

Comparação de Algoritmos de Categorização

Categorizar um veículo a partir das suas informações

Comparison of Categorization Algorithms

Categorize a vehicle based on its information

Martinho Caeiro - 23917 — Paulo Abade - 23919

Instituto Politécnico de Beja

Escola Superior de Tecnologia e Gestão

Beja, Portugal

23917@stu.ipbeja.pt — 23919@stu.ipbeja.pt

Resumo—Este artigo apresenta um estudo para a comparação entre algoritmos de categorização de veículos. O objetivo de cada um dos algoritmos é categorizar um veículo a partir das suas informações, sendo que este veículo será categorizado consoante o seu país de origem. Foi escolhido este tema para facilitar a nossa compreensão sobre o assunto e tornar mais agradável o estudo destes algoritmos. O estudo foi realizado no *Orange* (Orange, 2024) com base em algoritmos de aprendizagem supervisionada, nomeadamente o algoritmo *Binary Tree*, algoritmo *Random Forest*, algoritmo *Logistic Regression* e algoritmo *Neural Network*. Para fazer a comparação destes, foi utilizado um dataset, com cerca de 400 entradas, onde possui informações de veículos de diferentes países e este foi utilizado nos diferentes algoritmos para treinar e testar os mesmos. Este dataset, já foi alterado no primeiro trabalho da Disciplina de *Sistemas de Informação*, sendo ligeiramente diferente, por já ter sido tratado. Os resultados obtidos foram comparados e analisados para perceber qual o algoritmo que melhor categoriza um veículo a partir das suas informações. A comparação dos algoritmos foi feita com base na sua precisão, sensibilidade e especificidade. O estudo também abordou as limitações de cada algoritmo. Aplicações práticas incluem sistemas de recomendação de veículos, análise de dados ou apoio à criação de estratégias de marketing em diferentes regiões. A análise detalhada do comportamento dos algoritmos pode ser útil para investigadores e profissionais que desejam otimizar a categorização de grandes volumes de dados automóveis.

Palavras-chave—algoritmos; veículos; categorização; aprendizagem supervisionada; árvore binária; random forest; precisão; sensibilidade; especificidade; orange; datamining; machine learning; kaggle.

I. INTRODUÇÃO

A categorização de veículos é uma área relevante em aplicações práticas, como a otimização de cadeias de fornecimento, a personalização de ofertas comerciais ou o desenvolvimento de sistemas inteligentes de transporte. Estudos recentes demonstram que algoritmos de aprendizagem supervisionada podem oferecer soluções rápidas e eficazes para problemas de classificação, mas a escolha do algoritmo adequado depende de vários fatores, como o tipo de dados e o objetivo final. Este estudo procura preencher essa lacuna, analisando não apenas a precisão dos modelos, mas também os seus comportamentos sob diferentes métricas de avaliação. Para isso, foi utilizado um dataset com informações de veículos de diferentes países, onde o objetivo é determinar o país de origem dado as informações do veículo. Este dataset foi utilizado para treinar e testar

os diferentes algoritmos de categorização, nomeadamente os algoritmos *Tree*, *Random Forest*, *Logistic Regression* e *Neural Network*.

II. DATASET

O dataset "Car information dataset" (Elmetwally, 2023) utilizado neste estudo foi retirado do site *Kaggle* e contém informações de veículos de diferentes países. Estas informações incluem a marca/modelo, a economia de combustível, o número de cilindros, a cilindrada, a potência, o peso, a aceleração, o ano de fabrico e o país de origem. Este dataset possui cerca de 400 entradas em cada uma das colunas. Antes de aplicar os algoritmos, foi realizado um extenso pré-processamento do dataset, que incluiu a normalização de valores numéricos e a codificação de atributos categóricos, como a marca do veículo. A análise exploratória dos dados revelou uma distribuição não uniforme entre os diferentes países de origem, sendo os EUA responsáveis pela maior parte das entradas. Além disso, foi descartado o ano de fabrico e foi utilizada uma validação cruzada de 10 vezes para garantir a consistência dos resultados. Este procedimento foi adotado para reduzir a variabilidade e melhorar a fiabilidade das métricas.

III. METODOLOGIA

A metodologia adotada neste estudo envolveu as seguintes etapas principais:

- 1) **Definição do problema:** A categorização dos veículos foi definida como uma tarefa de classificação, com o país de origem como variável alvo.
- 2) **Seleção dos algoritmos:** Foram escolhidos algoritmos representativos de diferentes abordagens, como árvores de decisão, regressão e redes neurais.
- 3) **Preparação dos dados:** Foi descartado o ano de fabrico, e as variáveis foram normalizadas para melhorar a performance dos algoritmos.
- 4) **Treino dos modelos:** Cada algoritmo foi treinado utilizando um conjunto de treino com validação cruzada.
- 5) **Teste dos modelos:** Os modelos foram testados com um conjunto de teste para avaliar a sua capacidade de generalização, evitando assim bias e outliers.

- 6) **Avaliação dos modelos:** As métricas de desempenho (precisão, recall, AUC, entre outras) foram calculadas para comparar os algoritmos.
- 7) **Análise dos resultados:** Os resultados foram analisados de forma qualitativa e quantitativa, destacando os pontos fortes e fracos de cada modelo.

IV. ALGORITMOS DE DECISÃO

Nesta secção, vamos apresentar os diferentes algoritmos de decisão utilizados para a categorização dos veículos. Cada algoritmo foi avaliado numa matriz de confusão, que compara as previsões do modelo com os valores reais. As métricas que podemos retirar diretamente da matriz são:

- **Verdadeiros Positivos:** Número de observações corretamente classificadas como positivas, podem ser observadas na diagonal principal da matriz.
- **Falsos Positivos:** Número de observações incorretamente classificadas como positivas, são a soma da coluna exceto a pertencente à diagonal principal.
- **Verdadeiros Negativos:** Número de observações corretamente classificadas como negativas, são a soma de todas as observações exceto a linha e coluna da classe em questão.
- **Falsos Negativos:** Número de observações incorretamente classificadas como negativas, são a soma da linha exceto a pertencente à diagonal principal.

A. Tree (Wikipedia, 2025d)

É um modelo baseado numa estrutura hierárquica em forma de árvore, como é possível ver na tabela I. Cada nó representa uma condição ou regra (geralmente um atributo do conjunto de dados), e os ramos dividem os dados com base nessa regra. O objetivo é chegar a uma decisão ou classificação no final de cada ramo (folha). É simples, interpretável e útil para problemas de classificação e regressão.

| Atual/Previsão | Europeu | Japão | EUA | Total |
|----------------|-----------|-----------|------------|------------|
| Europeu | 29 | 9 | 10 | 48 |
| Japão | 16 | 34 | 5 | 55 |
| EUA | 12 | 6 | 154 | 172 |
| Total | 57 | 49 | 169 | 275 |

Tabela I. Matriz de Confusão do Algoritmo Tree

B. Random Forest (Wikipedia, 2025c)

Este algoritmo é um conjunto de árvores, como é possível ver na tabela II. Cria várias árvores independentes, cada uma treinada com um subconjunto dos dados e dos atributos selecionados aleatoriamente. No final, combina os resultados para melhorar a precisão e reduzir o risco de overfitting, comparado a uma única árvore.

| Atual/Previsão | Europeu | Japão | EUA | Total |
|----------------|-----------|-----------|------------|------------|
| Europeu | 25 | 16 | 7 | 48 |
| Japão | 12 | 34 | 9 | 55 |
| EUA | 11 | 7 | 154 | 172 |
| Total | 48 | 57 | 170 | 275 |

Tabela II. Matriz de Confusão do Algoritmo Random Forest

C. Logistic Regression (Wikipedia, 2025a)

Apesar do nome, é um método usado principalmente para classificação, como é possível ver na tabela III. Modela a probabilidade de um resultado pertencente a uma classe específica, usando uma função logística. É simples, rápido e eficaz em problemas de classificação binária, embora também possa ser estendido para múltiplas classes.

| Atual/Previsão | Europeu | Japão | EUA | Total |
|----------------|-----------|-----------|------------|------------|
| Europeu | 25 | 18 | 5 | 48 |
| Japão | 12 | 34 | 9 | 55 |
| EUA | 4 | 14 | 154 | 172 |
| Total | 41 | 66 | 168 | 275 |

Tabela III. Matriz de Confusão do Algoritmo Logistic Regression

D. Neural Network (Wikipedia, 2025b)

Inspiradas pelo cérebro humano, consistem em camadas de "neurónios" interligados, como é possível ver na tabela IV. Cada neurónio recebe entradas, aplica uma ponderação e uma função de ativação, e passa o resultado para os neurónios da camada seguinte. São altamente versáteis e podem lidar com problemas complexos, como reconhecimento de imagens ou processamento de linguagem natural, mas requerem mais dados e poder computacional.

| Atual/Previsão | Europeu | Japão | EUA | Total |
|----------------|-----------|-----------|------------|------------|
| Europeu | 20 | 22 | 6 | 48 |
| Japão | 12 | 33 | 10 | 55 |
| EUA | 7 | 11 | 154 | 172 |
| Total | 39 | 66 | 170 | 275 |

Tabela IV. Matriz de Confusão do Algoritmo Neural Network

V. COMPARAÇÕES FINAIS

Como podemos visualizar na tabela V, o algoritmo *Random Forest* obteve os melhores resultados avaliando a área sobre a curva do gráfico ROC (AUC), e o algoritmo *Tree* obteve os melhores resultados em termos de acurácia de classificação (CA), precisão e sensibilidade. O algoritmo *Logistic Regression* obteve resultados semelhantes ao *Random Forest*, enquanto o *Neural Network* apresentou o pior desempenho. Outra observação que é importante destacar é que o algoritmo *Logistic Regression* obteve um desempenho superior no contexto de F1 Score e MCC, sendo isto mais

importante para problemas de classificação, mostrando no geral um melhor desempenho que os algoritmos de *Random Forest* e *Neural Network*.

| Algoritmo | AUC | CA | F1 | Precision | Recall | MCC |
|---------------------|-------|-------|-------|-----------|--------|-------|
| Tree | 0.833 | 0.789 | 0.792 | 0.798 | 0.789 | 0.613 |
| Logistic Regression | 0.909 | 0.775 | 0.777 | 0.783 | 0.775 | 0.587 |
| Random Forest | 0.921 | 0.775 | 0.776 | 0.777 | 0.775 | 0.584 |
| Neural Network | 0.900 | 0.753 | 0.753 | 0.756 | 0.753 | 0.544 |

Tabela V. Comparação de Resultados dos Algoritmos

Estas métricas são importantes para avaliar o desempenho dos algoritmos, e a sua interpretação pode variar consoante o contexto do problema. A aplicação *Orange* forneceu automaticamente estas métricas, porém detalhando cada uma delas, obtemos o seguinte:

A. Acurácia de Classificação (CA)

A CA, como podemos visualizar na equação 1, é a proporção de observações corretamente classificadas pelo modelo. É uma métrica geral de desempenho, mas pode ser enganadora em conjuntos de dados desequilibrados.

$$CA = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Onde TN é o número de verdadeiros negativos e FN é o número de falsos negativos.

B. Precisão

A precisão, como podemos visualizar na equação 2, é a proporção de observações corretamente classificadas como positivas em relação ao total de observações classificadas como positivas.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Onde TP é o número de verdadeiros positivos e FP é o número de falsos positivos.

C. Sensibilidade/Recall

A sensibilidade, como podemos visualizar na equação 3, é a proporção de observações corretamente classificadas como positivas em relação ao total de observações reais positivas.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

Onde TP é o número de verdadeiros positivos e FN é o número de falsos negativos.

D. F1 Score

O F1 Score, como podemos visualizar na equação 4, é a média harmónica da precisão e da sensibilidade. É útil quando as classes estão desequilibradas, pois penaliza mais os falsos negativos e falsos positivos.

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

Onde Precision é a proporção de observações corretamente classificadas como positivas em relação ao total de observações classificadas como positivas e Recall é a proporção de observações corretamente classificadas como positivas em relação ao total de observações reais positivas.

E. Coeficiente de Correlação de Matthews (MCC)

O MCC, como podemos visualizar na equação 5, é uma métrica que varia entre -1 e 1, onde 1 indica uma previsão perfeita, 0 indica uma previsão aleatória e -1 indica uma previsão inversa.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (5)$$

Onde TP é o número de verdadeiros positivos, TN é o número de verdadeiros negativos, FP é o número de falsos positivos e FN é o número de falsos negativos.

F. Área sobre a Curva (AUC)

A AUC, como podemos visualizar na equação 6, é uma métrica que avalia a capacidade do modelo de distinguir entre classes positivas e negativas. Quanto maior o valor, melhor o desempenho do modelo. Esta métrica possui a seguinte formulação:

$$AUC = \frac{1 + TP - FP}{2} \quad (6)$$

Onde TP é o número de verdadeiros positivos e FP é o número de falsos positivos.

Na figura 1, é possível visualizar a curva ROC dos diferentes algoritmos para os veículos de origem nos EUA.

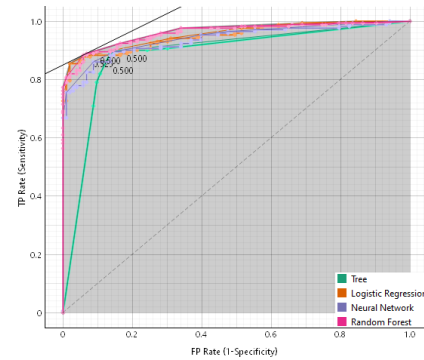


Figura 1: Curva ROC dos Algoritmos - EUA

Na figura 2, é possível visualizar a curva ROC dos diferentes algoritmos para os veículos de origem na Europa.

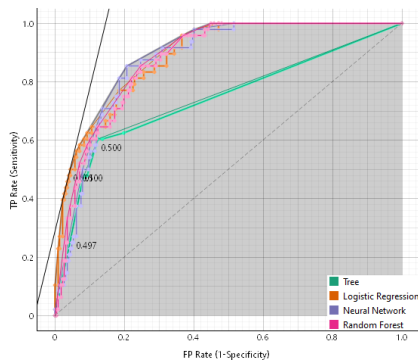


Figura 2: Curva ROC dos Algoritmos - Europa

Na figura 3, é possível visualizar a curva ROC dos diferentes algoritmos para os veículos de origem no Japão.

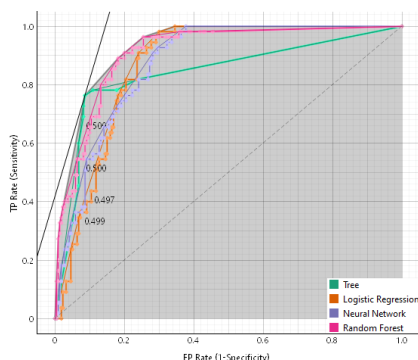


Figura 3: Curva ROC dos Algoritmos - Japão

VI. TRABALHOS RELACIONADOS

Os trabalhos "Automobile EDA" (Aboraida, 2024) e "EDA CAR INFORMATION DATA" (Salodkar, 2023) são exemplos de trabalhos relacionados com este dataset. Estes trabalhos tem uma abordagem e análise ao dataset do ponto de vista de estruturas de dados e algoritmos, que tem como objetivo analisar as diferentes características dos veículos, verificando quais os atributos dos automóveis de maior importância e com maiores relações, assim permitindo uma compreensão mais aprofundada do dataset assim garantindo um estudo mais rigoroso.

VII. CONCLUSÕES

Concluimos que embora o algoritmo *Tree* seja fácil de interpretar, apresenta limitações em datasets com alta variabilidade, onde tende a criar divisões muito específicas, resultando em overfitting. Por outro lado, o *Random Forest*, ao agregar várias árvores, resolve esse problema, mas o custo computacional aumenta significativamente. Já a *Logistic Regression*, apesar da simplicidade e robustez, pode apresentar dificuldades em modelar relações complexas entre as variáveis. Finalmente, as *Neural Networks* destacam-se pela capacidade de capturar padrões complexos, mas requerem grandes volumes de dados e longos períodos de treino, podendo ser menos práticas para

problemas pequenos. Em suma, concluímos que a escolha do algoritmo correto é de extrema importância dado que a precisão, sensibilidade e especificidade deve ser garantida e que a sua eficácia em grande escala tem um grande impacto no resultado final. No entanto, é importante também destacar algumas limitações do estudo. Primeiramente, o tamanho do dataset, com apenas 400 entradas por coluna, pode não ser representativo o suficiente para generalizações em larga escala. Além disso, fatores como o viés dos dados (mais veículos de origem norte-americana) podem ter influenciado os resultados.

REFERÊNCIAS

- Aboraida, A. (2024). *Automobile EDA* [Kaggle]. Obtido dezembro 2024, de <https://www.kaggle.com/code/ahmedaboraida/automobile-eda>
- Elmetwally, T. (2023). *Car information dataset* [Kaggle]. Obtido dezembro 2024, de <https://www.kaggle.com/datasets/tawfikelmetwally/automobile-dataset>
- Orange. (2024). *Orange* [Orange]. Obtido dezembro 2024, de <https://orangedatamining.com>
- Salodkar, V. (2023). *EDA CAR INFORMATION DATA* [Kaggle]. Obtido dezembro 2024, de <https://www.kaggle.com/code/vishweshsalodkar/eda-car-information-data>
- Wikipedia. (2025a). *Logistic regression* [Wikipedia]. Obtido janeiro 2025, de https://en.wikipedia.org/wiki/Logistic_regression
- Wikipedia. (2025b). *Neural network (machine learning)* [Wikipedia]. Obtido janeiro 2025, de [https://en.wikipedia.org/wiki/Neural_network_\(machine_learning\)](https://en.wikipedia.org/wiki/Neural_network_(machine_learning))
- Wikipedia. (2025c). *Random forest* [Wikipedia]. Obtido janeiro 2025, de https://en.wikipedia.org/wiki/Random_forest
- Wikipedia. (2025d). *Tree (abstract data type)* [Wikipedia]. Obtido janeiro 2025, de [https://en.wikipedia.org/wiki/Tree_\(abstract_data_type\)](https://en.wikipedia.org/wiki/Tree_(abstract_data_type))