UNIVERSIDADE DE SÃO PAULO

Instituto de Ciências Matemáticas e de Computação

Predição de Preços Imobiliários no Rio de Janeiro: Uma Comparação de Algoritmos e Métodos

Vinícius Viana Vieira

Trabalho de Conclusão de Curso - MBA em Ciência de Dados (CEMEAI)

ICMC USP SÃO CARLOS

UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Predição de Preços Imobiliários no Rio de Janeiro: Uma Comparação de Algoritmos e Métodos

Vinícius Viana Vieira

USP - São Carlos
2024

Vinícius Viana Vieira

Predição de Preços Imobiliários no Rio de Janeiro: Uma Comparação de Algoritmos e Métodos

Trabalho de conclusão de curso apresentado ao Centro de Ciências Matemáticas Aplicadas à Indústria do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, como parte dos requisitos para conclusão do MBA em Ciência de Dados.

Área de concentração: Ciências de Dados

Orientador: Prof. Dr. Diego Raphael

USP - São Carlos

2024

Esta página deve conter a ficha catalográfica e deve ser impressa no verso da folha de rosto.
Para elaborar, acesse o endereço:
https://www.icmc.usp.br/institucional/estrutura-administrativa/biblioteca/servicos/ficha
ou procure um bibliotecário na Seção de Atendimento ao Usuário da Biblioteca do ICMC

ERRATA

Errata			
Folha	Linha	Onde se lê	Leia-se

FOLHA DE AVALIAÇÃO OU APROVAÇÃO

DEDICATÓRIA

Aos meus pais, primos, familiares e amigos que me inspiraram a fazer esse trabalho. Em especial a vocês dedico este projeto de conclusão de curso.

Vocês têm minha eterna gratidão.

AGRADECIMENTOS

Gostaria de expressar minha profunda gratidão à minha mãe, Miriam, sem a qual nada disso seria possível. Ela sempre fez o possível e o impossível para ver meus sonhos se tornarem realidade. Com este trabalho de conclusão de curso, posso afirmar que seu filho está alcançando mais uma etapa importante em sua vida.

Em segundo lugar, à minha namorada Júlia, por me proporcionar todo o suporte emocional necessário ao longo desta trajetória, sempre me apoiando incondicionalmente e sem hesitação.

Em terceiro lugar, gostaria de agradecer ao meu irmão Victor, e ao meu pai Miguel por todo o apoio emocional e os conselhos que me deram ao longo da minha jornada.

Um agradecimento especial aos meus primos e tias que acompanharam minha trajetória na profissional e de alguma forma tornaram essa experiência mais leve e descontraída.

Expresso minha gratidão ao professor orientador Diego Raphael Leão pelos conselhos, ensinamentos e paciência durante a orientação do meu trabalho de conclusão de curso.

Agradeço também à Universidade de São Paulo e a todos os funcionários que trabalham incansavelmente para garantir seu pleno funcionamento, em especial aos professores que compartilharam comigo todo o conhecimento possível.

Por último, mas não menos importante, gostaria de agradecer à banca examinadora pelo seu valioso contributo e pelo aprimoramento do projeto.

Sou eternamente grato a todos vocês, pois deixaram uma marca inesquecível em minha vida.

“Se você pensa que pode ou se pensa que não pode, de qualquer forma você está certo.”

Henry Ford (1920)

RESUMO

Vieira, Vinicius. **Predição de Preços Imobiliários no Rio de Janeiro:** Uma Comparação de Algoritmos e Métodos. 2025. 52 f. Trabalho de conclusão de curso (MBA em Ciência de Dados) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2020.

Este projeto de conclusão de curso em Ciência de Dados tem como objetivo a análise e predição de preços de imóveis, tanto de aluguel quanto de venda, utilizando dados extraídos da internet por meio de técnicas avançadas de raspagem de dados e a integração com APIs externas. O processo iniciou com a coleta dos dados de diversas fontes online, incluindo informações sobre as características dos imóveis, como localização, tamanho, número de quartos e outros atributos. Além disso, dados complementares foram obtidos de APIs externas, agregando mais contexto às análises e melhorando a qualidade do modelo preditivo.

Após a raspagem e a integração com as APIs, os dados passaram por um rigoroso processo de tratamento e limpeza, removendo inconsistências e preparando-os para a aplicação de modelos preditivos. Para a predição dos preços, foram utilizados cinco modelos distintos: Regressão Linear, Regressão Ridge, Floresta Aleatória, XGBoost e Redes Neurais. Cada modelo foi testado e avaliado quanto à sua capacidade de generalização e precisão na previsão dos valores, levando em consideração as variáveis coletadas e as peculiaridades dos dados.

A combinação dessas técnicas de modelagem, aliada à adição de dados provenientes de fontes externas, permitiu uma análise robusta e uma comparação entre métodos clássicos e modernos de aprendizado de máquina. O projeto, portanto, contribui para a aplicação de Ciência de Dados no mercado imobiliário, oferecendo uma abordagem quantitativa para a avaliação e previsão de preços de imóveis, com grande potencial de uso em plataformas e serviços do setor.

Palavras-chave: Raspagem de Dados, Imóveis, Predição de Preços

ABSTRACT

Vieira, Vinicius. **Real Estate Price Prediction in Rio de Janeiro**: A Comparison of Algorithms and Methods. 2025. 52 pages. Master's Thesis (MBA in Data Science) – Institute of Mathematical and Computational Sciences, University of São Paulo, São Carlos, 2020.

This Data Science capstone project aims to analyze and predict real estate prices, both for rental and sales properties, using data extracted from the internet through advanced web scraping techniques and integration with external APIs. The process began with the collection of data from various online sources, including information about the properties' characteristics, such as location, size, number of rooms, and other attributes. Additionally, complementary data was obtained from external APIs, adding more context to the analysis and improving the quality of the predictive model.

After the web scraping and API integration, the data underwent a rigorous treatment and cleaning process to remove inconsistencies and prepare it for predictive modeling. For price prediction, five distinct models were used: Linear Regression, Ridge Regression, Random Forest, XGBoost, and Neural Networks. Each model was tested and evaluated for its generalization ability and accuracy in predicting prices, taking into account the collected variables and data peculiarities.

The combination of these modeling techniques, along with the addition of data from external sources, enabled a robust analysis and a comparison between classical and modern machine learning methods. Therefore, this project contributes to the application of Data Science in the real estate market, offering a quantitative approach for property price evaluation and prediction, with significant potential for use in platforms and services in the sector.

Keywords: Web Scraping, Real Estate, Price Prediction

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ABNT	–	Associação Brasileira de Normas Técnicas
ASTM	–	American Society for Testing and Materials
β_	–	Coeficiente de retenção ao cisalhamento
c	–	Coesão
d_t0	–	Escorregamento relativo à resistência máxima
E_c	–	Módulo de elasticidade do concreto
FLA	–	Flambagem Localizada da Alma
f_c	–	Resistência à compressão do concreto
G_c	–	Energia de fratura à compressão
h	–	Largura de banda de fissuras
K_t	–	Rigidez tangencial
K_n	–	Rigidez normal
μ_	–	Coeficiente de atrito
σ_	–	Tensão normal
τ_	–	Tensão de cisalhamento
L	–	Conector proposto de superfície lisa
R	–	Conector proposto de superfície com ranhuras
RP	–	Conector proposto de superfície com ranhuras e furos
AM	–	Aprendizado de Máquina
IBGE	–	Instituto Brasileiro de Geografia e Estatística

SUMÁRIO

Sumário

1 INTRODUÇÃO

Este trabalho surgiu de uma ideia a partir da principal dor enfrentada por jovens que migram da casa dos pais para uma moradia própria: encontrar um imóvel que seja bom e acessível. No estado do Rio de Janeiro, essa dificuldade é agravada por fatores como o elevado custo de vida, os altos preços praticados no mercado imobiliário e a precariedade na oferta de empregos, especialmente para quem está no início da vida profissional.

1.1 Motivação e Contextualização

A escolha do tema deste trabalho nasceu da percepção de uma dificuldade recorrente enfrentada por muitos moradores e investidores no mercado imobiliário do estado do Rio de Janeiro. A busca por bons imóveis, seja para compra ou aluguel, apresenta diversos desafios. Segundo (Crepaldi, 2024), os jovens têm tido maior dificuldade em comprar imóveis, reflexo de um cenário marcado por aumentos significativos nos preços das propriedades, contrastando com a redução do poder aquisitivo da população brasileira no mesmo período.

Como se não bastasse a discrepância entre os preços dos imóveis e o poder aquisitivo da geração atual em comparação à anterior, esta enfrenta outro desafio significativo: o tempo gasto com deslocamentos. De acordo com (IBGE, 2019), em média 6,4 horas semanais no trajeto de ida e volta ao trabalho. Essa realidade torna ainda mais crucial a escolha de um imóvel bem localizado, próximo ao local de trabalho, para melhorar a qualidade de vida e reduzir o tempo perdido no trânsito.

1.2 Hipótese da Pesquisa

A ausência de estudos na literatura que explorem a aplicação de algoritmos de aprendizado de máquina para a predição dos preços de venda e aluguel de imóveis no estado do Rio de Janeiro, somada às dificuldades na extração e tratamento dessas informações, gera incertezas sobre quais algoritmos seriam mais adequados para essa tarefa e quais características dos imóveis desempenham um papel mais relevante na modelagem preditiva.

Diante desse contexto, espera-se que modelos mais sofisticados e complexos, como redes neurais profundas (com múltiplas camadas) e algoritmos baseados em embeddings, apresentem resultados superiores em comparação com abordagens mais simples, como a Regressão Linear.

Com base nisso, este estudo tem como objetivo investigar a eficácia de diferentes modelos de aprendizado de máquina na previsão dos preços de imóveis, formulando as seguintes hipóteses:

Hipótese 1: Modelos preditivos baseados em redes neurais profundas ou embeddings possuem maior capacidade preditiva em comparação com modelos mais simples, como a Regressão Linear.

Hipótese 2: Acredita-se que os modelos conseguem generalizar bem os resultados, desde que atributos relevantes, como características físicas do imóvel (número de quartos, área total, presença de mobília), localização geográfica e infraestrutura disponível na região, sejam corretamente considerados no treinamento dos modelos.

1.3 Objetivos

Diante do contexto apresentado, o presente estudo, caracterizado como uma pesquisa empírica quantitativa, tem como principal objetivo comparar a capacidade preditiva de diferentes modelos supervisionados de aprendizado de máquina (AM) e redes neurais (RN). O foco é estimar, com maior precisão, o preço de um imóvel com base em seus atributos e na comparação com outros imóveis similares.

A partir dessa análise, será possível determinar se o imóvel está subvalorizado ou supervalorizado em relação ao mercado. Por fim, o estudo busca disponibilizar esses insights ao usuário por meio de uma interface prática e acessível.

Para alcançar o objetivo principal proposto, o estudo necessita atingir os seguintes objetivos específicos:

Coletar atributos estruturais de imóveis anunciados em plataformas online no estado do Rio de Janeiro, utilizando técnicas de raspagem de dados. Esses atributos incluem, por exemplo, área construída, número de quartos, banheiros, vagas de garagem, e outras características relevantes para a precificação.

a) Coletar informações complementares relacionadas às características da vizinhança e da localidade dos imóveis, utilizando dados disponibilizados pela Prefeitura do Rio de Janeiro. Esses dados incluem indicadores socioeconômicos da região, bem como distâncias até pontos de interesse, como o centro da cidade, shoppings, pontos de ônibus e outras infraestruturas relevantes.

b) Desenvolver modelos de aprendizado de máquina utilizando algoritmos amplamente reconhecidos na literatura para problemas de predição de preços de imóveis. Entre os modelos a serem empregados estão: Regressão Linear Múltipla (RLM), Florestas Aleatórias (Random Forest - RF), XGBoost e Redes Neurais Artificiais Perceptron Multi-Camada (RNAMLP).

c) Realizar a otimização dos modelos testando diferentes conjuntos de hiperparâmetros, com o objetivo de identificar a configuração que apresente o melhor desempenho

preditivo. A avaliação será feita com base em métricas consagradas, como o Coeficiente de Determinação (R^2), Erro Absoluto Médio (Mean Absolute Error - MAE), Erro Percentual Absoluto Médio (Mean Absolute Percentage Error - MAPE) e Raiz Quadrada do Erro Quadrático Médio (Root Mean Squared Error - RMSE).

d) Desenvolver uma e interpretar o melhor modelo para cada tipo de modelo e conjunto de dados extraido

1.4 Estrutura do Trabalho

A estrutura deste trabalho foi organizada de maneira a apresentar, de forma clara e progressiva, os elementos fundamentais para o estudo, desde a contextualização inicial até a conclusão dos resultados.

- **Capítulo 1: Introdução** – Introduz a motivação do estudo, o contexto em que está inserido, a hipótese de pesquisa, os objetivos gerais e específicos, além de uma visão geral sobre a organização do trabalho.

- **Capítulo 2: Revisão Bibliográfica** – Revisa estudos anteriores que tratam de problemas similares, destacando contribuições relevantes e identificando lacunas na literatura que justificam o presente estudo.

- **Capítulo 3: Metodologia** – Detalha o processo de coleta, pré-processamento e integração dos dados, além do treinamento, otimização e seleção dos modelos de

aprendizado de máquina. Também apresenta as métricas de avaliação e a abordagem para análise da interpretação dos modelos.

- **Capítulo 4: Resultados e Discussões** – Analisa os resultados obtidos a partir da aplicação dos modelos, comparando o desempenho das diferentes abordagens com base nas métricas definidas. Discute as implicações dos achados, a importância dos atributos identificados e as limitações do estudo.

- **Capítulo 5: Conclusão** – Resume os principais resultados, avalia a contribuição do trabalho e propõe direções para futuras pesquisas na área.

- **Capítulo 6: Recomendação para futuras pesquisas** – Aponta os aspectos que devem ser aprimorados ou aprofundados no projeto, a fim de alcançar os objetivos propostos de maneira mais eficaz.

Essa organização busca assegurar que o leitor compreenda os objetivos do estudo, os métodos utilizados e a relevância dos resultados, ao mesmo tempo em que destaca a importância da análise exploratória para fundamentar as etapas subsequentes.

2 REVISÃO BIBLIOGRÁFICA

Este capítulo apresenta as principais informações e conceitos que fundamentam este estudo, com base em uma revisão bibliográfica. São abordados aspectos relacionados aos dados utilizados, incluindo a metodologia de obtenção, bem como as técnicas de modelagem empregadas ao longo da pesquisa.

2.1 Dados do Mercado Imobiliário

Os dados utilizados neste estudo foram obtidos por meio de raspagem de informações da plataforma Zap Imóveis, através do site. O conjunto de dados deste estudo abrange exclusivamente a área do estado do Rio de Janeiro, contendo tanto informações sobre as vendas de imóveis quanto informações sobre aluguel, é importante salientar que os dados obtidos serão empregados apenas para fins acadêmicos.

Esta plataforma é integrante do Grupo OLX, que inclui também as marcas como Viva Real e OLX Imóveis. Juntas, essas plataformas totalizam 19 milhões de anúncios de imóveis e recebem cerca de 60 milhões de visitas mensais (OLX, s.d.). Além disso, o Zap Imóveis se destaca como “o marketplace com o maior número de anúncios de imóveis do país, com mais de 6,5 milhões de opções disponíveis em todo o Brasil.” (OLX, s.d.).

Os dados dos imóveis são inicialmente disponibilizados pela plataforma em formato tabular, onde cada linha representa um imóvel com suas informações mais relevantes, incluindo um link de acesso para a página específica do anúncio. Para realizar uma análise mais detalhada, foi necessário acessar individualmente as páginas de cada imóvel e extrair informações adicionais. Posteriormente, essas informações foram analisadas e selecionadas com base em critérios definidos durante o estudo.

2.2 Aprendizado de Máquina

De acordo com (Raschka & Mirjalili, Python Machine Learning, 2019), o aprendizado de máquina (do inglês *machine learning*) é um subcampo da inteligência artificial que se concentra no desenvolvimento de algoritmos capazes de aprender padrões e a partir de dados, sem depender exclusivamente de interações humanas.

O princípio central do aprendizado de máquina é o treinamento de modelos computacionais por meio de dados históricos, permitindo que esses modelos façam previsões ou classifiquem novas informações. Esses algoritmos, de acordo com (Janiesch, Zschech, & Heinrich, 2020) buscam aprender automaticamente relações e padrões significativos a partir de exemplos e observações.

Graças ao aprendizado de máquina, que se pode observar uma evolução no desenvolvimento de carros automáticos (Rosa, 2024), evoluções na medicina (Maraccini, 2024). Os métodos de aprendizado de máquina podem ser classificados em três categorias principais, de acordo com o (Raschka & Mirjalili, Python Machine Learning, 2019), sendo elas:

1. **Aprendizado Supervisionado**: Baseia-se em conjuntos de dados rotulados, onde a relação entre entradas e saídas é aprendida para prever resultados futuros. Exemplos incluem regressão linear, árvores de decisão e redes neurais.

2. **Aprendizado Não Supervisionado**: Não utiliza dados rotulados, mas busca identificar padrões ocultos ou agrupamentos nos dados. Métodos como clustering e análise de componentes principais (PCA) são amplamente utilizados.

3. **Aprendizado por Reforço**: Envolve agentes que tomam decisões em um ambiente dinâmico, maximizando uma função de recompensa por meio de tentativa e erro. Essa abordagem é comum em sistemas de recomendação e controle robótico.

Cada categoria mencionada anteriormente oferece uma variedade de algoritmos aplicáveis. No caso do aprendizado supervisionado, que foi a abordagem adotada neste estudo, os algoritmos podem ser divididos principalmente em tarefas de regressão e classificação. Esses

métodos variam desde os mais simples, como kNN e modelos lineares (Regressão Linear e Regressão Logística), passando por modelos baseados em Árvores de Decisão, até técnicas de ensemble, como Florestas Aleatórias e Boosting de Gradiente. Além disso, existem modelos mais complexos, como as Redes Neurais Artificiais (RNA), todas essas técnicas abordadas por (Muller & Guido, 2016).

2.3 Algoritmos de Aprendizado Supervisionado

De acordo com (Sugiyama, 2016) o aprendizado supervisionado, pode ser definido, por um modelo treinado com base em pares de entrada e saída previamente conhecidos. Esses pares são compostos por um conjunto de variáveis de entrada (x) e os valores correspondentes de saída (y), que representam o comportamento ou padrão que o modelo deve aprender. O objetivo é ajustar uma função f(x) que relacione as entradas às saídas, de forma que o modelo possa prever corretamente y para novos exemplos não observados.

Existem dois principais tipos de métodos no aprendizado supervisionado: regressão e classificação, sendo estes:

a) Na regressão, a saída y é contínua, e o objetivo é aproximar uma função f(x) que minimize o erro entre os valores previstos e os observados.

b) Na classificação, a saída y é categórica, pertencendo a um conjunto discreto de classes, e o foco está em atribuir corretamente cada entrada à sua classe correspondente.

Em seu livro, (Boehmke & Greenwell, 2020), são apresentados exemplos de problemas comumente abordados por algoritmos de aprendizado supervisionado, como: prever o preço de venda com base nos atributos de imóveis; calcular a probabilidade de desligamento a partir dos atributos de funcionários; e estimar o tempo para lançamento utilizando atributos de produção.

Em seu site, (Chugh, 2024) discute os principais problemas relacionados aos algoritmos de aprendizado supervisionado e destaca os algoritmos mais utilizados nesse contexto, muitos

dos quais são empregados neste estudo. Entre os algoritmos citados estão: regressão linear, árvore de decisão, floresta aleatória, máquinas de vetor de suporte e regressor de reforço de gradiente.

2.4 Regressão Linear

A regressão linear é amplamente reconhecida no campo da estatística devido à sua longa história e simplicidade, sendo um dos algoritmos de predição mais bem compreendidos (Gutierrez, 2015). Ela é frequentemente utilizada para modelar a relação entre uma variável dependente e uma ou mais variáveis independentes, assumindo uma relação linear entre elas. Apesar de sua simplicidade, a regressão linear continua sendo uma ferramenta poderosa, especialmente em abordagens iniciais, para resolver problemas que demandam predições rápidas e interpretáveis.

Em seu livro, (Raschka, Python Machine Learning: Unlock deeper insights into machine learning with this, 2015), Sebastian Raschka apresenta a fórmula para a regressão linear, em sua forma mais simples, da seguinte maneira:

$$y = \beta 0 + \beta 1 x1 + \varepsilon$$

que pode também ser escrita, na sua forma multilinear, como sendo:

$$y = \beta 0 + \beta 1 x1 + \beta 2 x2 + \cdots + \beta p x p + \varepsilon$$

Para Regressão Linear, presume-se que a variável resposta — a variável que desejamos prever — esteja em sua forma quantitativa. Já as variáveis preditoras, que são aquelas utilizadas para realizar a predição, podem ser tanto quantitativas quanto qualitativas. Essa flexibilidade nas variáveis preditoras permite a modelagem de diferentes tipos de dados (Gutierrez, 2015).

Um aspecto fundamental desse processo é o ajuste do modelo, que tem como objetivo encontrar a reta que minimiza os erros entre os valores reais e os valores previstos. Esses erros são medidos pelo método dos mínimos quadrados, que se baseia na soma dos quadrados das diferenças entre os valores observados e os valores preditos. A fórmula matemática que expressa essa relação é descrita da seguinte forma por (Gutierrez, 2015):

$$S = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

onde S representa a soma dos erros ao quadrado, y_i são os valores reais da variável resposta, e $\hat{y}_i$ são os valores previstos pelo modelo. O objetivo é minimizar S, ajustando os coeficientes da reta para que a diferença entre os valores observados e preditos seja a menor possível, resultando em um modelo de regressão linear otimizado (Gutierrez, 2015).

2.5 Regressão de Ridge

De acordo com (Miller, 2022), a regressão Ridge é uma técnica motivada pela observação de que estimativas de mínimos quadrados podem ser grandes, tornando a variância dos estimadores maiores, especialmente em cenários com multicolinearidade entre variáveis preditoras. Sua abordagem consiste em reduzir estocasticamente os coeficientes em direção a zero, criando um modelo mais parcimonioso e diminuindo a multicolinearidade. Um problema comum entre regressões com muitos fatores explicativos.

Este tipo de técnica frequentemente apresenta problemas de sobre ajuste, de acordo com (Muller & Guido, 2016) e se caracteriza por ser uma extensão da regressão linear que busca minimizar os erros ao mesmo tempo em que aplica uma restrição aos coeficientes do modelo introduzindo um fator de penalização, apresentando um estimador viesado, mas com ganho na diminuição da variância (Reynaldo, 1997). Essa abordagem é particularmente eficaz para evitar

o overfitting, pois restringe o impacto de variáveis preditoras individuais no resultado final (Shalev-Shwartz & Ben-David, 2014).

A solução para a regressão Ridge é obtida a partir da seguinte equação, de acordo com (Reynaldo, 1997):

$$b(K) = (W^{\top}W + K)^{-1}W^{\top}y$$

Na equação da regressão de Ridge, cada variável desempenha um papel fundamental para o ajuste do modelo. A variável $(b(K))$ representa os coeficientes ajustados pelo modelo de regressão de Ridge. A matriz (W) contém as variáveis preditoras, enquanto $(W^{\top})$ é a sua transposta, utilizada nos cálculos para ajustar os coeficientes. A matriz (K)é uma matriz de regularização diagonal, proporcional ao parâmetro (λ), que controla o grau de penalização aplicado aos coeficientes. O vetor (y)contém os valores da variável resposta, ou variável dependente. Por fim, (λ) é o parâmetro de regularização que determina a intensidade do encolhimento aplicado aos coeficientes, reduzindo o impacto de colinearidades entre os preditores (Reynaldo, 1997):

2.6 Florestas Aleatórias

Floresta aleatória é um modelo de aprendizado supervisionado utilizado para classificação e regressão, que de acordo com (Boehmke & Greenwell, 2020), é baseado no algoritmo de arvores de decisão. Este método, por ser desenvolvido a partir da combinação de outros métodos recebe a classificação de ensemble, (UFF, 2023).

Esta técnica é desenvolvida com base na tese da “Sabedoria das Multidões”, popularizado por “The Wisdom of Crowds”, de James Surowiecki, que sugere que uma agregação de julgamentos de um grupo de indivíduos pode levar a decisões, ou previsões, mais precisas do que as feitas por especialistas ou indivíduos isolados.

Esta técnica se tornou muito utilizada pelo seu alto poder de generalização, e consiste em um algoritmo que seleciona diversos atributos e características para calcular aleatoriamente múltiplas arvores de decisão, prevalecendo a moda ou média da classe, ou valor, que mais apareceu, (Gutierrez, 2015).

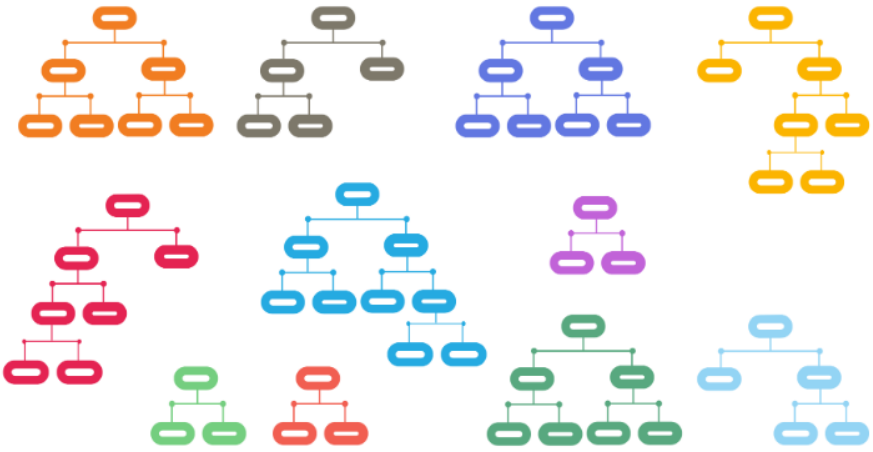

Figura 1: Representação Floresta Aleatória

A Floresta Aleatória apresenta diversas vantagens em relação a uma única árvore de decisão, especialmente em conjuntos de dados grandes e complexos. Ela se destaca pela alta precisão, fornecendo resultados mais robustos e confiáveis devido à combinação das previsões de múltiplas árvores. Além de reduzir significativamente o risco de overfitting, pois seu método de ensemble learning evita o ajuste excessivo aos dados de treinamento, promovendo melhor generalização para novos dados, (UFF, 2023).

2.7 Gradient Boosting

O Gradient Boosting, assim como a floresta aleatória, também é um método ensemble de aprendizado de máquina supervisionado, teve sua implementação original produzida na linguagem de programação R, (Gutierrez, 2015).

Este método, assim como a floresta aleatória, também é composto por uma combinação de arvores de decisão, contudo, enquanto o método de floresta aleatória cria arvores de decisões de maneira independente, o gradiente boosting constrói esse conjunto de árvores em sequência de forma dependente, o que gera, em teoria, um aprendizado e uma melhora a partir de cada árvore de decisão, (Boehmke & Greenwell, 2020).

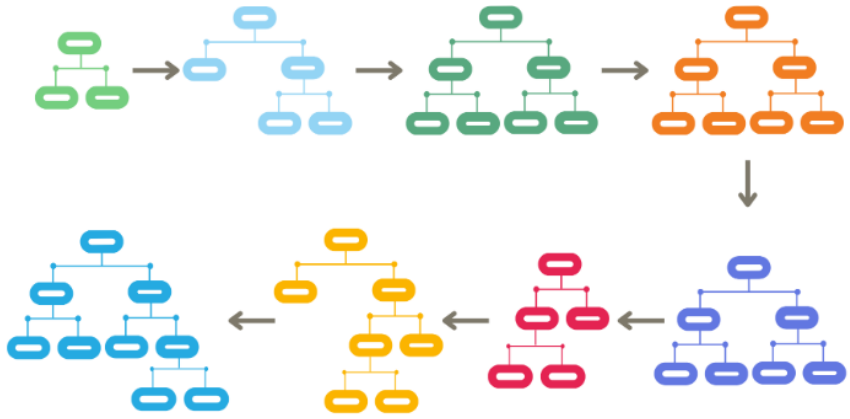

Figura 2: Representação Gradient Boosting

Podemos dizer então, que este método constrói árvores de decisão de maneira sequencial de forma que cada árvore é treinada para corrigir os erros cometidos pelas árvores anteriores. O resultado precisa passar por cada árvore na ordem em que elas foram construídas, (UFF, 2023)

2.8 Redes Neurais

Uma rede neural é um modelo preditivo baseado em algoritmos computacionais que são motivados pela forma como um cérebro humano funciona, (Joel, 2016). Este modelo é baseado em muitos dispositivos básicos de computação (neurônios) que estão conectados entre si em uma rede de comunicação complexa, (Shalev-Shwartz & Ben-David, 2014). Sua motivação surgiu a partir da necessidade de se resolver problemas de classificação não lineares, que muitas

vezes são subestimados por algoritmos mais simples, como a regressão logística, (Gutierrez, 2015).

Esta técnica surgiu em meados da década de 1980, mas por diversas razões, principalmente o custo elevado computacional exigido perdeu força no final dos anos 1990. Mas com a evolução das máquinas, mais especificamente do hardware do computador, esse movimento tem ganhado relevância. Desde 2006 essas redes neurais avançadas têm sido usada para implementar métodos classificados como aprendizado estruturado profundo, ou deep learning, (Gutierrez, 2015).

As redes neurais podem ter sua estrutura em sua forma mais simples, com apenas uma camada de saída ligada à camada de entrada, sendo chamada de rede neural de camada única. Estas em si, possuem bastante destaques para resolução de problemas lineares, (Raschka, Python Machine Learning: Unlock deeper insights into machine learning with this, 2015).

Contudo, as redes neurais podem também, ter sua estrutura em uma forma muito mais complexa, com dezenas de camadas e centenas de neurônios, estas em si possuem a nomenclatura de redes neurais de múltiplas camadas. Este tipo de algoritmo se caracteriza por resolver problemas não lineares. Cada camada possui uma função de ativação linear, fazendo com que o modelo capture relações e padrões mais profundos, (Raschka, Python Machine Learning: Unlock deeper insights into machine learning with this, 2015).

Entretando, diferentemente da rede neural de camada única, a rede neural de camada múltipla possui um problema de interpretação onde a adição de camadas ao algoritmo se transforma em um aumento exponencial de novas conexões, aumentando a complexidade e a interpretação das camadas. Esse problema em si, possui o nome de caixa preta, pois o modelo pode chegar em uma etapa onde a interpretação sobre o que acontece dentro, seja impossível, (Joel, 2016).

3 METODOLOGIA

Neste capítulo serão apresentadas e detalhadas a metodologia empregada para conclusão deste presente estudo. Os resultados obtidos ao longo do desenvolvimento deste projeto são resultado de uma série de etapas necessárias, sendo elas:

3.1 Origem e Aquisição dos dados

Os dados utilizados neste estudo foram obtidos a partir de duas fontes principais. Primeiramente, foram extraídos de sites externos do Zap Imóveis, empresa que faz parte do grupo OLX Brasil e tem como seu modelo de negócios o anúncio de imóveis por meio da internet (OLX, s.d.), utilizando técnicas de raspagem de dados. Essa abordagem, conhecida como web scraping, consiste em coletar informações por meio do protocolo de transferência de hipertexto (HTTP) (TechTarget, 2023).

Paralelamente, utilizaram-se API’s disponibilizadas pela cidade do Rio de Janeiro, por meio do portal Data.Rio que reúne diversas informações sobre a cidade do Rio de Janeiro utilizando tecnologia avançada, possibilitando um acesso à informação mais ágil e interativo para toda população (Passos, 2024). Estas informações podem conter, por exemplo, localização de hospitais e estações de metrô. As API’s atuam como um intermediário entre dois aplicativos, ou dois bancos de dados, diferentes permitindo que sistemas com estruturas distintas se comuniquem e centralizem informações de forma eficiente (MuleSoft, s.d.).

3.1.1 Plataforma de anúncio de imóveis

As características estruturais e principais dos imóveis foram extraídas a partir da lista de anúncios disponibilizada na plataforma Zap Imóveis. Essa plataforma é amplamente utilizada, especialmente na região Sudeste do Brasil, e se destaca como o marketplace com o maior número de anúncios de imóveis do país, atingindo uma média de 60 milhões de visitas mensais (OLX, s.d.).

A escolha dessa plataforma para o presente estudo baseia-se na sua consolidação no mercado e na confiabilidade de seu vasto volume de informações. O Zap Imóveis permite a veiculação de anúncios provenientes tanto de imobiliárias quanto de corretores credenciados, ampliando significativamente a variedade de imóveis disponíveis e proporcionando dados abrangentes para análise. Cabe salientar que existe uma verificação sobre os corretores e imobiliárias, trazendo maior confiabilidade nos dados.

Abaixo, apresenta-se uma figura que ilustra a interface do Zap Imóveis, utilizada para a visualização dos imóveis. A página inicial contém as informações mais relevantes de cada anúncio, como área útil, número de quartos, banheiros, vagas de garagem, endereço e preço do imóvel. Esses dados essenciais estão organizados em formato de lista, com os imóveis distribuídos por páginas, facilitando a navegação e a coleta das informações necessárias para este estudo.

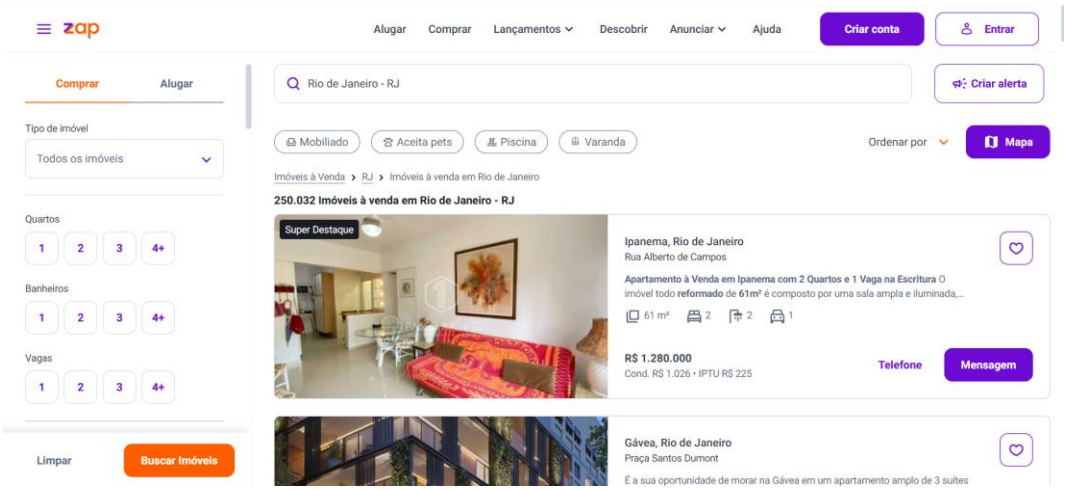

Figura 3 - Interface Zap Imóveis

Após clicar no imóvel de interesse, o usuário é redirecionado para uma página personalizada da Zap Imóveis, dedicada exclusivamente ao imóvel selecionado. Essa página apresenta informações adicionais que podem ser relevantes para um potencial comprador. Essas informações variam desde detalhes sobre as quadras esportivas presentes no condomínio, número de piscinas, até a presença de um elevador funcional na casa ou apartamento

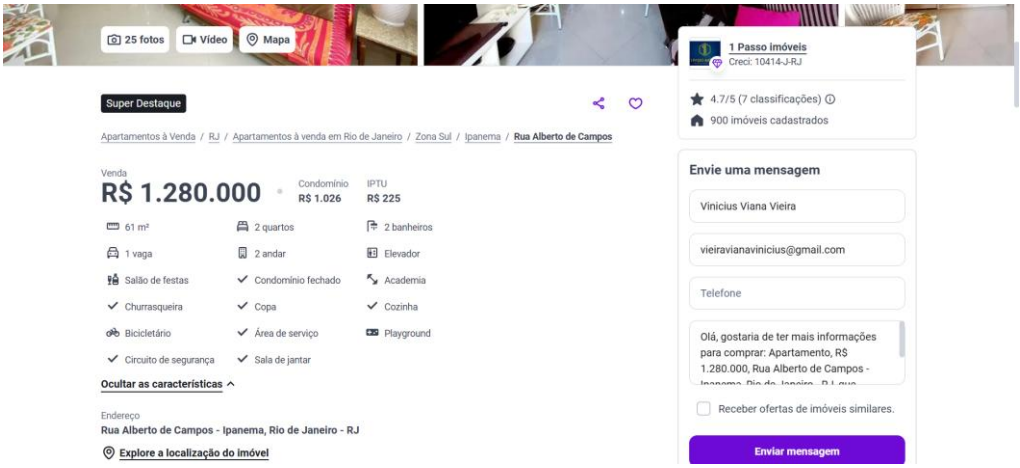

Figura 4 - Página personalizada do imóvel

Para realizar a raspagem completa dos dados da plataforma Zap Imóveis, foi necessário capturar tanto o ID, único para cada anúncio, quanto o link de redirecionamento para a página personalizada de cada imóvel. O processo de coleta foi estruturado em duas etapas:

a) **Captura Inicial:** Nesta etapa, o script coletou informações básicas, como o ID do imóvel e o link da página onde o anúncio está hospedado.

b) **Raspagem Detalhada:** Após a coleta inicial, foi possível acessar a página personalizada de cada imóvel e extrair as informações mais detalhadas e relevantes para o estudo.

O estudo abrangeu tanto anúncios de aluguel quanto de venda, permitindo uma análise comparativa entre esses dois segmentos do mercado imobiliário. No total, foram coletadas informações de 8.332 anúncios de imóveis disponíveis para aluguel e 8.578 anúncios de imóveis disponíveis para venda. Ressalta-se que os dados analisados contemplam exclusivamente imóveis localizados na cidade do Rio de Janeiro, garantindo um recorte específico e coerente para o objetivo deste trabalho.

Dada a grande quantidade de dados a serem processados, foi necessário implementar técnicas de paralelização no script de extração. Em vez de realizar a coleta de dados de forma linear, o que poderia gerar gargalos no processamento, utilizou-se a criação de múltiplas

threads. Como as informações de cada imóvel não eram interdependentes, essa abordagem foi viável e eficiente, permitindo que cada thread se encarregasse da extração de dados de um imóvel específico.

De acordo com (Tecnologia G. d., 2024), “threads são componentes fundamentais em sistemas operacionais modernos, permitindo que múltiplas operações sejam realizadas simultaneamente dentro de um único processo. Isso significa que uma aplicação pode executar várias tarefas ao mesmo tempo, melhorando a eficiência e a performance geral do sistema.”

No presente estudo, foram configurados ao entorno de 6 threads simultâneas para a extração dos dados. Essa técnica resultou em uma otimização significativa, reduzindo o tempo de execução do script para aproximadamente um sexto do que seria necessário em um processamento linear. A adoção da paralelização demonstrou ser crucial para lidar com o grande volume de informações, mantendo a eficiência e a agilidade do processo.

3.1.2 Plataforma municipal de dados georreferenciados

Os dados socioeconômicos, geográficos e as características de cada bairro assim como a localização de serviços disponibilizados na cidade do Rio de Janeiro são acessíveis por meio da plataforma online denominada: Data Rio. Desenvolvida pela Prefeitura do Rio de Janeiro em parceria com o Instituto Pereira Passos (IPP), a plataforma tem como objetivo principal oferecer suporte ao planejamento de políticas públicas e à promoção do desenvolvimento urbano com base em dados qualificados e confiáveis.

Conforme descrito pela (Prefeitura, 2023), o IPP é o instituto responsável por gerir informações estratégicas sobre a cidade, sendo uma referência nacional e internacional em gestão e planejamento urbano. Sua atuação é essencial para assegurar que decisões estratégicas sejam embasadas em dados robustos, promovendo um desenvolvimento urbano sustentável e eficiente.

A plataforma Data Rio apresenta uma interface gráfica amigável e intuitiva, disponibilizando uma ampla gama de informações geográficas e socioeconômicas sobre a cidade do Rio de Janeiro. Esses dados podem ser acessados em diferentes formatos, permitindo que sejam consumidos e utilizados de forma versátil para diversos fins.

Abaixo, estão listadas as categorias de informações disponíveis na plataforma, acompanhadas de uma visualização da página principal:

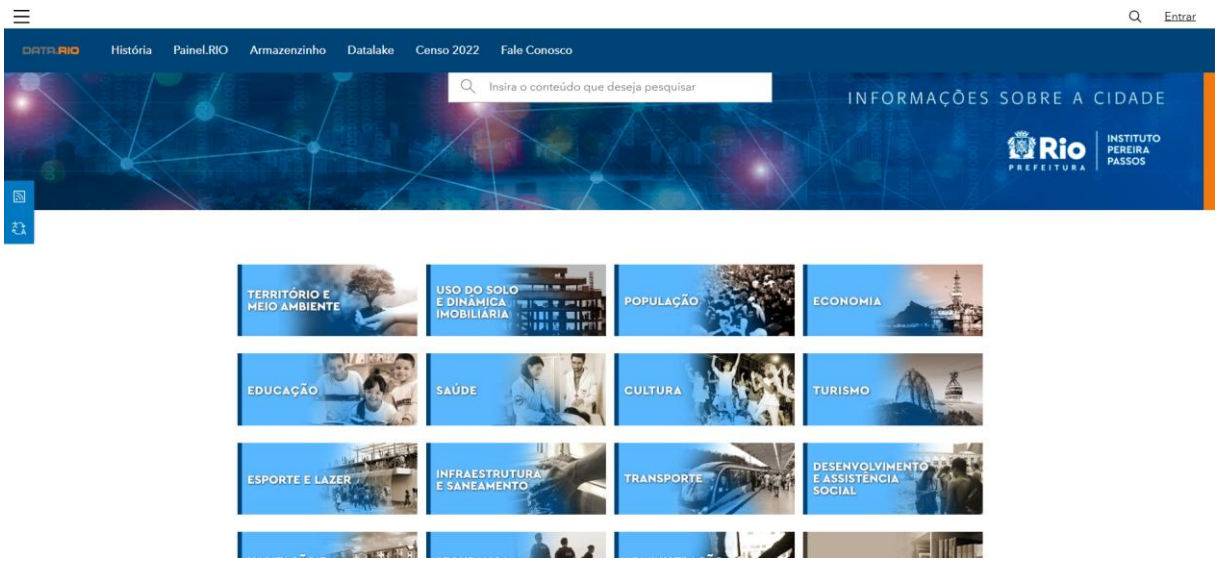

Figura 5 - Página inicial Data Rio

Após clicar em cada categoria, será redirecionado para uma página dedicada a categoria escolhida. Nesta página é possível encontrar os conjuntos de dados disponíveis para acesso

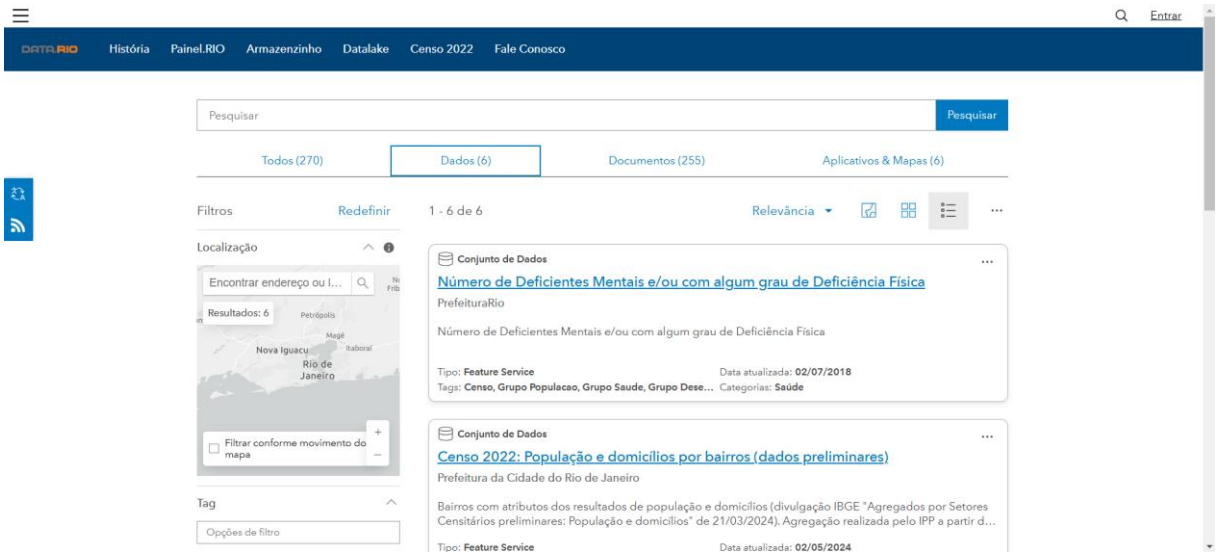

Figura 6 - Página específica Data Rio

Os dados são disponibilizados em seu formato tabular para facilitar a análise e a integração com ferramentas de ciência de dados. Esse formato permite manipular grandes volumes de informações de forma eficiente, seja para operações como filtragem, agregação e visualização, ou para treinamento de modelos preditivos e análises estatísticas.

3.2 Limpeza dos dados

Nesta seção será explicado quais foram as técnicas de pré-processamento de dados aplicadas neste presente estudo. Esta parte do projeto é considerada a parte mais importante de todo o pipeline do projeto, já que a qualidade e a quantidade de informações úteis são fatores extremamente importantes na hora de definir quão bem um algoritmo de aprendizado de máquina pode performar (Sugiyama, 2016).

Esta etapa do projeto envolve a aplicação de uma variedade de métodos estatísticos para melhorar a qualidade dos dados, sendo eles: gerenciamento de dados faltantes (eliminando-os, substituindo-os ou estimando-os), eliminação de valores duplicados, seleção de atributos, eliminação de outliers ou até mesmo a agregação dos dados e a redução da dimensionalidade (Gutierrez, 2015). Neste estudo foram utilizados os seguintes métodos:

3.2.1 Seleção de atributos

Esta etapa é dedicada à seleção das variáveis que, inicialmente, aparentam ser mais relevantes para o estudo. Essa fase é essencial, pois é comum que conjuntos de dados contenham colunas redundantes, ou seja, que fornecem a mesma informação de formas diferentes. O objetivo principal é garantir que os atributos mais adequados sejam escolhidos, de modo a alcançar a melhor performance possível no modelo.

Embora a escolha inicial seja, inevitavelmente, baseada em critérios arbitrários, técnicas mais avançadas podem ser aplicadas ao longo do estudo para refinar essa seleção. Uma abordagem comum é utilizar as variáveis em uma predição inicial e avaliar a importância de

cada coluna na performance do modelo, identificando aquelas que mais contribuem para os resultados. Isso permite uma escolha mais fundamentada e eficiente, reduzindo a complexidade do modelo e potencializando sua precisão. É uma boa prática reduzir a dimensionalidade do conjunto de dados a ser usado, por isso, essa técnica se torna relevante para o processo.

3.2.2 Gerenciamento dos dados faltantes

Após a escolha dos atributos mais relevantes para o projeto, realiza-se a etapa de limpeza dos dados, esta etapa, foca no gerenciamento dos valores ausentes no conjunto de dados. Essa parte do processo é essencial, pois muitos algoritmos de aprendizado de máquina não conseguem lidar adequadamente com valores faltantes, o que pode comprometer o desempenho e a confiabilidade do modelo. Entre as principais técnicas para tratar dados ausentes, destacam-se:

1. **Substituição dos valores ausentes:** Consiste em imputar os valores ausentes com estimativas calculadas a partir dos dados disponíveis. Técnicas comuns incluem o uso da média, mediana ou moda da coluna correspondente. Esse método preserva todas as amostras do conjunto de dados e é amplamente utilizado quando a quantidade de valores ausentes é pequena em relação ao total.

2. **Remoção dos valores ausentes:** Envolve a exclusão de linhas ou colunas que contenham valores ausentes. Essa abordagem é simples e direta. Tem como lado positivo a confiabilidade dos dados.

3. **Previsão dos valores ausentes:** Utiliza modelos estatísticos ou algoritmos de aprendizado de máquina para prever os valores ausentes com base em outras variáveis do conjunto de dados. Essa técnica pode ser particularmente útil para preencher lacunas em dados essenciais, preservando a integridade do conjunto. Métodos comuns incluem regressão, algoritmos de vizinhos mais próximos (KNN) e árvores de decisão.

Para este estudo, adotou-se a abordagem de remoção dos valores ausentes. Essa técnica pode aumentar a confiabilidade do conjunto de dados, eliminando inconsistências que poderiam impactar negativamente a análise. No entanto, essa abordagem deve ser utilizada com cautela,

pois reduz a quantidade de dados disponíveis, o que pode ser prejudicial em estudos onde a completude do conjunto de dados é essencial, especialmente ao aplicar algoritmos de previsão que dependem de uma amostra robusta para garantir resultados precisos.

3.2.3 Remoção de valores duplicados

A remoção de valores duplicados também é uma etapa essencial no pré-processamento de dados. Dados duplicados ocorrem quando uma mesma entrada é registrada mais de uma vez, o que pode acontecer devido a erros no processo de coleta, integração ou processamento. A presença de registros duplicados pode inflar artificialmente os resultados, distorcer estatísticas e prejudicar a eficiência dos algoritmos de aprendizado de máquina.

Neste projeto, foi realizada uma análise detalhada para identificar e eliminar registros duplicados no conjunto de dados. Para isso, utilizou-se ferramentas de processamento de dados que permitem detectar linhas com valores idênticos em todas ou em um subconjunto específico de colunas. Como cada imóvel possuía um ID exclusivo, previamente definido pelo responsável da divulgação, a identificação e eliminação de ocorrências repetidas foram realizadas de maneira simples e eficiente.

A execução dessa etapa resultou em um conjunto de dados mais confiável e enxuto, eliminando redundâncias que poderiam introduzir vieses ou aumentar o tempo de processamento sem adicionar valor à análise. Essa limpeza contribuiu para a qualidade geral do estudo, garantindo que os modelos desenvolvidos fossem treinados com dados únicos e representativos do problema em questão.

3.2.4 Remoção de outliers

A remoção de outliers é uma etapa fundamental no processo de pré-processamento de dados, especialmente em estudos que envolvem modelos preditivos. Outliers são valores extremos que diferem significativamente dos demais dados. Se não tratados, esses valores

podem distorcer métricas estatísticas, comprometer a precisão do modelo e dificultar a generalização dos resultados, (Tecnologia B. , 2018).

Para este projeto, foram analisadas as variáveis presentes no conjunto de dados a fim de identificar possíveis outliers. Essa identificação foi realizada utilizando métodos estatísticos, como o cálculo do intervalo interquartil (IQR), ou técnicas baseadas na visualização gráfica, como boxplots. Valores que se situaram além de limites definidos, como 1,5 vezes o IQR acima do terceiro quartil ou abaixo do primeiro quartil, foram classificados como potenciais outliers.

A remoção foi adotada como estratégia principal, uma vez que a presença desses valores poderia influenciar negativamente o desempenho dos modelos de regressão e aprendizado de máquina utilizados neste estudo.

Esse processo contribuiu para a obtenção de um conjunto de dados mais limpo e consistente, garantindo que as previsões fossem menos suscetíveis a distorções causadas por valores extremos e que os modelos desenvolvidos apresentassem maior confiabilidade e precisão.

3.3 Pré-processamento dos dados

O pré-processamento dos dados é uma etapa fundamental no desenvolvimento de modelos de aprendizado de máquina, pois visa transformar os dados brutos em um formato adequado para análise e modelagem. Essa etapa envolve normalização, padronização e transformação de variáveis categóricas para garantir que o conjunto de dados seja consistente, equilibrado e compatível com os requisitos dos algoritmos.

Essas técnicas foram essenciais para a preparação do conjunto de dados. A normalização e padronização garantiram consistência entre as variáveis numéricas, enquanto a codificação *One-Hot* viabilizou o uso de dados categóricos em algoritmos avançados. Essas transformações, combinadas às etapas de limpeza descritas anteriormente, contribuíram para a qualidade do conjunto de dados e a robustez dos modelos desenvolvidos.

3.3.1 Normalização de Dados

A normalização é uma técnica utilizada para escalonar os dados, assegurando que todas as variáveis estejam dentro de um mesmo intervalo ou escala, normalmente entre 0 e 1. Isso é particularmente relevante em algoritmos baseados em distância, como regressão logística, SVMs e redes neurais, que são sensíveis às magnitudes dos dados. Neste estudo, utilizou-se a técnica de normalização Min-Max, conforme descrito por (Jiawei, Micheline, & Jian, 2012), a qual transforma os valores xxx de cada variável utilizando a fórmula:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

3.3.2 Codificação One-Hot Encoding

A transformação de variáveis categóricas em variáveis numéricas foi realizada por meio da técnica de codificação *One-Hot Encoding*. Essa abordagem cria uma variável binária para cada categoria, atribuindo 1 para a presença e 0 para a ausência de uma categoria específica.

Por exemplo, para uma variável categórica como "Tipo de Imóvel" com valores "Casa", "Apartamento" e "Comercial", foram criadas três colunas separadas, cada uma representando uma das categorias. Essa técnica é amplamente utilizada por algoritmos que não conseguem trabalhar diretamente com dados categóricos, como redes neurais e árvores de decisão (Jiawei, Micheline, & Jian, 2012). Apesar do aumento na dimensionalidade, essa codificação foi essencial para preservar informações categóricas sem introduzir ordens arbitrárias nos dados.

3.3.3 Validação Cruzada

Para os modelos de aprendizado de máquina, excetuando as redes neurais, foi utilizado um algoritmo de validação cruzada. Essa técnica é amplamente reconhecida por sua robustez

na avaliação do desempenho dos modelos, ao dividir o conjunto de dados em múltiplos subconjuntos, denominados *folds*. Cada *fold* é utilizado como conjunto de teste em uma iteração, enquanto os demais são empregados para o treinamento. Essa abordagem permite que o modelo seja avaliado em diferentes partições dos dados, garantindo uma análise mais confiável e reduzindo o impacto de possíveis vieses oriundos de divisões específicas, conforme explicita em sua documentação.

No presente estudo, foi adotada a validação cruzada com cinco *folds* (5-fold cross-validation). Esse método consiste em dividir o conjunto de dados em cinco partes iguais, de modo que, em cada iteração, quatro partes sejam utilizadas para o treinamento e uma para a validação. Esse procedimento é repetido até que cada *fold* tenha sido utilizado uma vez como conjunto de teste.

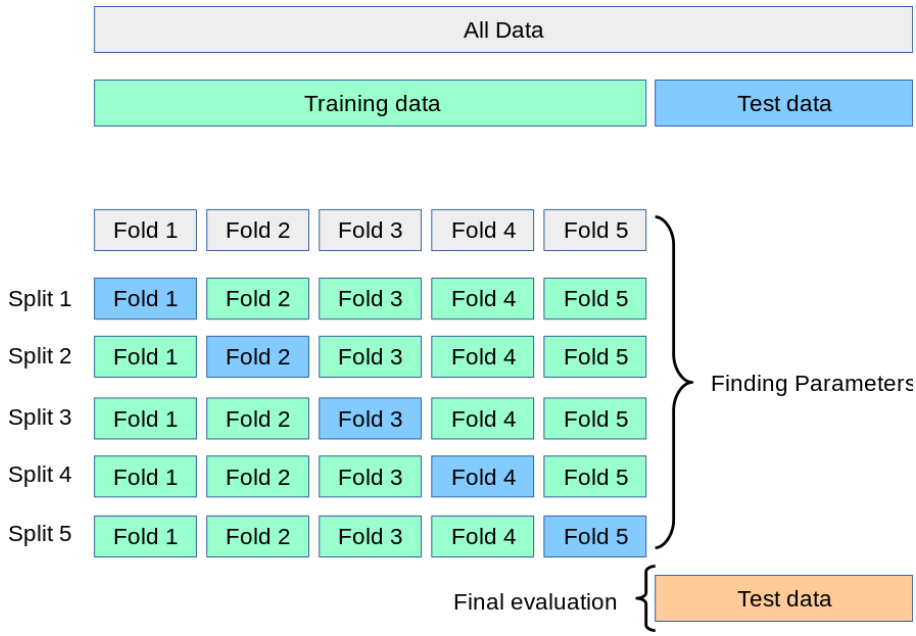

Figura 7 - Validação Cruzada

Essa abordagem foi fundamental para evitar problemas como *overfitting* ou *underfitting*, garantindo que os modelos fossem testados em diferentes divisões do conjunto de dados e avaliados de forma abrangente. (Learn, s.d.). A Validação Cruzada foi escolhida devido à sua capacidade de reduzir o viés e a variância nas estimativas das métricas de desempenho, proporcionando uma análise mais confiável e representativa do comportamento dos modelos.

3.4 Treinamento e Otimização dos Modelos

Esta seção descreve a metodologia utilizada para o treinamento e otimização dos modelos de previsão de valores, as técnicas utilizadas, a escolha das variáveis preditoras, os critérios de avaliação adotados e os ajustes realizados para garantir o melhor desempenho dos algoritmos. O processo de treinamento é dividido em três etapas principais, sendo elas:

3.4.1 Divisão do conjunto de dados

A divisão do conjunto de dados é uma etapa essencial no processo de treinamento de modelos de aprendizado de máquina. Ela garante que os dados sejam separados em partes distintas, com o objetivo de avaliar os resultados de maneira justa e imparcial, além de evitar problemas como o overfitting. Existem várias abordagens para realizar a divisão dos dados, sendo as mais comuns a divisão de 70% para treinamento e 30% para teste, ou 80% para treinamento e 20% para teste, entre outras. A escolha da proporção depende de diversos fatores, como o tamanho do conjunto de dados e a complexidade do modelo.

Para este projeto, foi adotada a estratégia de dividir o conjunto de dados em 80% para treinamento e 20% para teste. Esta divisão é amplamente utilizada na prática de aprendizado de máquina, pois proporciona uma quantidade adequada de dados para treinar o modelo, ao mesmo tempo em que preserva dados suficientes para avaliar seu desempenho de forma precisa e imparcial.

A maior parte do conjunto de dados é destinada ao treinamento do modelo, uma vez que é nesse subconjunto que o modelo aprenderá as relações entre as variáveis preditoras e a variável alvo. Durante esta fase, o modelo é ajustado e otimizado para identificar padrões e fazer previsões a partir dos dados. A robustez do conjunto de dados de treinamento é fundamental para garantir que o modelo seja capaz de aprender adequadamente e, assim, alcançar uma boa performance.

Por outro lado, o conjunto de teste, menor em tamanho, é utilizado para avaliar a performance do modelo após o treinamento. Esse conjunto de dados não é utilizado durante o processo de ajuste do modelo, garantindo que a avaliação da sua eficácia seja realizada de maneira justa, sem o risco de o modelo "decorar" os dados de treinamento. É importante que se tome cuidado na hora da divisão de dados para que não ocorra vazamento de dados, que é quando justamente os dados de treinamento se misturam com os dados de teste. A partir dos resultados obtidos no conjunto de teste, é possível medir a capacidade do modelo de generalizar para dados nunca vistos, o que é crucial para avaliar sua utilidade em um cenário do mundo real.

Além disso, existem outras técnicas relevantes para melhorar o processo de validação do modelo, sendo uma delas a **validação cruzada**. A validação cruzada é uma técnica que divide o conjunto de dados em múltiplos subconjuntos ou "folds" e treina o modelo em diferentes combinações de treinamento e teste, utilizando cada subconjunto como conjunto de teste em algum momento. Esse processo ajuda a obter uma avaliação mais robusta do modelo, minimizando o viés causado por uma divisão única de dados (Datacamp, 2024). A validação cruzada é especialmente útil quando o conjunto de dados é pequeno, pois permite aproveitar ao máximo as amostras disponíveis e garantir que o modelo seja avaliado de forma mais consistente.

3.4.2 Escolha dos hiperparâmetros

Durante o desenvolvimento deste estudo, a escolha inicial dos parâmetros dos modelos foi feita com base em uma análise preliminar, mas de forma arbitrária. A partir dessa escolha, diversos valores para os respectivos parâmetros foram definidos e utilizados dentro de um **pipeline de aprendizado de máquina**. O uso de um pipeline é fundamental, pois organiza e estrutura as etapas de pré-processamento, treinamento e ajuste de modelos, garantindo que todo o fluxo de trabalho seja automatizado e eficiente.

Dentro desse pipeline, foi aplicada a técnica de **Grid Search**, que é uma abordagem amplamente utilizada para a otimização de hiper parâmetros (Learning, 2024). O Grid Search consiste em uma busca exaustiva por meio de uma grade de combinações de diferentes valores para os hiperparâmetros de um modelo. O objetivo desse processo é testar de maneira

sistemática uma série de valores predefinidos para cada parâmetro, com o intuito de encontrar a combinação que melhore a performance do modelo em relação à predição de novos dados (Sapien, 2024).

No contexto deste estudo, o Grid Search foi utilizado para encontrar os melhores valores para parâmetros-chave, como a taxa de aprendizado, o número de árvores em modelos de Random Forest, a profundidade das árvores em modelos de Árvores de Decisão e outros hiperparâmetros críticos. A busca por esses parâmetros foi realizada de forma a otimizar o modelo e melhorar sua capacidade de generalização, minimizando o risco de overfitting e maximizando a precisão na previsão dos dados. Abaixo estão os parâmetros e valores utilizados no algoritmo de grid Search.

Modelo	**Hiperparâmetros**	**Valores Possíveis**
Linear Regression	Sem hiperparâmetros para ajustar	-
Ridge Regression	model__alpha	[0.01, 0.1, 1.0, 10.0, 100.0]
Random Forest	model__n_estimators model__max_depth model__min_samples_split model__min_samples_leaf	[50, 100, 200, 300] [5, 10, 20, 30] [2, 5, 10] [1, 2, 4]
Gradient Boosting	model__n_estimators model__learning_rate model__max_depth model__subsample	[50, 100, 200, 300] [0.01, 0.05, 0.1, 0.2] [3, 4, 5] [0.8, 0.9, 1.0]

Tabela 1 - Combinação Hiperparâmetros

3.4.3 Abordagem Utilizada

A abordagem adotada neste estudo consistiu inicialmente na escolha de um modelo simples, uma estratégia amplamente utilizada em projetos de aprendizado de máquina. Essa abordagem permite uma compreensão básica do problema e dos dados antes de avançar para modelos mais complexos. O objetivo principal foi identificar o desempenho mínimo aceitável e as limitações dos modelos iniciais, facilitando, assim, a decisão sobre a necessidade de modelos mais sofisticados.

Optou-se por iniciar com a Regressão Linear, um modelo simples que se destaca por sua interpretabilidade, facilidade de treinamento e rapidez na validação. Esse modelo proporcionou uma análise preliminar eficiente sobre as relações entre as variáveis independentes e a variável alvo. Além disso, seu desempenho inicial serviu como um indicador de quão bem o problema poderia ser resolvido de maneira simples e se seria necessário recorrer a modelos mais avançados, dada a complexidade dos dados.

Uma vez obtidos os resultados do modelo simples, a estratégia foi evoluir para modelos mais complexos, como Random Forest e Gradient Boosting. Esses modelos, embora mais poderosos, apresentam maior risco de overfitting e são mais desafiadores de interpretar, o que requer cuidados adicionais durante o treinamento e a validação. O progresso para modelos mais complexos ocorre somente quando é necessário aprimorar a performance do modelo, garantido um equilíbrio entre poder preditivo e complexidade.

4 RESULTADOS E DISCUSSÕES

Este capítulo reúne a apresentação e a análise dos resultados obtidos a partir das aplicações das etapas descritas no Capítulo 3, com o objetivo de identificar o algoritmo de aprendizado de máquina mais eficaz na predição de valores de venda, e aluguel, de imóveis no estado do Rio de Janeiro.

4.1 Conjunto de Dados

Após a conclusão de todas as etapas de preparação, incluindo a limpeza dos dados, a seleção de atributos e a adição de informações de geolocalização, foram obtidos dois conjuntos distintos para análise. O primeiro, com 518 registros, foi destinado à predição de preços de aluguel, enquanto o segundo, com 756 registros, foi utilizado para a análise de preços de venda. Para ambos os conjuntos, 80% dos dados foram alocados para os processos de treinamento e validação, enquanto os 20% restantes compuseram o conjunto de teste, assim como é referenciado em seu artigo (Shalev-Shwartz & Ben-David, 2014).

Os dados dos imóveis foram extraídos da plataforma Zap Imóveis, que disponibilizava o endereço dos mesmos, o que possibilitou a análise espacial das informações. Utilizando APIs especializadas, foi realizado o georreferenciamento dos imóveis, transformando os endereços em coordenadas de latitude e longitude. Além disso, integraram-se dados adicionais obtidos por meio da API do DataRio, que forneceu informações relevantes sobre as características dos bairros. Essas informações incluíram a localização de estações de metrô, hospitais e a presença de complexos urbanísticos, como favelas. Essa abordagem enriqueceu o conjunto de dados, permitindo uma análise mais detalhada e contextualizada do ambiente em que os imóveis estão inseridos.

Para este estudo, foram selecionados inicialmente 13 atributos preditivos. A escolha desses atributos foi baseada em uma análise de relevância conduzida para identificar as variáveis com maior impacto na predição de preços de venda e aluguel. Durante a etapa de pré-processamento dos dados, esses 13 atributos foram transformados em 16, devido à aplicação do algoritmo One Hot Encoding.

O One Hot Encoding foi utilizado para converter variáveis categóricas, como "tipo de imóvel" e "mobiliado", em colunas binárias. Essa transformação garante que os algoritmos de aprendizado de máquina possam interpretar as categorias sem introduzir relações ordinais inexistentes entre elas, o que seria um problema ao utilizar codificações baseadas em números inteiros. Por exemplo, a variável "tipo de imóvel" com três categorias (casa, apartamento e estúdio) foi desmembrada em três colunas distintas, cada uma representando a presença ou ausência de uma das categorias.

Além disso, foi aplicado o algoritmo de StandartScaler nos atributos numéricos, como área total, número de quartos e banheiros, para normalizar os valores em um intervalo entre 0 e 1. Essa normalização reduziu a influência de diferenças de escala entre os atributos, evitando que variáveis com magnitudes maiores dominassem o treinamento dos modelos.

4.2 Análise dos Dados

Esta etapa do estudo foi essencial para compreender as características principais do conjunto de dados, avaliar a distribuição das variáveis e identificar possíveis relações entre elas. Sendo elas análise de correlação dos atributos preditivos e distribuição dos dados.

4.2.1 Análise de correlação

A análise de correlação foi aplicada aos atributos preditivos de cada conjunto de dados, tanto de aluguel quanto de vendas. Essa etapa teve como objetivo identificar relações lineares entre as variáveis, restringindo-se exclusivamente aos atributos numéricos, com a exclusão dos categóricos. A matriz de correlação foi calculada e visualizada por meio de um mapa de calor gerado pela biblioteca Seaborn. Essa abordagem permitiu identificar as colunas com maior relevância potencial para os modelos de predição. A seguir o mapa de correlação entre as variáveis preditivas nos respectivos conjuntos de dados:

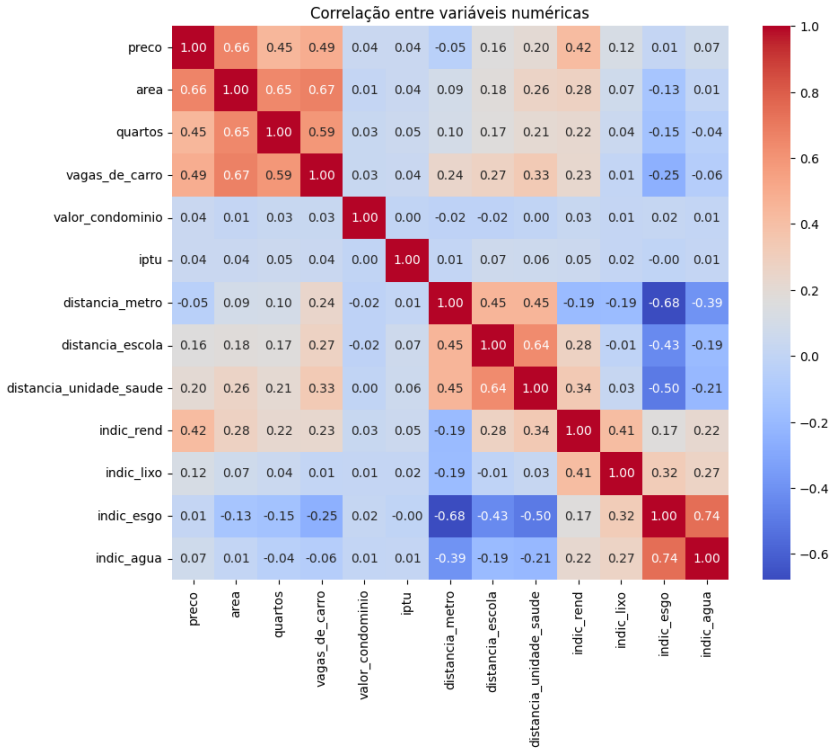

Figura 8 - Mapa de Calor das variáveis do conjunto de dados de Venda

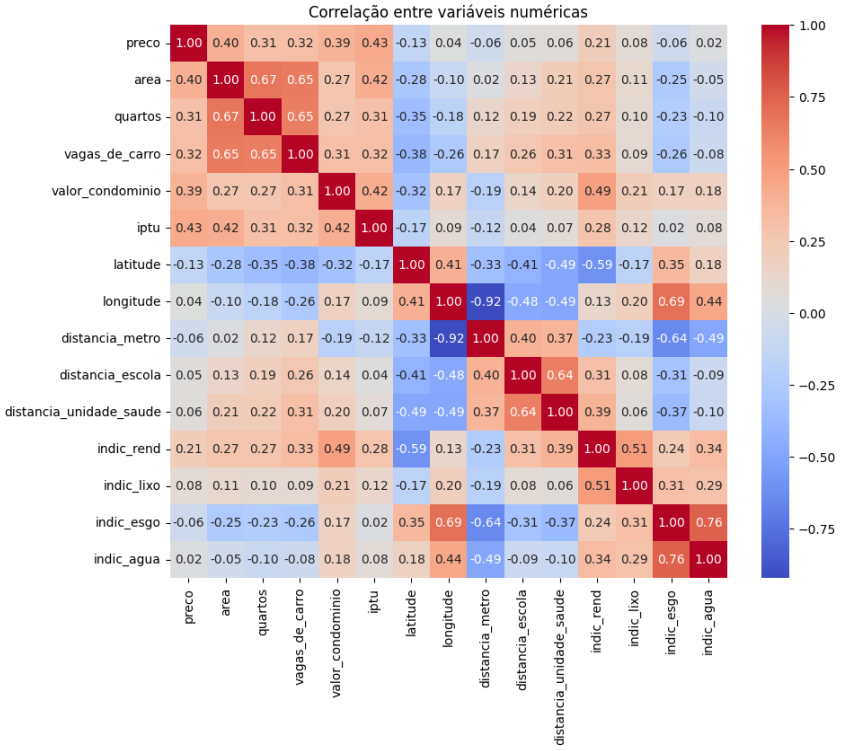

Figura 9 - Mapa de Calor das variáveis do conjunto de dados de Aluguel

A partir dessa análise, foi possível identificar padrões importantes, como a existência de variáveis altamente correlacionadas que podem influenciar diretamente as variáveis-alvo de predição. Além disso, atributos com baixa correlação com o alvo foram considerados para possível exclusão nos modelos finais, contribuindo para a redução da dimensionalidade e aprimorando o desempenho dos algoritmos. Esses resultados guiaram a seleção inicial de atributos relevantes, proporcionando um direcionamento mais eficiente para as etapas subsequentes do processo de modelagem preditiva.

4.2.2 Análise de distribuição e identificação de outliers

Esta análise pode ser considerada uma das mais importantes para a utilização de algoritmos de aprendizado de máquina, pois garante uma compreensão detalhada do comportamento dos dados, permitindo identificar padrões, anomalias e possíveis problemas que podem afetar a performance dos modelos.

A análise de distribuição é essencial para observar como os valores das variáveis estão distribuídos, verificando a presença de assimetrias, valores extremos e desvios em relação a uma distribuição normal. Para isso, foram gerados histogramas e gráficos de densidade que destacaram a forma das distribuições de cada atributo numérico.

A identificação de outliers, por sua vez, foi realizada por meio de métodos estatísticos, como o intervalo interquartil (IQR), e gráficos como boxplots, que evidenciaram pontos fora do padrão esperado. A detecção e o tratamento de outliers são cruciais, pois valores extremos podem impactar negativamente o treinamento de modelos sensíveis, como regressões, algoritmos baseados em distância e redes neurais.

Em um primeiro momento, o conjunto de dados apresentava uma distribuição que evidenciava a presença de assimetrias significativas e outliers extremos, como ilustrado nas imagens dos gráficos de distribuição e boxplots. Esses aspectos poderiam influenciar negativamente a performance dos algoritmos de aprendizado de máquina, especialmente aquelas sensíveis a escala e valores extremos, como regressões e métodos baseados em distância. A seguir as imagens das distribuições:

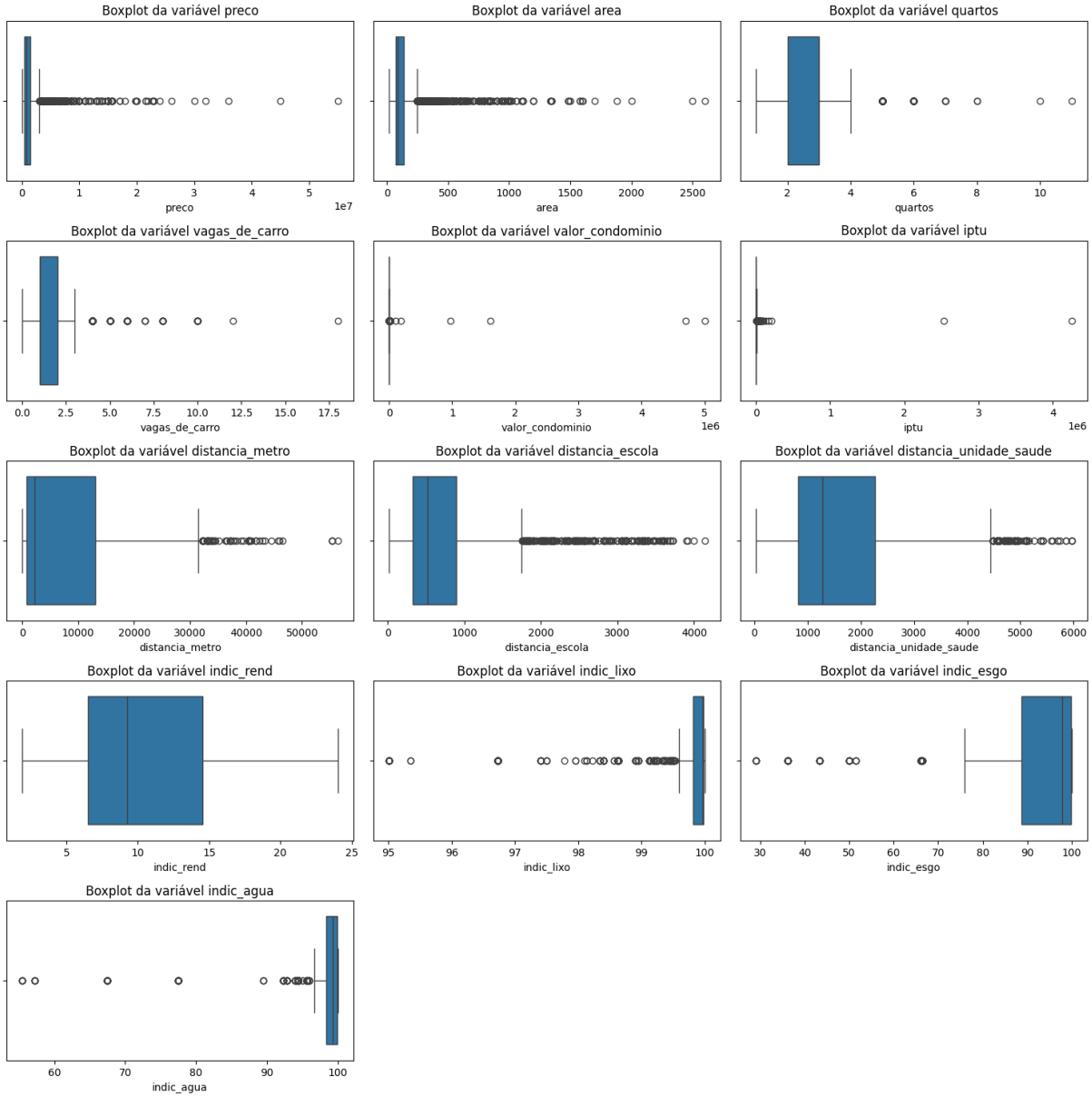

Figura 10 - Gráficos de distribuição das variáveis preditoras no conjunto de dados de venda antes da análise

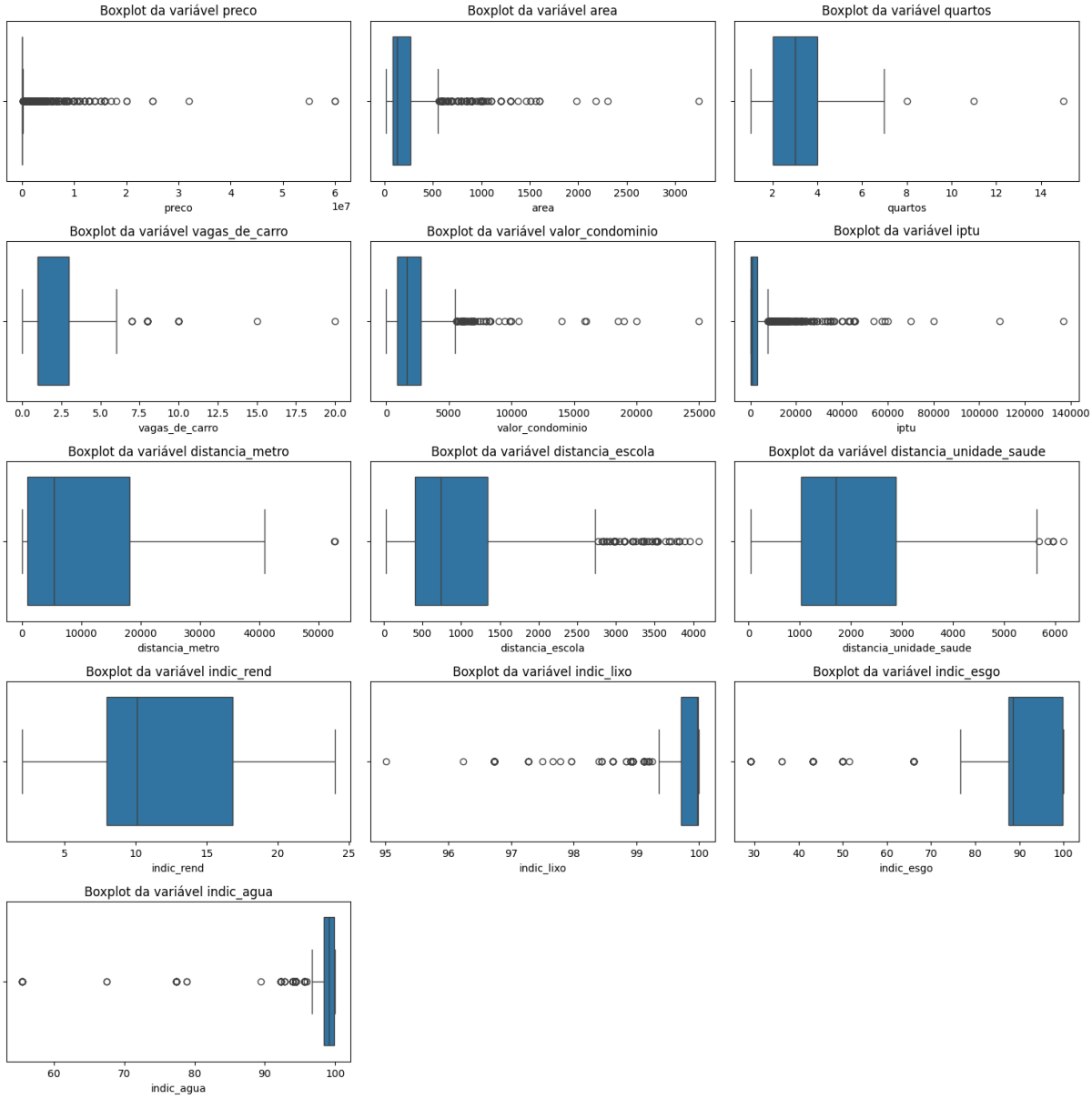

Figura 11 - Gráficos de distribuição das variáveis preditoras no conjunto de dados de aluguel antes da análise

Para resolver essas questões, foi aplicada a técnica de quartis, utilizando o intervalo interquartil (IQR) para identificar e tratar os outliers. O IQR é calculado como a diferença entre o terceiro quartil (Q3) e o primeiro quartil (Q1), representando a amplitude da faixa central dos dados, qualquer valor fora dessa faixa foi considerado um outlier e removido do conjunto de dados. A abordagem seguiu os seguintes passos:

- $Limite\ Inferior = Q1 - 1.5 \times IQR$
- $Limite\ Superior = Q1 + 1.5 \times IQR$

Após a utilização das técnicas de identificação de outliers, o conjunto de dados de aluguel e vendas foi ajustado, removendo-se as observações que estavam fora dos limites estabelecidos pelos quartis. Com isso, as distribuições das variáveis se tornaram mais consistentes e equilibradas, o que contribuiu para a melhoria da qualidade dos dados. Esse processo garantiu que os modelos de aprendizado de máquina fossem treinados com dados mais representativos, reduzindo a influência de valores extremos que poderiam distorcer as previsões e melhorar a precisão dos resultados.

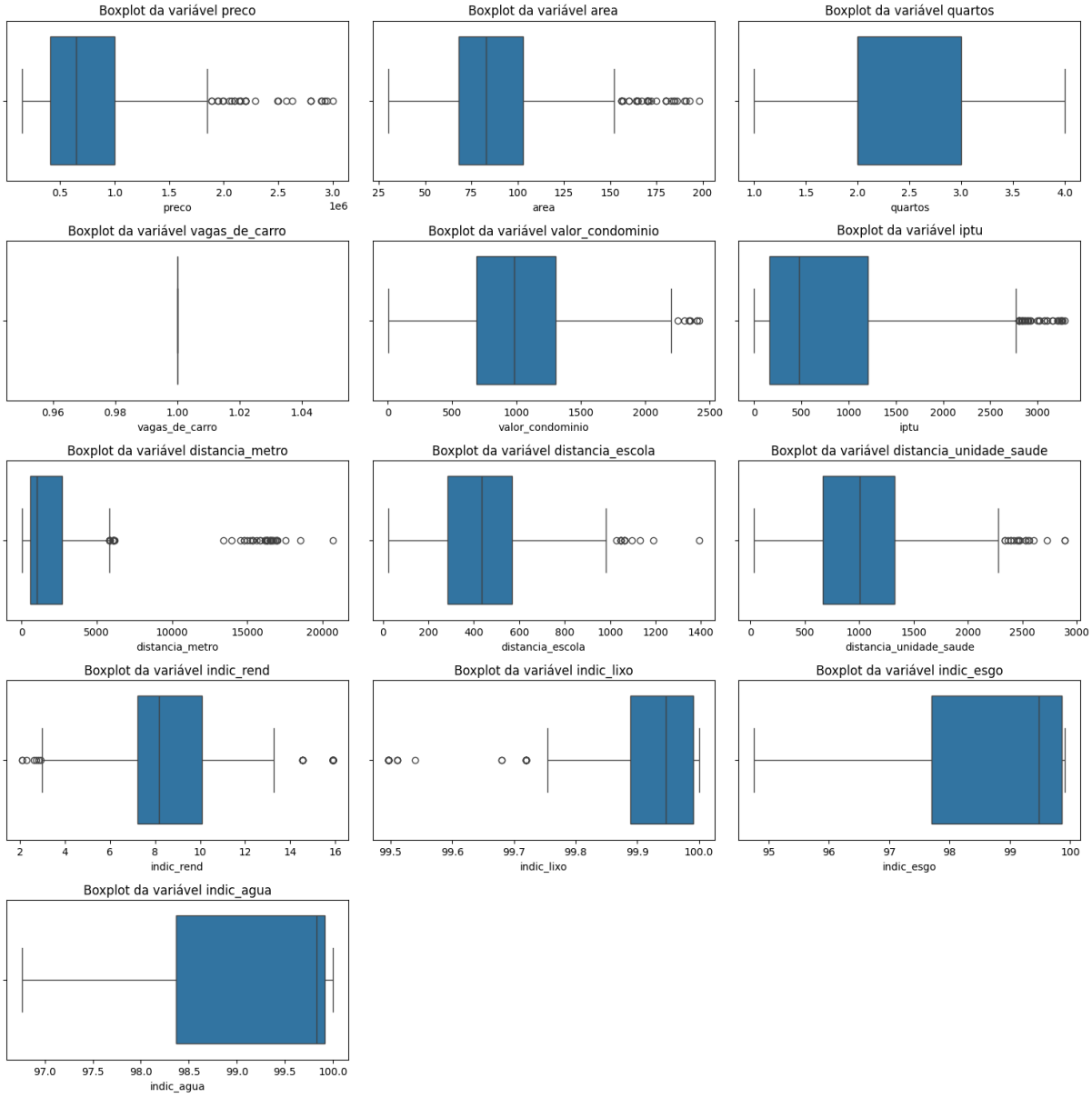

Figura 12 - Gráficos de distribuição das variáveis preditoras no conjunto de dados de venda após a análise

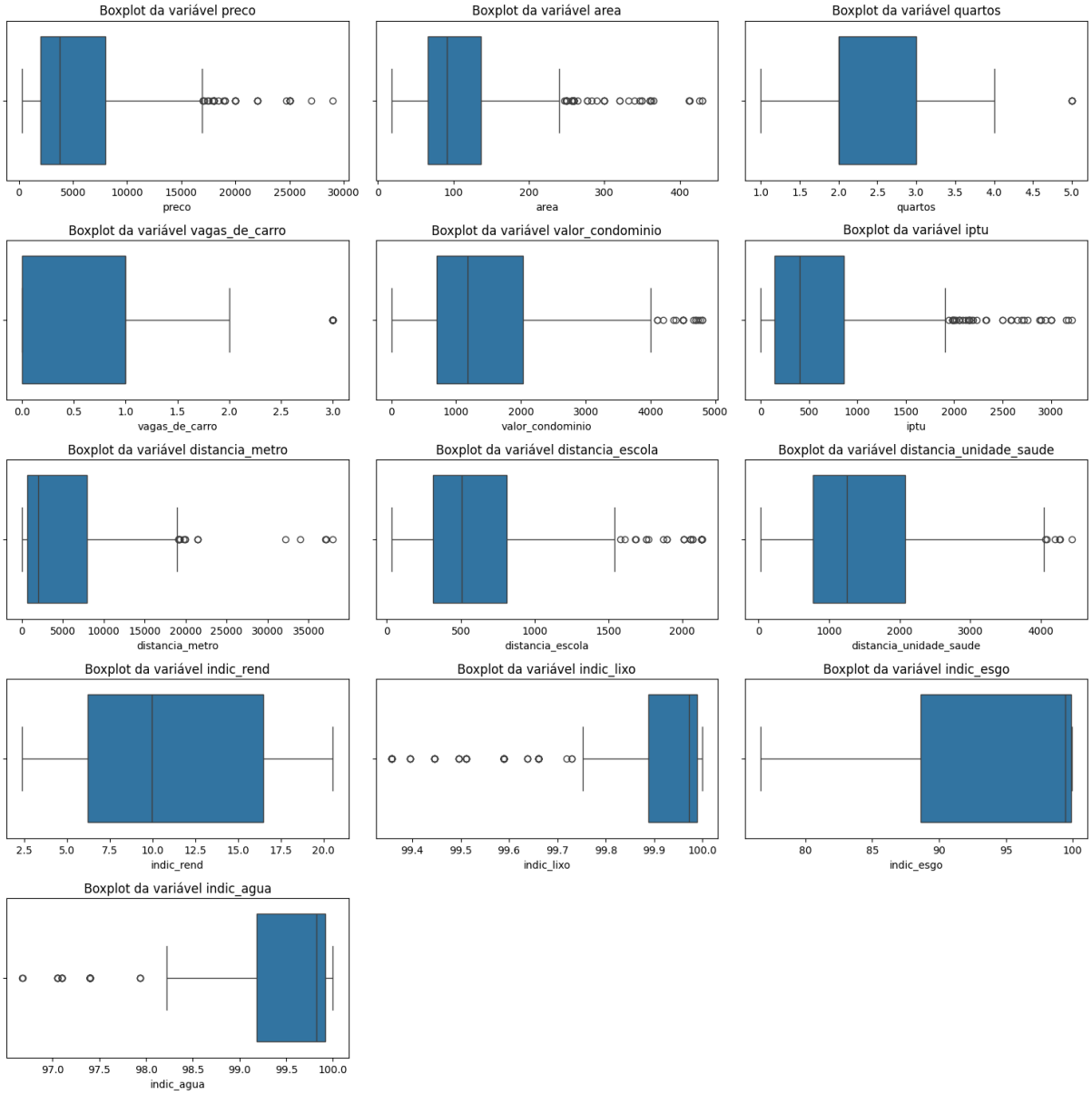

Figura 13 - Gráficos de distribuição das variáveis preditoras no conjunto de dados de aluguel após a análise

4.3 Métricas de Avaliação

Nesta seção, são apresentadas as abordagens, os resultados e as análises realizadas para cada modelo empregado no presente estudo. A metodologia adotada incluiu a aplicação de técnicas como Validação Cruzada, com o objetivo de aumentar a robustez e a confiabilidade dos resultados obtidos. Além disso, foram utilizados diferentes modelos preditivos, abrangendo abordagens lineares, representadas pela Ridge Regression, abordagens não lineares, como Random Forest e Gradient Boosting, e modelos baseados em redes neurais, adaptados ao problema em questão.

A avaliação do desempenho dos modelos foi realizada com base em métricas amplamente reconhecidas na análise preditiva. O coeficiente de determinação (R^2) foi utilizado para mensurar a proporção da variância explicada pelo modelo, enquanto o erro médio absoluto (MAE) permitiu avaliar o erro médio em unidades da variável dependente. Adicionalmente, o erro absoluto percentual médio (MAPE) foi empregado para quantificar a precisão das previsões em termos percentuais, e a raiz do erro quadrático médio (RMSE) foi utilizada para identificar a magnitude dos erros, atribuindo maior peso a desvios mais significativos.

4.3.1 Métricas de Avaliação

Cada modelo foi avaliado com base em métricas de desempenho, tais como MAE (Mean Absolute Error), RMSE (Root Mean Squared Error), R^2 (Coeficiente de Determinação) e MAPE (Mean Absolute Percentage Error).

- **Coeficiente de Determinação (R^2):** É uma métrica utilizada para avaliar o grau de explicação que um modelo de regressão fornece sobre a variabilidade de uma variável dependente em relação às variáveis independentes. Em termos simples, o R^2 indica a proporção da variabilidade total da variável dependente que pode ser explicada pelo modelo.

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

- **Erro Médio Absoluto (Mean Absolute Error - MAE):** É responsável por medir a média das diferenças absolutas entre os valores observados e os valores previstos. Ele é uma métrica intuitiva que avalia diretamente o erro médio em unidades da variável dependente.

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i|$$

- **Erro Absoluto Percentual Médio (Mean Absolute Percentage Error - MAPE):** É responsável por expressar o erro absoluto médio em termos percentuais, indicando o quão distante, em média, as previsões estão dos valores reais.

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i} \cdot 100\%$$

- **Raiz do Erro Quadrático Médio (Root Mean Squared Error - RMSE):** É responsável por medir a raiz quadrada da média dos quadrados dos erros. Ele é uma métrica que penaliza grandes erros mais severamente do que o MAE, devido à elevação dos resíduos ao quadrado.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

4.4 Resultados e Otimização dos modelos na etapa de treinamento

Durante o processo de treinamento e otimização dos modelos, tanto para previsão de vendas quanto para aluguel, foram aplicadas as etapas metodológicas previamente descritas neste estudo. A avaliação do desempenho dos modelos foi realizada com base nas seguintes métricas: Erro Absoluto Médio (MAE), Raiz do Erro Quadrático Médio (RMSE), Coeficiente de Determinação (R^2) e Erro Percentual Absoluto Médio (MAPE).

A partir dessas métricas, foi possível ajustar o conjunto de dados, selecionar os melhores atributos e comparar diferentes abordagens de modelagem, buscando a configuração mais eficiente. Vale destacar que esse processo não segue uma sequência linear fixa, mas exige uma abordagem iterativa, na qual cada escolha é continuamente avaliada e reavaliada com base nas métricas definidas. Esse método garante que as decisões tomadas ao longo da otimização sejam fundamentadas e direcionadas à obtenção dos melhores resultados possíveis.

Para se obter os resultados e uma melhor otimização e modelagem, foram escolhidos 4 algoritmos de aprendizado de máquina, sendo eles; Linear Regression, Ridge Regression, Random Florest Regression, XGboost Regresion e Redes Neurais. Para uma comparação justa e fidedigna os dados de todos esses algoritmos foram submetidos as mesmas etapas de análise, tratamento e pré-processamento de dados.

A primeira etapa do treinamento consiste na separação do conjunto de dados, seguindo a seguinte divisão de 80% dos dados para o conjunto de treinamento e 20% dos dados para o conjunto de validação. Em cima desses dados de treinamento, será aplicado o algoritmo de validação cruzada, com a divisão dos dados em 5 partes, utilizado para a escolha e otimização dos hiper parâmetros, os valores destes foram disponibilizados na tabela 1.

Nesta primeira etapa, após a aplicação deste treinamento, foram obtidos as médias de cada modelo, para cada métrica para vendas e aluguel de imóveis no Rio de Janeiro e foram dispostas no seguinte gráfico boxplot:

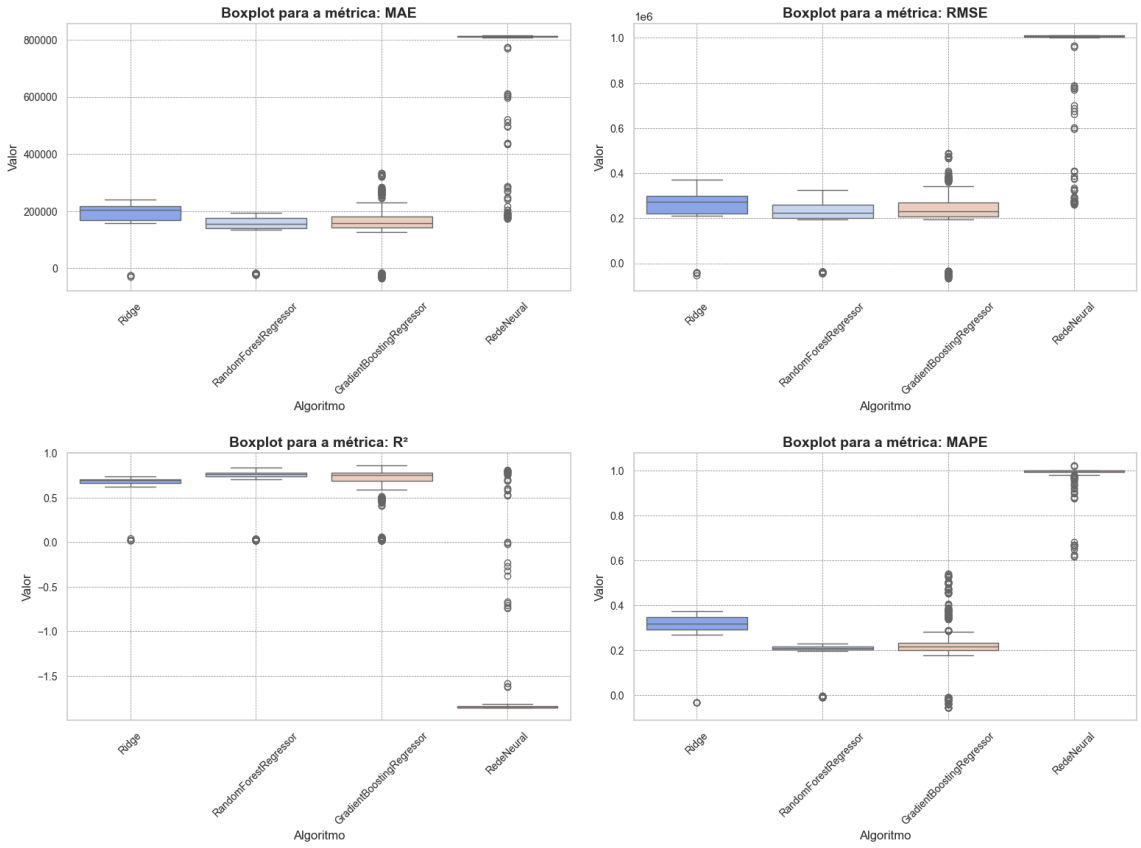

Figura 14 - Boxplot de métricas para os dados de vendas de imóveis

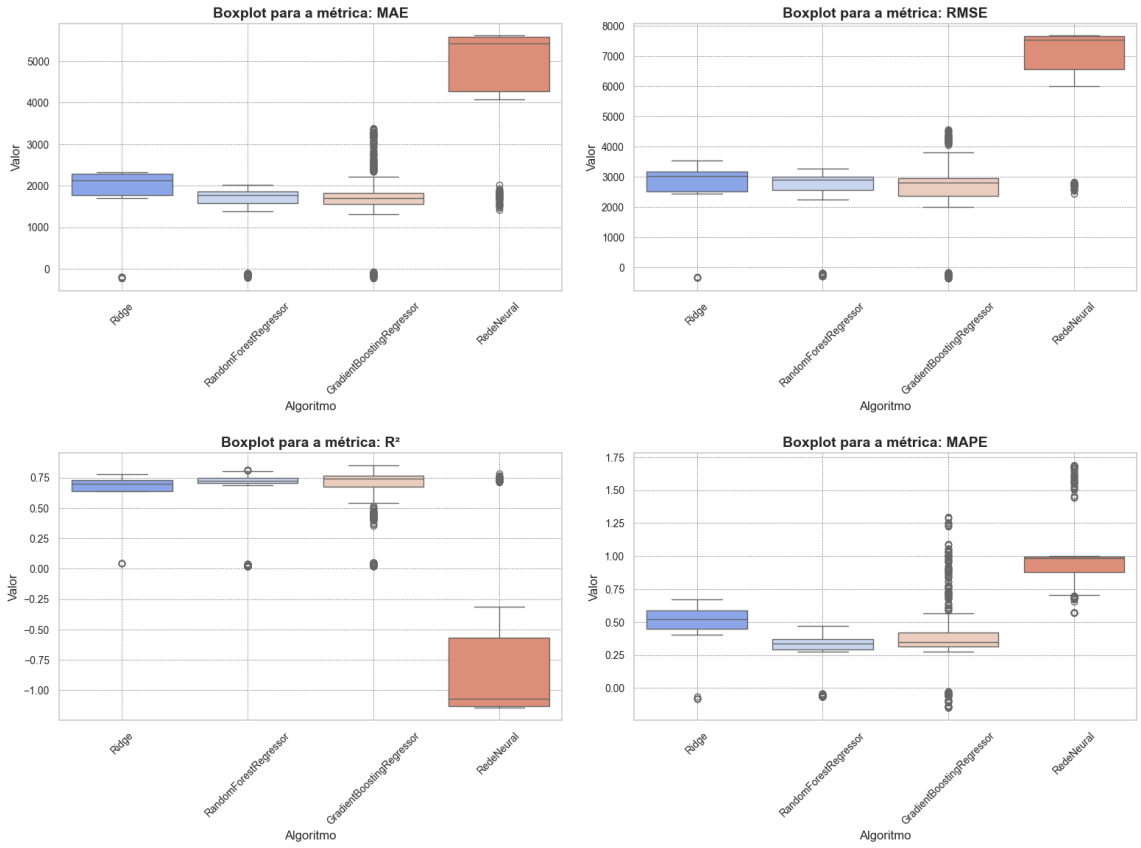

Figura 15 - Boxplot de métricas para os dados de aluguel de imóveis

A partir desta imagem, algumas observações podem ser feitas. Embora as redes neurais tenham apresentado, em média, o pior desempenho entre os modelos avaliados, isso não implica necessariamente que elas não irão generalizar bem. Esse resultado pode estar relacionado à complexidade do modelo e à necessidade de um ajuste mais refinado dos hiper parâmetros. Foram realizados diversos testes com diferentes combinações de parâmetros e camadas, aumentando o seu leque em estrutura e hiper parâmetros buscando aprimorar sua performance e minimizar possíveis problemas de sobreajuste ou subajuste, o mesmo pode ser evidenciado em modelos como os de Random Florest e XGBoost.

Contudo, para os hiper parâmetros que geraram os melhores modelos para cada algoritmo, se tem os seguintes resultados:

Modelo	Melhores Parâmetros	MAE	MAPE	RMSE	R^2
Linear Regression	Sem ajuste de hiperparâmetros	21.259,97	570,04	30.136,60	0.7080
Ridge Regression	model__alpha: 10.0	20.995,79	550,37	30.068,29	0.7094
Random Forest	**max_depth:** 20 **min_samples_leaf:** 1 **min_samples_split:** 2 **n_estimators:** 50	16.817,57	336,54	27.168,75	0.7630
Gradient Boston	**learning_rate:** 0.05 **max_depth:** 5 **n_estimators:** 300 **subsample:** 0.8	15.791,82	330,49	26.094,08	0.7812
Rede Neural	**learning_rate:** 1.23e-05 **hidden_layers:** 3 **dropout:** 0.3 **activation:** linear	1.426,44	150,63	2.448,74	0.7818

Tabela 2- Resultados com os melhores modelos dos dados de aluguel

Modelo	Melhores Parâmetros	MAE	MAPE	RMSE	R^2
Linear Regression	Sem ajuste de hiperparâmetros	205.152,95	36,16	280.929,04	0.7192
Ridge Regression	model__alpha: 10.0	201.687,75	34,04	284.019,34	0.7139
Random Forest	**max_depth:** 20 **min_samples_leaf:** 4 **min_samples_split:** 10 **n_estimators:** 300	155.543,30	22,34	243.088,86	0.7892
Gradient Boston	**learning_rate:** 0.05 **max_depth:** 4 **n_estimators:** 100 **subsample:** 0.9	156.044,90	22,62	241.979,25	0.7914
Rede Neural	**learning_rate:** 1.23e-05 **hidden_layers:** 3 **dropout:** 0.3 **activation:** linear	172.906,06	61,66	259.560,54	0.8109

Tabela 3- Resultados com os melhores modelos dos dados de venda

A Rede Neural apresentou o melhor desempenho entre os modelos testados, seguida de perto pelo Gradient Boosting, que também se mostrou muito eficaz. Embora os modelos de Regressão Linear e Ridge sejam simples e interpretáveis, eles não capturam toda a complexidade dos dados como os modelos de aprendizado mais avançado, como o Random Forest e o Gradient Boosting. Esses resultados indicam que técnicas mais sofisticadas, como

Redes Neurais, são adequadas para problemas mais complexos de predição de preços de imóveis.

4.5 Seleção do melhor modelo

A partir da análise comparativa dos modelos testados nos dados de validação, a Rede Neural se destacou como a abordagem mais eficaz para a predição de preços de imóveis, tanto para venda quanto para aluguel. Sua capacidade de capturar padrões não lineares e interações complexas entre as variáveis garantiu um desempenho superior em relação aos demais modelos.

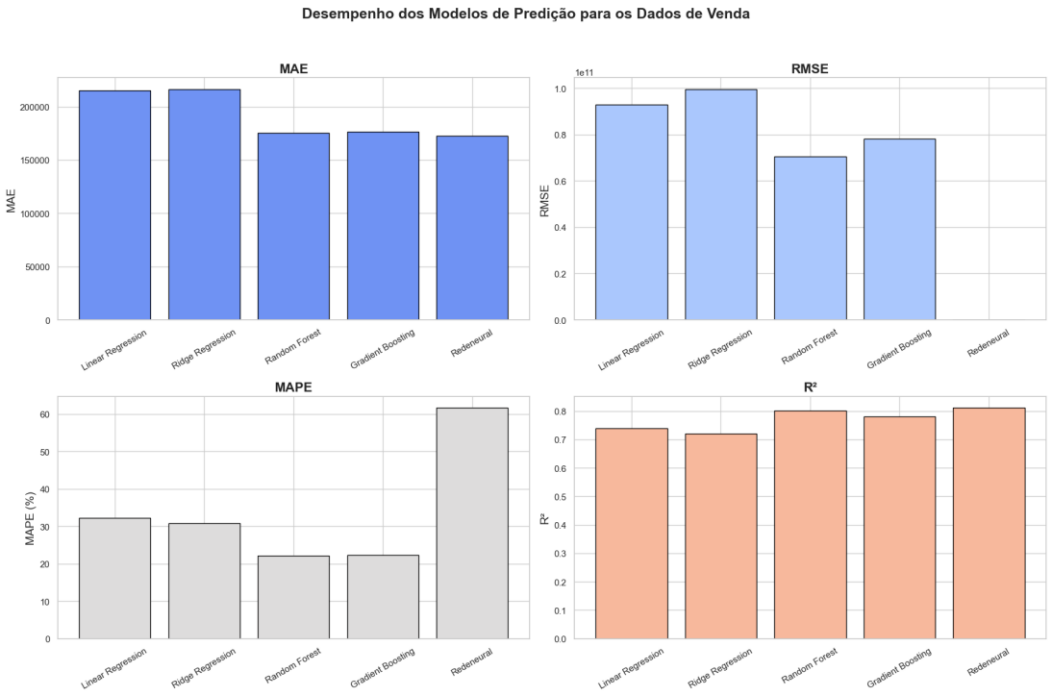

Figura 16 - Resultados dos modelos de venda para os dados de validação

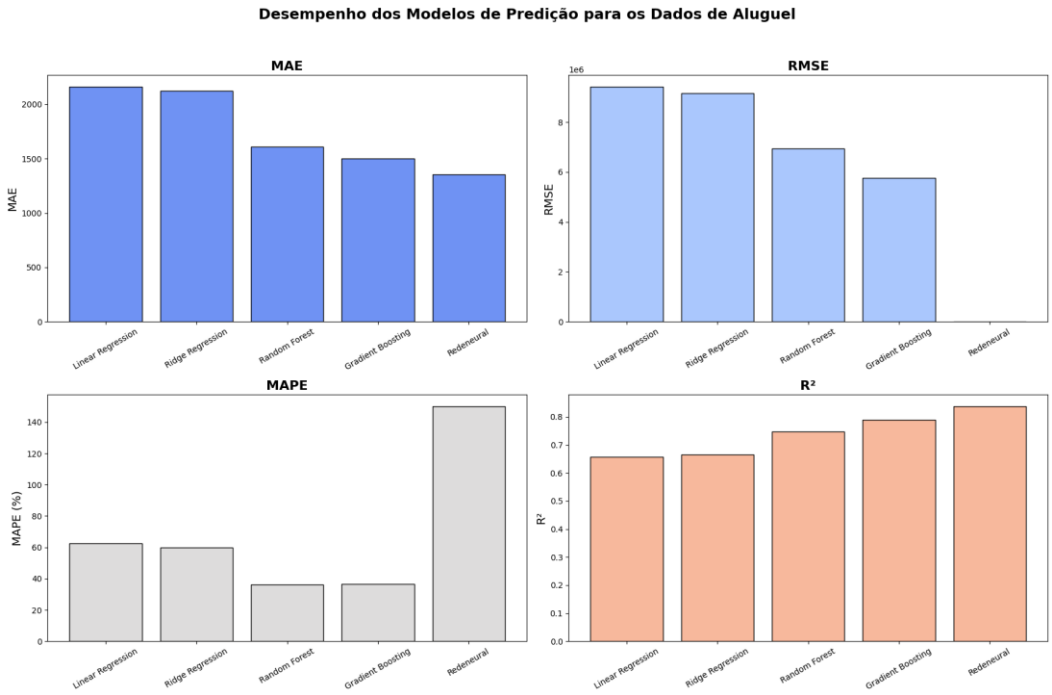

Figura 17 - Resultados dos modelos de aluguel para os dados de validação

No entanto, a escolha do modelo ideal não deve se basear apenas na métrica de desempenho, mas também em fatores como interpretabilidade, tempo de treinamento e aplicabilidade prática.

Os modelos de Regressão Linear e Ridge, apesar de apresentarem desempenho inferior, são alternativas interessantes para cenários em que a interpretabilidade das variáveis é essencial. Esses modelos permitem uma melhor compreensão dos fatores que influenciam os preços dos imóveis, sendo úteis para análises exploratórias e explicativas.

O Random Forest e o Gradient Boosting demonstraram alta capacidade preditiva, sendo competitivos com a Rede Neural. O Random Forest, em particular, mostrou-se robusto a outliers e variáveis irrelevantes, enquanto o Gradient Boosting, com seu aprendizado sequencial, conseguiu reduzir significativamente os erros de predição. Entretanto, ambos os modelos apresentam maior custo computacional em comparação com as regressões lineares.

Portanto, considerando a necessidade de um modelo altamente preditivo para este estudo, a Rede Neural foi selecionada como a melhor opção. Seu desempenho superior e capacidade de adaptação a padrões complexos justificam essa escolha.

5 CONCLUSÃO

O presente estudo demonstrou a importância da escolha adequada de modelos preditivos para a precificação de imóveis. Através da comparação entre diferentes abordagens, verificou-se que modelos mais avançados, como a Rede Neural e o Gradient Boosting, apresentam um desempenho significativamente superior em relação às abordagens mais simples, como a Regressão Linear e Ridge. No entanto, a escolha do modelo ideal deve considerar não apenas a precisão das previsões, mas também fatores como interpretabilidade, custo computacional e aplicabilidade ao contexto prático.

A Rede Neural, como modelo selecionado, mostrou-se altamente eficaz na captura de padrões complexos, garantindo previsões mais precisas. Sua aplicação pode contribuir significativamente para o desenvolvimento de soluções automatizadas de precificação de imóveis, permitindo maior assertividade em decisões de compra e venda.

Dessa forma, este trabalho reforça a importância do uso de técnicas avançadas de aprendizado de máquina no mercado imobiliário, proporcionando insights valiosos para profissionais da área e contribuindo para a evolução das ferramentas de análise e predição de preços.

6 RECOMENDAÇÃO PARA FUTURAS PESQUISAS

[a completar]

7 REFERÊNCIAS

Boehmke, B., & Greenwell, B. (2020). *Hands-On Machine Learning with R.* CRC Press.

Chugh, V. (setembro de 2024). *10 principais algoritmos de aprendizado de máquina e seus casos de uso*. Fonte: datacamp: https://www.datacamp.com/pt/blog/top-machine-learning-use-cases-and-algorithms

Crepaldi, R. (Setembro de 2024). “Logo na minha vez, ficou mais caro”: Geração Z enfrenta desafios para comprar primeiro imóvel. (Exame, Ed.) Fonte: https://exame.com/mercado-imobiliario/logo-na-minha-vez-ficou-mais-caro-geracao-z-enfrenta-desafios-para-comprar-primeiro-imovel/

Datacamp. (30 de Julho de 2024). *Datacamp*. Fonte: Um guia abrangente para a validação cruzada K-Fold: https://www.datacamp.com/pt/tutorial/k-fold-cross-validation

Gutierrez, D. (2015). *Machine Learning and Data Science: An Introduction to Statistical Learning Methods with R.* Basking Ridge: Technics Publications.

Hughes, B., Wallis, R., & Bishop, C. (March de 2022). Yearning for machine learning: applications for the classification.

IBGE. (2019). Percepção do Estado de Saúde, Estilo de Vida, Doenças Crônicas e Saúde Bucal. Fonte: https://www.pns.icict.fiocruz.br/wp-content/uploads/2021/02/liv101764.pdf

Janiesch, C., Zschech, P., & Heinrich, K. (Outubro de 2020). Machine learning and deep learning.

Jiawei, H., Micheline, K., & Jian, P. (2012). *Data Mining: Concepts and Techniques.* Massachusetts: Morgan Kaupman.

Joel, G. (2016). *Data Science do Zero: Primeiras Regras com Python.* Rio de Janeiro: Alta Books.

Learn, S. (s.d.). *Scikit Learn*. Fonte: Cross-validation: evaluating estimator performance: https://scikit-learn.org/1.5/modules/cross_validation.html

Learning, M. (4 de outubro de 2024). *Machine Learning*. Fonte: Grid Search: https://machinelearning.org.in/grid-search/

Maraccini, G. (Setembro de 2024). *Diagnóstico precoce e redução de riscos: como IA pode ser usada na medicina*. Fonte: CNN Brasil: https://www.cnnbrasil.com.br/saude/diagnostico-precoce-e-reducao-de-riscos-como-ia-pode-ser-usada-na-medicina/

Miller, A. (2022). *Neurocomputing.*

MuleSoft. (s.d.). *MuleSoft*. Fonte: MuleSoft: https://www.mulesoft.com/pt/resources/api/what-is-an-api

Muller, A., & Guido, S. (2016). *Introduction to Machine Learning with Python.*

OLX, G. (s.d.). *grupo OLX - Nossas Marcas*. Fonte: Nossas Marcas: https://olxbrasil.com.br/nossas-marcas/zap/

Passos, I. P. (dezembro de 2024). *data rio*. Fonte: data rio: https://www.data.rio/

Prefeitura, R. d. (fevereiro de 2023). *Prefeitura.Rio*. Fonte: CONHEÇA O INSTITUTO: https://ipp.prefeitura.rio/home/conheca-o-instituto/

Raschka, S. (2015). *Python Machine Learning: Unlock deeper insights into machine learning with this.* Birmingham: Packt Publishing.

Raschka, S., & Mirjalili, V. (2019). *Python Machine Learning.* Dezembro: Packt Publishing LTd.

Reynaldo, C. (1997). Regressão "Ridge": Um Método Alternativo para o Mal Condicionamento da Matriz das Regressoras .

Rosa, B. (Outubro de 2024). *Indústria automotiva acelera com IA e conectividade de olho no futuro dos carros autônomos e personalizados*. Fonte: O Globo:

https://oglobo.globo.com/economia/tecnologia/noticia/2024/10/23/industria-automotiva-acelera-com-ia-e-conectividade-de-olho-no-futuro-dos-carros-autonomos-e-personalizados.ghtml

Sapien. (26 de novembro de 2024). *Sapien*. Fonte: Grid Search: https://www.sapien.io/glossary/definition/grid-search

Shalev-Shwartz, S., & Ben-David, S. (2014). Understanding Machine Learning: From Theory to Algorithms. *Cambridge University Press.*

Sugiyama, M. (2016). *Introduction to Statistical Learning.*

TechTarget. (Março de 2023). *TechTarget*. Fonte: computerweekly: https://www.computerweekly.com/br/definicoe/O-que-e-e-como-funciona-a-web-scraping

Tecnologia, B. (setembro de 2018). *Bix Tecnologia*. Fonte: Outliers: Descubra o que são e como contorná-los em sua análise de dados: https://bixtecnologia.com.br/o-que-sao-outliers/

Tecnologia, G. d. (Abril de 2024). *Glossário de Tecnologia*. Fonte: O que é Thread e para que serve?: https://programae.org.br/termos/glossario/o-que-e-thread-e-para-que-serve/

UFF. (Dezembro de 2023). Introdução ao Machine Learning - I. *Laboratório de Estatística*.