DESAFIO TÉCNICO

INTRODUÇÃO

O aprendizado de máquina tem desempenhado um papel fundamental na resolução de desafios em diversos setores, oferecendo soluções inovadoras e oportunidades promissoras. A 29.ª Competição Baja SAE BRASIL representa uma oportunidade única para aplicar o conhecimento acumulado nessa revolução tecnológica. Este desafio não se limita à implementação de algoritmos, ele busca compreender integralmente a complexidade do problema e validar a robustez das soluções propostas.

DESENVOLVIMENTO

No início do desafio, foi entregue um conjunto de dados composto por diversas planilhas e parâmetros destinados à avaliação, treinamento e teste do modelo. O primeiro passo foi conduzir uma avaliação detalhada, analisando os parâmetros e atributos disponíveis. Durante essa análise, identificou-se a presença de campos vazios e parâmetros que não exerciam influência significativa em determinado tipo de prova.

Diante desse cenário, tornou-se imperativo realizar um processo de pré-processamento nos dados. Essa etapa consistiu em uma análise detalhada, visando identificar os parâmetros verdadeiramente relevantes por meio de uma análise técnica. De acordo com Hasan [16], uma maneira eficiente e frequentemente implementada é tratar os campos vazios com a média da coluna, portanto, os dados foram tratados utilizando essa técnica, preenchendo-os com a média das colunas. Nos casos em que não era possível calcular essa média, optamos por preencher o dado faltante com zero. Por meio da biblioteca Pandas, do Python, essas operações foram executadas garantindo eficiência e precisão na manipulação de dados.

Após o pré-processamento da planilha, foi dada sequência para a etapa de carregamento. Durante esse processo, foi efetuada a exclusão da coluna alvo, que representa a variável a ser prevista. Dessa forma, foram definidas as variáveis X (atributos preditores) e variável Y (variável alvo) para a subsequente divisão do *dataset* entre treino, teste e validação.

Com X e Y definidos, foi dada a sequência para a etapa de divisão do conjunto de dados. Utilizando o método *train_test_split* da biblioteca *sklearn,* foi definido 30% do conjunto de dados para teste e 70% para treinamento, conforme prática estabelecida por James [17].

Além disso, foi inserido o parâmetro *random state,* com um valor 42. A escolha do *random state 42* não está diretamente relacionada ao valor em si, mas a prática comum é utilizar um número constante para que diferentes execuções do código gerem sempre a mesma divisão nos conjuntos de treino e teste. Isso é fundamental para garantir a consistência dos resultados, facilitar a comparação entre diferentes modelos e evitar variações indesejadas.

A divisão do *dataset* é um passo crucial na criação de um modelo de aprendizado de máquina. Garantir que o modelo não tenha acesso aos dados de teste durante o treinamento é vital para avaliar a capacidade de generalização do modelo para novos dados, evitando viés excessivo e assegurando que o modelo não esteja ajustado apenas aos dados específicos do conjunto de treino.

O próximo passo consistiu em selecionar o modelo mais apropriado para a tarefa de classificação. Inicialmente, os dados foram analisados e a natureza do conjunto de dados foi observada para identificar o modelo mais flexível. Testes foram conduzidos com Árvore de Decisão, Regressão Logística e Random Forest, aplicados à tarefa de classificação. Após a análise desses parâmetros e com base no trabalho de Liaw e Wiener [18], o modelo escolhido foi o Random Forest, devido à sua melhor aplicabilidade.

Com o modelo selecionado, foi aplicado o método *GridSearchCV,* pertencente à biblioteca *sklearn,* que visa otimizar os hiperparâmetros com base no conjunto de dados e no modelo escolhido. Essa abordagem possibilitou a exploração de várias combinações de parâmetros, identificando assim a configuração mais eficaz do modelo para cada tipo de prova. Essa estratégia de busca sistemática contribuiu significativamente para aprimorar o desempenho do modelo, garantindo uma melhor adaptação aos dados específicos de cada avaliação.

Com o modelo devidamente ajustado, foi empregado o método fit, também integrante da biblioteca *sklearn,* a partir do conjunto de dados previamente dividido entre treino e teste, para conduzir o treinamento, que permite a assimilação e adaptação do modelo aos padrões presentes no conjunto de dados, capacitando-o para realizar previsões em novos dados com eficácia. Essa fase é essencial para garantir que o modelo adquira o conhecimento necessário e possa generalizar bem para instâncias não vistas anteriormente

Com o modelo ajustado e devidamente treinado, chegou o momento de realizar as previsões e avaliar o desempenho do modelo durante a fase de treinamento. Para tanto, foi utilizado o método *predict* da biblioteca *sklearn* para efetuar as previsões. Com base nos resultados obtidos, foi construída uma matriz de comparação entre os dados reais e os dados previstos. Essa abordagem proporcionou uma visão detalhada da performance do modelo durante o treinamento, permitindo avaliar sua capacidade de generalização e a precisão das previsões em relação aos dados reais. Essa análise é crucial para garantir a confiabilidade do modelo e identificar possíveis áreas de melhoria.

Conforme destacado por Hasan [16], uma das métricas essenciais e amplamente utilizadas para avaliar o desempenho de um modelo é a MicroF1 (Figura 1). Essa métrica é representada pela seguinte fórmula:

$$Micro\ F1 = \frac{2\ x\ Precisão\ x\ Revocação}{Precisão\ +\ Revocação}$$

Dessa forma, foi adotada a MicroF1 como parâmetro fundamental para avaliar a performance do modelo. Essa métrica é especialmente relevante, pois considera tanto a precisão quanto a sensibilidade (recall) do modelo, proporcionando uma visão abrangente da sua eficácia em relação aos dados de teste.

Após a análise das métricas do modelo, foram constatados resultados satisfatórios. Para validar a robustez do modelo, foram aplicados testes com novos dados provenientes do nosso veículo, com o intuito de avaliar a performance do modelo com dados inéditos, assegurando que ele não esteja superajustado ou subajustado.

Para avaliar o desempenho com esses novos dados, utilizamos novamente o método *predict* aplicado ao modelo já treinado. Com base nas previsões desses dados, conseguimos simular os grupos nos quais o nosso modelo previu que estaríamos. Esse processo é essencial para verificar a capacidade de generalização do modelo em situações não vistas durante o treinamento, garantindo sua aplicabilidade em cenários do mundo real. Dessa forma, conseguimos prever os grupos com base no modelo.

RESULTADOS

Os resultados provenientes das métricas e modelos são apresentados de maneira detalhada conforme Tabela 1:

	Aceleração	Velocidade	Frenagem	Tração	Manobrabilidade	Suspensão
Desempenho do Modelo ML	70%	85%	96%	66%	67%	85%
Grupo Previsto	4	3	4	4	3	3

A análise dos resultados confirma a eficácia do modelo, demonstrando uma precisão notável. Este modelo destaca-se por sua capacidade de lidar de maneira concisa com novos dados, apresentando uma notável generalização. Os resultados obtidos são otimistas e coesos, alinhando-se de maneira consistente com a realidade. Essa eficiência valida a robustez e confiabilidade do modelo, indicando seu potencial para aplicações práticas e sua capacidade de oferecer insights valiosos.

Uma análise adicional realizada foi a exibição das árvores de decisão geradas pelo modelo para conseguir ilustrar como o modelo atuou, assim como na Figura 1. Essa abordagem visual não apenas oferece uma compreensão mais tangível do funcionamento interno do modelo, mas também proporciona uma ferramenta valiosa para interpretação e validação de suas decisões.

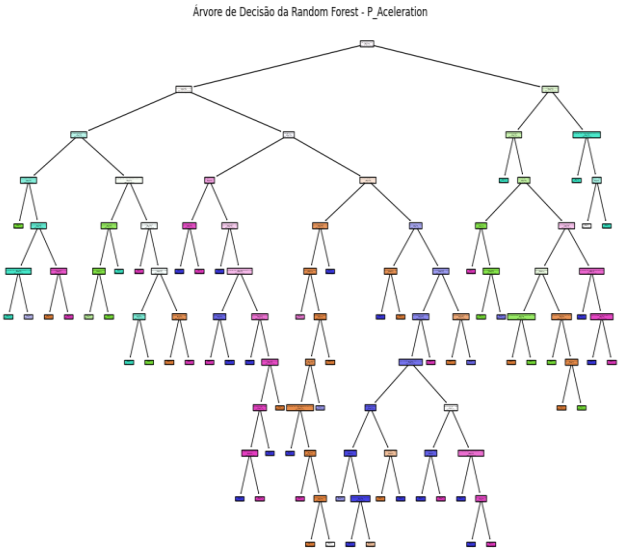

Figura 1 – Árvore de Decisão do modelo

CONCLUSÃO

A abordagem de aprendizado de máquina adotada para o desafio técnico na 29.ª Competição Baja SAE BRASIL revelou-se eficaz e promissora. Desde o pré-processamento dos dados até a seleção do modelo e a avaliação do desempenho, cada etapa foi estrategicamente conduzida para compreender e solucionar o problema de forma robusta.

O tratamento cuidadoso de campos vazios e parâmetros pouco influentes durante o pré-processamento estabeleceu uma base sólida para o desenvolvimento do modelo. A escolha do Random Forest, respaldada pela análise da natureza do conjunto de dados, demonstrou ser precisa e a otimização de hiperparâmetros através do GridSearchCV contribuiu para personalizar o modelo de acordo com as características de cada tipo de prova.

Os resultados satisfatórios da métrica MicroF1, considerando precisão e sensibilidade, validam a eficácia do modelo em dados de teste. A aplicação bem-sucedida a novos dados, provenientes do veículo, ressalta a capacidade de generalização do modelo para cenários do mundo real.

A tabela de resultados reflete a consistência e eficiência do modelo, alinhando-se de maneira otimista com a realidade. Este modelo não só atende às expectativas como também oferece uma base sólida para aplicações práticas, destacando seu potencial para fornecer *insights* valiosos. Em resumo, a abordagem adotada mostra o impacto positivo do aprendizado de máquina na resolução de desafios práticos.