UNIVERSIDADE REGIONAL INTEGRADA DO ALTO URUGUAI E DAS MISSÕES
PRÓ-REITORIA DE ENSINO, PESQUISA, EXTENSÃO E PÓS GRADUAÇÃO
DEPARTAMENTO DE ENGENHARIA E CIÊNCIA DA COMPUTAÇÃO
CURSO DE CIÊNCIA DA COMPUTAÇÃO

CARLOS EDUARDO BUENO

TRABALHO FINAL

MINERAÇÃO DE DADOS - CLASSIFICAÇÃO DE CARROS

ERECHIM
2025

1. INTRODUÇÃO

Este projeto tem como objetivo aplicar técnicas de mineração de dados na classificação de veículos, utilizando como base um conjunto de dados reais de carros. O objetivo principal é prever se um veículo é considerado “caro” ou “barato”, com base em suas características técnicas e de mercado. A classificação é feita a partir da mediana do preço (MSRP). Para isso, utilizam-se bibliotecas amplamente adotadas na área de Ciência de Dados, como pandas, scikit-learn, matplotlib e seaborn.

O projeto foi desenvolvido em Python e está disponível publicamente no GitHub:

Repositório: https://github.com/Carlos10198/MineracaoDeDados

2. DESCRIÇÃO DO DATASET

Fonte: Kaggle – Car Features and MSRP Dataset (https://www.kaggle.com/datasets/CooperUnion/cardataset)

- Total de registros: 11.914

Colunas utilizadas na análise:

- Make (Marca), Model (Modelo), Year (Ano), Engine Fuel Type (Tipo de combustível), Engine HP (Potência do motor), Engine Cylinders (Cilindros), Transmission Type (Tipo de transmissão), Driven_Wheels (Tração), Number of Doors (Número de portas), Vehicle Size (Tamanho do veículo), Vehicle Style (Estilo do veículo), highway MPG / city mpg (Consumo), MSRP (Preço sugerido pelo fabricante).

-A variável-alvo “categoria_preco” foi definida como:

caro se o preço do veículo for acima da mediana

barato caso contrário

3. PRÉ-PROCESSAMENTO

Para garantir a qualidade dos dados utilizados na análise, foi realizado um processo de pré-processamento com diversas etapas. Inicialmente, a coluna de preço dos veículos (MSRP) foi convertida para um formato numérico, tratando possíveis inconsistências ou valores inválidos. Em seguida, todas as linhas com dados ausentes nas colunas relevantes foram removidas, assegurando que apenas registros completos fossem utilizados no modelo. Após essa limpeza, foi criada a variável categoria_preco, que classifica os veículos como “caro” ou “barato” com base na mediana dos preços. Para tornar as variáveis categóricas compatíveis com os algoritmos de machine learning, foi aplicada a codificação com LabelEncoder. Por fim, os dados foram divididos em dois conjuntos: 70% destinados ao treinamento do modelo e 30% reservados para testes, utilizando a função train_test_split da biblioteca scikit-learn.

4. MODELO E AVALIAÇÃO

Para a tarefa de classificação, foi escolhido o algoritmo Random Forest Classifier, uma técnica de aprendizado supervisionado baseada em múltiplas árvores de decisão. Esse algoritmo é amplamente utilizado devido à sua robustez, capacidade de lidar com dados heterogêneos e baixo risco de overfitting. A avaliação do modelo foi realizada por meio de métricas estatísticas clássicas: Acurácia, Precisão, Revocação e F1-Score. Além dessas métricas, foi gerada uma matriz de confusão utilizando a biblioteca Seaborn, que possibilita uma visualização clara da performance do modelo ao distinguir corretamente as classes "caro" e "barato" no conjunto de testes.

5. RESULTADOS

Os resultados obtidos demonstraram que o modelo Random Forest apresentou desempenho satisfatório, com uma acurácia global de 93%. Após o pré-processamento, o total de registros válidos utilizados na análise foi de 11.812. No conjunto de testes, a distribuição das classes ficou dividida entre 1.736 veículos classificados como “caro” e 1.808 como “barato”. A matriz de confusão gerada ilustra

o número de acertos e erros cometidos pelo modelo ao classificar os veículos. Observou-se uma boa capacidade do modelo em generalizar os padrões, mesmo diante de diferentes marcas, estilos e configurações de veículos, evidenciando sua eficácia na tarefa proposta.

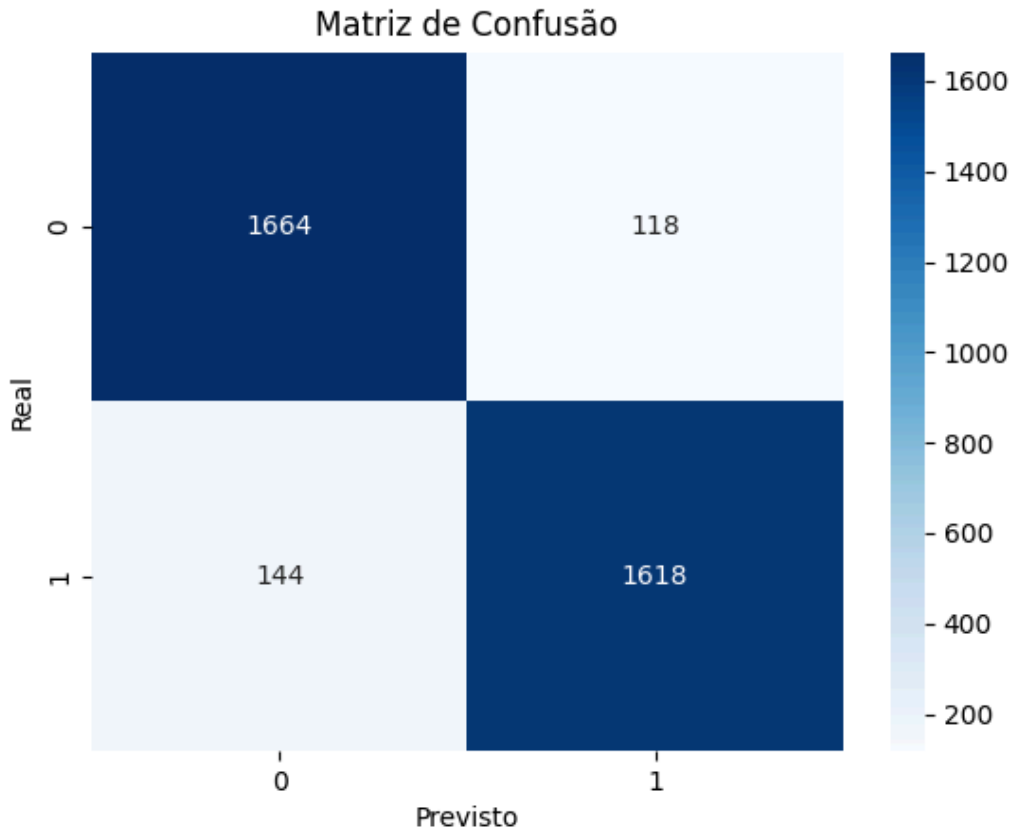

Figura 1 - Matriz de Confusão (Autor Próprio, 2025)

```
Colunas disponíveis: Index(['Make', 'Model', 'Year', 'Engine Fuel Type', 'Engine HP',
       'Engine Cylinders', 'Transmission Type', 'Driven_Wheels',
       'Number of Doors', 'Market Category', 'Vehicle Size', 'Vehicle Style',
       'highway MPG', 'city mpg', 'Popularity', 'MSRP'],
      dtype='object')

Colunas utilizadas na análise (com MSRP): 14 colunas
Registros com dados faltantes (nas colunas usadas): 102
Registros utilizados após limpeza (linhas completas nas colunas usadas): 11812

Quantidade de carros classificados pelo modelo:
barato: 1808
caro: 1736

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.92      0.93      0.93      1782
           1       0.93      0.92      0.93      1762

    accuracy                           0.93      3544
   macro avg       0.93      0.93      0.93      3544
weighted avg       0.93      0.93      0.93      3544
```

Figura 2 - Resultado Terminal (Autor Próprio, 2025)

6. CONCLUSÃO

A aplicação de Random Forest permitiu uma classificação eficaz dos veículos quanto ao seu preço. O projeto demonstra como técnicas básicas de mineração de dados podem ser utilizadas com sucesso em problemas reais e destaca a importância da preparação dos dados para alcançar bons resultados.

7. REFERÊNCIAS

- https://www.kaggle.com/datasets/CooperUnion/cardataset

- Documentação do scikit-learn: https://scikit-learn.org/

- Documentação do seaborn: https://seaborn.pydata.org/

- Python Software Foundation: https://www.python.org/