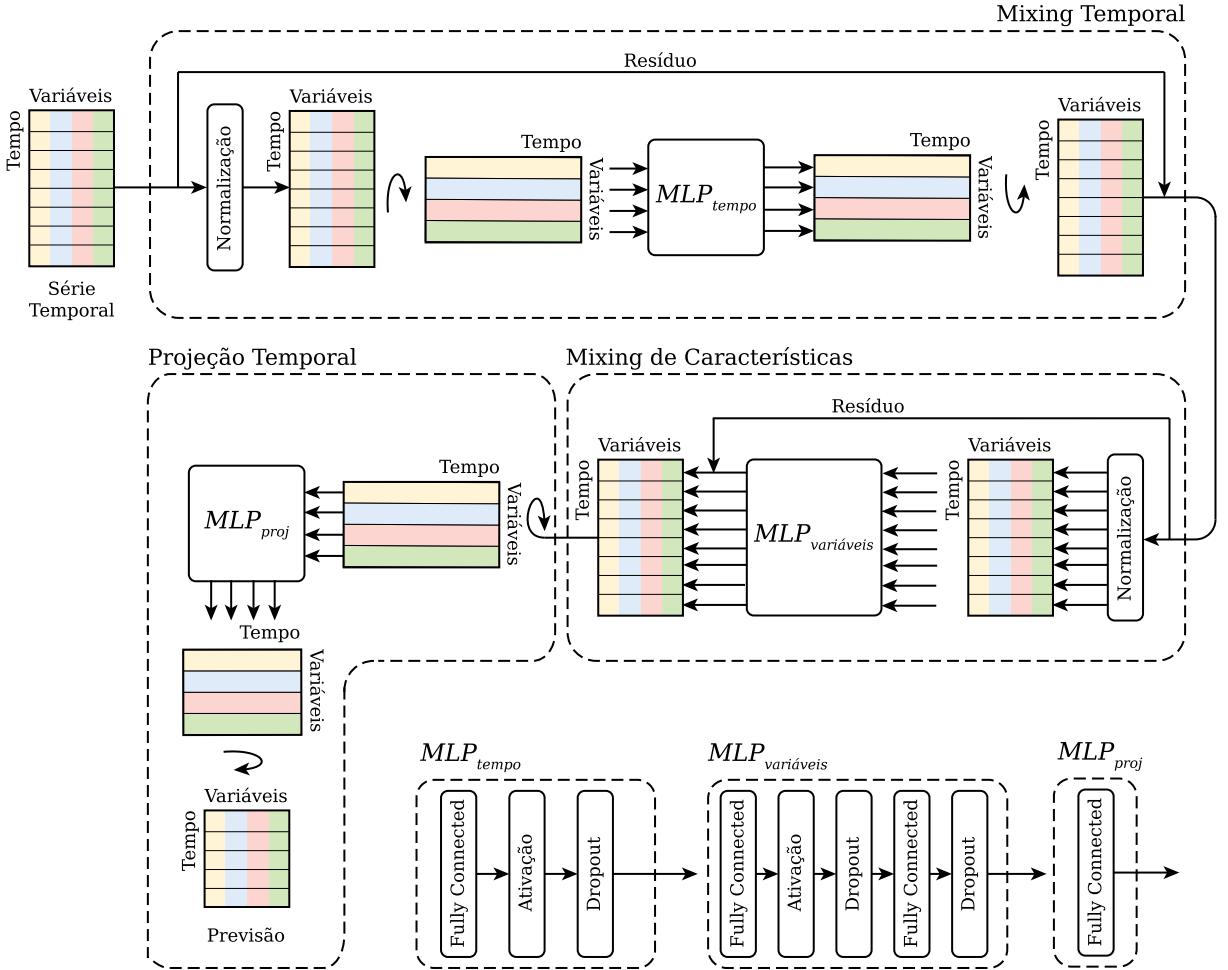
A figura ?? ilustra a arquitetura do modelo. As colunas das entradas representam diferentes variáveis e as linhas são os instantes de tempo. As operações de mistura são realizadas linha a linha.

As MLPs no domínio do tempo tem seus pesos compartilhados entre todas as variáveis, enquanto as MLPs no domínio das características tem seus pesos compartilhados entre todos os instantes de tempo. Para a projeção final, uma camada totalmente conectada é usada para mapear o tamanho da entrada (contexto) para o tamanho da saída (previsão). As etapas estão descritas a seguir:

- 1. Mixing Temporal:** modela padrões temporais da série. Constitui uma única camada de neurônios totalmente conectada (MLP de camada única), seguida de uma função de ativação e *dropout*. A entrada é transposta para aplicar as conexões no domínio do tempo, ou seja, as entradas $X_i = [x_i(t_1), x_i(t_2), \dots, x_i(t_n)]$ são cada variável em todos os instantes de tempo, permitindo que o modelo capture os padrões temporais de cada variável utilizando os mesmos pesos. A saída é transposta novamente para manter a forma original da entrada.

- 2. Mixing de Características:** modela relações entre as variáveis em cada instante de tempo. Constitui duas camadas totalmente conectadas (MLPs de camada dupla),

Figura 11 - Arquitetura do modelo neural usado



Fonte: o autor (adaptado de ??)).

também com uma função de ativação e *dropout*. As conexões são aplicadas no domínio das características, ou seja, as entradas $T_i = [t_i(x_1), t_i(x_2), \dots, t_i(x_n)]$ são todas as variáveis em um instante de tempo, permitindo ao modelo capturar as relações entre elas, também utilizando os mesmos pesos.

3. Projeção Temporal: projetam a saída, mapeando o tamanho da janela de entrada (contexto) para o tamanho da janela de saída (previsão). Constitui uma camada totalmente conectada aplicada no domínio do tempo.

??) também propõem uma variante (*TSMixer-Ext*) para uso de variáveis exógenas, que são variáveis auxiliares que podem explicar o comportamento da variável de interesse. Considerando que essa variante ainda não foi disponibilizada publicamente, para esse projeto foi utilizada uma implementação da arquitetura TSMixer com suporte a variáveis exógenas, disponibilizada por ??), denominada de *Tiny Time Mixer* (TTM).

3.4.3.1 Extensão para Variáveis Exógenas (TSMixer-Ext)

Essa variante da arquitetura é capaz de trabalhar com recursos estáticos, que não variam ao longo do tempo, como localização e características da usina; e também recursos variáveis no tempo, como índices climáticos. A TSMixer-Ext introduz os processos de Alinhamento e Mistura, antes de aplicar as camadas de mistura principais:

1. **Alinhamento:** o objetivo é fazer com que todas as entradas (históricas, futuras e estáticas) tenham o mesmo comprimento de sequência para que possam ser combinadas. Recursos estáticos, por exemplo, são repetidos ao longo da dimensão do tempo para se alinharem ao horizonte de previsão. Após o alinhamento, todas as variáveis são concatenadas em uma única matriz de características.
2. **Mistura:** a matriz de características combinada é então processada pelas mesmas Mixing Layers (temporal e de características) descritas anteriormente. Essa abordagem permite que o modelo aprenda não apenas os padrões da série principal, mas também as complexas interações e influências que as variáveis exógenas exercem sobre ela.

No contexto deste projeto, as variáveis exógenas são os dados referentes ao fenômeno ENSO, ou seja, os dados de SST.

Capítulo 4

Metodologia

4.1 Abordagem Computacional

O projeto foi desenvolvido em *Python* versão 3.12. Toda as etapas, da obtenção dos dados de entrada à implementação dos modelos computacionais foram organizadas em módulos. Todas as etapas de processamento intensivo foram realizadas em paralelo, utilizando todas as *threads* disponíveis do sistema. As etapas referentes ao modelo neural foram realizadas com aceleração da *Graphics Processing Unit* (GPU).

Considerando que o suporte a aceleração por GPU é limitado no Windows, foi necessário realizar as etapas de treinamento do modelo neural em Linux, utilizando a distribuição Fedora 42. A aceleração por GPU foi necessária para reduzir o tempo de treinamento do modelo. O sistema utilizado possui um processador AMD Ryzen 5900X, 32 GB de memória RAM e placa de vídeo AMD RX 6800 XT. A tabela ?? mostra as bibliotecas utilizadas no projeto, suas finalidades e versões.

Tabela 1 – Bibliotecas utilizadas no projeto

Biblioteca	Descrição	Versão
numpy	Cálculos numéricos e manipulação de arrays	1.26.4
pandas	Manipulação e análise de dados (DataFrames)	2.2.3
requests	Requisições HTTP	2.32.3
urllib3	Gerenciamento de conexões HTTP	2.2.3
alive_progress	Barra de progresso para loops	3.2.0
netCDF4	Leitura de arquivos NetCDF	1.7.2
cdsapi	API para download de dados do ECMWF	0.7.5
geopandas	Manipulação de dados geoespaciais	1.0.1
matplotlib	Visualização de dados	3.9.2
scikit-learn	Aplicação de modelos iniciais	1.5.2
scipy	Ferramentas e algoritmos matemáticos	1.14.1
transformers	Modelos Neurais Pré-treinados	4.52.3
torch	Processamento de Redes Neurais	2.7.0

Fonte: o autor.

4.2 Obtenção e Pré Processamento dos Dados

4.2.1 Obtenção dos Dados Energéticos e Hidrológicos

A primeira etapa do projeto consiste na consolidação das séries históricas de geração, carga e variáveis hidrológicas, que são disponibilizadas publicamente no portal Dados Abertos do ONS, a partir do ano 2000. As séries referentes às variáveis hidrológicas são disponibilizadas em base diária, e os dados de geração e carga são disponibilizados em base horária. As Tabelas ??, ?? e ?? mostram os parâmetros dos dados hidrológicos, carga e geração, respectivamente.

Os dados de geração são disponibilizados em Mega Watt médio (MWmed) por fonte de energia, subsistema, estado, modalidade de operação, entre outras variáveis. Os dados de carga também são disponibilizados em MWmed e contêm informações sobre a carga em cada subsistema do SIN.

Tabela 2 – Parâmetros dos dados hidrológicos

Parâmetro	Descrição	Tipo
din_instante	Instante de aferição	Datetime
nom_subistema	Subsistema	String
tip_reservatorio	Tipo de reservatório	String
nom_bacia	Bacia hidrográfica	String
nom_ree	Nome do REE	String
val_nivelmontante	Valor do nível montante (m)	Float
val_niveljusante	Valor do nível jusante (m)	Float
val_volumetutilcon	Volume útil consistido (%)	Float
val_vazaoafluente	Vazão afluente (m ³ /s)	Float
val_vazaoturbinada	Vazão turbinada (m ³ /s)	Float
val_vazaovertida	Vazão vertida (m ³ /s)	Float
val_vazaodefluente	Vazão defluente (m ³ /s)	Float
val_vazaoevaporacaoliquida	Vazão de evaporação líquida (m ³ /s)	Float

Fonte: ??

Nota: Variáveis não utilizadas foram omitidas.

Para as séries de geração, os dados de 2000 a 2021 são agrupados pelos respectivos anos, e a partir de 2022, as informações estão agrupadas em arquivos por mês e ano. Para as séries de carga, os dados são disponibilizados por ano. Como o ONS não disponibiliza *Application Programming Interface* (API) para a obtenção dos dados diretamente, foi necessário uma outra abordagem, a fim de evitar o *download* manual dos dados.

Um script foi desenvolvido para fazer o download dos dados por meio de requisições HTTP, utilizando as bibliotecas `requests` e `urllib3` para gerenciar as conexões. Todos os downloads foram realizados em paralelo, utilizando todas as threads disponíveis do sistema.

Tabela 3 - Parâmetros dos dados de carga

Parâmetro	Descrição	Tipo
din_instante	Instante de aferição	Datetime
nom_subistema	Subsistema da usina	String
val_cargaenergiahomwmed	Carga de energia (MWmed)	Float

Fonte: ??)

Ao todo, cerca de 10 GB de dados em arquivos Comma Separated Values (CSV) foram consolidados.

Tabela 4 - Parâmetros dos dados de geração

Parâmetro	Descrição	Tipo
id_subistema	Instante de aferição	Datetime
nom_subistema	Subsistema da usina	String
id_estado	Estado onde a usina está localizada	String
nom_tipousina	Tipo de usina	String
nom_tipocombustivel	Tipo de combustível	String
nom_usina	Nome da usina	String
val_geracao	Geração de energia (MWmed)	Float

Fonte: ??)

Nota: Variáveis não utilizadas foram omitidas.

Os dados de geração contém informações que permitem uma análise detalhada em diferentes níveis de granulidade. Dessa maneira, possíveis impactos em diferentes escalas geográficas e temporais poderão ser avaliados.

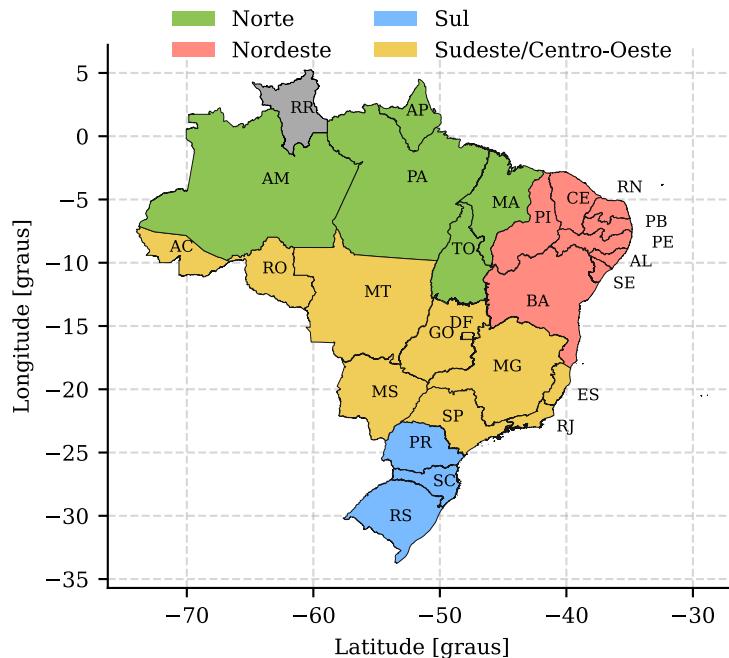
4.2.2 Pré Processamento dos Dados de Geração e Carga

Com as séries históricas de geração e carga consolidadas, se faz necessário preparar os dados para que possam ser utilizados nos modelos computacionais. Essa etapa envolve a verificação, transformação e limpeza dos dados. Utilizando a biblioteca `pandas`, todos os arquivos com o histórico de geração e carga foram lidos e consolidados em dois `DataFrames` distintos, estrutura de dados tabulares da biblioteca. Com isso, as seguintes operações foram realizadas:

- Seleção das colunas relevantes;
- Verificação de valores inválidos e tratamento de valores ausentes;
- Agrupamento dos tipos de usinas por classes: hidráulica, térmica, eólica, fotovoltaica e outras;
- Reamostragem para diferentes bases temporais, considerando a geração média (MWmed) e energia gerada (MWh);

- Agrupamento dos dados de geração por subsistema e classe.

Figura 12 – Subsistemas do SIN segundo o ONS



Fonte: o autor.

Optou-se por fazer o agrupamento dos dados de geração por subsistema e fonte para permitir avaliar os impactos das variáveis do ENSO em diferentes regiões do Brasil. Além disso, a representação por subsistema também é utilizada pelo modelo NEWAVE. A figura ?? mostra os subsistemas do SIN, na qual o estado de Roraima (RR) ainda é considerado um subsistema isolado, ainda que as obras de interligação com o sistema nacional tenham sido iniciadas.

Diferentes arquivos CSV foram gerados, considerando a geração média (MWmed) e energia gerada (MWh) para as escalas diária e mensal. Os dados de geração contém o histórico de geração por subsistema e fonte, enquanto que os dados de carga contém o histórico de carga por subsistema. Considerando o período de 2000 a 2024, foram consolidados ao todo 9132 amostras diárias e 300 amostras mensais.

4.2.3 Pré Processamento dos Dados Hidrológicos

Os dados hidrológicos contém informações sobre o nível dos reservatórios e vazões. Para o contexto do projeto, espera-se que essas informações sejam altamente correlacionados com a geração hidrelétrica, o que será verificado a seguir na seção de análise exploratória.

Sendo assim, os dados hidrológicos foram utilizados para verificar a performance dos modelos quando as variáveis de entrada possuem alta correlação com a variável de saída. Dessa forma, é estabelecida uma referência para comparação entre os resultados dos modelos.

As séries históricas foram consolidadas em um único `DataFrame`, contendo as vazões totais em cada subsistema do SIN, considerando a soma das vazões de todos os reservatórios em cada subsistema, em base diária.

4.2.4 Obtenção dos Dados do ENSO

Os dados do ENSO foram obtidos a partir do *Climate Data Store* (CDS) do ECMWF. O CDS é um banco de dados com diversos *datasets* de variáveis climáticas de diferentes regiões do mundo. Para esse projeto, o dataset utilizado foi o *ERA5 post-processed daily statistics on single levels from 1940 to present*, que contém dados diários de diversas variáveis, incluindo a temperatura da superfície do mar (SST).

Considerando que ENSO é um fenômeno definido pela temperatura da superfície do mar em regiões específicas do Oceano Pacífico, e que os dados de geração, carga e hidrológicos são disponibilizados em base diária, optou-se por utilizar a temperatura absoluta da superfície do mar em base diária para cada uma das regiões do ENSO. As regiões do ENSO e suas coordenadas geográficas são mostradas na Tabela ??.

Tabela 5 – Regiões do ENSO

Região	Latitude	Longitude
Niño 1+2	-10° a 0°	-90° a -80°
Niño 3	-5° a 5°	-150° a -90°
Niño 3.4	-5° a 5°	-170° a -120°
Niño 4	-5° a 5°	-160° a -150°

Fonte: ??)

Os dados foram obtidos usando a biblioteca `cdsapi`, que permite acessar o CDS através da API do ECMWF. O script desenvolvido para essa etapa realiza o download dos dados de temperatura da superfície do mar para cada uma das regiões do ENSO, considerando o período de 2000 a 2024. Os dados são obtidos em formato NetCDF, que é um formato de arquivo utilizado para armazenar dados científicos multidimensionais.

4.2.5 Pré Processamento dos Dados do ENSO

Após o download, os arquivos NetCDF são processados para extrair as informações relevantes para cada região do ENSO. Utilizando a biblioteca `netCDF4`, cada arquivo anual é lido para extrair as dimensões de tempo, latitude e longitude, além da variável de interesse, a temperatura da superfície do mar (SST).

O processamento segue as seguintes etapas:

- Conversão do formato das coordenadas de longitude, de 0 a 360 graus, para -180 a 180 graus;
- Para cada uma das regiões do ENSO, um subconjunto geográfico dos dados globais é criado, selecionando os pontos de latitude e longitude que se encontram dentro dos limites de cada região;
- A média espacial da variável é calculada para cada dia sobre este subconjunto. Esse processo resulta em uma única série temporal diária, que representa o valor médio da variável para aquela região específica;
- Ao final, as séries temporais anuais de cada região são consolidadas, formando um conjunto de dados único que abrange todo o período de análise, de 2000 a 2024.

Em resumo, o processo transforma os dados brutos multidimensionais em séries temporais diárias para cada região do ENSO, que agora estão prontas para serem utilizadas como variáveis exógenas nos modelos computacionais.

4.3 Análise Exploratória dos Dados

A análise exploratória dos dados é uma etapa crucial para entender as características e padrões dos dados antes de aplicar modelos computacionais. Essa etapa envolve a visualização e análise estatística, permitindo identificar tendências, sazonalidades, correlações e possíveis outliers. Além das bibliotecas `pandas` e `matplotlib`, a biblioteca `statsmodels` foi utilizada para realizar a decomposição das séries temporais, permitindo avaliar as sazonalidades e tendências presentes nos dados.

As análises foram realizadas em base temporal mensal visando otimizar a relação sinal-ruído. Na prática, a agregação mensal funciona como um filtro passa-baixas, suavizando as variações diárias devido a ajustes operacionais (alta frequência), permitindo

uma melhor visualização das tendências e sazonalidades de longo prazo (baixas frequências).

Para a decomposição das séries temporais, a função `MSTL` foi utilizada, que realiza a decomposição em componentes de tendência, sazonalidade e resíduo. Essa função utiliza o método de *local polynomial regression fitting*, ou ajuste de regressão polinomial local (LOESS) para decompor a série, sendo capaz de capturar sazonalidades de séries temporais não estacionárias, que é o caso das séries de geração e carga do SIN. Para todas as análises a seguir, os dados estão em base temporal mensal e foram normalizados utilizando o `StandardScaler` da biblioteca `scikit-learn`.

4.3.1 Dados de Geração e Carga

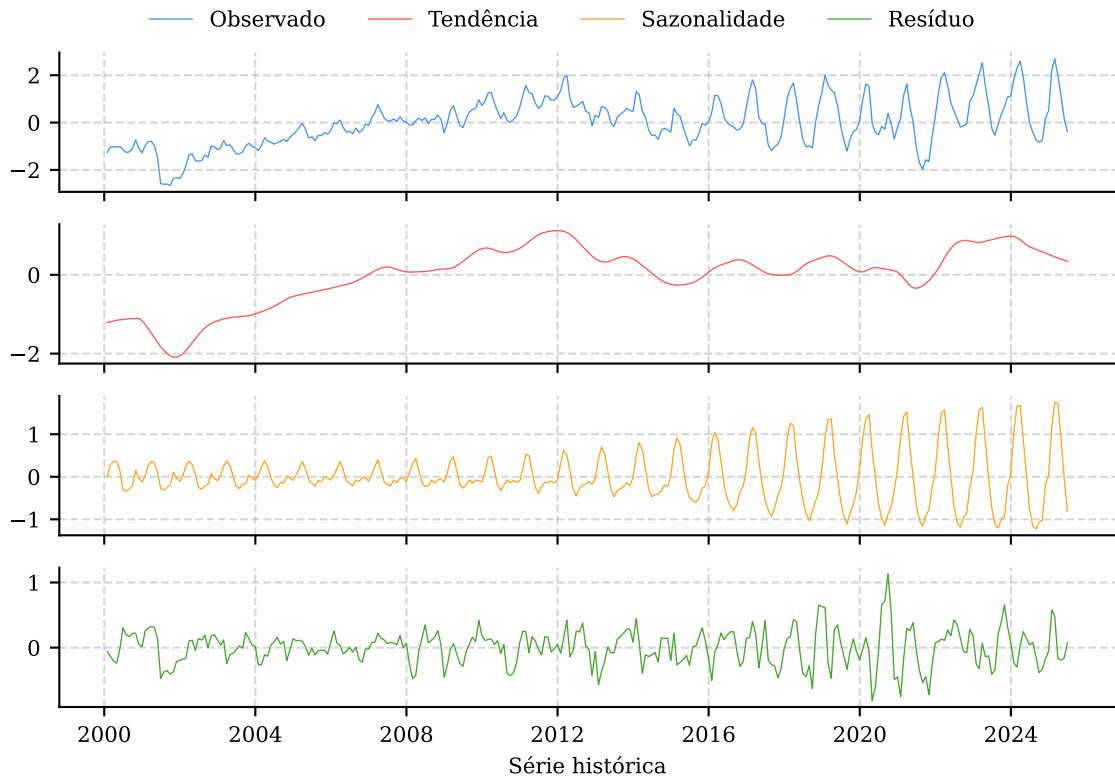
4.3.1.1 Dados de Geração

A Figura ?? mostra a decomposição da série temporal de geração hidráulica, que revela uma tendência de aumento dos anos de 2000 a 2012, seguida por uma período de estabilidade de 2012 a 2024. Ao mesmo tempo, o regime de sazonalidade apresenta maior amplitude após o ano de 2012, sugerindo uma maior variabilidade da contribuição hidráulica para a geração do SIN. Esse comportamento pode ser explicado pela introdução em larga escala das fontes renováveis variáveis, como a eólica e a solar fotovoltaica, na matriz elétrica. (??)

Conforme descrito no capítulo 1, a geração dessas fontes é intermitente e não despachável, ou seja, dependente da disponibilidade de vento e sol. Nesse cenário, as usinas hidrelétricas, especialmente aquelas com reservatórios, passaram a atuar como a principal ferramenta de flexibilidade do sistema, compensando a variabilidade das fontes intermitentes.

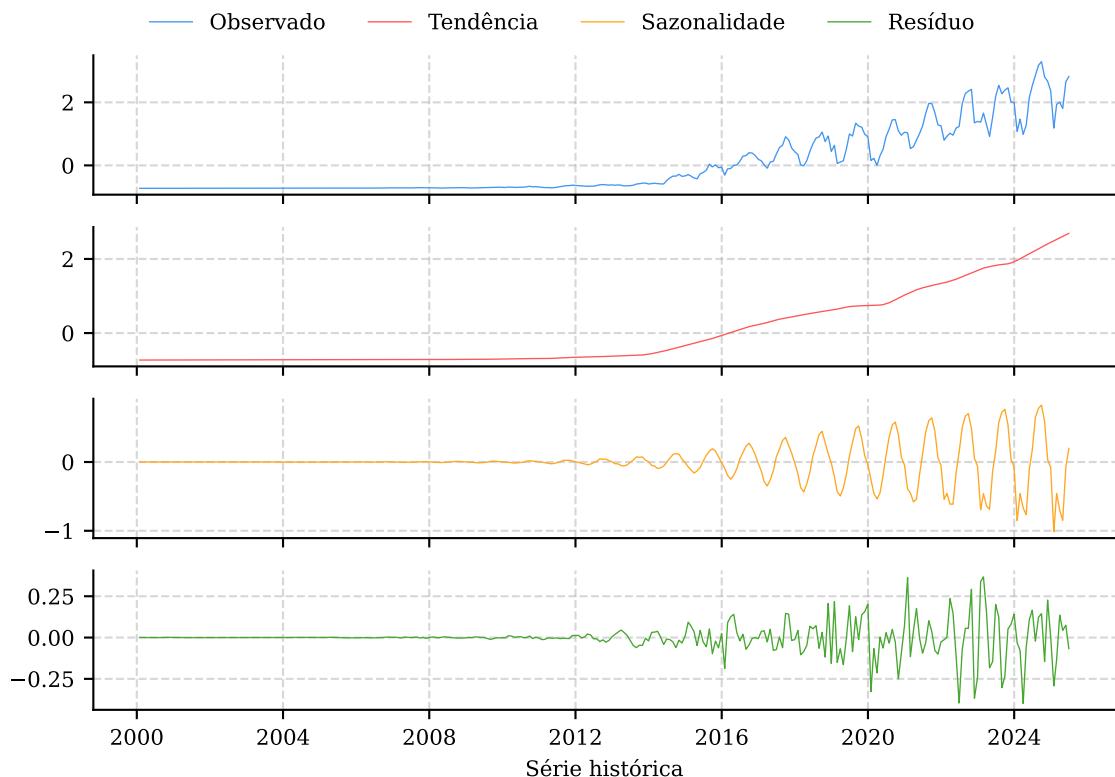
Em momentos de alta geração eólica e solar, o ONS reduz a produção hidrelétrica para absorver a energia renovável, criando vales de geração mais profundos. Inversamente, em períodos de baixa geração renovável, como durante a noite, a geração hidrelétrica é acionada para suprir a demanda, resultando em picos mais elevados. Portanto, o aumento da amplitude da sazonalidade após 2012 é a assinatura visual desse novo papel operacional, no qual a fonte hidráulica não apenas segue o ciclo hidrológico, mas também responde dinamicamente à intermitência e situações de pico de demanda para garantir a estabilidade do SIN. (??)

Figura 13 – Decomposição da série temporal de geração hidráulica



Fonte: o autor.

Figura 14 – Decomposição da série temporal de geração eólica

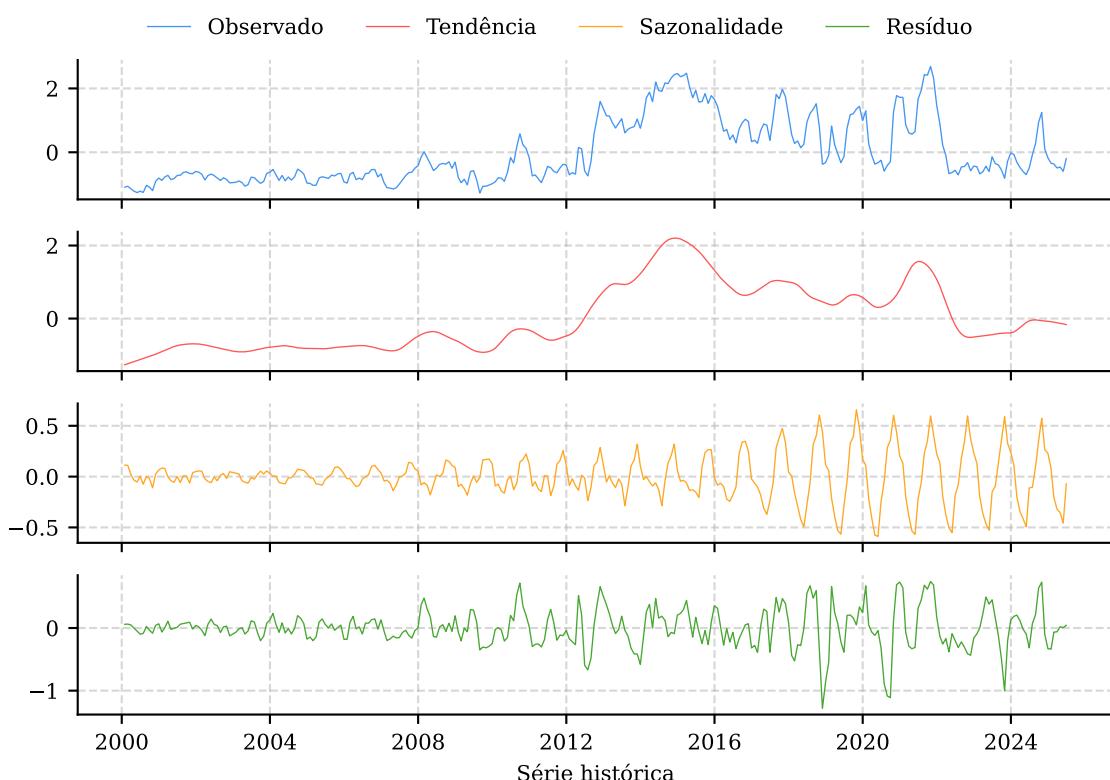


Fonte: o autor.

A Figura ?? mostra a decomposição da série temporal de geração eólica. A tendência evidencia o crescimento exponencial da capacidade instalada a partir de 2012, reflexo das políticas de incentivo e dos leilões de energia. Diferentemente da fonte hidráulica, a sazonalidade da geração eólica é ditada exclusivamente pela disponibilidade do recurso, com picos correspondentes à "safra dos ventos", que ocorre tipicamente no segundo semestre.

A amplitude crescente da sazonalidade e a maior variância do resíduo são consequências diretas da expansão da capacidade instalada. Em conjunto, a sazonalidade e o resíduo da geração eólica são as principais causas da mudança no perfil operacional da geração hidráulica, que ajusta sua produção para compensar a intermitência da fonte eólica.

Figura 15 - Decomposição da série temporal de geração térmica



Fonte: o autor.

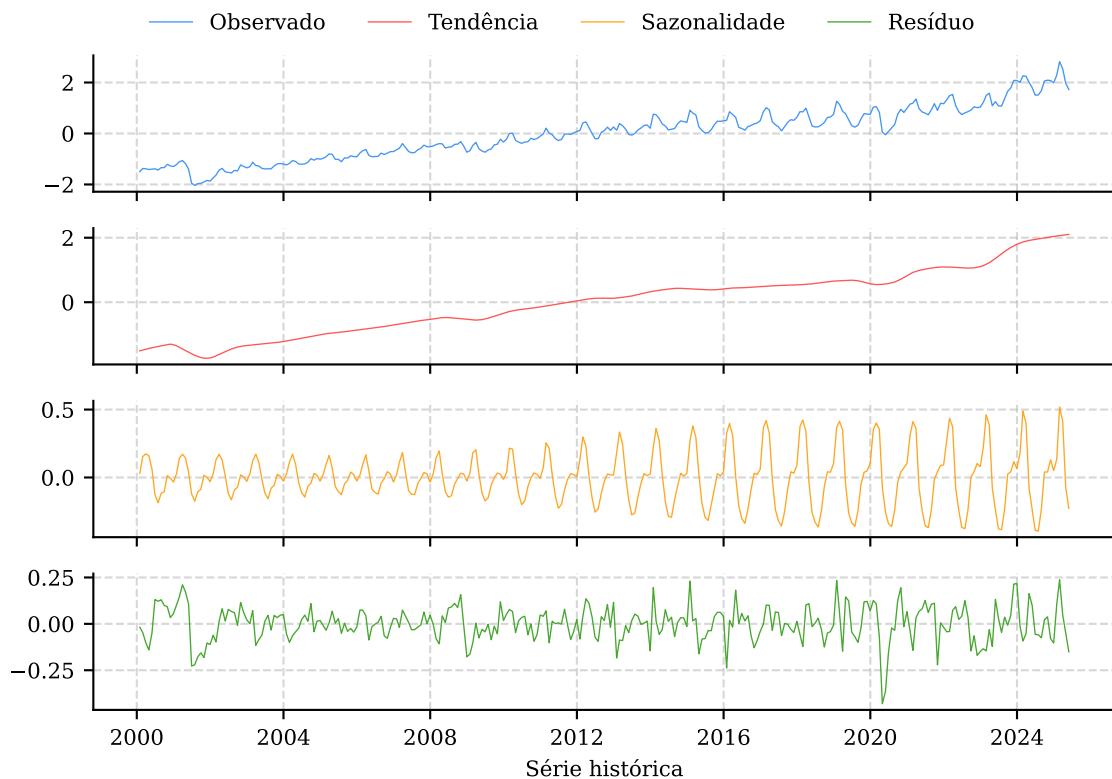
A Figura ?? apresenta a decomposição da geração térmica. A tendência atua como um termômetro do risco hidrológico do país, com picos proeminentes que coincidem diretamente com períodos de crise hídrica, como em 2014-2015 e 2021, quando o despacho térmico é massivo para garantir o suprimento.

A sazonalidade opera de forma inversa ao ciclo hidrológico, com maior geração no

período seco para poupar os reservatórios. O resíduo, por sua vez, exibe alta volatilidade e representa o papel da fonte térmica como recurso de ponta e de emergência, acionada para cobrir a intermitência de outras fontes, atender picos de demanda e responder a indisponibilidades no sistema.

4.3.1.2 Dados de Carga

Figura 16 – Decomposição da série temporal de carga



Fonte: o autor.

A Figura ?? mostra a decomposição da série temporal de carga. A tendência reflete diretamente a atividade econômica do país, com um crescimento contínuo até 2014, seguido por uma estagnação durante a recessão de 2014-2016 e uma queda abrupta em 2020 devido à pandemia de COVID-19. (??)

A sazonalidade é a mais regular entre todas as séries, ditada pelo clima, com picos de consumo consistentes no verão devido ao uso de sistemas de refrigeração. O resíduo captura variações de curto prazo, como ondas de temperatura e feriados, e possui uma volatilidade relativamente baixa.

4.4 Implementação dos Modelos de Regressão

Para todos os casos, os códigos referentes às implementações dos modelos estão disponíveis no Apêndice A. A seguir, será apresentada uma visão geral das etapas de implementação, métricas de avaliação e os modelos computacionais utilizados.

4.4.1 Métricas de Avaliação

Para todos os casos, duas métricas de avaliação foram utilizadas: o erro quadrático médio (MSE) e o coeficiente de determinação (R^2). O MSE é uma medida que quantifica a média dos erros quadráticos entre os valores reais e as previsões do modelo. O R^2 , por outro lado, é uma medida que indica a proporção da variabilidade dos dados que é explicada pelo modelo, variando de 0 a 1, onde valores mais próximos de 1 indicam um modelo mais preciso.

As equações para o cálculo dessas métricas são apresentadas a seguir:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (4.1)$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (4.2)$$

em que y_i são os valores reais, \hat{y}_i são os valores previstos pelo modelo, \bar{y} é a média dos valores reais e n é o número de amostras. Essas métricas estão disponíveis na biblioteca `scikit-learn` através das funções `mean_squared_error` e `r2_score`, respectivamente.

4.4.2 Etapas Comuns

As etapas iniciais de implementação são comuns a todos os modelos, consistindo em:

1. **Entrada de dados:** leitura dos arquivos CSV contendo os dados de geração, carga e variáveis do ENSO;
2. **Consolidação do dataset:** alinhamento temporal dos dados, garantindo que as séries estejam sincronizadas;
3. **Definição das variáveis:** seleção das clunas contendo as variáveis dependentes (geração por fonte e subsistema) e independentes (carga e variáveis do ENSO);
4. **Normalização:** normalização das variáveis independentes utilizando o `StandardScaler` do `scikit-learn`;

5. **Definição dos conjuntos para treino:** divisão dos dados em conjuntos de treino, teste e validação (para o modelo neural);
6. **Avaliação:** avaliação através das métricas escolhidas e visual dos resultados através de gráficos de dispersão e linha.

A normalização é importante para evitar que o modelo atribua maior peso às variáveis com maior amplitude, o que poderia resultar em resultados não representativos. Para a etapa de treinamento, na qual os modelos atualizam seus parâmetros para minimizar o erro, os dados são divididos em conjuntos de treino, teste e validação (para o caso do modelo neural).

O conjunto de treino é utilizado para ajustar os parâmetros do modelo, enquanto o conjunto de teste é utilizado para avaliar a performance do modelo em dados não vistos. Para o modelo neural, o conjunto de validação é utilizado para monitorar o desempenho do modelo durante o treinamento e evitar o sobreajuste. Diferentes tamanhos da janela de treinamento foram avaliados e serão apresentados no capítulo 5.

Os gráficos de dispersão mostram os valores reais e estimados junto a uma linha ideal, de modo que valores próximos à linha indicam resultados mais precisos. O outro gráfico demonstra a separação dos períodos de treino e teste, permitindo uma avaliação mais direta entre os resultados reais e estimados. Para facilitar a visualização, os gráficos são apresentados em base temporal mensal, considerando a média mensal dos valores.

4.4.3 Modelo Linear

O modelo `LinearRegression` da biblioteca `scikit-learn` foi implementado com a abordagem de regressão linear múltipla, uma técnica estatística que busca modelar a relação entre uma variável dependente e várias variáveis independentes, conforme descrito pela equação $??$. Nesse contexto, a variável dependente é a geração de uma determinada fonte em um subsistema do SIN, e as variáveis independentes são as variáveis auxiliares (ou exógenas), os dados de SST e carga.

Foi considerado apenas o período de 2010 a 2024, já que a fonte eólica era pouco presente no SIN até então. Utilizar todo o dataset poderia resultar em resultados não representativos, uma vez que o modelo assume que a relação entre as variáveis é linear e constante ao longo do tempo, dado que as séries de geração são não estacionárias. Assim, a escolha do período de 2010 a 2024 é justificada pela necessidade de garantir que os

dados utilizados sejam representativos do comportamento atual do sistema, permitindo uma análise mais precisa e confiável.

Após as etapas comuns, de carregamento de dados e tratamento inicial, os datasets de treino e teste foram definidos, de modo que o conjunto de treino contenha 70% dos dados e o conjunto de teste 30%. O modelo é então instanciado com paralelismo do CPU habilitado e treinado no conjunto de treino. A seguir, o modelo é avaliado no conjunto de teste, e os resultados são apresentados através dos gráficos.

O modelo linear permite uma interpretação direta dos resultados, uma vez que é possível obter os coeficientes da equação linear que descreve a relação entre as variáveis dependentes e independentes. Os demais modelos não permitem essa análise diretamente.

4.4.4 Modelo Não Linear

O modelo `RandomForestRegressor` da biblioteca `scikit-learn` foi implementado com a abordagem de previsão multivariada. O modelo é um dos poucos da biblioteca que permite prever múltiplas variáveis dependentes simultaneamente, sem considerar os modelos neurais, e foi selecionado por este motivo.

Para este caso, também foi considerado o período de 2010 a 2024, e as etapas comuns foram realizadas. Após a consolidação dos dados de geração, carga e ENSO, foram criados atributos temporais adicionais, como ano, mês e dia do ano, para enriquecer o conjunto de dados. As variáveis exógenas (carga e ENSO) foram normalizadas com o `StandardScaler`. O conjunto de dados foi então dividido cronologicamente, com 70% dos dados para treino e 30% para teste, utilizando a função `train_test_split` com o parâmetro `shuffle = False`.

Para definir os melhores hiperparâmetros do modelo, foi utilizado o `GridSearchCV`, que realiza uma busca exaustiva por meio de validação cruzada em grade. A validação cruzada foi adaptada para séries temporais com o uso do `TimeSeriesSplit`, que garante que os dados de treino sempre ocorram antes dos dados de validação em cada divisão. O modelo foi treinado no conjunto de treino, e os melhores parâmetros foram selecionados com base na métrica de erro quadrático médio (MSE).

Após a aplicação do `GridSearchCV`, o modelo foi instanciado com os parâmetros otimizados, com número de árvores de decisão `n_estimators = 1000` e profundidade máxima `max_depth = 30`, com o paralelismo habilitado `n_jobs = -1`. O modelo treinado foi então utilizado para realizar as previsões no conjunto de teste. Os resultados foram avaliados

com as métricas R^2 e MSE. Por fim, para cada variável alvo, foram gerados e salvos gráficos de dispersão e de série temporal para a análise visual dos resultados.

4.5 Implementação do Modelo Neural

O modelo neural utilizado foi o *Tiny Time Mixer* (TTM), versão 2.1, disponível na biblioteca `transformers` do *Hugging Face*. Ele é um modelo de previsão de séries temporais pré-treinado baseado na arquitetura *TSMixer*, apresentada no capítulo 3. Modelos pré-treinados são modelos que já foram treinados em grandes conjuntos de dados. Ou seja, os pesos das camadas de *perceptrons* já foram ajustados para capturar padrões gerais em séries temporais.

Além disso, o modelo permite o processo de *fine tuning*, que é a atualização dos pesos para adequar-se a um conjunto de dados específico, considerando variáveis dependentes e independentes, que no contexto do projeto são a geração de energia por subsistema e fonte, carga e variáveis do ENSO.

O modelo possui diferentes variações, com diferentes tamanhos de janelas de contexto e previsão e tem suporte para escalas de tempo semanal, diária, horária e de minutos. Para o projeto, foram utilizadas as variantes com janelas de contexto e previsão de 512 e 96 dias, bem como a de 90 e 30 semanas.

A biblioteca `tsfm_public`, que contém funções auxiliares para a implementação dos modelos publicados pela IBM, foi utilizada para definir os conjuntos de treino, teste e validação, além de realizar o pré processamento dos dados, incluindo o processo de normalização.

Vale destacar que, embora ??) sugira o congelamento de todo o *backbone* do modelo, ou seja, não atualizar todos os pesos das camadas escondidas durante o ajuste fino, de modo a preservar o conhecimento prévio do modelo, isso não foi seguido para esse projeto, dado que resultados superiores foram obtidos ao permitir que todas as camadas fossem atualizadas. Dessa forma, o modelo é capaz de aprender padrões mais complexos e específicos do conjunto de dados, o que pode levar a uma melhor performance nas previsões.

4.5.1 Modelo Pré-treinado

Inicialmente, foi implementado o modelo sem considerar as variáveis exógenas, ou seja, apenas com as variáveis dependentes, os dados de geração por subsistema e fonte. Essa abordagem, conhecida como *one-shot forecasting*, consiste em aplicar um modelo já treinado diretamente sobre a série temporal de interesse para realizar a inferência, sem uma nova etapa de treinamento.

Para esta análise, os dados de 2000 a 2024 foram considerados, devido a capacidade do modelo capturar relações mais complexas. O conjunto de dados foi então dividido em frações de treino (70%), validação (10%) e teste (20%), utilizando a função `prepare_data_splits`. Diferentes variantes do modelo foram avaliadas para diferentes amostragens temporais, e os resultados serão apresentados no capítulo 5.

O pré-processamento foi realizado pela classe `TimeSeriesPreprocessor`, configurada para as variáveis de geração como alvo `target_columns`. A classe ajusta os dados para o formato esperado pelo modelo, com janelas de contexto definidas considerando a variante selecionada, além de aplicar a normalização `StandardScaler`, cujos parâmetros são aprendidos a partir do conjunto de treino.

Os modelos `TinyTimeMixerForPrediction` utilizados nesta etapa foram diferentes variantes pré-treinadas, carregadas a partir de suas respectivas identificações do *Hugging Face*. Apenas as camadas de entrada e saída do modelo foram adaptadas para corresponder ao número de variáveis de geração.

A previsão é executada através de uma `TimeSeriesForecastingPipeline`, que aplica o modelo pré-processado sobre o conjunto de teste. Como as previsões são geradas em frequência semanal, um pós-processamento é realizado para agrregar os resultados em médias mensais, permitindo uma comparação direta com os demais modelos. As métricas R² e MSE são calculadas sobre os dados com a mesma base temporal. Por fim, para cada variável, são gerados e salvos gráficos de dispersão e de série temporal para análise visual dos resultados.

4.5.2 Ajuste Fino

Para a implementação do modelo com variáveis exógenas, faz-se necessário realizar o processo de *fine tuning*, que é o processo de atualização dos pesos do modelo pré-treinado para adequar-se ao conjunto de dados específico, considerando a presença de variáveis

dependentes e independentes.

Após a consolidação dos dados de geração (variáveis dependentes), carga e ENSO (variáveis exógenas), o conjunto de dados é dividido em frações de treino (70%), validação (10%) e teste (20%), utilizando a função `prepare_data_splits`. A biblioteca `tsfm_public` é novamente empregada para o pré-processamento através da classe `TimeSeriesPreprocessor`, que desta vez é configurada para tratar tanto as variáveis alvo `target_columns` quanto as exógenas `control_columns`, aplicando a normalização `StandardScaler` em ambas.

O modelo `TinyTimeMixerForPrediction` é carregado, e sua arquitetura é ajustada para o novo conjunto de dados, especificando os canais de entrada para as variáveis dependentes e exógenas. Uma característica central do modelo é a sua capacidade de *mixing*, ou mistura de canais, que permite aprender as interdependências entre as múltiplas séries temporais. Isso é habilitado pelos parâmetros `decoder_mode = 'mix_channel'` e `enable_forecast_channel_mixing = True`. Adicionalmente, o parâmetro `fcm_prepended_past = True` é utilizado para que os valores passados das séries também sejam considerados no processo de mistura, enriquecendo o contexto disponível para a previsão.

Para o treinamento, a taxa de aprendizado `learning_rate` é definida dinamicamente pela função `optimal_lr_finder`, que busca um valor otimizado para a convergência do modelo. O otimizador utilizado foi o `AdamW`, uma variante do otimizador Adam que desacopla a regularização de decaimento de peso (*weight decay*) da atualização do gradiente.

O treinamento é gerenciado pela classe `Trainer` da biblioteca `transformers`, que recebe os hiperparâmetros, como o número máximo de épocas, definido em 500, e `batch_size` (tamanho do lote) `batch_size = 64`. O tamanho do lote define o número de amostras que são propagadas através da rede antes da atualização dos pesos. A escolha do tamanho do lote envolve um compromisso entre a estabilidade da convergência e a capacidade de generalização do modelo. Em tese, lotes maiores podem resultar em um treinamento mais rápido, mas podem fazer com que o modelo converja para um mínimo local, enquanto lotes menores podem levar a uma convergência lenta, mas com maior chance de encontrar um mínimo global.

Adicionalmente, o `OneCycleLR`, é empregado para variar a taxa de aprendizado de forma cíclica durante o treinamento, começando com um valor baixo, aumentando até um máximo e depois diminuindo novamente, o que pode acelerar a convergência.

Uma estratégia de parada antecipada `EarlyStoppingCallback` é implementada para

monitorar a perda no conjunto de validação e interromper o treinamento caso não haja melhora significativa de ao menos 0,001 por 50 épocas consecutivas, evitando o sobreajuste (*overfitting*). O modelo com o menor erro de validação é salvo ao final do processo.

Capítulo 5

Resultados

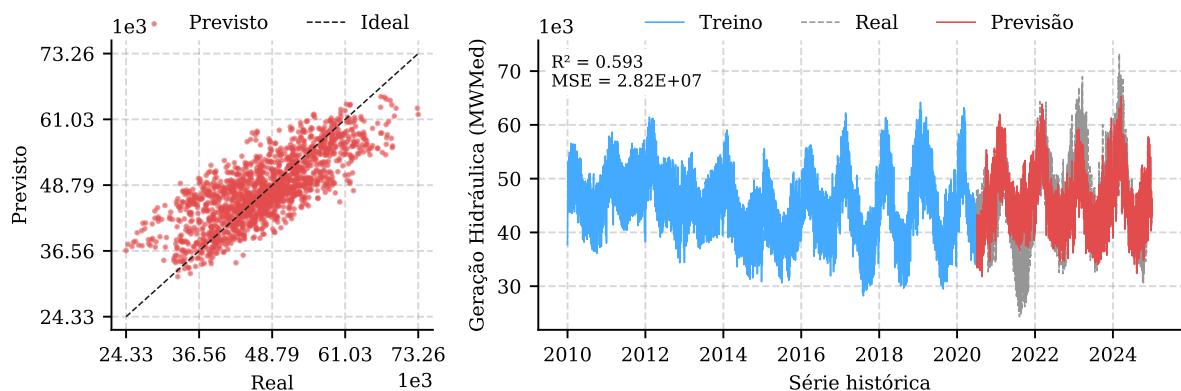
Este capítulo apresenta os resultados obtidos na estimativa da geração das fontes hidráulica, eólica e térmica. A análise se inicia com modelos de regressão linear e não linear (Random Forest) para estabelecer um baseline de performance e avaliar a linearidade das relações (Seções 5.1 e 5.2). Em seguida, na Seção 5.3, avalia-se o desempenho do modelo neural *Tiny Time Mixer* (TTM) em sua aplicação direta. Por fim, na Seção 5.3.2, demonstra-se o impacto e a eficácia do ajuste fino (*finetuning*) deste modelo com a inclusão de variáveis do fenômeno ENSO, quantificando o ganho de acurácia obtido com a especialização do modelo. Serão apresentados os gráficos de geração total das fontes. Demais gráficos dos resultados para os subsistemas estão disponíveis no Apêndice B.

5.1 Modelo Linear

5.1.1 Fonte Hidráulica

A Figura ?? apresenta os resultados da regressão linear para a fonte hidráulica. Para esse caso, foram considerados os dados de 2010 em diante. O conjunto de treino foi constituído pelos dados de 2010 a 2021, e o conjunto de teste pelos dados restantes.

Figura 17 – Regressão linear para fonte hidráulica (carga + ENSO) em base diária



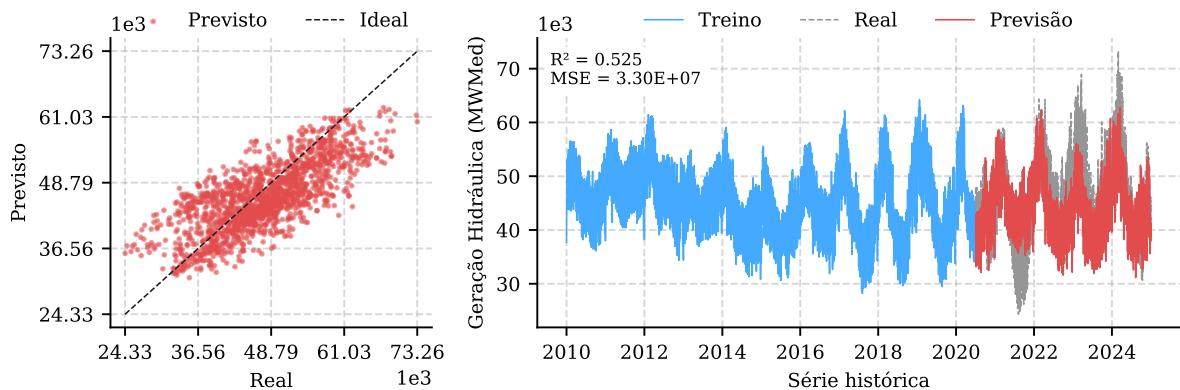
Fonte: o autor.

Fica evidente que, para a geração hidráulica total, o modelo é capaz de estimar os valores de geração do conjunto de teste com um valor de R^2 de 0,593, o que indica

uma correlação moderada entre os valores estimados e os valores reais, considerando as limitações do modelo linear.

Para verificar a utilidade dos dados do ENSO, uma nova regressão foi realizada, dessa vez considerando apenas a carga como variável independente, conforme a Figura ???. Observa-se uma queda no valor de R^2 de 0,593 para 0,525, uma redução de 11,5% na capacidade explicativa do modelo, indicando que a inclusão do ENSO melhora significativamente a predição. Além disso, o resultado é compatível com a fundamentação teórica apresentada no Capítulo 3, na qual a influência do ENSO na geração hidráulica é direta e significativa.

Figura 18 – Regressão linear para fonte hidráulica (apenas carga) em base diária



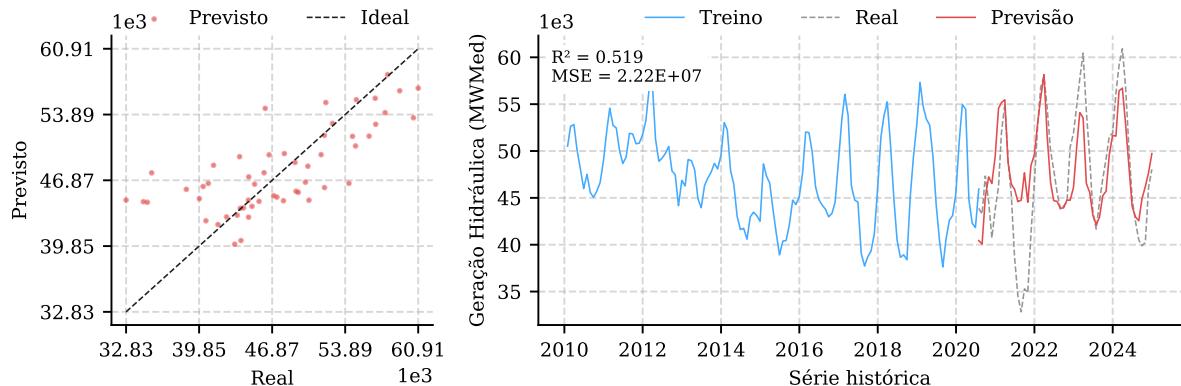
Fonte: o autor.

Embora o modelo linear atinja um R^2 de 0,593, o gráfico de dispersão (à esquerda) mostra que os pontos se afastam da linha ideal especialmente nos extremos. Nota-se que o modelo tende a subestimar os picos de geração (pontos acima da linha) e a superestimar os vales (pontos abaixo da linha), uma limitação característica da regressão linear ao tentar capturar a amplitude total de uma série com forte sazonalidade.

A fim de melhor avaliar os resultados, os demais gráficos serão apresentados em base temporal mensal, indicando a média dos valores de geração para cada mês em MWmed, considerando os valores médios das variáveis independentes para o mesmo período. A Figura ?? apresenta os resultados mensais da regressão linear para a fonte hidráulica, considerando a carga e o ENSO.

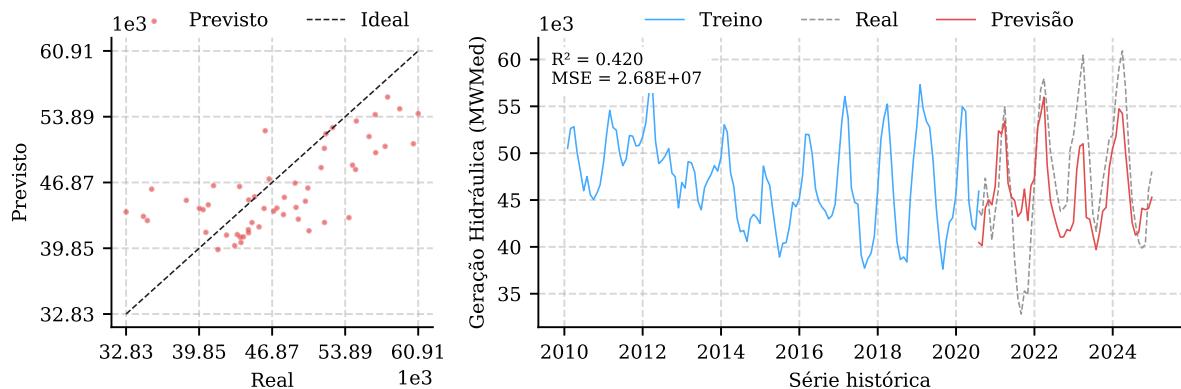
Novamente, é realizado uma avaliação do modelo considerando apenas a carga, conforme a Figura ???. Observa-se que, para a regressão mensal, a queda na performance do modelo é ainda maior: o valor de R^2 cai de 0,519 para 0,420. Isso pode indicar que a inclusão de variáveis do ENSO é ainda mais significante para a base temporal mensal.

Figura 19 – Regressão linear para fonte hidráulica (carga + ENSO) em base mensal



Fonte: o autor.

Figura 20 – Regressão linear para fonte hidráulica (apenas carga) em base mensal



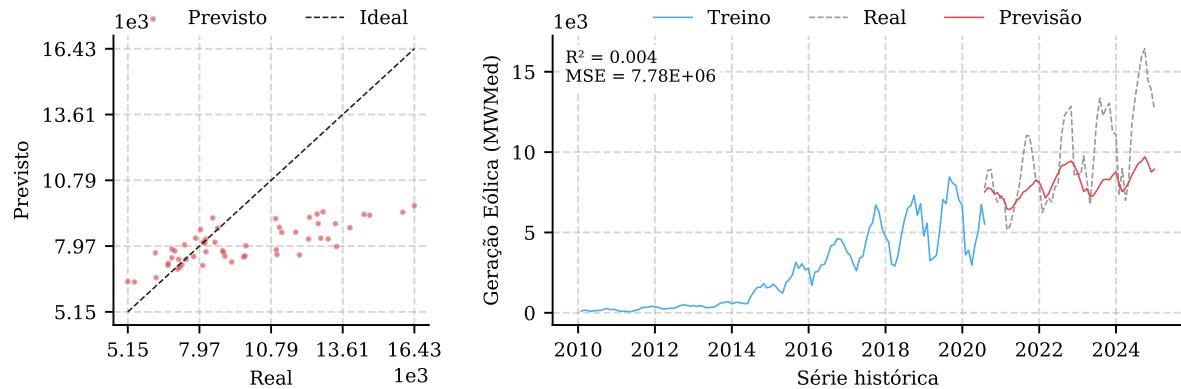
Fonte: o autor.

5.1.2 Fonte Eólica e Térmica

As Figuras ?? e ?? apresentam os resultados para as fontes eólica e térmica, nos quais pode-se observar que o modelo não foi capaz de apresentar resultados satisfatórios, sugerindo que a relação entre as variáveis independentes e dependentes pode ser não linear, validando a hipótese levantada nos capítulos anteriores.

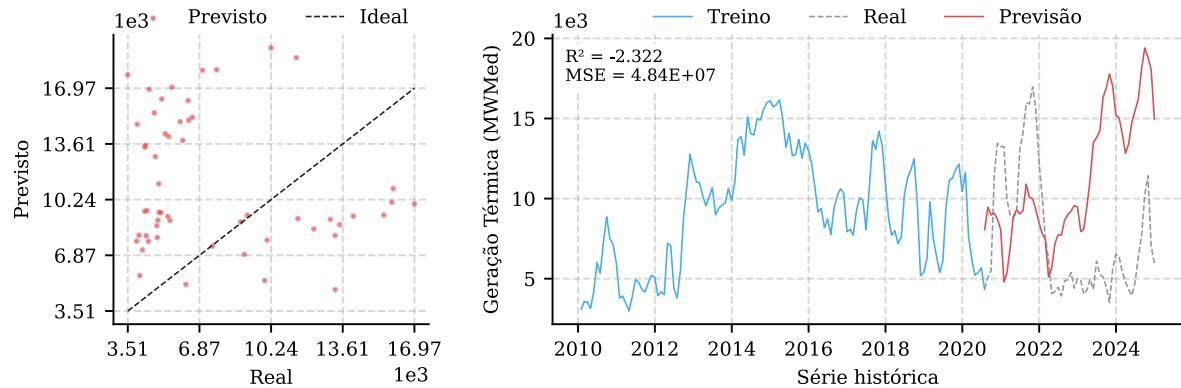
A regressão para a fonte eólica resultou em um valor de R^2 de 0,004, o que indica que o modelo não é capaz de explicar a variabilidade dos dados, enquanto que para a fonte térmica, o valor de R^2 foi negativo, indicando que o modelo é pior do que uma média simples dos dados. Esses resultados corroboram a hipótese de que a relação entre as variáveis independentes e dependentes é não linear, o que justifica a escolha de modelos mais complexos para essas fontes.

Figura 21 – Regressão linear para fonte eólica (carga + ENSO) em base mensal



Fonte: o autor.

Figura 22 – Regressão linear para fonte térmica (carga + ENSO) em base mensal

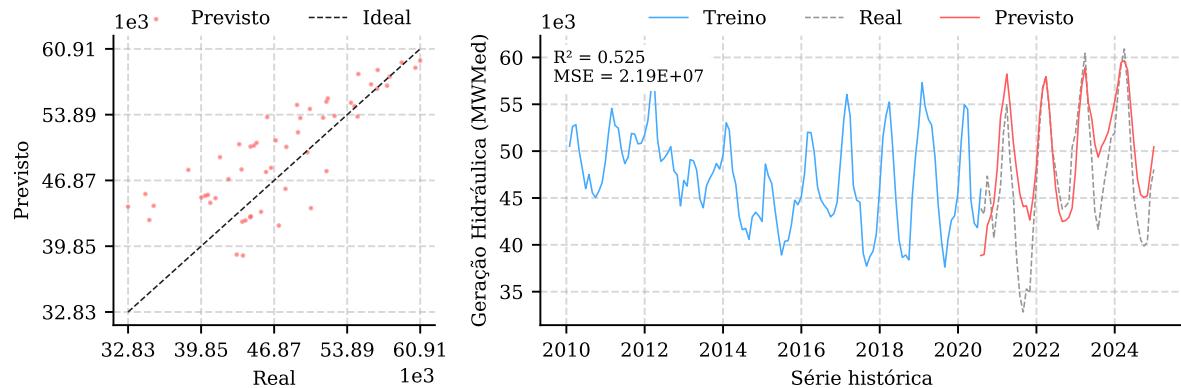


Fonte: o autor.

5.2 Modelo Não Linear (Random Forest Regressor)

5.2.1 Fonte Hidráulica

Figura 23 – Random Forest para fonte hidráulica (carga + ENSO)

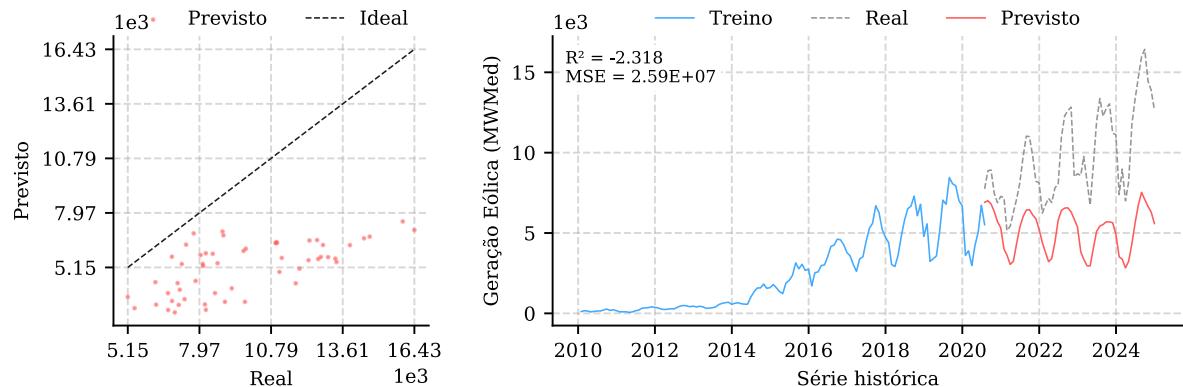


Fonte: o autor.

A Figura ?? apresenta os resultados do modelo não linear para a fonte hidráulica, considerando os dados de carga e ENSO como variáveis independentes. O dataset utilizado contém dados a partir de 2010. Os resultados são ligeiramente superiores aos do modelo de regressão linear, com um valor de R^2 de 0,525.

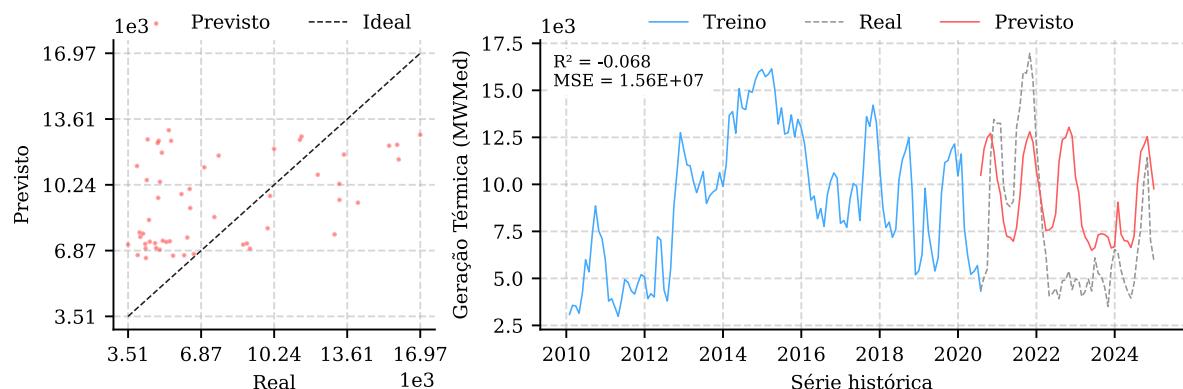
5.2.2 Fonte Eólica e Térmica

Figura 24 - Random Forest para fonte eólica (carga + ENSO)



Fonte: o autor.

Figura 25 - Random Forest para fonte térmica (carga + ENSO)



Fonte: o autor.

As Figuras ?? e ?? apresentam os resultados do modelo não linear para as fontes eólica e térmica, respectivamente. De maneira similar ao modelo linear, o modelo não linear também não é capaz de produzir resultados satisfatórios para essas fontes.

Ainda que o modelo Random Forest apresente uma capacidade superior para modelar não-linearidades, os resultados para as fontes eólica e térmica permaneceram insatisfatórios. Essa dificuldade sugere que as relações temporais são complexas e podem ser mais bem capturadas por modelos como o avaliado a seguir.

5.3 Modelo Neural TTM

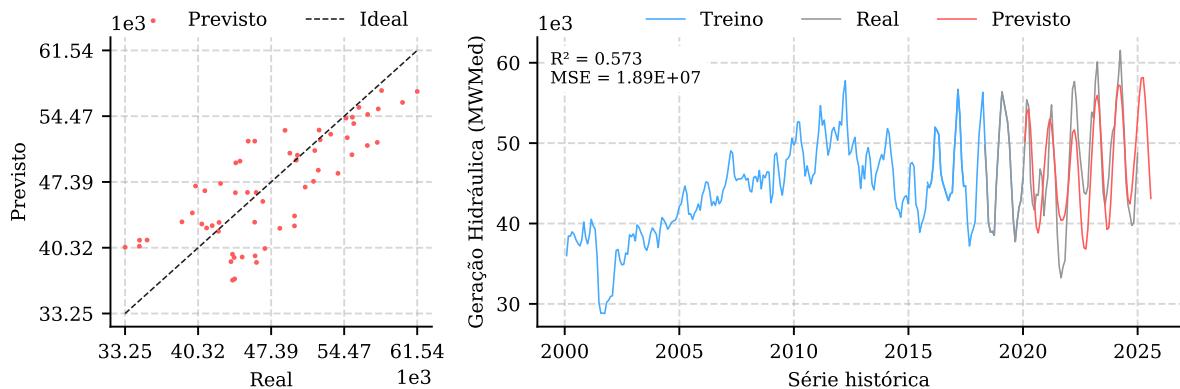
5.3.1 Modelo Pré-treinado

Ao fazer a implementação *oneshot*, todos os pesos do *backbone*, as camadas escondidas, permanecem inalterados em relação ao modelo pré treinado, e apenas a *head*, ou cabeça de previsão, é ajustada para gerar a saída desejada, considerando o número de variáveis dependentes.

A seguir serão apresentados os resultados da previsão *oneshot* para cada fonte de geração, considerando os dados de carga e ENSO como variáveis exógenas. Para cada fonte, serão apresentados os resultados considerando as janelas de contexto e previsão de 90/30 semanas e 512/96 dias. Demais gráficos dos resultados para os subsistemas estão disponíveis no Apêndice B.

5.3.1.1 Fonte Hidráulica

Figura 26 – Oneshot para fonte hidráulica (contexto/previsão: 90/30 semanas)

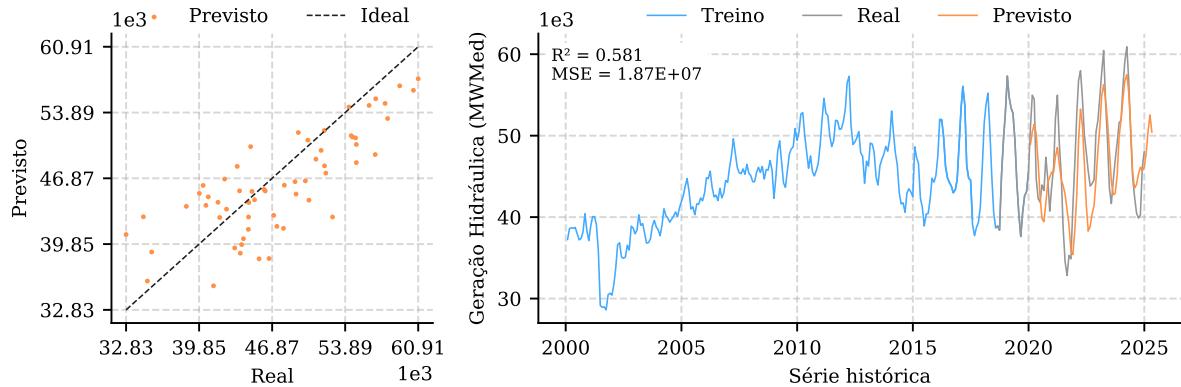


Fonte: o autor.

As Figuras ?? e ?? apresentam os resultados da previsão *oneshot* para a fonte hidráulica, considerando as janelas de contexto e previsão de 90/30 semanas e 512/96 dias, respectivamente. Observa-se que as métricas de avaliação foram superiores para o caso de 512/96 dias. Isso pode ser explicado devido à taxa de amostragem diária, que permite uma maior quantidade de dados para o treinamento do modelo e maior número de janelas de contexto e previsão.

Para o caso de 90/30 semanas, o valor de R^2 foi de 0,573, enquanto que para o caso de 512/96 dias, o valor de R^2 foi de 0,581. Esses valores são razoáveis, considerando

Figura 27 – Oneshot para fonte hidráulica (contexto/previsão: 512/96 dias)

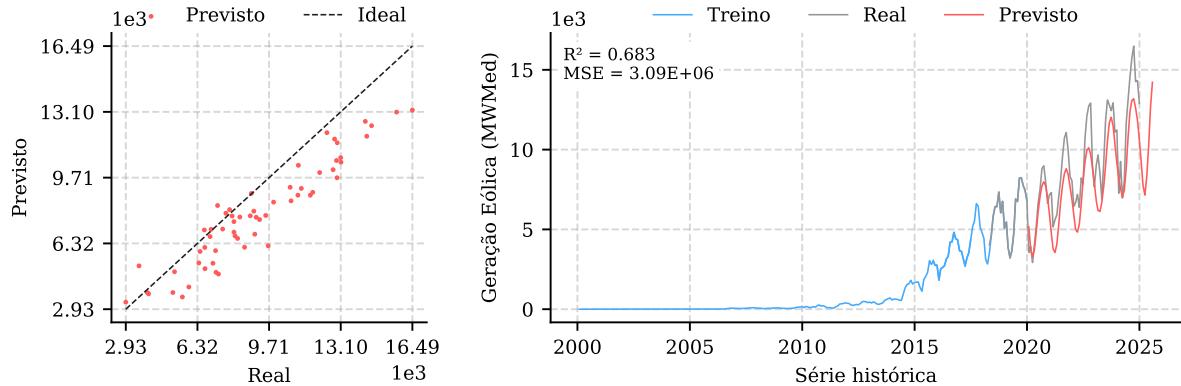


Fonte: o autor.

a implementação *oneshot*, sem ajuste fino para o dataset. Além disso, os resultados são comparáveis aos resultados do modelo linear, o que sugere que a relação da geração hidráulica com as variáveis consideradas pode ser mais linear do que as demais fontes de geração.

5.3.1.2 Fonte Eólica

Figura 28 – Oneshot para fonte eólica (contexto/previsão: 90/30 semanas)

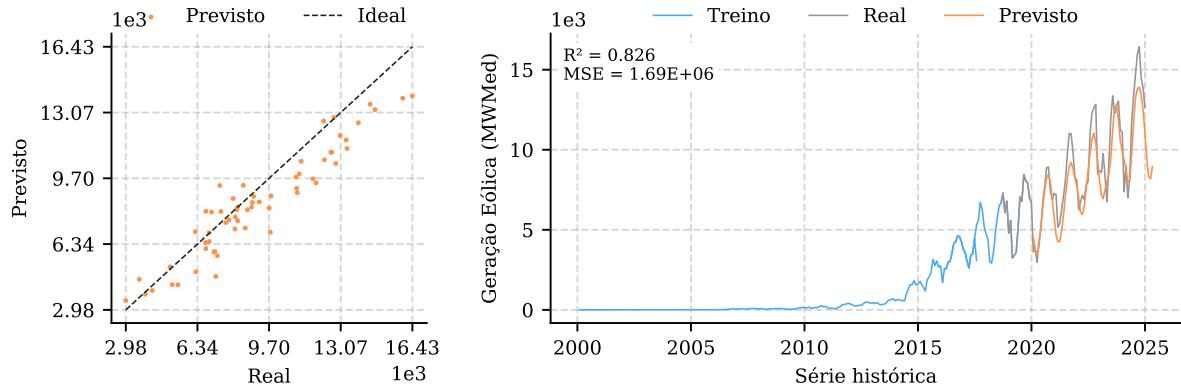


Fonte: o autor.

As Figuras ?? e ?? apresentam os resultados da previsão *oneshot* para a fonte eólica, considerando as janelas de contexto e previsão de 90/30 semanas e 512/96 dias, respectivamente. Observa-se que, para o caso de 90/30 semanas, o valor de R^2 foi de 0,683, enquanto que para o caso de 512/96 dias, o valor de R^2 foi de 0,826.

Esses valores são superiores aos resultados da fonte hidráulica, sugerindo que o modelo consegue capturar melhor a relação entre as variáveis. Além disso, as métricas de

Figura 29 – Oneshot para fonte eólica (contexto/previsão: 512/96 dias)

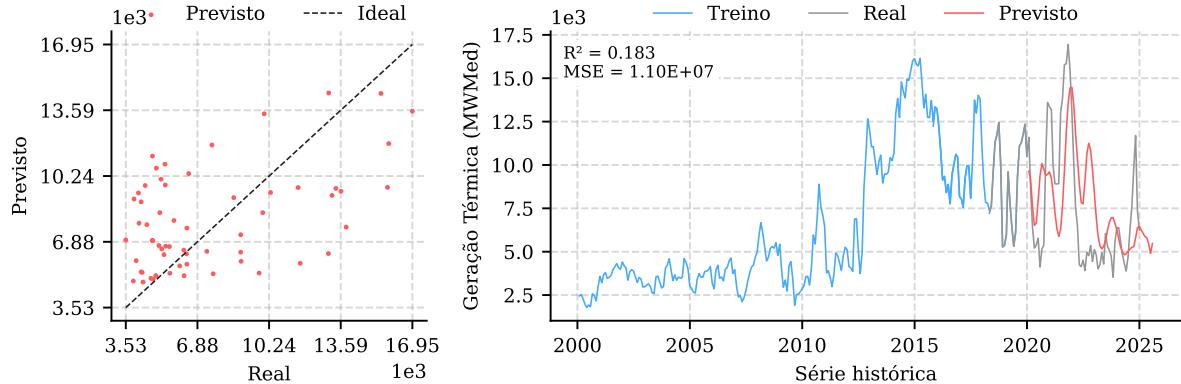


Fonte: o autor.

avaliação foram consideravelmente superiores em comparação com os modelos linear e não linear, sugerindo que, de fato, a relação entre as variáveis é extremamente não linear.

5.3.1.3 Fonte Térmica

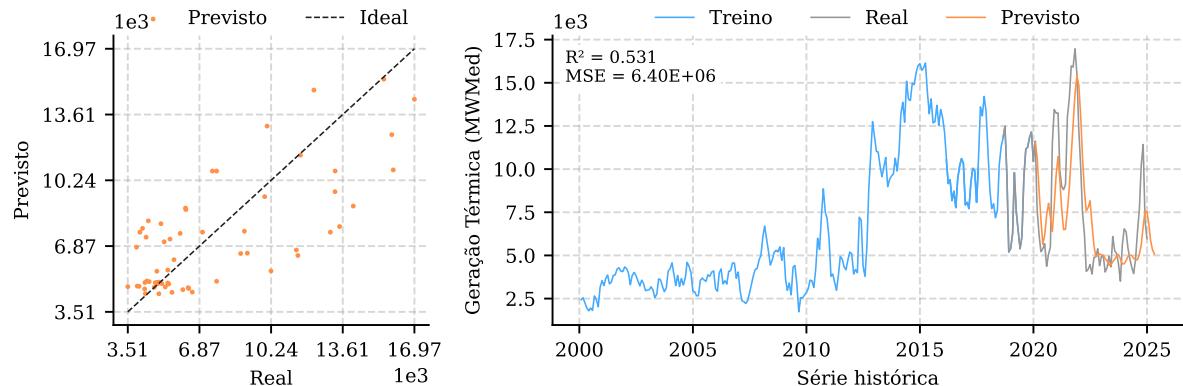
Figura 30 – Oneshot para fonte térmica (contexto/previsão: 90/30 semanas)



Fonte: o autor.

As Figuras ?? e ?? apresentam os resultados da previsão *oneshot* para a fonte térmica, considerando as janelas de contexto e previsão de 90/30 semanas e 512/96 dias, respectivamente. A geração térmica apresentou o pior resultado entre as fontes, com um valor de R^2 de 0,183 para o caso de 90/30 semanas e 0,531 para o caso de 512/96 dias. Ainda assim, os valores são consideravelmente superiores em comparação com os os modelos linear e não linear.

Figura 31 – Oneshot para fonte térmica (contexto/previsão: 512/96 dias)

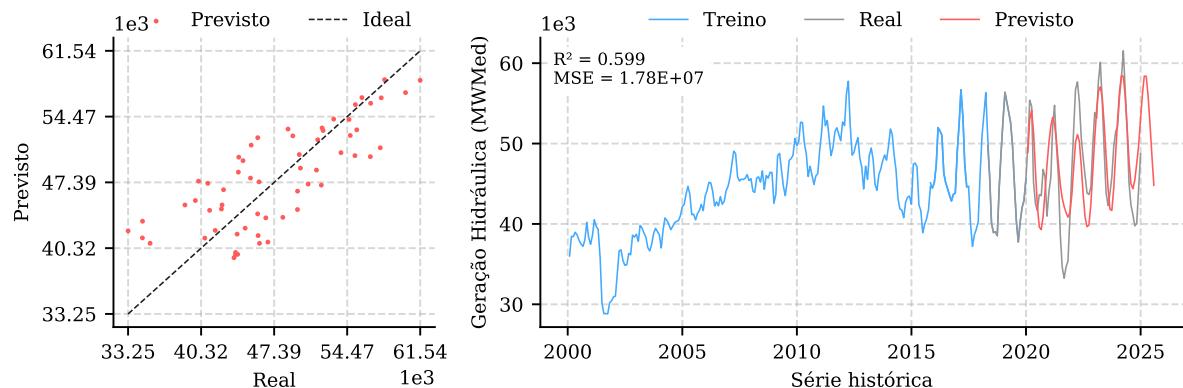


Fonte: o autor.

5.3.2 Impacto e Análise do Ajuste Fino

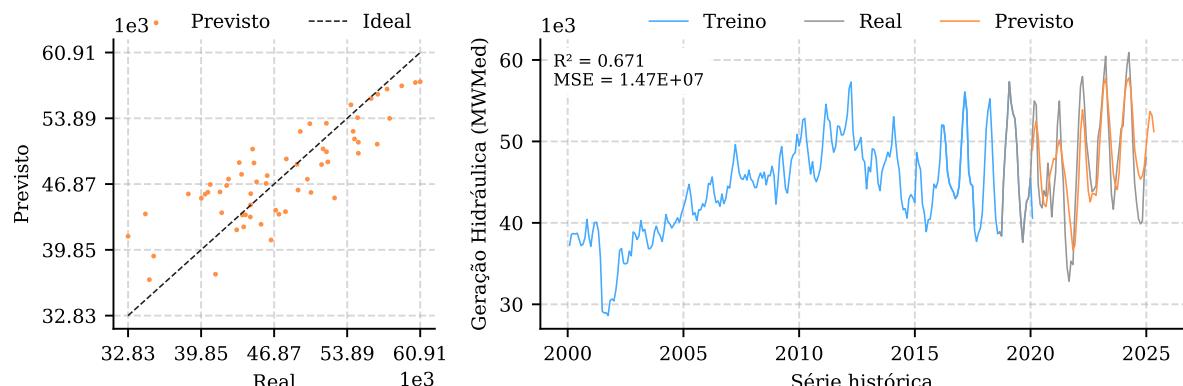
5.3.2.1 Fonte Hidráulica

Figura 32 – Finetune para fonte hidráulica (contexto/previsão: 90/30 semanas)



Fonte: o autor.

Figura 33 – Finetune para fonte hidráulica (contexto/previsão: 512/96 dias)



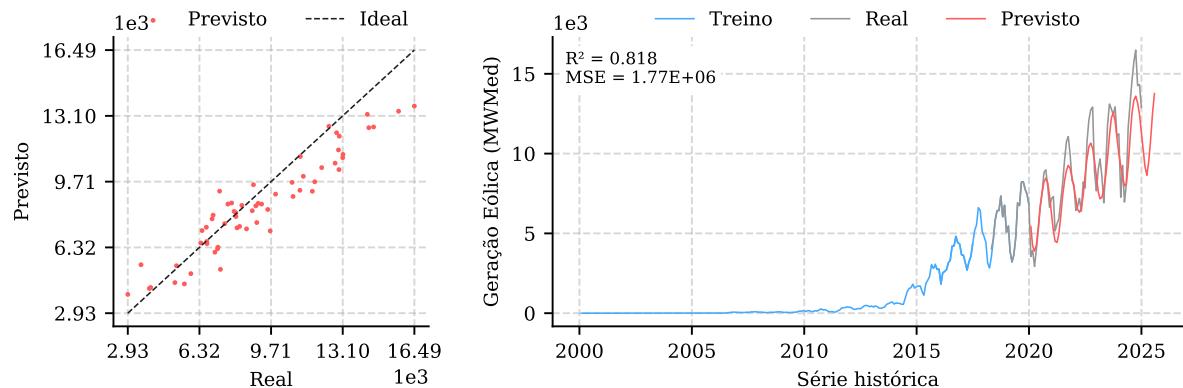
Fonte: o autor.

As Figuras ?? e ?? apresentam os resultados do ajuste fino para a fonte hidráulica, considerando as janelas de contexto e previsão de 90/30 semanas e 512/96 dias, respectivamente.

Observa-se que o ajuste fino melhorou em 4,53% o valor de R^2 para o caso de 90/30 semanas, passando de 0,573 para 0,599. Para o caso de 512/96 dias, o ajuste fino melhorou em 15,49% o valor de R^2 , passando de 0,581 para 0,671. Os resultados indicam que o processo de ajuste dos pesos do modelo é eficaz em melhorar a performance do modelo.

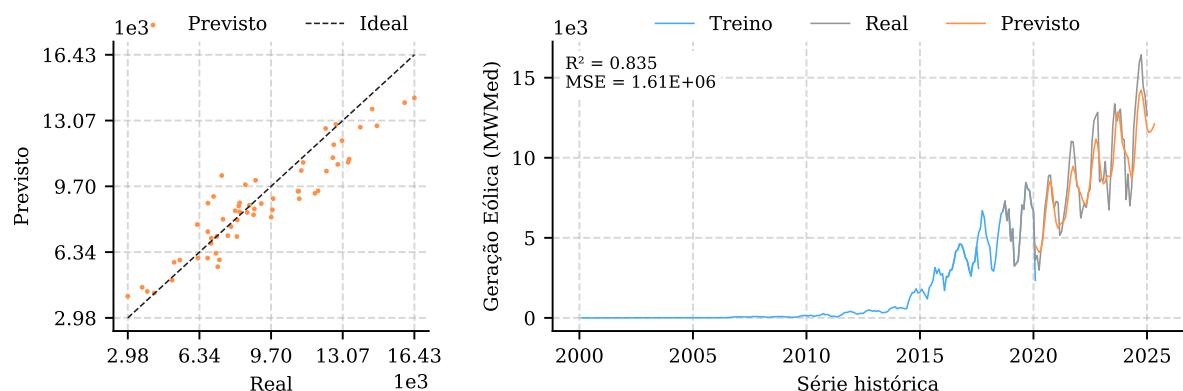
5.3.2.2 Fonte Eólica

Figura 34 - Finetune para fonte eólica (contexto/previsão: 90/30 semanas)



Fonte: o autor.

Figura 35 - Finetune para fonte eólica (contexto/previsão: 512/96 dias)



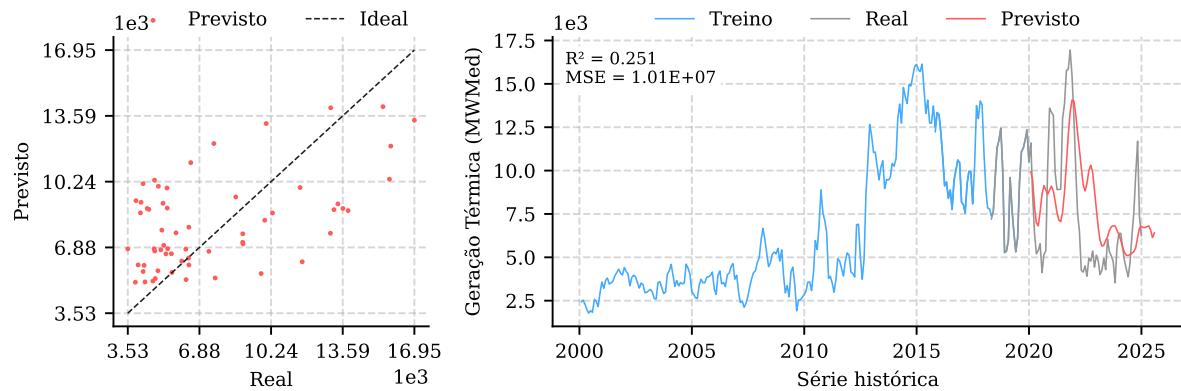
Fonte: o autor.

As Figuras ?? e ?? apresentam os resultados do ajuste fino para a fonte eólica, considerando as janelas de contexto e previsão de 90/30 semanas e 512/96 dias, respectivamente.

Observa-se que o ajuste fino melhorou em 19,76% o valor de R^2 para o caso de 90/30 semanas, passando de 0,683 para 0,818. Para o caso de 512/96 dias, o ajuste fino melhorou em 1,09% o valor de R^2 , passando de 0,826 para 0,835. Os resultados também indicam que o processo de ajuste dos pesos foi eficaz para a fonte eólica, em especial para o caso de 90/30 semanas.

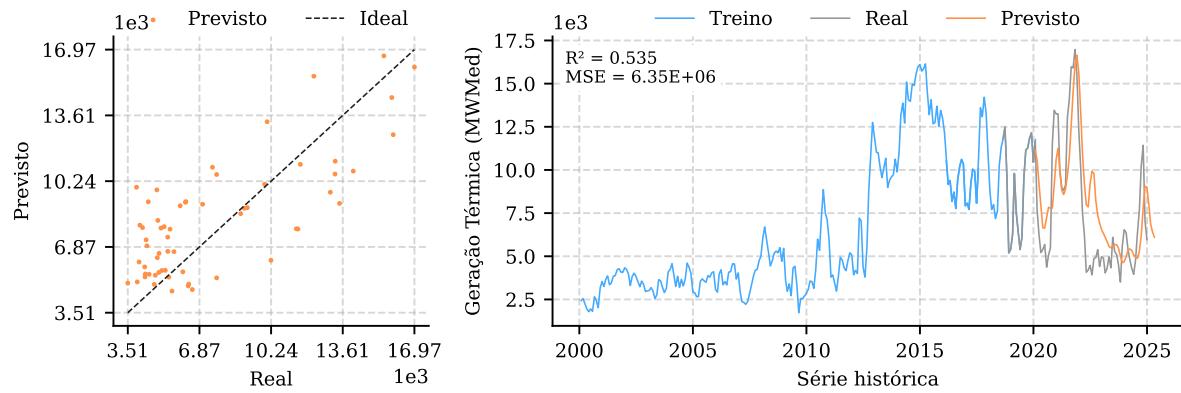
5.3.2.3 Fonte Térmica

Figura 36 – Finetune para fonte térmica (contexto/previsão: 90/30 semanas)



Fonte: o autor.

Figura 37 – Finetune para fonte térmica (contexto/previsão: 512/96 dias)



Fonte: o autor.

As Figuras ?? e ?? apresentam os resultados do ajuste fino para a fonte térmica, considerando as janelas de contexto e previsão de 90/30 semanas e 512/96 dias, respectivamente.

A fonte térmica apresentou melhoria de 37,15% no valor de R^2 para o caso de 90/30 semanas, passando de 0,183 para 0,251. Para o caso de 512/96 dias, o ajuste fino não apresentou melhoria significativa.

5.3.3 Análise Geral

Os resultados demonstram uma clara hierarquia de performance entre os modelos testados. Foi evidenciado que modelos lineares e não-lineares tradicionais são insuficientes para capturar a complexa dinâmica das gerações eólica e térmica, sendo amplamente superados pelo modelo TTM. As Tabelas ??, ??, ?? e ?? resumem as métricas de performance dos modelos testados, incluindo os valores de R^2 e MSE para cada fonte de geração.

Tabela 6 – Métricas de performance dos modelos

Modelo	Hidráulica		Eólica		Térmica	
	R^2	MSE	R^2	MSE	R^2	MSE
Linear	0,519	$2,22 \times 10^7$	0,004	$7,78 \times 10^6$	-2,322	$4,84 \times 10^7$
Não Linear	0,525	$2,19 \times 10^7$	-2,318	$2,59 \times 10^7$	-0,068	$1,56 \times 10^7$
Neural Oneshot	0,581	$1,87 \times 10^7$	0,826	$1,69 \times 10^6$	0,531	$6,63 \times 10^6$
Neural Finetune	0,671	$1,47 \times 10^7$	0,835	$1,61 \times 10^6$	0,535	$6,35 \times 10^6$

Fonte: o autor.

Tabela 7 – Comparaçao percentual de MSE para a fonte hidráulica

Modelo	Linear	Não Linear	Neural Oneshot	Neural Finetune
Linear	-	-1,37%	-18,72%	-51,02%
Não Linear	+1,35%	-	-17,11%	-48,98%
Neural Oneshot	+15,77%	+14,61%	-	-27,21%
Neural Finetune	+33,78%	+32,88%	+21,39% -	-

Fonte: o autor.

Tabela 8 – Comparaçao percentual de MSE para a fonte eólica

Modelo	Linear	Não Linear	Neural Oneshot	Neural Finetune
Linear	-	+232,91%	-360,36%	-383,23%
Não Linear	-232,91%	-	+93,47%	+93,78%
Neural Oneshot	+78,28%	-93,47%	-	+4,73%
Neural Finetune	+79,31%	-93,78%	-4,97%	-

Fonte: o autor.

Tabela 9 – Comparaçao percentual de MSE para a fonte térmica

Modelo	Linear	Não Linear	Neural Oneshot	Neural Finetune
Linear	-	-210,26%	-630,02%	-662,20%
Não Linear	+67,77%	-	+57,50%	+59,29%
Neural Oneshot	+86,30%	-135,29%	-	+4,22%
Neural Finetune	+86,88%	-145,67%	-4,41%	-

Fonte: o autor.

Capítulo 6

Conclusão

Este trabalho teve como objetivo central investigar e quantificar o impacto de variáveis climáticas externas, associadas ao fenômeno ENSO, na estimativa da geração de energia das fontes hidráulica, térmica e eólica no SIN. Para tal, foram implementados e comparados modelos computacionais de complexidade crescente, desde regressões lineares e não-lineares até uma arquitetura de rede neural pré-treinada, avaliando o ganho de performance ao especializar o modelo com os dados do ENSO.

Foi demonstrado que modelos de regressão tradicionais, tanto lineares quanto o Random Forest, são insuficientes para capturar a dinâmica complexa das fontes eólica e térmica, apresentando valores de R^2 insatisfatórios e até negativos, o que valida a hipótese de não-linearidade das relações envolvidas. A introdução do modelo neural pré-treinado TTM representou um salto qualitativo significativo, especialmente para a fonte eólica, que atingiu um R^2 de 0,826, evidenciando o poder de generalização dessa arquitetura.

Foram observados ganhos de performance substanciais ao realizar o ajuste fino do modelo neural, o que responde afirmativamente à questão central da pesquisa: a incorporação de variáveis climáticas externas, aliada a uma arquitetura neural adequada, aprimora de forma mensurável e significativa a estimativa da geração energética.

Reconhece-se, contudo, as limitações deste estudo. A análise se restringiu às variáveis do fenômeno ENSO, enquanto outras variáveis climáticas e indicadores econômicos poderiam ser incorporados para enriquecer os modelos. Adicionalmente, foi utilizada uma arquitetura neural específica (TSMixer), e a exploração de outras arquiteturas, como as baseadas em Transformers ou LSTMs, poderia trazer resultados distintos.

Para trabalhos futuros, sugere-se a expansão do conjunto de variáveis exógenas, incluindo outros índices climáticos relevantes para o território brasileiro e indicadores macroeconômicos. Além disso, a aplicação da metodologia de fine-tuning de modelos pré-treinados a outras tarefas, como a previsão de carga ou de preços de energia (PLD), poderá ser estudada.

Apêndice A: Códigos

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
from sklearn.metrics import mean_squared_error as mse_error
from sklearn.preprocessing import StandardScaler

import matplotlib_config

# Entrada das séries
geracao = (
    pd.read_csv(
        'Exportado/geracao_fontes_diario_MWmed.csv',
        parse_dates = ['Data'])
    .set_index('Data')
)

carga = (
    pd.read_csv(
        'Exportado/carga_subsist_diario_MWmed.csv',
        parse_dates = ['Data'])
    .set_index('Data')
)
```

```

enso = (
    pd.read_csv(
        'Exportado/ECMWF/derived-era5-single-levels-daily-statistics_sea_surface_temperature_reanalysis.csv',
        parse_dates = ['Data'])
    .set_index('Data')
)

geracao = geracao.fillna(0) # sem geração eólica e fotovoltaica no início dos anos 2000
geracao = geracao.drop(columns = [cols for cols in geracao.columns if 'Fotovoltaica' in cols or 'Outras' in cols])

# Feature engineering
carga['ano'] = carga.index.year
carga['mes'] = carga.index.month
carga['dia_mes'] = carga.index.day
carga['dia_semana'] = carga.index.dayofweek
carga['dia_ano'] = carga.index.dayofyear
carga['quadrimestre'] = carga.index.quarter

dataset = pd.concat([geracao, carga, enso], axis = 1).dropna()
dataset = dataset[dataset.index.year > 2009]
dataset = dataset[dataset.index.year < 2025]

target_cols = geracao.columns.to_list()
exog_cols = carga.columns.append(enso.columns).to_list()
print(f'Colunas objetivo:\n {target_cols}')
print(f'\nColunas exógenas:\n {exog_cols}')
print(dataset.head())

# Normalização e separação das séries de treino/teste
scaler = StandardScaler()
dataset[exog_cols] = scaler.fit_transform(dataset[exog_cols])

X_train, X_test, y_train, y_test = train_test_split(

```

```

dataset[exog_cols], dataset[target_cols],
test_size = .3, shuffle = False
)

# Aplicação do modelo
modelo = LinearRegression(n_jobs = -1)
reg = modelo.fit(X_train, y_train)
print(f'Coeficientes do modelo: {reg.coef_}')
print(f'Intercepto do modelo: {reg.intercept_}')
y_pred = reg.predict(X_test)
y_pred = pd.DataFrame(y_pred, columns = target_cols, index = y_test.index)
y_pred[y_pred < 0] = 0

# Resample mensal para visualização
y_pred = y_pred.resample('ME').mean()
y_test = y_test.resample('ME').mean()
y_train = y_train.resample('ME').mean()

layout = [['a', 'b']]
for col in target_cols:
    r2 = r2_score(y_test[col], y_pred[col])
    mse = mse_error(y_test[col], y_pred[col])

    fig, ax = plt.subplot_mosaic(layout, figsize = (7, 2.5), width_ratios = [1, 2])

    ax['a'].scatter(y_test[col], y_pred[col], s = 2, color = "#E24C4C", alpha = .5, label = 'Previsto')
    ax['a'].plot([y_test[col].min(), y_test[col].max()], [y_test[col].min(), y_test[col].max()],
                color = "#202020", linewidth = .66, ls = '--', label = 'Ideal')

    ax['a'].set_xticks(np.linspace(y_test[col].min(), y_test[col].max(), num = 5))
    ax['a'].set_yticks(np.linspace(y_test[col].min(), y_test[col].max(), num = 5))
    ax['a'].legend(loc = 'upper center', bbox_to_anchor = (.5, 1.15), ncol = 2, frameon = False, fancybox = True)
    ax['a'].ticklabel_format(axis = 'both', style = 'sci', scilimits = (3, 3))

```

```
ax['a'].set_xlabel('Real')
ax['a'].set_ylabel('Previsto')

ax['b'].plot(y_train.index, y_train[col], label = 'Treino', color = "#43AAFF", linewidth = .66)
ax['b'].plot(y_test.index, y_test[col], label = 'Real', color = "#969696", linewidth = .66, ls =
ax['b'].plot(y_pred.index, y_pred[col], label = 'Previsão', color = "#E24C4C", linewidth = .66)

ax['b'].legend(loc = 'upper center', bbox_to_anchor = (.5, 1.15), ncol = 3, frameon = False, fancybox = True)
ax['b'].set_xlabel('Série histórica')
ax['b'].set_ylabel(f'Geração {col.replace('_', ' ')} (MWMed)')
ax['b'].ticklabel_format(axis = 'y', style = 'sci', scilimits = (3, 3))

ax['b'].text(.02, .95, (f'R2 = {r2:.3f}\nMSE = {mse:.2E}'),
             transform = ax['b'].transAxes, verticalalignment = 'top', fontsize = 7,
             bbox = dict(boxstyle = 'square', facecolor = 'white', edgecolor = 'none'))

plt.tight_layout()
# plt.show()

plt.savefig(f'Graficos/Linear/linear_{col}_mensal_CARGA.svg', bbox_inches = 'tight')
```

Apêndice B: Resultados por Subsistema

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.