



INSTITUTO TECNOLÓGICO DE AERONÁUTICA

CMC-16 — PRÁTICAS DE BANCO DE DADOS

---

**Exame — *House prices***

---

*Professor:*

Felipe A. N. Verri

*Alunos:*

Antônio Silveira

Leonado Friedrich

Lucca Haddad

Rafael Hoffmann

9 de julho de 2024

## Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Metodologia</b>	<b>2</b>
2.1	Descrição dos Dados . . . . .	2
2.2	Pré-processamento dos Dados . . . . .	2
2.3	Exploração de Dados . . . . .	2
2.4	Seleção de Modelos . . . . .	3
<b>3</b>	<b>Modelagem e Avaliação</b>	<b>3</b>
3.1	Implementação dos Modelos . . . . .	3
3.2	Avaliação dos Modelos . . . . .	4
<b>4</b>	<b>Resultados e Discussão</b>	<b>4</b>
4.1	Interpretação dos Resultados . . . . .	4
<b>5</b>	<b>Aplicação</b>	<b>5</b>
<b>6</b>	<b>Conclusão</b>	<b>5</b>

## 1 Introdução

A previsão de preços de imóveis é uma tarefa crucial no mercado imobiliário, impactando desde compradores e vendedores até investidores e corretores. Utilizando o dataset do Kaggle "House Prices: Advanced Regression Techniques", este artigo tem como objetivo desenvolver um modelo preditivo para estimar os preços das casas com base em características específicas dos imóveis.

## 2 Metodologia

### 2.1 Descrição dos Dados

O dataset utilizado contém informações detalhadas sobre propriedades residenciais em Ames, Iowa, incluindo 79 variáveis que descrevem aspectos como qualidade geral, área útil, número de quartos, entre outros. O objetivo é prever a variável *SalePrice*, que representa o preço de venda dos imóveis.

### 2.2 Pré-processamento dos Dados

O pré-processamento é uma etapa essencial para garantir a qualidade dos dados utilizados no treinamento do modelo. As ações realizadas incluem:

- Tratamento de valores ausentes. Os valores numéricos ausentes foram preenchidos com o valor da média e os valores categóricos ausentes foram preenchidos com o valor mais frequente.
- Transformação de variáveis categóricas em numéricas. As variáveis com menos de 10 valores únicos foram tratadas para receber um equivalente numérico para cada um de seus valores únicos.
- Normalização dos dados utilizando *MinMaxScaler*. Os dados foram normalizados para possibilitar a utilização uniforme dos valores para os cálculos de predição.
- Aplicado log no target. O log aplicado no valor de venda das casas busca minimizar a degradação do erro causada por erro em casas de maior valor, onde o erro absoluto teria muito impacto nas métricas.

### 2.3 Exploração de Dados

Uma análise exploratória foi realizada para identificar correlações entre as variáveis e a variável alvo (*SalePrice*). Com isso, foram listadas as features de maior correlação com o target.

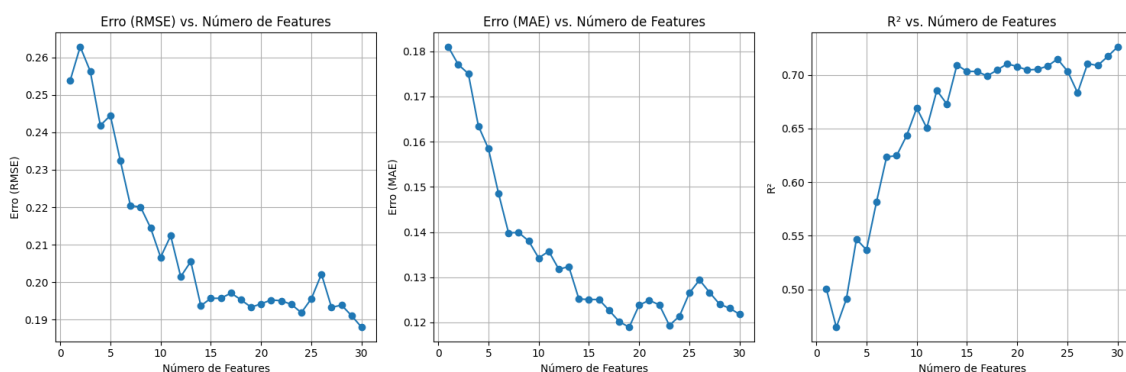


Figura 1: Gráfico desempenho x nº de features

Com a lista das features mais correlacionadas com o target, foram realizados testes com uma versão inicial do modelo de Machine Learning para verificar o erro final de cada conjunto de features, iniciando com a feature de maior correlação e, de maneira iterativa, sendo adicionada a próxima feature na lista de maior correlação. Com isso chegou-se ao seguinte resultado:

Como se desejava o menor número de features possível, para aumentar a usabilidade, sem que houvesse uma degradação muito grande do desempenho, observando qualitativamente os resultados, optou-se pelo número de 14 features.

## 2.4 Seleção de Modelos

A escolha do modelo de ML foi o XGBoost. A escolha final recaiu sobre o XGBoost devido ao seu desempenho superior em tarefas de regressão e sua capacidade de lidar com dados complexos.

# 3 Modelagem e Avaliação

## 3.1 Implementação dos Modelos

O modelo XGBoost foi treinado utilizando as variáveis mais significativas identificadas na análise exploratória. O código de treinamento foi implementado em um notebook Jupyter (*model\_training.ipynb*), onde foram ajustados os hiperparâmetros para otimização do desempenho.

## 3.2 Avaliação dos Modelos

A avaliação do modelo foi realizada utilizando métricas como *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE) e  $R^2$ , variando os hiperparâmetros em uma busca exploratória para verificar o melhor conjunto de hiperparâmetros considerando cada uma das métricas. A validação cruzada foi empregada para garantir a robustez dos resultados, e os modelos foram comparados quanto ao seu desempenho preditivo.

# 4 Resultados e Discussão

## 4.1 Interpretação dos Resultados

O modelo XGBoost com seus hiperparâmetros definidos pela busca exploratória foi então treinado e aplicado ao dataset de validação. Com isso, os resultados apresentados foram os seguintes:

```
1 Metrics for MAE:
2 Best parameters: {'colsample_bytree': 0.4, 'gamma': 0.1,
3   'learning_rate': 0.1, 'max_depth': 6, 'min_child_weight': 1, 'n_estimators': 200, 'reg_alpha': 0.5, '
4   reg_lambda': 0.7, 'subsample': 0.5}
5 Log Mean Absolute Error (MALE): 0.0959910453184553
6 Log Mean Squared Error (MSLE): 0.019443165274490134
7 Log Root Mean Squared Error (RMSLE): 0.13943875097866495
8 R-squared ( $R^2$ ): 0.8816015234128173
9
10 Metrics for RMSE:
11 Best parameters: {'colsample_bytree': 0.5, 'gamma': 0.1,
12   'learning_rate': 0.1, 'max_depth': 6, 'min_child_weight': 1, 'n_estimators': 300, 'reg_alpha': 0.5, '
13   reg_lambda': 0.8, 'subsample': 0.5}
14 Log Mean Absolute Error (MALE): 0.09544310176557419
15 Log Mean Squared Error (MSLE): 0.01946295450319423
16 Log Root Mean Squared Error (RMSLE): 0.13950969322306686
17 R-squared ( $R^2$ ): 0.8814810175950494
18
19 Metrics for  $R^2$ :
20 Best parameters: {'colsample_bytree': 0.5, 'gamma': 0.1,
21   'learning_rate': 0.1, 'max_depth': 6, 'min_child_weight': 1, 'n_estimators': 300, 'reg_alpha': 0.5, '
22   reg_lambda': 0.8, 'subsample': 0.5}
23 Log Mean Absolute Error (MALE): 0.09544310176557419
```

```
18   Log Mean Squared Error (MSLE): 0.01946295450319423
19   Log Root Mean Squared Error (RMSLE): 0.13950969322306686
20   R-squared (R^2): 0.8814810175950494
```

Os resultados finais mostram que, apesar da redução do número de features, o erro logarítimo foi baixo, representando um erro proporcional de aproximadamente 10% do valor de venda. Observa-se ainda um valor de 0.88 de  $R^2$ , o que comprova que o modelo é explicativo em relação aos dados previstos

## 5 Aplicação

Com os dados e o modelo prontos, pôde-se então desenvolver a aplicação capaz de prever o valor de venda do imóvel de acordo com as features selecionadas. A aplicação foi desenvolvida ainda com a possibilidade de lidar com valores ausentes, para os casos de preenchimento incorreto ou ausência de dados. A aplicação foi desenvolvida em Flask (backend) e react (frontend).

## 6 Conclusão

Este artigo desenvolveu e avaliou um modelo preditivo eficiente para estimar os preços de imóveis, utilizando técnicas avançadas de *machine learning* e um robusto processo de pré-processamento e análise de dados. Os resultados obtidos pelo modelo preditivo foram considerados satisfatórios. Isso se deve às boas técnicas envolvidas no pré-processamento dos dados e na avaliação e treinamento do modelo de ML.

Além disso, uma aplicação de utilização simples foi desenvolvida, sem a necessidade de todos os dados envolvidos no dataset inicial, tendo uma usabilidade agradável, considerando, inclusive a possibilidade de ausência de dados para realizar a predição do valor.