

Universidade Federal de Juiz de Fora  
Instituto de Ciências Exatas/Faculdade de Engenharia  
Programa de Pós-Graduação em Modelagem Computacional

**Gisele Goulart Tavares da Silva**

**Aplicação de técnicas de inteligência computacional para a previsão de  
cargas de aquecimento e resfriamento em edificações**

Juiz de Fora  
12 de Setembro de 2019

Gisele Goulart Tavares da Silva

**Aplicação de técnicas de inteligência computacional para a previsão de cargas de aquecimento e resfriamento em edificações**

Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Computacional da Universidade Federal de Juiz de Fora, na área de concentração em Sistemas Computacionais, como requisito parcial para obtenção do título de Mestre em Modelagem Computacional.

Orientadora: Dra. Priscila Vanessa Zabala Capriles Goliatt

Coorientador: Dr. Leonardo Goliatt da Fonseca

Juiz de Fora

12 de Setembro de 2019

Ficha catalográfica elaborada através do programa de geração automática da Biblioteca Universitária da UFJF, com os dados fornecidos pelo(a) autor(a)

da Silva, Gisele Goulart Tavares .

Aplicação de técnicas de inteligência computacional para a previsão de cargas de aquecimento e resfriamento em edificações / Gisele Goulart Tavares da Silva. -- 2019.

81 f. : il.

Orientadora: Priscila Vanessa Zabala Capriles Goliatt

Coorientador: Leonardo Goliatt da Fonseca

Dissertação (mestrado acadêmico) - Universidade Federal de Juiz de Fora, ICE/Engenharia. Programa de Pós-Graduação em Modelagem Computacional, 2019.

1. Eficiência Energética. 2. Cargas de Aquecimento e Resfriamento. 3. Extreme Gradient Boosting. 4. Evolução Diferencial. 5. Seleção de Modelos. I. Goliatt, Priscila Vanessa Zabala Capriles, orient. II. da Fonseca, Leonardo Goliatt, coorient. III. Título.

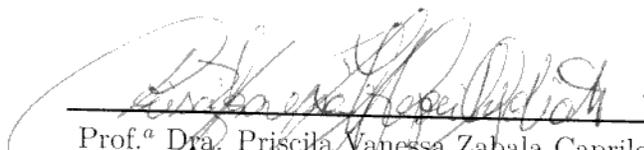
Gisele Goulart Tavares da Silva

Aplicação de técnicas de inteligência computacional para a previsão de cargas de aquecimento e resfriamento em edificações

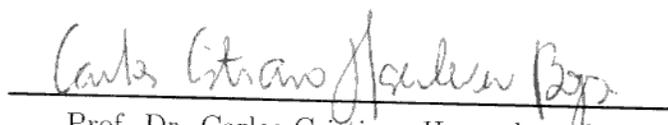
Dissertação apresentada ao Programa de Pós-Graduação em Modelagem Computacional da Universidade Federal de Juiz de Fora, na área de concentração em Sistemas Computacionais, como requisito parcial para obtenção do título de Mestre em Modelagem Computacional.

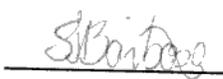
Aprovada em: 12 de Setembro de 2019

BANCA EXAMINADORA

  
\_\_\_\_\_  
Prof.<sup>a</sup> Dra. Priscila Vanessa Zabala Capriles Goliatt -  
Orientadora  
Universidade Federal de Juiz de Fora

  
\_\_\_\_\_  
Prof. Dr. Leonardo Goliatt da Fonseca - Coorientador  
Universidade Federal de Juiz de Fora

  
\_\_\_\_\_  
Prof. Dr. Carlos Cristiano Hasenclever Borges  
Universidade Federal de Juiz de Fora

  
\_\_\_\_\_  
Prof.<sup>a</sup> Dra. Sabrina Andrade Barbosa  
Universidade do Estado do Rio de Janeiro

*Dedico este trabalho à minha família, pelo apoio incondicional.*

## AGRADECIMENTOS

A Deus por me conceder forças em toda a caminhada. Aos meus pais Jorge e Cleonice, meu irmão Gabriel e meu noivo Leandro pela paciência e apoio durante toda a jornada. A todos os meus familiares, pelo encorajamento.

A todos os colegas do PGMC pelos auxílios prestados e pelo apoio durante este período de mestrado.

Aos professores Leonardo Goliatt e Priscila Capriles pela orientação, amizade, incentivo e principalmente, pela paciência, sem a qual este trabalho não se realizaria. Aos professores Carlos Cristiano e Sabrina Barbosa por terem aceitado o convite para a banca e pelas contribuições na avaliação do trabalho.

Aos professores do Programa de Pós-Graduação em Modelagem Computacional pelos seus ensinamentos e aos funcionários do curso, que durante esses anos, contribuíram de algum modo para o meu enriquecimento pessoal e profissional.

Por fim, agradeço às agências Capes e Fapemig pelo suporte financeiro oferecido neste período de mestrado.

## RESUMO

Devido às mudanças climáticas, até 2040 os edifícios podem vir a consumir 30% mais energia, sendo o desempenho energético o elemento chave para o alcance do desenvolvimento sustentável no setor da construção civil. Custos de projeto, dano ambiental e características construtivas como geometria, propriedades térmicas dos materiais e condições climáticas são necessárias para avaliar a performance energética de uma edificação. Uma forma de solucionar este problema de avaliação é a aplicação de Métodos de Aprendizado de Máquina para estimar uma resposta a partir de uma entrada de dados. O uso de Métodos de Aprendizado de Máquina pode vir a auxiliar especialistas da área da construção civil na análise de cenários ainda na fase inicial do projeto e trazer economia para a edificação. O presente trabalho avalia a aplicação dos modelos *Elastic Net*, Máquina de Aprendizado Extremo e *Extreme Gradient Boosting* para a previsão das cargas de aquecimento e resfriamento em edificações residenciais. Duas bases de dados são utilizadas para avaliar o desempenho dos métodos, contendo variáveis geométricas de entrada e duas variáveis térmicas de saída. Para a seleção de parâmetros dos métodos, o algoritmo de otimização Evolução Diferencial foi aplicado, com o objetivo de encontrar os conjuntos de hiperparâmetros que reforcem as capacidades preditivas dos modelos. As comparações dos resultados ocorreram através do uso das métricas MAE, MAPE, RMSE e  $R^2$ . Os resultados mostraram que o método *Extreme Gradient Boosting* obteve uma melhor performance dentre os métodos testados e também em comparação com a literatura, apresentando os menores valores para as métricas de erro e diferença significativa nos testes estatísticos. Além disso, o algoritmo de Evolução Diferencial se mostrou eficaz para a otimização dos parâmetros dos modelos testados, podendo vir a ser aplicado também a outros modelos da literatura. Dessa forma, a combinação dos métodos Evolução Diferencial e *Extreme Gradient Boosting* pode vir a ser aplicada na previsão das cargas térmicas em edificações, auxiliando em projetos que visem economia de energia e sustentabilidade.

Palavras-chave: Eficiência Energética. Cargas de Aquecimento e Resfriamento. *Extreme Gradient Boosting*. Evolução Diferencial. Seleção de Modelos.

## ABSTRACT

Due to climate change, by 2040 buildings may consume 20% more energy, with energy performance being the key element for achieving sustainable development in the building sector. Design costs, environmental damage and constructive characteristics, such as geometry, thermal properties of materials and weather conditions, are necessary to assess the energy performance of a building. One way to solve this problem is to apply Machine Learning Methods to estimate a response from data. The use of Machine Learning Methods can assist construction specialists in scenario analysis at an early stage of the project and bring savings to the building. The present work evaluates the application of the Elastic Net, Extreme Learning Machine and Extreme Gradient Boosting models for the forecasting of heating and cooling loads in residential buildings. Two databases are used to evaluate the performance of the methods, including geometric input variables and two thermal output variables. For the selection of method parameters, the Differential Evolution optimization algorithm was applied to find the sets of hyperparameters that reinforce the predictive capabilities of the models. Comparisons of the results were made using the MAE, MAPE, RMSE and  $R^2$  metrics. The results showed that the Extreme Gradient Boosting achieved a better performance among the tested methods and also compared to the literature, presenting the lowest values for the error metrics and significant difference in the statistical tests. In addition, the Differential Evolution algorithm showed to be effective for the optimization of the parameters of the tested models, and may also be applied to other models in the literature. Thus, a combination of Differential Evolution and Extreme Gradient Boosting methods can be applied to predict thermal loads in buildings, assisting in projects that aim at energy saving and sustainability.

Key-words: Energy Efficiency. Heating and Cooling Loads. Extreme Gradient Boosting. Differential Evolution. Model Selection.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Dimensões propostas para o consumo de energia. . . . .	18
Figura 2 – Geração das geometrias a partir dos 18 blocos. . . . .	26
Figura 3 – Exemplos das geometrias das edificações. . . . .	27
Figura 4 – Definição das áreas da edificação, onde AE representa a altura total, AC a área de piso, AT a área de telhado e AP a área de parede. . . . .	28
Figura 5 – Modelo base utilizado para a geração da base de dados de edificações residenciais. . . . .	29
Figura 6 – Conectividades para uma rede do tipo ELM com 8 entradas, uma camada oculta com 10 neurônios e uma saída. . . . .	34
Figura 7 – Diagrama do método de validação cruzada $K$ -fold com $K = 5$ . . . . .	39
Figura 8 – Fluxograma de execução do algoritmo de otimização DE em conjunto com os modelos de regressão NET, ELM e XGB. . . . .	40
Figura 9 – Distribuição do parâmetro <code>l1_ratio</code> do método <i>Elastic Net</i> aplicado a predição das cargas de aquecimento da BD1. . . . .	46
Figura 10 – Distribuições dos parâmetros <code>alpha</code> , <code>rbf_width</code> , <code>n_hidden</code> e <code>activation_func</code> utilizados no método ELM para a predição das cargas de aquecimento da BD1. . . . .	47
Figura 11 – Distribuições dos parâmetros <code>colsample_bytree</code> , <code>learning_rate</code> , <code>subsample</code> , <code>n_estimators</code> e as frequências dos parâmetros <code>min_child_weight</code> e <code>max_depth</code> utilizados no método XGB para a predição das cargas de resfriamento da BD1. . . . .	48
Figura 12 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de aquecimento da BD1. . . . .	49
Figura 13 – Distribuição de frequências para o parâmetro <code>l1_ratio</code> utilizado no método NET para predição das cargas de resfriamento da BD1. . . . .	50
Figura 14 – Distribuições dos parâmetros <code>alpha</code> , <code>rbf_width</code> , <code>n_hidden</code> e <code>activation_func</code> utilizados no método ELM para a predição das cargas de resfriamento da BD1. . . . .	51
Figura 15 – Distribuições dos parâmetros <code>colsample_bytree</code> , <code>learning_rate</code> , <code>subsample</code> , <code>n_estimators</code> , <code>min_child_weight</code> e <code>max_depth</code> utilizados no método XGB para a predição das cargas de resfriamento da BD1. . . . .	52
Figura 16 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de resfriamento da BD1. . . . .	53
Figura 17 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de aquecimento da BD1. . . . .	57
Figura 18 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de resfriamento da BD1. . . . .	57

Figura 19 – Distribuições dos parâmetros <code>alpha</code> e <code>l1_ratio</code> utilizados no método NET para a previsão das cargas de aquecimento para a BD2. . . . .	59
Figura 20 – Distribuições dos hiperparâmetros <code>alpha</code> , <code>rbf_width</code> , <code>activation_func</code> e <code>n_hidden</code> utilizados no método ELM para a previsão das cargas de aquecimento da BD2. . . . .	60
Figura 21 – Distribuições dos parâmetros <code>colsample_bytree</code> , <code>subsample</code> , <code>learning_rate</code> , <code>min_child_weight</code> , <code>max_depth</code> e <code>n_estimators</code> utilizados no método XGB para a predição das cargas de aquecimento para a BD2. . . . .	61
Figura 22 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de aquecimento da BD2.	62
Figura 23 – Distribuições dos parâmetros <code>alpha</code> e <code>l1_ratio</code> utilizados no método NET para a previsão das cargas de resfriamento para a BD2. . . . .	62
Figura 24 – Distribuições dos hiperparâmetros escolhidos pelo algoritmo evolutivo para o método ELM para a previsão das cargas de resfriamento da BD2.	63
Figura 25 – Distribuições dos parâmetros <code>colsample_bytree</code> e <code>subsample</code> , <code>learning_rate</code> , <code>min_child_weight</code> , <code>max_depth</code> e <code>n_estimators</code> utilizados no método XGB para a predição das cargas de resfriamento para a BD2. . . . .	64
Figura 26 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de resfriamento da BD2.	65
Figura 27 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de aquecimento da BD2. . . . .	69
Figura 28 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de resfriamento da BD2. . . . .	69

## LISTA DE TABELAS

Tabela 1 – Representação matemática, mínimos, máximos e total de valores possíveis para as variáveis de entrada e saída. . . . .	27
Tabela 2 – Representações matemáticas das variáveis de entrada e saída e o número de possíveis valores. . . . .	30
Tabela 3 – Funções de ativação para ELM. . . . .	32
Tabela 4 – Hiperparâmetros do método NET. . . . .	41
Tabela 5 – Hiperparâmetros do método ELM. . . . .	41
Tabela 6 – Hiperparâmetros do método XGB. . . . .	42
Tabela 7 – Média das métricas estatísticas obtidas ao longo das 30 execuções dos algoritmos para as cargas de aquecimento (HL) e resfriamento (CL) da BD1. Os melhores valores são mostrados em negrito e os desvios padrão entre parênteses. . . . .	46
Tabela 8 – Resultados dos p-valores para o teste de Shapiro-Wilk para as métricas referentes aos resultados da BD1. Em negrito são apresentados os valores que indicam a existência de normalidade. . . . .	52
Tabela 9 – Resultados dos p-valores para o teste de Lilliefors para as métricas referentes aos resultados da BD1. Em negrito são apresentados os valores que indicam a existência de normalidade. . . . .	53
Tabela 10 – Resumo dos resultados dos testes de Shapiro e Lilliefors sobre a normalidade das distribuições dos resultados das métricas avaliadas para a BD1. . . . .	54
Tabela 11 – Resultados dos p-valores retornados pelo teste não paramétrico de Kruskal-Wallis para as métricas obtidas a partir da BD1. . . . .	54
Tabela 12 – Resultados dos p-valores e diferenças obtidos através do teste de Dunn para a métrica MAPE da BD1. . . . .	55
Tabela 13 – Resultados das somas e médias das classificações obtidas através do teste de Dunn para a métrica MAPE da BD1. . . . .	55
Tabela 14 – Cargas de aquecimento – Comparação dos resultados obtidos pelos modelos deste estudo na BD1 com a dados da literatura. Os melhores valores são apresentados em negrito. . . . .	56
Tabela 15 – Carga de resfriamento – Comparação dos resultados obtidos pelos modelos deste estudo na BD1 com a dados da literatura. Os melhores valores são apresentados em negrito. . . . .	56
Tabela 16 – Média das métricas estatísticas obtidas ao longo das 30 execuções dos algoritmos para as cargas de aquecimento (HL) e resfriamento (CL) para BD2. Os melhores valores são mostrados em negrito e os desvios padrão entre parênteses. . . . .	58

Tabela 17 – Resultados dos p-valores para o teste de Shapiro-Wilk para as métricas referentes aos resultados da BD2. Em negrito são apresentados os valores que indicam a existência de normalidade. . . . .	66
Tabela 18 – Resultados dos p-valores para o teste de Lilliefors para as métricas referentes aos resultados da BD2. Em negrito são apresentados os valores que indicam a existência de normalidade. . . . .	66
Tabela 19 – Resumo dos resultados dos testes Shapiro e Lilliefors sobre normalidade das distribuições dos resultados das métricas avaliadas. . . . .	66
Tabela 20 – Resultados dos p-valores retornados pelo teste não paramétrico de Kruskal-Wallis para as métricas obtidas a partir da BD2. . . . .	67
Tabela 21 – Resultados dos p-valores e diferenças obtidos através do teste de Dunn para a métrica MAPE da BD2. . . . .	67
Tabela 22 – Resultados das somas e médias das classificações obtidas através do teste de Dunn para a métrica MAPE da BD2. . . . .	67
Tabela 23 – Cargas de aquecimento e resfriamento – Comparação dos resultados obtidos pelos modelos testados na BD2. Os melhores valores são apresentados em negrito. . . . .	68

## LISTA DE ABREVIATURAS E SIGLAS

DE	Evolução Diferencial, do inglês <i>Differential Evolution</i>
NET	<i>Elastic Net</i>
ELM	Máquina de Aprendizado Extremo, do inglês <i>Extreme Learning Machine</i>
XGB	<i>Extreme Gradient Boosting</i>
NZEB	Edificação de balanço energético nulo, do inglês <i>Net-Zero Energy Building</i>
KNN	K-vizinhos mais próximos, do inglês <i>K-NearestNeighbors</i>
SVM	Máquinas de Vetor Suporte, do inglês <i>Support Vector Machines</i>
DT	Árvores de Decisão, do inglês <i>Decision Trees</i>
RF	Floresta Aleatória, do inglês <i>RandomForest</i>
PCA	Análise de Componentes Principais, do inglês <i>Principal Component Analysis</i>
MLP	Perceptron Multicamadas, do inglês <i>Multilayer Perceptron</i>
EMARS	Splines de regressão adaptativa multivariada evolutiva, do inglês <i>Evolutionary Multivariate Adaptive Regression Splines</i>
CR	Compactação Relativa
LASSO	<i>Least Absolute Shrinkage and Selection Operator</i>
ANN	Redes Neurais Artificiais, do inglês <i>Artificial Neural Network</i>
RBF	Função de Base Radial, do inglês <i>Radial Basis Function</i>
GB	<i>Gradient Boosting</i>
MAPE	Erro Absoluto Percentual Médio, do inglês <i>Mean Absolute Percentage Error</i>
R <sup>2</sup>	Coefficiente de Determinação
RMSE	Raiz do Erro Quadrático Médio, do inglês <i>Root Mean Square Error</i>
MAE	Erro Médio Absoluto, do inglês <i>Mean Absolute Error</i>
BD1	Base de Dados 1

BD2	Base de Dados 2
HL	Carga de Aquecimento, do inglês <i>Heating Load</i>
CL	Carga de Resfriamento, do inglês <i>Cooling Load</i>

## SUMÁRIO

<b>1</b>	<b>Introdução</b>	<b>14</b>
1.1	Apresentação do Tema e Contextualização do Problema	14
1.2	Justificativa	16
1.3	Objetivos Gerais e Específicos	16
<b>2</b>	<b>Revisão da Literatura</b>	<b>18</b>
2.1	Aprendizado de Máquina para a Previsão de Cargas energéticas	21
2.2	Emprego de Redes Neurais Profundas em Aprendizado Supervisionado	22
2.3	Algoritmos Evolutivos aplicados à Eficiência Energética	23
2.4	Regressores <i>Ensemble</i>	24
2.5	Otimização de Parâmetros em Métodos de Aprendizado de Máquina	24
<b>3</b>	<b>Material e Métodos</b>	<b>26</b>
3.1	Bases de Dados	26
3.1.1	Base de Dados 1	26
3.1.2	Base de Dados 2	29
3.2	Métodos de Regressão	30
3.2.1	Regressão <i>Elastic Net</i>	30
3.2.2	Máquina de Aprendizado Extremo	31
3.2.3	<i>Extreme Gradient Boosting</i>	33
3.3	Evolução Diferencial	36
3.4	Validação Cruzada	38
3.5	Seleção de Modelos Baseada em Evolução Diferencial	38
3.6	Métricas de Avaliação	42
3.7	Testes Estatísticos	43
<b>4</b>	<b>Resultados e Discussão</b>	<b>45</b>
4.1	Base de Dados 1 (BD1)	45
4.2	Base de Dados 2 (BD2)	58
<b>5</b>	<b>Conclusões e Trabalhos Futuros</b>	<b>70</b>
<b>6</b>	<b>Publicações Científicas</b>	<b>72</b>
	<b>REFERÊNCIAS</b>	<b>74</b>

# 1 Introdução

## 1.1 Apresentação do Tema e Contextualização do Problema

A construção civil é uma das atividades humanas que mais demanda energia e recursos naturais, causando graves impactos ambientais ao modificar o ambiente natural para o construído. Além disso, possui papel ativo no desenvolvimento sustentável, reduzindo disparidades sociais e conservando o meio ambiente. Para o alcance deste desenvolvimento, o setor da construção precisa adotar uma postura inovadora, introduzindo novos conceitos e procedimentos para o planejamento e a construção de edificações sustentáveis [53].

O aumento da população e o crescente uso de novas tecnologias têm resultado no surgimento de maiores demandas energéticas, podendo acarretar um aumento no consumo de cerca de 30% até 2040 [58]. No Brasil, apenas no setor residencial, estima-se que a posse de ar condicionado pelas famílias tenha mais que duplicado entre 2005 e 2017, despertando o interesse em estratégias que visem a redução do consumo de energia aliado à manutenção do conforto ambiental [47]. Neste contexto, a economia de energia faz-se cada vez mais necessária para a redução dos impactos ambientais e sociais associados à sua geração.

O desempenho energético de edificações é indicado como um elemento chave para o alcance do desenvolvimento sustentável, uma vez que pode potencialmente reduzir cerca de 20% nas emissões de gases de efeito estufa e 20% de economia de energia primária [74]. Em edifícios comerciais, devido à influência negativa que um ambiente desconfortável provoca na performance dos usuários, a busca pelo conforto térmico associado ao baixo consumo de energia tem resultado em um crescimento nas pesquisas neste campo [80]. O uso contínuo de equipamentos eletrônicos e a alta densidade de pessoas nos escritórios aumentam o desafio de manter estes ambientes termicamente confortáveis.

Para a obtenção do melhor desempenho de uma construção, três fatores devem ser considerados: projeto arquitetônico, sistemas de aquecimento e resfriamento, e ocupação. O projeto arquitetônico é desenvolvido de forma iterativa, com uma equipe que repassa todos os aspectos da edificação e repensa sobre decisões relacionadas a arquitetura [92]. Com um projeto bastante otimizado em mãos, especialistas da área da construção civil podem reduzir a capacidade de sistemas de refrigeração e aquecimento, e minimizar a necessidade deste conjunto de serviços.

Porém, apesar do projeto arquitetônico figurar como importante aspecto na performance da edificação, determinar materiais e configurações que otimizem o consumo e conforto na estrutura não é uma tarefa fácil. Considerações sobre a localização, estratégias de ventilação, iluminação e materiais a serem utilizados, aumentam a complexidade da concepção de um projeto energeticamente eficiente.

Como o ar condicionado pode ser um dos maiores consumidores de energia em

edificações [58], uma das principais maneiras de reduzir seu uso é a busca de soluções passivas, como o uso de ventilação natural, favorecendo a execução de projetos arquitetônicos que, por sua vez, devem levar em consideração diversos aspectos, desde variáveis técnico-construtivas até fatores climáticos em que o edifício de análise está inserido. A combinação e aplicação correta dessas variáveis resultará no conforto térmico adequado dentro do edifício [70].

Neste cenário, estudos que consideram simulações computacionais de modelos que tratam do consumo de edificações, estão se tornando cada vez mais presentes. Diversas abordagens são propostas para modelos computacionais de edificações, como:

- O uso de redes neurais e máquinas de vetor suporte para a predição do uso de energia elétrica em sistemas de aquecimento de residências [85];
- A geração de base de dados através da parametrização de residência com o objetivo de utilizar algoritmos de aprendizado de máquina para previsão de cargas de aquecimento<sup>1</sup> e resfriamento<sup>2</sup> [37];
- A criação de sistemas de controle em tempo real que utilizam algoritmos bi-objetivo para a aproximação da demanda elétrica do consumidor e minimização do gasto com eletrodomésticos [39];
- A aplicação de algoritmos genéticos para a minimização do consumo de energia e das horas de desconforto em uma residência tipicamente italiana simulada em diferentes zonas climáticas da Itália [5].

A necessidade de se planejar residências e edifícios energeticamente eficientes e confortáveis trouxe à tona o conceito de edificação de balanço energético nulo, conhecido como NZEB (do inglês, *Net-Zero Energy Building*). Este tipo de edificação se destaca por ter como principal objetivo mitigar as emissões de CO<sub>2</sub> e reduzir o consumo de energia. Sudbrack [73] identificou limitações para o balanço energético nulo em uma tipologia residencial pré-fabricada em Brasília. Levando em conta aspectos relacionados ao uso, ocupação e ventilação natural, a autora realizou alguns testes no modelo através do *software* proprietário DesignBuilder utilizando um conjunto de parâmetros predefinidos. A partir da realização dos testes, um conjunto de materiais foi definido como mais adequado ao clima, oferecendo o menor número de horas de desconforto na residência ao longo do ano.

Assim como Sudbrack [73], Harkouss et al. [30] realizaram simulações computacionais de diferentes cenários a partir da troca de materiais e das características geométricas

<sup>1</sup> As cargas de aquecimento se referem à quantidade de energia térmica que precisaria ser adicionada a um espaço para manter a temperatura em uma faixa aceitável.

<sup>2</sup> As cargas de resfriamento se referem à quantidade de energia térmica que precisaria ser removida de um espaço para manter a temperatura em uma faixa aceitável.

de uma NZEB, considerando, por exemplo, tipos diferentes de vidro, dimensões das janelas e espessura das paredes e do sistema de cobertura. O autor utilizou uma metodologia evolucionária multiobjetivo para minimizar as demandas térmicas e elétricas da residência, automatizando o processo de tomada de decisão dos materiais e das configurações arquitetônicas a serem utilizados no projeto.

## 1.2 Justificativa

A literatura apresenta uma diversidade de trabalhos que realizaram a modelagem e a simulação de diferentes cenários com várias tipologias arquitetônicas. Contudo, as alternativas são, com frequência, testadas uma a uma, de forma separada, e os resultados referem-se à comparações dentre saídas geradas. Esse processo demanda numerosos testes e considerável tempo de execução, tornando, muitas vezes, a análise de muitas variáveis de forma simultânea, inviável. Assim, a combinação de métodos de otimização e algoritmos inteligentes tem se mostrado promissora [49, 2]. A integração desses métodos pode trazer melhorias às predições relacionadas ao mercado de energia, auxiliando prestadoras de serviço a entender as demandas dos diferentes consumidores e usuários a economizar energia através do conhecimento de seus hábitos de uso. A predição do comportamento da edificação a partir de parâmetros de projeto pode vir a auxiliar na tomada de decisão de especialistas, de modo que não seja necessária uma análise manual e operacionalmente custosa.

O desenvolvimento de um método automatizado deste tipo representa uma contribuição significativa na modelagem energética de edificações, pois auxiliará especialistas da área da construção a projetarem construções mais eficientes energeticamente, facilitando a tomada de decisões quanto à adoção de parâmetros de projeto. O estudo reduzirá o trabalho manual de modelagem e a análise operacionalmente custosa.

## 1.3 Objetivos Gerais e Específicos

O presente trabalho tem como objetivo geral propor um método combinado de aprendizado de máquina e algoritmos evolucionários para a previsão de cargas térmicas na construção civil.

Os objetivos específicos que se destacam no trabalho são:

- Otimizar o conjunto de hiperparâmetros a fim de melhorar a previsão das cargas térmicas com o uso dos métodos Regressão *Elastic-Net*, Máquina de Aprendizado Extremo e *Extreme Gradient Boosting*;
- Validar e comparar os resultados obtidos com dados da literatura;

- Avaliar o custo computacional e a viabilidade da aplicação do modelo em diferentes escalas para a aproximação das cargas térmicas a partir de dados térmicos e relacionados à geometria da estrutura.

## 2 Revisão da Literatura

A integração entre as áreas de sistemas de informação, ciências sociais e ciência energética tem proporcionado um crescimento no número de pesquisas que possuem como objetivo a conservação eficiente de energia [4]. Por conta desta aproximação, dados referentes a produção e consumo de energia têm sido armazenados e utilizados em sistemas de energia inteligentes, visando o aumento da eficiência dos sistemas de geração e distribuição, e buscando reduções no consumo [94]. Watson et al. [86] relataram que a eficiência energética possui como objetivo o aumento da eficiência dos sistemas de fornecimento e otimização das redes de distribuição de energia através de coleta, análise e implementação de soluções computacionais, integrando sistemas de energia e informática.

Zhou and Yang [94] propõem a descrição do comportamento do consumo de energia com o uso de três dimensões: dimensão espacial, do tempo e do usuário. A Figura 1 apresenta as variáveis relacionadas a cada uma das dimensões, que determinam a granularidade dos dados que podem vir a serem coletados. Além disso Zhou and Yang [94], dividem o campo de pesquisas relacionadas ao consumo de energia em duas categorias, relacionadas ao paradigma econômico e ao orientado ao comportamento. O paradigma econômico sugere que as pessoas tendem a fazer decisões relacionadas ao seu consumo baseadas nos custos e benefícios, de modo que estratégias baseadas no preço e em incentivos podem ser caminhos efetivos para promover mudanças positivas no custo da energia. O paradigma orientado ao comportamento indica que os consumidores tendem a tomar determinadas decisões a partir de complexos fatores interpessoais, de modo que a sensação de obrigatoriedade torna a redução do consumo de energia mais provável.

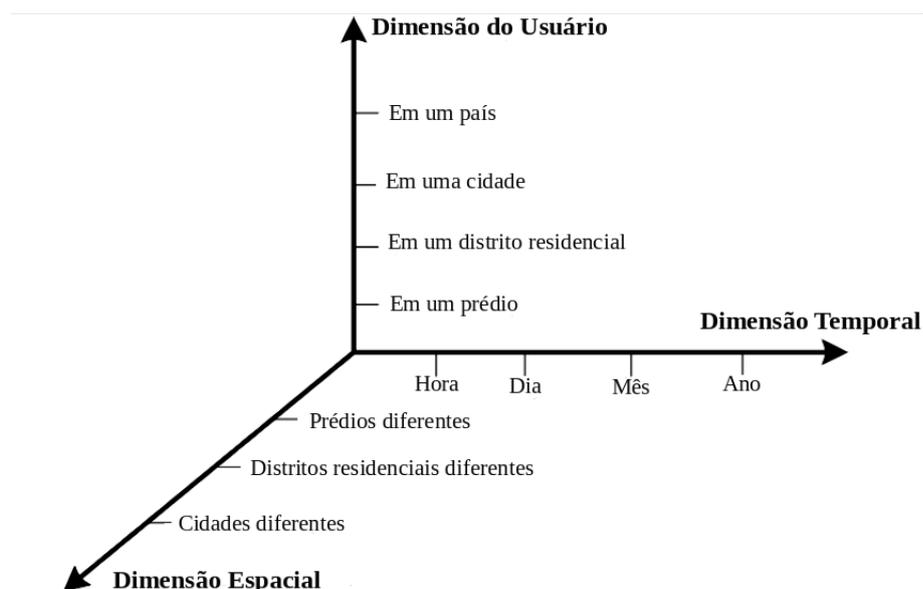


Figura 1 – Dimensões propostas para o consumo de energia.

Fonte: Adaptada de Zhou and Yang [94].

Métodos de aprendizado de máquina têm sido utilizados para realizar previsões em diversas áreas do conhecimento [23, 65, 89]. Como exemplo do que vem sendo desenvolvido em nosso grupo de pesquisa podemos citar o trabalho de Saporetti et al. [65], nesse trabalho foram utilizados seis métodos de aprendizado de máquina combinados com técnicas de balanceamento de dados para classificar dados geológicos da Bacia Provença do Sul. Com o objetivo de verificar a melhor abordagem para a classificação de dados petrográficos, utilizaram as seguintes técnicas: K-vizinhos mais próximos (KNN, do inglês *K-Nearest Neighbors*), Máquinas de Vetor Suporte (SVM, do inglês *Support Vector Machines*), Árvores de Decisão (DT, do inglês *Decision Trees*), Floresta Aleatória (RF, do inglês *Random Forest*) e Perceptron Multicamadas (MLP, do inglês *Multilayer Perceptron*).

O algoritmo KNN [88] realiza a classificação de uma amostra baseada nas  $K$  amostras mais próximas que possuem maior similaridade, enquanto o SVM [69] combina linearmente uma série de atributos de funções *kernel*<sup>1</sup> para desempenhar a classificação de amostras. As Árvores de Decisão [31] constroem modelos de classificação semelhantes à estrutura de uma árvore, dividindo o conjunto de dados em subconjuntos gradativamente menores enquanto a árvore associada cresce de forma incremental, resultando em nós de decisão e folhas.

O método RF [6] executa a classificação dos dados a partir da construção de  $K$  árvores de decisão em  $K$  iterações, sendo  $K$  um parâmetro livre que pode ser determinado a partir da observação dos erros dos subconjuntos de treinamento. A rede neuronal<sup>2</sup> do tipo Perceptron Multicamadas [54] possui arquitetura composta de ao menos três camadas de neurônios<sup>3</sup>, sendo essas chamadas de camadas de entrada, escondida e de saída. O método utiliza o número de camadas e neurônios escondidos, um algoritmo para a otimização dos pesos, a conectividade das camadas e a renormalização dos dados como parâmetros, e possui como alvo a minimização dos erros.

A aplicação da técnica de *oversampling*<sup>4</sup> para o balanceamento das amostras melhorou a performance geral dos classificadores testados por Saporetti et al. [65]. Dentre os seis modelos testados, DT, SVM e RF apresentaram melhores resultados, que demonstraram a eficiência da abordagem para a classificação de petrofácies.

Os conjuntos de dados de entrada podem conter uma volumosa quantidade de características, que não necessariamente possuem grande influência nos resultados das

<sup>1</sup> *Kernel* é um termo normalmente utilizado na área de aprendizado de máquina para se referir a um método que utiliza um modelo linear para resolver um problema não-linear.

<sup>2</sup> Uma rede neuronal pode ser interpretada como uma série de algoritmos que buscam relacionamentos em um conjunto de dados através de um processo que tenta reproduzir o modo de operação do cérebro humano.

<sup>3</sup> Nesse caso, um neurônio é definido como uma operação matemática que recebe sua entrada, multiplica por pesos e passa o resultado através da função de ativação para outros neurônios.

<sup>4</sup> *Oversampling* é uma técnica utilizada na análise de dados para ajustar a distribuição de classes em um conjunto de dados.

predições obtidas pelas redes neurais, tornando sua execução computacionalmente custosa. Para isso, técnicas de seleção de características como Análise de Componentes Principais (PCA, do inglês *Principal Component Analysis*) e de regressão penalizada como LASSO (do inglês *Least Absolute Shrinkage and Selection Operator*), podem auxiliar na seleção de variáveis em modelos de regressão [93, 95].

O algoritmo de Evolução Diferencial (DE, do inglês *Differential Evolution*) é um dos algoritmos mais eficientes do tipo genético para resolver problemas com variáveis no domínio dos reais. O DE é uma ferramenta de otimização de grande utilidade, acessível para aplicações práticas e tem sido utilizada para descobrir soluções eficazes para problemas quase intratáveis, sem a necessidade de conhecimento especializado ou algoritmos de análises complexas. O método DE tem sido aplicado em problemas mono e multiobjetivo, com o intuito de buscar soluções mais eficientes para problemas com ou sem restrições de projeto [91].

Wang et al. [84] utilizaram uma estratégia híbrida que combina o algoritmo DE e Máquinas de Vetores Suporte para prever o consumo elétrico industrial anual. O DE foi utilizado para escolher um conjunto apropriado de parâmetros para o modelo de Vetores Suporte aplicado ao problema de previsão de demanda elétrica, pois a determinação do conjunto de hiperparâmetros<sup>5</sup> que melhora o desempenho do modelo é um elemento chave para a efetividade da previsão.

Yang et al. [91] aplicaram algoritmos de Evolução Diferencial Discreta e Contínua para a seleção de dados de entrada e hiperparâmetros, respectivamente. O processo de otimização visou melhorar o desempenho de um modelo de Máquina de Aprendizado Extremo (ELM, do inglês *Extreme Learning Machine*) aplicado à análise de desgaste de componentes industriais a fim de prever o limite de desgaste para a realização da substituição e assim prevenir possíveis danos a equipamentos. Além destas aplicações, o método DE tem sido aplicado a previsão de custos e a confiabilidade de sistemas híbridos de distribuição de eletricidade [63]. Além disso, melhorias tem sido propostas com o objetivo de melhorar a velocidade e acurácia do método DE [13, 3, 66].

O *Extreme Gradient Boosting* (XGB) é um algoritmo de aprendizado de máquina, baseado em árvore de decisão [9]. Desde a sua introdução por Chen et al. [9], este algoritmo tem sido o mais vitorioso na plataforma Kaggle [75], uma plataforma de competições em que cientistas de dados resolvem desafios por meio de aprendizado de máquina. Além disso, também tem sido utilizado em várias aplicações de ponta na indústria [41, 28, 68]. O algoritmo se diferencia de outros por conta de ser aderente a uma ampla variedade de aplicações e possuir bibliotecas implementadas para diversas linguagens de programação.

<sup>5</sup> Hiperparâmetros são variáveis que controlam o processo de treinamento de algoritmos inteligentes. Essas variáveis não estão diretamente relacionadas aos dados de treinamento, sendo variáveis de configuração.

A implementação de estratégias que realizem previsões e caracterizem o gasto de energia de edificações podem vir a desempenhar um importante papel na gestão da energia e na criação de iniciativas eficientes para o controle de emissões [29]. Marcu et al. [45] utilizaram sensores sem fio para a aquisição de dados de potência, voltagem e corrente elétrica consumida com o objetivo de monitorar e analisar o consumo.

O sistema de Marcu et al. [45] foi configurado para o uso em residências, *data centers*<sup>6</sup> e edifícios com escritórios, também contando com uma interface gráfica que permite a geração de relatórios e gráficos com diferentes perspectivas sobre o consumo de energia em diferentes dispositivos. A arquitetura de monitoramento utiliza assinaturas elétricas dos dispositivos, que a partir dos testes preliminares demonstraram serem parâmetros adequados para a administração do consumo dos aparelhos. Dados deste tipo têm sido utilizados principalmente no setor comercial, pois a disponibilidade de informações adquiridas a partir de sensores para residências familiares ainda é limitada.

Jain et al. [36] citam a necessidade de pesquisas que foquem em residências que acomodem múltiplas famílias, pois, dentro do campo limitado de pesquisas que tratam de residências, a maior parte dos estudos utilizam dados de lares que acomodem uma única família. A partir do uso de dados de sensores, Jain et al. [36] monitoraram e posteriormente analisaram a eficácia de um algoritmo na previsão do consumo em unidades residenciais. O algoritmo foi responsável pela busca de padrões de consumo e relações entre características específicas e níveis de ocupação nas diferentes unidades residenciais para a aproximação de um método preditivo mais abrangente e eficaz. O modelo desenvolvido para a análise dos dados de sensores foi baseado em SVM [67]. A validação e teste ocorreram com o uso de um conjunto de dados empíricos, sendo desempenhada uma avaliação do impacto da granularidade dos dados nas escalas temporal e espacial.

A partir dos resultados obtidos, Jain et al. [36] conseguiram concluir que a granularidade espacial possui grande impacto no poder preditivo do modelo, apesar dos resultados também disporem de influência da granularidade temporal. Contudo, a utilização de uma única residência multi-familiar é apresentada como uma limitação, sendo necessárias mais pesquisas neste campo e aplicação do modelo em outros conjuntos de dados. Além disso, Jain et al. [36] propõem o uso de outras técnicas de aprendizado de máquina e de seleção de características.

## 2.1 Aprendizado de Máquina para a Previsão de Cargas energéticas

Tsanas and Xifara [81] desenvolveram um framework baseado em métodos estatísticos para estudar o efeito de oito variáveis de entrada na previsão de duas variáveis de saída.

---

<sup>6</sup> Um *data center* é uma instalação que centraliza as operações e equipamentos de Tecnologia da Informação de uma organização, bem como onde se armazena, gerencia e dissemina seus dados.

A partir do uso de métodos estatísticos clássicos e não paramétricos, analisou o efeito de cada uma das variáveis de entrada nas cargas de aquecimento e resfriamento previstas. Simulações de cenários através do software proprietário Ecotect permitiram comparar o desempenho preditivo do método de aprendizado de máquina Florestas Aleatórias e concluir que modelos estatísticos podem ser utilizados para aproximar cargas térmicas a partir de parâmetros de projeto.

Chou and Bui [11] e Duarte et al. [15] utilizaram a mesma base de dados estudada por [81] para aplicar diferentes métodos de aprendizado de máquina. Chou and Bui [11] aplicou Regressão por Vetores Suporte, Redes Neurais Artificiais, Árvore de Regressão, Regressão Linear e um modelo *ensemble*<sup>7</sup> de inferência a fim de determinar a melhor abordagem ser aplicada para a previsão das cargas de aquecimento e resfriamento. A comparação dos resultados permitiu concluir que o modelo *ensemble*, que combinava Vetores Suporte e Redes Neurais Artificiais, obteve o melhor desempenho em relação aos dados presentes na literatura. Duarte et al. [15] comparou o desempenho dos métodos Árvore de Decisão, Máquinas de Vetores Suporte, Florestas Aleatórias e Rede Perceptron Multicamadas nas 768 amostras testadas por Tsanas and Xifara [81]. Além disso, utilizou busca exaustiva para determinar o conjunto de parâmetros que aprimorem o desempenho dos métodos preditivos e técnica de validação cruzada para prevenir o sobreajuste no conjunto de dados.

Jihad and Tahiri [37] aplicaram Redes Neurais Artificiais para realizar a previsão da energia necessária em edifícios residenciais localizados na zona climática de Agadir, no Marrocos. Através de parametrização de um modelo base, foram geradas 5625 amostras de acordo com o modo de construção e uso de edificações residenciais da cidade estudada. Para testar o desempenho da rede, três edificações foram selecionadas e simuladas através do software proprietário Design Builder para calcular a demanda energética, alcançando resultados satisfatórios com acurácia de 98.7% e 97.6% para os conjuntos de treino e teste, respectivamente.

## 2.2 Emprego de Redes Neurais Profundas em Aprendizado Supervisionado

A melhora da infraestrutura elétrica e a necessidade uso eficiente de fontes de energia renováveis tem originado um novo paradigma para as distribuidoras de energia, levando ao estudo e criação de *smart grids*. Smart grids são sistemas de distribuição e transmissão de energia elétrica que utilizam recursos digitais e da Tecnologia da Informação. Por meio delas, o sistema opera de forma mais eficiente, com maior controle do fluxo de energia e, consequentemente, passa a ser mais sustentável [48]. Previsões realizadas de forma precisa

<sup>7</sup> Modelos *ensemble* na área de aprendizado de máquina combinam as decisões de múltiplos modelos simplificados com o objetivo de melhorar a performance geral e criar um método mais robusto.

no nível do usuário refletem diretamente em melhorias na eficiência do sistema como um todo. Porém, a previsão do consumo elétrico é uma tarefa complexa devido aos diversos fatores, como condições climáticas, ocupação e performance dos sistemas térmicos. Diante disto, Mocanu et al. [48] propõem o uso de modelos estocásticos para a previsões em séries temporais do consumo de energia. Os métodos de aprendizado profundo Máquina Boltzmann Restrita Condicional e Máquina Boltzmann Restrita Condicional Fatorada foram aplicados a fim de diminuir as incertezas nas predições causadas por flutuações nas variáveis de interesse. Os resultados mostraram que os métodos empregados melhoraram de forma significativa as previsões obtidas por métodos tradicionais de aprendizado de máquina.

### 2.3 Algoritmos Evolutivos aplicados à Eficiência Energética

Cheng and Cao [10] propôs a aplicação de *splines* de regressão adaptativa multivariada evolutiva (EMARS, do inglês *Evolutionary Multivariate Adaptive Regression Splines*) e modelos de inteligência artificial para melhorar as previsões realizadas no conjunto de dados proposto por Tsanas and Xifara [81]. O modelo EMARS combina *splines* de regressão adaptativa multivariada e um algoritmo evolutivo inspirado em colônia de abelhas para minimizar o erro de predição através da otimização do conjunto de parâmetros de configuração do modelo. Os testes foram realizados nos dados estudados por Tsanas and Xifara [81] e resultaram em uma melhora de 65% e 45% para as cargas de resfriamento e aquecimento, respectivamente.

O consumo de energia tem sido evidenciado como uma importante questão política em termos de economia e sustentabilidade. Castelli et al. [7] citam que eficiência energética de edificações é considerada uma prioridade nas políticas energéticas de diversos países e propõem o uso de um *framework* baseado em programação genética para estimar a performance energética de residências. A predição acurada das cargas de aquecimento e resfriamento de edificações permitem um maior controle do consumo de energia e auxiliam na escolha de fontes mais adequadas de acordo com a necessidade energética da população. O *framework* de programação genética combinado com um método de busca local resultou em um sistema com previsões acuradas para as variáveis de saída, competitivas com a literatura existente.

Neto and Hippert [52] utilizaram a meta-heurística de otimização por enxame de partículas para buscar os melhores parâmetros de um modelo estatístico que minimizem o erro médio quadrático das previsões. A partir de uma base de dados de cargas elétricas da cidade do Rio de Janeiro, Neto and Hippert [52] conduziram simulações computacionais a fim de verificar o desempenho do método. O erro médio obtido encontrado e a variabilidade dos resultados foi baixa, levando os autores a concluírem que o método de enxame de partículas alcançou melhores resultados em relação a métodos mais simples testados,

apesar da presença de alguns erros discrepantes em algumas execuções.

## 2.4 Regressores *Ensemble*

Robinson et al. [64] utilizaram o método *ensemble Gradient Boosting* (GB) para estimar o consumo de energia de uma edificação comercial a partir de um conjunto reduzido de variáveis de projeto. O modelo de regressão realizou previsões que se aproximaram por um fator de 2 dos valores observados para o consumo de energia e com um coeficiente de determinação de 0.82. O modelo foi testado e validado utilizando um conjunto de dados de consumo energético da cidade de Nova Iorque e aplicado a dados da cidade de Atlanta, Geórgia, para estimativas do consumo energético.

Goliatt et al. [24] avaliaram a performance do método Processo Gaussiano, conhecido como *Kriging* para estimar cargas de aquecimento e resfriamento nos dados disponibilizados por Tsanas and Xifara [81]. Os parâmetros do método foram determinados a partir de busca exaustiva e validação cruzada. Com o uso de quatro métricas estatísticas, o método testado superou os resultados obtidos por outras técnicas de aprendizado de máquina, como Redes Neurais Artificiais, Máquinas de Vetores Suporte e Florestas Aleatórias previamente apresentadas na literatura. A proposta apresentada pode vir a auxiliar na redução de custos e melhora do desempenho energético ainda na fase de projeto.

Goliatt et al. [25] citam que o interesse pelos modelos conhecidos como *black box* tem crescido ao longo dos anos. Modelos *black box* são sistemas em que um objeto de estudo pode ser analisado em termos das entradas e saídas, sem o conhecimento aprofundado das operações internas. Máquinas de *Gradient Boosting* tem sido utilizadas com sucesso para realizar tarefas de previsão, de forma que Goliatt et al. [25] propõem o uso deste modelo para realizar previsões das cargas de térmicas em edificações ainda na fase de projeto. Assim como Goliatt et al. [25], Al-Rakhami et al. [1] propõem o uso de um modelo *ensemble* para realizar a previsão de cargas de aquecimento e resfriamento em edificações residenciais. A abordagem proposta consiste na aplicação do algoritmo XGB para melhorar as previsões realizadas por métodos de aprendizado de máquina tradicionais da literatura, que sofrem de problemas de sobreajuste e levam a previsões imprecisas.

## 2.5 Otimização de Parâmetros em Métodos de Aprendizado de Máquina

Edificações mais modernas e inteligente tem surgido ao longo dos anos, onde as temperaturas internas e externas podem se obtidas e posteriormente previstas através da instalação de sensores. Modelos preditivos baseados em dados podem realizar previsões de temperaturas de forma precisa e economizar energia através da otimização do uso dos sistemas de aquecimento, resfriamento e ar condicionado [44]. Kumar et al. [44] realizaram

um experimento em uma residência com diversos sensores instalados, onde um modelo híbrido baseado em otimização por enxame de partículas e ELM foi utilizado para prever a temperatura operacional da estrutura e otimizar o uso de energia futuro. O experimento analisou o efeito da combinação dos diversos sensores para a previsão da temperatura, determinando a relevância de cada um deles com a temperatura prevista e a observada através de métodos convencionais de medição.

### 3 Material e Métodos

#### 3.1 Bases de Dados

A fim de analisar a performance dos métodos de aprendizado de máquina testados, é proposta a utilização de duas bases de dados disponíveis na literatura. A base de dados 1 (BD1) foi utilizada por Tsanas and Xifara [81] e a base de dados 2 (BD2) por Jihad and Tahiri [37]. Detalhes sobre os conjuntos de dados são descritos a seguir.

##### 3.1.1 Base de Dados 1

Os dados utilizado neste trabalho foram disponibilizados por Tsanas and Xifara [81] e obtidos a partir da simulação de uma série de edificações utilizando o *software* proprietário Ecotect. O Ecotect é uma ferramenta para análise ambiental, compatível com informações provenientes de *softwares* de modelagem, como Autodesk Revit Architecture, e utilizado para executar análises preliminares compreensivas relacionadas a performance energética de edificações. O *software* inclui uma ampla variedade de funções de análise com uma interface iterativa e visual que faz com que resultados analíticos estejam diretamente ligados ao contexto do modelo da edificação [90]. O *software* Ecotect dispõe de análises térmica, acústica e de iluminação, e foi utilizado para simular as cargas de aquecimento e resfriamento durante o estágio operacional da estrutura a partir da atribuição da geometria, das propriedades dos materiais e das condições climáticas locais para garantir a precisão dos resultados [57].

O conjunto de dados é composto por oito variáveis de entradas ( $X_1, \dots, X_8$ ) e duas variáveis de saída ( $Y_1$  e  $Y_2$ ), apresentadas na Tabela 1. Um sistema de geometria modular foi gerado a partir de um cubo base de  $3,5 \times 3,5 \times 3,5$  m. Para a geração das diferentes geometrias de edificações, foram utilizados dezoito desses elementos cúbicos (Figura 2) e um subconjunto de doze geometrias com valores de Compacidade Relativa ( $CR$ ) distintos foram selecionadas para as simulações, como pode ser visto na Figura 3.

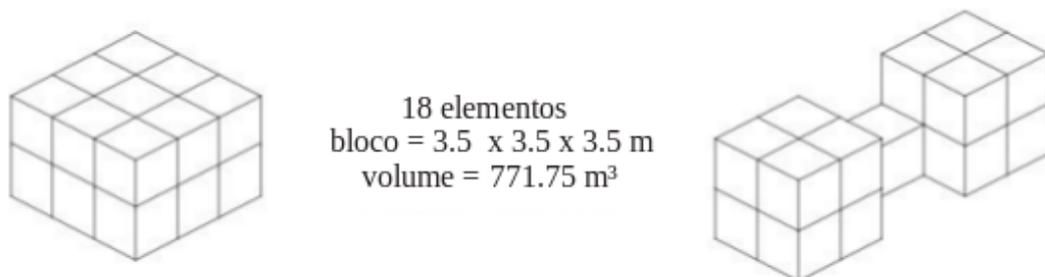


Figura 2 – Geração das geometrias a partir dos 18 blocos.

Fonte: Adaptado de Duarte et al. [15].

Tabela 1 – Representação matemática, mínimos, máximos e total de valores possíveis para as variáveis de entrada e saída.

Representação Matemática	Variável de Entrada ou Saída	Mínimo	Máximo	Número de Valores Possíveis
$X_1$	Compacidade Relativa (-)	0,62	0,98	12
$X_2$	Área de Superfície ( $m^2$ )	514,5	808,5	12
$X_3$	Área de Parede ( $m^2$ )	245	416,5	7
$X_4$	Área de Telhado ( $m^2$ )	110,25	220,5	4
$X_5$	Altura (m)	3,5	7	2
$X_6$	Orientação ( $^\circ$ )	2	5	4
$X_7$	Área Envidraçada ( $m^2$ )	0	0,4	4
$X_8$	Distribuição de Envidraçamento (-)	0	5	6
$Y_1$	Carga de Aquecimento ( $KWh/m^2/an$ )	6,01	43,1	586
$Y_2$	Carga de Resfriamento ( $KWh/m^2/an$ )	10,9	48,03	636

Fonte: Adaptada de Duarte et al. [15].

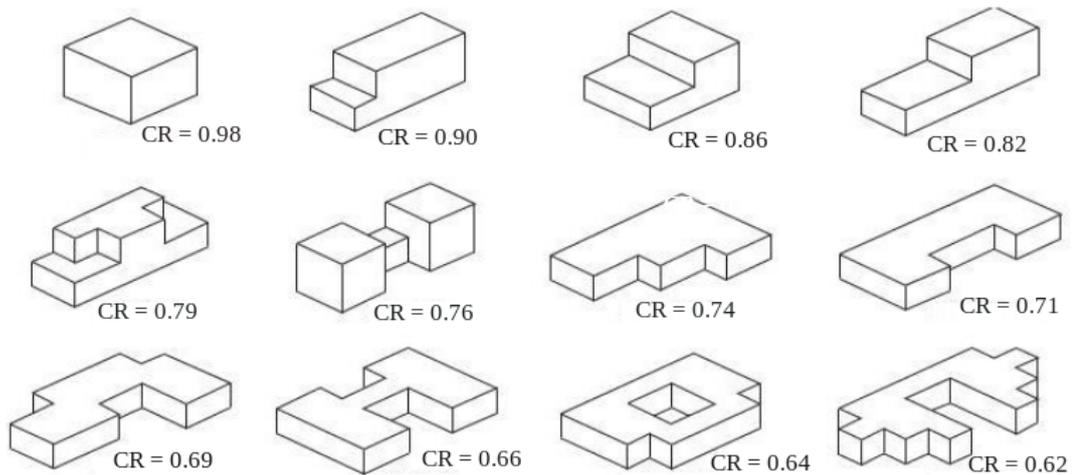


Figura 3 – Exemplos das geometrias das edificações.

Fonte: Adaptado de Duarte et al. [15].

O indicador de Compacidade Relativa é utilizado para explicitar diferentes tipos de construções e pode ser calculado a partir da Equação 3.1 [15].

$$CR = 6V^{2/3}(A^{-1}) \quad (3.1)$$

onde  $A$  representa a área de superfície e  $V$  o volume da construção. A área de superfície foi calculada como a soma da área total das paredes, área de telhado e piso. A Figura 4 mostra os detalhes da área das paredes, do telhado, do piso e a altura da edificação. Foram utilizadas quatro orientações principais nos experimentos: norte, leste, oeste, e sul. As porcentagens de área de vidro em relação a área de piso foram 10%, 25% e 40%. Além disso, cinco distribuições de envidraçamento foram simuladas:

- uniforme: com 25% de envidraçamento em cada face;

- norte: 55% para a face norte e 15% para as demais faces;
- leste: 55% para a face leste e 15% para as demais faces;
- sul: 55% para a face sul e 15% para as demais faces;
- oeste: 55% para a face oeste e 15% para as demais faces;

Adicionalmente, áreas sem envidraçamento foram simuladas no experimento e todas os modelos foram rotacionados nas quatro direções cardeais. A partir dessas informações, o conjunto de dados possui  $12 \times 3 \times 5 \times 4 + 12 \times 4 = 768$  amostras de edificações.

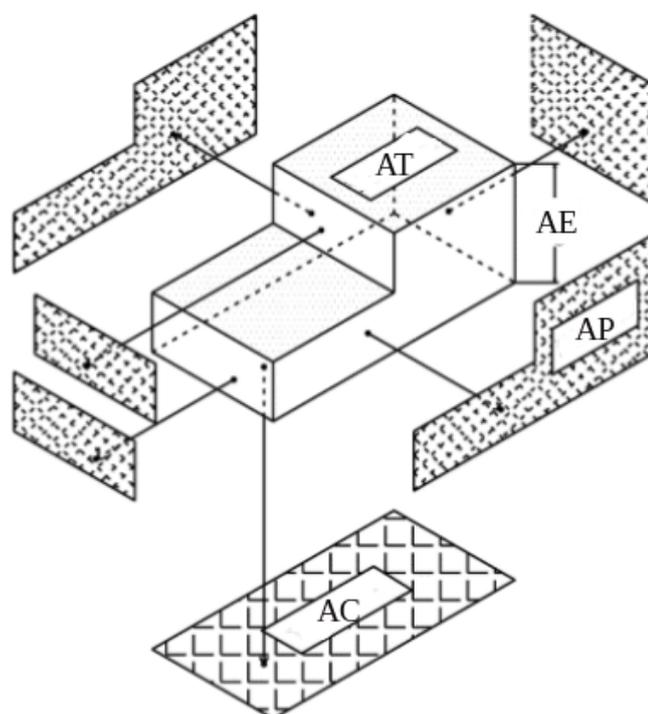


Figura 4 – Definição das áreas da edificação, onde AE representa a altura total, AC a área de piso, AT a área de telhado e AP a área de parede.

Fonte: Adaptado de Duarte et al. [15].

A simulação considerou que todos os edifícios estavam localizados em Atenas na Grécia, e que cada bloco era ocupado por sete pessoas, totalizando uma média de consumo de 70 W. As configurações internas dos blocos foram definidas como: revestimento: 0,6; umidade: 60%; velocidade do ar: 0,3 m/s; nível de iluminação: 300 lux (equivalente a cinco lâmpadas de LED de 9 W). Os ganhos de calor sensível e latente foram assumidos como 5 W/m<sup>2</sup> e 2 W/m<sup>2</sup>, respectivamente. Para as propriedades térmicas, um modo misto com 95% de eficiência foi utilizado, com uma faixa de temperatura de 19°C a 24°C, com 15-20h de operação durante a semana e 10-20h nos finais de semana. Detalhes adicionais dos experimentos de simulação são fornecidos por Tsanas and Xifara [81].

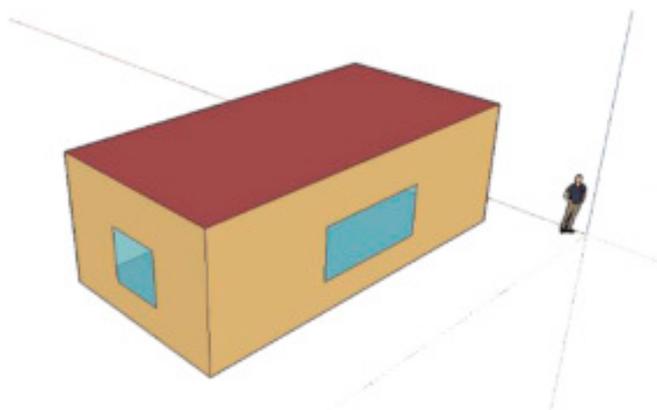


Figura 5 – Modelo base utilizado para a geração da base de dados de edificações residenciais.

Fonte: Jihad and Tahiri [37].

### 3.1.2 Base de Dados 2

A segunda base de dados a utilizada neste trabalho foi estudada por Jihad and Tahiri [37]. Através da criação de uma edificação base de dimensões  $10 \times 5 \times 3,5\text{m}$  (Figura 5) com o uso do software proprietário Design Builder, [37] utilizaram uma análise paramétrica para gerar diferentes cenários da edificação por meio da modificação de variáveis de projeto. A geração da base de dados com os diversos cenários ocorreu através da modificação do arquivo base de extensão .idf que contém dados de entrada para a simulação termodinâmica no software Design Builder. A partir das simulações, valores das cargas de aquecimento e resfriamento da estrutura foram extraídos. As edificações simuladas foram modeladas com o uso de materiais na indústria de construção do Marrocos e coeficientes de transferência de calor determinados pela regulação térmica do Marrocos para a zona climática de Agadir.

Foram simulados cenários com nove taxas de envidraçamento, expressadas como uma porcentagem da área total de paredes, e sendo estas: 5%, 10%, 15%, 20%, 25%, 30%, 35%, 40% e 45%. Para cada proporção de área envidraçada, foram simuladas cinco orientações para a residência em relação ao norte, sendo estas: 0, 45, 90, 135 e 180°. Cinco diferentes alturas foram atribuídas através da parametrização: 3,5; 7, 10,5; 14 e 17,5m. Além disso, o comprimento e largura da edificação foram variados criando 25 cenários para estas características. Dessa forma, a partir da parametrização foi obtida uma base de dados com 5625 amostras. As amostras simuladas são caracterizadas por meio de seis parâmetros construtivos ( $X_1, X_2, \dots, X_6$ ), que serão utilizados como variáveis de entrada para os métodos de aprendizado de máquina e possuem duas variáveis de saída, que representam as cargas de resfriamento ( $Y_1$ ) e aquecimento ( $Y_2$ ). A Tabela 2 apresenta as representações matemáticas das variáveis, as interpretações das variáveis de entrada e saída e o número de possíveis valores em cada uma delas.

Tabela 2 – Representações matemáticas das variáveis de entrada e saída e o número de possíveis valores.

Representação Matemática	Variável	Número de Valores Possíveis
$X_1$	Altura (m)	5
$X_2$	Compacidade Relativa (-)	125
$X_3$	Área de Parede (m <sup>2</sup> )	1125
$X_4$	Área de Superfície (m <sup>2</sup> )	25
$X_5$	Orientação(°)	5
$X_6$	Proporção de Envidraçamento (%)	9
$Y_1$	Carga de Resfriamento (KWh/m <sup>2</sup> /an)	5625
$Y_2$	Carga de Aquecimento (KWh/m <sup>2</sup> /an)	5625

Fonte: Adaptada de Jihad and Tahiri [37].

## 3.2 Métodos de Regressão

### 3.2.1 Regressão *Elastic Net*

A Regressão Ridge [32] é um tipo de modelo adequado na presença de muitos preditores, com coeficientes diferentes, de modo que impede que os coeficientes de modelos de regressão linear com variáveis correlacionadas sejam mal determinados e resultem em alta variação [20]. Este tipo de regressão soluciona o problema:

$$\mathbf{y} = \mathbf{X}\beta + e \quad (3.2)$$

onde  $\mathbf{y}$  é o vetor de observações,  $\mathbf{X}$  é a matriz de entradas,  $\beta$  é o vetor de coeficientes de regressão e  $e$  o vetor de erros residuais. Utilizando a penalização do tipo  $L2$ :

$$\hat{\beta}(\text{ridge}) = \arg \min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda \|\beta\|_2^2 \quad (3.3)$$

onde

$$\|\mathbf{y} - \mathbf{X}\beta\|_2^2 = \sum_{i=1}^n (y_i - \mathbf{x}_i^T \beta)^2 \quad (3.4)$$

é a norma- $L2$  da função de perda (por exemplo, soma dos quadrados dos resíduos) [17],  $\mathbf{x}_i^T$  a  $i$ -ésima linha de  $\mathbf{X}$ ,

$$\|\beta\|_2^2 = \sum_{j=1}^p \beta_j^2 \quad (3.5)$$

a penalização da norma- $L2$  em  $\beta$  e  $\lambda \geq 0$  o parâmetro de ajuste que regula a intensidade da penalização através da importância relativa do erro empírico e do termo de penalização. Quanto maior o valor de  $\lambda$ , maior a redução. Como o valor de  $\lambda$  depende dos dados, este pode ser determinado através de métodos orientados aos dados, como validação cruzada.

Os métodos de regressão LASSO são amplamente utilizados em domínios com conjuntos de dados massivos, onde algoritmos eficientes e rápidos são essenciais [20]. No entanto, esta técnica não é robusta para altas correlações entre os preditores, resultando

em erro quando esses forem idênticos. A penalização LASSO espera que muitos coeficientes sejam próximos de zero e apenas um pequeno subconjunto seja maior. O estimador [79] utiliza o critério de mínimos quadrados penalizados  $L1$  para obter uma solução esparsa para o seguinte problema de otimização:

$$\hat{\beta}(lasso) = \arg \min_{\beta} \|y - X\beta\|_2^2 + \lambda \|\beta\|_1 \quad (3.6)$$

onde

$$\|\beta\|_1 = \sum_{j=1}^p |\beta_j| \quad (3.7)$$

é a norma- $L1$ , que introduz dispersão na solução e  $\lambda \geq 0$  o parâmetro de ajuste que regula a intensidade de penalização. A penalização  $L1$  permite ao método regularizar o ajuste de mínimos quadrados e diminuir alguns componentes de  $\hat{\beta}(lasso)$  a zero a partir da escolha de valores de  $\lambda$  adequados. O algoritmo de descida de coordenadas cíclicas [20] calcula com eficiência os caminhos da solução para  $\lambda$  e é mais rápido que o algoritmo *Least Angle Regression* [16].

A técnica *Elastic Net* (NET) é uma extensão do método LASSO, robusta a correlações entre os preditores [20]. O *Elastic Net* utiliza uma mistura das penalidades  $L1$  (LASSO) e  $L2$  (Ridge) e pode ser formulado como:

$$\hat{\beta}(enet) = \left(1 + \frac{\lambda_2}{n}\right) \left\{ \arg \min_{\beta} \|y - \mathbf{X}\beta\|_2^2 + \lambda_2 \|\beta\|_2^2 + \lambda_1 \|\beta\|_1 \right\} \quad (3.8)$$

Setando  $\alpha = \lambda_2 / (\lambda_1 + \lambda_2)$ , o estimador se equivale a minimização de:

$$\hat{\beta}(enet2) = \arg \min_{\beta} \|y - \mathbf{X}\beta\|_2^2, \text{ subject to } P_{\alpha}(\beta) = (1 - \alpha) \|\beta\|_1 + \alpha \|\beta\|_2^2 \leq s \text{ for some } s \quad (3.9)$$

onde  $P_{\alpha}(\beta)$  é a penalização *Elastic Net* [96]. O método é simplificado a uma regressão Ridge quando  $\alpha = 1$  e a uma regressão LASSO quando  $\alpha = 0$ . A parte  $L1$  do método NET faz a seleção automática de variáveis, enquanto a parte  $L2$  incentiva a seleção agrupada e estabiliza os caminhos da solução em relação à amostragem aleatória, melhorando assim a previsão. Ao induzir um efeito de agrupamento durante a seleção de variáveis, o método pode selecionar grupos de características correlacionadas quando os grupos não são conhecidos antecipadamente.

### 3.2.2 Máquina de Aprendizado Extremo

Os modelos ELM podem ser definidos como um tipo de rede neuronal sem realimentação que possui uma única camada oculta [35]. Redes do tipo ELM buscam balancear velocidade e desempenho na generalização, apresentando vantagens em relação a métodos tradicionais como Redes Neurais Artificiais (ANN, do inglês *Artificial Neural Network*) e SVM por conta do treinamento ser linear nos parâmetros ajustáveis [76]. Nas redes ELM

existem três tipos de aleatoriedade: (i) totalmente conectada, os parâmetros da camada oculta são randomicamente gerados; (ii) as conexões podem ser randomicamente geradas, nem todos os nós de entrada são conectados a um nó oculto em particular; (iii) um nó oculto pode ser uma sub-rede formada por vários nós, resultando em aprendizados de características locais. A função de saída utilizada neste trabalho pode ser descrita pela Equação 3.10.

$$\begin{aligned}\hat{y}(\mathbf{x}) &= \sum_{i=1}^L \beta_i G(\alpha, \gamma, \mathbf{w}_i, b_i, \mathbf{c}, \mathbf{x}) \\ &= \sum_{i=1}^L \beta_i G(\alpha \text{MLP}(\mathbf{w}_i, b_i, \mathbf{x}) + \gamma(1 - \alpha) \text{RBF}(\mathbf{x}, \mathbf{c}))\end{aligned}\quad (3.10)$$

onde  $\hat{y}$  é a previsão retornada pela ELM associada ao vetor de entrada  $\mathbf{x}$ ,  $\mathbf{w}_i$  é o vetor de pesos do  $i$ -ésimo nó oculto,  $b_i$  são os desvios dos neurônios da camada oculta,  $\beta_i$  são os pesos de saída e  $\mathbf{c}$  é o vetor de centros da função de base radial. MLP (Perceptron multicamadas, do inglês *Multilayer Perceptron*) e RBF (Função de Base Radial, do inglês *Radial Basis Function*) são funções de ativação de entrada, enquanto  $\alpha$  e  $\gamma$  são parâmetros definidos pelo usuário que multiplica MLP( $\cdot$ ) and RBF( $\cdot$ ).  $G(\cdot)$  é a função de ativação não-linear de saída e  $L$  é o número de neurônios na camada oculta. As funções de ativação de saída  $G(\alpha, \mathbf{w}_i, b_i, \mathbf{c}, \mathbf{x})$  são apresentadas na Tabela 3.2.2.

Tabela 3 – Funções de ativação para ELM.

#	Nome	Função de Ativação $G$
1	<i>Identity</i>	$G(x) = x$
2	ReLU	$G(x) = \max(0, x_i; i = 1, \dots, D)$
3	<i>Swish</i>	$G(x) = x/(1 + \exp(-x))$
5	<i>Gaussian</i>	$G(x) = \exp(-x^2)$
6	<i>Multiquadric</i>	$G(x) = \sqrt{x^2 + b^2}$
7	<i>Inverse Multiquadric</i>	$G(x) = 1/(x^2 + b^2)^{1/2}$

Fonte: Elaborada pelo autor.

Os parâmetros  $(\mathbf{w}, b)$  são randomicamente gerados (distribuídos de acordo com uma distribuição normal com média zero e desvio padrão igual a um) e os pesos  $\beta_i$  da camada de saída são determinados analiticamente. MLP e RBF são funções de ativação descritas nas Equações 3.11 e 3.12, respectivamente.

$$\text{MLP}(\mathbf{w}_i, b_i, \mathbf{x}) = \sum_{k=1}^D w_{ik} x_k + b_i \quad (3.11)$$

$$\text{RBF}(\mathbf{x}, \mathbf{c}) = \sum_{j=1}^D \frac{x_j - c_{ij}}{r_i} \quad (3.12)$$

onde  $D$  é o número de características de entrada, os centros  $c_{ij}$  são atribuídos uniformemente a partir do hiper-retângulo delimitador das variáveis de entrada [42] e  $r = \max(\|\mathbf{x} - \mathbf{c}\|)/\sqrt{2D}$ .

A Equação 3.10 apresenta uma função de saída que combina as funções de ativação MLP e RBF apresentadas acima, podendo ser considerada uma extensão da função apresentada na Equação 3.13.

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^L \beta_i G(\text{MLP}(\mathbf{w}_i, b_i, \mathbf{x})) \quad (3.13)$$

O uso do modelo apresentado na Equação 3.13 dispensa o uso das variáveis  $\alpha$  (que define a proporção entre as funções MLP e RBF) e  $\gamma$  (multiplicador da função de ativação RBF, atua como um regularizador), tornando o modelo computacionalmente menos complexo. Apesar do aumento da complexidade, a inclusão da função de ativação RBF traz mais flexibilidade o modelo, combinando um conjunto de funções gaussianas para a melhora do desempenho do método.

O vetor de pesos de saída  $[\beta_1, \dots, \beta_L]$  pode ser determinado a partir da minimização do erro de aproximação mostrado na Equação 3.14 [33].

$$\min_{\beta \in \mathbb{R}^L} \|\mathbf{H}\beta - \mathbf{y}\| \quad (3.14)$$

onde  $\mathbf{H}$  é a matriz de saída da camada oculta e  $\mathbf{y}$  é o vetor de saída, apresentados nas Equações 3.15 e 3.16, respectivamente, sendo  $N$  o tamanho do conjunto de dados utilizado.

$$\mathbf{H} = \begin{bmatrix} G_1(\alpha, \gamma, \mathbf{w}_1, b_1, \mathbf{c}, \mathbf{x}_1) & \cdots & G_L(\alpha, \gamma, \mathbf{w}_L, b_L, \mathbf{c}, \mathbf{x}_1) \\ \vdots & \ddots & \vdots \\ G_1(\alpha, \gamma, \mathbf{w}_1, b_1, \mathbf{c}, \mathbf{x}_N) & \cdots & G_L(\alpha, \gamma, \mathbf{w}_L, b_L, \mathbf{c}, \mathbf{x}_N) \end{bmatrix} \quad (3.15)$$

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \quad (3.16)$$

A solução ótima é indicada pela Equação 3.17.

$$\beta = (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \mathbf{y} = \mathbf{H}^\dagger \mathbf{y} \quad (3.17)$$

onde  $\mathbf{H}^\dagger$  é a pseudoinversa de  $\mathbf{H}$ . A Figura 6 apresenta as conectividades em uma rede do tipo ELM com 8 entradas, uma camada oculta com 10 neurônios e uma saída.

### 3.2.3 Extreme Gradient Boosting

*Boosting* é uma estratégia utilizada para aprimorar algoritmos de aprendizado de máquina, proposta inicialmente para tratar problemas de classificação de padrões. A partir da introdução do método *AdaBoost* [19], diversas generalizações surgiram, dentre elas o método GB que pode ser aplicado a problemas de classificação e regressão a partir da utilização de uma função objetivo diferenciável [31].

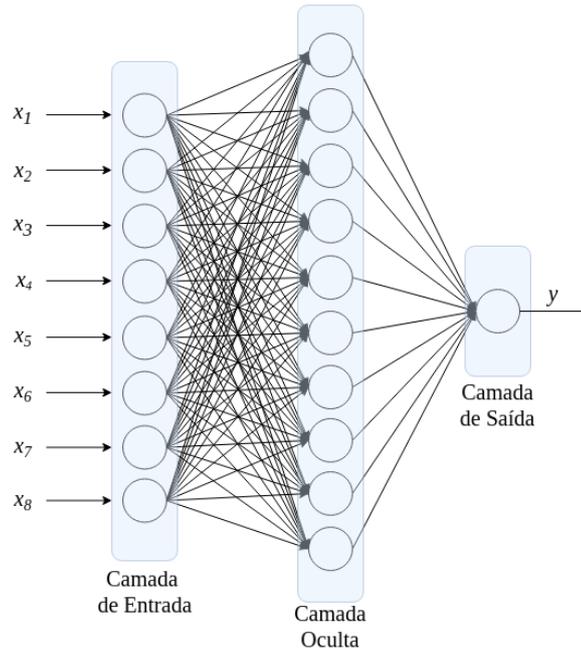


Figura 6 – Conectividades para uma rede do tipo ELM com 8 entradas, uma camada oculta com 10 neurônios e uma saída.

Fonte: Elaborada pelo autor.

O modelo GB é classificado como um tipo de método *ensemble*, que corresponde a combinação de vários modelos chamados *weak learners* (tipicamente árvores de decisão) para a criação de um modelo mais efetivo conhecido como *strong learner*, com capacidade de aumento da estabilidade e redução de erros relacionados a variância, ruído e viés. A ideia geral da maior parte dos métodos do tipo *boosting* é treinar preditores sequencialmente, de modo que cada uma tente corrigir seu predecessor.

Partindo de um conjunto de treino  $\{(x_i, y_i)\}_{i=1}^N$  com  $N$  amostras, o objetivo é encontrar uma aproximação  $\hat{f}(x)$  para uma função  $f(x)$  que minimize o valor esperado da função de perda

$$L(y, \hat{f}(x)) = \sum_i^N [y_i - \hat{f}(x_i)]^2. \quad (3.18)$$

GB [21] aproxima  $f$  a partir de uma expansão aditiva da forma:

$$\hat{f} = \sum_{m=1}^M \beta_m h(x, a_m) \quad (3.19)$$

onde a função  $h(x, a_m)$  é um nó  $K$  da árvore de regressão e os parâmetros  $\{\beta_m, a_m\}$  são ajustados conjuntamente aos dados de treinamento de maneira inteligente [31]. A cada iteração  $m$ , uma árvore de regressão particiona o espaço em regiões disjuntas  $\{R_{km}\}_{k=1}^K$  na  $m$ -ésima iteração. Uma constante  $\gamma_{jm}$  é atribuída a cada região e a regra preditiva é  $x \in R_{jm} \Rightarrow f(x) = \gamma_{jm}$ . Utilizando a notação indicial, a saída de  $h$  para a entrada  $x$  pode

ser escrita como:

$$h(x, \{R_{km}\}_{k=1}^K) = \sum_{k=1}^K \gamma_{km} I(x \in R_{km}), \quad (3.20)$$

onde,

$$I(\cdot) = 1 \text{ se } x \in R_{km} \text{ senão } 0 \quad (3.21)$$

com os parâmetros  $\{R_{km}, \gamma_{km}\}$ ,  $k = 1, 2, \dots, J$ ,  $m = 1, \dots, M$ , onde  $\gamma_{km}$  é o valor predito na região  $R_{km}$ .

Como o modelo (3.20) prevê um valor constante em cada região  $R_{km}$ , a solução se reduz a:

$$\gamma_{km} = \gamma \arg \min_{f_{m-1}} \sum_{x_i \in R_{km}} L(y_i, f_{m-1}(x_i) + \gamma), \quad \gamma \text{ constante.} \quad (3.22)$$

A atual aproximação  $f_{m-1}(x)$  é então atualizada seguindo a regra

$$\hat{f}_m(x) = \hat{f}_{m-1}(x) + \lambda \sum_{k=1}^K \gamma_{km} I(x \in R_{km}) \quad (3.23)$$

onde o parâmetro  $0 < \lambda \leq 1$  é chamado de *learning rate*.

Uma melhora considerável na acurácia do GB pode ser alcançada quando, a cada iteração do algoritmo, o modelo base é ajustado a uma subamostra do conjunto de treinamento sorteada aleatoriamente sem reposição. O tamanho da subamostra é uma fração constante  $\eta$  do tamanho do conjunto de treino. Valores pequenos para  $\eta$  introduzem aleatoriedade ao algoritmo e auxiliam na prevenção de *overfitting*<sup>1</sup> [21]. O algoritmo também se torna mais rápido, pois as árvores de regressão são aproximadas a conjuntos de dados menores a cada iteração.

O Algoritmo 1 apresenta o método *gradient tree-boosting* genérico para regressão. O parâmetro  $0 < \lambda \leq 1$  controla a taxa de aprendizado (*learning rate*) do procedimento de *boosting* e atua como um parâmetro de regularização, enquanto  $0 < \lambda \leq 1$  é a fração das amostras de treino (sem reposição) utilizada para calcular os parâmetros ótimos do modelo na iteração  $m$ . Diferentes algoritmos podem ser obtidos a partir da inserção de outras funções de perda  $L(y, f(x))$ .

O método XGB segue os mesmos princípios do GB, com algumas diferenças em detalhes da modelagem que performam aproximações mais acuradas utilizando a derivada de segunda ordem da função de perda (no caso do uso de função logística), regularização  $L1$  e  $L2$  e computação paralela. XGB é a forma mais regularizada do GB, utilizando regularizações semelhantes as do método NET, que melhoram as capacidades de generalização do modelo. Apresenta melhor performance computacional por conta de ser capaz de realizar treinamentos mais rápidos que podem ser distribuídos através de

<sup>1</sup> *Overfitting* é um termo usado em estatística para descrever quando um modelo estatístico se ajusta muito bem ao conjunto de dados anteriormente observado, mas se mostra ineficaz para prever novos resultados.

---

**Algoritmo 1:** Algoritmo XGB.
 

---

**Input:** Conjunto de treino  $\{(x_i, y_i)\}_{i=1}^N$  com  $N$  amostras onde  $y_i = f(x_i)$ , uma função de perda diferenciável  $L(y, f(x))$ , a profundidade de cada árvore  $K$ , a taxa de reamostragem  $\eta$ , learning rate  $\lambda$  e o número de iterações  $M$

1 . **Output:** Aproximar  $\hat{f}(x)$  a  $f(x)$

2 Inicializar  $\hat{f}_0(x) = \arg \min_{\gamma} \sum_{i=1}^N L(y_i, \gamma)$ , com  $\gamma$  constante

3 **for**  $m = 1, \dots, M$  **do**

4     1. Calcular os resíduos como o gradiente negativo da resposta

$$r_{im} = - \left[ \frac{\partial L(y_i, f(x_i))}{\partial f(x_i)} \right]_{f(x)=\hat{f}_{m-1}(x)}, \text{ for } i = 1, 2, \dots, N$$

      2. Randomicamente selecionar  $\eta \times N$  amostras do conjunto de dados (passo de reamostragem)

5     3. Aproximar uma árvore de regressão com  $K$  nós aos resíduos  $r_{im}$  utilizando  $\eta \times N$  amostras. Essa árvore é aproximada utilizando somente as amostras selecionadas randomicamente.

6     4. Calcular os parâmetros ótimos  $\gamma_{1m}, \dots, \gamma_{km}$  como

$$\gamma_{km} = \arg \min_{\gamma} \left( \sum_{x_i \in R_{km}} L(y_i, \hat{f}_{m-1}(x_i) + \gamma) \right) + \Omega(f_{m-1}), \text{ for } k = 1, 2, \dots, K$$

      onde  $R_{km}$  é o conjunto de  $x$  que define o nó  $k$ .

7     5. Atualizar  $\hat{f}_m(x) = \hat{f}_{m-1}(x) + \lambda \sum_{k=1}^K \gamma_{km} I(x \in R_{km})$ .

8 **end**

9 **return**  $\hat{f}(x) = \hat{f}_M(x)$

---

diferentes núcleos [8]. Utiliza estruturas de dados aprimoradas para melhor utilização da memória cache<sup>2</sup> do processador, o que o torna mais rápido. Além disso, a biblioteca que implementa o método XGB [9] está preparada para lidar com dados faltantes internamente, não sendo necessário um tratamento separado para esses casos.

### 3.3 Evolução Diferencial

O algoritmo DE é um método de programação evolutiva desenvolvido por Storn and Price [72] para a otimização de problemas em domínios contínuos. No DE, o valor de cada variável é representado por um número real. Algumas das vantagens do DE são sua estrutura simples, facilidade de uso e velocidade. O algoritmo utiliza a mutação como mecanismo de busca e seleção para direcionar a pesquisa para as regiões mais promissoras dentro da região viável. Algoritmos genéticos geram uma sequência de populações usando

<sup>2</sup> A memória cache é um tipo de memória que trabalha em conjunto com o processador, com o objetivo de potencializar o desempenho do chip de processamento, evitando que fique ocioso por longos períodos.

mecanismos de seleção, mutação e cruzamento. A principal diferença entre os algoritmos genéticos e a evolução diferencial é que os algoritmos genéticos se baseiam no cruzamento, um mecanismo de troca probabilística de informações para localizar as melhores soluções, enquanto as estratégias evolutivas usam a mutação como o principal mecanismo de busca.

DE é uma técnica de pesquisa baseada em população que utiliza uma população  $P$  de vetores de parâmetros  $D$ -dimensionais para cada geração. A população inicial é escolhida aleatoriamente se nenhuma informação estiver disponível sobre o problema. No caso da solução preliminar disponível, a população inicial é frequentemente gerada adicionando desvios aleatórios normalmente distribuídos à solução preliminar [72]. A ideia básica do DE é o esquema para geração de vetores de parâmetros de teste. O algoritmo gera novos vetores de parâmetros adicionando o vetor de diferenças ponderadas entre dois membros da população a um terceiro membro. Se o vetor resultante produz um valor de função objetivo mais baixo do que um membro da população predeterminado, o vetor recém-gerado substitui o vetor com o qual foi comparado. Além disso, o melhor vetor de parâmetros é avaliado para cada geração, a fim de acompanhar o progresso feito durante o processo de otimização. Extrair as informações de distância e direção da população para gerar desvios aleatórios resulta em um esquema adaptativo com excelentes propriedades de convergência [61].

O algoritmo DE não possui um paradigma natural, isto é, não é biologicamente inspirado. Para a evolução da população de possíveis soluções, inicialmente uma população de  $p$  indivíduos é randomicamente dispersa no espaço de busca. A população, denotada por  $P_{\mathbf{x},g}$ , pode ser definida como:

$$\mathbf{x}_L \leq \mathbf{x}_{i,0} \leq \mathbf{x}_U \quad \forall i \in \{1, 2, \dots, P\} \quad (3.24)$$

$$\mathbf{P}_{\mathbf{x},g} = (\mathbf{x}_{i,g}), \quad i \in \{1, 2, \dots, P\}, \quad g \in \{0, 1, \dots, g_{max}\} \quad (3.25)$$

$$\mathbf{x}_{i,g} = (x_{j,i,g}), \quad j \in \{1, 2, \dots, D\} \quad (3.26)$$

onde,  $P_{x,g}$  é uma matriz com  $p$  vetores (indivíduos),  $\mathbf{x}_{i,g}$  é um vetor  $D$ -dimensional que representa a solução,  $i$  representa o índice para os vetores,  $g$  o índice para as gerações,  $j$  o índice para as variáveis a serem otimizadas e  $g_{max}$  o número máximo de gerações. Para cada passo evolucionário (geração), uma população modificada  $P_{v,g}$  é formada a partir da população atual  $P_{x,g}$ , da seguinte forma:

$$\mathbf{v}_{i,g} = \mathbf{x}_{best,g} + F(\mathbf{x}_{r0,g} - \mathbf{x}_{r1,g}) + F(\mathbf{x}_{r2,g} - \mathbf{x}_{r3,g}) \quad (3.27)$$

onde,  $r0, r1, r2, r3$  são índices randômicos em  $0, 1, \dots, p - 1$  e  $F$  é um parâmetro escalar do algoritmo. A seguir, uma população teste  $P_{u,g}$  é formada:

$$\mathbf{u}_{i,g} = (u_{j,i,g}) = \begin{cases} v_{j,i,g}, & \text{se } (r \leq C_r \text{ ou } j = j_{rand}) \\ x_{j,i,g}, & \text{caso contrário} \end{cases} \quad (3.28)$$

onde,  $r$  é um número randômico com distribuição uniforme no intervalo  $(0, 1)$  que é re-amostrado a cada iteração,  $j_{rand}$  é um índice randômico em  $0, 1, \dots, p - 1$  que garante que pelo menos uma variável a ser otimizada seja originada do vetor modificado e  $C_r$  é um parâmetro escalar no intervalo  $(0, 1]$  do algoritmo DE. O passo final é o critério de seleção guloso, que realiza uma escolha localmente ótima. No caso do problema de minimização:

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g}, & \text{se } f(\mathbf{u}_{i,g}) \leq f(\mathbf{x}_{i,g}) \\ \mathbf{x}_{i,g}, & \text{caso contrário} \end{cases} \quad (3.29)$$

A formulação descrita corresponde a variante do algoritmo DE conhecida como *best2bin* [61], onde "best" indica que o vetor base é atualmente o melhor vetor na população, "2" significa que a diferença entre dois vetores é utilizada para formar a população modificada e o termo "bin" (de distribuição binomial) indica que o *crossover* uniforme<sup>3</sup> é empregado durante a formação da população teste.

### 3.4 Validação Cruzada

Validação cruzada é uma técnica de amostragem estatística para avaliar a habilidade de generalização de um modelo a partir de um conjunto de dados [40]. As técnicas de validação cruzada buscam estimar o quão preciso é o modelo para um novo conjunto de dados. Entre as técnicas de validação cruzada, a técnica *K-Fold* [31] é uma das mais utilizadas. A técnica *K-Fold* divide os dados disponíveis em  $K$  partes, utiliza alguns conjuntos para a treinar o modelo e outras partes diferentes para testá-lo. Um conjunto de dados contendo  $N$  amostras é randomicamente dividido em  $K$  subconjuntos, onde  $K \geq 1$ ; dos  $K$  subconjuntos,  $K - 1$  são utilizados para treino e o conjunto restante é usado para teste. Esse processo é repetido  $K$  vezes, utilizando um conjunto diferente para teste a cada iteração.

A validação *K-fold* reduz a variação na estimativa do desempenho do modelo para diferentes amostras de dados e, por conta disso, o desempenho se torna menos sensível ao particionamento dos dados. A Figura 7 apresenta um exemplo de conjunto de dados dividido em  $K = 5$  subconjuntos a cada iteração do processo de validação cruzada.

### 3.5 Seleção de Modelos Baseada em Evolução Diferencial

Diversos modelos da literatura possuem importantes parâmetros que não podem ser diretamente estimados dos dados de entrada [43]. Chamados de hiperparâmetros, possuem valores definidos antes do início do processo de aprendizado dos algoritmos. A performance do modelo pode ser afetada diretamente pela escolha dos hiperparâmetros, mas escolher

<sup>3</sup> Recombinação uniforme onde os elementos dos vetores são comparados individualmente e trocados com uma probabilidade fixada de 50%.

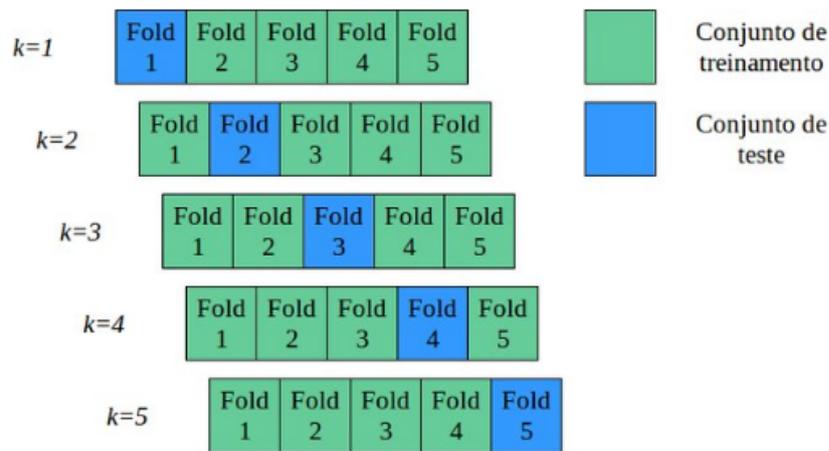


Figura 7 – Diagrama do método de validação cruzada  $K$ -fold com  $K = 5$ .

Fonte: Soares et al. [71]

o melhor conjunto de valores pode se tornar uma tarefa complexa [12]. Frequentemente, estes hiperparâmetros são definidos de modo empírico através de testes manuais ou com o uso de busca exaustiva (*Grid Search*) [56]. Importante ressaltar que o método de busca exaustiva pode ser computacionalmente custoso dependendo do tamanho do espaço de hiperparâmetros e gastar um tempo impraticável para encontrar o conjunto de parâmetros que forneça a melhor performance do modelo testado.

Uma alternativa é o uso de algoritmos de otimização evolucionária, que permitem encontrar o conjunto de hiperparâmetros que produza um modelo ótimo ou sub-ótimo que minimize uma função objetivo pré-determinada a partir dos dados de teste. No presente trabalho foi empregado o método de DE [51] para encontrar conjuntos de parâmetros para os métodos NET, ELM e XGB. Para a execução do processo de otimização, os indivíduos são representações dos modelos NET, ELM e XGB para a execução do processo descrito na Seção 3.3. Os parâmetros utilizados no DE foram os seguintes: número de iterações=300, tamanho da população=30 indivíduos, mutação(0,5; 1) e recombinação=0,7. Como o algoritmo DE é estruturado para a solução de problemas em domínios contínuos, para os hiperparâmetros discretos, a estratégia utilizada para a transformação dos valores contínuos em discretos foi a consideração da parte inteira do valor retornado pelo DE durante o processo de otimização. A Figura 8 apresenta um fluxograma de execução do acoplamento do algoritmo evolutivo aos métodos de regressão testados. Dessa forma, a partir do uso da métrica MAPE [78] como função objetivo, os parâmetros dos métodos são otimizados com o uso do DE a fim de minimizar o erro MAPE e melhorar as previsões dos algoritmos.

Para cada um dos algoritmos, diferentes conjuntos de hiperparâmetros foram otimizados. No caso do modelo de regressão NET, foram otimizados valores para os parâmetros `l1_ratio` e `alpha` apresentados na Tabela 4, onde `l1_ratio` é um parâmetro

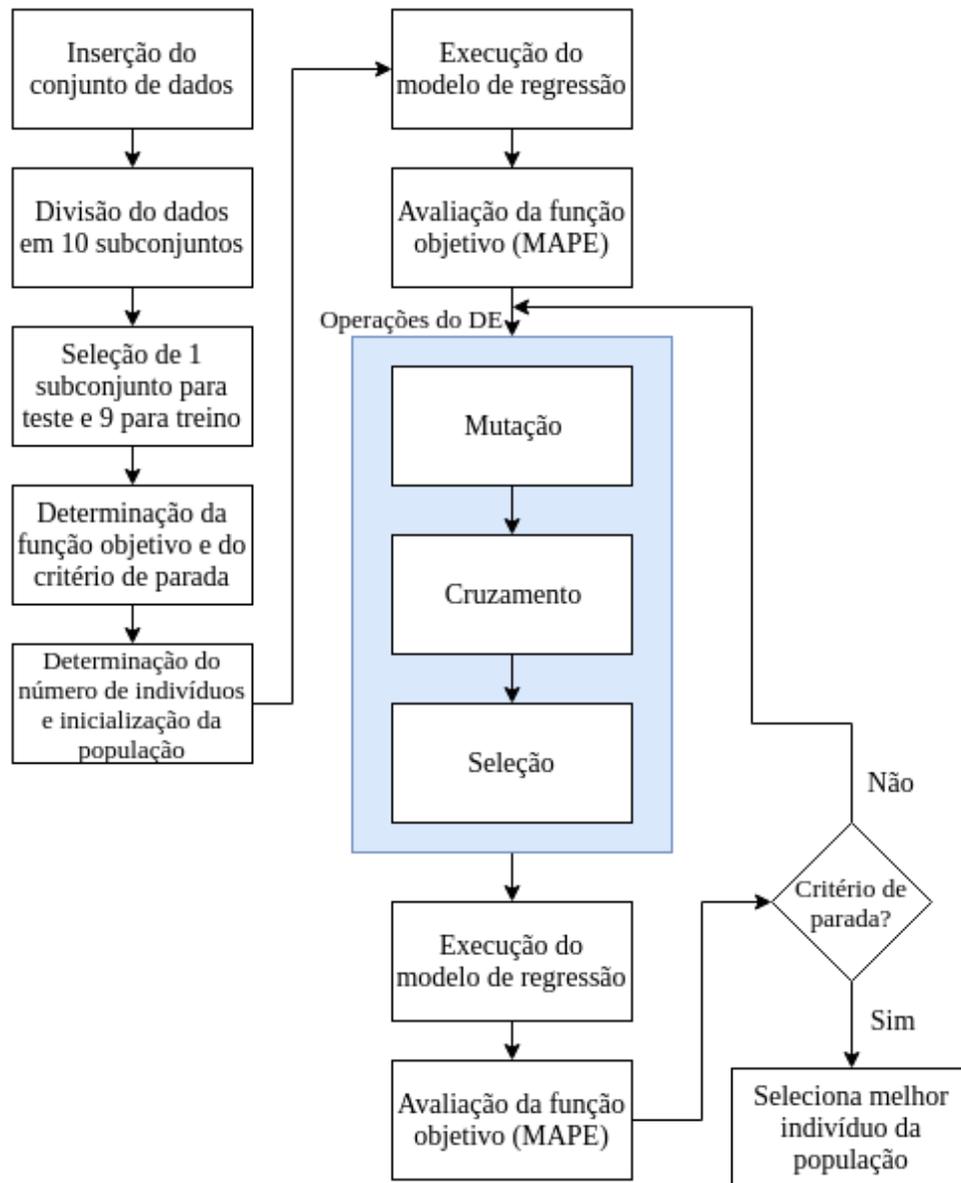


Figura 8 – Fluxograma de execução do algoritmo de otimização DE em conjunto com os modelos de regressão NET, ELM e XGB.

Fonte: Elaborada pelo autor.

de penalização compartilhado, `alpha` define a proporção entre a regularização  $L1$  e  $L2$ , `max_iter` é o número máximo de iterações do modelo de aproximação, `tol` é a tolerância para a aproximação do modelo, de forma que se as atualizações forem menores que o parâmetro `tol` o processo de aproximação se encerra antes de alcançar `max_iter`, e `normalize` se refere a normalização dos dados antes da execução do modelo.

Tabela 4 – Hiperparâmetros do método NET.

Modelo	Parâmetros	Conjunto de Hiperparâmetros
NET	<code>l1_ratio</code>	[0,1]
	<code>alpha</code>	[0.1,1]
	<code>max_iter</code>	1000
	<code>tol</code>	0.0001
	<code>normalize</code>	false

Fonte: Elaborada pelo autor.

Para o método ELM, os hiperparâmetros otimizados foram `n_hidden`, `rbf_width`, `activation_func` e `alpha`, apresentados na Tabela 5, onde `n_hidden` é o número de neurônios para geração da camada oculta, `activation_func` é a função de ativação não-linear de saída, `rbf_width` e `alpha` são os hiperparâmetros  $\gamma$  e  $\alpha$ , respectivamente, utilizados na composição do vetor de previsões apresentado na Equação 3.10 na Seção 3.2.2.

Tabela 5 – Hiperparâmetros do método ELM.

Modelo	Parâmetros	Conjunto de Hiperparâmetros
ELM	<code>n_hidden</code>	1, 2, 3 ... , 500
	<code>rbf_width</code>	[0.01, 10]
	<code>activation_func</code>	identity, relu, swish, gaussian, multiquadric, inv_multiquadric
	<code>alpha</code>	[0, 1]

Fonte: Elaborada pelo autor.

No modelo XGB, os hiperparâmetros otimizados foram `learning_rate`, `n_estimators`, `colsample_bytree`, `min_child_weight`, `subsample` e `max_depth`, apresentados na Tabela 6. O parâmetro `learning_rate` (apresentado como  $\lambda$  na Seção 3.2.3) se refere a diminuição do tamanho da passo utilizado na atualização para impedir o ajuste excessivo (*overfitting*), tornando o processo de *boosting* mais conservador, `n_estimators` é o número de árvores de regressão (apresentado como  $M$  na Seção 3.2.3), `colsample_bytree` é a taxa de subamostragem utilizada ao construir cada árvore (apresentado como  $\eta$  na Seção 3.2.3). O hiperparâmetro `min_child_weight` é a soma mínima de pesos necessária para o particionamento em um nó folha, `subsample` é a proporção de subamostras das instâncias de treinamento, que também auxilia na prevenção de *overfitting*, `max_depth` representa a profundidade máxima que uma sub-árvore pode alcançar, e `objective` especifica a tarefa

de aprendizado e o objetivo correspondente (neste caso a regressão é realizada com o valor de perda ao quadrado) [9].

Tabela 6 – Hiperparâmetros do método XGB.

Modelo	Parâmetros	Conjunto de Hiperparâmetros
XGB	<code>learning rate</code>	[0, 1]
	<code>n_estimators</code>	10, 11, 12, \dots, 900
	<code>colsample_bytree</code>	[0, 1]
	<code>min_child_weight</code>	1, 2, 3, \dots, 10
	<code>subsample</code>	[0, 1]
	<code>max_depth</code>	1, 2, 3, \dots, 30
	<code>objective</code>	<code>squared_error</code>

Fonte: Elaborada pelo autor.

### 3.6 Métricas de Avaliação

O Erro Absoluto Percentual Médio (MAPE, do inglês *Mean Absolute Percent Error*) [78] é utilizado neste trabalho como função objetivo no algoritmo DE, além de ser utilizado para análise e comparação das saídas dos modelos de regressão testados com os valores observados. Além de ser empregado em trabalhos que realizem previsões, os resultados do MAPE são apresentados em pontos percentuais, sendo um benefício desta medida que torna sua interpretação fácil e direta.

$$MAPE = 100 \times \frac{1}{N} \sum_{i=0}^{N-1} \frac{|y_i - \hat{y}_i|}{|y_i|} \quad (3.30)$$

O coeficiente de determinação  $R^2$ , é uma medida do ajuste de um modelo estatístico linear generalizado, como a regressão linear, em relação aos valores observados. O  $R^2$  varia entre 0 e 1, indicando o quanto o modelo consegue explicar os valores observados. Quanto maior o  $R^2$ , mais explicativo é o modelo, melhor ele se ajusta à amostra [50].

$$R^2 = 1 - \frac{\sum_{i=0}^{N-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{N-1} (y_i - \bar{y})^2} \quad (3.31)$$

A métrica Raiz do Erro Quadrático Médio (RMSE, do inglês *Root Mean Squared Error*) é a raiz quadrada da variação dos resíduos. Indica a adequação absoluta do modelo aos dados - quão próximos os pontos de dados observados estão dos valores previstos do modelo. Enquanto  $R^2$  é uma medida relativa do ajuste, o RMSE é uma medida absoluta do ajuste. Como raiz quadrada de uma variação, o RMSE pode ser interpretado como o desvio padrão da variação e tem a propriedade útil de estar nas mesmas unidades que a variável de resposta. Valores mais baixos de RMSE indicam melhor ajuste e sendo uma boa medida da precisão com que o modelo prevê a resposta.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (y_i - \hat{y}_i)^2} \quad (3.32)$$

O Erro Médio Absoluto (MAE do inglês, *Mean Absolute Error*) mede a magnitude média dos erros em um conjunto de previsões, sem considerar sua direção. É a média na amostra de teste das diferenças absolutas entre previsão e observação real em que todas as diferenças individuais têm peso igual.

$$MAE = \frac{1}{N} \sum_{i=0}^{N-1} |y_i - \hat{y}_i| \quad (3.33)$$

### 3.7 Testes Estatísticos

O teste Shapiro-Wilk permite verificar se uma amostra aleatória vem de uma distribuição normal [22]. O teste fornece um p-valor; onde valores pequenos indicam que a amostra não possui uma distribuição normal (rejeitando a hipótese nula de que a população é normalmente distribuída se seus valores estiverem abaixo de um determinado limite, normalmente 0,05). A fórmula para o valor p é:

$$p = \frac{(\sum_{i=1}^n a_i x_i)^2}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (3.34)$$

onde  $x_i$  representa os valores ordenados da amostra,  $a_i$  as constantes geradas a partir das covariâncias, variâncias e médias da amostra a partir de uma amostra normalmente distribuída.

O teste de Lilliefors é uma adaptação do teste de Kolmogorov-Smirnov [87], que pode ser usado quando parâmetros da distribuição teórica são estimados a partir dos dados, em vez de serem conhecidos *a priori*.

A estatística do teste é a mesma do teste de Kolmogorov-Smirnov, ou seja, a diferença máxima entre a função de distribuição empírica e a função de distribuição cumulativa teórica. É um teste de normalidade, corrigido em relação ao Kolmogorov-Smirnov para pequenos valores nas caudas das distribuições de probabilidade. A hipótese nula ( $H_0$ ) para o teste é que os dados provêm de uma distribuição normal. A hipótese alternativa ( $H_1$ ) é que os dados não provêm de uma distribuição normal e o teste pressupõe que seja apresentada uma amostra aleatória.

Se o resultado do teste for significativamente grande ( $p - valor > 0,05$ , a hipótese nula é rejeitada e pode-se concluir que os dados não são normais.

Para comparar a distribuição de três ou mais grupos de amostras independentes originadas de métodos de regressão e indicar grupos em que a diferença é significativa, são utilizados testes paramétricos ou não paramétricos dependendo da normalidade das amostras. Nas estatísticas paramétricas, assumimos que as amostras são coletadas a partir de distribuições totalmente especificadas caracterizadas por um ou mais parâmetros desconhecidos sobre o qual queremos fazer inferência. Em um método não paramétrico,

assumimos que a distribuição de origem da amostra é indeterminada e que estamos interessados em fazer inferência sobre o centro da distribuição. Métodos não paramétricos são úteis quando a suposição de normalidade não se sustenta e seu tamanho da amostra é pequeno.

O teste de Kruskal-Wallis é não paramétrico e pode ser utilizado para determinar se há diferenças estatisticamente significativas entre dois ou mais grupos de uma variável independente considerando contínuas ou ordinais. É considerado como alternativa ao teste paramétrico ANOVA para permitir a comparação de mais de dois grupos independentes [83]. O teste determina se as medianas de dois ou mais grupos são diferentes. Como a maioria dos testes estatísticos, uma estatística de teste é calculada e comparada com um ponto de corte da distribuição (considerando normalmente o nível de significância igual a 0,05). A estatística de teste usada no Kruskal-Wallis é chamada de estatística  $H$ . As hipóteses para o teste são:

$H_0$  : as medianas das populações são iguais.

$H_1$  : as medianas das populações não são iguais.

O teste de Kruskal-Wallis informa se há uma diferença significativa entre os grupos. No entanto, não informa quais grupos são diferentes. Para isso, é necessário o uso de um teste *Post Hoc*.

O teste de Dunn é um método estatístico *Post Hoc* não paramétrico usado para fazer um número específico de comparações entre grupos de dados e descobrir qual deles é significativo [59, 14]. Após o teste de Kruskal-Wallis indicar uma diferença significativa em três ou mais métodos, o teste de Dunn pode ser utilizado para identificar quais métodos específicos são mais significativos dos que os outros. A hipótese nula para o teste é que não há diferença entre os grupos. A hipótese alternativa é a de que há uma diferença entre os grupos. Para indicar os métodos mais significativos, o teste calcula diferenças entre os pares de métodos, p-valores e ordena de forma crescente os dados de entrada do teste, a fim de retornar a média dessa classificação e ordenar os melhores métodos a partir dos valores de erro.

## 4 Resultados e Discussão

Os experimentos computacionais para cada um dos modelos de regressão testados (NET, ELM e XGB) foram executados 30 vezes de forma independente. Os algoritmos foram implementados na linguagem Python em sua versão 3.5, e com o uso das bibliotecas Scikit-learn [56] (NET e ELM), XGBoost [9] (XGB) e Scipy [38] (DE), para a implementação dos métodos de aprendizado de máquina, e as bibliotecas Numpy [82] e Pandas [46], para manipulação dos dados. Os experimentos foram conduzidos em computadores com as seguintes especificações: CPU Intel(R) Xeon(R) E5620 (8 cores de 2,40GHz e memória cache de 2MB), memória RAM de 8GB e sistema operacional Linux Ubuntu 14.04 LTS.

Durante a apresentação dos resultados, a Base de Dados 1 disponibilizada por Tsanas and Xifara [81] será referida como BD1 e a Base de Dados 2 disponibilizada por Jihad and Tahiri [37] será mencionada como BD2. As cargas de aquecimento serão representadas nas tabelas por HL (do inglês *Heating Load*) e as cargas de resfriamento por CL (do inglês *Cooling Load*).

Para o método ELM, inicialmente foram testadas duas configurações da função de saída: utilizando apenas a função de ativação MLP (Equação 3.11) e com uma combinação de MLP e RBF (Equação 3.12). Estes testes tinham como objetivo escolher a configuração do ELM a ser utilizada, baseadas nas configurações propostas por Huang et al. [35] e Huang and Siew [34]. No caso das cargas de aquecimento, os melhores modelos obtidos para a BD1 e BD2 obtiveram os valores 2,316 e 19,214 para a métrica MAPE, respectivamente. Para as cargas de resfriamento, os melhores modelos para BD1 e BD2 obtiveram 4,469 e 1,895 de MAPE, respectivamente. Por conta dos resultados obtidos nos testes preliminares, optou-se por utilizar o modelo ELM que combina as funções de ativação MLP e RBF. Dessa forma, os resultados descritos para o modelo ELM nos gráficos e tabelas apresentados neste trabalho são referentes a implementação do método com a função de saída descrita na Equação 3.10.

### 4.1 Base de Dados 1 (BD1)

A Tabela 7 apresenta um resumo das médias das métricas estatísticas obtidas ao longo das 30 execuções independentes para as cargas de aquecimento e resfriamento. No caso das métricas MAE, RMSE e MAPE os menores valores indicam melhor desempenho e para o coeficiente  $R^2$ , os melhores modelos devem apresentar seu valor mais próximo de 1. Através da observação das métricas, é possível verificar que o método XGB alcançou melhores resultados, tanto para as cargas de aquecimento quanto para as de resfriamento. No caso das cargas de aquecimento, o método ELM obteve valores ligeiramente próximos dos alcançados pelo XGB nas métricas MAE, RMSE e  $R^2$ . Para todas as métricas apresentadas é possível notar um comportamento comum entre os três métodos testados,

sendo este o melhor desempenho médio na previsão das cargas de aquecimento.

Tabela 7 – Média das métricas estatísticas obtidas ao longo das 30 execuções dos algoritmos para as cargas de aquecimento (HL) e resfriamento (CL) da BD1. Os melhores valores são mostrados em negrito e os desvios padrão entre parênteses.

Saída	Modelo	Métrica			
		MAE	RMSE	MAPE	R2
HL	NET	2,236(0,007)	3,188(0,005)	10,213(0,019)	0,900(0,001)
	ELM	0,337(0,009)	0,474(0,016)	1,669(0,057)	0,998(0,001)
	XGB	<b>0,155(0,005)</b>	<b>0,254(0,009)</b>	<b>0,785(0,024)</b>	<b>0,999(4 × 10<sup>-5</sup>)</b>
CL	NET	2,386(0,003)	3,386(0,004)	9,437(0,013)	0,873(0,001)
	ELM	0,897(0,024)	1,288(0,046)	3,599(0,079)	0,982(0,001)
	XGB	<b>0,249(0,010)</b>	<b>0,367(0,036)</b>	<b>1,054(0,037)</b>	<b>0,998(0,001)</b>

Fonte: Elaborada pelo autor.

A Figura 9 apresenta o gráfico da distribuição do parâmetro `l1_ratio` retornado pelo algoritmo evolutivo para o método NET aplicado a previsão das cargas de aquecimento ao longo das 30 execuções. É possível notar que o valor `l1_ratio=1` foi selecionado em 29 de 30 execuções. Para o parâmetro `alpha`, os valores selecionados se distribuíram no intervalo  $[0,114; 0,193]$  com média 0,186, desvio padrão 0,014, mediana 0,189 e sem *outliers*.

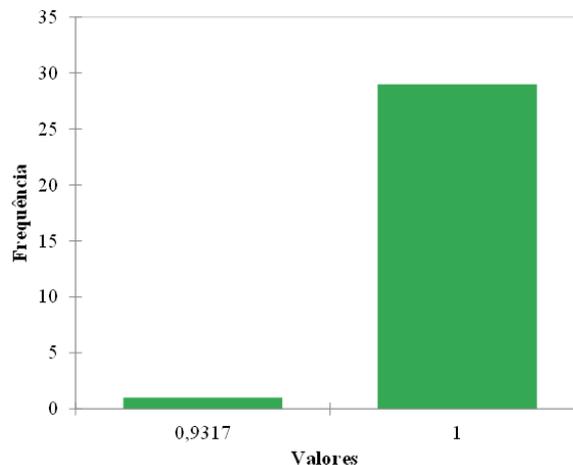


Figura 9 – Distribuição do parâmetro `l1_ratio` do método *Elastic Net* aplicado a previsão das cargas de aquecimento da BD1.

Fonte: Elaborada pelo autor.

A Figura 10 apresenta as distribuições dos parâmetros `alpha`, `rbf_width`, `n_hidden` e `activation_func` utilizados no método ELM aplicado a previsão das cargas de aquecimento. É possível notar que o parâmetro `alpha` variou no intervalo  $[0; 0,53]$  enquanto o parâmetro `rbf_width`, apesar de possuir um intervalo de possibilidades maior ( $[0,01; 10]$ ) variou apenas entre os valores  $[0,010; 0,067]$ , indicando que o intervalo utilizado no algoritmo de otimização pode ser diminuído e assim consequentemente diminuir o espaço de busca. O número de neurônios na camada oculta da ELM variou entre 190 e 314, com

média de 254,87 e mediana de 259,5. A função de ativação com maior frequência foi a *gaussian*, sendo selecionada em 20 de 30 execuções.

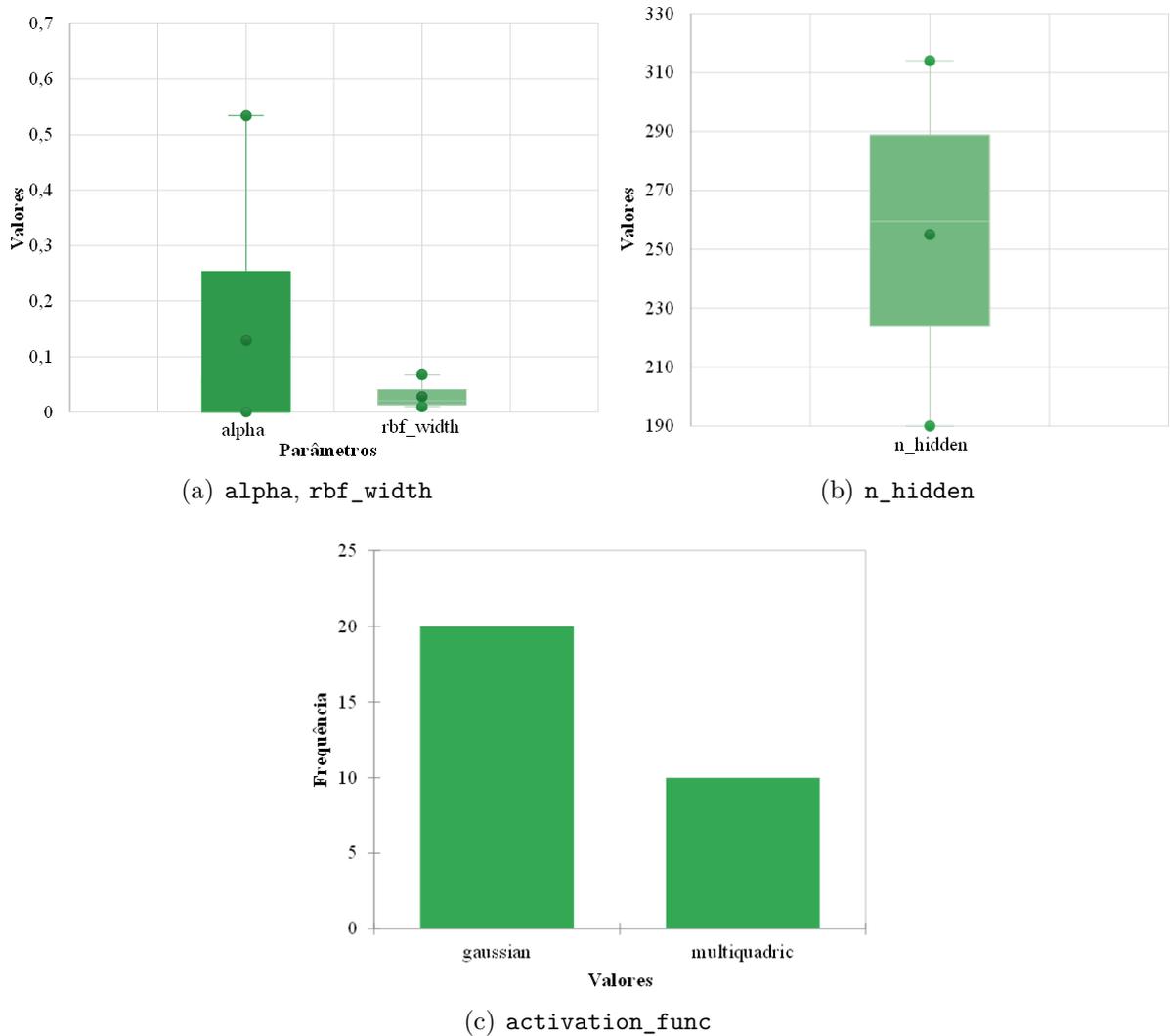


Figura 10 – Distribuições dos parâmetros `alpha`, `rbf_width`, `n_hidden` e `activation_func` utilizados no método ELM para a predição das cargas de aquecimento da BD1.

Fonte: Elaborada pelo autor.

A Figura 11 apresenta (a) as distribuições dos parâmetros `colsample_bytree`, `learning_rate`, `subsample`, (b) a distribuição do parâmetro `n_estimators`, (c) e (d) as frequências dos parâmetros `min_child_weight` e `max_depth` utilizados no método XGB para a predição das cargas de aquecimento. O hiperparâmetro `colsample_bytree` apresentou valores no intervalo  $[0,630; 0,867]$ , com média 0,728 e mediana 0,719. O `learning_rate` teve seus valores distribuídos entre  $[0,043; 0,422]$ , média 0,183 e mediana 0,159, enquanto o `subsample` possuiu valores bem próximos de 1, com distribuição entre  $[0,915; 0,999]$ , média 0,968 e mediana 0,976. O `n_estimators`, que possui como intervalo de variação  $[10; 900]$  obteve média 654,633 e mediana 720,500. Para o parâmetro `min_child_weight` o valor 1 foi selecionado em 29 de 30 execuções e para o

`max_depth` o valor 7 obteve maior frequência, sendo selecionado em 16 execuções.

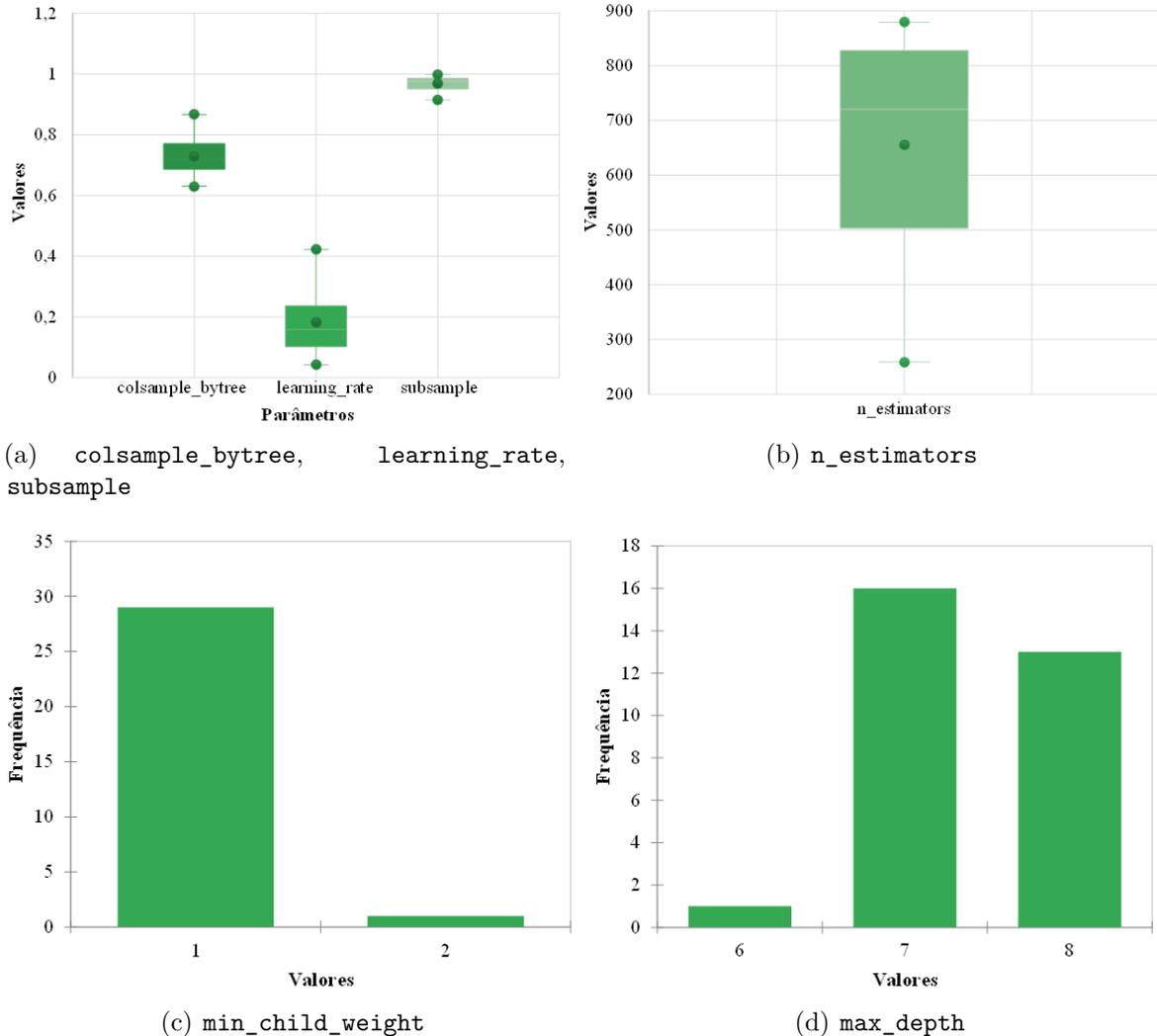


Figura 11 – Distribuições dos parâmetros `colsample_bytree`, `learning_rate`, `subsample`, `n_estimators` e as frequências dos parâmetros `min_child_weight` e `max_depth` utilizados no método XGB para a predição das cargas de resfriamento da BD1.

Fonte: Elaborada pelo autor.

A Figura 12 apresenta os gráficos de convergência da função objetivo para os métodos testados obtidos através da média dos valores retornados nas 30 execuções independentes. O método NET converge para o valor final de função objetivo em torno da vigésima quinta iteração, enquanto os métodos ELM e XGB alcançam essa convergência posteriormente. Isto pode ser explicado pela simplicidade do método NET em comparação com os outros dois testados. Para o método ELM é possível notar que os valores de função objetivo diminuem lentamente, não sendo possível atestar convergência para este método com o número de iterações máximo igual a 300.

A Figura 13 apresenta a distribuição de frequências para o parâmetro `l1_ratio` utilizado no método NET para predição das cargas de resfriamento. O valor 1 foi escolhido

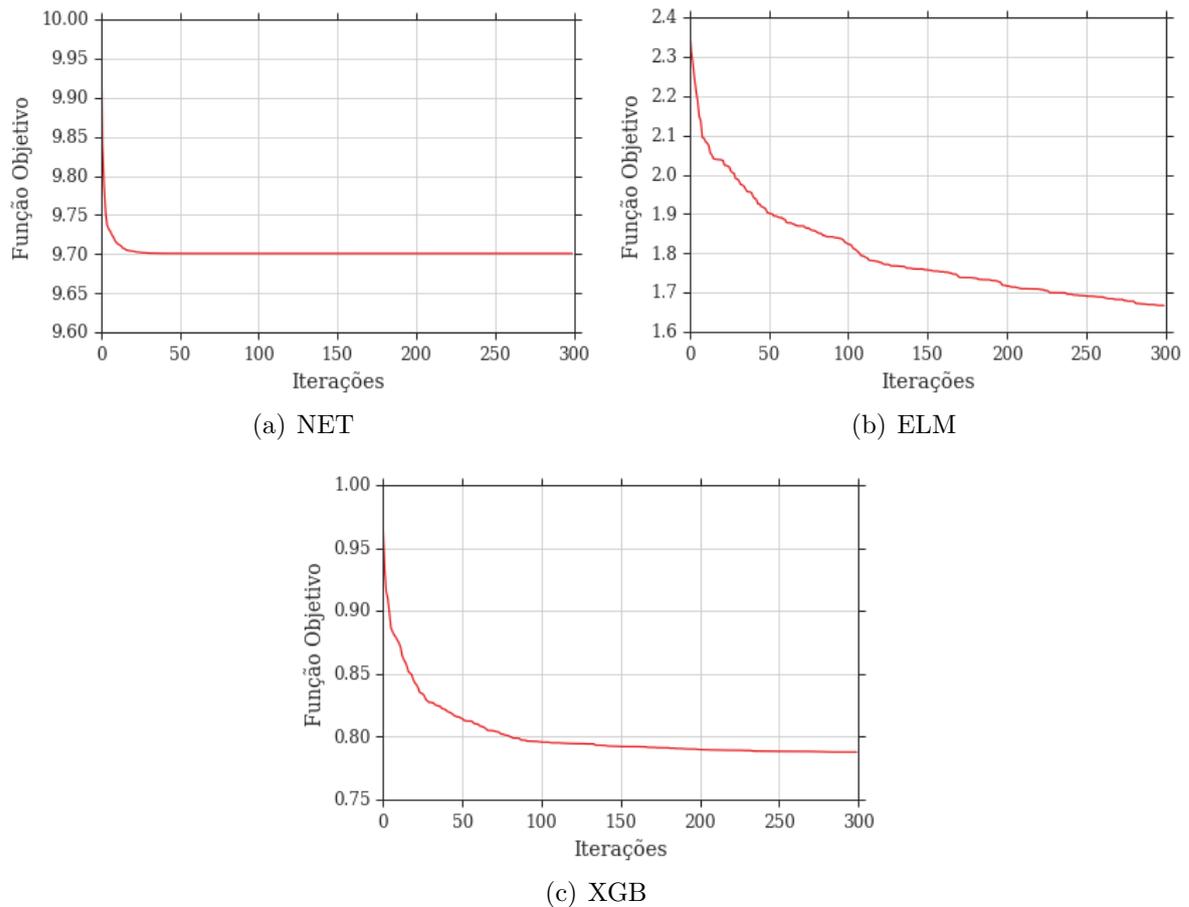


Figura 12 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de aquecimento da BD1.

Fonte: Elaborada pelo autor.

pelo algoritmo de otimização nas 30 execuções, enquanto para o parâmetro `alpha` os valores atribuídos foram distribuídos no intervalo  $[0,124; 0,134]$  com média 0,128 e mediana 0,128 sem *outliers*, indicando que valores próximos de 0,128 melhoram o desempenho preditivo do método NET para as cargas de resfriamento da BD1.

As distribuições dos hiperparâmetros escolhidos pelo algoritmo evolutivo para o método ELM são apresentadas na Figura 14. Os valores para `rbf_width` se distribuíram no intervalo  $[0,064; 9,996]$ , com média 1,378, mediana 0,182 e 1 *outlier*. Os valores de `alpha` se distribuíram em  $[0,001; 0,941]$ , com média 0,293 e mediana 0,122, sem *outliers* e contemplando a maior parte do intervalo de possibilidades  $[0; 1]$ .

A Figura 15 apresenta as distribuições dos parâmetros `colsample_bytree`, `learning_rate`, `subsample`, (b) a distribuição do parâmetro `n_estimators`, (c) e (d) as frequências dos parâmetros `min_child_weight` e `max_depth` utilizados no método XGB para a predição das cargas de resfriamento. O parâmetro `colsample_bytree` apresentou distribuição no intervalo  $[0,382; 0,604]$ , com média 0,475 e mediana 0,452, sem *outliers*. O `subsample` apresentou distribuição no intervalo  $[0,812; 0,995]$ , com média 0,475 e

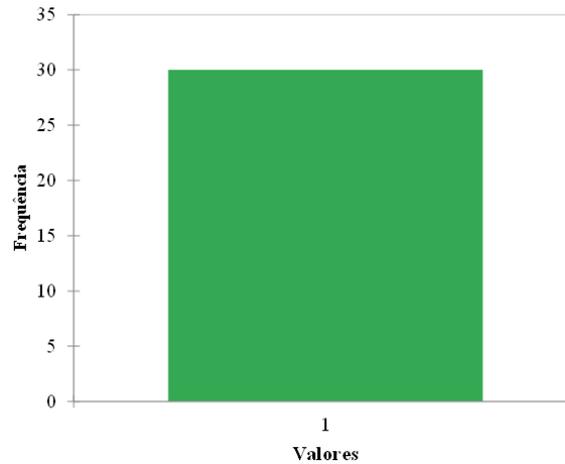


Figura 13 – Distribuição de frequências para o parâmetro `l1_ratio` utilizado no método NET para predição das cargas de resfriamento da BD1.

Fonte: Elaborada pelo autor.

mediana 0,452, enquanto o `learning_rate` se distribuiu no intervalo  $[0,150; 0,978]$ , com média 0,463 e mediana 0,439. O `n_estimators` apresentou distribuição concentrada em valores próximos do limite superior utilizado pelo algoritmo evolutivo, possuindo valores no intervalo  $[189,000; 894,000]$ , com média 769,933, mediana 829,500 e 1 valor *outlier*. Para `min_child_weight` o valor 2 apresentou maior frequência, sendo retornado em 17 das 39 execuções, enquanto para `max_depth` as frequências se distribuíram no intervalo  $[6; 29]$ , com a maior frequência ocorrendo no valor 7.

A Figura 16 apresenta os gráficos de convergência da função objetivo para os métodos testados obtidos através da média dos valores retornados nas 30 execuções independentes. O método NET converge para o valor final de função objetivo em torno da décima quinta iteração, enquanto os métodos ELM e XGB alcançam essa convergência posteriormente. Para o método ELM é possível notar que os valores de função objetivo diminuem lentamente, não sendo possível atestar convergência para este método com o número de iterações máximo igual a 300. O algoritmo XGB alcança valores convergentes próximo da iteração 225.

Para analisar as diferenças entre os métodos testados, inicialmente é necessário analisar a normalidade das distribuições das métricas estatísticas ao longo das execuções independentes. Para isso, foi aplicado os testes de Shapiro-Wilk e Lilliefors, que possuem como objetivo analisar a normalidade das distribuições. Para cada um dos métodos e cada uma das métricas, os valores retornados pelos testes que forem maiores que 0,05 indicam normalidade de distribuição na métrica de erro.

A Tabela 8 apresenta os p-valores calculados pelo método de Shapiro-Wilk, onde os itens em **negrito** indicam a existência de normalidade. Porém, ainda se faz necessário considerar os resultados retornados pelo teste de Lilliefors apresentados na Tabela 9. Desse

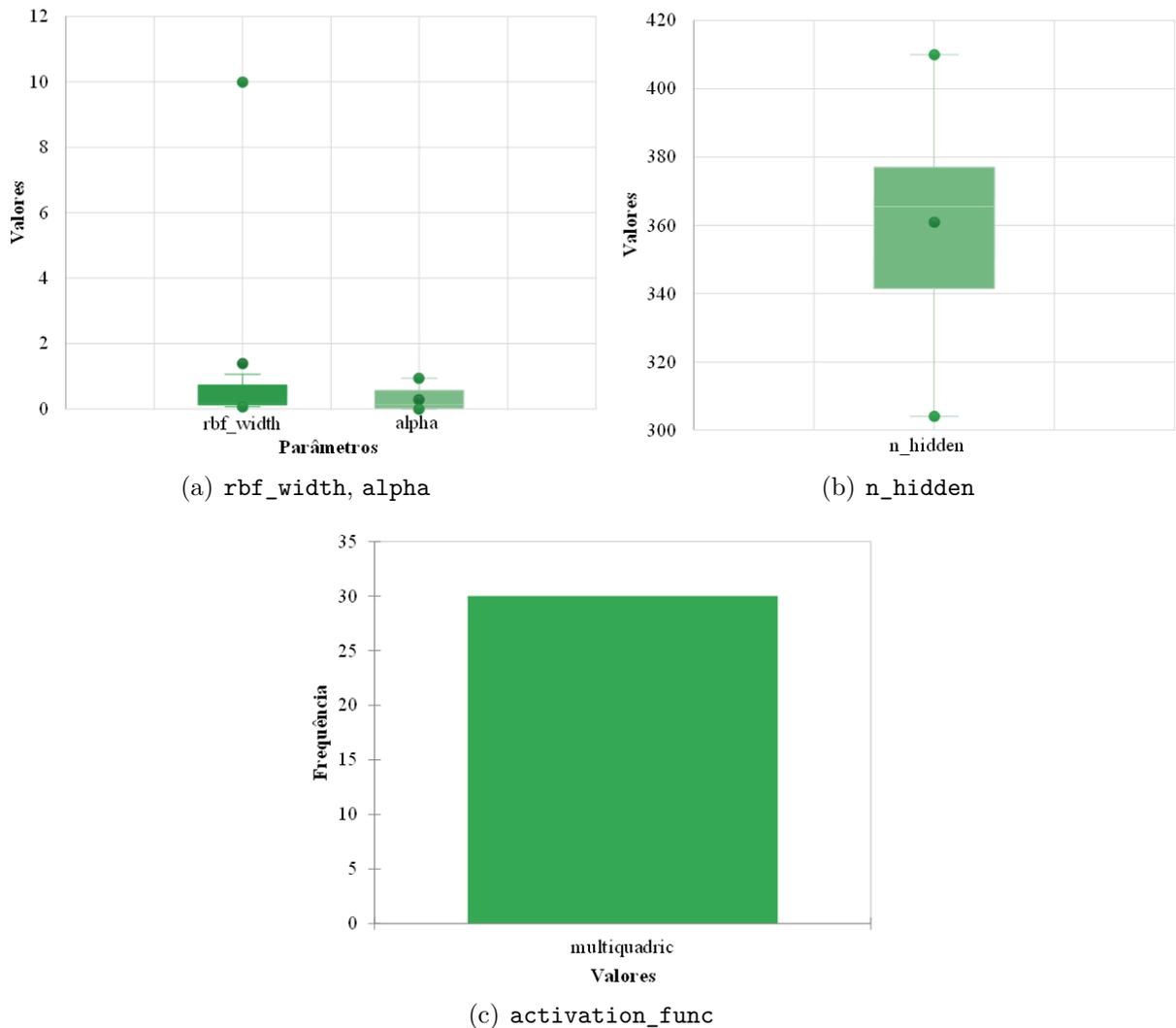


Figura 14 – Distribuições dos parâmetros `alpha`, `rbf_width`, `n_hidden` e `activation_func` utilizados no método ELM para a predição das cargas de resfriamento da BD1.

Fonte: Elaborada pelo autor.

modo, a indicação de normalidade retornada pelos dois testes indica a possibilidade de uso de testes estatísticos paramétricos para verificar as diferenças entre os métodos. No caso de não ser possível atestar normalidade, testes não paramétricos devem ser aplicados [55].

A Tabela 10 apresenta um resumo dos resultados dos testes estatísticos de Shapiro-Wilk e Lilliefors. Para o uso de testes paramétricos em dada métrica, é necessário que todos os métodos testados apresentem distribuição normal. Dessa forma, através da análise da Tabela 10 é possível verificar que será necessária a aplicação de testes não paramétricos, pois nenhuma das métricas apresentou distribuições normais nos três métodos de aprendizado de máquina de forma concomitante.

O teste de Kruskal-Wallis realiza a comparação entre as médias das métricas obtidas em cada um dos métodos com o objetivo de verificar se os modelos possuem diferenças estatisticamente significantes. A Tabela 11 apresenta os resultados dos p-valores retornados

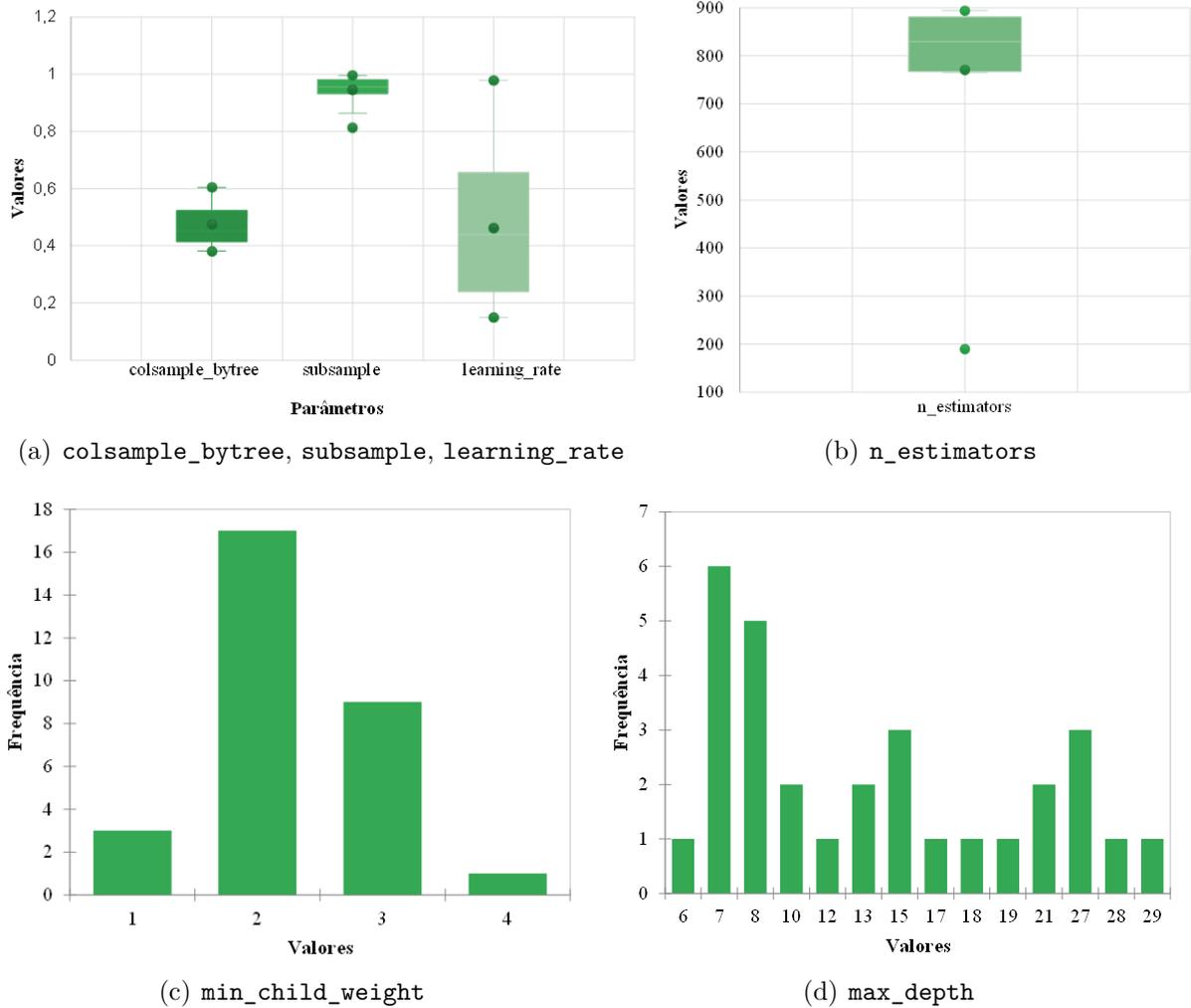


Figura 15 – Distribuições dos parâmetros colsample\_bytree, learning\_rate, subsample, n\_estimators, min\_child\_weight e max\_depth utilizados no método XGB para a predição das cargas de resfriamento da BD1.

Fonte: Elaborada pelo autor.

Tabela 8 – Resultados dos p-valores para o teste de Shapiro-Wilk para as métricas referentes aos resultados da BD1. Em negrito são apresentados os valores que indicam a existência de normalidade.

Saída	Métrica	NET	ELM	XGB
HL	MAE	0,000	0,008	0,031
	MAPE	<b>0,075</b>	<b>0,222</b>	<b>00,202</b>
	R <sup>2</sup>	<b>0,300</b>	0,010	<b>0,202</b>
	RMSE	<b>0,304</b>	0,013	<b>0,311</b>
CL	MAE	<b>0,746</b>	<b>0,162</b>	<b>0,088</b>
	MAPE	<b>0,103</b>	<b>0,781</b>	<b>0,717</b>
	R <sup>2</sup>	<b>0,144</b>	<b>0,302</b>	0,000
	RMSE	<b>0,145</b>	<b>0,268</b>	0,000

Fonte: Elaborada pelo autor.

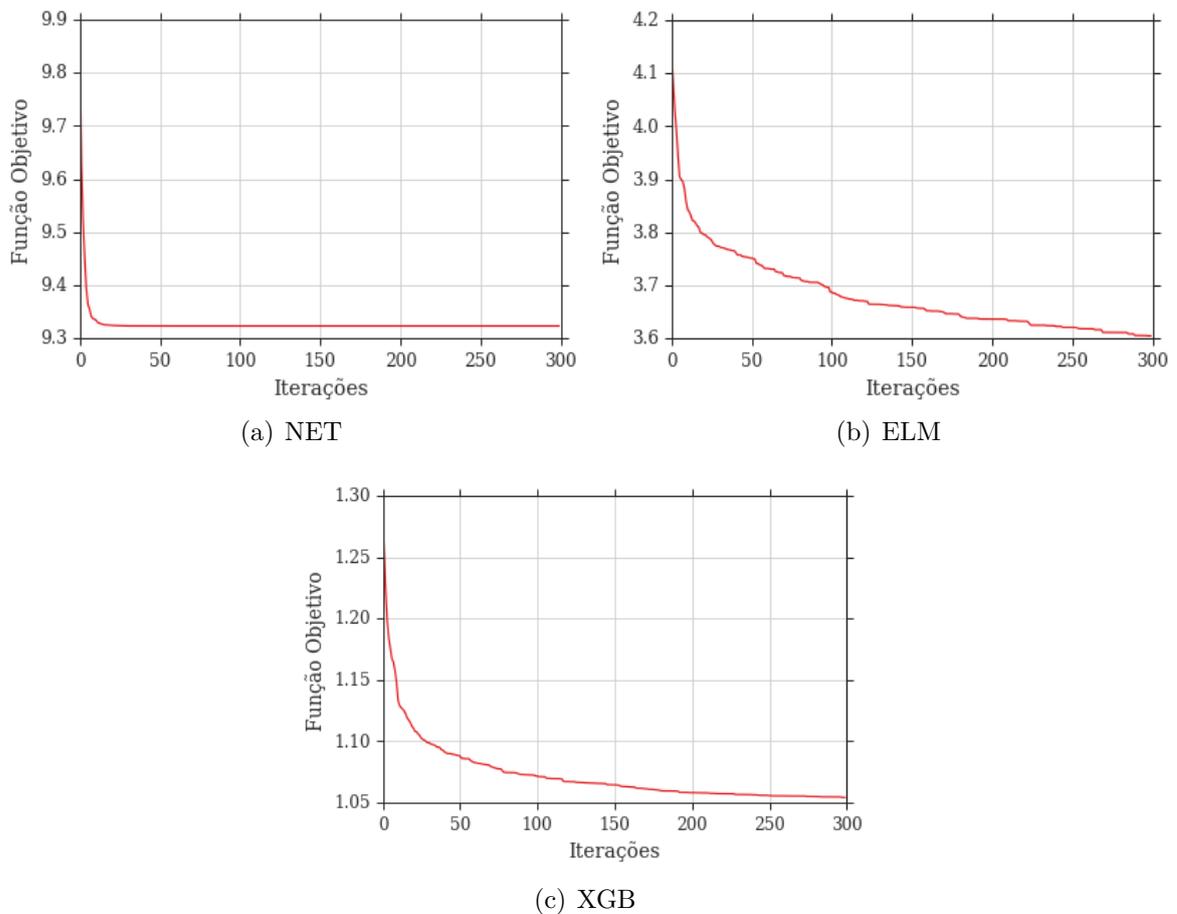


Figura 16 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de resfriamento da BD1.

Fonte: Elaborada pelo autor.

Tabela 9 – Resultados dos p-valores para o teste de Lilliefors para as métricas referentes aos resultados da BD1. Em negrito são apresentados os valores que indicam a existência de normalidade.

Saída	Métrica	NET	ELM	XGB
HL	MAE	0,000	0,018	0,019
	MAPE	<b>0,200</b>	<b>0,200</b>	<b>0,150</b>
	R <sup>2</sup>	<b>0,200</b>	0,037	<b>0,150</b>
	RMSE	<b>0,200</b>	0,037	<b>0,198</b>
CL	MAE	<b>0,200</b>	<b>0,200</b>	<b>0,200</b>
	MAPE	0,037	<b>0,200</b>	<b>0,200</b>
	R <sup>2</sup>	<b>0,200</b>	<b>0,200</b>	0,000
	RMSE	<b>0,200</b>	<b>0,200</b>	0,000

Fonte: Elaborada pelo autor.

pelo teste não paramétrico de Kruskal-Wallis para as métricas obtidas a partir da BD1. Pode-se observar que o  $p\text{-valor} < 0,0001$  foi calculados para as métricas MAE, MAPE, R<sup>2</sup> e MSE, sendo este menor que o nível de significância fixado em 0.05, o teste indica que os métodos de aprendizado de máquina possuem médias diferentes.

Tabela 10 – Resumo dos resultados dos testes de Shapiro e Lilliefors sobre a normalidade das distribuições dos resultados das métricas avaliadas para a BD1.

Saída	Métrica	NET	ELM	XGB
HL	MAE	-	-	-
	MAPE	normal	normal	-
	R <sup>2</sup>	normal	-	normal
	RMSE	normal	-	normal
CL	MAE	normal	normal	normal
	MAPE	-	normal	normal
	R <sup>2</sup>	normal	normal	-
	RMSE	normal	normal	-

Fonte: Elaborada pelo autor.

Tabela 11 – Resultados dos p-valores retornados pelo teste não paramétrico de Kruskal-Wallis para as métricas obtidas a partir da BD1.

Métrica	p-valor	
	HL	CL
MAE	<0,0001	<0,0001
MAPE	<0,0001	<0,0001
R <sup>2</sup>	<0,0001	<0,0001
RMSE	<0,0001	<0,0001

Fonte: Elaborada pelo autor.

O teste de Dunn, utilizado após o teste de Kruskal-Wallis, determina se as médias individuais são significativamente diferentes das outras médias. Além de mostrar a existência de diferenças significantes, o teste determina onde a diferença existe. A Tabela 12 apresenta as diferenças entre os pares de métodos e os p-valores para as cargas de aquecimento e resfriamento considerando a métrica MAPE. O teste de Dunn foi performado apenas para a métrica MAPE por conta da constatação através de testes preliminares, do fornecimento as mesmas informações encontradas através do MAPE para os resultados das outras métricas. A partir da Tabela 12 pode-se notar que os métodos testados não produzem resultados similares para as cargas de aquecimento (HL) e resfriamento (CL), pois todos os p-valores apresentados são menores que o nível de significância fixado em 0,05. O teste indicou uma diferença de 30 entre os métodos NET e ELM, e ELM e XGB, ocorrendo a maior diferença entre os métodos XGB e NET, -60.

Além dos p-valores e das diferenças, o teste de Dunn retorna a soma e a média das classificações dos dados utilizados no teste. Essa classificação se refere ao ordenamento dos valores de MAPES de todos os métodos juntos de forma crescente. Após esse ordenamento, as posições gerais dos MAPES são somadas de acordo com o método ao qual se referem e a média dessas posições é calculada. Dessa forma, é possível verificar o método que obteve desempenho superior aos outros através da análise das médias das classificações, criando um tipo de ordenamento para o desempenho preditivo dos modelos. A Tabela 13 apresenta os resultados das somas e médias das classificações obtidas através do teste

Tabela 12 – Resultados dos p-valores e diferenças obtidos através do teste de Dunn para a métrica MAPE da BD1.

Saída	Métodos	Diferença	p-valor
HL	NET ELM	30	<0,0001
	ELM XGB	30	<0,0001
	XGB NET	-60	<0,0001
CL	NET ELM	30	<0,0001
	ELM XGB	30	<0,0001
	XGB NET	-60	<0,0001

Fonte: Elaborada pelo autor.

de Dunn para a métrica MAPE. A menor média de classificação foi obtida pelo método XGB, concluindo-se que, tanto para as cargas de aquecimento (HL) quanto para as de resfriamento (CL), o XGB se mostrou estatisticamente superior aos demais. Além disso, pode ser observado também que o ELM é estatisticamente superior ao método NET.

Tabela 13 – Resultados das somas e médias das classificações obtidas através do teste de Dunn para a métrica MAPE da BD1.

Saída	Métodos	Soma das Classificações	Médias das Classificações
HL	NET	2265	75,5
	ELM	1365	45,5
	XGB	465	15,5
CL	NET	2265	75,5
	ELM	1365	45,5
	XGB	465	15,5

Fonte: Elaborada pelo autor.

As Tabelas 14 e 15 apresentam as métricas estatísticas obtidas pelos melhores modelos (ao longo das 30 execuções) testados neste trabalho. Para fins de comparação, foram incluídos nas tabelas os resultados obtidos por outros modelos apresentados na literatura para a mesma base de dados utilizada. Como pode ser visto na Tabela 14, o modelo DE+XGB obteve resultados competitivos em relação a trabalhos anteriores. Na Tabela 15 podemos verificar que o acoplamento do DE ao XGB alcançou a melhor performance em todas as métricas calculadas, refletindo sua habilidade de capturar relações não lineares a partir dos dados.

As Figuras 27 e 28 apresentam a distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de aquecimento e resfriamento, respectivamente. O tempo computacional do algoritmo XGB se mostrou consideravelmente maior do que os modelos ELM e NET. Isso pode ser explicado pela complexidade do método e pelo tamanho do espaço de busca de parâmetros. Enquanto o DE otimiza 2 e 4 parâmetros para os métodos NET e ELM, respectivamente, o método XGB conta com a otimização de 6 parâmetros de forma simultânea, que, aliada a robustez do modelo, aumenta de forma

Tabela 14 – Cargas de aquecimento – Comparação dos resultados obtidos pelos modelos deste estudo na BD1 com a dados da literatura. Os melhores valores são apresentados em negrito.

Referência	Modelo	MAE (kW)	RMSE (kW)	MAPE (%)	R <sup>2</sup>
[81]	Florestas Aleatórias	0,510	–	2,180	–
[10]	Modelo <i>Ensemble</i>	0,340	0,460	–	1,000
[11]	Modelo <i>Ensemble</i>	0,236	0,346	–	<b>0,999</b>
[7]	Programação Genética	0,380	–	<b>0,430</b>	–
[15]	Florestas Aleatórias	0,315	<b>0,223</b>	1,350	0,998
[24]	Processos Gaussianos	0,262	0,404	1,395	0,998
[25]	<i>GB</i>	0,224	0,341	1,114	<b>0,999</b>
[1]	<i>XGB</i>	0,175	0,265	0,913	<b>0,999</b>
Modelo 1	DE+NET	2,202	3,178	10,157	0,901
Modelo 2	DE+ELM	0,329	0,329	1,573	0,998
Modelo 3	DE+XGB	<b>0,150</b>	0,243	0,753	<b>0,999</b>

Fonte: Adaptada de Al-Rakhami et al. [1].

Tabela 15 – Carga de resfriamento – Comparação dos resultados obtidos pelos modelos deste estudo na BD1 com a dados da literatura. Os melhores valores são apresentados em negrito.

Referência	Modelo	MAE (kW)	RMSE (kW)	MAPE (%)	R <sup>2</sup>
[81]	Florestas Aleatórias	1,420	–	4,620	–
[10]	Modelo <i>Ensemble</i>	0,680	0,970	–	0,990
[11]	Modelo <i>Ensemble</i>	0,890	1,566	–	0,986
[7]	Programação Genética	0,970	–	3,40	–
[15]	Florestas Aleatórias	0,565	0,837	2,342	0,991
[24]	Processos Gaussianos	0,486	0,763	1,924	0,994
[25]	<i>GB</i>	0,491	0,722	1,973	0,994
[1]	<i>XGB</i>	0,307	0,461	1,197	0,998
Modelo 1	DE+NET	2,384	3,387	9,416	0,873
Modelo 2	DE+ELM	0,861	1,222	3,434	0,986
Modelo 3	DE+XGB	<b>0,231</b>	<b>0,327</b>	<b>0,983</b>	<b>0,999</b>

Fonte: Adaptada de Al-Rakhami et al. [1].

expressiva seu tempo de execução em relação aos demais.

Para as cargas de aquecimento, o NET obteve distribuição de tempos no intervalo [1,849; 2,855] com média de 2,439 min e mediana 2,478 min, o ELM no intervalo [178,475; 354,333] com média de 257,382 min e mediana de 257,926 min, e o XGB no intervalo [482,457, 959,340] com média 743,286 min e mediana 776,939 min. Para as cargas de resfriamento, o NET obteve distribuição de tempos no intervalo [1,505; 2,262] com média de 2,193 min e mediana de 2,236 min, o ELM no intervalo [271,259; 364,771] com média de 332,119 min e mediana de 334,199 min, e o XGB no intervalo [226,109; 1029,750] com média de 771,933 e mediana de 809,457.

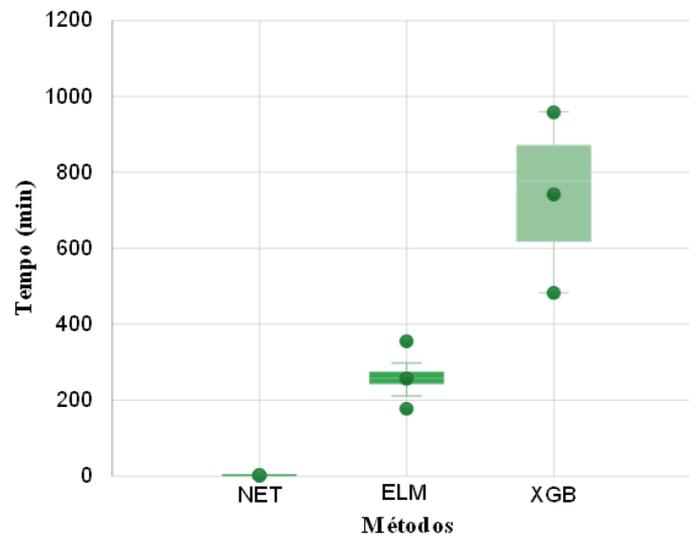


Figura 17 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de aquecimento da BD1.

Fonte: Elaborada pelo autor.

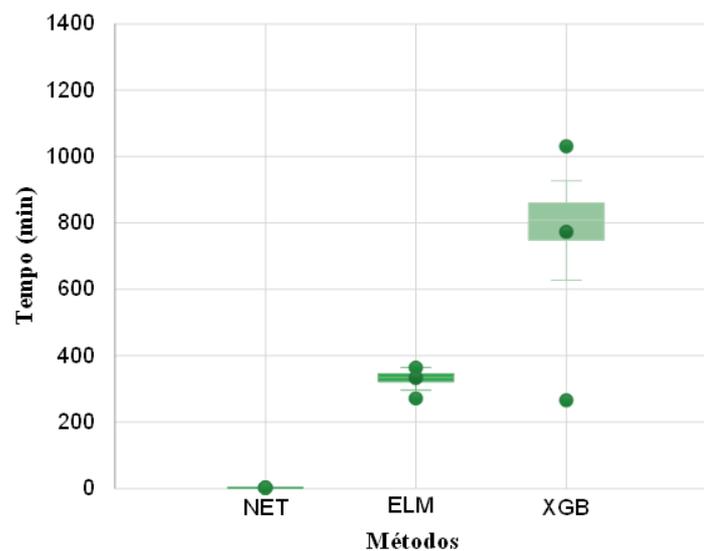


Figura 18 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de resfriamento da BD1.

Fonte: Elaborada pelo autor.

## 4.2 Base de Dados 2 (BD2)

A Tabela 16 apresenta um resumo das médias das métricas estatísticas obtidas ao longo das 30 execuções independentes para as cargas de aquecimento e resfriamento. No caso das métricas MAE, RMSE e MAPE os menores valores indicam melhor desempenho e para o coeficiente  $R^2$ , os melhores modelos devem apresentar seu valor mais próximo de 1. Através da observação das métricas, é possível verificar que o método XGB alcançou melhores resultados, tanto para as cargas de aquecimento quanto para as de resfriamento. No caso das cargas de aquecimento, o método ELM obteve valores ligeiramente próximos dos alcançados pelo XGB nas métricas RMSE e  $R^2$ . Para as cargas de resfriamento, é possível verificar que o método NET obteve valores próximos dos alcançados pelo ELM para as métricas MAE, RMSE e MAPE. Para todas as métricas apresentadas é possível notar um comportamento comum entre os três métodos testados, sendo este o melhor desempenho médio na previsão das cargas de aquecimento.

Tabela 16 – Média das métricas estatísticas obtidas ao longo das 30 execuções dos algoritmos para as cargas de aquecimento (HL) e resfriamento (CL) para BD2. Os melhores valores são mostrados em negrito e os desvios padrão entre parênteses.

Saída	Modelo	Métrica			
		MAE	RMSE	MAPE	$R^2$
HL	NET	5,037(0,116)	11,351(0,051)	21,470(0,498)	0,448(0,069)
	ELM	0,309(0,003)	0,429(0,005)	1,774(0,019)	0,996( $7 \times 10^{-5}$ )
	XGB	<b>0,081(0,003)</b>	<b>0,146(0,010)</b>	<b>0,418(0,014)</b>	<b>1,000(<math>5 \times 10^{-5}</math>)</b>
CL	NET	3,247(0,001)	5,362(0,002)	16,571(0,005)	0,453(0,001)
	ELM	3,097(0,043)	6,040(0,092)	15,741(0,146)	0,794(0,041)
	XGB	<b>0,596(0,013)</b>	<b>1,322(0,050)</b>	<b>6,903(0,106)</b>	<b>0,997(0,001)</b>

Fonte: Elaborada pelo autor.

A Figura 19 apresenta as distribuições dos parâmetros `alpha` e `l1_ratio` utilizados no método NET para a previsão das cargas de aquecimento. O parâmetro `alpha` obteve distribuição no intervalo  $[0,298; 0,311]$  com média 0,306 e mediana 0,305, enquanto o parâmetro `l1_ratio` se distribuiu no intervalo  $[0,978; 0,987]$  com média e mediana de 0,983.

As distribuições dos hiperparâmetros escolhidos pelo algoritmo evolutivo para o método ELM são apresentadas na Figura 20. Os valores de `alpha` se distribuíram em  $[0,430; 0,685]$ , com média 0,590 e mediana 0,599, sem *outliers*. O algoritmo evolutivo retornou o valor 0,01 para `rbf_width` e escolheu a função *gaussian* para `activation_func` em todas as execuções. O parâmetro `n_hidden` obteve valores entre 480 e 498 para o número de neurônios na camada oculta.

A Figura 11 apresenta (a) as distribuições dos parâmetros `colsample_bytree` e `subsample`, (b) a distribuição do parâmetro `learning_rate`, (c), (d) e (e) as frequências dos parâmetros `min_child_weight`, `max_depth` e `n_estimators`, respectivamente, utili-

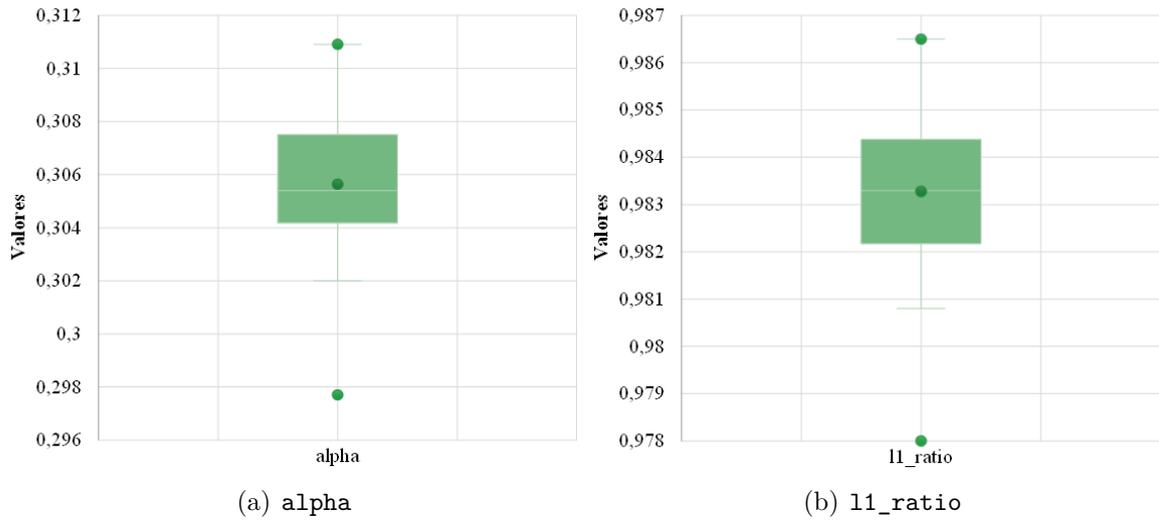


Figura 19 – Distribuições dos parâmetros `alpha` e `l1_ratio` utilizados no método NET para a previsão das cargas de aquecimento para a BD2.

Fonte: Elaborada pelo autor.

zados no método XGB para a predição das cargas de aquecimento. O hiperparâmetro `colsample_bytree` apresentou valores no intervalo  $[0,501; 0,937]$ , com média 0,775 e mediana 0,765. O `subsample` teve seus valores distribuídos entre  $[0,901; 0,986]$ , média 0,775 e mediana 0,765, enquanto o `learning_rate` apresentou valores em torno de 0,1, com distribuição entre  $[0,058; 0,187]$ , média 0,096 e mediana 0,080. O `n_estimators`, que possui valores entre 851 e 898, com a ocorrência da maior frequência em 898 escolhida em 10 de 30 execuções. Para o parâmetro `min_child_weight` o valor 10 foi selecionado em 16 de 30 execuções e para o `max_depth` o valor 26 obteve maior frequência, sendo selecionado em 6 execuções.

A Figura 22 apresenta os gráficos de convergência da função objetivo obtidos através da média dos valores retornados nas 30 execuções independentes para a previsão das cargas de aquecimento. Os métodos NET e ELM convergem para o valor final de função objetivo próximo da iteração 150, enquanto o método XGB alcança essa convergência posteriormente, próximo da iteração 225. Os três métodos testados alcançaram convergência dentro das 300 iterações realizadas.

A Figura 23 apresenta as distribuições dos parâmetros `alpha` e `l1_ratio` utilizados no método NET para a previsão das cargas de resfriamento. O parâmetro `alpha` obteve distribuição no intervalo  $[0,135; 0,281]$  com média 0,209 e mediana 0,198, enquanto o parâmetro `l1_ratio` se distribuiu no intervalo  $[0,904; 1,000]$  com média de 0,959 e mediana de 0,943.

As distribuições dos hiperparâmetros escolhidos pelo algoritmo evolutivo para o método ELM são apresentadas na Figura 24. Os valores de `alpha` para as cargas de resfriamento se distribuíram em  $[0,003; 0,901]$ , com média 0,565 e mediana 0,597, sem

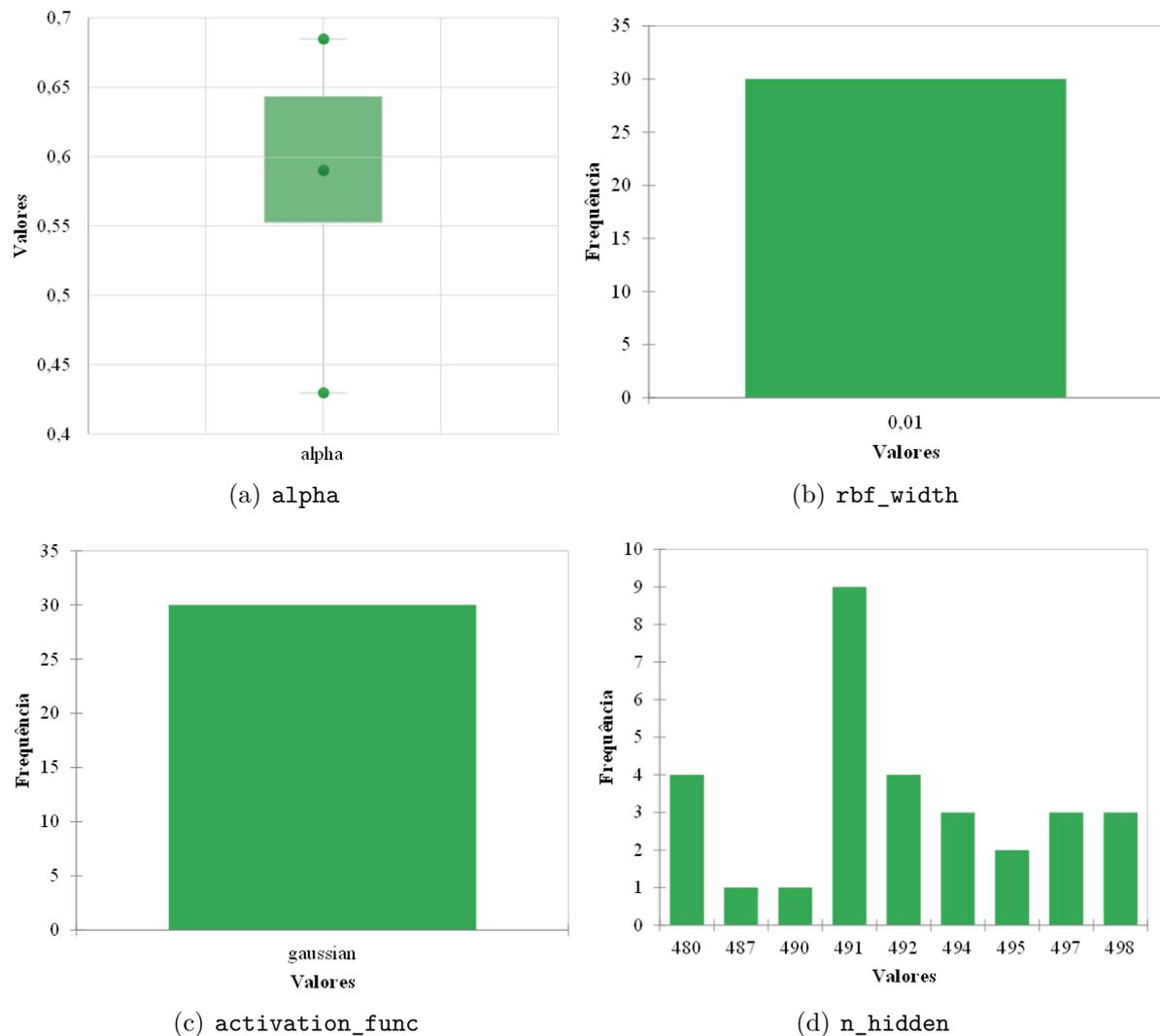
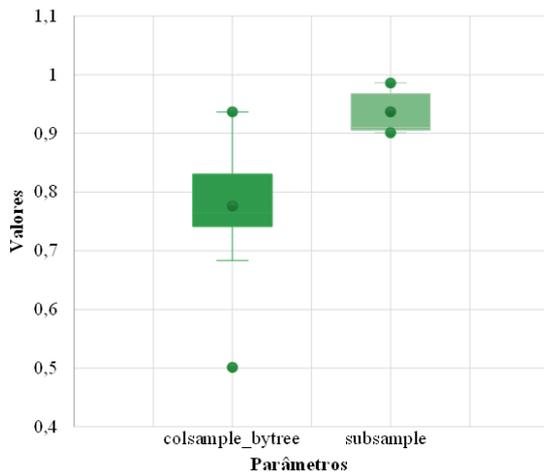


Figura 20 – Distribuições dos hiperparâmetros `alpha`, `rbf_width`, `activation_func` e `n_hidden` utilizados no método ELM para a previsão das cargas de aquecimento da BD2.

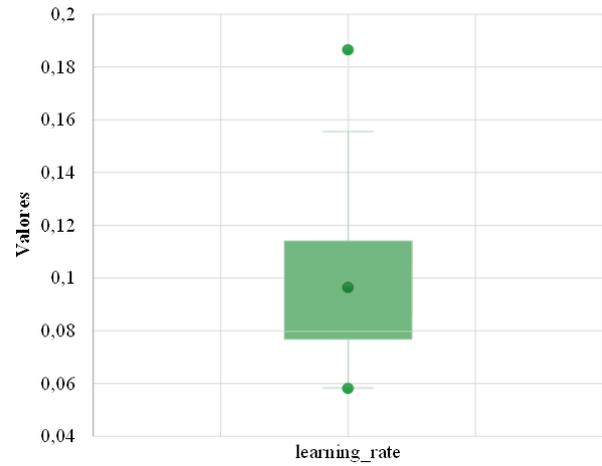
Fonte: Elaborada pelo autor.

*outliers*, enquanto o parâmetro `n_hidden` obteve valores no intervalo  $[4,000; 12,000]$  com média 6.800 e mediana 7,000. O algoritmo evolutivo retornou valores distribuídos no intervalo  $[0,351; 10]$  para o parâmetro `rbf_width`, com média 4,032 e mediana 3,639, e escolheu a função *multiquadric* para `activation_func` em todas as execuções.

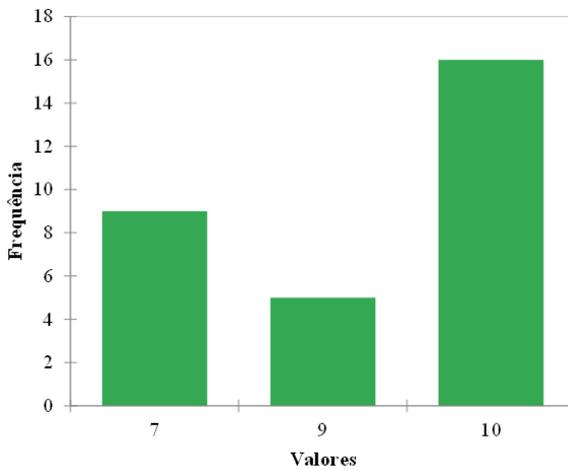
A Figura 25 apresenta (a) as distribuições dos parâmetros `colsample_bytree` e `subsample`, (b) a distribuição do parâmetro `learning_rate`, (c), (d) e (e) as frequências dos parâmetros `min_child_weight`, `max_depth` e `n_estimators`, respectivamente, utilizados no método XGB para a predição das cargas de resfriamento. O hiperparâmetro `colsample_bytree` apresentou valores no intervalo  $[0,845; 0,958]$ , com média 0,884 e mediana 0,870. O `subsample` teve seus valores distribuídos entre  $[0,644; 0,868]$ , média 0,696 e mediana 0,671, enquanto o `learning_rate` apresentou valores com distribuição entre  $[0,017; 0,034]$ , média 0,027 e mediana 0,028, ocupando uma pequena parte do intervalo



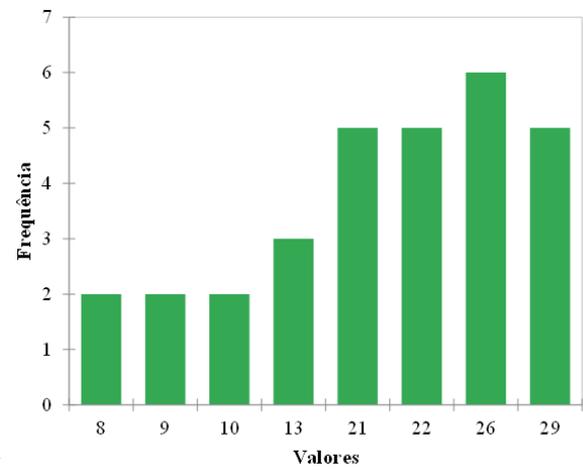
(a) colsample\_bytree, subsample



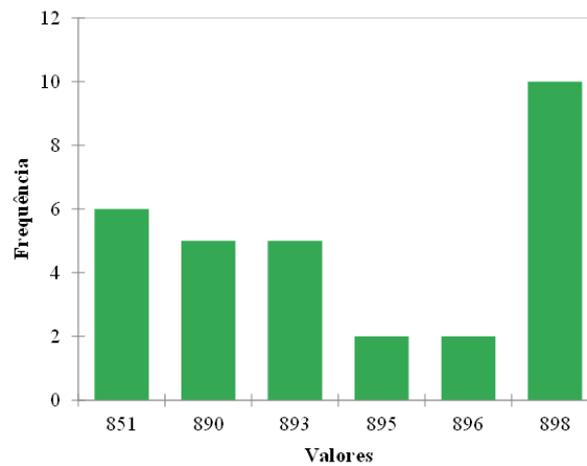
(b) learning\_rate



(c) min\_child\_weight



(d) max\_depth



(e) n\_estimators

Figura 21 – Distribuições dos parâmetros `colsample_bytree`, `subsample`, `learning_rate`, `min_child_weight`, `max_depth` e `n_estimators` utilizados no método XGB para a predição das cargas de aquecimento para a BD2.

Fonte: Elaborada pelo autor.

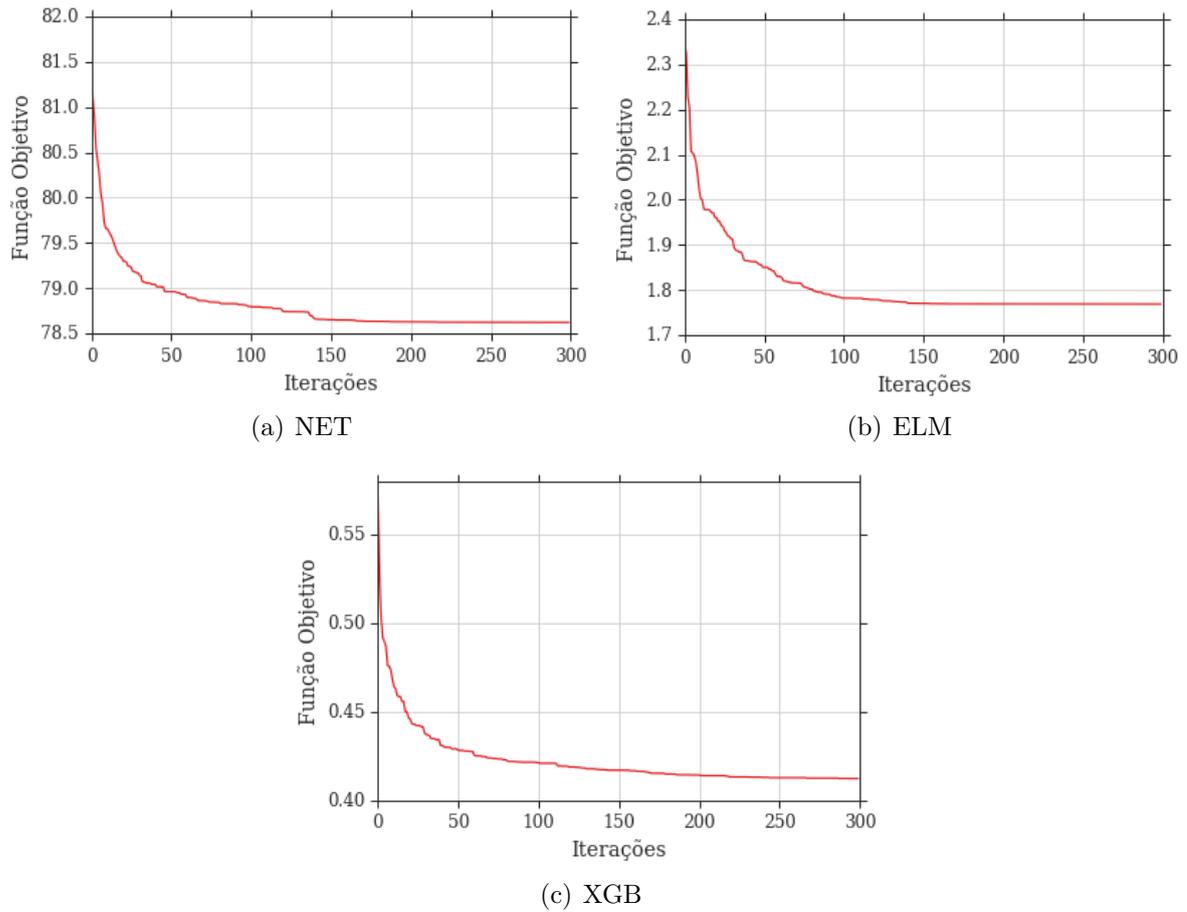


Figura 22 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de aquecimento da BD2.

Fonte: Elaborada pelo autor.

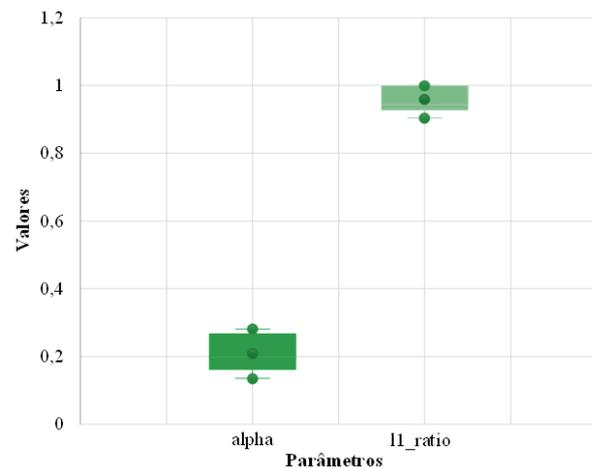


Figura 23 – Distribuições dos parâmetros alpha e l1\_ratio utilizados no método NET para a previsão das cargas de resfriamento para a BD2.

Fonte: Elaborada pelo autor.

de valores possíveis  $[0; 1]$ . Para o parâmetro `min_child_weight` o valor 2 foi selecionado em 26 de 30 execuções e para o `max_depth` o valor 27 obteve maior frequência, sendo

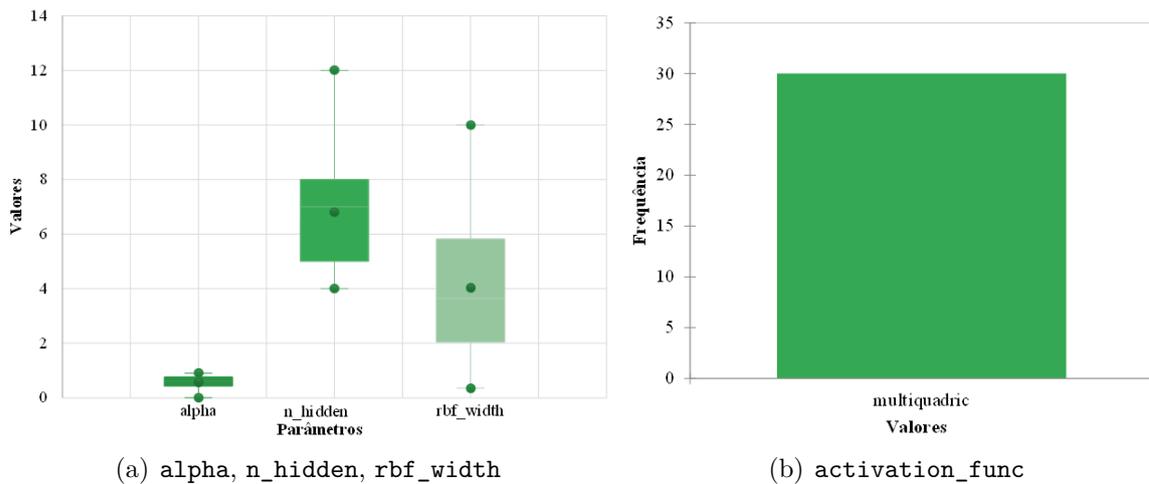


Figura 24 – Distribuições dos hiperparâmetros escolhidos pelo algoritmo evolutivo para o método ELM para a previsão das cargas de resfriamento da BD2.

Fonte: Elaborada pelo autor.

selecionado em 12 execuções. O  $n\_estimators$ , que alcançou valores entre 877 e 895, com a ocorrência da maior frequência em 895, escolhida em 8 das 30 execuções.

A Figura 26 apresenta os gráficos de convergência da função objetivo obtidos através da média dos valores retornados nas 30 execuções independentes para a previsão das cargas de resfriamento. O método NET converge para o valor final de função objetivo em torno da décima quinta iteração, enquanto os métodos ELM e XGB alcançam essa convergência posteriormente. Para o método ELM é possível notar que os valores de função objetivo diminuem lentamente, não sendo possível atestar convergência para este método com o número de iterações máximo igual a 300. O algoritmo XGB alcança valores convergentes próximo da iteração 225.

A Tabela 17 apresenta os p-valores calculados pelo método de Shapiro-Wilk, onde os itens em negrito indicam a existência de normalidade. Porém, ainda se faz necessário considerar os resultados retornados pelo teste de Lilliefors apresentados na Tabela 18. Desse modo, a indicação de normalidade retornada pelos dois testes indica a possibilidade de uso de testes estatísticos paramétricos para verificar as diferenças entre os métodos. No caso de não ser possível atestar normalidade, testes não paramétricos devem ser aplicados.

A Tabela 19 apresenta um resumo dos resultados dos testes estatísticos de Shapiro-Wilk e Lilliefors. Para o uso de testes paramétricos em dada métrica, é necessário que todos os métodos testados apresentem distribuição normal. Dessa forma, através da análise da Tabela 19 é possível verificar que será necessária a aplicação de testes não paramétricos, pois nenhuma das métricas apresentou distribuições normais nos três métodos de aprendizado de máquina de forma concomitante.

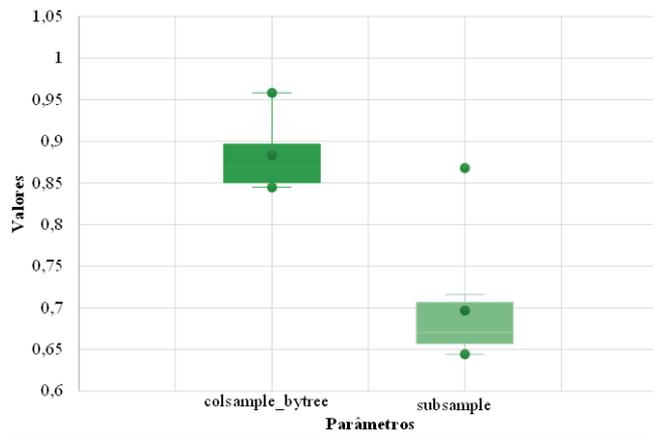
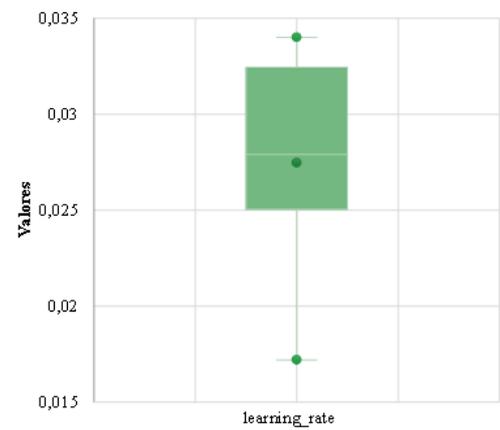
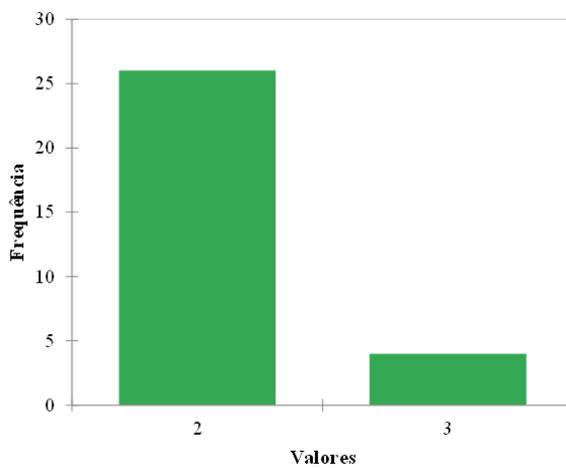
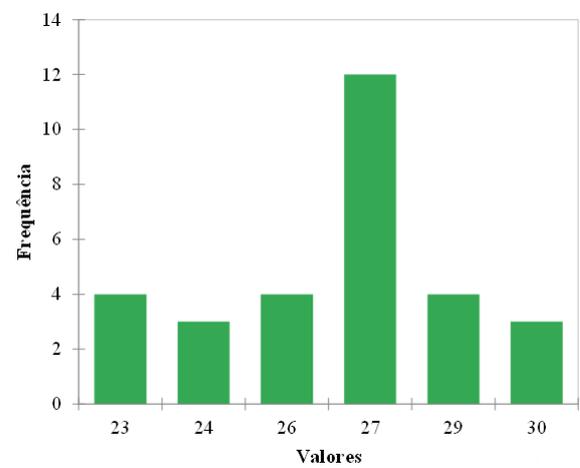
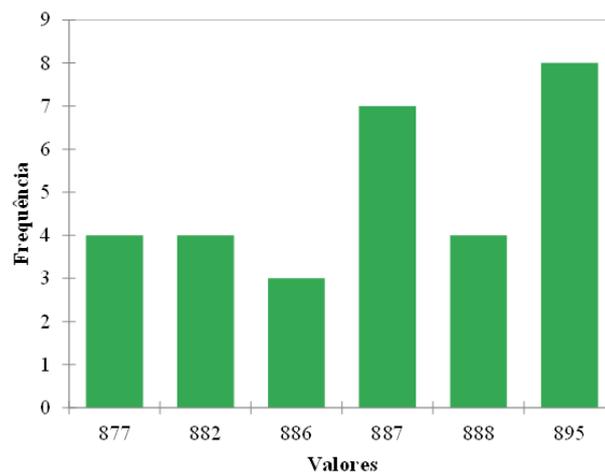
(a) `colsample_bytree`(b) `subsample`(c) `min_child_weight`(d) `max_depth`(e) `n_estimators`

Figura 25 – Distribuições dos parâmetros `colsample_bytree` e `subsample`, `learning_rate`, `min_child_weight`, `max_depth` e `n_estimators` utilizados no método XGB para a predição das cargas de resfriamento para a BD2.

Fonte: Elaborada pelo autor.

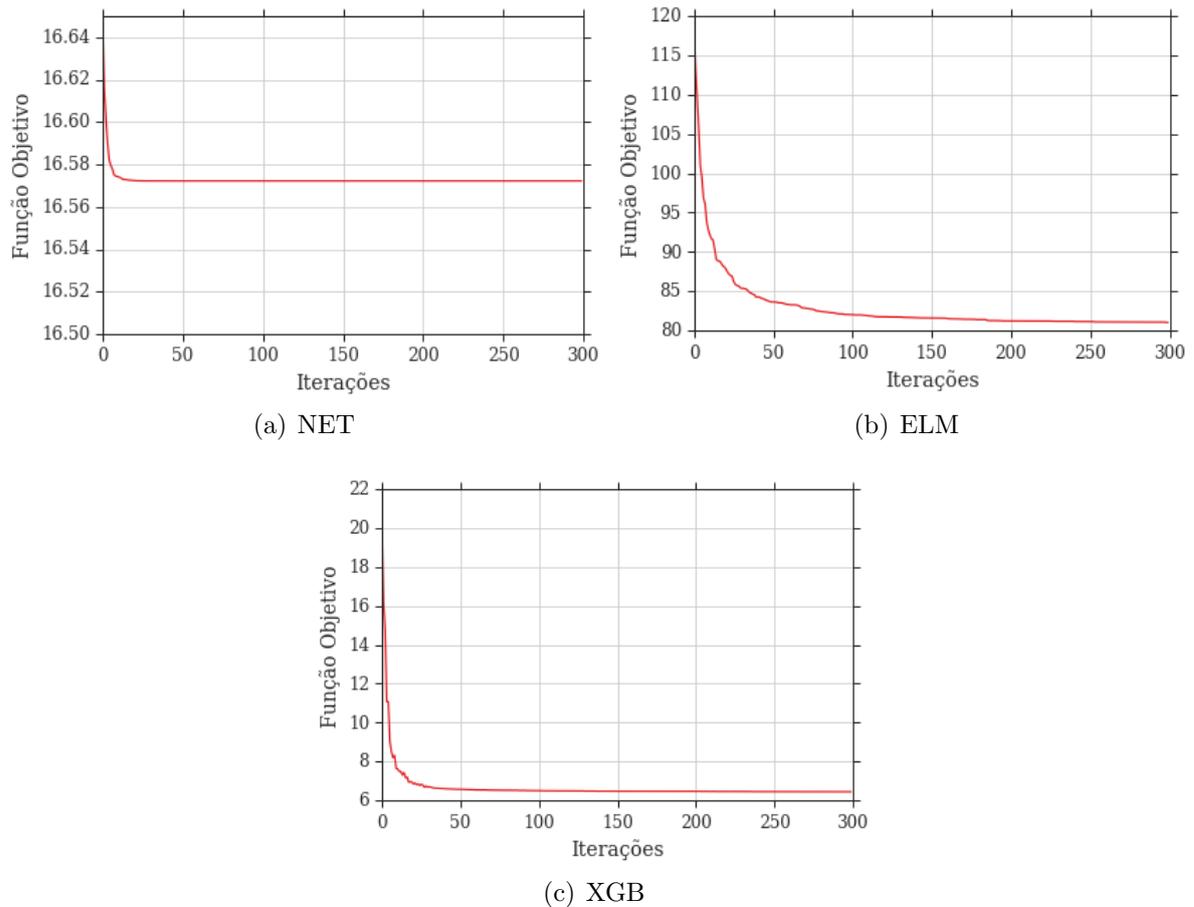


Figura 26 – Convergência média da função objetivo ao longo das 300 iterações do algoritmo evolutivo para as previsões das cargas de resfriamento da BD2.

Fonte: Elaborada pelo autor.

O teste de Kruskal-Wallis realiza a comparação entre as médias das métricas obtidas em cada um dos métodos com o objetivo de verificar se os modelos possuem diferenças estatisticamente significantes. A Tabela 20 apresenta os resultados dos p-valores retornados pelo teste não paramétrico de Kruskal-Wallis para as métricas obtidas a partir da BD2. Pode-se observar que o p-valor  $< 0,0001$  foi calculado para as métricas MAE, MAPE,  $R^2$  e RMSE, sendo este menor que o nível de significância fixado em 0,05, o teste indica que os métodos de aprendizado de máquina possuem médias diferentes.

A Tabela 21 apresenta as diferenças entre os pares de métodos e os p-valores para as cargas de aquecimento e resfriamento considerando a métrica MAPE. O teste de Dunn foi performado apenas para a métrica MAPE por conta da constatação através de testes preliminares, do fornecimento das mesmas informações encontradas através do MAPE para os resultados das outras métricas. A partir da Tabela 21 pode-se notar que os métodos testados não produzem resultados similares para as cargas de aquecimento (HL) e resfriamento (CL), pois todos os p-valores apresentados são menores que o nível de significância fixado em 0,05. Para as cargas de aquecimento (HL), o teste indicou uma

Tabela 17 – Resultados dos p-valores para o teste de Shapiro-Wilk para as métricas referentes aos resultados da BD2. Em negrito são apresentados os valores que indicam a existência de normalidade.

Saída	Métrica	NET	ELM	XGB
HL	MAE	<b>0,706</b>	0,010	0,001
	MAPE	0,004	0,003	0,001
	R <sup>2</sup>	<b>0,668</b>	0,005	0,009
	RMSE	<b>0,670</b>	0,006	0,005
CL	MAE	0,000	<b>0,603</b>	0,011
	MAPE	0,000	<b>0,483</b>	0,006
	R <sup>2</sup>	0,000	<b>0,649</b>	0,001
	RMSE	0,000	<b>0,660</b>	0,002

Fonte: Elaborada pelo autor.

Tabela 18 – Resultados dos p-valores para o teste de Lilliefors para as métricas referentes aos resultados da BD2. Em negrito são apresentados os valores que indicam a existência de normalidade.

Saída	Métrica	NET	ELM	XGB
HL	MAE	<b>0,200</b>	0,013	0,000
	MAPE	<b>0,142</b>	0,004	0,002
	R <sup>2</sup>	<b>0,200</b>	0,001	0,040
	RMSE	<b>0,200</b>	0,001	0,017
CL	MAE	0,000	<b>0,200</b>	0,032
	MAPE	0,000	<b>0,200</b>	0,005
	R <sup>2</sup>	0,000	<b>0,200</b>	0,001
	RMSE	0,000	<b>0,200</b>	0,001

Fonte: Elaborada pelo autor.

Tabela 19 – Resumo dos resultados dos testes Shapiro e Lilliefors sobre normalidade das distribuições dos resultados das métricas avaliadas.

Saída	Métrica	NET	ELM	XGB
HL	MAE	normal	-	-
	MAPE	-	-	-
	R <sup>2</sup>	normal	-	-
	RMSE	normal	-	-
CL	MAE	-	normal	-
	MAPE	-	normal	-
	R <sup>2</sup>	-	normal	-
	RMSE	-	normal	-

Fonte: Elaborada pelo autor.

diferença de 30,1 entre os métodos NET e ELM, e 29,8 entre ELM e XGB, ocorrendo a maior diferença entre os métodos XGB e NET, -59,9. Para as cargas de resfriamento (CL), o teste indicou uma diferença de 20 entre os métodos NET e ELM, e 40 entre ELM e XGB, ocorrendo a maior diferença entre os métodos XGB e NET, -60.

A Tabela 22 apresenta os resultados das somas e médias das classificações obtidas

Tabela 20 – Resultados dos p-valores retornados pelo teste não paramétrico de Kruskal-Wallis para as métricas obtidas a partir da BD2.

Métrica	p-valor	
	HL	CL
MAE	<0,0001	<0,0001
MAPE	<0,0001	<0,0001
R2	<0,0001	<0,0001
RMSE	<0,0001	<0,0001

Fonte: Elaborada pelo autor.

Tabela 21 – Resultados dos p-valores e diferenças obtidos através do teste de Dunn para a métrica MAPE da BD2.

Saída	Métodos	Diferença	p-valor
HL	NET ELM	30,1	<0,0001
	ELM XGB	29,8	<0,0001
	XGB NET	-59,9	<0,0001
CL	NET ELM	20	<0,0001
	ELM XGB	40	<0,0001
	XGB NET	-60	<0,0001

Fonte: Elaborada pelo autor.

através do teste de Dunn para a métrica MAPE. A menor média de classificação foi obtida pelo método XGB, concluindo-se que, tanto para as cargas de aquecimento (HL) quanto para as de resfriamento (CL), o XGB se mostrou estatisticamente superior aos demais. Além disso, pode ser observado também que o ELM é estatisticamente superior ao método NET.

Tabela 22 – Resultados das somas e médias das classificações obtidas através do teste de Dunn para a métrica MAPE da BD2.

Saída	Métodos	Soma das Classificações	Médias das Classificações
HL	NET	2262	75,4
	ELM	1368	45,6
	XGB	465	15
CL	NET	2265	75,5
	ELM	1365	45,5
	XGB	465	15,5

Fonte: Elaborada pelo autor.

A Tabela 23 apresenta as métricas estatísticas obtidas pelos melhores modelos (ao longo das 30 execuções) testados neste trabalho. Pode-se notar que o modelo DE+XGB obteve os melhores resultados em comparação com os outros dois métodos, tanto para as cargas de aquecimento quanto para as de resfriamento.

Jihad and Tahiri [37] executou um único modelo de Rede Neuronal na BD2 para a previsão das cargas de aquecimento e resfriamento. Após 25 milhões de iteração, seu

melhor modelo obteve  $R^2=0,986$  e  $MAPE=0,650$ . Em comparação com os resultados obtidos pelos modelos do XGB apresentados na Tabela 23, para a previsão do HL o DE+XGB obteve maior acurácia considerando as métricas  $R^2$  e MAPE, se comparado com o modelo de Jihad and Tahiri [37]. Para o CL, o  $R^2$  obtido pelo DE+XGB foi maior do que o encontrado pela referência, porém, para a métrica MAPE o resultado encontrado pelo melhor modelo desse trabalho para BD2 foi pior, pois apresentou um MAPE de 6,457 em comparação com o MAPE de 0,650 encontrado por Jihad and Tahiri [37]. O método XGB é conhecido por ser bastante robusto na captura de relações não lineares, de modo que possam ser necessárias mais iterações no algoritmo evolutivo e aumento da população de indivíduos, assim como modificações nos limites superiores e inferiores dos hiperparâmetros para melhorar o desempenho preditivo nesta base de dados de 5625 amostras.

Tabela 23 – Cargas de aquecimento e resfriamento – Comparação dos resultados obtidos pelos modelos testados na BD2. Os melhores valores são apresentados em negrito.

Saída	Modelo	Métrica			
		MAE	RMSE	MAPE	$R^2$
HL	DE+NET	3,280	8,509	18,716	0,528
	DE+ELM	0,305	0,424	1,737	0,997
	DE+XGB	<b>0,075</b>	<b>0,127</b>	<b>0,392</b>	<b>0,999</b>
CL	DE+NET	3,447	5,161	16,033	0,453
	DE+ELM	2,115	4,006	13,251	0,818
	DE+XGB	<b>0,586</b>	<b>1,299</b>	<b>60,457</b>	<b>0,997</b>

Fonte: Elaborada pelo autor.

As Figuras 27 e 28 apresentam a distribuição dos tempos em minutos referentes aos modelos de regressão testados para as cargas de aquecimento e resfriamento, respectivamente. O tempo computacional do algoritmo XGB se mostrou consideravelmente maior (cerca de 100 vezes maior) do que os modelos ELM e NET para as cargas de resfriamento. Isso pode ser explicado pela complexidade do método, pelo tamanho do conjunto de dados a ser processado e pelo tamanho do espaço de busca de parâmetros. Enquanto o DE otimiza 2 e 4 parâmetros para os métodos NET e ELM, respectivamente, o método XGB conta com a otimização de 6 parâmetros de forma simultânea, que, aliada a robustez do modelo e tamanho do conjunto de dados, aumenta de forma expressiva seu tempo de execução em relação aos demais. Para a previsão das cargas de aquecimento, os métodos ELM e XGB obtiveram tempo computacional elevado, podendo significar que ambos os métodos apresentaram dificuldades para capturar as relações não lineares entre os dados de entrada e a saída  $Y_2$ .

Para as cargas de aquecimento, o NET obteve distribuição de tempos no intervalo [1,673; 3,193] com média de 2,657 min e mediana 3,139 min, o ELM no intervalo [1615,309; 14978,667] com média de 12621,985 min e mediana de 13348,881 min, e o XGB no intervalo [6680,275; 13126,244] com média 10813,791 min e mediana 12091,638 min. Para as cargas

de resfriamento, o NET obteve distribuição de tempos no intervalo [4,362; 4,984] com média de 4,591 min e mediana de 4,547 min, o ELM no intervalo [53,582; 137,416] com média de 91,960 min e mediana de 92,851 min, e o XGB no intervalo [14611,085; 16739,006] com média de 15595,916 e mediana de 15668,548.

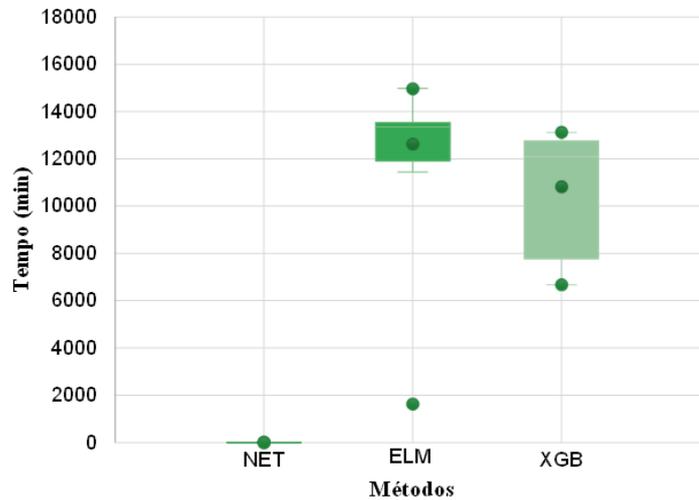


Figura 27 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de aquecimento da BD2.

Fonte: Elaborada pelo autor.

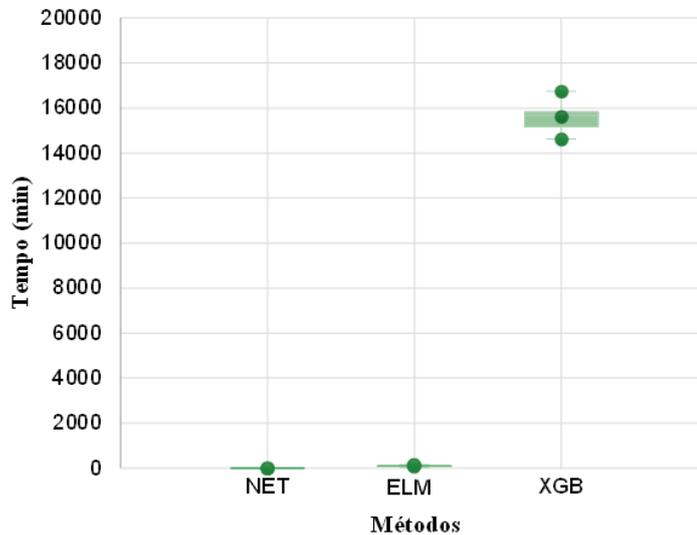


Figura 28 – Distribuição dos tempos em minutos referentes os modelos de regressão testados para as cargas de resfriamento da BD2.

Fonte: Elaborada pelo autor.

## 5 Conclusões e Trabalhos Futuros

No presente trabalho foi realizada a previsão das cargas de aquecimento e resfriamento para duas bases de dados disponíveis na literatura (BD1 e BD2). Para as essas bases, foram utilizados os modelos *Elastic Net*, ELM e XGB, acoplados ao algoritmo de Evolução Diferencial para a otimização dos hiperparâmetros dos modelos.

No caso da BD1, o uso do algoritmo evolutivo em conjunto com os métodos de aprendizado de máquina apresentou resultados satisfatórios em comparação com os dados da literatura. O modelo XGB alcançou os melhores resultados para todas as métricas testadas, considerando os três modelos testados no HL e CL. Além disso, obteve resultados competitivos com trabalhos recentes da literatura para as cargas de aquecimento e melhor performance em todas as métricas nas aproximações das cargas de resfriamento. Em relação ao tempo de execução, o XGB apresentou um custo computacional elevado em relação aos métodos testados. Desse modo, para aplicações práticas em grande escala que necessitem de resultados imediatos, o uso do método ELM pode ser suficiente, desde que sejam consideradas as métricas de erro, de modo que exista um balanço entre o erro cometido na previsão e o tempo computacional gasto.

Como trabalho futuro, para o modelo XGB, que apresentou um melhor desempenho geral na BD1, é proposta a aplicação de métodos de redução de dimensionalidade para a análise da importância de cada uma das variáveis de entrada, sendo possível assim reduzir o tempo de computação gasto. Também, é proposto o teste do método DE+ELM com um número maior de iterações para a otimização, a fim de analisar a convergência do método e verificar se menores valores de erro podem ser encontrados.

Para a previsão das cargas de aquecimento da BD2, o acoplamento dos modelos ELM e XGB ao algoritmo evolutivo DE apresentou resultados satisfatórios em comparação com os dados da literatura, sendo o XGB o método que conseguiu os melhores valores para as métricas estatísticas. No caso das cargas de resfriamento, o XGB apresentou um MAPE cerca de 10 vezes maior do que o encontrado na literatura, o que pode significar uma convergência prematura do método para um mínimo local. Em relação ao tempo computacional, nas cargas de aquecimento o ELM obteve custo computacional próximo ao do XGB, o que não ocorreu na previsão das cargas de resfriamento, onde o XGB apresentou um tempo de computação bastante elevado. Nos dois casos o método NET apresentou os menores tempos de computação por conta da simplicidade do modelo linear ao realizar a regressão nos dados, o que também resultou em um baixo desempenho em todas as métricas quando comparado aos modelos ELM e XGB.

Como trabalhos futuros, para analisar a ocorrência de convergência prematura na BD2, é proposta a aplicação de modelos de dinâmica caótica no algoritmo de Evolução Diferencial com o intuito de introduzir diversidade na população ao longo das iterações.

Além disso, técnicas de análise de componentes principais e de extração de características podem ser aplicadas para a redução da dimensionalidade do conjunto de dados de entrada e assim diminuir o tempo computacional gasto pelos métodos XGB e ELM. Será também avaliada a viabilidade de paralelização do código no intuito de diminuir o custo computacional.

Para as bases BD1 e BD2, é proposto como trabalho futuro o uso de *Automated Machine Learning* para, além de otimização dos hiperparâmetros, também gerenciar a busca automática dos métodos a serem utilizados em determinado problema. Além disso, o uso de *Automated Machine Learning* pode vir a proporcionar o ajuste de redes que retornem as previsões para aquecimento e resfriamento ao mesmo tempo, sem a necessidade de treino e teste de dois modelos separadamente.

## 6 Publicações Científicas

Trabalhos desenvolvidos durante o curso de Mestrado em Modelagem Computacional sem relação direta com o tema da dissertação.

1. G. G. Tavares; N. Resende; L. G. Fonseca; F. S. Bastos; G. H. C. Neves; G. L. Marques; M. Farage. Escaneamento tridimensional a laser como alternativa para aquisição de parâmetros de rugosidade de pavimentos - estudo de técnica computacional. Apresentado no *XXXVIII Iberian Latin American Congress on Computational Methods in Engineering*, Florianópolis, 2017 [77].
2. Y. Gorodestskaya; L. G. Fonseca; G. G. Tavares; C. B. M. Ribeiro. *Daily streamflow forecasting for Paraíba do Sul river using machine learning methods with hydrologic inputs*. Apresentado no XV Encontro Nacional de Inteligência Artificial e Computacional (ENIAC), São Paulo, 2018 [27].
3. Y. Gorodestskaya; G. G. Tavares; L. Goliatt ; C. B. M. Ribeiro. Comparação de métodos de aprendizado de máquina para a previsão de curto prazo de vazão do baixo curso do rio Paraíba do Sul. Apresentado no XXI Encontro Nacional de Modelagem Computacional e IX Encontro de Ciência e Tecnologia de Materiais (ENMC), Búzios, 2018 [26].
4. A. C. L. C. Queiroz; G.G. Tavares; F. O. Chaves; T. L. R. Sabino; L. G. Fonseca; P. V. Z. C. Goliatt. Método Computacional para segmentação automática do dossel de árvores de mangue a partir de dados de perfilamento tridimensional a laser. Apresentado no XXI Encontro Nacional de Modelagem Computacional e IX Encontro de Ciência e Tecnologia de Materiais (ENMC), Búzios, 2018 [62].
5. A. C. E. Polessa; G. G. T. Silva; L. G. Fonseca; P. C. Goliatt; R. M. Carvalho. Uso de dados abertos para pesquisa e desenvolvimento de cidades conectadas: um breve debate sobre o conceito de *smart cities*. Apresentado no XI Simpósio Nacional da ABCiber, Juiz de Fora, 2018 [60].

Trabalhos desenvolvidos durante o curso de Mestrado relacionados a este trabalho de conclusão.

1. G. G. Tavares; M. V. Ferraz; E. Vargas; L. Goliatt ; P. V. Z. C. Goliatt. Análise do desempenho térmico de edificação de balanço energético nulo via algoritmo evolutivo. Apresentado no XI Workshop do Departamento de Ciência da Computação da Universidade Federal de Juiz de Fora, Juiz de Fora, 2019.

2. T. L. Fonseca; Y. Gorodestskaya; G. G. T. Silva; C. B. M. Ribeiro; L. G. Fonseca. *A Gradient Boosting Model Optimized by a Genetic Algorithm for Short-term Riverflow Forecast*. Publicado na Revista Mundi Engenharia, Tecnologia e Gestão (ISSN: 2525-4782), v. 4, p. 151, 2019 [18].
3. L. Goliatt; P. V. Z. Capriles; G. G. Tavares. *Gradient boosting ensembles for predicting heating and cooling loads in building design*. Aceito para apresentação e publicação na *19th EPIA Conference on Artificial Intelligence*, Portugal, 2019.
4. G. G. Tavares; M. V. Ferraz; E. Vargas; V. C. Nobre; L. Goliatt; P. Capriles. *Thermal performance analysis of net-zero energy building using evolutionary algorithms*. Aceito para apresentação e publicação no XXII Encontro Nacional de Modelagem Computacional e X Encontro de Ciências e Tecnologia de Materiais, Juiz de Fora, 2019.

## REFERÊNCIAS

- [1] M. Al-Rakhami, A. Gumaedi, A. Alsanad, A. Alamri, and M. M. Hassan. An ensemble learning approach for accurate energy load prediction in residential buildings. *IEEE Access*, 7:48328–48338, 2019.
- [2] F. AlFaris, A. Juaidi, and F. Manzano-Agugliaro. Intelligent homes’ technologies to optimize the energy performance for the net zero energy home. *Energy and Buildings*, 153:262–274, 2017.
- [3] M. Ali and M. Pant. Improving the performance of differential evolution algorithm using cauchy mutation. *Soft Computing*, 15(5):991–1007, 2011.
- [4] H. Allcott and S. Mullainathan. Behavior and energy policy. *Science*, 327(5970):1204–1205, 2010.
- [5] F. Ascione, N. Bianco, G. M. Mauro, and D. F. Napolitano. Building envelope design: Multi-objective optimization to minimize energy consumption, global cost and thermal discomfort. application to different italian climatic zones. *Energy*, 174:359–374, 2019.
- [6] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- [7] M. Castelli, L. Trujillo, L. Vanneschi, and A. Popovič. Prediction of energy performance of residential buildings: A genetic programming approach. *Energy and Buildings*, 102:67–74, 2015.
- [8] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.
- [9] T. Chen, T. He, M. Benesty, V. Khotilovich, and Y. Tang. Xgboost: extreme gradient boosting. *R package version 0.4-2*, pages 1–4, 2015.
- [10] M.-Y. Cheng and M.-T. Cao. Accurately predicting building energy performance using evolutionary multivariate adaptive regression splines. *Applied Soft Computing*, 22:178–188, 2014.
- [11] J.-S. Chou and D.-K. Bui. Modeling heating and cooling loads by artificial intelligence for energy-efficient building design. *Energy and Buildings*, 82:437–446, 2014.
- [12] M. Claesen and B. De Moor. Hyperparameter search in machine learning. *arXiv preprint arXiv:1502.02127*, 2015.
- [13] S. Das and S. Sil. Kernel-induced fuzzy clustering of image pixels with an improved differential evolution algorithm. *Information Sciences*, 180(8):1237–1256, 2010.

- [14] A. Dinno. Nonparametric pairwise multiple comparisons in independent groups using dunn's test. *The Stata Journal*, 15(1):292–300, 2015.
- [15] G. R. Duarte, L. Fonseca, P. Goliatt, and A. Lemonge. Comparison of machine learning techniques for predicting energy loads in buildings. *Ambiente Construído*, 17(3):103–115, 2017.
- [16] B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.
- [17] D. Figueiredo Filho, F. Nunes, E. C. da Rocha, M. L. Santos, M. Batista, and J. A. Silva Júnior. O que fazer e o que não fazer com a regressão: pressupostos e aplicações do modelo linear de mínimos quadrados ordinários (mqo). *Revista Política Hoje*, 20(1), 2011.
- [18] T. Fonseca, Y. Gorodestskaya, G. Goulart, C. Ribeiro, and L. Goliatt. A gradient boosting model optimized by a genetic algorithm for short-term riverflow forecast. *Revista Mundi Engenharia, Tecnologia e Gestão*, 2019.
- [19] Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139, 1997.
- [20] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.
- [21] J. H. Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 29:1189–1232, 2001.
- [22] A. Ghasemi and S. Zahediasl. Normality tests for statistical analysis: a guide for non-statisticians. *International journal of endocrinology and metabolism*, 10(2):486, 2012.
- [23] L. Goliatt and M. Farage. An extreme learning machine with feature selection for estimating mechanical properties of lightweight aggregate concretes. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–7. IEEE, 2018.
- [24] L. Goliatt, P. Capriles, and G. R. Duarte. Modeling heating and cooling loads in buildings using gaussian processes. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–6. IEEE, 2018.
- [25] L. Goliatt, P. Capriles, and G. Tavares. Gradient boosting ensembles for predicting heating and cooling loads in building design. *Lecture Notes in Computer Science*, 2019.

- [26] Y. Gorodetskaya, G. G. Tavares, L. G. da Fonseca, and C. B. de Melo Ribeiro. Comparação de métodos de aprendizado de máquina para a previsão de curto prazo de vazão do baixo curso do rio paraíba do sul. 2018.
- [27] Y. Gorodetskaya, L. G. da Fonseca, G. G. Tavares, and C. B. de Melo Ribeiro. Daily streamflow forecasting for paraíba do sul river using machine learning methods with hydrologic inputs. In *Anais do XV Encontro Nacional de Inteligência Artificial e Computacional*, pages 162–173. SBC, 2018.
- [28] L. Guelman. Gradient boosting trees for auto insurance loss cost modeling and prediction. *Expert Systems with Applications*, 39(3):3659–3667, 2012.
- [29] Y. Guo, J. Wang, H. Chen, G. Li, J. Liu, C. Xu, R. Huang, and Y. Huang. Machine learning-based thermal response time ahead energy demand prediction for building heating systems. *Applied energy*, 221:16–27, 2018.
- [30] F. Harkouss, F. Fardoun, and P. H. Biwolé. Multi-objective optimization methodology for net zero energy buildings. *Journal of Building Engineering*, 16:57–71, 2018.
- [31] T. Hastie, R. Tibshirani, and J. Friedman. The elements of statistical learning: data mining, inference, and prediction, springer series in statistics, 2009.
- [32] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.
- [33] G. Huang, G.-B. Huang, S. Song, and K. You. Trends in extreme learning machines: A review. *Neural Networks*, 61(Supplement C):32 – 48, 2015. ISSN 0893-6080.
- [34] G.-B. Huang and C.-K. Siew. Extreme learning machine: Rbf network case. In *ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, 2004.*, volume 2, pages 1029–1036. IEEE, 2004.
- [35] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, et al. Extreme learning machine: a new learning scheme of feedforward neural networks. *Neural networks*, 2:985–990, 2004.
- [36] R. K. Jain, K. M. Smith, P. J. Culligan, and J. E. Taylor. Forecasting energy consumption of multi-family residential buildings using support vector regression: Investigating the impact of temporal and spatial monitoring granularity on performance accuracy. *Applied Energy*, 123:168–178, 2014.
- [37] A. S. Jihad and M. Tahiri. Forecasting the heating and cooling load of residential buildings by using a learning algorithm “gradient descent”, morocco. *Case studies in thermal engineering*, 12:85–93, 2018.
- [38] E. Jones, T. Oliphant, P. Peterson, et al. Scipy: Open source scientific tools for python. 2001.

- [39] A. Khalid, N. Javaid, M. Guizani, M. Alhussein, K. Aurangzeb, and M. Ilahi. Towards dynamic coordination among home appliances using multi-objective energy optimization for demand side management in smart buildings. *Ieee Access*, 6:19509–19529, 2018.
- [40] R. Kohavi et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai*, volume 14, pages 1137–1145, 1995.
- [41] B. Kriegler and R. Berk. Small area estimation of the homeless in los angeles: An application of cost-sensitive stochastic gradient boosting. *The Annals of Applied Statistics*, pages 1234–1255, 2010.
- [42] M. Kubat. A hyperrectangle-based method that creates rbf networks. In *Radial basis function networks 1*, pages 31–50. Physica Verlag Rudolf Liebing KG, 2001.
- [43] M. Kuhn and K. Johnson. *Applied predictive modeling*, volume 26. Springer, 2013.
- [44] S. Kumar, S. K. Pal, and R. Singh. A novel hybrid model based on particle swarm optimisation and extreme learning machine for short-term temperature prediction using ambient sensors. *Sustainable Cities and Society*, 49:101601, 2019.
- [45] M. Marcu, C. Stangaciu, A. Topirceanu, D. Volcinschi, and V. Stangaciu. Wireless sensors solution for energy monitoring, analyzing, controlling and predicting. In *International Conference on Sensor Systems and Software*, pages 1–19. Springer, 2010.
- [46] W. McKinney. pandas: a foundational python library for data analysis and statistics. *Python for High Performance and Scientific Computing*, 14, 2011.
- [47] Ministério de Minas e Energia do Brasil. Uso de ar condicionado no setor residencial brasileiro: Perspectivas e contribuições para o avanço em eficiência energética. Technical report, 2018.
- [48] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling. Deep learning for estimating building energy consumption. *Sustainable Energy, Grids and Networks*, 6:91–99, 2016.
- [49] Y. L. Murphey, J. Park, L. Kiliaris, M. L. Kuang, M. A. Masrur, A. M. Phillips, and Q. Wang. Intelligent hybrid vehicle power control—part ii: Online intelligent energy management. *IEEE Transactions on Vehicular Technology*, 62(1):69–79, 2012.
- [50] N. J. Nagelkerke et al. A note on a general definition of the coefficient of determination. *Biometrika*, 78(3):691–692, 1991.
- [51] F. Neri and V. Tirronen. Recent advances in differential evolution: a survey and experimental analysis. *Artificial Intelligence Review*, 33(1-2):61–106, 2010.

- [52] G. G. Neto and H. Hippert. *Gestão da Produção em Foco*, 31:155–163, 2019.
- [53] R. G. Newell, Y. Qian, and D. Raimi. Global energy outlook 2015. Technical report, National Bureau of Economic Research, 2016.
- [54] S. Nissen. Neural networks made simple. *Software*, 2(2):14–19, 2005.
- [55] D. Normando, L. Tjäderhane, and C. C. A. Quintão. A escolha do teste estatístico-um tutorial em forma de apresentação em powerpoint. *Dental Press J. Orthod*, 15(1): 101–106, 2010.
- [56] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
- [57] C. Peng. Calculation of a building’s life cycle carbon emissions based on Ecotect and building information modeling. *Journal of Cleaner Production*, 112, Part 1:453 – 465, 2016. ISSN 0959-6526. doi: <http://dx.doi.org/10.1016/j.jclepro.2015.08.078>.
- [58] L. Pérez-Lombard, J. Ortiz, and C. Pout. A review on buildings energy consumption information. *Energy and buildings*, 40(3):394–398, 2008.
- [59] J. Perlmutter and J. L. Myers. A comparison of two procedures for testing multiple contrasts. *Psychological Bulletin*, 79(3):181, 1973.
- [60] A. Polessa, G. Silva, L. G. Fonseca, P. C. Goliatt, and R. M. Carvalho. Uso de dados abertos para pesquisa e desenvolvimento de cidades conectadas: um breve debate sobre o conceito de *smart cities*. 2018.
- [61] K. Price, R. M. Storn, and J. A. Lampinen. *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media, 2006.
- [62] A. C. L. C. Queiroz, G. G. Tavares, F. O. Chaves, T. R. Sabino, L. G. da Fonseca, and P. C. Goliatt. Método computacional para segmentação automática do dossel de Árvores de mangue a partir de dados de perfilamento tridimensional a laser. 2018.
- [63] M. A. Ramli, H. Bouchekara, and A. S. Alghamdi. Optimal sizing of pv/wind/diesel hybrid microgrid system using multi-objective self-adaptive differential evolution algorithm. *Renewable energy*, 121:400–411, 2018.
- [64] C. Robinson, B. Dilkina, J. Hubbs, W. Zhang, S. Guhathakurta, M. A. Brown, and R. M. Pendyala. Machine learning approaches for estimating commercial building energy consumption. *Applied energy*, 208:889–904, 2017.

- [65] C. M. Saporetti, L. G. da Fonseca, E. Pereira, and L. C. de Oliveira. Machine learning approaches for petrographic classification of carbonate-siliciclastic rocks using well logs and textural information. *Journal of Applied Geophysics*, 155:217–225, 2018.
- [66] J. P. Sarkar, I. Saha, A. Sarkar, and U. Maulik. Improving modified differential evolution for fuzzy clustering. In *International Conference on Health Information Science*, pages 136–146. Springer, 2017.
- [67] B. Scholkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA, 2001. ISBN 0262194759.
- [68] I. Semanjski and S. Gautama. Smart city mobility application—gradient boosting trees for mobility prediction and analysis based on crowdsourced data. *Sensors*, 15(7):15974–15987, 2015.
- [69] R. Shanmugamani, M. Sadique, and B. Ramamoorthy. Detection and classification of surface defects of gun barrels using computer vision and machine learning. *Measurement*, 60:222–230, 2015.
- [70] F. M. Silveira et al. Análise do desempenho térmico de edificações residenciais ventiladas naturalmente: Nbr 15.575 e ashrae 55. 2014.
- [71] N. Soares, L. Goliatt, R. Santis, and E. P. de Aguiar. Máquinas de aprendizado extremo para a predição do volume de clones de eucaliptos. 11 2017. doi: 10.20906/CPS/CILAMCE2017-1298.
- [72] R. Storn and K. Price. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.
- [73] L. O. Sudbrack. Casa zero: diretrizes de projeto para casas pré-fabricadas de balanço energético nulo em Brasília. 2017.
- [74] J. G. T. Boermans. Economic effects of investing in ee in buildings – the beam2 model. *Background paper for EC Workshop on Cohesion policy*, 2011.
- [75] S. B. Taieb and R. J. Hyndman. A gradient boosting approach to the kaggle load forecasting competition. *International journal of forecasting*, 30(2):382–394, 2014.
- [76] J. Tapson, P. De Chazal, and A. van Schaik. Explicit computation of input weights in extreme learning machines. In *Proceedings of ELM-2014 Volume 1*, pages 41–49. Springer, 2015.

- [77] G. Tavares, L. Resende, Natália Goliatt, F. Bastos, Gabriel, G. Marques, and M. Farage. Escaneamento tridimensional a laser como alternativa para aquisição de parâmetros de rugosidade de pavimentos - estudo de técnica computacional. In *Anais do XXXVIII Iberian-Latin American Congress on Computational Methods in Engineering*, 2017.
- [78] J. Tayman and D. A. Swanson. On the validity of mape as a measure of population forecast accuracy. *Population Research and Policy Review*, 18(4):299–322, 1999.
- [79] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.
- [80] S. Touzani, J. Granderson, and S. Fernandes. Gradient boosting machine for modeling the energy consumption of commercial buildings. *Energy and Buildings*, 158:1533–1543, 2018.
- [81] A. Tsanas and A. Xifara. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, 49:560–567, 2012.
- [82] S. Van Der Walt, S. C. Colbert, and G. Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22, 2011.
- [83] A. Vargha and H. D. Delaney. The kruskal-wallis test and stochastic homogeneity. *Journal of Educational and behavioral Statistics*, 23(2):170–192, 1998.
- [84] J. Wang, L. Li, D. Niu, and Z. Tan. An annual load forecasting model based on support vector regression with differential evolution algorithm. *Applied Energy*, 94:65–70, 2012.
- [85] Z. Wang, R. S. Srinivasan, and J. Shi. Artificial intelligent models for improved prediction of residential space heating. *Journal of Energy Engineering*, 142(4):04016006, 2016.
- [86] R. T. Watson, M.-C. Boudreau, and A. J. Chen. Information systems and environmentally sustainable development: Energy informatics and new directions for the is community. *MIS quarterly*, 34(1), 2010.
- [87] C. Whitnall, E. Oswald, and L. Mather. An exploration of the kolmogorov-smirnov test as a competitor to mutual information analysis. In *International Conference on Smart Card Research and Advanced Applications*, pages 234–251. Springer, 2011.

- [88] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip, et al. Top 10 algorithms in data mining. *Knowledge and information systems*, 14(1):1–37, 2008.
- [89] Y. Xie, C. Zhu, W. Zhou, Z. Li, X. Liu, and M. Tu. Evaluation of machine learning methods for formation lithology identification: A comparison of tuning processes and model performances. *Journal of Petroleum Science and Engineering*, 160:182–193, 2018.
- [90] L. Yang, B.-J. He, and M. Ye. Application research of Ecotect in residential estate planning. *Energy and Buildings*, 72:195 – 202, 2014. ISSN 0378-7788.
- [91] W.-A. Yang, Q. Zhou, and K.-L. Tsui. Differential evolution-based feature selection and parameter optimisation for extreme learning machine in tool wear estimation. *International Journal of Production Research*, 54(15):4703–4721, 2016.
- [92] W. Yu, B. Li, H. Jia, M. Zhang, and D. Wang. Application of multi-objective genetic algorithm to optimize energy efficiency and thermal comfort in building design. *Energy and Buildings*, 88:135–143, 2015.
- [93] T. Zhang and B. Yang. Big data dimension reduction using pca. In *2016 IEEE International Conference on Smart Cloud (SmartCloud)*, pages 152–157. IEEE, 2016.
- [94] K. Zhou and S. Yang. Understanding household energy consumption behavior: The contribution of energy big data analytics. *Renewable and Sustainable Energy Reviews*, 56:810–819, 2016.
- [95] J. Zhu, Z. Ge, and Z. Song. Distributed parallel pca for modeling and monitoring of large-scale plant-wide processes with big data. *IEEE Transactions on Industrial Informatics*, 13(4):1877–1885, 2017.
- [96] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the royal statistical society: series B (statistical methodology)*, 67(2):301–320, 2005.