**ANÁLISE PREDITIVA DO DESEMPENHO NA FÓRMULA 1 COM MACHINE LEARNING**

Luciano Franzoi Filho

Acadêmico do Curso de Engenharia de Software do Centro Universitário Cesumar – UNICESUMAR, Curitiba – PR, Abril de 2025.

**RESUMO:**

Esse artigo traz uma análise preditiva usando machine learning pra prever a pontuação dos pilotos da Fórmula 1 em corridas específicas, com base nos dados históricos de 1950 até 2020 (dataset do Kaggle). O processo envolveu autenticação via API, download automático da base, EDA, pré-processamento e dois caminhos pra treinar os modelos: um manual (com cinco algoritmos de regressão — Gradient Boosting, Random Forest, AdaBoost, LightGBM e SVR) e outro automatizado com PyCaret. Os modelos que se saíram melhor em R² foram o Gradient Boosting e o Random Forest. A análise com SHAP ajudou a entender o peso de cada variável nas previsões. Os gráficos mostraram padrões como o domínio da Mercedes na era híbrida, a regularidade da Ferrari ao longo do tempo e como a idade do piloto impacta na performance. No geral, o uso de ciência de dados nesse cenário mostra como o machine learning pode ajudar a identificar tendências, dar suporte em decisões estratégicas e aprofundar a visão sobre desempenho no automobilismo.

**Palavras-chave**: Fórmula 1; Machine Learning; Análise de Dados; Regressão; Data Science.

**1. Introdução**

A Fórmula 1 é considerada o ápice do automobilismo mundial, combinando tecnologia, engenharia de ponta, estratégia e habilidade dos pilotos. Com mais de sete décadas de história, o esporte acumula uma imensa quantidade de dados, tornando-se um campo fértil para análise com técnicas de ciência de dados. A categoria é regulamentada pela FIA (Fédération Internationale de l’Automobile), entidade responsável pelas normas e segurança do esporte.

Este artigo tem como objetivo explorar, modelar e interpretar dados históricos do Campeonato Mundial de Fórmula 1 (1950–2024) utilizando algoritmos de aprendizado de máquina. A análise visa prever a pontuação dos pilotos em corridas e identificar os principais fatores que influenciam o desempenho esportivo.

**2. Fundamentação Teórica**

O uso de aprendizado de máquina (machine learning) em análises esportivas tem crescido significativamente nos últimos anos, principalmente devido à disponibilidade de dados detalhados e ferramentas computacionais acessíveis. Em especial, algoritmos de regressão supervisionada têm sido amplamente empregados para tarefas que envolvem previsão de valores numéricos, como pontuações de jogadores, tempo de voltas, ou distância percorrida.

A regressão supervisionada parte do princípio de que existe uma relação entre um conjunto de variáveis independentes (features) e uma variável dependente (target). No contexto deste trabalho, a variável alvo é a pontuação obtida pelo piloto em uma corrida, e as variáveis independentes incluem dados como data de nascimento do piloto, equipe (construtor), posição de largada, status da corrida (abandono, finalizado, etc.), tempo da volta mais rápida e resultados de classificação.

Dentre os algoritmos usados neste projeto estão: Random Forest, Gradient Boosting, LightGBM, AdaBoost e SVR. Todos esses modelos são conhecidos por sua robustez e boa performance em tarefas de regressão, especialmente quando bem ajustados com técnicas como GridSearchCV e validação cruzada. Tais algoritmos também foram amplamente discutidos por Géron (2019), que destaca o Gradient Boosting como um dos métodos mais eficazes quando se trata de dados tabulares com múltiplas variáveis correlacionadas.

A análise também utilizou a técnica SHAP (SHapley Additive exPlanations), que é baseada em teoria dos jogos e permite quantificar o impacto de cada feature nas previsões do modelo. Isso contribui para a interpretabilidade dos resultados, um aspecto essencial em aplicações reais onde é importante compreender por que um modelo tomou determinada decisão. Lundberg e Lee (2017) demonstraram que SHAP fornece uma base matemática sólida para explicações consistentes, sendo altamente recomendada para análise de modelos de machine learning.

**3. Metodologia**

A metodologia do projeto foi dividida em etapas bem estruturadas para garantir clareza e reprodutibilidade no pipeline. A seguir, detalho o desenvolvimento de cada uma:

1. **Autenticação e Download:** Foi usado a API oficial do Kaggle com credenciais (kaggle.json), através da função autenticar\_kaggle(). O script dataset\_downloader.py cuida do download e extração automática dos arquivos CSV do dataset de Fórmula 1.
2. **Tratamento e Consolidação:** A função ajustar\_datasets() trata e consolida os dados, conectando tabelas como drivers, races, results, constructors, circuits, status e qualifying. Substitui valores inválidos (\N) por NaN, converte IDs para numéricos e tempos de volta (como fastestLapTime, q1, q2, q3) para milissegundos. Também ajusta a pontuação conforme as regras da FIA para diferentes períodos e adiciona dados como tempos de treino, classificação e regulamento técnico. Ao final, organiza e seleciona as colunas para gerar um dataset pronto para análise.
3. **Análise Exploratória (EDA):** O arquivo eda\_service.py usa bibliotecas como AutoViz, ydata\_profiling e D-Tale para uma análise exploratória dos dados. Ele ajuda a entender a distribuição das variáveis, identificar valores nulos, detectar outliers e achar correlações. Relatórios interativos e visuais são gerados pela função gerar\_relatorio(), enquanto exibir\_informacoes\_base() apresenta estatísticas descritivas diretamente no console.
4. **Pré-processamento:** O script data\_preprocessor.py lida com valores ausentes, aplica o StandardScaler nas variáveis numéricas e o OneHotEncoder nas variáveis categóricas. O ColumnTransformer é utilizado para aplicar transformações no conjunto de dados, seguindo as boas práticas do Scikit-learn, como descrito por Pedregosa et al. (2011).
5. **Treinamento de Modelos Manuais:** Em model\_service.py, a função train\_model() treina cinco modelos distintos com validação cruzada e busca de hiperparâmetros via GridSearchCV. Os resultados são armazenados e comparados com base em métricas como R², MAE e RMSE.
6. **Treinamento com PyCaret:** O pipeline também inclui o PyCaret, através da função train\_with\_pycaret(). Esta função automatiza o processo de configuração, comparação e seleção do melhor modelo de regressão, salvando o modelo final para uso posterior. O PyCaret realiza a comparação automática entre os modelos, retornando o de melhor desempenho baseado em múltiplas métricas de avaliação.
7. **Geração de Relatórios e Visualizações:** A função gerar\_graficos\_relatorios() cria gráficos que mostram a evolução de pontos por equipe, os Top 20 pilotos com mais pontos, a relação entre a idade do piloto e a posição final, os status de abandono mais comuns e os melhores pilotos e equipes de todos os tempos. Esses gráficos são gerados com Matplotlib e Seaborn e salvos em um diretório de relatórios, facilitando a análise visual e insights, como recomendado por Brownlee (2020).
8. **Simulação do Campeonato de F1 2025:** A função simular\_campeonato\_f1\_2025() usa o modelo Random Forest Regressor para prever os resultados do campeonato de 2025, com base em dados históricos. A simulação considera variáveis como posição de largada, velocidade da volta mais rápida, tempo de classificação, idade do piloto e confiabilidade histórica de pilotos e equipes (baseada em abandonos e finalizações). Os pontos são atribuídos aos 10 primeiros de cada corrida, conforme o regulamento da FIA, gerando rankings finais que destacam os melhores desempenhos da temporada.
9. **Execução Geral:** O Main.py conecta todos os módulos, permitindo a execução do pipeline completo com um único comando. Ele inclui dois fluxos paralelos de modelagem: um manual, com análise SHAP, e outro automatizado com PyCaret, oferecendo uma avaliação robusta e a seleção do melhor modelo.

**4. Coleta e Tratamento dos Dados**

**4.1 Obtenção do Dataset**

Utilizando a API oficial do Kaggle, o dataset foi baixado e armazenado localmente. As tabelas principais incluem: races.csv, results.csv, drivers.csv, constructors.csv, circuits.csv, qualifying.csv e status.csv. O processo de download e extração foi automatizado pela função baixar\_dataset().

**4.2 Pré-processamento**

As seguintes transformações foram aplicadas ao dataset:

* Substituição de valores nulos: Valores inválidos (\N) foram substituídos por NaN.
* Conversão de tempos: Colunas como fastestLapTime, q1, q2 e q3 foram convertidas para milissegundos para facilitar a análise.
* Criação de nova coluna: A Coluna regulamento foi criada com base numa tabela de regulamentos.
* Ajuste de pontuações: As pontuações foram corrigidas com base nas regras da FIA, considerando exceções específicas como Abu Dhabi 2014 e Bélgica 2021.
* Preenchimento de valores nulos: Colunas de tempo e posição foram preenchidas com valores padrão (ex.: 0.0 para tempos e -1 para posições ausentes).
* Seleção de colunas relevantes: Foram selecionadas colunas como position, points, age, name\_constructors, status, fastestLapTime, fastestLapSpeed, entre outras, para a modelagem.

Essas transformações garantem que os dados estejam limpos, consistentes e prontos para análise e modelagem.

**5. Análise Exploratória de Dados (EDA)**

A análise exploratória de dados (EDA) foi fundamental para compreender a estrutura do dataset e identificar padrões relevantes. Utilizando ferramentas como AutoViz, YData Profiling e D-Tale, foram gerados relatórios interativos e gráficos descritivos que facilitaram a interpretação dos dados.

As análises realizadas incluíram:

**Status da Corrida:** Foi classificado os tipos de abandono, como falha de motor, colisão e desistência. Esses status têm impacto direto na confiabilidade do carro e, consequentemente, na pontuação final.

**Gráfico, Gráfico de barras

O conteúdo gerado por IA pode estar incorreto.Gráfico 1**: Status de maior incidência de abandonos  
**Fonte**: dados da pesquisa

**Gráfico, Gráfico de dispersão

O conteúdo gerado por IA pode estar incorreto.Idade x Desempenho:** Um gráfico de dispersão revelou uma leve correlação entre a idade do piloto e sua colocação final, sugerindo que a experiência pode ser um fator diferencial em corridas.

**Gráfico 2**: Relação entre idade e posição final na corrida  
**Fonte**: dados da pesquisa

**Gráfico, Gráfico de linhas

O conteúdo gerado por IA pode estar incorreto.Top Equipes:** Ferrari, Mercedes e Red Bull foram identificadas como as equipes que mais bem pontuaram ao longo dos anos, porém Mclaren vem logo atrás, melhorando muitos nos últimos anos, validando a consistência dos dados históricos.

**Gráfico 3**: Evolução dos pontos por temporada 2010-2024  
**Fonte**: dados da pesquisa

Essas observações foram cruciais para validar a qualidade dos dados e orientar a seleção de variáveis relevantes para o modelo preditivo.

**6. Modelagem Preditiva**

Gráfico, Gráfico de barras

O conteúdo gerado por IA pode estar incorreto.Nesta etapa, foram aplicados cinco modelos supervisionados de regressão, cada um com suas particularidades:

**Gráfico 4**: Comparativo dos modelos de regressão   
**Fonte**: dados da pesquisa

* **Gradient Boosting Regressor**: Ideal para lidar com dados heterogêneos. Utiliza uma série de árvores de decisão fracas para formar um modelo forte, corrigindo os erros das etapas anteriores. Essa abordagem foi inicialmente proposta por Friedman (2001) e se tornou um padrão-ouro para competições de dados.
* **Random Forest Regressor**: Combina múltiplas árvores de decisão de forma paralela. Mostrou-se eficaz e rápido. Originalmente descrito por Breiman (2001), é amplamente usado em benchmarks de regressão.
* **LightGBM**: Usado por sua eficiência em grandes volumes de dados. Trabalha com histogramas para acelerar o processo. Desenvolvido por Ke et al. (2017), LightGBM é uma escolha popular em competições do Kaggle.
* **AdaBoost Regressor**: Foca nas instâncias mal classificadas em cada iteração. Boa performance, mas inferior ao Random Forest. Introduzido por Freund e Schapire (1997), é um dos algoritmos de boosting mais antigos.
* **SVR (Support Vector Regressor)**: Embora promissor teoricamente, teve desempenho fraco neste caso (R² de 0.02). SVR é eficiente em espaços de alta dimensionalidade, mas sensível à escala e parametrização.

Todos os modelos foram treinados com validação cruzada k-fold (k=5) e hiperparâmetros otimizados via GridSearchCV. As métricas principais para comparação foram:

* R² (coeficiente de determinação)
* MAE (erro absoluto médio)
* RMSE (raiz do erro quadrático médio)

As melhores performances foram obtidas com Gradient Boosting e Random Forest, ambos com R² superior a 0.82.

**7. Interpretação dos Resultados**

A análise dos **SHAP Values** permitiu identificar a influência de cada variável nas previsões do modelo. O gráfico SHAP revelou que fatores como **Status\_Finished, Position e Grid** foram os mais relevantes para determinar o desempenho dos pilotos.

Tabela

O conteúdo gerado por IA pode estar incorreto.

**Gráfico 5**: Análise SHAP dos dados   
**Fonte**: dados da pesquisa

Além disso, gráficos complementares destacaram insights importantes:

**Top 20 Pilotos com Mais Pontos**: Gráfico

O conteúdo gerado por IA pode estar incorreto.

**Gráfico 6**: Top 20 pilotos com mais pontos na história   
**Fonte**: dados da pesquisa

* + Pilotos como **Lewis Hamilton**, **Michael Schumacher** e **Sebastian Vettel** lideram o ranking histórico de pontos acumulados.

**Top 10 Equipes com Mais Pontos**: Gráfico

O conteúdo gerado por IA pode estar incorreto.

**Gráfico 7**: Equipes com mais pontos na história   
**Fonte**: dados da pesquisa

* + Equipes como **Ferrari**, **McLaren** e **Mercedes** dominam o cenário, refletindo sua consistência e desempenho ao longo dos anos.

**Top 10 Pilotos com Mais Pontos por Temporada**: Gráfico, Gráfico de barras

O conteúdo gerado por IA pode estar incorreto.

**Gráfico 8**: Top pilotos com mais pontos por temporada últimos 10 anos   
**Fonte**: dados da pesquisa

* Destaque para pilotos que tiveram temporadas excepcionais, consolidando sua posição entre os melhores.

Essas análises reforçam a importância de variáveis como **experiência do piloto**, **confiabilidade do carro** e **desempenho em classificações** para prever resultados futuros. Os gráficos fornecem uma visão clara e detalhada, ajudando a validar o modelo e orientar decisões baseadas em dados.

**8. Conclusão**

Este estudo demonstrou a eficiência de modelos de aprendizado de máquina aplicados à análise esportiva, especialmente no contexto da Fórmula 1. Ao longo da pesquisa, foi possível extrair insights relevantes a partir de dados históricos, entender os principais fatores que influenciam o desempenho dos pilotos e prever pontuações.

A análise revelou que variáveis como equipe, posição de largada, idade do piloto e status final da corrida têm impacto direto na pontuação. Gráficos como SHAP Values permitiram uma visão clara da contribuição de cada feature.

Recomenda-se, como trabalho futuro, a incorporação de variáveis contextuais como condições climáticas, estratégias de pit-stop e tipo de circuito (rua ou fechado). Também é possível explorar redes neurais recorrentes para prever desempenho ao longo de múltiplas corridas sequenciais. Além disso, explorar dados adicionais de sensores em tempo real, como telemetria dos carros, pode trazer avanços significativos nas previsões.

Por fim, este projeto contribui para o avanço do uso de inteligência artificial no esporte, fornecendo um modelo replicável que pode ser aplicado a outras competições ou modalidades esportivas.

**Referências**

BROWNLEE, J. *Machine Learning Mastery with Python*. [S.l.]: Machine Learning Mastery, 2020.

BUNKER, R.; THABTAH, F. A machine learning framework for sport result prediction. *Applied Computing and Informatics*, v. 15, n. 1, p. 27–33, 2019.

CHEN, T.; GUESTRIN, C. XGBoost: A scalable tree boosting system. In: *ACM SIGKDD INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING*, 22., 2016, San Francisco. *Proceedings* [...]. New York: ACM, 2016. p. 785–794.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, v. 55, n. 1, p. 119–139, 1997.

FRIEDMAN, J. H. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, v. 29, n. 5, p. 1189–1232, 2001.

GÉRON, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. 2. ed. Sebastopol: O’Reilly Media, 2019.

GRANDI, T. Formula 1 Winner Prediction. *GitHub*, 2020. Disponível em: <https://github.com/thomasgrandi/Formula1-Winner-Prediction>. Acesso em: 10 abr. 2025.

KE, G. et al. LightGBM: A highly efficient gradient boosting decision tree. In: *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS (NeurIPS)*, 2017. *Proceedings* [...]. [S.l.: s.n.], 2017. p. 3146–3154.

LUNDBERG, S. M.; LEE, S. I. A unified approach to interpreting model predictions. In: *NEURAL INFORMATION PROCESSING SYSTEMS (NeurIPS)*, 2017. *Proceedings* [...]. [S.l.: s.n.], 2017.

MDPI. A review of gradient boosting methods for predictive modeling. *Diagnostics*, 2021. Disponível em: <https://www.mdpi.com/2075-4418/11/9/1714>. Acesso em: 03 abr. 2025.

NIGRO, V. Formula 1 Analysis with Python. *Medium*, 2020. Disponível em: <https://medium.com/@vincent.nigro/formula-1-analysis-with-python-d8e7f0c040ef>. Acesso em: 08 abr. 2025.

PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011.

RAO, R. Formula 1 World Championship 1950–2020. *Kaggle*, 2020. Disponível em: <https://www.kaggle.com/datasets/rohanrao/formula-1-world-championship-1950-2020>. Acesso em: 05 abr. 2025.

SCIENCEDIRECT. A survey of gradient boosting models in predictive analytics. *Journal of Machine Learning Research*, 2021. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0020025521002875>. Acesso em: 15 abr. 2025.

SPRINGER. A comparative analysis of gradient boosting algorithms. *Artificial Intelligence Review*, 2020. Disponível em: <https://link.springer.com/article/10.1007/s10462-020-09896-5>. Acesso em: 17 abr. 2025.

TOWARDS DATA SCIENCE. Formula 1 Race Predictor. *Medium*, 2020. Disponível em: <https://medium.com/towards-data-science/formula-1-race-predictor-5d4bfae887da>. Acesso em: 06 abr. 2025.