



INSTITUTO FEDERAL DE CIÊNCIA E TECNOLOGIA DE PERNAMBUCO

Campus Garanhuns

Coordenação de Engenharia Elétrica

GABRIEL DA SILVA BELÉM

**PREVISÃO DE CURTO PRAZO DA DEMANDA DE ENERGIA ELÉTRICA NO
NORDESTE BRASILEIRO UTILIZANDO SÉRIES TEMPORAIS**

Garanhuns

2024

GABRIEL DA SILVA BELÉM

**PREVISÃO DE CURTO PRAZO DA DEMANDA DE ENERGIA ELÉTRICA NO
NORDESTE BRASILEIRO UTILIZANDO SÉRIES TEMPORAIS**

Trabalho de conclusão de curso apresentado à
Coordenação de Engenharia Elétrica do
Instituto Federal de Ciência e Tecnologia de
Pernambuco, como requisito para obtenção do
título de Bacharel em Engenharia Elétrica. Sob
orientação da Profa. Ma. **Camila Almeida
Diniz.**

Garanhuns

2024

B428p

Belém, Gabriel da Silva.

Previsão de curto prazo da demanda de energia elétrica no nordeste brasileiro utilizando séries temporais / Gabriel da Silva Belém ; orientador Camila Almeida Diniz, 2024.

75f. : il.

Orientador: Camila Almeida Diniz.

Trabalho de Conclusão de Curso (Graduação) – Instituto Federal de Pernambuco. Pró-Reitoria de Ensino. Diretoria de Ensino. Campus Garanhuns. Coordenação do Curso Superior em Engenharia. Curso de Bacharelado em Engenharia Elétrica, 2024.

1. Energia elétrica – Previsão - Brasil, Nordeste – Modelos matemáticos. 2. Análise de séries temporais. 3. Redes neurais (Computação) . I. Título.

CDD 621.31

Andréa Maria Lidington Lins –CRB4/868

**PREVISÃO DE CURTO PRAZO DA DEMANDA DE ENERGIA ELÉTRICA NO
NORDESTE BRASILEIRO UTILIZANDO SÉRIES TEMPORAIS**

Trabalho aprovado. Garanhuns, 19 de setembro de 2024.

Profa. Ma. Camila Almeida Diniz (orientadora)

Prof. Dr. Diego Soares Lopes (examinador interno)

Prof. Me. Geronimo Barbosa Alexandre (examinador externo)

Garanhuns

2024

Dedico este trabalho a três pessoas extraordinárias que sempre estiveram ao meu lado com amor incondicional e apoio inabalável. Ao meu querido avô, cuja sabedoria e gentileza são fontes de inspiração diária. À minha amada avó, cujo carinho e presença constante me ensinaram o verdadeiro significado de ternura e cuidado. E à minha mãe, que é o meu exemplo de força, coragem e perseverança. Sem o amor e o suporte de vocês, esta jornada não teria sido possível. Obrigado por serem a luz constante no meu caminho. Amo vocês profundamente.

AGRADECIMENTOS

Agradeço a Deus pelo dom da vida, pelo privilégio de conviver com pessoas maravilhosas e por sua proteção constante.

Aos meus familiares, deixo um agradecimento especial. Meu avô, Antônio de França Belém e minha avó, Elza da Silva Belém, por toda a sabedoria e ensinamentos transmitidos ao longo dos anos, e por sempre acreditarem em mim. Cada conselho, cada palavra de incentivo e cada gesto de amor foram indispensáveis para minha formação acadêmica e pessoal. Minha mãe, Maria Cristina da Silva Belém, com seu apoio incondicional e amor eterno, foi a base sólida sobre a qual construí minha trajetória. Suas orientações e suporte foram essenciais para que eu pudesse prosseguir com determinação e confiança.

À minha namorada, Brunna dos Santos Oliveira, sou imensamente grato pelo apoio inabalável e pelo amor que você me proporcionou durante toda essa jornada. Seu carinho, compreensão e companheirismo foram fontes de grande conforto e motivação. Você esteve ao meu lado em cada passo deste caminho, e sua presença constante foi uma fonte inestimável de força e encorajamento. Agradeço profundamente por tudo que você fez por mim.

Também quero estender meus agradecimentos à família dela, que me acolheu com tanto carinho e consideração. A maneira como vocês me receberam e me apoiaram fez toda a diferença, e eu sou muito grato por todo o calor e acolhimento que me proporcionaram.

“Honestidade não é virtude: é obrigação”.
Antônio de França Belém

RESUMO

Visando uma gestão eficiente e sustentável dos sistemas elétricos, é essencial ter uma previsão da demanda de energia, principalmente devido à crescente integração de fontes renováveis e às mudanças climáticas. Posto isso, o presente estudo objetivou desenvolver um algoritmo para prever a demanda de energia elétrica no nordeste brasileiro utilizando técnicas de análise de séries temporais e aprendizado de máquina, com foco no modelo MLP (Perceptron de Multicamadas). Empregou-se a metodologia CRISP (Processo Padrão Intersetorial) para estruturar processos de análise preditiva, garantindo uma abordagem sistemática desde a compreensão do problema até a implementação e monitoramento do modelo. Os experimentos avaliaram o desempenho de diferentes cenários de previsões da carga elétrica para horizontes de 24, 72, 168 e 720 horas. Foram criados três modelos: (i) modelo contendo uma camada de 24 neurônios; (ii) modelo contendo duas camadas: a primeira com 24 neurônios e a segunda com 48 neurônios; e (iii) modelo contendo três camadas: a primeira com 24 neurônios, a segunda com 48 neurônios e a terceira 96 neurônios. Dentre os modelos, os que possuem duas camadas ocultas usando o otimizador Nadam e função de ativação Linear apresentou o melhor desempenho geral com um Erro Absoluto Percentual Médio (MAPE) de 1,49% e um Coeficiente de Determinação (R^2) de 0,94 para previsões de 24 horas, indicando alta precisão. Esses modelos também foram eficazes para horizontes de 72 e 168 horas, com um MAPE de 1,79% e 1,87% e R^2 de 0,92 e 0,91, respectivamente. No entanto, o modelo enfrentou dificuldades para o horizonte de 720 horas, apresentando um MAPE de 3,17% e R^2 de 0,77, exibindo menor precisão e maior tempo de processamento. Os resultados destacam a importância da escolha adequada de otimizadores e funções de ativação para maximizar a precisão e a eficiência dos modelos. A arquitetura otimizada do banco de dados contribuiu para a alta performance do modelo, evidenciando a necessidade de testes contínuos de configurações para aprimorar a previsão de longo prazo.

Palavras-chave: Inteligência Artificial; Metodologia CRISP; MLP; Previsão de Demanda; Redes Neurais Artificiais

ABSTRACT

Aiming for efficient and sustainable management of electrical systems, it is essential to forecast energy demand, especially due to the increasing integration of renewable sources and climate change. Accordingly, this study aimed to develop an algorithm to predict electricity demand in the Brazilian Northeast using time series analysis techniques and machine learning, focusing on the MLP (Multilayer Perceptron) model. The CRISP methodology (Cross-Industry Standard Process for Data Mining) was employed to structure predictive analysis processes, ensuring a systematic approach from problem understanding to model implementation and monitoring. The experiments evaluated the performance of different scenarios for forecasting electrical load for horizons of 24, 72, 168, and 720 hours. Three models were created: (i) a model with one layer of 24 neurons; (ii) a model with two layers: the first with 24 neurons and the second with 48 neurons; and (iii) a model with three layers: the first with 24 neurons, the second with 48 neurons, and the third with 96 neurons. Among the models, those with two hidden layers using the Nadam optimizer and a linear activation function demonstrated the best overall performance, achieving a Mean Absolute Percentage Error (MAPE) of 1.49% and a Coefficient of Determination (R^2) of 0.94 for 24-hour forecasts, indicating high accuracy. These models were also effective for 72- and 168-hour horizons, with MAPE values of 1.79% and 1.87%, and R^2 values of 0.92 and 0.91, respectively. However, the model faced challenges for the 720-hour horizon, presenting a MAPE of 3.17% and R^2 of 0.77, showing lower precision and longer processing time. The results highlight the importance of choosing appropriate optimizers and activation functions to maximize model accuracy and efficiency. The optimized database architecture contributed to the model's high performance, underscoring the need for continuous testing of configurations to improve long-term forecasting.

Keywords: Artificial Intelligence; CRISP Methodology; MLP; Demand Forecasting; Artificial Neural Networks.

LISTA DE ABREVIACOES

AIC	Criterio de Informaco de Akaike
ADAM	Estimativa Adaptativa de Momentos
ARIMA	Media Movel Integrada Auto-Regressiva
BIC	Criterio de Informaco Bayesiano
CNN	Rede Neural Convolutacional
CRISP-DM	Processo Padro Intersetorial para Minerao de Dados
IA	Inteligncia Artificial
LSTM	Memria de Longo e Curto Prazo
MAPE	Erro Mdio Percentual Absoluto
ML	Aprendizado de Mquina
MLP	Perceptron de Mltiplas Camadas
MMGD	Mini e Microgeraco Distribuda
MSE	Erro Quadrtico Mdio
NADAM	Estimativa Adaptativa de Momentos Acelerada por Nesterov
ONS	Operador Nacional do Sistema Eltrico
RELU	Unidade Linear Retificada
RMSPROP	Propagaco da Raiz do Erro Quadrtico Mdio
RNA	Rede Neural Artificial
RNN	Rede Neural Recorrente
SARIMA	SARIMA Sazonal
SIN	Sistema Interligado Nacional

LISTA DE FIGURAS

Figura 2.1 – Quatro exemplos de séries temporais mostrando padrões diferentes.	24
Figura 2.2 – Curva de Carga do SIN.	26
Figura 2.3 – Representação em diagrama de blocos do sistema nervoso.	29
Figura 2.4 – Representação de um neurônio biológico	30
Figura 2.5 – Representação de um neurônio artificial.	30
Figura 2.6 – Rede neural feedforward MLP.	32
Figura 2.7 – Esquemático da metodologia CRISP	36
Figura 3.1 – Gráfico referente à carga horária do Nordeste no mês de janeiro de 2023.	39
Figura 3.2 – Preparação dos dados.	40
Figura 3.3 – Estrutura base do algoritmo de previsão.	44
Figura 4.1 – Caso 1: Evolução do erro com o passar das épocas (uma camada oculta).	47
Figura 4.2 – Caso 1: Comparação entre a previsão e os dados reais (uma camada oculta).	47
Figura 4.3 – Caso 1: Evolução do erro com o passar das épocas (duas camadas ocultas).	48
Figura 4.4 – Caso 1: Comparação entre a previsão e os dados reais (duas camadas ocultas).	48
Figura 4.5 – Caso 1: Evolução do erro com o passar das épocas (três camadas ocultas).	49
Figura 4.6 – Caso 1: Comparação entre a previsão e os dados reais (três camadas ocultas).	49
Figura 4.7 – Caso 2: Evolução do erro com o passar das épocas (uma camada oculta).	50
Figura 4.8 – Caso 2: Comparação entre a previsão e os dados reais (uma camada oculta).	50
Figura 4.9 – Caso 2: Evolução do erro com o passar das épocas (duas camadas ocultas).	51
Figura 4.10 – Caso 2: Comparação entre a previsão e os dados reais (duas camadas ocultas).	51
Figura 4.11 – Caso 2: Evolução do erro com o passar das épocas (três camadas ocultas).	52
Figura 4.12 – Caso 2: Comparação entre a previsão e os dados reais (três camadas ocultas).	52

Figura 4.13 – Caso 3: Evolução do erro com o passar das épocas (uma camada oculta).	53
Figura 4.14 – Caso 3: Comparação entre a previsão e os dados reais (uma camada oculta).	53
Figura 4.15 – Caso 3: Evolução do erro com o passar das épocas (duas camadas ocultas).	53
Figura 4.16 – Caso 3: Comparação entre a previsão e os dados reais (duas camadas ocultas).	53
Figura 4.17 – Caso 3: Evolução do erro com o passar das épocas (três camadas ocultas).	54
Figura 4.18 – Caso 3: Comparação entre a previsão e os dados reais (três camadas ocultas).	54
Figura 4.19 – Caso 4: Evolução do erro com o passar das épocas (uma camada oculta).	55
Figura 4.20 – Caso 4: Comparação entre a previsão e os dados reais (uma camada oculta).	55
Figura 4.21 – Caso 4: Evolução do erro com o passar das épocas (duas camadas ocultas).	55
Figura 4.22 – Caso 4: Comparação entre a previsão e os dados reais (duas camadas ocultas).	55
Figura 4.23 – Caso 4: Evolução do erro com o passar das épocas (três camadas ocultas).	56
Figura 4.24 – Caso 4: Comparação entre a previsão e os dados reais (três camadas ocultas).	56
Figura 4.25 – Caso 5: Evolução do erro com o passar das épocas (uma camada oculta).	57
Figura 4.26 – Caso 5: Comparação entre a previsão e os dados reais (uma camada oculta).	57
Figura 4.27 – Caso 5: Evolução do erro com o passar das épocas (duas camadas ocultas).	58
Figura 4.28 – Caso 5: Comparação entre a previsão e os dados reais (duas camadas ocultas).	58
Figura 4.29 – Caso 5: Evolução do erro com o passar das épocas (três camadas ocultas).	58

Figura 4.30 – Caso 5: Comparação entre a previsão e os dados reais (três camadas ocultas).	58
Figura 4.31 – Caso 6: Evolução do erro com o passar das épocas (uma camada oculta).	59
Figura 4.32 – Caso 6: Comparação entre a previsão e os dados reais (uma camada oculta).	59
Figura 4.33 – Caso 6: Evolução do erro com o passar das épocas (duas camadas ocultas).	60
Figura 4.34 – Caso 6: Comparação entre a previsão e os dados reais (duas camadas ocultas).	60
Figura 4.35 – Caso 6: Evolução do erro com o passar das épocas (três camadas ocultas).	61
Figura 4.36 – Caso 6: Comparação entre a previsão e os dados reais (três camadas ocultas).	61
Figura 4.37 – Cenário 1: Evolução do erro com o passar das épocas.	64
Figura 4.38 – Cenário 1: Comparação entre a previsão e os dados reais.	64
Figura 4.39 – Cenário 2: Evolução do erro com o passar das épocas.	65
Figura 4.40 – Cenário 2: Comparação entre a previsão e os dados reais.	65
Figura 4.41 – Cenário 3: Evolução do erro com o passar das épocas.	66
Figura 4.42 – Cenário 3: Comparação entre a previsão e os dados reais.	66

LISTA DE QUADROS

Quadro 3.1 – Código referente à implementação das variáveis de teste e treinamento.	45
Quadro 3.2 – Código referente à implementação da rede neural MLP com uma camada oculta.	46

LISTA DE TABELAS

Tabela 3.1 – Demanda elétrica por macrorregião com amostragem horária.	39
Tabela 3.2 – Base de dados utilizado para a RNA.....	41
Tabela 4.1 – Caso 1: MAPE, R^2 e tempo de processamento.	47
Tabela 4.2 – Caso 2: MAPE, R^2 e tempo de processamento.	50
Tabela 4.3 – Caso 3: MAPE, R^2 e tempo de processamento.	52
Tabela 4.4 – Caso 4: MAPE, R^2 e tempo de processamento.	54
Tabela 4.5 – Caso 5: MAPE, R^2 e tempo de processamento.	56
Tabela 4.6 – Caso 6: MAPE, R^2 e tempo de processamento.	59
Tabela 4.7 – Comparação entre os valores de MAPE, R^2 e tempo de processamento dos casos para a rede com duas camadas ocultas.....	63
Tabela 4.8 – Cenário 1: MAPE, R^2 e tempo de processamento.	64
Tabela 4.9 – Cenário 2: MAPE, R^2 e tempo de processamento.	65
Tabela 4.10 – Cenário 3: MAPE, R^2 e tempo de processamento.	66
Tabela 4.11 – MAPE, R^2 e tempo de processamento dos três cenários.....	67

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Motivação	19
1.2	Objetivos	19
1.2.1	Objetivo Geral	19
1.2.2	Objetivos Específicos	19
1.3	Estrutura do Trabalho	20
2	FUNDAMENTAÇÃO TEÓRICA	21
2.1	A Demanda Energética do Nordeste Brasileiro	21
2.2	Séries Temporais	22
2.3	Previsão de Carga	25
2.3.1	Classificação da Previsão de Carga	27
2.4	Aprendizado de Máquina em Séries Temporais	28
2.4.1	Redes Neurais Artificiais	29
2.4.2	Perceptron de Multicamadas	31
2.5	Otimizadores	33
2.6	Software e Ferramentas para Análise de Séries Temporais	34
2.7	Metodologia CRISP-DM	35
3	METODOLOGIA	38
3.1	Compreensão do Problema	38
3.2	Compreensão dos dados	38
3.3	Preparação dos dados	40
3.4	Modelagem	42
3.5	Avaliação	43
3.6	Implementação	44

4	RESULTADOS E DISCUSSÕES	47
4.1	Caso 1 – Otimizador: Adam; Função de Ativação: Linear	47
4.2	Caso 2 – Otimizador: Adam; Função de Ativação: ReLU	50
4.3	Caso 3 – Otimizador: Nadam; Função de Ativação: Linear	52
4.4	Caso 4 – Otimizador: Nadam; Função de Ativação: ReLU	54
4.5	Caso 5 – Otimizador: RMSprop; Função de Ativação: Linear	56
4.6	Caso 6 – Otimizador: RMSprop; Função de Ativação: ReLU	59
4.7	Análise comparativa	61
4.8	Cenário 1 – Previsão de 72 horas	63
4.9	Cenário 2 – Previsão de 168 horas	64
4.10	Cenário 3 – Previsão de 720 horas	65
4.11	Comentários	67
5	CONSIDERAÇÕES FINAIS	68
6	REFERÊNCIAS	70
7	APÊNDICES	75
	Apêndice A – Link GitHub	75

1 INTRODUÇÃO

Ser capaz de prever a demanda por um produto ou serviço é uma técnica estratégica que pode auxiliar a tomada de decisões, o planejamento, a gestão de estoque, como a otimização de recursos, por exemplo, e até mesmo a satisfação do cliente. São muitas as aplicações em que esse tipo de ferramenta pode ser aplicado e em empresas dos mais diversos setores (EPE, 2020; Achermann et al., 2022).

A análise de séries temporais é uma abordagem fundamental para a previsão de demanda em diversos contextos empresariais. Em particular, permite o estudo de padrões e variações ao longo do tempo, essenciais para entender comportamentos sazonais, tendências de mercado e ciclos de consumo (Lopes, 2023). Empresas podem utilizar modelos como ARIMA (*Auto-Regressive Integrated Moving Average*) e SARIMA (*Seasonal ARIMA*) para capturar essas nuances temporais e projetar com maior precisão as necessidades futuras. Esses métodos não apenas fornecem *insights* sobre o volume de vendas esperado, mas também ajudam na elaboração de estratégias proativas para ajustar a produção, melhorar a eficiência operacional e, conseqüentemente, fortalecer a competitividade no mercado (Montgomery et al., 2020; García et al., 2019).

No contexto energético, a previsão de demanda elétrica desempenha um papel crucial na gestão eficiente e sustentável de sistemas elétricos modernos (Pires, 2016; Pessanha e CEPTEL, 2013). O Nordeste é caracterizado por uma variabilidade climática significativa, que inclui longos períodos de seca intercalados com chuvas intensas, e uma alta dependência de fontes de energia renováveis, como as energias eólica e solar (França, 2019; Santos, 2021). Essa variabilidade climática e a intermitência das fontes renováveis representam desafios adicionais na previsão da demanda de energia

Com o advento da IA (Inteligência Artificial), especialmente através de técnicas avançadas como MLP (*Multilayer Perceptron*) e LSTM (*Long Short-Term Memory*), novas abordagens têm sido exploradas para melhorar a precisão e a robustez dos modelos de previsão (Tolstikhin et al., 2021; Ramesh e Ramkumar, 2024; Wang, 2022). Estes estudos investigam como essas técnicas podem ser aplicadas de forma eficaz na previsão de carga elétrica, aproveitando sua capacidade de capturar padrões complexos e dinâmicos nos dados históricos de consumo.

Além disso, devido a sua natureza variável e distribuída, a MGD (mini e microgeração distribuída), também possuem impactos significativos na previsão de carga elétrica (Duarte, 2021; MME, 2023). Enquanto tradicionalmente a previsão de carga

lidava principalmente com grandes centrais elétricas, a incorporação crescente de sistemas distribuídos introduz novos desafios e oportunidades (Falcão, 2013; Oliveira, 2023). Métodos avançados como MLP e LSTM se destacam nesse contexto. MLP é capaz de aprender padrões complexos de dados históricos, enquanto LSTM é especialmente eficaz na modelagem de séries temporais com dependências de longo prazo, permitindo previsões mais precisas e adaptáveis às flutuações imprevisíveis da geração distribuída. Essas técnicas não apenas melhoram a precisão das previsões de carga, mas também facilitam a integração harmoniosa de fontes de energia variáveis, promovendo a eficiência e a sustentabilidade dos sistemas elétricos modernos.

Em geral, os estudos de previsão têm se concentrado predominantemente em dois métodos principais: os modelos estatísticos, que tradicionalmente formulam a curva de carga matematicamente e são amplamente utilizados na previsão global da carga do sistema; e os métodos de inteligência computacional, enfocados neste estudo, que se destacam pela capacidade de aprendizado e adaptação (Reis, 2007; Soares, 2013).

Estes métodos oferecem uma abordagem flexível na resolução de problemas ao capturar as complexas interações entre variáveis para alcançar previsões mais precisas. Apesar do desenvolvimento de sistemas estatísticos adaptativos que continuam aprendendo durante o uso, os modelos de inteligência computacional têm consistentemente demonstrado um desempenho superior (Spiliotis, 2023; Luo et al., 2020).

A crescente demanda por energia, juntamente com a necessidade de integrar fontes renováveis intermitentes, impõe desafios significativos aos sistemas de previsão tradicionais (ENERGIA, 2024; MME, 2024). A IA surge como uma solução promissora devido à sua capacidade de aprender padrões não lineares e capturar dependências temporais complexas, características essenciais para previsões precisas em cenários variáveis e dinâmicos como os sistemas elétricos (Manyika e Bughin, 2018; IBM, 2024).

O objetivo deste trabalho é desenvolver um algoritmo avançado de previsão de carga elétrica utilizando análise de séries temporais e técnicas de aprendizado de máquina, especificamente MLP. Essas abordagens serão aplicadas para capturar padrões complexos e dinâmicos nos dados históricos de consumo elétrico, visando melhorar significativamente a precisão das previsões. O algoritmo proposto busca não apenas otimizar a gestão de sistemas elétricos, mas também enfrentar os desafios impostos pela integração crescente de fontes renováveis intermitentes, contribuindo assim para uma operação mais eficiente e sustentável.

1.1 Motivação

A motivação para este trabalho surge da necessidade premente e crescente de prever com precisão a carga elétrica em sistemas modernos, especialmente diante da complexidade introduzida pela integração de fontes renováveis intermitentes, como solar e eólica. A utilização de técnicas avançadas como redes neurais artificiais, em análise de séries temporais oferece uma oportunidade única para melhorar significativamente a precisão das previsões. Isso não só permite uma gestão mais eficiente e sustentável dos recursos energéticos, mas também facilita a integração harmoniosa de fontes de energia variáveis, contribuindo assim para a adaptação dos sistemas elétricos às demandas variáveis e dinâmicas do mercado atual. Ao desenvolver um sistema robusto que integra essas metodologias, espera-se não apenas otimizar operações em tempo real, mas também fornecer informações valiosas para o planejamento estratégico de longo prazo no setor energético regional.

1.2 Objetivos

1.2.1 Objetivo Geral

Desenvolver um algoritmo de previsão de curto prazo da demanda elétrica no Nordeste do Brasil, utilizando a metodologia CRISP-DM em uma rede MLP.

1.2.2 Objetivos Específicos

Os objetivos específicos deste trabalho são:

- i. Coletar e preparar dados históricos confiáveis de carga elétrica no Nordeste brasileiro;
- ii. Realizar uma análise exploratória dos dados para identificar padrões sazonais e tendências relevantes;
- iii. Implementar um modelo de rede neural artificial usando Perceptron de Múltiplas Camadas para realizar a previsão de curto prazo da demanda de energia elétrica no Nordeste brasileiro;
- iv. avaliar o desempenho dos modelos utilizando métricas apropriadas como o Erro Médio Percentual Absoluto (MAPE), Coeficiente de Determinação (R^2) e o Tempo de Processamento.

1.3 Estrutura do Trabalho

Este trabalho está estruturado em cinco capítulos, começando por este capítulo introdutório.

No Capítulo 2, encontra-se a fundamentação teórica que aborda o estudo da arte sobre séries temporais, incluindo as principais técnicas utilizadas na previsão de carga, além de uma discussão sobre o modelo de Rede Neural Artificial (RNA).

No Capítulo 3, é apresentada a metodologia CRISP-DM. Também são feitas considerações sobre os métodos utilizados para a obtenção dos resultados.

O Capítulo 4, é dedicado à exposição e análise dos resultados.

No Capítulo 5, são apresentadas as considerações finais, acompanhadas de sugestões de trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, são apresentados os conceitos teóricos fundamentais para a compreensão da metodologia utilizada neste trabalho, e da análise e discussão dos resultados obtidos. São abordados os conceitos de séries temporais, previsão de carga, técnicas de previsão por aprendizado de máquina, bem como os *softwares* e ferramentas empregados na análise.

2.1 A Demanda Energética do Nordeste Brasileiro

O Nordeste brasileiro engloba nove estados (Maranhão, Piauí, Ceará, Rio Grande do Norte, Paraíba, Pernambuco, Alagoas, Sergipe e Bahia). A abrangência territorial dessa região corresponde a cerca de 18% do território brasileiro e a população em 2022, segundo o IBGE, era de 54.658.515, representando 26,9% do total de habitantes do Brasil (IBGE, 2024; Marques, 2010).

A dinâmica populacional e econômica do Nordeste é influenciada por fatores como a agricultura, o turismo e a indústria, que desempenham papéis cruciais no desenvolvimento regional. O Nordeste é conhecido pela produção de culturas como cana-de-açúcar, algodão e frutas tropicais, que são vitais para a economia local e nacional (IBGE, 2023). O Nordeste, com sua rica herança cultural e potencial econômico, continua a ser uma área vital para o futuro do Brasil, destacando a necessidade de investimentos em infraestrutura e educação para fortalecer seu papel na economia nacional (Silva, 2022; Santos e Almeida, 2021).

O crescimento econômico de um país está fortemente ligado à demanda energética, uma vez que a energia é essencial para a produção industrial, a construção de infraestrutura e a oferta de serviços. À medida que uma economia se expande, a necessidade de energia para máquinas, transporte e atividades diárias aumenta, refletindo diretamente na capacidade produtiva e na competitividade do país. Segundo a IEA (2020), a expansão da produção industrial e a urbanização geram um aumento significativo no consumo de energia, criando uma correlação entre crescimento econômico e demanda energética.

De acordo com Carregosa (2024), a geração de energia elétrica tem aumentado a uma taxa superior ao seu consumo, mas grande parte dessa energia não é devidamente aproveitada. Como resultado, durante os picos de demanda, é necessário acionar usinas

termelétricas para garantir o fornecimento. Essa situação ocorre porque as fontes de energia renováveis, como a solar e a eólica, não conseguem oferecer uma produção contínua e estável, dependendo de condições climáticas variáveis.

Prever a demanda por energia é fundamental para garantir a segurança do abastecimento e a eficiência do sistema energético. Essa previsão permite que governos e empresas planejem adequadamente a infraestrutura necessária, como usinas e redes de distribuição, evitando assim crises de abastecimento que podem impactar a economia e a qualidade de vida da população. Além disso, uma previsão com um bom grau de assertividade auxilia em investimentos em fontes renováveis e tecnologias emergentes, contribuindo para a sustentabilidade e a redução das emissões de carbono. Em um cenário de crescente urbanização e consumo, a capacidade de antecipar a demanda energética torna-se essencial para um desenvolvimento equilibrado e resiliente.

2.2 Séries Temporais

As séries temporais desempenham um papel fundamental em uma variedade de disciplinas, fornecendo compreensões cruciais sobre a evolução de fenômenos ao longo do tempo. De forma geral, uma série temporal pode ser definida como um conjunto de observações organizadas em ordem cronológica (Makridakis et al., 1998; Hyndman e Athanasopoulos, 2018). Segundo Souza e Camargo (2004), essas observações são registradas em intervalos de tempo regular e podem abranger uma ampla gama de fenômenos, incluindo aspectos físicos, biológicos, econômicos e sociais.

A análise de séries temporais, conforme discutido por Box e Jenkins (2008), parte da suposição fundamental de que existe um sistema causal, em geral constante, que influenciou os dados no passado e pode influenciá-los no futuro. Esse sistema causal frequentemente gera padrões não aleatórios, discerníveis em um gráfico da série temporal ou através de métodos estatísticos.

Além da compreensão teórica das séries temporais, diversos modelos estatísticos são empregados para sua análise e previsão. Dentre esses, destacam-se os modelos ARIMA e SARIMA: O ARIMA é utilizado principalmente para séries que não apresentam sazonalidade, permitindo capturar a dinâmica dos dados através de suas componentes autorregressiva, integrada e de média móvel. Por outro lado, o SARIMA expande essa abordagem ao incorporar a sazonalidade, sendo especialmente útil para

dados que mostram padrões sazonais, como vendas ao longo do ano ou temperaturas mensais.

Esses modelos são ferramentas poderosas na previsão de séries temporais, como evidenciado por suas aplicações em áreas que vão da economia à meteorologia. A escolha adequada dos parâmetros dos modelos ARIMA e SARIMA é essencial para garantir previsões precisas, sendo que métodos como o AIC e o BIC são frequentemente utilizados para essa finalidade (Hyndman & Athanasopoulos, 2018).

O AIC é calculado como:

$$AIC = 2k - 2\ln(L) \quad (2.1)$$

Onde k é o número de parâmetros do modelo e L é a verossimilhança máxima. Um valor de AIC mais baixo indica um modelo melhor, pois sugere um bom ajuste aos dados com menor complexidade.

Já o BIC é dado pela seguinte equação:

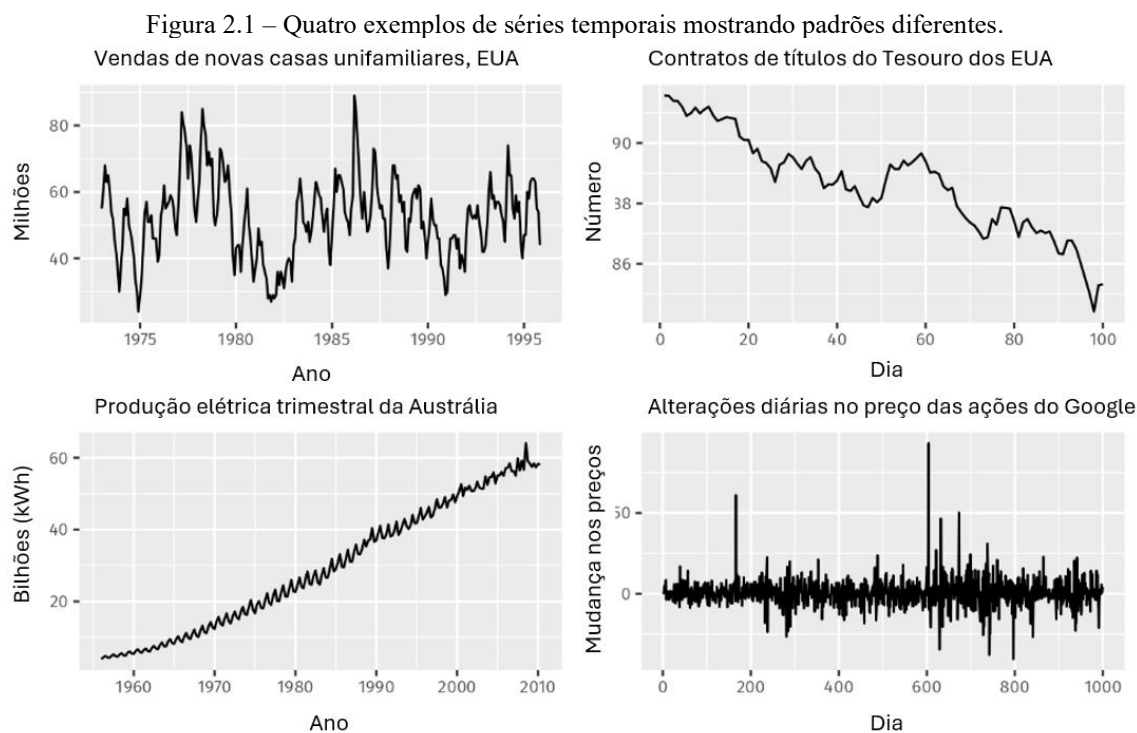
$$BIC = \ln(n)k - 2\ln(L) \quad (2.2)$$

onde n é o número de observações. Assim como no AIC, um valor mais baixo de BIC indica um modelo mais adequado. O BIC penaliza modelos complexos de forma mais severa do que o AIC, o que pode ser útil para evitar o sobreajuste.

Assim, ao analisar séries temporais, a aplicação de modelos estatísticos como ARIMA e SARIMA, juntamente com critérios como AIC e BIC, pode revelar padrões significativos e auxiliar na tomada de decisões informadas.

De acordo com Filho (2023), as séries temporais podem ser decompostas em quatro componentes principais:

1. Tendência: reflete o padrão de comportamento dos dados ao longo do tempo, indicando se a sequência é crescente, decrescente ou constante;
2. Sazonalidade: refere-se a variações nos dados que ocorrem de forma regular e repetitiva dentro de períodos específicos conhecidos;
3. Ciclo: representa uma variação dos dados que ocorre em períodos específicos, repetindo-se, embora não tenha uma frequência fixa como na sazonalidade.
4. Ruído (erro): é a variação aleatória dos dados que não pode ser explicada pelos outros componentes (exemplo: erro intrínseco a leitura de um termômetro).



Fonte: Adaptado de Hyndman e Athanasopoulos (2018).

É apresentado na Figura 2.1 combinações dessas componentes:

1. As vendas mensais de habitação exibem uma sazonalidade marcante dentro de cada ano, além de apresentarem um comportamento cíclico pronunciado com um período variando entre 6 e 10 anos. Não foi identificada uma tendência clara nos dados durante este período.
2. Os contratos de títulos do Tesouro dos EUA mostraram resultados consistentes do mercado de Chicago ao longo de 100 dias consecutivos de negociação em 1981. Neste caso, não há padrões sazonais, mas observa-se uma tendência de queda claramente definida.
3. A produção trimestral de eletricidade na Austrália demonstra uma tendência robusta de crescimento, acompanhada por variações sazonais significativas. Não foram identificados comportamentos cíclicos evidentes neste contexto.
4. A variação diária no preço de fechamento das ações do Google não exhibe tendência, sazonalidade ou comportamento cíclico discerníveis. As flutuações são predominantemente aleatórias e parecem pouco previsíveis, não revelando padrões consistentes que poderiam ser úteis para o desenvolvimento de modelos de previsão.

A capacidade de uma série temporal em prever valores futuros é uma das suas características mais importantes. A partir de um histórico de dados disponível até o momento t é possível utilizar essas informações para estimar o valor da variável no instante $t + 1$, conforme expresso pela equação 2.1:

$$x_{t+1} = f(x_t + x_{t-1} + \dots + x_{t-k}) + \epsilon_{t+1}. \quad (2.3)$$

Na qual x_{t+1} é o valor futuro da variável, f é uma função que captura a relação entre os valores passados e ϵ_{t+1} é o erro associado à previsão para o instante $t + 1$.

A análise de séries temporais não apenas descreve padrões passados, mas também proporciona métodos para previsão futura. Essas previsões são cruciais para orientar a tomada de decisões em uma variedade de áreas, desde o planejamento econômico até a gestão de operações e recursos (Loffredo, 2023).

Assim, tem-se que a análise de séries temporais é uma ferramenta poderosa para entender a dinâmica temporal de diversos fenômenos, permitindo não só a interpretação histórica, mas também a antecipação de eventos futuros com base em dados empíricos e modelos estatísticos robustos (Brockwell e Davis, 2002).

2.3 Previsão de Carga

A previsão de carga é um componente crucial para o planejamento e operação eficiente do sistema elétrico, desempenhando um papel fundamental na garantia da segurança e estabilidade do fornecimento de energia elétrica. Essa prática envolve a estimativa da demanda futura de energia com base em análises detalhadas de séries temporais históricas de consumo, considerando sazonalidades, variações climáticas e padrões de uso diário e semanal (Mamlook, 2009).

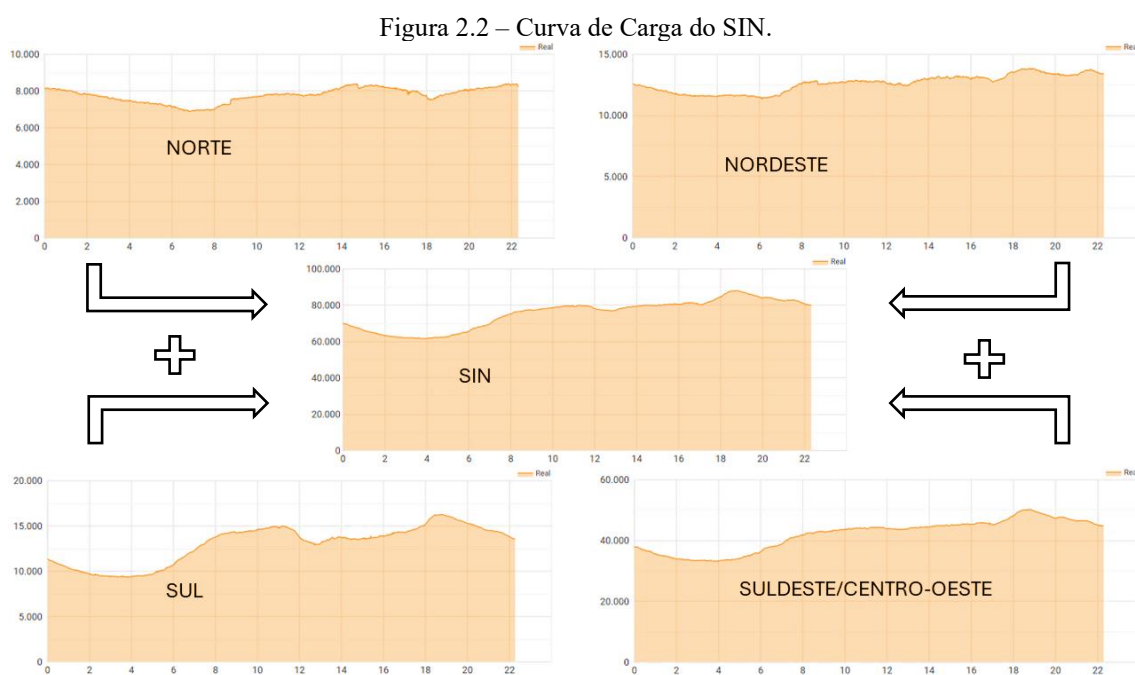
No contexto brasileiro, o ONS (Operador Nacional do Sistema Elétrico) utiliza modelos avançados como NEWAVE, DECOMP e DESSEM para realizar diferentes aspectos da previsão de carga, desde o planejamento de longo prazo até a operação em tempo real (CEPEL, 2024a).

O NEWAVE, por exemplo, é aplicado para prever a operação de usinas hidrelétricas e termelétricas, integrando dados sobre a disponibilidade de recursos hídricos e a demanda esperada. Já o DECOMP foca na decomposição da carga e na análise do sistema elétrico ao longo do tempo, permitindo uma compreensão detalhada das necessidades de expansão da infraestrutura, enquanto o DESSEM simula o despacho de

geração, considerando as restrições operacionais das usinas e as características específicas do sistema (CEPEL, 2024b; CEPEL, 2024c; CEPEL, 2024d).

Esses modelos não operam isoladamente; ao contrário, eles se complementam, integrando previsões de carga com variáveis críticas, como as previsões hidrológicas para usinas hidrelétricas, que são fundamentais, especialmente em um país onde a matriz energética é predominantemente hídrica. Além disso, eles consideram as restrições operacionais do sistema, como limites de transmissão e capacidade das usinas, permitindo que o ONS tome decisões informadas sobre o despacho de energia e o uso eficiente dos recursos disponíveis (ONS, 2022).

Gross e Galiana (1987) discutem que a carga total de um sistema de energia é a soma das demandas de todos os seus usuários. Embora teoricamente fosse viável prever o padrão de carga se o consumo de cada usuário fosse conhecido, na prática, as demandas individuais são aleatórias e imprevisíveis. No entanto, a soma dessas cargas individuais forma um padrão de consumo que pode ser estimado estatisticamente.



Fonte: Adaptado de ONS (2024).

A previsão de carga é, portanto, uma tarefa complexa que combina o comportamento imprevisível das cargas individuais com padrões estatísticos discerníveis a partir da agregação dessas demandas. No contexto operacional, como destacado por Gross e Galiana (1987), a soma das demandas individuais em todos os pontos do sistema define a carga total, cuja previsão é essencial para a operação eficiente do sistema elétrico. Essa previsão não apenas considera variações sazonais e climáticas, mas também integra

modelos avançados que incorporam previsões hidrológicas e outras variáveis críticas para otimizar o despacho de energia e garantir a estabilidade do fornecimento.

É apresentado na Figura 2.2 uma visão abrangente das cargas de cada subsistema do SIN (Sistema Interligado Nacional), revelando tendências individuais que podem ser previstas. A precisão das previsões de carga não apenas otimiza a operação do sistema elétrico, mas também contribui significativamente para a redução de custos operacionais e para a minimização de impactos ambientais associados à geração de energia. Antecipando variações na demanda, os agentes geradores podem ajustar a produção de energia de forma mais eficiente, reduzindo desperdícios e maximizando o uso de fontes renováveis com menor impacto ambiental (Kong et al., 2019).

Ainda segundo Gross e Galiana (1987), o comportamento da carga do sistema elétrico é influenciado por diversos fatores. Esses fatores podem ser agrupados em quatro categorias principais:

1. Fatores econômicos: referem-se às mudanças no padrão de consumo;
2. Fatores temporais: são padrões cíclicos ou sazonais que impactam o consumo;
3. Fatores climáticos: variáveis como chuva, velocidade do vento ou temperatura;
4. Fatores aleatórios: eventos imprevisíveis como greves, desligamentos inesperados da rede elétrica etc.

Esses elementos têm uma correlação significativa com os dados futuros da série temporal. Portanto, os métodos de previsão podem utilizar tanto as variáveis exógenas quanto os valores anteriores da série para inferir dados futuros. Como resultado, os modelos de previsão dependem não apenas de informações históricas, mas também da suposição de que eventos passados podem se repetir ou influenciar o futuro (Antunis, 2013).

2.3.1 Classificação da Previsão de Carga

Como foi apresentado, a previsão de carga temporal utiliza os fatores que influenciam o consumo de energia e os dados históricos para prever os dados futuros. Essa previsão pode ocorrer em diversas frentes e com diferentes objetivos. Deste modo, a depender do horizonte de tempo, a previsão de carga é dividida em quatro tipos (Medeiros, 2019; IBM, 2024; EPE, 2024; CCEE, 2024; ONS, 2024):

1. **Previsão de Longo Prazo:** é um processo de estimativa da demanda futura de energia que abrange períodos superiores a um ano, podendo variar de um a dez anos, conforme os diferentes enfoques. Essa previsão utiliza uma combinação de modelos econômicos, demográficos e análises de cenários futuros para projetar como o consumo de energia pode evoluir ao longo do tempo;
2. **Previsão de Médio Prazo:** horizonte de tempo de semanas a poucos anos. O objetivo deste tipo de previsão está voltado a operação do sistema ou mesmo planejamento de médio prazo. Os dados podem ser usados na tomada de decisões, desde armazenamento de combustível em usinas geradoras até programação de manutenções. O passo de tempo é da ordem de dias;
3. **Previsão de Curto Prazo:** previsões entre meia hora até alguns dias. Está muito relacionado e tem fundamental importância na operação econômica do sistema e na garantia de fornecimento. O passo de tempo das previsões são de até uma hora;
4. **Previsão de Curtíssimo Prazo:** previsão com antecedência de minutos, com passo de previsão da ordem de segundos. Este tipo de previsão é utilizado para operação e controle online do sistema.

2.4 **Aprendizado de Máquina em Séries Temporais**

O aprendizado de máquina, também conhecido como ML (*Machine Learning*), é um campo da ciência da computação que se insere dentro da IA. Ele se destaca por utilizar algoritmos que permitem aos computadores aprenderem a partir de dados, sem necessidade de programação explícita para cada possível cenário (Gomes, 2019). Essa capacidade de aprendizado automático é fundamental para a criação de modelos analíticos que podem reconhecer padrões nos dados e realizar previsões ou decisões com base nesses padrões (Waltrick, 2020). Em essência, o ML capacita os sistemas a modelarem eventos e reconhecerem padrões complexos a partir de grandes volumes de dados, tornando possível a automação de processos analíticos e a tomada de decisões com intervenção mínima humana (Silva, 2019).

A aplicação do ML possibilita a criação rápida e automática de modelos que lidam com grandes volumes de dados complexos, oferecendo resultados mais precisos e eficientes, mesmo em escala significativa (Waltrick, 2020). Isso significa que esses modelos podem ser utilizados para análise preditiva e tomada de decisões em uma ampla variedade de campos.

No contexto deste trabalho, o foco reside na aplicação específica de técnicas avançadas de aprendizado de máquina para a previsão de carga elétrica. Em particular, serão explorados métodos baseados em redes neurais artificiais, como o MLP.

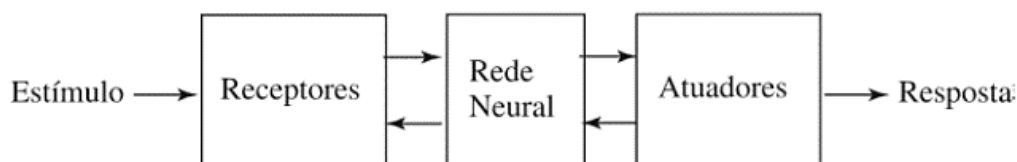
2.4.1 Redes Neurais Artificiais

Uma RNA (rede neural artificial) é um modelo computacional inspirado na forma como o cérebro humano processa informações. Ela é usada para reconhecer padrões, fazer previsões e tomar decisões baseadas em dados (Shah, 2024).

Shah (2024) define redes neurais artificiais como sistemas de processamento de dados que imitam o funcionamento do sistema neural, permitindo a identificação de relações não lineares em grandes volumes de dados. Esses dados podem ser provenientes de diversas fontes sensoriais e se manifestar em diferentes formatos, como texto, imagens ou áudio.

A melhor maneira de entender como uma rede neural artificial funciona é compreendendo como uma rede neural natural no cérebro funciona e traçando um paralelo entre elas. Neurônios são o componente fundamental do cérebro humano e são responsáveis pelo aprendizado e pela retenção de conhecimento e informações como conhecemos (Santos, 2008; Souza, 2023).

Figura 2.3 – Representação em diagrama de blocos do sistema nervoso.



Fonte: Arbib, 1987.

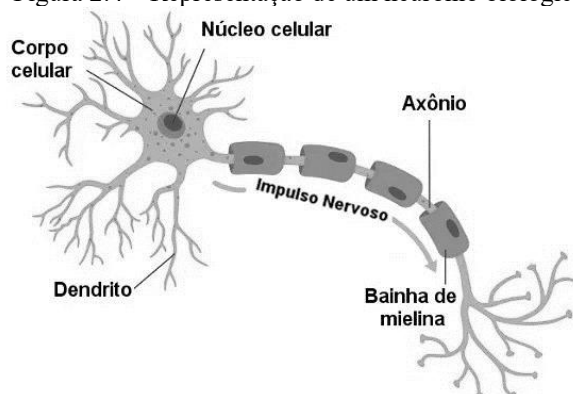
É apresentado na Figura 2.3 o sistema nervoso humano como um sistema integrado de três componentes principais, conforme descrito por Arbib (1987): receptores, a rede neural (cérebro) e atuadores. Os receptores são responsáveis por detectar estímulos do ambiente externo ou interno e convertê-los em impulsos elétricos. Esses impulsos são transmitidos ao cérebro, onde são processados e avaliados. Após o processamento, o cérebro envia comandos aos atuadores, que então convertem os impulsos elétricos em respostas físicas ou comportamentais visíveis. Assim, o sistema nervoso opera em um ciclo contínuo: os estímulos são recebidos, processados e resultam em respostas apropriadas do corpo. Esse processo permite que o organismo reaja de maneira adequada

a mudanças internas e externas, mantendo a coordenação e o equilíbrio essenciais para a adaptação e sobrevivência.

Haykin (2000) descreve o neurônio como uma unidade essencial de processamento de informações em uma rede neural. Eles recebem dados sensoriais como entrada, processam-nos e fornecem os dados de saída utilizados por outros neurônios.

É apresentado na Figura 2.4 o neurônio biológico, composto por dendritos que captam os estímulos de outros neurônios e os transmitem ao corpo celular. O dendrito avalia se a informação deve ser repassada através das sinapses e dos axônios, que estabelecem conexão com os dendritos dos neurônios subsequentes (Santos, 2008; Haykin, 2000).

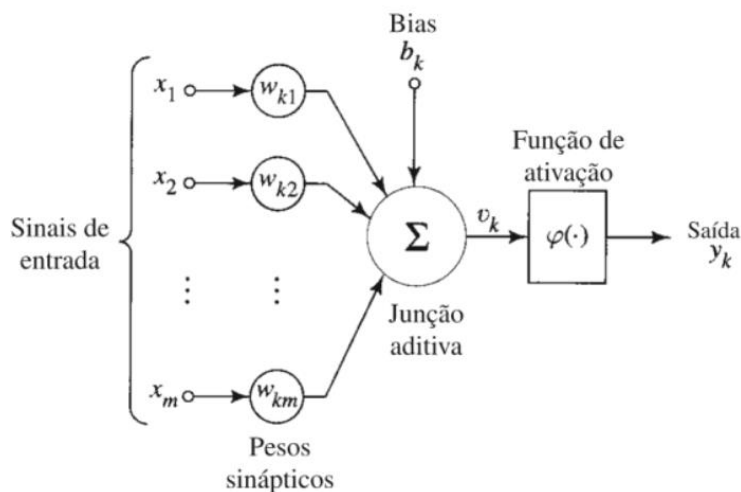
Figura 2.4 – Representação de um neurônio biológico



Fonte: Santos, 2008.

É apresentado na Figura 2.5 o neurônio artificial, que de maneira semelhante ao neurônio real, recebe os estímulos da entrada, determina se um novo estímulo deve ou não ser gerado e, caso deva, o envia para o próximo neurônio.

Figura 2.5 – Representação de um neurônio artificial.



Fonte: Haykin, 2000.

Os estímulos x_m recebidos na entrada são multiplicados pelos pesos sinápticos ω_{km} – ao contrário de uma sinapse do cérebro, o peso sináptico de um neurônio artificial pode estar em um intervalo que inclui valores negativos e positivos. Os resultados são somados e passados para uma função de ativação responsável por restringir a amplitude de saída do neurônio – o intervalo normalizado da amplitude da saída de um neurônio é escrito como o intervalo unitário fechado $[0, 1]$ ou alternativamente $[-1, 1]$ (Haykin, 2000).

Ainda na Figura 2.5, é possível perceber a presença de um *bias* b_k que possui o efeito de aumentar ou diminuir a entrada líquida da função de ativação, dependendo se ele for positivo ou negativo. Ele serve para ajustar o valor da ativação, permitindo que o modelo se adapte melhor aos dados e aprenda padrões mais complexos (Haykin, 2000; Nielsen, 2019).

Em termos matemáticos, pode-se descrever um neurônio k pelas seguintes equações:

$$\mu = \sum_{j=1}^m \omega_{kj}x_j, \quad (2.4)$$

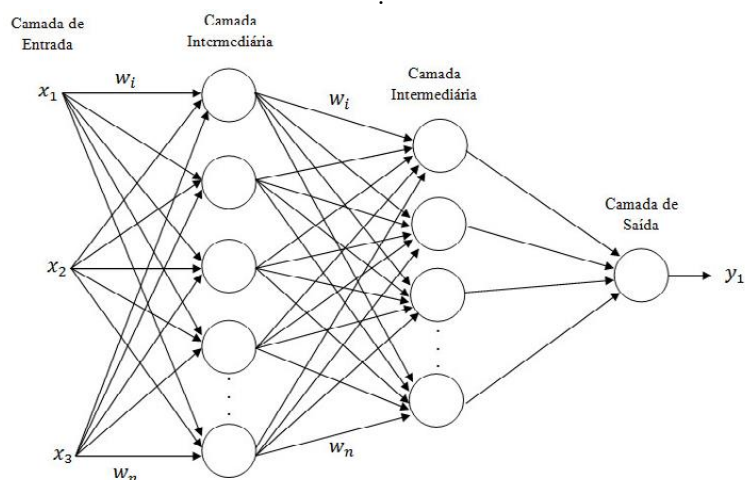
$$y_k = \varphi(u_k + b_k). \quad (2.5)$$

Na Equação 2.2 tem-se que: x_j representa os sinais de entrada; ω_j são os pesos sinápticos do neurônio k ; e u_k é a saída do combinador linear devido aos sinais de entrada. Já na Equação 2.3: b_k são os bias; $\varphi(\cdot)$ é a função de ativação; e y_k é o sinal de saída do neurônio.

2.4.2 Perceptron de Multicamadas

Uma Rede Neural MLP é um tipo fundamental de rede neural *feedforward*, representando uma das formas mais básicas e amplamente aplicadas de redes neurais artificiais (RNA). Nesta arquitetura, a informação flui de maneira unidirecional, movendo-se da camada de entrada para a camada de saída, passando por uma ou mais camadas ocultas (Jaiswal, 2024).

Figura 2.6 – Rede neural feedforward MLP.



Fonte: Delgado e Ventura, 2020.

As redes MLP têm se destacado por sua eficácia em uma ampla gama de aplicações práticas. Elas são utilizadas para tarefas como classificação de padrões, reconhecimento de padrões e controle de processos, além de desempenharem um papel crucial em sistemas de reconhecimento de voz e imagem. Sua utilidade se estende ao controle e processamento de sinais em sistemas complexos, graças à sua habilidade de modelar relações não lineares e aprender representações sofisticadas a partir de grandes volumes de dados (Rumelhart, Hinton, & Williams, 1986; Bishop, 2006; Haykin, 2009).

A estrutura de uma rede MLP, como ilustrado na Figura 2.6, é composta por três tipos principais de camadas, conforme a seguir.

1. Camada de Entrada: A primeira camada da rede, responsável por receber e introduzir os dados a serem processados. Cada neurônio nesta camada representa uma característica ou variável dos dados de entrada.
2. Camadas Ocultas: As camadas intermediárias, ou camadas ocultas, estão localizadas entre a camada de entrada e a camada de saída. Essas camadas realizam o processamento das informações recebidas da camada de entrada, aplicando transformações e extraindo características significativas. Elas são compostas por neurônios que realizam cálculos ponderados e aplicam funções de ativação para introduzir não linearidades no modelo.
3. Camada de Saída: A camada final da rede neural. Ela recebe os resultados do processamento realizado pelas camadas ocultas e produz a saída final da rede. A configuração dos neurônios nesta camada depende do tipo de tarefa que a rede está realizando, como regressão, classificação binária etc.

Essas camadas trabalham em conjunto para transformar os dados de entrada em uma saída útil, aprendendo e ajustando os pesos das conexões entre os neurônios durante o processo de treinamento. O sucesso das redes MLP em diversas aplicações pode ser atribuído à sua capacidade de capturar padrões complexos e relações não lineares nos dados, possibilitando soluções precisas e eficientes para problemas variados (Freiman, 2004).

2.5 Otimizadores

A escolha do otimizador adequado pode influenciar significativamente a eficácia e a velocidade de convergência de um modelo. Entre os otimizadores mais proeminentes estão o Adam, o Nadam e o RMSprop, cada um com características únicas que os tornam apropriados para diferentes tipos de problemas e dados (Kingma e Ba, 2014).

Dentre os otimizadores utilizados, estão:

1. Adam (*Adaptive Moment Estimation*): um otimizador amplamente utilizado em treinamento de redes neurais, conhecido por sua capacidade de combinar as vantagens do algoritmo de Gradiente Descendente com Momentos (*Momentum*) e o algoritmo de Gradiente Descendente com Escalonamento Adaptativo de Taxa de Aprendizado (RMSprop). O Adam ajusta a taxa de aprendizado de forma adaptativa para cada parâmetro, utilizando estimativas dos momentos de primeira e segunda ordem do gradiente. Isso geralmente resulta em uma convergência mais rápida e estável em comparação com métodos tradicionais (Kingma e Ba, 2014).
2. Nadam (*Nesterov-accelerated Adaptive Moment Estimation*): uma variante do Adam que incorpora a aceleração de Nesterov. Ele combina as propriedades adaptativas do Adam com a abordagem de aceleração de Nesterov, onde o gradiente é estimado levando em conta a direção de aceleração. Isso pode proporcionar uma atualização mais eficiente dos pesos e uma convergência potencialmente mais rápida ao minimizar a função de custo (Ruder, 2016).
3. RMSprop (*Root Mean Square Propagation*): é um otimizador que ajusta a taxa de aprendizado para cada parâmetro, normalizando o gradiente pelo quadrado da média móvel das atualizações recentes. Ele é projetado para lidar com o problema de decaimento da taxa de aprendizado e é eficaz para treinar redes neurais em tarefas que envolvem dados não estacionários ou muito variados. O RMSprop

pode ser particularmente útil quando o treinamento enfrenta dificuldades de otimização devido a gradientes muito grandes ou muito pequenos (Goodfellow et.al, 2016).

A função linear é a mais simples, porém raramente utilizada nas camadas ocultas, sendo mais apropriada para camadas de saída em problemas de regressão. Em contraste, a função ReLU (Rectified Linear Unit) tem se destacado por sua eficiência e capacidade de resolver problemas como o gradiente desvanecido, permitindo que redes neurais aprendam representações complexas dos dados (Nair & Hinton, 2010).

1. Linear: a mais simples, onde a saída é diretamente proporcional à entrada. Embora raramente seja usada em camadas ocultas devido à sua limitação em modelar não-linearidades complexas, a função linear pode ser aplicada na camada de saída para problemas de regressão, onde o objetivo é prever um valor contínuo (Nair & Hinton, 2010).
2. ReLU: é uma função de ativação não-linear que retorna o valor da entrada se for positivo e zero caso contrário. Essa função é amplamente utilizada em redes neurais devido à sua simplicidade e eficiência computacional. A ReLU ajuda a resolver o problema do gradiente desvanecido e permite que as redes neurais aprendam representações complexas e não-lineares dos dados (Nair & Hinton, 2010).

2.6 Software e Ferramentas para Análise de Séries Temporais

Na análise de séries temporais, a escolha do software é essencial para obter resultados precisos e eficientes. O *Python* se destaca como uma ferramenta preferencial, especialmente quando usado em conjunto com o *Jupyter*. Essa combinação permite unir código, visualizações e documentação em um único documento, ideal para análises complexas (McKinney, 2023; Driscoll, 2018).

O *Jupyter* facilita a experimentação, permitindo a execução de trechos de código em células independentes. Isso é particularmente útil na previsão da demanda elétrica, pois possibilita o ajuste dinâmico de parâmetros e a visualização imediata dos resultados. Ferramentas como *Pandas* e *NumPy* são fundamentais para manipulação e processamento de dados, enquanto bibliotecas como *Matplotlib* e *Seaborn* permitem criar gráficos

detalhados para validar modelos e identificar padrões sazonais (Pandas, 2024; Numpy, 2024; Matplotlib, 2024; Seaborn, 2024).

Além disso, bibliotecas como *Statsmodels* e *Prophet* oferecem técnicas avançadas de modelagem e previsão, como ARIMA e modelos de suavização exponencial, permitindo a aplicação de técnicas estatísticas robustas. A *scikit-learn* também se destaca ao suportar redes neurais, capturando padrões complexos e não lineares nas séries temporais (Hyndman e Athanasopoulos, 2018).

A capacidade do *Jupyter* de documentar todo o processo, com descrições e explicações sobre as escolhas de modelagem, torna o trabalho mais acessível a outros pesquisadores. Assim, a combinação de *Python* e *Jupyter* fornece um ambiente robusto e flexível para a implementação de modelos de previsão, especialmente em contextos dinâmicos como o da demanda elétrica, onde a interatividade e a visualização desempenham papéis cruciais (Jupyter, 2024).

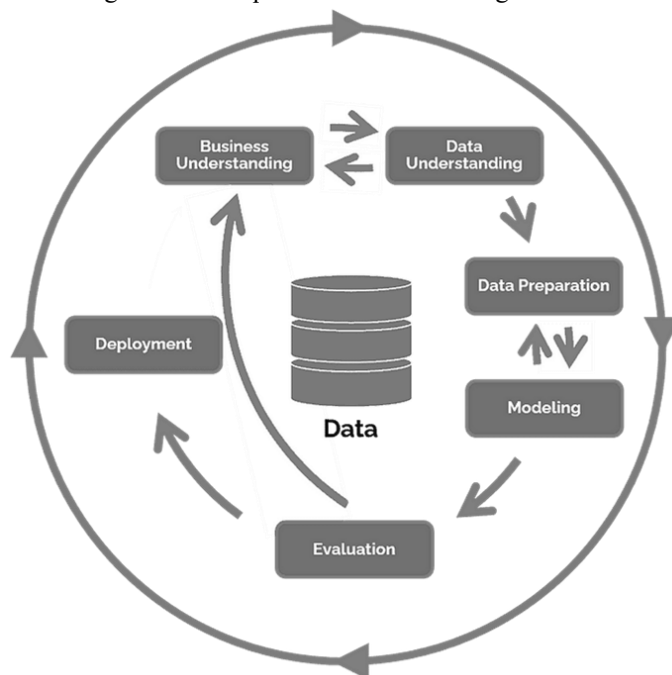
Uma das principais vantagens do *Python* é sua natureza livre e de código aberto, que oferece um acesso gratuito ao código-fonte e a uma ampla gama de pacotes desenvolvidos pela comunidade. Isso não só reduz os custos associados a programas proprietários como o *MATLAB*, mas também fomenta a inovação e permite uma personalização mais ampla das ferramentas para atender às necessidades específicas dos projetos (Python, 2024; Sanchez, 2024).

Embora o *Python* ofereça uma solução poderosa e flexível, é importante destacar que a utilização eficaz da linguagem requer um sólido conhecimento técnico. Além disso, o desempenho pode ser impactado ao lidar com grandes volumes de dados. Portanto, enquanto *Python* e suas bibliotecas ofereçam um conjunto robusto de ferramentas para a análise de séries temporais, a proficiência na sua aplicação e a gestão adequada dos dados são fundamentais para obter sucesso (Python, 2024; Sanchez, 2024).

2.7 Metodologia CRISP-DM

A metodologia CRISP é uma abordagem estruturada para a realização de projetos de mineração de dados. Ela foi desenvolvida para fornecer um *framework* que pode ser aplicado a diferentes indústrias e tipos de dados, facilitando a análise e a extração de informações valiosas. Sua utilização é amplamente empregada em projetos de ciência de dados e análise preditiva (Souza, 2023; Pellegrino, 2020; Roberto, 2023).

Figura 2.7 – Esquemático da metodologia CRISP



Fonte: Souza, 2023.

Conforme apresentado no Figura 2.7, a metodologia CRISP-DM é dividida em seis fases principais:

1. **Compreensão do Problema:** Esta fase envolve a definição dos objetivos do projeto e a compreensão do contexto. É essencial entender o problema e como a análise de dados pode contribuir para resolver esse problema.
2. **Compreensão dos Dados:** Após entender o problema, a próxima etapa é coletar e explorar os dados disponíveis. Isso inclui a avaliação da qualidade dos dados, a identificação de padrões e a compreensão das variáveis relevantes.
3. **Preparação dos Dados:** Nesta fase, os dados são limpos e transformados para que estejam prontos para análise. Isso pode incluir a remoção de dados inconsistentes, a transformação de variáveis e a criação de novos atributos.
4. **Modelagem:** Nessa etapa, diferentes técnicas de modelagem são aplicadas para criar modelos preditivos ou descritivos. Isso pode envolver a seleção de algoritmos, a construção de modelos e a avaliação de seu desempenho.
5. **Avaliação:** Após a construção dos modelos, é necessário avaliar os resultados para garantir que atendam aos objetivos de negócios. Esta fase pode envolver a validação dos modelos e a interpretação dos resultados.

6. Implantação: A fase final é a implementação dos modelos e das soluções desenvolvidas no ambiente real. Isso pode envolver a criação de relatórios, a integração com sistemas existentes e a comunicação dos resultados para os interessados.

A metodologia CRISP-DM é iterativa, o que significa que pode ser necessário revisar e ajustar as fases conforme o projeto avança e novas informações são descobertas. Ela é amplamente valorizada por sua flexibilidade e adaptabilidade a diferentes tipos de problemas e setores (Menna, 2024; Hotz, 2024).

3 METODOLOGIA

Nesta seção, é apresentada a abordagem metodológica adotada para a realização deste estudo sobre previsão de carga elétrica utilizando técnicas de aprendizado de máquina. O processo metodológico é baseado na metodologia CRISP-DM (*Cross Industry Standard Process for Data Mining*), um modelo amplamente aceito e eficaz para a mineração de dados e análise preditiva (Schröer, et al, 2021). A escolha dessa metodologia visa garantir uma abordagem sistemática e robusta para a análise e modelagem dos dados de demanda elétrica, desde a compreensão do problema até a implementação e monitoramento do modelo preditivo.

3.1 Compreensão do Problema

Como dito anteriormente, a região Nordeste, ao longo dos anos, tornou-se um importante polo de geração de energias renováveis, com uma participação crescente no SIN. No entanto, a natureza intermitente dessas fontes de energia somado a crescente expansão da MMDG faz com que a demanda por uma previsão precisa da carga elétrica seja ainda mais crítica. Quando as condições climáticas mudam repentinamente, a geração de energia pode cair ou aumentar de forma imprevisível, exigindo uma resposta rápida e precisa da rede elétrica para evitar sobrecargas ou déficits de energia.

3.2 Compreensão dos dados

O ONS disponibiliza de forma aberta diversas informações pertinentes à gestão e operação do SIN. Dentre essas informações, destaca-se a carga elétrica por macrorregião, que é apresentada em *Watts médios* (W_{med}). Esses dados são essenciais para a análise do comportamento da demanda nas regiões, permitindo uma melhor compreensão das necessidades energéticas locais. A disponibilização dessas informações contribui para a tomada de decisões mais precisa sobre a operação e planejamento do sistema elétrico, além de auxiliar no desenvolvimento de estratégias de otimização e eficiência no fornecimento de energia para a população.

Na Tabela 3.1, é apresentada a estrutura de dados fornecida pelo ONS para a curva de carga do SIN, que ilustra o perfil de consumo de energia elétrica com discretização horária.

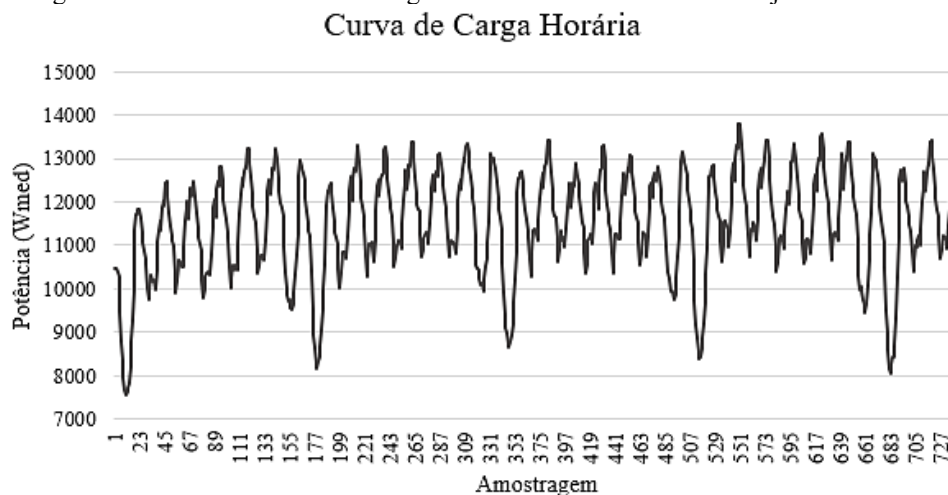
Tabela 3.1 – Demanda elétrica por macrorregião com amostragem horária.

id_subsistema	nom_subsistema	din_instante	val_cargaenergiamwmed
N	NORTE	01/01/2023 00:00:00	5.990,55300000
NE	NORDESTE	01/01/2023 00:00:00	10.455,81799999
S	SUL	01/01/2023 00:00:00	10.273,82699999
SE	SUDESTE	01/01/2023 00:00:00	32.183,88499999
...
N	NORTE	31/12/2023 23:00:00	6.641,41000000
NE	NORDESTE	31/12/2023 23:00:00	12.144,91700000
S	SUL	31/12/2023 23:00:00	10.365,64700000
SE	SUDESTE	31/12/2023 23:00:00	36.060,07599999

Fonte: ONS, 2023.

Os dados são, então, organizados por subsistema (nom_subsistema), data e hora da medição (din_instante), e valor de carga elétrica em megawatts médios (val_cargaenergiamwmed). Vale ressaltar que, conforme informado pelo ONS, os dados disponibilizados passam por um processo contínuo de verificação e podem ser atualizados após sua publicação. Embora os valores brutos permaneçam inalterados, a forma de apresentação dos dados pode sofrer modificações para refletir a precisão e a consistência das informações.

Figura 3.1 – Gráfico referente à carga horária do Nordeste no mês de janeiro de 2023.



Fonte: Autoria própria com base nos dados de ONS, 2023.

É apresentado na Figura 3.1 um gráfico que ilustra o comportamento da carga, com os dados coletados em intervalos de uma hora. Essa discretização horária permite uma análise detalhada das variações ao longo do tempo.

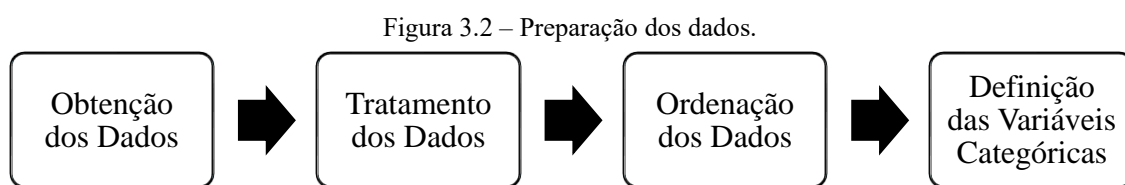
A partir do gráfico na Figura 3.1, é possível identificar visualmente uma forte tendência constante, acompanhada por variações sazonais que se repetem ao longo dos dias do mês. Essa tendência constante é observada pela estabilidade na linha de base do

gráfico, onde os valores de carga elétrica mantêm um comportamento relativamente uniforme ao longo do tempo, sem grandes desvios abruptos ou declínios acentuados.

As variações sazonais, por outro lado, são evidenciadas pelos picos e vales que ocorrem em intervalos regulares, correspondendo a determinados períodos do dia. Por exemplo, os picos de carga são frequentemente registrados nas horas de maior atividade humana, como durante o início da manhã e no final da tarde, quando o consumo de energia tende a aumentar significativamente. Esses padrões se repetem de forma consistente, indicando um ciclo sazonal diário bem definido, onde as flutuações de carga se mantêm em um ritmo previsível ao longo do mês.

3.3 Preparação dos dados

A Figura 3.1 apresenta um gráfico que ilustra o comportamento da carga elétrica com discretização horária. A partir desse gráfico, é possível observar visualmente que os dados possuem uma forte tendência constante, acompanhada por variações sazonais que se repetem ao longo dos dias do mês. Essas variações refletem padrões diários e sazonais que são característicos do consumo de energia na região, evidenciando a importância de modelar essas flutuações para previsões precisas.



Fonte: Autoria própria.

A partir do Figura 3.2, o processo é iniciado pela obtenção dos dados, em que os dados brutos são coletados a partir de diversas fontes relevantes. Para este trabalho, a principal fonte é a base de dados pública disponibilizada pelo ONS, garantindo a qualidade e a integridade das informações obtidas nessa etapa.

Após a obtenção, os dados passam por uma fase de tratamento essencial. Nessa etapa, são removidos dados irrelevantes, duplicados ou errôneos, enquanto valores ausentes podem ser tratados ou imputados. O principal objetivo deste processo é limpar e refinar o conjunto de dados, tornando-o mais representativo e livre de ruídos que possam distorcer os resultados da análise.

Na ordenação dos dados, estes são organizados de maneira lógica e estruturada, garantindo uma disposição que facilita o acesso e a manipulação nas etapas subsequentes

de modelagem e análise. Uma ordenação correta dos dados é essencial para assegurar que as informações estejam alinhadas e prontas para serem processadas de forma eficiente.

Por fim, as variáveis do conjunto de dados são identificadas e classificadas. Nesta etapa, também se pode criar variáveis que têm o potencial de enriquecer o modelo, oferecendo uma compreensão mais detalhada e precisa dos padrões presentes nos dados.

A relevância das novas variáveis em um contexto de demandas elétricas, ao construir um banco de dados para alimentar um modelo MLP deve ser avaliada cuidadosamente. Se forem irrelevantes ou redundantes, essas variáveis podem introduzir ruído e aumentar a complexidade desnecessariamente, comprometendo o desempenho e a capacidade de generalização do modelo.

Para definir essa relevância, várias abordagens podem ser utilizadas. A análise de correlação, por exemplo, avalia a relação entre as novas variáveis e a demanda elétrica, permitindo identificar aquelas com alta correlação. Outra técnica é a importância das características, que utiliza métodos de aprendizado de máquina, como árvores de decisão, para identificar quais variáveis têm maior impacto nas previsões. Testes estatísticos, como ANOVA ou teste t, podem verificar se as novas variáveis apresentam efeito significativo sobre a demanda elétrica.

É apresentado na Tabela 3.2 a estrutura final dos dados utilizados para a modelagem, que consiste em um arquivo Excel com 8.761 linhas (incluindo o cabeçalho) e 35.040 entradas.

Tabela 3.2 – Base de dados utilizado para a RNA.

din_instante	val_cargaenergiahomwmed	hora_dia	dia_semana	hora_pico
01/01/2023	10455818000	0	0	0
01/01/2023	10481710000	1	0	0
01/01/2023	10475740000	2	0	0
...
16/08/2023	12138344000	23	3	0
17/08/2023	11554734000	0	4	0
17/08/2023	11229219000	1	4	0
...
31/12/2023	13150272000	21	0	1
31/12/2023	12641152000	22	0	0
31/12/2023	12144917000	23	0	0

Fonte: Autoria própria.

Como o foco do estudo é a carga elétrica na região Nordeste, os dados das demais macrorregiões foram excluídos, mantendo-se apenas o registro temporal das amostragens

e os valores correspondentes em *megawatts* foram convertidos e arredondados para *watts*. Além disso, três novas colunas foram criadas: a coluna "hora_dia", que contém valores de 0 a 23, indicando o horário de registro da carga; a coluna "dia_semana", com valores de 0 a 6, representando os dias da semana de domingo a sábado; e a coluna "hora_pico", que sinaliza os horários de pico na região Nordeste, definidos como o período entre 17h e 21h. As informações adicionais foram criadas com o intuito do algoritmo identificar possíveis sazonalidades da carga elétrica de forma mais fácil.

3.4 Modelagem

O processo de modelagem utiliza a rede MLP para realizar a previsão de curto prazo da carga. Vale ressaltar que o MLP é uma RNA *feedforward* composta por uma ou mais camadas ocultas entre a camada de entrada e a camada de saída. Essa estrutura permite ao modelo capturar relações não lineares complexas entre as variáveis de entrada e a variável alvo, a carga elétrica.

Inicialmente, o modelo é ajustado para a previsão de um horizonte de vinte e quatro horas à frente, ou seja, um horizonte de curto prazo. Para esta análise, foram realizados experimentos com seis configurações distintas entre otimizadores e funções de ativação.

O modelo MLP desenvolvido neste trabalho consiste em diversas camadas: uma camada de entrada, camadas ocultas e uma camada de saída. As camadas ocultas empregam funções de ativação lineares e não lineares, como a ReLU (*Rectified Linear Unit*), para permitir que o modelo aprenda as complexidades dos dados.

Na literatura atual, não existe um método universalmente aceito para definir de maneira precisa a quantidade ideal de camadas e neurônios em uma RNA. Essa ausência de um método definitivo deve-se à natureza altamente dependente do problema das arquiteturas de redes neurais. Cada problema apresenta características únicas, como a complexidade dos padrões nos dados e o volume de informações a serem processadas, que tornam inviável a aplicação de uma única fórmula ou regra para todos os casos (Brownlee, 2019).

Diante dessa ausência, a construção da rede neural para este trabalho será realizada de forma empírica. Isso significa que diferentes configurações de camadas e neurônios serão testadas sistematicamente, avaliando-se o desempenho de cada configuração por meio de métricas, como o erro percentual absoluto (MAPE); o indicador R^2 ; e o tempo

de processamento. A escolha final da arquitetura será aquela que apresentar o melhor equilíbrio entre capacidade de aprendizado, generalização e tempo de processamento.

Essa abordagem empírica é justificada pela necessidade de adaptar a rede às especificidades do problema de previsão de carga elétrica, onde os padrões nos dados podem variar significativamente e requerer um ajuste fino da complexidade da rede. A experimentação permite explorar diferentes combinações e identificar a configuração que melhor se adapta aos dados e ao objetivo do modelo, assegurando um desempenho robusto e eficaz.

3.5 Avaliação

Para a avaliação do modelo de previsão de carga elétrica, utiliza-se três indicadores: MAPE, R^2 e tempo de processamento.

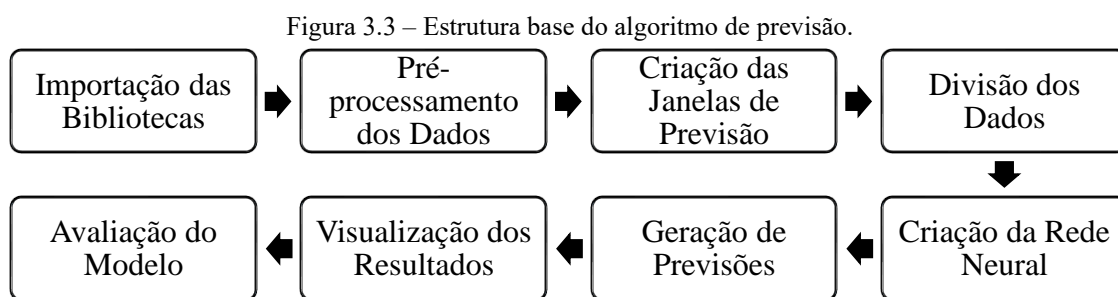
- **MAPE (Erro Percentual Absoluto):** O MAPE mede a precisão do modelo, calculando a média dos erros percentuais absolutos entre os valores previstos e os valores reais. Ele é expresso como uma porcentagem, o que facilita a interpretação dos erros em relação ao valor real. Um MAPE menor indica maior precisão (Ren e Glasure, 2009). Um valor menor que 10% é, em geral, considerado um valor expressivamente bom para a grande maioria das aplicações.
- **R^2 (Coeficiente de Determinação):** O R^2 avalia o quão bem o modelo explica a variabilidade dos dados. Ele varia de 0 a 1, onde 1 indica que o modelo explica toda a variabilidade dos dados e 0 indica que o modelo não explica nada. Um valor de R^2 mais próximo de 1 sugere que o modelo tem um bom ajuste aos dados. Em grande parte dos casos, um valor de R^2 acima de 0,7 é aceitável, pois abaixo disso, o modelo pode refletir a não adequabilidade do modelo à aplicação em estudo (Montgomery, 1992).
- **Tempo de Processamento:** Esse indicador mede o tempo necessário para treinar o modelo e gerar previsões. Em aplicações práticas, o tempo de processamento é crucial, especialmente quando há a necessidade de previsões em tempo real ou para grandes volumes de dados. Um modelo eficiente deve ter um bom equilíbrio entre precisão (medida por MAPE e R^2) e tempo de processamento (Bishop, 2006).

Esses indicadores juntos permitem uma avaliação abrangente do desempenho do modelo, considerando tanto a precisão das previsões quanto a eficiência do processamento.

3.6 Implementação

A implementação do modelo de previsão de curto prazo da demanda elétrica foi realizada utilizando a interface Jupyter associada a linguagem de programação Python, assim, obtendo um ambiente interativo e flexível para desenvolvimento e experimentação. As bibliotecas selecionadas oferecem um conjunto poderoso de ferramentas para manipulação de dados, pré-processamento, construção e treinamento do modelo de rede neural MLP.

É descrito na Figura 3.3 o passo a passo para a construção do algoritmo com horizonte de previsão de vinte e quatro horas.



Fonte: Autoria própria.

Para a etapa de importação das bibliotecas, é necessário preparar o ambiente com as ferramentas para cada etapa subsequente, assim, importando as bibliotecas necessárias para a execução do modelo: (i) *time*: serve para medir o tempo de execução do código; (ii) *numpy* e *pandas*: utilizados para manipulação e análise dos dados; (iii) *matplotlib*: para a criação de gráficos; (iv) *sklearn*: necessário para o pré-processamento dos dados e para dividir os dados entre treino e teste; e (v) *tensorflow*: para construir e treinar a rede neural.

Na fase de pré-processamento dos dados, os dados brutos são normalizados para assegurar que todas as variáveis estejam na mesma escala, o que melhora o desempenho do modelo. Na normalização os dados são escalados entre 0 e 1 utilizando *MinMaxScaler* para evitar que variáveis com maior amplitude dominem o aprendizado da rede neural.

A etapa de criação das janelas de previsão serve para transformar a série temporal em um conjunto de previsores e alvos, ou seja, cria-se blocos de observações anteriores que servem como entrada para prever os valores futuros.

Na etapa de divisão dos dados, os dados são divididos em conjunto de treinamento e teste: (i) treinamento: o modelo aprende a partir desses dados; e (ii) teste: avaliação do desempenho do modelo em dados não vistos. Os dados são divididos de forma a garantir que o modelo possa ser avaliado com dados o qual ele não foi exposto durante o treinamento (Bishop, 2006; Arlot, 2010). O trecho do código que descreve essa etapa é descrito no Quadro 3.1.

Quadro 3.1 – Código referente à implementação das variáveis de teste e treinamento.

```
n = 24

previsores = []
target_previsao = []

for i in range(n, len(saida_normalizada)-24, 1):
    previsores.append(entradas_normalizadas[i-n:i,:])
    target_previsao.append(saida_normalizada[i:i+24])

previsores, target_previsao = np.array(previsores),
np.array(target_previsao)

previsores = previsores.reshape(-1,24*4)
target_previsao = target_previsao.reshape(-1,24)

I_treinamento, I_teste, t_treinamento, t_teste =
train_test_split(previsores, target_previsao, test_size=0.25,
shuffle=False)
```

Fonte: Autoria Própria

Finalmente, a rede neural é criada utilizando a arquitetura MLP. Nessa etapa é possível definir as camadas, e funções de ativação, além da aplicação de técnicas como o *dropout* e *early stopping* para lidar com o problema de *overfitting*. O *dropout* ajuda a promover a generalização ao prevenir a dependência excessiva de neurônios específicos, enquanto o *early stopping* interrompe o treinamento quando o modelo começa a sobre ajustar os dados de treinamento, garantindo que o modelo seja eficaz e robusto ao lidar com novos dados.

O modelo treinado é utilizado para prever os valores no conjunto de teste, prevendo, assim, as próximas 24 horas. Por fim, é possível analisar visualmente a precisão das previsões do modelo e quantificar sua eficiência.

No Quadro 3.2, apresentamos o código referente à implementação da rede neural em três configurações: (1) uma camada oculta, (2) duas camadas ocultas e (3) três camadas ocultas.

A estrutura dessa rede neural foi projetada de maneira a otimizar o processamento de informações, começando com uma camada de entrada composta por 24 neurônios. Esta primeira camada é responsável por capturar as características iniciais dos dados de entrada.

Em seguida, a rede possui uma segunda camada que contém o dobro de neurônios em relação à primeira, totalizando 48 neurônios. Essa duplicação na quantidade de neurônios permite à rede ter uma capacidade maior de aprendizado e representação, facilitando a extração de padrões mais complexos nos dados.

Por último, a rede apresenta uma terceira camada, que também segue a lógica de duplicação, contendo 96 neurônios. Essa terceira camada amplia ainda mais a capacidade da rede de processar e refinar as informações, possibilitando um aprendizado mais profundo e uma maior precisão nas previsões.

Quadro 3.2 – Código referente à implementação da rede neural MLP com uma camada oculta.

```
Previsor = Sequential()
Previsor.add(Dense(units=48 , activation='linear', input_dim =
I_treinamento.shape[1]))

Previsor.compile(optimizer=Nadam(learning_rate=1e-3),
loss='mean_squared_error', metrics=['mean_absolute_error'])

early_stop = EarlyStopping(monitor='val_loss', patience=3)
historico = Previsor.fit(I_treinamento, t_treinamento, epochs=50,
callbacks=[early_stop])
```

Fonte: Autoria Própria.

Após a descrição detalhada da metodologia empregada neste estudo, que abrangeu desde a seleção dos dados até a implementação das técnicas de regularização, como o descarte (*dropout*), é essencial agora analisar os resultados obtidos. Nesta próxima seção, apresentaremos os desempenhos do modelo nas diferentes etapas do treinamento e validação, destacando como as escolhas metodológicas impactaram a eficácia do modelo em prever os resultados. A análise dos dados permitirá uma compreensão mais profunda do desempenho da rede neural e suas implicações práticas.

4 RESULTADOS E DISCUSSÕES

A partir de combinações entre otimizadores e funções de ativação, é possível examinar como cada configuração impacta o desempenho do modelo em termos de precisão, velocidade de convergência e comportamento geral durante o treinamento. A estrutura do modelo foi cuidadosamente ajustada para otimizar o desempenho. Especificamente, foi aplicado um descarte de 10% para ajudar a prevenir o sobre ajuste e melhorar a generalização do modelo. As análises a seguir detalham os resultados obtidos e discutem as implicações de cada escolha de configuração.

4.1 Caso 1 – Otimizador: Adam; Função de Ativação: Linear

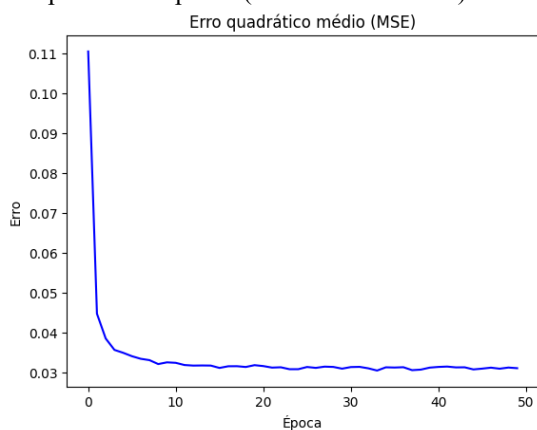
É apresentado na tabela 4.1 as simulações do Caso 1, que utiliza o otimizador Adam e a função de ativação linear em três situações: (1) uma camada oculta; (2) duas camadas ocultas; e (3) três camadas ocultas.

Tabela 4.1 – Caso 1: MAPE, R² e tempo de processamento.

Número de Camadas	MAPE (%)	R ²	Tempo de Processamento (s)
1	4,95	0,57	15,74
2	1,59	0,93	14,62
3	2,17	0,89	16,55

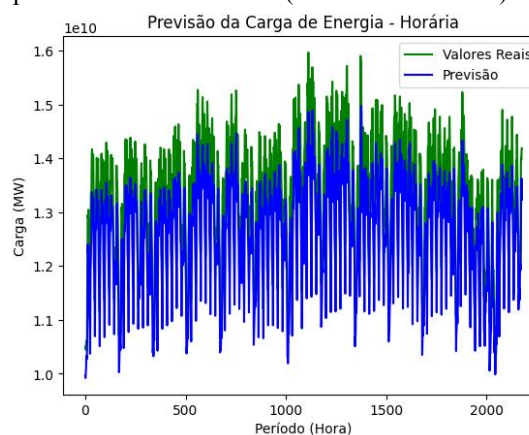
Fonte: Autoria própria.

Figura 4.1 – Caso 1: Evolução do erro com o passar das épocas (uma camada oculta).



Fonte: Autoria própria.

Figura 4.2 – Caso 1: Comparação entre a previsão e os dados reais (uma camada oculta).

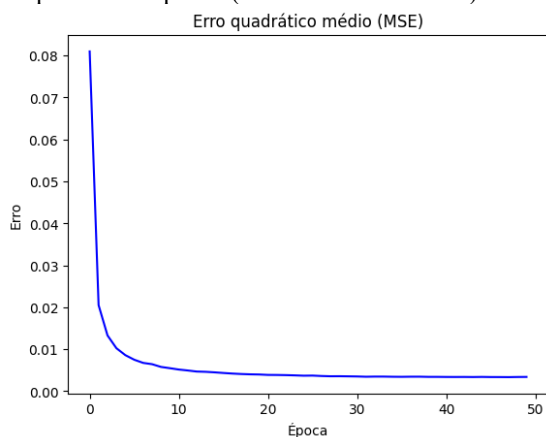


Fonte: Autoria própria.

Para o modelo com uma camada, o MAPE foi de 4,95%, o que indica um erro percentual relativamente alto em relação aos demais resultados. Enquanto isso, o R² de 0,57 demonstra que o modelo explica apenas 57% da variabilidade dos dados, refletindo

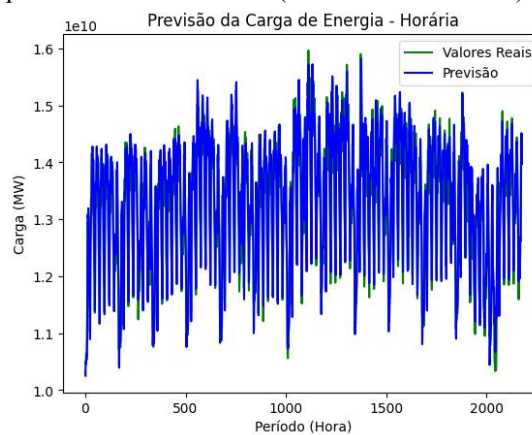
um ajuste moderado. Já o tempo de processamento foi de 15,74 segundos, o que é aceitável para um modelo relativamente simples. É ilustrado na figura 4.1 o decaimento do erro ao longo de cada iteração. O formato da curva indica que o modelo consegue aprender a partir dos dados e com isso melhorar a previsão. É possível comparar na figura 4.2 as previsões do modelo com os dados reais. A partir dela, é possível perceber que a curva de previsão possui o comportamento dos dados reais, mas a amplitude real ainda não é alcançada pelos dados treinados.

Figura 4.3 – Caso 1: Evolução do erro com o passar das épocas (duas camadas ocultas).



Fonte: Autoria própria.

Figura 4.4 – Caso 1: Comparação entre a previsão e os dados reais (duas camadas ocultas).



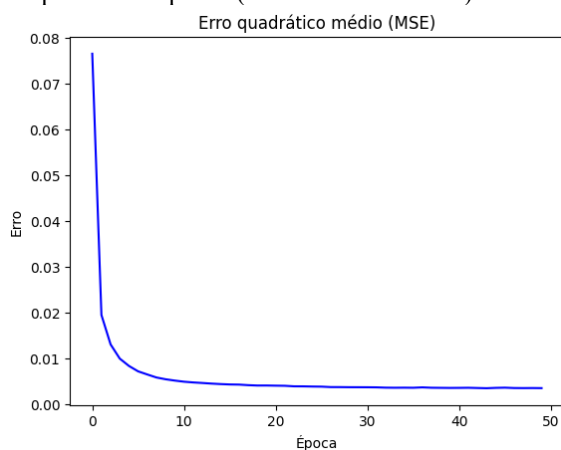
Fonte: Autoria própria.

Ao adicionar uma segunda camada, observa-se uma melhoria significativa. O MAPE reduziu para 1,59%, mostrando uma precisão consideravelmente melhorada. O R^2 aumentou para 0,93, o que significa que o modelo com duas camadas consegue explicar 93% da variabilidade dos dados, oferecendo um ajuste mais robusto. O tempo de processamento, por outro lado, diminuiu ligeiramente para 14,62 segundos, o que pode indicar que o aumento da complexidade não afetou negativamente a eficiência computacional. É ilustrado na figura 4.3 o decaimento do erro ao longo de cada iteração, enquanto a figura 4.4 compara as previsões do modelo com os dados reais. É possível perceber que a curva de previsão, nesse caso, segue próximo aos valores reais de forma mais aprimorada em relação ao primeiro caso.

Com a adição de uma terceira camada, o MAPE aumentou para 2,17%, indicando um decaimento na precisão em comparação com o modelo de duas camadas. O R^2 diminuiu um pouco para 0,89, ainda indicando um bom ajuste, mas não tão eficaz quanto o modelo com duas camadas. O tempo de processamento aumentou para 16,55 segundos, possivelmente devido ao maior número de cálculos necessários para treinar um modelo

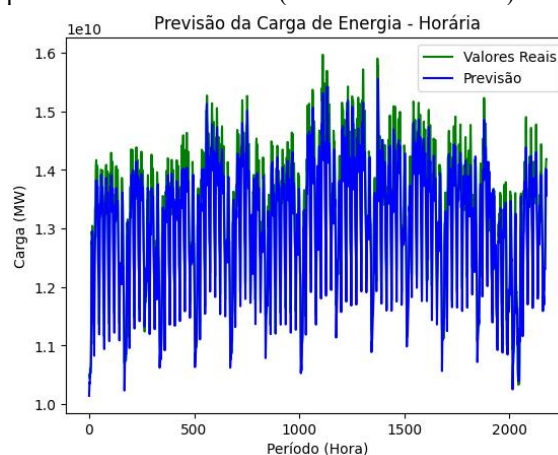
mais profundo. É ilustrado na figura 4.5 o decaimento do erro ao longo de cada iteração. Com relação aos demais casos, é possível perceber que o decaimento do erro ocorre de forma mais suave ao longo de cada época, de forma semelhante ao que foi observado no segundo caso. É possível comparar na figura 4.6 as previsões do modelo com os dados reais. A curva de previsão se ajusta aos valores reais de forma satisfatória, porém, com menor precisão do que o modelo com apenas duas camadas.

Figura 4.5 – Caso 1: Evolução do erro com o passar das épocas (três camadas ocultas).



Fonte: Autoria própria.

Figura 4.6 – Caso 1: Comparação entre a previsão e os dados reais (três camadas ocultas).



Fonte: Autoria própria.

Com a adição de uma terceira camada, o MAPE aumenta para 2,17%, indicando um decaimento na precisão em comparação com o modelo de duas camadas. O R^2 diminuiu um pouco para 0,89, ainda indicando um bom ajuste, mas não tão eficaz quanto o modelo com duas camadas. O tempo de processamento aumenta para 16,55 segundos, possivelmente devido ao maior número de cálculos necessários para treinar um modelo mais profundo. É ilustrado na figura 4.5 o decaimento do erro ao longo de cada iteração. Com relação aos demais casos, é possível perceber que o decaimento do erro ocorre de forma mais suave ao longo de cada época, de forma semelhante ao que foi observado no segundo caso. É possível comparar na figura 4.6 as previsões do modelo com os dados reais. A curva de previsão se ajusta aos valores reais de forma satisfatória, porém, com menor precisão do que o modelo com apenas duas camadas.

As simulações para o caso 1 mostram que adicionar uma segunda camada ao modelo resulta em uma melhoria significativa na precisão (redução do MAPE) e no ajuste (aumento do R^2). Entretanto, a adição de uma terceira camada não contribuiu para uma melhoria adicional no desempenho. Esses resultados indicam que, para o modelo específico com otimizador Adam e função de ativação Linear, o uso de duas camadas é

mais eficiente, oferecendo um bom equilíbrio entre precisão, ajuste e eficiência computacional. Adicionar mais camadas não justifica o aumento no tempo de processamento e não proporciona benefícios significativos em termos de precisão ou ajuste.

4.2 Caso 2 – Otimizador: Adam; Função de Ativação: ReLU

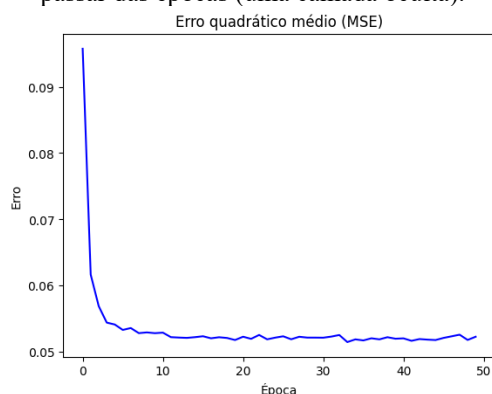
Os resultados relativos ao segundo caso são apresentados na tabela 4.2, a qual expõe os valores referentes às métricas utilizando o otimizador Adam e a função de ativação ReLU em todas as camadas.

Tabela 4.2 – Caso 2: MAPE, R^2 e tempo de processamento.

Número de Camadas	MAPE (%)	R^2	Tempo de Processamento (s)
1	5,47	0,49	14,39
2	2,11	0,89	21,88
3	2,45	0,85	17,24

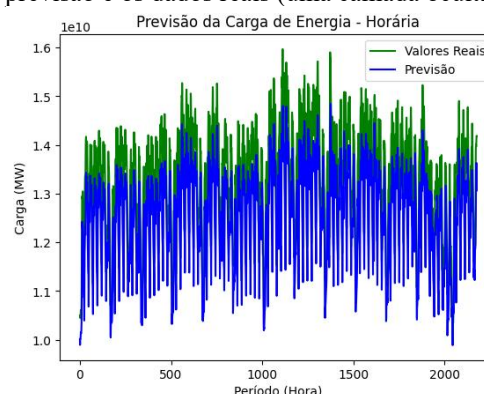
Fonte: Autoria própria.

Figura 4.7 – Caso 2: Evolução do erro com o passar das épocas (uma camada oculta).



Fonte: Autoria própria.

Figura 4.8 – Caso 2: Comparação entre a previsão e os dados reais (uma camada oculta).

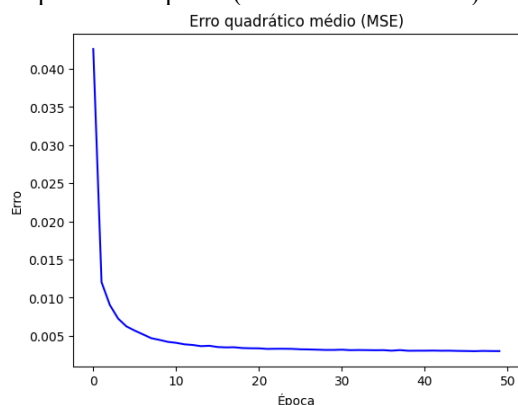


Fonte: Autoria própria.

Inicialmente, com uma camada, o modelo apresenta um MAPE de 5,47%, indicando uma precisão relativamente baixa nas previsões, e um R^2 de 0,49, o que sugere que o modelo explica apenas 49% da variabilidade dos dados. O tempo de processamento é de 14,39 segundos, que é relativamente rápido e reflete a simplicidade do modelo. É ilustrado na Figura 4.7 o decaimento do erro ao longo das iterações de treinamento, enquanto é possível comparar na Figura 4.8 as previsões do modelo com os dados reais. Apesar de a curva de previsão seguir o comportamento geral dos dados reais, ela não consegue capturar totalmente a amplitude desses dados.

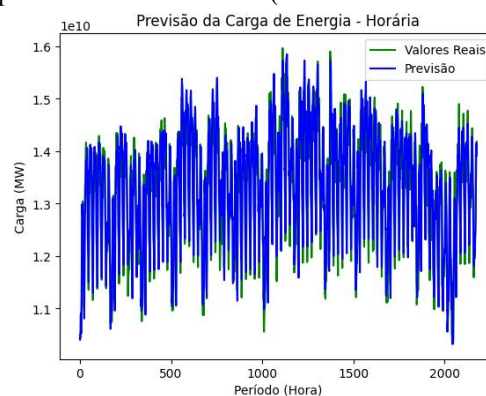
Ao adicionar uma segunda camada, o MAPE melhora significativamente para 2,11%, evidenciando uma precisão muito melhor. O R^2 também aumenta para 0,89, indicando que o modelo com duas camadas é capaz de explicar 89% da variabilidade dos dados, o que representa uma melhoria substancial no ajuste. No entanto, o tempo de processamento aumenta consideravelmente para 21,88 segundos, possivelmente devido à maior complexidade do modelo com duas camadas, que requer mais cálculos durante o treinamento. É apresentado na figura 4.9 o decaimento do erro ao longo de cada iteração. É possível perceber que o decaimento possui menos picos ao longo de cada época, ou seja, a curva ao longo das iterações é mais suave. A figura 4.10 compara as previsões do modelo com os dados reais.

Figura 4.9 – Caso 2: Evolução do erro com o passar das épocas (duas camadas ocultas).



Fonte: Autoria própria.

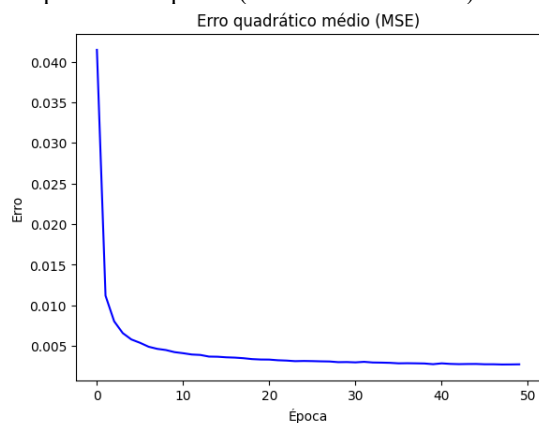
Figura 4.10 – Caso 2: Comparação entre a previsão e os dados reais (duas camadas ocultas).



Fonte: Autoria própria.

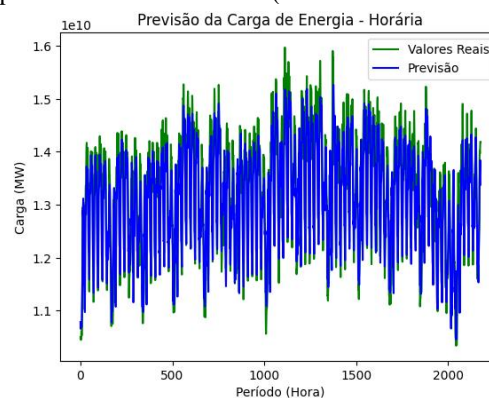
Com três camadas, o MAPE aumenta ligeiramente para 2,45%, sugerindo uma leve degradação na precisão em comparação com o modelo de duas camadas. O R^2 diminui um pouco para 0,85, ainda indicando um bom ajuste, mas não tão eficaz quanto o modelo com duas camadas. O tempo de processamento é reduzido para 17,24 segundos, o que é mais baixo do que o tempo de processamento para o modelo com duas camadas, mas ainda maior do que o tempo registrado para o modelo com uma camada. É apresentado na figura 4.11 o decaimento do erro ao longo de cada iteração. É possível comparar na figura 4.12 compara as previsões do modelo com os dados reais. É possível perceber que a curva de previsão se ajusta aos valores reais de forma satisfatória, porém, com menor precisão do que o modelo com apenas duas camadas.

Figura 4.11 – Caso 2: Evolução do erro com o passar das épocas (três camadas ocultas).



Fonte: Autoria própria.

Figura 4.12 – Caso 2: Comparação entre a previsão e os dados reais (três camadas ocultas).



Fonte: Autoria própria.

Portanto, o modelo com duas camadas oferece o melhor equilíbrio entre precisão, ajuste e eficiência computacional. A adição de uma terceira camada não proporciona uma melhoria significativa na precisão ou no ajuste, e o aumento no tempo de processamento sugere que a complexidade adicional não é justificável.

4.3 Caso 3 – Otimizador: Nadam; Função de Ativação: Linear

É apresentado na tabela 4.3 as simulações do Caso 3, que utiliza o otimizador Nadam e a função de ativação Linear em todas as camadas.

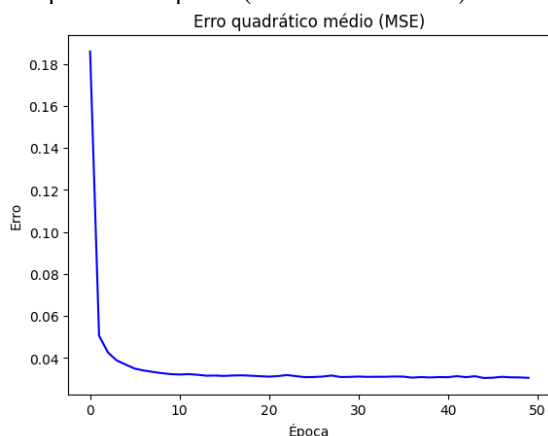
Tabela 4.3 – Caso 3: MAPE, R^2 e tempo de processamento.

Número de Camadas	MAPE (%)	R^2	Tempo de Processamento (s)
1	5,83	0,42	14,52
2	1,49	0,94	16,41
3	4,24	0,67	18,81

Fonte: Autoria própria.

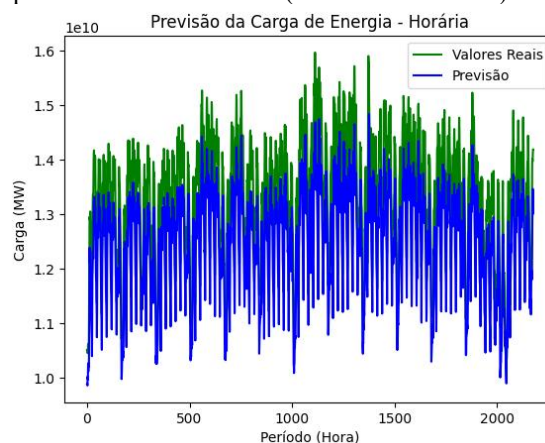
Para o modelo com uma camada, o MAPE é de 5,83%, refletindo um erro percentual relativamente alto nas previsões, e o R^2 é de 0,42, indicando que o modelo explica apenas 42% da variabilidade dos dados. Isso sugere um ajuste relativamente fraco. O tempo de processamento é de 14,52 segundos, o que é razoável para um modelo com uma camada. É apresentado na Figura 4.13 o decaimento do erro ao longo das iterações de treinamento, enquanto é possível comparar na Figura 4.14 as previsões do modelo com os dados reais. Apesar de a curva de previsão seguir o comportamento geral dos dados reais, ela não consegue acompanhar de forma adequada a amplitude desses dados.

Figura 4.13 – Caso 3: Evolução do erro com o passar das épocas (uma camada oculta).



Fonte: Autoria própria.

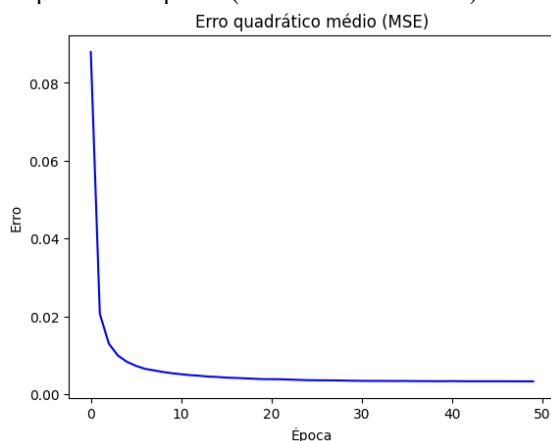
Figura 4.14 – Caso 3: Comparação entre a previsão e os dados reais (uma camada oculta).



Fonte: Autoria própria.

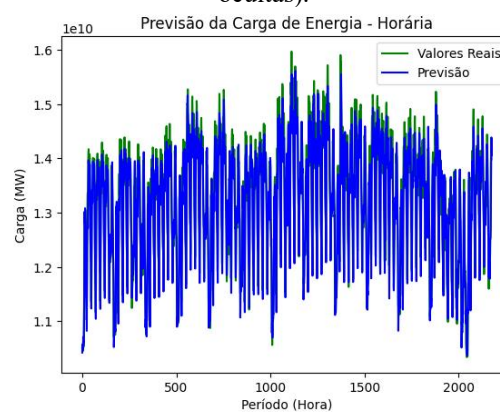
Ao adicionar uma segunda camada, o modelo apresenta uma melhora notável. O MAPE diminui para 1,49%, demonstrando uma precisão muito melhor nas previsões. O R^2 aumenta significativamente para 0,94, o que indica que o modelo com duas camadas é capaz de explicar 94% da variabilidade dos dados, proporcionando um ajuste muito mais robusto. O tempo de processamento aumenta para 16,41 segundos, refletindo a maior complexidade do modelo com duas camadas, mas ainda dentro de uma faixa aceitável. É apresentado na figura 4.15 o decaimento do erro ao longo de cada iteração. É possível comparar na figura 4.16 as previsões do modelo com os dados reais. É possível perceber que a curva de previsão segue muito próximo aos valores reais.

Figura 4.15 – Caso 3: Evolução do erro com o passar das épocas (duas camadas ocultas).



Fonte: Autoria própria.

Figura 4.16 – Caso 3: Comparação entre a previsão e os dados reais (duas camadas ocultas).

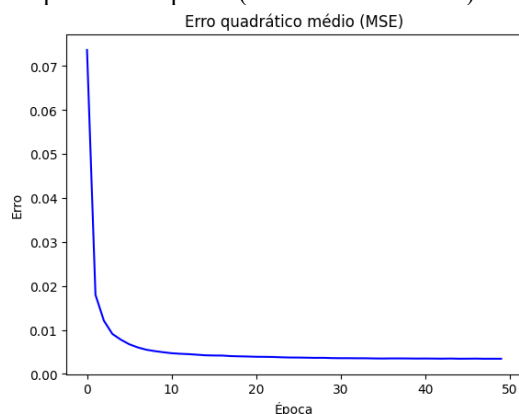


Fonte: Autoria própria.

Com a adição de uma terceira camada, o MAPE aumenta para 4,24%, indicando um decaimento na precisão das previsões em comparação com o modelo de duas camadas. O R^2 diminui para 0,67, ou seja, não foi tão eficaz quanto o modelo de duas

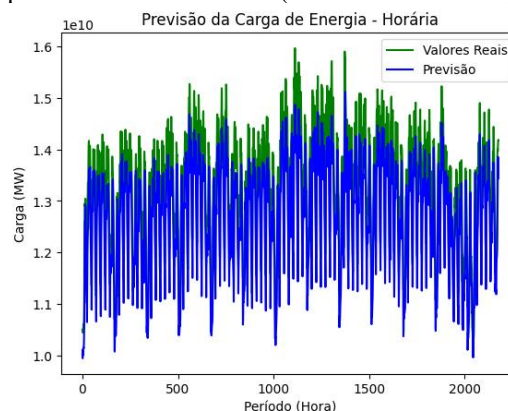
camadas. O tempo de processamento é o mais alto entre as configurações, atingindo 18,81 segundos, o que pode ser atribuído ao aumento da complexidade computacional com a adição da terceira camada. É apresentado na figura 4.17 o decaimento do erro ao longo de cada iteração, enquanto é possível comparar na figura 4.18 as previsões do modelo com os dados reais. É possível perceber que a curva de previsão se ajusta aos valores reais de forma satisfatória, porém, com menor precisão do que o modelo com apenas duas camadas.

Figura 4.17 – Caso 3: Evolução do erro com o passar das épocas (três camadas ocultas).



Fonte: Autoria própria.

Figura 4.18 – Caso 3: Comparação entre a previsão e os dados reais (três camadas ocultas).



Fonte: Autoria própria.

O modelo com duas camadas se destaca como a melhor configuração. Ele oferece a combinação mais favorável de precisão, ajuste e eficiência computacional. A adição de uma terceira camada não resulta em melhorias adicionais.

4.4 Caso 4 – Otimizador: Nadam; Função de Ativação: ReLU

A tabela 4.4 apresenta as simulações do Caso 3, que utiliza o otimizador Nadam e a função de ativação ReLU em todas as camadas. Nesse caso, para o modelo com uma camada, o MAPE é de 4,71%, indicando um erro percentual relativamente alto nas previsões em relação aos demais, e o R^2 é de 0,61, o que sugere um ajuste razoável. O tempo de processamento é de 14,83 segundos, um valor compatível com a simplicidade do modelo.

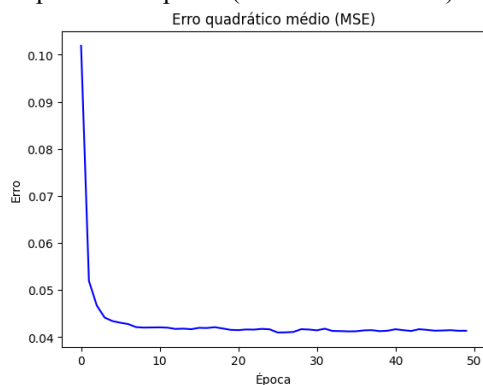
Tabela 4.4 – Caso 4: MAPE, R^2 e tempo de processamento.

Número de Camadas	MAPE (%)	R^2	Tempo de Processamento (s)
1	4,71	0,61	14,83
2	2,40	0,87	17,27
3	2,66	0,83	21,80

Fonte: Autoria própria.

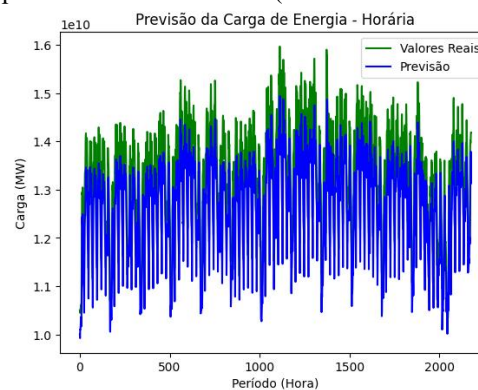
É apresentado na Figura 4.19 o decaimento do erro ao longo das iterações de treinamento, enquanto é possível comparar na Figura 4.20 as previsões do modelo com os dados reais.

Figura 4.19 – Caso 4: Evolução do erro com o passar das épocas (uma camada oculta).



Fonte: Autoria própria.

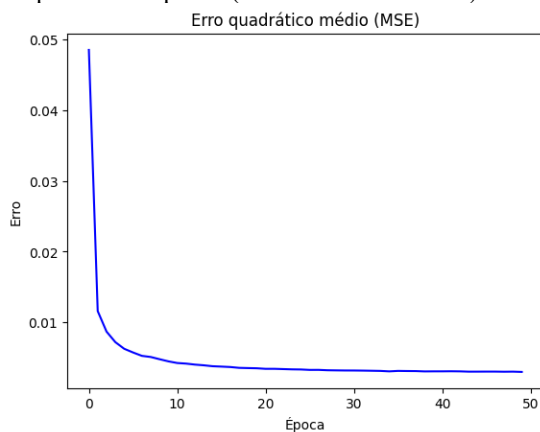
Figura 4.20 – Caso 4: Comparação entre a previsão e os dados reais (uma camada oculta).



Fonte: Autoria própria.

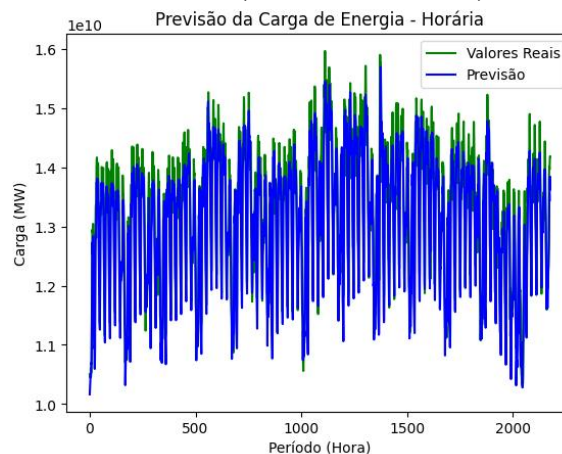
A adição de uma segunda camada resulta em melhorias significativas. O MAPE reduz para 2,40%, o que demonstra uma precisão aprimorada nas previsões. O R^2 sobe para 0,87, indicando que o modelo com duas camadas explica 87% da variabilidade dos dados, oferecendo um ajuste substancialmente melhor. No entanto, o tempo de processamento aumenta para 17,27 segundos, refletindo a maior complexidade do modelo com duas camadas. É apresentado na Figura 4.21 o decaimento do erro ao longo de cada iteração. Enquanto é possível comparar na figura 4.22 as previsões do modelo com os dados reais. É possível perceber que a curva de previsão segue muito próximo aos valores reais.

Figura 4.21 – Caso 4: Evolução do erro com o passar das épocas (duas camadas ocultas).



Fonte: Autoria própria.

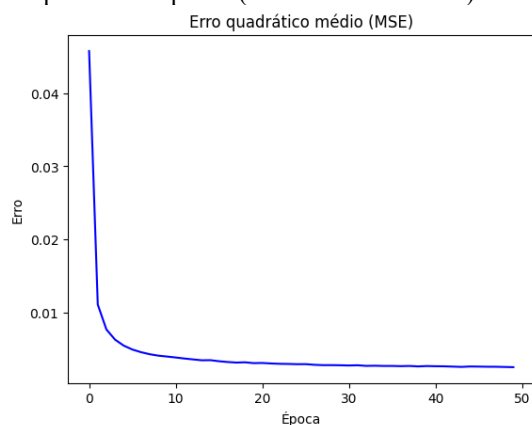
Figura 4.22 – Caso 4: Comparação entre a previsão e os dados reais (duas camadas ocultas).



Fonte: Autoria própria.

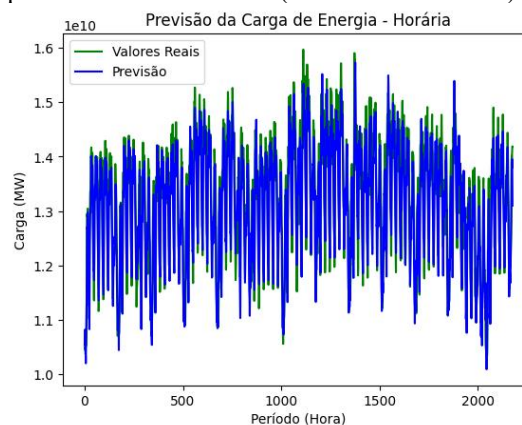
Com a adição de uma terceira camada, o MAPE aumenta ligeiramente para 2,66%, indicando uma leve redução na precisão das previsões em comparação com o modelo de duas camadas. O R^2 também diminuiu para 0,83, sugerindo que o modelo com três camadas explica 83% da variabilidade dos dados, mas não tão eficientemente quanto o modelo de duas camadas. O tempo de processamento subiu para 21,80 segundos. É apresentado na Figura 4.23 ilustra o decaimento do erro ao longo de cada iteração. A figura 4.24 compara as previsões do modelo com os dados reais. É possível perceber que a curva de previsão se ajusta aos valores reais de forma satisfatória, porém, com menor precisão do que o modelo com apenas duas camadas.

Figura 4.23 – Caso 4: Evolução do erro com o passar das épocas (três camadas ocultas).



Fonte: Autoria própria.

Figura 4.24 – Caso 4: Comparação entre a previsão e os dados reais (três camadas ocultas).



Fonte: Autoria própria.

Para o caso 4, o modelo com duas camadas apresenta o melhor equilíbrio entre precisão, ajuste e eficiência computacional. A adição de uma terceira camada não melhora significativamente a precisão ou o ajuste além de aumentar o tempo de processamento.

4.5 Caso 5 – Otimizador: RMSprop; Função de Ativação: Linear

É apresentado na tabela 4.5 as simulações do Caso 5, que utiliza o otimizador RMSprop e a função de ativação linear em todas as camadas.

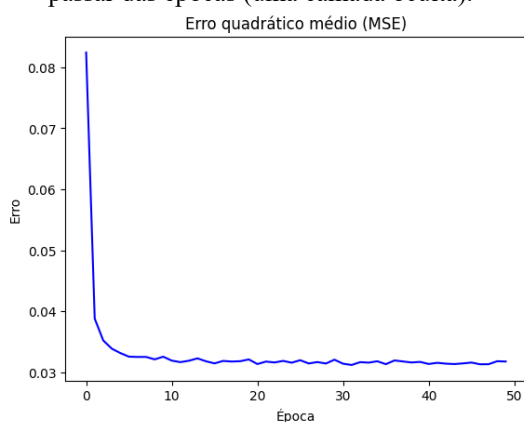
Tabela 4.5 – Caso 5: MAPE, R^2 e tempo de processamento.

Número de Camadas	MAPE (%)	R^2	Tempo de Processamento (s)
1	4,86	0,56	13,20
2	2,22	0,89	14,97
3	3,20	0,80	16,83

Fonte: Autoria própria.

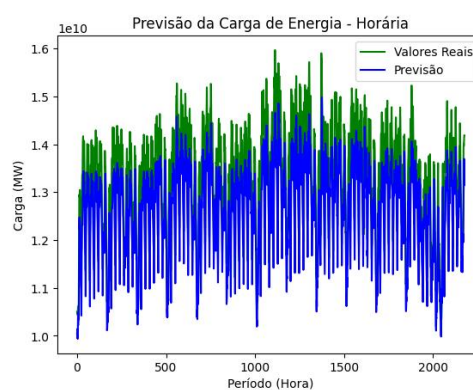
Para o modelo com uma camada, o MAPE é de 4,86%, indicando um erro percentual relativamente alto nas previsões. O R^2 é de 0,56, o que demonstra um ajuste moderado. O tempo de processamento é de 13,20 segundos, que é relativamente baixo e adequado para um modelo simples. É apresentado na Figura 4.25 o decaimento do erro ao longo das iterações de treinamento, enquanto na Figura 4.26 é possível comparar as previsões do modelo com os dados reais. Em relação aos demais métodos de otimização, é perceptível a piora na suavidade da curva da Figura 4.25, ou seja, no ajuste do erro ao decorrer do processamento.

Figura 4.25 – Caso 5: Evolução do erro com o passar das épocas (uma camada oculta).



Fonte: Autoria própria.

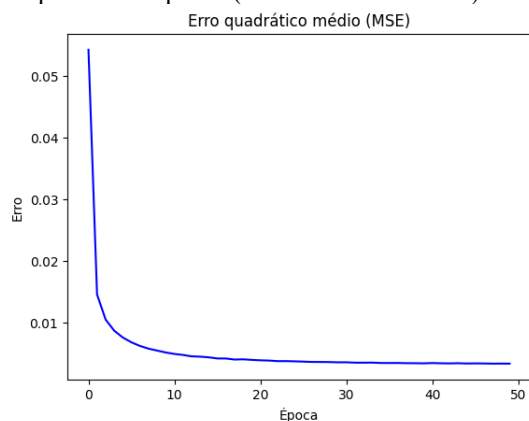
Figura 4.26 – Caso 5: Comparação entre a previsão e os dados reais (uma camada oculta).



Fonte: Autoria própria.

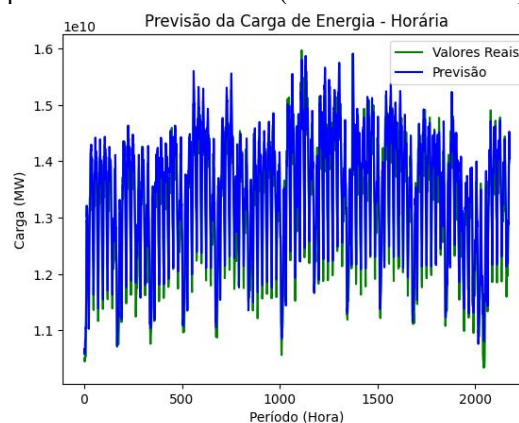
Com a adição de uma segunda camada, há uma melhoria significativa no desempenho. O MAPE reduz para 2,22%, indicando uma precisão aprimorada nas previsões. O R^2 aumenta para 0,89, indicando que o modelo com duas camadas explica 89% da variabilidade dos dados, proporcionando um ajuste significativamente melhor. O tempo de processamento aumenta para 14,97 segundos, o que reflete a maior complexidade do modelo. A figura 4.27 ilustra o decaimento do erro ao longo de cada iteração. Nesse caso, a referida curva volta a apresentar o comportamento parecido aos casos dos métodos de Adam e Nadam. A figura 4.28 compara as previsões do modelo com os dados reais.

Figura 4.27 – Caso 5: Evolução do erro com o passar das épocas (duas camadas ocultas).



Fonte: Autoria própria.

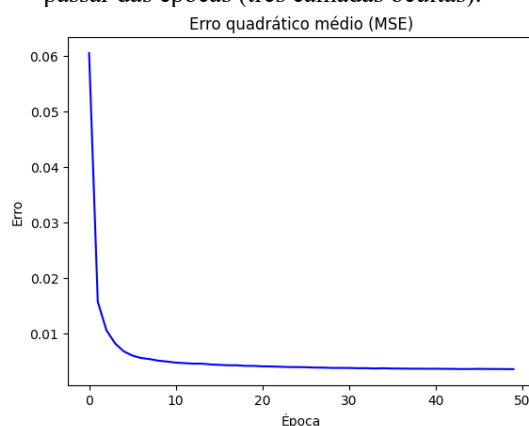
Figura 4.28 – Caso 5: Comparação entre a previsão e os dados reais (duas camadas ocultas).



Fonte: Autoria própria.

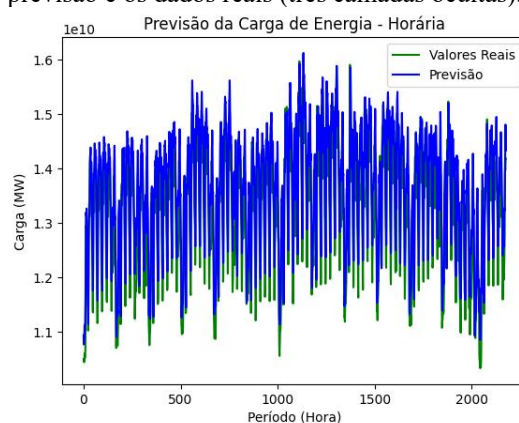
Ao adicionar uma terceira camada, o MAPE aumenta para 3,20%, sugerindo uma leve deterioração na precisão das previsões em comparação com o modelo de duas camadas. O R^2 diminui para 0,80, indicando que o modelo com três camadas explica 80% da variabilidade dos dados, mas não tão eficazmente quanto o modelo de duas camadas. O tempo de processamento sobe para 16,83 segundos, o mais alto entre as configurações, possivelmente devido à complexidade adicional do modelo. É apresentado na figura 4.29 o decaimento do erro ao longo de cada iteração. Enquanto é possível comparar na figura 4.30 as previsões do modelo com os dados reais. É possível perceber que a curva de previsão se ajusta aos valores reais de forma satisfatória, porém, com menor precisão do que o modelo com apenas duas camadas.

Figura 4.29 – Caso 5: Evolução do erro com o passar das épocas (três camadas ocultas).



Fonte: Autoria própria.

Figura 4.30 – Caso 5: Comparação entre a previsão e os dados reais (três camadas ocultas).



Fonte: Autoria própria.

Como visto nos casos anteriores, o modelo com duas camadas continua oferecendo o melhor equilíbrio entre precisão, ajuste e eficiência computacional. Embora

o modelo de três camadas ofereça um ajuste razoável, a adição de mais uma camada não resulta em uma melhoria significativa na precisão ou no ajuste.

4.6 Caso 6 – Otimizador: RMSprop; Função de Ativação: ReLU

A tabela 4.6 apresenta as simulações do Caso 6, que utiliza o otimizador RMSprop e a função de ativação ReLU em todas as camadas. Os dados possuem um comportamento distinto em comparação com os outros casos analisados, com um ajuste negativo sendo apresentado para o modelo com uma única camada.

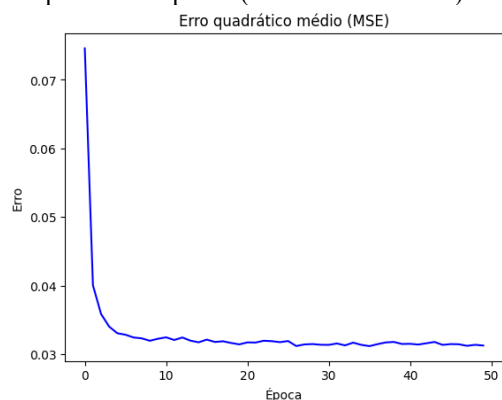
Tabela 4.6 – Caso 6: MAPE, R^2 e tempo de processamento.

Número de Camadas	MAPE (%)	R^2	Tempo de Processamento (s)
1	9,60	-0,45	18,43
2	2,04	0,90	14,96
3	2,37	0,87	17,02

Fonte: Autoria própria.

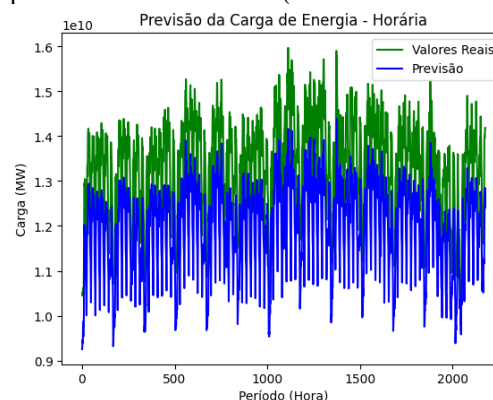
Para o modelo com uma camada, o MAPE é de 9,60%, o que indica uma precisão relativamente baixa nas previsões. Mais notável é o valor negativo do R^2 , de -0,45. Um R^2 negativo sugere que o modelo não está apenas mal ajustado, mas também é inferior a uma simples média dos valores reais, indicando que o modelo com uma camada não está capturando a variabilidade dos dados de forma eficiente e, portanto, apresenta um desempenho muito insatisfatório. É apresentado na figura 4.31 o decaimento do erro ao longo de cada iteração. Enquanto é possível comparar na figura 4.32 as previsões do modelo com os dados reais. É possível perceber que a curva de previsão possui o comportamento dos dados reais, mas suas amplitudes divergem de forma acentuada.

Figura 4.31 – Caso 6: Evolução do erro com o passar das épocas (uma camada oculta).



Fonte: Autoria própria.

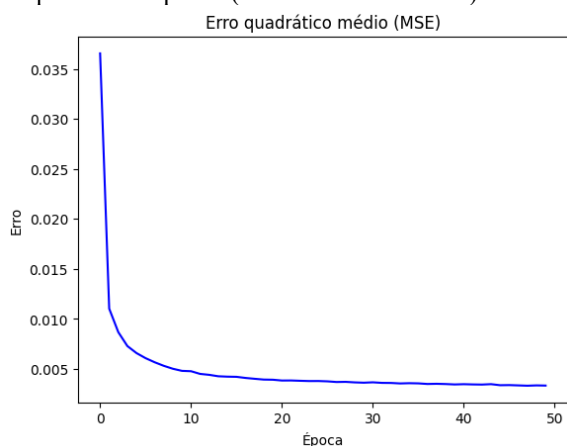
Figura 4.32 – Caso 6: Comparação entre a previsão e os dados reais (uma camada oculta).



Fonte: Autoria própria.

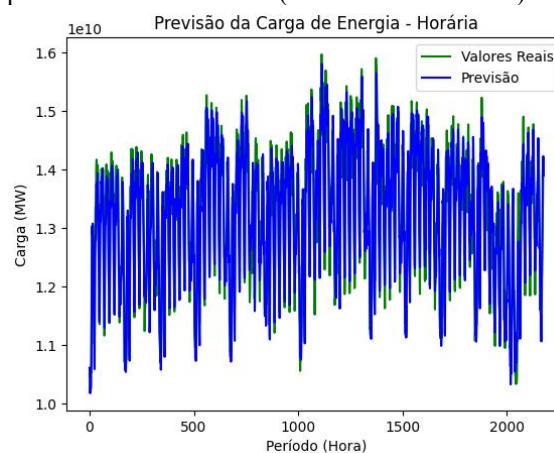
Ao adicionar uma segunda camada, observa-se uma melhoria considerável. O MAPE reduz para 2,04%, o que indica uma precisão significativamente melhor nas previsões. O R^2 aumenta para 0,90, o que indica que o modelo com duas camadas é capaz de explicar 90% da variabilidade dos dados, proporcionando um ajuste muito mais robusto. O tempo de processamento é de 14,96 segundos, que é aceitável e não tão elevado em comparação com o modelo de uma camada. É apresentado na figura 4.33 o decaimento do erro ao longo de cada iteração. Enquanto é possível comparar na figura 4.34 as previsões do modelo com os dados reais. É possível perceber que a curva de previsão segue muito próximo aos valores reais.

Figura 4.33 – Caso 6: Evolução do erro com o passar das épocas (duas camadas ocultas).



Fonte: Autoria própria.

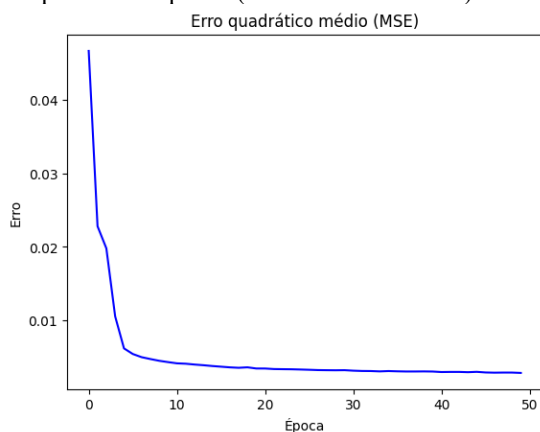
Figura 4.34 – Caso 6: Comparação entre a previsão e os dados reais (duas camadas ocultas).



Fonte: Autoria própria.

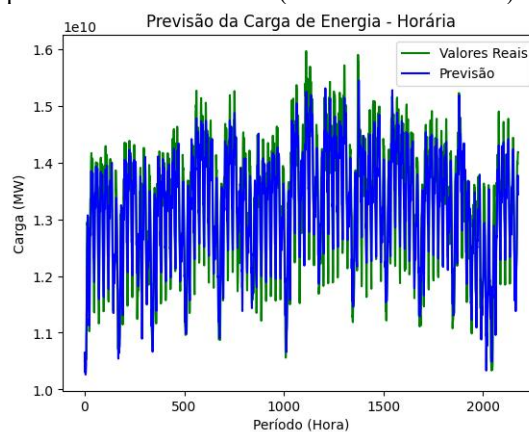
Com a adição de uma terceira camada, o MAPE aumenta ligeiramente para 2,37%, sugerindo uma leve degradação na precisão em comparação com o modelo de duas camadas. O R^2 diminui para 0,87, indicando que o modelo com três camadas explica 87% da variabilidade dos dados, ainda um bom ajuste, mas não tão eficaz quanto o modelo de duas camadas. O tempo de processamento sobe para 17,02 segundos, o que é maior do que o tempo registrado para o modelo de duas camadas, possivelmente devido à maior complexidade computacional. É apresentado na figura 4.35 o decaimento do erro ao longo de cada iteração. Enquanto é possível comparar na figura 4.36 as previsões do modelo com os dados reais. É possível perceber que a curva de previsão se ajusta aos valores reais de forma satisfatória, porém, com menor precisão do que o modelo com apenas duas camadas.

Figura 4.35 – Caso 6: Evolução do erro com o passar das épocas (três camadas ocultas).



Fonte: Autoria própria.

Figura 4.36 – Caso 6: Comparação entre a previsão e os dados reais (três camadas ocultas).



Fonte: Autoria própria.

O caso 6 apresenta um comportamento atípico para o modelo de uma camada, que apresentou um ajuste negativo e uma alta taxa no MAPE. No entanto, a adição de camadas melhorou o desempenho do modelo de forma significativa. O modelo com duas camadas ofereceu o melhor equilíbrio entre precisão, ajuste e eficiência computacional. A adição de uma terceira camada não trouxe melhorias significativas adicionais na precisão ou no ajuste.

4.7 Análise comparativa

A análise dos resultados obtidos nas diferentes configurações de modelos com redes neurais e MLP revelou informações importantes sobre o impacto das combinações de otimizadores e funções de ativação no desempenho do modelo. A seguir, sintetizam-se as conclusões principais de cada caso estudado.

1. Caso 1 – Otimizador: Adam; Função de Ativação: Linear

O modelo com duas camadas apresentou o melhor equilíbrio entre precisão e ajuste, com um MAPE de 1,59% e um R^2 de 0,93. Adicionar camadas além da segunda resultou em uma leve degradação na precisão e no ajuste, além de aumentar o tempo de processamento.

2. Caso 2 – Otimizador: Adam; Função de Ativação: ReLU

Similar ao Caso 1, o modelo com duas camadas também se destacou. O MAPE foi reduzido para 2,11% e o R^2 subiu para 0,87, demonstrando um desempenho eficaz. A adição de camadas adicionais não trouxe melhorias substanciais e aumentou o tempo de processamento.

3. Caso 3 – Otimizador: Nadam; Função de Ativação: Linear

O modelo com duas camadas mostrou o melhor desempenho com um MAPE de 1,49% e um R^2 de 0,94. A inclusão de uma terceira camada resultou em uma redução na precisão e no ajuste, além de um aumento significativo no tempo de processamento.

4. Caso 4 – Otimizador: Nadam; Função de Ativação: ReLU

Aqui, o modelo com duas camadas também apresentou o melhor equilíbrio. O MAPE foi de 2,40% e o R^2 de 0,87. A terceira camada não contribuiu para melhorar o desempenho e levou a um aumento no tempo de processamento.

5. Caso 5 – Otimizador: RMSprop; Função de Ativação: Linear

O modelo com duas camadas ofereceu o melhor desempenho com um MAPE de 2,22% e um R^2 de 0,89. A adição de uma terceira camada não melhorou a precisão e aumentou o tempo de processamento.

6. Caso 6 – Otimizador: RMSprop; Função de Ativação: ReLU

Este caso apresentou um comportamento atípico com o modelo de uma camada, que teve um ajuste negativo (R^2 de -0,45) e um MAPE de 9,60%. No entanto, o modelo com duas camadas melhorou substancialmente, com um MAPE de 2,04% e um R^2 de 0,90. A terceira camada não trouxe benefícios adicionais e aumentou o tempo de processamento.

Através da comparação entre os diferentes casos, observou-se que, em geral, o modelo com duas camadas se destacou como a configuração mais eficaz em termos de precisão e ajuste, independentemente da combinação de otimizador e função de ativação utilizada. A escolha da função de ativação e do otimizador impactou a eficiência do modelo, mas a adição de camadas além da segunda não resultou em melhorias significativas e, em todos os casos, piorou o desempenho ou aumentou o tempo de processamento.

É apresentado na tabela 4.7 uma síntese dos desempenhos dos modelos de rede neurais avaliados, considerando as diferentes combinações de otimizadores e funções de ativação.

Tabela 4.7 – Comparação entre os valores de MAPE, R^2 e tempo de processamento dos casos para a rede com duas camadas ocultas.

Otimizador	Função de Ativação	MAPE (%)	R^2	Tempo de Processamento (s)
Adam	Linear	1,59	0,93	14,62
Adam	ReLU	2,11	0,87	21,88
Nadam	Linear	1,49	0,94	16,41
Nadam	ReLU	2,40	0,87	17,27
RMSprop	Linear	2,22	0,89	14,97
RMSprop	ReLU	2,04	0,90	14,96

Fonte: Autoria própria.

A análise dos dados revela que a combinação Nadam com a função de ativação linear é a mais eficaz, apresentando o menor MAPE e o maior R^2 , refletindo a melhor precisão e ajuste dos modelos testados. Essa configuração também oferece um tempo de processamento aceitável, equilibrando desempenho e eficiência.

Por outro lado, as combinações com a função ReLU, especialmente com o otimizador Adam, mostram um desempenho inferior em termos de precisão e ajuste, além de um aumento no tempo de processamento. A configuração RMSprop com ativação ReLU oferece uma performance aceitável, mas ainda não é tão eficaz quanto outras combinações.

De posse da melhor configuração, baseada na combinação entre otimizador e função de ativação, para o horizonte de previsão de vinte e quatro horas, seguirão agora cenários baseados em novos horizontes de previsão: (i) previsão de 72 horas (3 dias); (ii) previsão de 168 horas (7 dias); e (iii) previsão de 720 horas (30 dias).

4.8 Cenário 1 – Previsão de 72 horas

É apresentado na tabela 4.8 os dados obtidos da simulação para o horizonte de 72 horas. Como observado, o modelo de previsão teve um desempenho notável com um MAPE baixo e um R^2 alto, indicando alta precisão e boa capacidade de explicação dos dados com um tempo de processamento razoável, mostrando que o modelo ainda consegue apresentar uma boa previsão para o cenário em estudo.

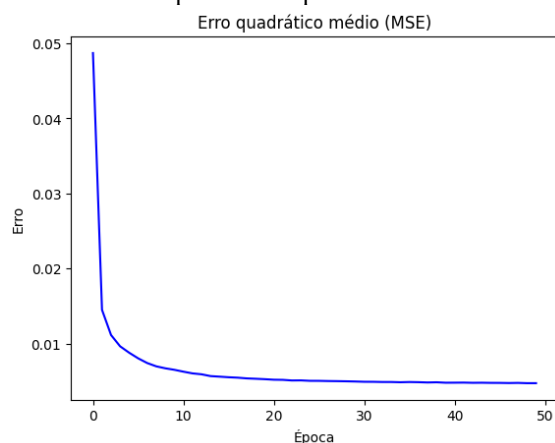
Tabela 4.8 – Cenário 1: MAPE, R² e tempo de processamento.

Horizonte de Previsão	MAPE (%)	R ²	Tempo de Processamento (s)
72 horas	1,79	0,92	22,19

Fonte: Autoria própria.

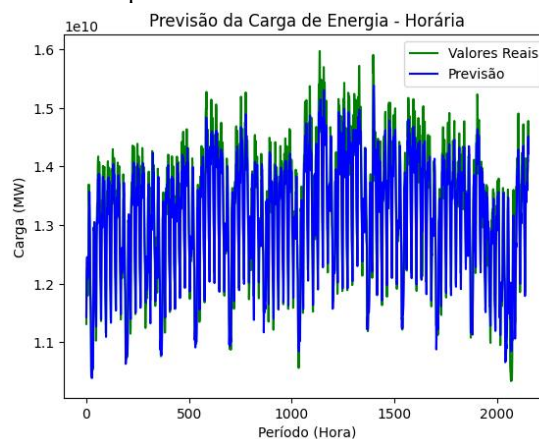
É ilustrado na figura 4.37 a evolução do erro com o passar das épocas. De forma semelhante ao obtido no cenário de 24 horas, o desempenho do erro sofreu um decaimento ao longo das iterações, o que demonstra uma boa adaptação do modelo à previsão sugerida. E é apresentado na figura 4.38 a eficiência do modelo de previsão de carga de energia, comparando os valores reais e previstos ao longo de um período de 72 horas.

Figura 4.37 – Cenário 1: Evolução do erro com o passar das épocas.



Fonte: Autoria própria.

Figura 4.38 – Cenário 1: Comparação entre a previsão e os dados reais.



Fonte: Autoria própria.

A sobreposição significativa entre os valores reais e previsões reflete a alta precisão do modelo, conforme indicado por um MAPE de 1,79% e um R² de 0,92, que foram destacados na tabela 4.8. Essa precisão mostra que o modelo consegue captar adequadamente as variações na demanda de energia, apresentando previsões que seguem de perto os padrões reais observados. Além disso, o tempo de processamento de 22,19 segundos reforça a eficiência do modelo, tornando-o adequado para aplicações que exigem previsões rápidas e confiáveis.

4.9 Cenário 2 – Previsão de 168 horas

A Tabela 4.9 apresenta os resultados da simulação para o horizonte de previsão de 168 horas. O modelo de previsão apresentou, de forma semelhante ao anterior, um desempenho elevado, com um MAPE de 1,87%, refletindo um baixo erro percentual e, portanto, alta precisão. O R² de 0,91 indica uma forte capacidade do modelo em explicar a variação dos dados de carga de energia. Além disso, o tempo de processamento de 35,99

segundos é bastante razoável para um horizonte de previsão de sete dias, demonstrando que o modelo é eficiente e adequado para gerar previsões precisas em um prazo semanal.

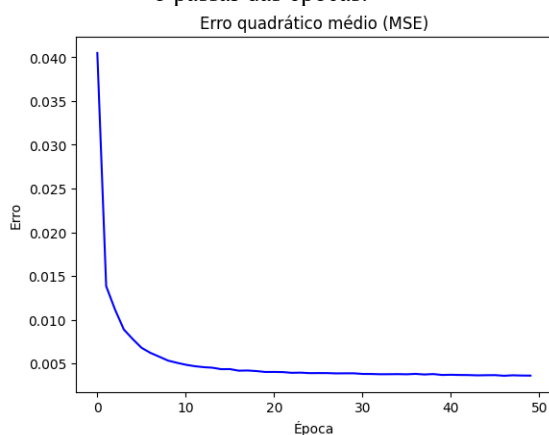
Tabela 4.9 – Cenário 2: MAPE, R^2 e tempo de processamento.

Horizonte de Previsão	MAPE (%)	R^2	Tempo de Processamento (s)
168 horas	1,87	0,91	35,99

Fonte: Autoria própria.

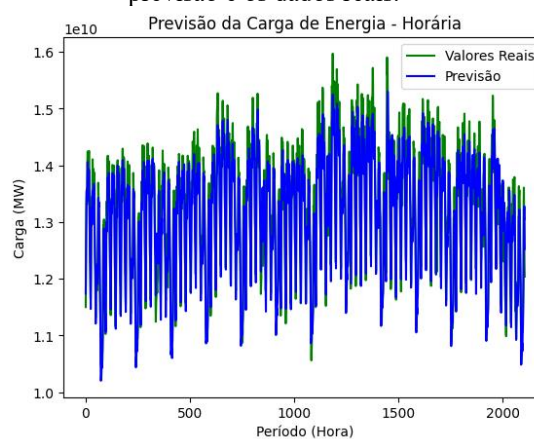
É apresentado na figura 4.39 o decaimento do erro com o passar das épocas. É possível visualizar na figura 4.40 a precisão do modelo de previsão de carga de energia ao comparar os valores reais e previstos para um período de 168 horas.

Figura 4.39 – Cenário 2: Evolução do erro com o passar das épocas.



Fonte: Autoria própria.

Figura 4.40 – Cenário 2: Comparação entre a previsão e os dados reais.



Fonte: Autoria própria.

A estreita correspondência entre os dados reais e as previsões evidencia a eficiência do modelo, conforme refletido pelo MAPE de 1,87% e o R^2 de 0,91 mencionados na Tabela 4.9. Esses indicadores confirmam que o modelo captura de forma eficaz as flutuações na demanda de energia, gerando previsões que acompanham de perto os dados observados.

4.10 Cenário 3 – Previsão de 720 horas

É apresentado na tabela 4.10 os resultados da simulação para o horizonte de previsão de 720 horas. O modelo de previsão demonstra um desempenho satisfatório, com um MAPE de 3,17%, indicando um erro percentual moderado. O R^2 de 0,77 sugere uma boa capacidade do modelo em explicar a variação dos dados, embora com menor precisão em comparação a horizontes mais curtos. O tempo de processamento de 483,90 segundos é significativamente maior, o que é esperado devido ao aumento do horizonte

de previsão, mas ainda assim viável para aplicações que exigem previsões de médio prazo.

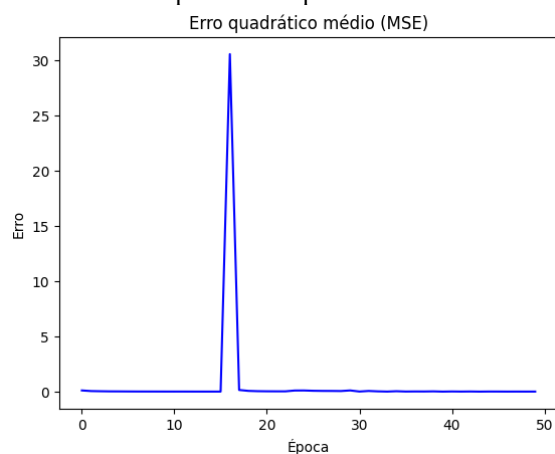
Tabela 4.10 – Cenário 3: MAPE, R^2 e tempo de processamento.

Horizonte de Previsão	MAPE (%)	R^2	Tempo de Processamento (s)
720 horas	3,17	0,77	483,90

Fonte: Autoria própria.

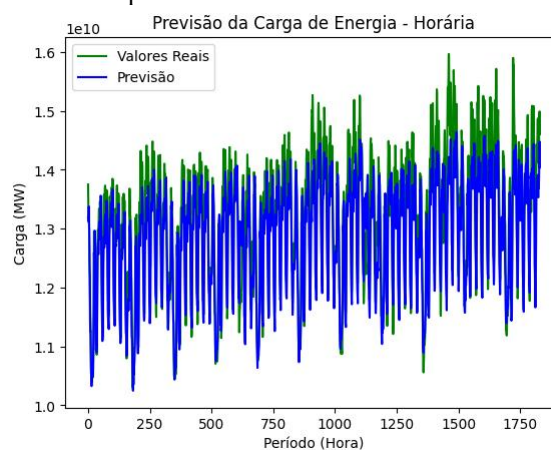
É ilustrado na figura 4.41 a curva do erro com o passar das épocas. Nesse caso, a evolução do erro não ocorreu conforme os demais casos, de forma a decair ao longo das épocas.

Figura 4.41 – Cenário 3: Evolução do erro com o passar das épocas.



Fonte: Autoria própria.

Figura 4.42 – Cenário 3: Comparação entre a previsão e os dados reais.



Fonte: Autoria própria.

O erro aparenta possuir um comportamento constante até próximo da 15ª época. O pico subsequente pode sinalizar que o modelo começou a se ajustar excessivamente aos dados de treinamento, capturando ruídos ou padrões que não generalizam de forma adequada para novos dados. Após o pico, o erro retorna ao comportamento anterior, sugerindo que o modelo se estabilizou após o *overfitting* inicial, possivelmente por meio de algum mecanismo de regularização ou ajuste no treinamento.

E é apresentado na figura 4.42 a eficiência do modelo de previsão de carga de energia ao longo de um período de 720 horas, comparando os valores reais e previstos. Embora haja uma sobreposição entre os valores reais e as previsões, refletindo a capacidade do modelo, a precisão é menor em comparação a horizontes mais curtos, conforme indicado pelos indicadores destacados na Tabela 4.10. Esses resultados sugerem que o modelo ainda capta as variações na demanda de energia, mas com menor acurácia. Além disso, o tempo de processamento elevado, de 483,9 segundos

(aproximadamente 8 minutos), torna essa previsão de 720 horas menos interessante, especialmente quando comparada a modelos de menor horizonte que oferecem maior precisão e eficiência.

4.11 Comentários

Os resultados para os cenários de horizontes de previsão analisados são sintetizados na Tabela 4.11.

Tabela 4.11 – MAPE, R² e tempo de processamento dos três cenários.

Horizonte de Previsão	MAPE (%)	R²	Tempo de Processamento (s)
24 horas	1,49	0,94	16,41
72 horas	1,79	0,92	22,19
168 horas	1,87	0,91	35,99
720 horas	3,17	0,77	483,90

Fonte: Autoria própria.

O modelo de previsão mostrou-se altamente eficiente para os horizontes de 24, 72 e 168 horas, apresentando baixo erro percentual e um ajuste robusto aos dados reais, o que o torna confiável para previsões de curto e médio prazo, sendo esse último em até poucas semanas. No entanto, para o horizonte de 720 horas, ou seja, um mês, os resultados foram menos satisfatórios. Nesse cenário, o MAPE foi significativamente maior, indicando um aumento no erro percentual das previsões, e o R² apresentou um ajuste mais baixo, sugerindo uma menor capacidade do modelo em capturar a variabilidade dos dados ao longo de um período mais extenso. Apesar disso, é importante destacar que esses valores ainda são considerados aceitáveis pela comunidade científica, especialmente para previsões de longo prazo, onde a complexidade e a incerteza tendem a aumentar.

5 CONSIDERAÇÕES FINAIS

A previsão precisa da demanda de energia é essencial para a gestão eficiente e sustentável dos sistemas elétricos, especialmente em contextos de variabilidade climática e aumento da integração de fontes renováveis intermitentes. Neste trabalho foi proposto o desenvolvimento de um algoritmo de previsão de carga elétrica, combinando técnicas de análise de séries temporais com aprendizado de máquina, com ênfase no MLP.

Realizou-se uma série de experimentos para avaliar o impacto de diferentes combinações de otimizadores e funções de ativação na performance dos modelos de redes neurais artificiais para a previsão de carga no horizonte de 24 horas à frente. A análise revelou que os modelos com duas camadas ocultas, utilizando o otimizador Nadam e a função de ativação Linear apresentaram o melhor desempenho geral, com o menor erro percentual (MAPE) e um coeficiente de determinação robusto (R^2). Esses modelos alcançaram um MAPE de 1,49% e um R^2 de 0,94, evidenciando a alta precisão e robustez na captura das variações na carga de energia elétrica.

Os resultados evidenciam que a escolha adequada do otimizador e da função de ativação é crucial para otimizar a precisão e o ajuste dos modelos preditivos. O modelo com duas camadas mostrou um equilíbrio favorável entre precisão, ajuste e eficiência computacional. No entanto, observou-se um desempenho atípico no Caso 6, onde um R^2 negativo de $-0,45$ foi registrado para o modelo com uma camada, destacando a importância de testar múltiplas configurações para identificar a melhor solução.

Além disso, os modelos com duas camadas demonstraram excelente desempenho em outros horizontes de previsão. Para 72 horas, o MAPE foi de 1,79% e R^2 foi de 0,92. No caso de 168 horas, o MAPE foi de 1,87% e o R^2 de 0,91. No entanto, a performance do modelo para o horizonte de 720 horas (30 dias) mostrou um aumento significativo no MAPE, de 3,17% e uma diminuição no R^2 de 0,77, indicando que o modelo possui uma menor precisão em previsões de longo prazo. O aumento no tempo de processamento também destaca as limitações do modelo quando aplicado a períodos extensos.

A eficácia do modelo é particularmente notável considerando a arquitetura do banco de dados, que integra e organiza dados de forma otimizada, garantindo que o modelo tenha acesso rápido e eficaz as informações necessárias, potencializando a performance do otimizador Nadam maximizando a utilidade da função de ativação Linear.

Como sugestões de trabalhos futuros, sugere-se:

- Aplicação de novas arquiteturas de redes neurais, como CNN (*Convolutional Neural Network*) para extrair padrões temporais complexos e RNN (*Recurrent Neural Network*) ou LSTM (*Long Short-Term Memory*) para capturar dependências de longo prazo.
- Além disso, a incorporação de dados adicionais meteorológicos, como temperatura, umidade e outros fatores, podem enriquecer os modelos preditivos, oferecendo uma visão mais holística das variáveis que influenciam a demanda de energia elétrica.

6 REFERÊNCIAS

ACKERMANN, A. E. F.; SELBITTO, M. A. Métodos de previsão de demanda: uma revisão da literatura. *Innovar*, v. 32, n. 85, p. 83–99, 2022. Disponível em: <https://doi.org/10.15446/innovar.v32n85.100979>. Acesso em: 05 abr. 2024.

Almada, Marco. (2019). Modelos de neurônios em redes neurais artificiais. 10.13140/RG.2.2.32042.16323.

Arlot, S., & Celisse, A. (2010). "A Survey of Cross-Validation Procedures". *Statistics Surveys*.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

BOX, G. E. P. and JENKINS, G. M. (2008). *Time series analysis: forecasting and control*, 4th ed. San Francisco: Holden-Day.

Brockwell, Peter J. and Davis, Richard A. (2002). *Introduction to Time Series and Forecasting*, 2nd ed. Springer-Verlag.

Brasil, Ministério de Minas e Energia, Empresa de Pesquisa Energética. Plano Nacional de Energia 2050 / Ministério de Minas e Energia. Empresa de Pesquisa Energética. Brasília: MME/EPE, 2020. Disponível em: <https://www.epe.gov.br/pt/publicacoes-dados-abertos/publicacoes/Plano-Nacional-de-Energia-2050>. Acesso em: 05 abr. 2024.

CAMELO, H. DO N. et al. Métodos de Previsão de Séries Temporais e Modelagem Híbrida ambos Aplicados em Médias Mensais de Velocidade do Vento para Regiões do Nordeste do Brasil. *Revista Brasileira de Meteorologia*, v. 32, n. 4, p. 565–574, 2017.

CASTRO, N; et al. Impactos Sistêmicos da Micro e Minigeração Distribuída. Disponível em: https://agora.ie.ufrj.br/pdf/Nivalde_de_Castro/12.tdse79_2018_Impactos_sistemicos_da_micro.pdf. Acesso em: 30 jul. 2024.

CCEE – 1ª Revisão Quadrimestral das Previsões de carga para o Planejamento Anual da Operação Energética 2023-2027. Disponível em: https://www.ccee.org.br/documents/80415/26530371/Boletim%20Tecnico%20PLAN_2024_2028.pdf/e7ba819c-577f-b415-1604-5ce39e65b5aa. Acesso em: 19 jul. 2024.

CEPEL – Centro de Pesquisas de Energia Elétrica. Disponível em: <https://see.cepel.br/manual/libs/latest/apresentacao/apresentacao.html>. Acesso em: 15 jul 2024a.

CEPEL – Centro de Pesquisas de Energia Elétrica. Disponível em: <https://www.cepel.br/linhas-de-pesquisa/newave/#:~:text=O%20NEWAVE%20%C3%A9%20um%20programa%20computacional%20de>. Acesso em: 02 out. 2024b.

CEPEL – Centro de Pesquisas de Energia Elétrica. Disponível em: <https://www.cepel.br/linhas-de-pesquisa/decomp/#:~:text=DECOMP:%20Modelo%20de%20Planejamento%20de%20Curto>. Acesso em: 02 out. 2024c.

CEPEL – Centro de Pesquisas de Energia Elétrica. Disponível em: <https://www.cepel.br/linhas-de-pesquisa/dessem-saibamais/#:~:text=O%20programa%20DESSEM%20%C3%A9%20um%20modelo%20de>. Acesso em: 02 out. 2024d.

Conte, S.; Limão, R.; Oliveira. Aplicação Híbrida com Redes Neurais Profundas e Algoritmo Genético para Previsão de Séries Temporais do Sistema de Energia Elétrica Brasileira. [s.l.: s.n.]. Disponível em: https://sbic.org.br/wp-content/uploads/2021/09/pdf/CBIC_2021_paper_104.pdf. Acesso em: 19 jul. 2024.

DRANKA, G. G.; FERREIRA, P. Planning for a renewable future in the Brazilian power system. *Energy*, v. 164, p. 496–511, 1 dez. 2018. Disponível em: <https://doi.org/10.1016/j.energy.2018.08.164>. Acesso em: 15 jul. 2024.

Driscoll, M. Jupyter notebook: An introduction. Disponível em: <https://realpython.com/jupyter-notebook-introduction/>. Acesso em: 26 set. 2024.

EPE – Planejamento Energético Planejamento de Longo Prazo. Disponível em: <https://www.epe.gov.br/pt/areas-de-atuacao/planejamento-energetico/planejamento-de-longo-prazo>. Acesso em: 19 jul. 2024.

ENERGIA, A. C. Demanda Global por Eletricidade em Rápida Ascensão: Energias Renováveis e Desafios à Frente. *Cenário Energia*, 19 de julho de 2024. Disponível em: <https://cenarioenergia.com.br/2024/07/19/demanda-global-por-eletricidade-em-rapida-ascensao-energias-renovaveis-e-desafios-a-frente/>. Acesso em: 19 jul. 2024.

FALCÃO, P. D. M. Impacto da mini e microgeração distribuída nas redes de distribuição de energia elétrica. Disponível em: https://www.ieee.org.br/wp-content/uploads/2014/05/EspacoIEEE_set2013.pdf. Acesso em: 30 jul. 2024.

GOMES, P. C. T. Introdução ao aprendizado de máquina. Disponível em: <https://www.datageeks.com.br/aprendizado-de-maquina/>. Acesso em: 19 jul. 2024.

G. Gross and F. D. Galiana, "Short-term load forecasting," in *Proceedings of the IEEE*, vol. 75, no. 12, pp. 1558-1573, Dec. 1987. doi: 10.1109/PROC.1987.13927.

Goodfellow, I; Bengio, Y; Courville, A. *Deep Learning*. Disponível em: <https://www.deeplearningbook.org/>. Acesso em: 2 out. 2024.

Haykin, S. (2009). *Neural Networks and Learning Machines* (3rd ed.). Prentice Hall.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: principles and practice*. [s.l.] Otexts, 2018. Disponível em: <https://otexts.com/fpp2/>. Acesso em: 05 abr. 2024.

IBGE. (2023). Análise da Agricultura Familiar no Nordeste. Disponível em: <https://www.ibge.gov.br>. Acesso em 21 set. 2024.

IBGE. Disponível em: <https://www.ibge.gov.br/nordeste>. Acesso em: 21 set. 2024.

IBM – Soluções de inteligência artificial (IA). Disponível em: <https://www.ibm.com/br-pt/artificial-intelligence>. Acesso em: 19 jul. 2024.

JUPYTER. Jupyter Documentation. Disponível em: <https://jupyter.org/documentation>. Acesso em: 25 set. 2024.

KONG, W. et al. Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network. IEEE Transactions on Smart Grid, v. 10, n. 1, p. 841–851, jan. 2019. Disponível em: <https://ieeexplore.ieee.org/document/8039509>. Acesso em: 15 jul 2024.

MARQUES, V. Região Nordeste. Disponível em: <https://www.todamateria.com.br/regiao-nordeste/>. Acesso em: 21 set. 2024.

MATPLOTLIB. Matplotlib Documentation. Disponível em: <https://matplotlib.org/stable/contents.html>. Acesso em: 15 jul. 2024.

McKinney, W. Python for data analysis, 2023, 3 Ed. Disponível em: <https://wesmckinney.com/book/>. Acesso em: 15 jul. 2024.

MEDEIROS, A. Aplicação de redes neurais artificiais para a previsão de demanda em subestações de distribuição considerando a influência da temperatura. Disponível em: <https://www.cear.ufpb.br/juan/wp-content/uploads/2022/07/2019-Aplica%C3%A7%C3%A3o-de-redes-neurais-artificiais-para-a-previs%C3%A3o-de-demanda-em-subesta%C3%A7%C3%B5es-de-distribui%C3%A7%C3%A3o-considerando-a-influ%C3%Aancia-da-temperatura.pdf>. Acesso em: 03 abr. 2024.

MME – Aneel regulamenta marco legal da micro e da minigeração distribuída. Disponível em: <https://www.gov.br/mme/pt-br/assuntos/noticias/aneel-regulamenta-marco-legal-da-micro-e-da-minigeracao-distribuida>. Acesso em: 30 jul. 2024.

MME – Fontes renováveis responderam por 93,1% da geração de energia elétrica em 2023. Disponível em: <https://www.gov.br/mme/pt-br/assuntos/noticias/fontes-renovaveis-responderam-por-93-1-da-geracao-de-energia-eletrica-em-2023>. Acesso em: 19 jul. 2024.

NOGUEIRA, L. C.; SALGADO, R. M. Comparação de Estratégias de Previsão de Carga Elétrica Utilizando Support Vector Machine. Disponível em: https://www.unifal-mg.edu.br/dcc/wp-content/uploads/sites/221/2022/01/ArtigoFinal_LeandraCarvalhoNogueira.pdf. Acesso em: 19 jul. 2024.

NUMPY. NumPy Documentation. Disponível em: <https://numpy.org/doc/stable/>. Acesso em: 15 jul. 2024.

ONS – Operador Nacional do Sistema Elétrico. Disponível em: <https://www.ons.org.br/paginas/energia-agora/carga-e-geracao>. Acesso em: 15 jul 2024.

ONS – Operador Nacional do Sistema Elétrico. Disponível em: <https://www.ons.org.br/Paginas/Noticias/20230816---ONS-iniciou-processo-de-licita%C3%A7%C3%A3o-para-contrata%C3%A7%C3%A3o-de-consultoria-para-o-subprojeto-de-previs%C3%A3o-de-gera%C3%A7%C3%A3o-de-fo.aspx>. Acesso em: 19 jul. 2024.

ORIGO – Questão energética no Brasil: entenda a situação do país. Disponível em: <https://origoenergia.com.br/blog/energia/questao-energetica-no-brasil-entenda-a-situacao-atual-do-pais>. Acesso em: 19 jul. 2024.

PANDAS. Pandas Documentation. Disponível em: <https://pandas.pydata.org/docs/>. Acesso em: 15 jul. 2024.

PESSANHA, J. F. M.; CEPTEL, B. UM MODELO PARA PREVISÃO DE CARGA DE CURTO-PRAZO. Disponível em: <https://www.aedb.br/seget/arquivos/artigos13/44218518.pdf>. Acesso em: 19 jul. 2024.

PIRES, C. L. Metodologia para previsão de carga e geração no horizonte de curtíssimo prazo. 2016. Disponível em: <https://repositorio.ufsm.br/handle/1/860>. Acesso em: 17 jul 2024.

PYTHON. Python Documentation. Disponível em: <https://docs.python.org/3/>. Acesso em: 10 jul 2024.

RUDER, S. An overview of gradient descent optimization algorithms. 2016. Disponível em: <http://arxiv.org/abs/1609.04747>.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533-536.

SCHRÖER, C.; KRUSE, F.; GÓMEZ, J. M. A Systematic Literature Review on Applying CRISP-DM Process Model. *Procedia Computer Science*, Volume 181, 2021, Pages 526-534. Disponível em: <https://doi.org/10.1016/j.procs.2021.01.199>. Acesso em: 19 jul. 2024.

SEABORN. Seaborn Documentation. Disponível em: <https://seaborn.pydata.org/>. Acesso em: 15 jul 2024.

Sanchez, G. MATLAB vs. Python: Which one is right for you? Disponível em: <https://www.mathworks.com/products/matlab/matlab-vs-python.html>. Acesso em: 3 out. 2024.

SILVA, L. C. Aprendizado de máquina com treinamento continuado aplicado à previsão de demanda de curto prazo: o caso do restaurante universitário da Universidade Federal de Uberlândia. Mestrado (Dissertação) — Uberlândia: Universidade Federal de Uberlândia, 27 maio 2019.

SILVEIRA, J.; DA, F.; ALEGRE, S. UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL ESCOLA DE ENGENHARIA PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO MODELAGEM DE EQUAÇÕES ESTRUTURAIS: apresentação de uma metodologia. Disponível em: <https://lume.ufrgs.br/bitstream/handle/10183/8628/000583213.pdf>. Acesso em: 19 jul. 2024.

Spiliotis, E. (2023). Time Series Forecasting with Statistical, Machine Learning, and Deep Learning Methods: Past, Present, and Future. In: Hamoudia, M., Makridakis, S., Spiliotis, E. (eds) Forecasting with Artificial Intelligence. Palgrave Advances in the Economics of Innovation and Technology. Palgrave Macmillan, Cham. https://doi.org/10.1007/978-3-031-35879-1_3

Tieleman, T. e Hinton, G. Aula 6.5 - RMSProp, COURSERA: Neural Networks for Machine Learning. Relatório técnico, 2012.

WALTRICK, C. Machine Learning – O que é, tipos de aprendizagem de máquina, algoritmos e aplicações. Disponível em: <https://medium.com/camilawaltrick/introducao-machine-learning-o-que-e-tipos-de-aprendizado-de-maquina-445dcfb708f0>. Acesso em: 19 jul. 2024.

ZHONG, G. et al. Long Short-Term Attention. 2018. Disponível em: <http://arxiv.org/abs/1810.12752>. Acesso em: 19 jul. 2024.

7 APÊNDICES

Apêndice A – Link GitHub

Este apêndice contém informações referente aos códigos utilizados durante a elaboração deste trabalho, bem como os arquivos de banco de dados. Essa disponibilização visa facilitar a reprodução dos experimentos e a verificação dos resultados apresentados.

Os arquivos podem ser acessados através do seguinte link: [gabrielsbgr/TCC-Multilayer-Perceptron \(github.com\)](https://github.com/gabrielsbgr/TCC-Multilayer-Perceptron).

É recomendável aos usuários que verifiquem as instruções contidas no repositório para garantir uma correta configuração do ambiente e a execução dos códigos.