Rodrigo Lorente Kauer

Previsão de resultados de partidas de futebol da série A do campeonato brasileiro utilizando aprendizado de máquina: uma análise comparativa de modelos

PROJETO FINAL DE GRADUAÇÃO

DEPARTAMENTO DE INFORMÁTICA
Centro Tecnológico Científico - CTC
Graduação em Engenharia de Computação

Rio de Janeiro
Dezembro de 2025

Pontifícia Universidade Católica
do Rio de Janeiro

Rodrigo Lorente Kauer

Previsão de resultados de partidas de futebol da série A do campeonato brasileiro utilizando aprendizado de máquina: uma análise comparativa de modelos

PROJETO FINAL DE GRADUAÇÃO

Relatório de Projeto Final, apresentado ao programa de Graduação do Departamento de Informática da PUC-Rio como requisito parcial para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador : Professor Alberto Barbosa Raposo
Co-orientador: Professor Cesar Augusto Sierra Franco

Rio de Janeiro
Dezembro de 2025

Resumo

Kauer, Rodrigo; Alberto, Raposo; . **Previsão de resultados de partidas de futebol da série A do campeonato brasileiro utilizando aprendizado de máquina: uma análise comparativa de modelos**. Rio de Janeiro, 2025. 65p. Relatório de Projeto Final (Graduação) – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

A previsão de resultados de partidas de futebol constitui um problema desafiador, em virtude da natureza dinâmica e multifatorial do esporte. No contexto brasileiro, a Série A do Campeonato Brasileiro apresenta características específicas, como elevado equilíbrio competitivo, influência regional e variabilidade de desempenho ao longo da temporada, que dificultam a modelagem preditiva. Este trabalho propõe o desenvolvimento e a avaliação de uma pipeline de aprendizado de máquina voltada à previsão dos resultados de partidas da Série A. Para a realização do estudo, foi construído um conjunto de dados abrangente a partir de bases históricas, incluindo estatísticas de jogo, informações de desempenho recente das equipes, confrontos diretos, fatores regionais e métricas relacionadas ao mercado de apostas esportivas. Um processo sistemático de engenharia de atributos foi aplicado com o objetivo de capturar padrões temporais e contextuais relevantes para o futebol brasileiro. Diversos modelos de classificação supervisionada foram avaliados, incluindo Regressão Logística, Naive Bayes, K-Nearest Neighbors, Support Vector Machines, Random Forest, Gradient Boosting, AdaBoost, Multilayer Perceptron e XGBoost. Os modelos foram comparados com base em métricas apropriadas para problemas potencialmente desbalanceados, como acurácia, precisão, recall e F1-score, com ênfase nas médias ponderadas e macro.

Palavras-chave

Aprendizado de Máquina; Classificação; Modelos de apredizado supervisionado.

Abstract

Kauer, Rodrigo; Alberto, Raposo (Advisor); (Co-Advisor). **Prediction of results of football matches in the Brazilian championship Série A using machine learning: a comparative analysis of models**. Rio de Janeiro, 2025. 65p. Final Project Report (Undergraduate) – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Predicting the outcomes of football matches is a challenging problem due to the sport's dynamic and multifactorial nature. In the Brazilian context, the Campeonato Brasileiro Série A exhibits specific characteristics, such as a high level of competitive balance, regional influences, and performance variability throughout the season that make predictive modeling particularly difficult. This work proposes the development and evaluation of a machine learning pipeline aimed at predicting match outcomes in Série A. To conduct the study, a comprehensive dataset was built from historical sources, including match statistics, recent team performance indicators, head-to-head records, regional factors, and metrics related to the sports betting market. A systematic feature engineering process was applied to capture temporal and contextual patterns relevant to Brazilian football. Several supervised classification models were evaluated, including Logistic Regression, Naive Bayes, K-Nearest Neighbors, Support Vector Machines, Random Forest, Gradient Boosting, AdaBoost, Multilayer Perceptron, and XGBoost. The models were compared using metrics suitable for potentially imbalanced problems, such as accuracy, precision, recall, and F1-score, with emphasis on weighted and macro averages.

Keywords

Machine Learning; Clasification; Supervised models.

Sumário

Lista de figuras

Lista de tabelas

Lista de Códigos

1
Introdução

O futebol ultrapassa a condição de simples prática esportiva, consolidando-se como um fenômeno de grande relevância cultural, social e econômica em escala global. No contexto brasileiro, a Série A do Campeonato Brasileiro representa o mais alto nível dessa manifestação, reunindo os principais clubes do país e movimentando setores estratégicos, como a mídia, a publicidade e, mais recentemente, o mercado de apostas esportivas. Diante da expressiva dimensão financeira e da elevada competitividade do campeonato, a análise de dados deixou de ser apenas um diferencial e passou a exercer papel estratégico, possibilitando a extração de informações relevantes sobre desempenho esportivo e comportamento de mercado.

Nesse contexto, as técnicas de aprendizado de máquina assumem papel de destaque ao possibilitar a transição de uma análise essencialmente descritiva para uma abordagem preditiva. Por meio do uso de dados históricos estruturados, como estatísticas de partidas, rankings de equipes e probabilidades, torna-se viável a antecipação de resultados com maior nível de assertividade. Entretanto, a aplicação dessas metodologias ao cenário do futebol brasileiro apresenta desafios específicos. A Série A caracteriza-se pelo elevado equilíbrio técnico entre as equipes, pela alta competitividade e pela influência significativa de fatores regionais e logísticos, aspectos que a diferenciam das ligas europeias, amplamente abordadas na literatura especializada.

Embora dados estruturados brutos, como número de gols, posse de bola e finalizações, forneçam uma base informacional relevante, eles, de forma isolada, não são suficientes para representar toda a complexidade inerente ao jogo. A natureza dinâmica do futebol exige a aplicação de um processo sistemático de engenharia de atributos, capaz de capturar padrões mais sutis, como o desempenho recente das equipes, o desgaste físico decorrente de longos deslocamentos e o impacto psicológico proveniente de confrontos históricos entre adversários.

Dessa forma, este trabalho propõe o desenvolvimento e a análise comparativa de modelos de classificação supervisionada voltados à previsão dos resultados de partidas da Série A do Campeonato Brasileiro, considerando as classes vitória do mandante, empate ou vitória do visitante. Para tanto, são utilizados dados históricos compreendendo o período de 2003 a 2024, com o auxílio da biblioteca scikit-learn, explorando algoritmos como Random Forest, Gradient Boosting e Regressão Logística. Além da avaliação das métricas de

desempenho dos modelos, o estudo investiga a importância relativa dos atributos empregados, considerando as particularidades do futebol nacional.

1.1 A Previsão Esportiva

A previsão de resultados de partidas de futebol é um problema desafiador, em razão de sua natureza multifacetada. De fato, o desfecho de uma partida é influenciado por uma combinação de fatores quantitativos, como número de gols e desempenho estatístico, bem como qualitativos, tais quais motivação dos jogadores, decisões táticas e condições climáticas. Embora dados estruturados sejam ricos, muitas vezes não capturam elementos dinâmicos, como mudanças de escalação ou eventos inesperados durante o jogo. Além disso, o problema apresenta desafios técnicos específicos, pois envolve:

- Desbalanceamento de classes: Empates são menos frequentes do que vitórias, de modo que podem enviesar os modelos.
- Não-estacionariedade dos dados: O desempenho dos times varia ao longo da temporada, logo exige estratégias que respeitem a ordem temporal dos dados.
- Alta dimensionalidade: A inclusão de múltiplos atributos, como estatísticas de jogo, rankings e odds, aumenta a complexidade computacional.
- O desempenho do time pode variar devido a lesões, mudanças de treinador ou até condições climáticas.

Consequentemente, esses desafios tornam a previsão de resultados de futebol um problema atraente para a aplicação de aprendizado de máquina, visto que exige não apenas a escolha adequada de algoritmos, mas também uma engenharia de atributos robusta para que se maximize a capacidade preditiva dos modelos.

1.2 Objetivo geral

O objetivo geral deste trabalho é desenvolver e avaliar uma pipeline de aprendizado de máquina para prever os resultados de partidas da Série A do Campeonato Brasileiro, a partir de dados históricos e atributos derivados, realizando uma análise comparativa de diferentes modelos de classificação supervisionada.

1.2.1 Objetivos específicos

Os objetivos específicos do estudo incluem:

- Construir um conjunto de dados integrado da Série A do Campeonato Brasileiro, no período de 2003 a 2024, combinando informações de resultados, estatísticas de jogo, gols e cartões.
- Projetar e implementar um conjunto de atributos derivados que representem aspectos relevantes do contexto brasileiro, como forma recente das equipes, confrontos diretos, fatores regionais e desgaste logístico.
- Treinar e otimizar diferentes modelos de classificação supervisionada, utilizando bibliotecas consolidadas de aprendizado de máquina em Python.
- Avaliar o desempenho dos modelos por meio de métricas adequadas a problemas desbalanceados, com destaque para o F1-score por classe e métricas globais.
- Comparar os resultados obtidos com evidências presentes na literatura, discutindo convergências, divergências e limitações do estudo.

1.3 Atividades realizadas

A fim de cumprir com os objetivos do projeto, a execução foi dividida em quarto partes, descritas na lista a seguir.

- Escolha dos modelos a serem testados
- Escolha dos datasets
- Criação dos arquivos de treino e teste
- Execução e análise dos arquivos

Nos próximos capítulos do relatório serão explicados quais modelos foram utilizados e no que eles se baseiam, detalhando os datasets utilizados, revelando assim, os resultados e conclusões do estudo. Todos os notebooks utilizados se encontram no seguinte link: https://github.com/Rodk32/Projeto-Final-II---Rodrigo-Kauer.git

2
Trabalhos relacionados

A previsão de resultados em partidas de futebol tem sido amplamente investigada nas áreas de estatística, ciência de dados e aprendizado de máquina, sobretudo em razão da natureza complexa e imprevisível do esporte, aliada ao crescente interesse econômico do setor esportivo. Estudos pioneiros utilizaram modelos estatísticos clássicos para modelar a ocorrência de gols e estimar probabilidades de resultados. Maher, do Departamento de Probabilidade e Estatística da Universidade de Sheffield na Inglaterra, propôs um dos primeiros modelos baseados em distribuições de Poisson para representar a força ofensiva e defensiva das equipes (1). Posteriormente, Mark Dixon e Stuart Coles, também britânicos aprimoraram essa abordagem ao introduzir ajustes para dependências temporais entre os gols, além de investigarem ineficiências no mercado de apostas (2).

Esses trabalhos inaugurais estabeleceram as bases teóricas para a modelagem quantitativa do futebol, demonstrando que variáveis como mando de campo, desempenho recente e eficiência ofensiva possuem influência estatisticamente significativa sobre os resultados. O holandês, Ruud Koning, por sua vez, analisou o equilíbrio competitivo em esportes coletivos, contribuindo para a compreensão do impacto da competitividade nas previsões esportivas. (3)

Com o avanço do aprendizado de máquina, métodos mais sofisticados passaram a ser empregados. Josip Hucaljuk e Alen Rakipović aplicaram técnicas de classificação supervisionada para prever resultados de partidas de futebol europeu, destacando o potencial dos métodos computacionais para problemas esportivos complexos.(4)

O uso de modelos baseados em árvores de decisão e métodos de ensemble também se consolidou na literatura. O matemático e estatístico da universidade de Berkeley na Califórnia, Leo Breiman introduziu o algoritmo Random Forest, que combina múltiplas árvores de decisão para reduzir variância e melhorar a capacidade de generalização.(5). O americano Jerome Friedman, por sua vez, propôs o Gradient Boosting, técnica que constrói modelos de forma incremental, corrigindo os erros dos classificadores anteriores. Ambos os métodos passaram a ser amplamente utilizados em tarefas de previsão esportiva, apresentando desempenho superior em diferentes cenários experimentais.(6)

A engenharia de atributos constitui um dos elementos centrais nos trabalhos recentes voltados à previsão de resultados no futebol. Rahul Baboota desenvolveu um modelo preditivo para a Premier League, a liga inglesa utili-

zando aprendizado de máquina e engenharia de atributos, combinando estatísticas históricas de desempenho com informações de mercado de apostas. O autor destacou que atributos derivados, como métricas agregadas de desempenho e indicadores de forma recente, conduzem a melhorias significativas nas métricas de acurácia e F1-score. (7)

Outro aspecto amplamente explorado na literatura é o uso de probabilidades do mercado de apostas como atributos auxiliares. Também analisou-se a eficiência de mercados de apostas esportivas, por Joseph Golec e Maurry Tamarkin, da Clarck University, demonstrando que as odds refletem, ainda que de forma imperfeita, expectativas agregadas dos agentes econômicos.(8)

Grande parte desses estudos concentra-se em ligas europeias, como Premier League, La Liga, Serie A italiana e Bundesliga, a liga alemã. Nessas competições, há ampla disponibilidade de dados estruturados e alta padronização estatística, o que favorece a aplicação de métodos de aprendizado de máquina. Entretanto, observa-se uma lacuna de pesquisas voltadas especificamente ao contexto do futebol brasileiro.

No que se refere ao contexto brasileiro, ainda é relativamente reduzido o número de estudos acadêmicos voltados à previsão de resultados do Campeonato Brasileiro, em comparação com o volume de pesquisas sobre ligas europeias. Trabalhos recentes em conferências e periódicos nacionais têm explorado modelos de classificação aplicados à Série A, frequentemente utilizando dados estatísticos de partidas e, em alguns casos, informações de odds de casas de apostas.

Entretanto, muitos desses estudos se limitam a períodos temporais curtos ou a conjuntos de atributos menos abrangentes, o que dificulta a generalização dos resultados para longas séries históricas. Revisões recentes sobre aprendizado de máquina aplicado ao futebol apontam que a incorporação explícita de fatores contextuais e logísticos ainda é relativamente rara na literatura, apesar de seu potencial impacto sobre o desempenho preditivo dos modelos.

Diante desse cenário, o presente trabalho diferencia-se ao aplicar uma pipeline completa de aprendizado de máquina à Série A do Campeonato Brasileiro, abrangendo um período histórico extenso de 2003 a 2024. Além disso, enfatiza-se a construção de atributos adaptados à realidade do futebol nacional, incorporando forma recente das equipes, histórico de confrontos diretos e fatores logísticos relacionados a deslocamentos. Diferentemente de parte dos estudos existentes, este trabalho não se limita à análise de métricas de desempenho, mas também investiga a importância relativa dos atributos, buscando compreender quais fatores mais influenciam os resultados das partidas no contexto brasileiro.

3
Base Teórica e Metodologia

Este capitulo servirá para mostrar um pouco da história do campeonato brasileiro e como ele funciona e para mostrar toda a base teórica e de exploração estatística e de ciência de dados até a execução dos pipelines de predição dos dados.

3.1
Estrutura competitiva do Campeonato Brasileiro Série A

O Campeonato Brasileiro Série A, desde 2003, adota o formato de pontos corridos. Com exceção do campeonato brasiliero de 2003, onde havia 24 equipes, todos os outros anos, vinte equipes disputam 38 rodadas distribuídas em turno e returno. Cada equipe enfrenta todas as demais como mandante e como visitante, o que resulta em uma estrutura equilibrada, sem fases eliminatórias ou critérios de desempate baseados em grupos. A pontuação segue o padrão internacional: três pontos para vitória, um ponto para empate e zero pontos para derrota.

Essa dinâmica faz com que o desempenho acumulado ao longo da temporada seja determinante para posições na tabela, o que permite a construção de séries temporais de desempenho, métricas de forma recente e indicadores de consistência. O Brasil, entretanto, possui características geográficas e logísticas que diferenciam seu campeonato de outras ligas frequentemente estudadas, como a Premier League ou La Liga, ligas da Inglaterra e Espanha respectivamente.

As distâncias entre sedes dos clubes são amplas, variando de deslocamentos curtos a viagens superiores a três mil quilômetros. Essa realidade introduz elementos adicionais de variabilidade no desempenho das equipes, como desgaste físico, adaptação climática e impacto de viagens longas. Além disso, clubes da Série A apresentam alta rotatividade de elenco, mudanças frequentes de comissões técnicas e forte influência do mando de campo, tanto pelo ambiente dos estádios quanto por características regionais.

Esses fatores tornam o Brasileirão um ambiente altamente competitivo e estatisticamente ruidoso, o que aumenta a complexidade da previsão de resultados. Assim, a metodologia adotada neste trabalho busca capturar essa complexidade por meio de engenharia de atributos, análise estatística e uso de modelos supervisionados capazes de lidar com variabilidade elevada.

3.2
Fundamentos estatísticos e exploração de dados

A análise exploratória de dados constitui uma etapa estruturante em estudos quantitativos. Seu objetivo é compreender a distribuição, variabilidade e relacionamentos entre as variáveis envolvidas, permitindo detectar padrões, inconsistências, outliers, correlações espúrias e eventuais necessidades de transformação de dados.

A exploração de dados fornece a base necessária para garantir que a modelagem posterior seja consistente e alicerçada em um entendimento profundo da estrutura informacional. (9)

3.2.1
Classificação das variaveis

O conjunto de variáveis utilizadas na modelagem preditiva pode ser agrupado conforme sua natureza:

Tipo de Variável	Exemplos no contexto do futebol	Implicações analíticas
Quantitativas contínuas	posse de bola, odds, distâncias percorridas	permitem cálculo de média, desvio padrão e correlações
Quantitativas discretas	gols, finalizações, escanteios	distribuições frequentemente assimétricas; adequadas para modelos de contagem
Qualitativas nominais	equipe mandante, estádio, adversário	exigem codificação para uso em modelos supervisionados
Qualitativas ordinais	posição na tabela, classificação recente	permitem comparações relativas e uso de rankings

Tabela 3.1: Classificação e Implicações Analíticas de Variáveis no Contexto do Futebol.

A correta classificação dessas variáveis é fundamental para decisões posteriores de engenharia de atributos e seleção de algoritmos.

3.2.2
Estatística descritiva

A estatística descritiva fornece medidas fundamentais para caracterização dos dados:

Medida	Interpretação	Aplicação no futebol
Média	Tendência central	Média de gols marcados por partida
Mediana	Robustez contra *outliers*	Posição central do desempenho recente
Variância / Desvio padrão	Grau de dispersão	Variabilidade de finalizações ou posse
Assimetria	Comportamento de cauda	Assimetria típica de distribuição de gols
Curtose	Concentração de valores	Intensidade de partidas com placares comuns

Tabela 3.2: Medidas Estatísticas e sua Interpretação no Futebol.

3.2.3
Análise multivariada

A etapa multivariada busca identificar relações entre variáveis, reduzindo a complexidade e orientando a engenharia de atributos.

1. Calcular correlação entre variáveis contínuas.
2. Identificar redundâncias, como altos coeficientes entre atributos semelhantes.
3. Analisar relações não lineares por meio de gráficos de dispersão.
4. Avaliar colinearidade para evitar atributos duplicados na modelagem.
5. Selecionar variáveis com relevância teórica e estatística.

3.3
Limpeza e preparação de dados

O processo de limpeza é essencial para garantir a integridade e a confiabilidade do dataset, especialmente ao lidar com séries históricas longas e provenientes de fontes diversas.

3.3.1
Critérios de qualidade dos dados

Os principais critérios aplicados incluem:

Tabela 3.3: Critérios de Qualidade de Dados e Problemas Típicos.

Critério	Descrição	Exemplos de problemas encontrados
Validade	Valores devem ser possíveis	Gols negativos, rodadas inexistentes
Consistência	Ausência de conflitos entre registros	Duplicação de partidas, placares divergentes
Completude	Ausência de lacunas críticas	Odds faltantes, escalações incompletas
Uniformidade	Padronização de formatos	Codificação textual inconsistente

3.3.2
Detecção e tratamento de outliers

Outliers foram identificados por inspeção gráfica e análise estatística. Exemplos típicos:

- Finalizações acima de 40 (incomuns e sugerem erro de coleta).
- Odds igual a 1.00 para ambos os times (erro na origem dos dados).
- Datas sem relação com o calendário oficial.

Esses registros foram tratados conforme sua natureza: correção quando possível, exclusão quando prejudiciais à modelagem.

3.3.3
Esquema de processo de preparação dos dados

$$\text{Coleta} \dashrightarrow \text{Consolidação} \dashrightarrow \text{Padronização} \dashrightarrow \text{Limpeza} \dashrightarrow \text{Tratamento de outliers} \dashrightarrow \text{Verificação final}$$

3.4
Engenharia de atributos

A engenharia de atributos visa transformar dados brutos em variáveis capazes de captar características relevantes do fenômeno estudado. Esse processo é apontado como um dos fatores que mais influenciam o desempenho de modelos preditivos modernos. A descrição completa dos atributos criados está no próximo capítulo após a apresentação dos datasets.

3.4.1
Esquema explicativo do processo de engenharia de atributos

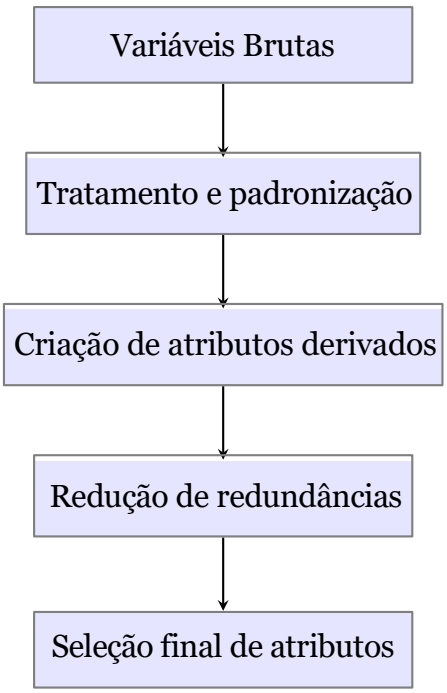


Figura 3.1: Esquema do Processo de Engenharia e Seleção de Atributos.

3.5
Fundamentos de Aprendizado de Máquina e modelos utilizados

A seleção de modelos utilizada nesta pesquisa buscou contemplar diferentes abordagens teóricas do aprendizado supervisionado, permitindo avaliar como distintos mecanismos de aprendizado interpretam os atributos derivados das partidas do Campeonato Brasileiro.

Como cada algoritmo opera de maneira própria e possui hipóteses distintas sobre os dados, a comparação entre eles oferece uma visão abrangente do problema de previsão esportiva. Nos subtópicos a seguir, apresenta-se uma explicação conceitual e introdutória sobre cada tipo de modelo utilizado, destacando suas características fundamentais e a razão de sua inclusão neste estudo, apenas para o leitor ter uma ideia de como é a matemática por tras de cada modelo.

3.5.1
Modelos lineares

3.5.1.1
Regressão Logística

A Regressão Logística é um modelo estatístico clássico utilizado para estimar a probabilidade de ocorrência de classes em problemas de classificação. Em sua essência, trata-se de um modelo linear que transforma combinações lineares dos atributos em probabilidades por meio da função logística. A ideia central é encontrar coeficientes que indiquem o quanto cada atributo contribui para aumentar ou reduzir a chance de determinado resultado.

Por ser matematicamente simples e altamente interpretável, a Regressão Logística permite compreender diretamente a influência de variáveis como posse de bola, forma recente e probabilidades implícitas das odds. Apesar de sua estrutura linear, frequentemente alcança desempenho comparável a métodos mais complexos em bases tabulares, sendo uma escolha natural como modelo de referência em estudos preditivos.

A Regressão Logística parte do pressuposto de que a relação entre os atributos de entrada e a probabilidade do evento pode ser expressa por uma transformação logística. Embora seja um modelo linear nos parâmetros, sua saída é uma probabilidade, garantida pela função sigmoide.

Seja $X = (x_1, x_2, \ldots, x_k)$ o vetor de atributos e Y a variável resposta.

A probabilidade de $Y = 1$ é dada por:

$$P(Y = 1 \mid X) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k)}} \tag{3-1}$$

O termo linear $\beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k$ é chamado de *logit*, pois representa o logaritmo da razão de chances ($p/(1-p)$):

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 x_1 + \cdots + \beta_k x_k \tag{3-2}$$

O processo de ajuste consiste em encontrar os coeficientes β que maximizam a verossimilhança dos dados observados. (10)

3.5.2
Modelos baseados em árvores e métodos de ensenble

3.5.2.1 Random Forest

O Random Forest é um método de ensemble que combina diversas árvores de decisão independentes para produzir uma predição final. Cada árvore é construída a partir de amostras aleatórias dos dados, e o modelo final corresponde ao voto majoritário das árvores individuais. Árvores de decisão isoladas tendem a apresentar alta variância, mas ao combiná-las, o Random Forest reduz efeitos de ruído e melhora a generalização. Essa abordagem é particularmente útil quando há interações complexas entre atributos, já que as árvores conseguem modelar relações não lineares sem necessidade de transformações matemáticas adicionais.

Uma Árvore de Decisão particiona o espaço de atributos em regiões homogêneas. Em cada divisão, escolhe-se o atributo que melhor separa as classes com base em critérios como Gini ou entropia.

A impureza de Gini é definida por:

$$G = 1 - \sum_{c} p_c^2 \tag{3-3}$$

onde p_c é a proporção de elementos da classe c no nó.

O *Random Forest* expande essa lógica treinando diversas árvores independentes, cada uma com amostras e atributos selecionados aleatoriamente. A predição final é dada por votação entre as árvores:

$$\hat{y} = \text{modo}\{T_1(X), T_2(X), \ldots, T_m(X)\} \tag{3-4}$$

onde $T_i(X)$ é a predição da i-ésima árvore e m é o número total de árvores.

Essa combinação reduz variância e torna o modelo mais estável que uma árvore individual.(5)

3.5.2.2 Gradient Boosting

O Gradient Boosting segue uma lógica distinta do Random Forest. Em vez de treinar árvores independentes, ele constrói uma sequência de árvores onde cada uma tenta corrigir os erros cometidos pelas anteriores. Assim, o modelo evolui de forma incremental, ficando progressivamente mais especializado naquilo que ainda não foi aprendido. O resultado é um classificador capaz de capturar padrões sutis e diversas interações entre variáveis. Embora mais sensível a hiperparâmetros e ao risco de sobreajuste, apresenta excelente desempenho quando adequadamente configurado.

O *Gradient Boosting* constrói modelos de forma sequencial. Cada nova árvore tenta corrigir os erros cometidos pela combinação anterior.

De forma simplificada, o modelo final ($F_M(X)$) é:

$$F_M(X) = F_0(X) + \sum_{m=1}^{M} \gamma_m h_m(X) \tag{3-5}$$

onde:

- $F_0(X)$ é uma predição inicial (como a média),
- $h_m(X)$ é a árvore treinada na etapa m,
- γ_m é o peso da árvore.

A cada iteração, o algoritmo ajusta-se ao gradiente do erro, por isso o nome *gradient boosting*.(6)

3.5.2.3 XGBoost

O XGBoost é uma versão otimizada do Gradient Boosting que utiliza melhorias computacionais significativas, como paralelização, regularização explícita e manipulação eficiente de valores ausentes. Sua popularidade deriva do fato de frequentemente apresentar desempenho superior em bases tabulares e ser amplamente utilizado em competições de ciência de dados. É considerado um dos métodos mais poderosos para problemas estruturados, especialmente quando os dados apresentam grande número de atributos derivados

O *XGBoost* (eXtreme Gradient Boosting) é uma implementação otimizada do *Gradient Boosting*, em que a função de custo (L) inclui um termo de regularização.

A função de custo global é dada por:

$$\mathrm{L} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{3-6}$$

Onde:

- $\sum_{i=1}^{n} l(y_i, \hat{y}_i)$ é a função de perda (erro) no conjunto de treinamento.
- $\sum_{k=1}^{K} \Omega(f_k)$ é o termo de regularização, que penaliza a complexidade dos modelos (árvores).

O termo de regularização $\Omega(f)$ para uma única árvore f é definido como:

$$\Omega(f) = \alpha T + \frac{1}{2}\lambda \|w\|^2 \tag{3-7}$$

Onde:

- T é o número de folhas (nós terminais) na árvore,
- $\|w\|^2$ é a norma L2 dos pesos das folhas,
- α e λ são hiperparâmetros de regularização.

A regularização controla a complexidade das árvores e evita sobreajuste (*overfitting*), tornando o modelo mais robusto.(11)

3.5.2.4 AdaBoost

O AdaBoost, um dos primeiros métodos de boosting, funciona ajustando pesos maiores a exemplos classificados incorretamente nas iterações anteriores. Dessa forma, o modelo concentra seu aprendizado nos casos mais difíceis. Embora simples quando comparado a abordagens modernas, o AdaBoost oferece uma perspectiva histórica sobre o boosting e opera de forma eficiente em cenários com separações razoavelmente definidas entre as classes.

O *AdaBoost* (Adaptive Boosting) ajusta sucessivos classificadores fracos, reforçando a atenção nos exemplos mal classificados. A cada iteração, exemplos com erro recebem maior peso.

O peso (α_m) de cada classificador h_m é calculado por:

$$\alpha_m = \frac{1}{2} \ln \left[\frac{1 - \epsilon_m}{\epsilon_m} \right] \tag{3-8}$$

onde ϵ_m é a taxa de erro daquele classificador.

A predição final $H(X)$ é dada pela combinação ponderada dos classificadores:

$$H(X) = \text{sign} \left[\sum_{m=1}^{M} \alpha_m h_m(X) \right] \tag{3-9}$$

onde M é o número total de classificadores e sign é a função sinal.(12)

3.5.3 Modelos Probabilísticos

3.5.3.1 Naive Bayes

O Naive Bayes é um classificador baseado no Teorema de Bayes, que calcula a probabilidade de uma classe ocorrer dado o conjunto de atributos observados. Sua característica mais marcante é a suposição de independência condicional entre os atributos, o que simplifica drasticamente os cálculos. Mesmo que tal suposição não seja totalmente verdadeira na prática, o modelo costuma obter resultados surpreendentemente eficazes, especialmente quando

os padrões dos dados são bem definidos. Por sua velocidade, simplicidade e capacidade de generalização, tornou-se uma escolha adequada para compor a base comparativa desta pesquisa.

O *Naive Bayes* utiliza o Teorema de Bayes e assume independência condicional entre atributos.

A probabilidade posterior $P(Y = c \mid X)$ é dada por:

$$P(Y = c \mid X) = \frac{P(Y = c) \prod_{i=1}^{k} P(x_i \mid Y = c)}{P(X)} \tag{3-10}$$

Devido ao fato de o denominador $P(X)$ ser constante para todas as classes, o modelo escolhe a classe com maior probabilidade posterior (a regra de decisão de Bayes simplificada):

$$\hat{y} = \arg\max_{c} P(Y = c) \prod_{i=1} P(x_i \mid Y = c) \tag{3-11}$$

Mesmo com a suposição simplificadora, o método costuma apresentar desempenho surpreendente em dados estruturados.(13)

3.5.4
Modelos Geométricos e Baseados em Instância

3.5.4.1
Suport Vector Machine (SVM)

O Support Vector Machine é um modelo que busca identificar o hiperplano que melhor separa as classes, maximizando a margem entre os grupos. Quando a separação não é linear, o SVM utiliza funções de kernel para projetar os dados em um espaço de maior dimensionalidade, permitindo capturar relações mais complexas. O kernel RBF, utilizado neste estudo, é uma das escolhas mais comuns para esse tipo de tarefa, pois transforma os dados de forma flexível, possibilitando que o modelo identifique padrões não triviais no desempenho esportivo.

O *Support Vector Machine* (*SVM*) procura o hiperplano que maximize a margem entre duas classes.

A margem (margem) do classificador é definida por:

$$\text{margem} = \frac{2}{\|\mathbf{w}\|} \tag{3-12}$$

O classificador encontra o vetor de pesos **w** e o *bias b* que satisfaçam a restrição de separação para os vetores de suporte (os pontos mais próximos do

hiperplano):

$$y_i(\mathbf{w}^T x_i + b) \geq 1 \tag{3-13}$$

Para dados não separáveis linearmente, utiliza-se um *kernel* (função de mapeamento para espaços de maior dimensão). No caso deste trabalho, o *kernel* RBF (*Radial Basis Function*) é dado por:

$$K(\mathbf{x}, \mathbf{x}') = e^{-\gamma\|\mathbf{x}-\mathbf{x}'\|^2} \tag{3-14}$$

onde γ é um hiperparâmetro que controla a influência do mapeamento.(14)

3.5.4.2
K-Nearest Neighbors (KNN)

O K-Nearest Neighbors é um modelo baseado na ideia de que exemplos semelhantes têm maior probabilidade de pertencer à mesma classe. Em vez de treinar um classificador explícito, o KNN armazena os dados e determina a classe de uma nova observação com base nas partidas historicamente mais parecidas. Sua lógica é simples e intuitiva: se jogos anteriores com características semelhantes resultaram em vitória do mandante, por exemplo, então o modelo tende a prever o mesmo desfecho para uma partida semelhante. Contudo, sua eficiência diminui à medida que o conjunto de dados cresce, e seu desempenho depende fortemente da escolha adequada do número de vizinhos.

O *k-Nearest Neighbors* (*k-NN*) não possui um modelo interno (é um classificador baseado em instâncias). A ideia é que a classe de um novo ponto **x** é determinada pelas classes dos *k* vizinhos mais próximos, de acordo com uma métrica de distância, sendo a mais comum a distância Euclidiana:

$$d(\mathbf{x}, \mathbf{x}') = \sqrt{\sum_{i=1}^{n} (x_i - x_i')^2} \tag{3-15}$$

onde *n* é o número de atributos.

A classe predita ($\hat{y}$) para o novo ponto é determinada pela votação da maioria simples:

$$\hat{y} = \text{modo}\{y_{(1)}, y_{(2)}, \ldots, y_{(k)}\} \tag{3-16}$$

onde $y_{(j)}$ é a classe do *j*-ésimo vizinho mais próximo(15).

3.5.5
Redes Neurais Artificiais

3.5.5.1
Multi-Layer Perceptron (MLP)

O Multi-Layer Perceptron é uma rede neural composta por camadas de neurônios artificiais capazes de aprender relações não lineares por meio de transformações sucessivas dos dados. Cada camada aplica uma função de ativação que permite ao modelo aprender padrões complexos que modelos tradicionais podem não capturar. Apesar dessa expressividade, redes neurais exigem volumes maiores de dados, cuidado no ajuste de hiperparâmetros e maior capacidade computacional. No presente estudo, o MLP foi incluído com o propósito de avaliar se abordagens baseadas em aprendizado profundo ofereceriam alguma vantagem significativa na identificação de padrões complexos do futebol. Contudo, seu desempenho foi limitado pela escala da base de dados e pela natureza predominantemente tabular do problema.

O *Multi-Layer Perceptron* (*MLP*) é uma rede neural composta por camadas de neurônios artificiais, incluindo uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída.

Um neurônio artificial recebe múltiplas entradas ($\mathbf{x}$), aplica pesos ($\mathbf{w}$), soma um *bias* (b) e, em seguida, utiliza uma função de ativação (f) para produzir uma saída (a):

$$a = f(\mathbf{w}^{\top}\mathbf{x} + b) \tag{3-17}$$

As funções de ativação (f) mais comuns são a ReLU (*Rectified Linear Unit*) e a sigmoide.

O treinamento ajusta os pesos por *backpropagation* (retropropagação), minimizando uma função de perda (como a entropia cruzada). A rede aprende representações progressivamente mais abstratas dos dados, permitindo capturar relações complexas e não lineares(16).

3.6
Métricas de Validação

A avaliação dos modelos de classificação utilizados neste estudo baseou-se em métricas estatísticas amplamente reconhecidas na literatura de aprendizado de máquina, complementadas por uma abordagem metodológica de validação temporal. Essa escolha visa garantir não apenas precisão numérica, mas também realismo no processo de previsão, evitando vieses decorrentes de uso inadequado dos dados.

3.6.1
Métricas de Desempenho de Classificação

Para quantificar o desempenho dos algoritmos, foram empregadas métricas derivadas diretamente da matriz de confusão, ferramenta que organiza as predições em verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos. Essa estrutura conceitual está detalhada nos materiais de referência, que destacam sua importância para interpretar a natureza dos erros cometidos pelos modelos.

A seguir serão descritas as métricas utilizadas.

3.6.1.1
Acurácia

A acurácia representa a proporção total de predições corretas do modelo em relação ao total de observações:

$$\text{Acurácia} = \frac{TP + TN}{P + N} \tag{3-18}$$

onde TP é o número de verdadeiros positivos, TN é o número de verdadeiros negativos, P é o número total de positivos e N é o número total de negativos.

Essa métrica foi escolhida porque, após a remoção da classe "empate", o problema tornou-se binário, com distribuição relativamente próxima ao balanceamento entre vitórias do mandante e do visitante. Assim, a acurácia se torna uma medida estável e interpretável. Nos materiais teóricos, ela é apresentada como a métrica central em problemas básicos de classificação binária.

3.6.1.2
Precisão

A precisão quantifica a proporção de predições positivas que de fato pertencem à classe positiva:

$$\text{Precisão} = \frac{TP}{TP + FP} \tag{3-19}$$

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

No contexto do futebol, essa métrica responde à pergunta: "Quando o modelo prevê vitória do mandante, com que frequência essa previsão está

correta?" Sua utilidade prática é evidente, especialmente quando decisões de investimento (como apostas) podem ser sensíveis a falsos positivos.

3.6.1.3 Recall

O recall, também conhecida como Sensibilidade, mede a capacidade do modelo de identificar corretamente os casos positivos reais:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3-20}$$

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

No problema tratado, isso equivale a medir quantas vitórias reais do mandante foram efetivamente detectadas pelo modelo. Essa métrica é fundamental para evitar que o classificador seja excessivamente conservador, ignorando oportunidades evidentes.

3.6.1.4 F1-Score

O *F1-Score* combina a precisão e a revocação em uma média harmônica, oferecendo uma visão balanceada do desempenho do modelo:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precisão} \cdot \text{Recall}}{\text{Precisão} + \text{Recall}} \tag{3-21}$$

Essa métrica é particularmente útil quando se busca um equilíbrio entre essas duas dimensões, evitando interpretações distorcidas que poderiam surgir quando se analisa apenas uma medida isolada. Modelos que "apostam" sempre no mandante podem inflar precisão ou revocação, mas não o *F1-Score*.

3.6.1.5 Matriz de confusão

A matriz de confusão sintetiza o desempenho do modelo ao discriminar corretamente cada classe. Sua estrutura auxilia na compreensão das assimetrias inerentes ao problema, especialmente o conhecido efeito do mando de campo, amplamente documentado em estudos esportivos.

Em nossos experimentos, a matriz revelou uma tendência clara: o modelo apresenta maior acurácia ao prever vitórias do mandante do que vitórias do visitante. Esse comportamento se alinha ao fato estatístico de que o mando de campo aumenta a probabilidade de vitória em competições como a Série A, criando um padrão que os classificadores tendem a captar.

A matriz de confusão assume a seguinte forma (onde a classe positiva de interesse é a "Vitória do Mandante"):

Resultado Real \ Predito	**Vitória do Mandante**	**Vitória do Visitante**
Vitória do Mandante	Verdadeiro Positivo (TP)	Falso Negativo (FN)
Vitória do Visitante	Falso Positivo (FP)	Verdadeiro Negativo (TN)

Tabela 3.4: Estrutura da Matriz de Confusão para Classificação Binária.

A seguir, cada elemento é explicado em detalhe.

3.6.1.6 Verdadeiro Positivo (TP)

Um verdadeiro positivo ocorre quando o modelo prevê vitória do mandante e essa vitória realmente ocorre. No contexto do futebol, isso representa um acerto consistente: o modelo identificou corretamente o favorito dentro de casa. Um exemplo de um TP seria: O modelo prevê que o Internacional vencerá em casa, e o Internacional realmente vence.

3.6.1.7 Verdadeiro Negativo (TN)

Um verdadeiro negativo ocorre quando o modelo prevê vitória do visitante e o visitante realmente vence. Essa categoria indica que o modelo conseguiu reconhecer corretamente condições desfavoráveis ao mandante, o que geralmente é mais difícil em competições como o Brasileirão, onde o mando de campo exerce forte influência. Um exemplo de um TN seria: O modelo prevê vitória do Palmeiras em um jogo fora de casa, e o Palmeiras realmente vence.

3.6.1.8 Falso Positivo (FP)

O falso positivo acontece quando o modelo prevê vitória do mandante, mas o visitante vence. É um tipo de erro relevante, pois representa casos em que o modelo depositou “confiança excessiva” no fator casa ou interpretou erroneamente a força do mandante. Um exemplo de um FP seria: O modelo prevê vitória do Grêmio na Arena, mas o Grêmio perde para o Botafogo. Este erro indica previsões otimistas demais em favor do mandante.

3.6.1.9 Falso Negativo (FN)

O falso negativo ocorre quando o modelo prevê vitória do visitante, mas o mandante vence. Esse é um tipo de erro que sinaliza que o modelo não

captou algum fator que favorecia o time da casa — forma recente, desgaste do adversário, histórico no confronto, entre outros. Um exemplo de um FN seria: O modelo prevê vitória do visitante, mas o mandante vence o jogo. Esse tipo de erro significa que o modelo "subestimou" o mandante.

3.6.1.10
Imporatância Analítica da Matriz de Confusão

A matriz de confusão fornece uma visão mais rica do desempenho do modelo do que a acurácia isolada. Enquanto a acurácia resume a proporção total de acertos, a matriz permite observar:

- Se o modelo está enviesado em favor de uma classe (por exemplo, sempre prevê vitória do mandante).
- Se há tendência a cometer mais erros do tipo Falso Positivo (FP) ou Falso Negativo (FN).
- Se o modelo identifica corretamente situações desafiadoras, como vitórias do visitante.
- Se a classificação está equilibrada ou distorcida.

No presente estudo, a matriz revelou um padrão típico de competições de futebol: os modelos demonstram maior facilidade em prever vitórias do mandante, refletindo fatores como familiaridade com o estádio, apoio da torcida e menor desgaste logístico. Já as vitórias do visitante tendem a ser menos frequentes e mais difíceis de identificar, o que se reflete nos valores de Verdadeiro Negativo (TN) e Falso Positivo (FP).

4
Desenvolvimento

O desenvolvimento começou primeiramente na investigação e procura por bons *datasets* que contemplem o máximo de informações estatísticas a respeito do Campeonato Brasileiro. Além de suprir uma boa quantidade de informações, foi buscado um *dataset* com um histórico interessante. Após procurar em diversas plataformas, como por exemplo o *Kaggle* e o *GitHub*, foi encontrado um *dataset* interessante, que contempla dados desde 2003 até 2024.

O *dataset* do Transfermarkt, um site alemão que contempla diversas informações estatísticas sobre os campeonatos ao redor do mundo, se tornou uma boa opção por esses dois principais fatores: um bom histórico recheado de colunas que podem se tornar boas informações ou não. Para isso servirá toda a análise exploratória nesses dados, citada no capítulo 3.2. (17)

4.1
Dataset - Transfermarket

Os conjuntos de dados utilizados neste estudo oferecem uma visão ampla e detalhada do Campeonato Brasileiro, contemplando informações que representam diferentes dimensões do jogo. O *dataset* de gols registra individualmente cada ocorrência ofensiva, incluindo autor, minuto e características do lance, o que permite identificar padrões de produtividade e o impacto temporal dos gols nas partidas. O *dataset* completo das partidas funciona como a base central da análise, reunindo variáveis estruturais como formacões das equipes, placares, mandos de campo, estádios e escalações, possibilitando a reconstrução contextual de cada jogo. O conjunto de dados referente aos cartões acrescenta informações disciplinares que refletem intensidade, postura tática e nível de confronto físico entre as equipes.

Por fim, o *dataset* de estatísticas apresenta métricas quantitativas de desempenho como finalizações, posse de bola, precisão de passes e escanteios, permitindo mensurar eficiência, domínio territorial e comportamentos estratégicos. Quando integrados, esses quatro conjuntos formam uma base robusta e representativa do futebol brasileiro, viabilizando análises preditivas que consideram aspectos ofensivos, defensivos, disciplinares, contextuais e táticos de maneira conjunta e rigorosa.

Coluna	Descrição
ID	Identificador único da partida.
Rodada	Número da rodada em que a partida ocorreu.
Data	Data de realização da partida (formato DD/MM/AAAA).
Horário	Horário de início da partida (formato HH:MM).
Dia	Dia da semana em que a partida foi realizada.
Mandante	Nome do clube atuando como mandante.
Visitante	Nome do clube atuando como visitante.
formacao_mandante	Formação tática utilizada pelo mandante.
formacao_visitante	Formação tática utilizada pelo visitante.
tecnico_mandante	Nome do técnico do clube mandante.
tecnico_visitante	Nome do técnico do clube visitante.
Vencedor	Clube vencedor (-" indica empate).
Arena	Local (estádio) onde a partida foi realizada.
mandante_Placar	Quantidade de gols marcados pelo mandante.
visitante_Placar	Quantidade de gols marcados pelo visitante.

Tabela 4.1: Estrutura do Dataset Central de Partidas

Coluna	Descrição
partida_id	Identificador da partida.
rodada	Rodada da partida em que o cartão foi aplicado.
clube	Clube do jogador punido.
atleta	Nome do atleta que recebeu o cartão.
cartao	Tipo do cartão recebido (amarelo/vermelho).
num_camisa	Número da camisa do atleta.
posicao	Posição do jogador em campo.
minuto	Minuto em que o cartão foi aplicado.

Tabela 4.2: Estrutura do Dataset campeonato-brasileiro-cartoes.csv

Coluna	Descrição
partida_ID	Identificador único da partida.
Rodada	Número da rodada da partida.
Clube	Nome do clube que marcou o gol.
Atleta	Nome do jogador que marcou o gol.
Minuto	Minuto do jogo em que o gol foi marcado.

Tabela 4.3: Estrutura do Dataset campeonato-brasileiro-gols.csv

Coluna	Descrição
partida_ID	Identificador único da partida.
Rodada	Número da rodada da partida.
Clube	Nome do clube ao qual as estatísticas pertencem.
Chutes	Total de finalizações realizadas pelo clube.
Chutes_a_gol	Total de finalizações na direção do gol.
Posse_de_bola	Percentual de posse de bola durante a partida.
Passes	Quantidade total de passes realizados.
precisao_passes	Percentual de precisão nos passes.
Faltas	Quantidade de faltas cometidas pelo clube.
cartao_amarelo	Quantidade de cartões amarelos recebidos.
cartao_vermelho	Quantidade de cartões vermelhos recebidos.
Impedimentos	Quantidade de impedimentos marcados.
Escanteios	Quantidade de escanteios conquistados.

Tabela 4.4: campeonato-brasileiro-estatisticas-full.csv

4.1.1 Exploração do dataset do Transfermarkt

A etapa de exploração dos dados revelou uma descontinuidade significativa na qualidade e completude das estatísticas ao longo da série histórica do Campeonato Brasileiro. A análise do arquivo campeonato-brasileiro-estatisticas-full.csv evidenciou que as temporadas compreendidas entre 2003 e 2013 não apresentam registros técnicos válidos, uma vez que variáveis fundamentais como chutes, escanteios, passes e posse de bola possuem médias iguais a zero ou valores nulos em todos os jogos. Esses resultados indicam que, nesse período, a fonte original continha apenas informações básicas como data e placar, sem qualquer detalhamento dos *scouts* de desempenho. O ano de 2014 também não pôde ser aproveitado integralmente, pois apresentou valores incompatíveis com partidas reais, como média de apenas 0,51 chutes por jogo por equipe, revelando que a coleta de dados foi parcial e insuficiente para compor análises estatisticamente confiáveis.

A partir de 2015 observa-se estabilidade nos indicadores técnicos, com valores médios condizentes com o comportamento real do futebol brasileiro, resultando em um conjunto de dados consistente entre 2015 e 2023. Dessa forma, definiu-se que apenas esse intervalo seria utilizado para caracterizar o perfil estatístico das equipes e construir os modelos preditivos, totalizando aproximadamente 3.400 partidas com registro completo das variáveis relevantes.

No período validado, algumas tendências se destacam e ajudam a delinear o estilo de jogo predominante no Brasil. Cada equipe produziu, em média, 11,53

finalizações por partida, com desvio padrão elevado, indicando grande variação na agressividade ofensiva entre equipes e entre confrontos. O número médio de finalizações certas foi de apenas 3,01, o que corresponde a aproximadamente 26% de precisão, revelando baixa eficiência no momento da finalização.

A posse de bola apresentou distribuição equilibrada, com média próxima de 50% e desvio de 10,6%, sugerindo que a competição não apresenta dominância estrutural de um estilo de jogo específico. O volume de passes por equipe, em torno de 377 por jogo, aliado à precisão média de 80,4%, indica um futebol relativamente cadenciado e com alta frequência de trocas de posse. Em termos de eventos geradores de interrupções, a média de 13,36 faltas por equipe demonstra um ritmo de jogo fragmentado, acumulando aproximadamente 26 infrações por partida. O comportamento disciplinar também se destaca, com equipes recebendo cerca de 2 cartões amarelos por jogo. Além disso, o número médio de escanteios por equipe, 4,68 por partida, contribui para caracterizar o volume de jogadas ofensivas e lances de bola parada.

A exploração dos resultados das partidas desse período validado confirma a existência de um forte viés favorável ao mandante no Campeonato Brasileiro. Entre 2015 e 2023, 48,07% dos jogos terminaram com vitória do time da casa, enquanto empates representaram 26,95% e vitórias dos visitantes corresponderam a apenas 24,98%. Essa distribuição revela um desbalanceamento natural das classes e estabelece um parâmetro mínimo de desempenho para qualquer modelo preditivo. Um classificador trivial que previsse vitória do mandante em todas as partidas teria acurácia de 48% o que implica que qualquer abordagem de aprendizado de máquina deve superar significativamente essa marca para ser considerada eficaz.

Esse resultado reforça tanto a importância de se utilizar métricas complementares à acurácia quanto a necessidade de métodos capazes de lidar com padrões assimétricos característicos do futebol brasileiro.

4.1.2 Síntese do Processamento de Dados

O processo de preparação dos dados iniciou-se com a seleção criteriosa das fontes utilizadas no estudo, concentrando-se exclusivamente em dois arquivos principais provenientes do repositório analisado. O primeiro, campeonato-brasileiro-full.csv, reúne os metadados essenciais de cada partida, incluindo data, horário, equipes envolvidas, estádio e placar final, constituindo-se como a base a partir da qual foi extraída a variável alvo do modelo. O segundo, campeonato-brasileiro-estatisticas-full.csv, contém os dados de scout referentes às equipes, reunindo informações técnicas como finalizações, posse de bola,

número de passes, escanteios e faltas. A combinação dessas duas tabelas foi considerada necessária e suficiente para o escopo do trabalho, uma vez que a primeira descreve o desfecho da partida, enquanto a segunda detalha o desempenho que levou a esse resultado. Outras tabelas disponíveis no repositório, como aquelas contendo escalações nominais de jogadores, foram deliberadamente excluídas a fim de evitar a elevação excessiva da dimensionalidade e manter o foco na performance coletiva, não na individualidade dos atletas.

A integração entre os datasets foi realizada por meio de uma operação de junção utilizando o identificador único da partida. Inicialmente, um Left Join foi empregado para diagnóstico e verificação de inconsistências, sendo substituído posteriormente por um Inner Join na fase final da modelagem, o que garantiu a utilização apenas de partidas que possuíam simultaneamente o resultado e o conjunto completo de estatísticas.

Código 1: Código Python para Merge de DataFrames

```
df_merged = df_partidas.merge(
    df_estatisticas,
    left_on='ID',
    right_on='partida_id',
    how='inner'
)
```

Após a seleção do intervalo temporal válido, realizou-se a limpeza e a tipagem das variáveis. Colunas como posse de bola e precisao passes continham percentuais armazenados como texto em virtude da presença do caractere %, exigindo sua remoção e conversão para valores numéricos. A demonstração dessa etapa é dada abaixo.

Código 2: Tratamento de Percentuais

```
df['posse_de_bola'] = df['posse_de_bola'].str.replace('%', '').astype(float)
df['precisao_passes'] = df['precisao_passes'].str.replace('%', '').astype(
    float)
```

Valores textuais como "None"foram substituídos por NaN, viabilizando seu tratamento posterior, e a coluna de datas foi convertida para o formato datetime, garantindo a correta ordenação cronológica e prevenindo vazamentos de informação entre períodos distintos.

Código 3: Substituição de Strings 'None' por NaN

```
df = df.replace("None", np.nan)
```

Código 4: Conversão da Coluna de Data para o Tipo datetime

```
df['data'] = pd.to_datetime(df['data'], format='%Y-%m-%d')
```

Na sequência, procedeu-se à análise de outliers, que revelou a presença de partidas com número atípico de cartões vermelhos ou faltas. Optou-se por manter esses valores, pois representam eventos reais e relevantes no contexto esportivo, cuja exclusão reduziria a capacidade do modelo de lidar com situações extremas. Para mitigar seu impacto em algoritmos sensíveis à escala, aplicou-se normalização via StandardScaler.

Código 5: Aplicação do Standard Scaler para Padronização dos Dados

```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
df[cols_numericas] = scaler.fit_transform(df[cols_numericas])
```

4.1.3 Engenharia de Atributos

A engenharia de atributos representou uma das etapas mais importantes do processo, pois consistiu em transformar dados brutos em informações capazes de expressar tendências reais do desempenho das equipes e o contexto das partidas. Para ampliar o poder explicativo do conjunto de variáveis além das estatísticas isoladas de uma partida, foram incorporados atributos temporais, relacionais e contextuais, sempre garantindo que apenas informações anteriores ao jogo fossem utilizadas, evitando vazamento temporal.

O primeiro passo consistiu em reestruturar os dados no nível "time-partida", isto é, cada partida foi convertida em duas observações: uma do ponto de vista do mandante e outra do ponto de vista do visitante. Essa transformação viabilizou a criação de variáveis derivadas como gols pró, gols sofridos e pontos obtidos em cada contexto, permitindo a agregação do desempenho histórico por equipe.

Em seguida, foi aplicado o cálculo de médias móveis (Rolling Mean) em janela fixa de cinco jogos, representando a forma recente. Todas as variáveis foram defasadas com shift(1) para garantir que a previsão do jogo no instante t utilizasse apenas dados disponíveis até $t - 1$. O código abaixo ilustra a construção das médias móveis defasadas por time.

Código 6: Cálculo de médias móveis (5 jogos) com defasagem temporal

```
window_size = 5
features_to_roll = ['gols_pro', 'gols_sofridos', 'pontos',
                    'chutes', 'chutes_no_alvo', 'posse_de_bola',
                    'passes', 'precisao_passes', 'escanteios', 'faltas']

rolling_stats = (
    df_team_matches.groupby('time')[features_to_roll]
    .apply(lambda x: x.rolling(window=window_size, min_periods=1).mean().
        shift(1))
    .reset_index(level=0, drop=True)
)
```

Além das estatísticas gerais, foram construídas também médias móveis condicionais ao contexto de mando de campo, de modo a separar o desempenho do clube quando atua como mandante e quando atua como visitante. Essa decisão foi baseada na hipótese de que a vantagem de jogar em casa altera significativamente o comportamento de muitas equipes.

Outra adição importante foi a incorporação de atributos relacionais de confronto direto. Para cada partida, foram computadas estatísticas dos últimos cinco confrontos anteriores entre as duas equipes, como número de vitórias do mandante atual, vitórias do visitante atual, empates e saldo de gols. O trecho abaixo representa a lógica aplicada.

Código 7: Exemplo de construção de atributos de confronto direto

```
def h2h_features(row, window=5):
    a, b, dt = row['mandante'], row['visitante'], row['data']
    hist = df_partidas[
        (((df_partidas['mandante']==a)&(df_partidas['visitante']==b)) |
         ((df_partidas['mandante']==b)&(df_partidas['visitante']==a))) &
        (df_partidas['data'] < dt)
    ].sort_values('data').tail(window)
    # retorna vitorias, empates e saldo no recorte
```

Adicionalmente, foram incorporadas variáveis contextuais para aproximar nuances comportamentais do campeonato. A primeira foi o *momento do campeonato*, calculado por meio de uma proxy de rodada por temporada (ano), gerando uma variável contínua normalizada (fase_campeonato) e uma indicação binária de reta final (reta_final). O código abaixo ilustra essa construção.

Código 8: Rodada e momento do campeonato

```
df_partidas['ano'] = df_partidas['data'].dt.year
df_partidas['rodada'] = df_partidas.groupby('ano').cumcount() + 1
df_partidas['rodadas_no_ano'] = df_partidas.groupby('ano')['rodada'].
    transform('max')
```

```
df_partidas['fase_campeonato'] = df_partidas['rodada'] / df_partidas['
    rodadas_no_ano']
df_partidas['reta_final'] = (df_partidas['fase_campeonato'] >= 0.75).astype(
    int)
```

Por fim, foi criada uma variável indicadora de confronto regional, a partir de um mapeamento time–UF, assumindo que jogos entre equipes do mesmo estado tendem a apresentar dinâmicas específicas. Dessa forma, o conjunto final de atributos passou a representar: (i) forma recente e desempenho técnico, (ii) diferenças de mando/visitante, (iii) histórico do confronto direto e (iv) momento competitivo do campeonato, resultando em uma base mais robusta e metodologicamente adequada para a etapa de modelagem.

4.2
Aplicação dos modelos e Resultados

A primeira etapa da avaliação experimental consistiu na modelagem do problema em sua forma nativa: a classificação multiclasse (Mandante, Empate, Visitante).

4.2.1
Cenário com empates

A primeira etapa da avaliação experimental consistiu na modelagem do problema em sua forma nativa: a classificação multiclasse (Mandante, Empate, Visitante). Nesta configuração, os modelos foram treinados para prever o desfecho da partida em três categorias: vitória do mandante (2), empate (1) e vitória do visitante (0).

A distribuição de classes no conjunto de teste apresentou 808 vitórias do mandante, 483 empates e 466 vitórias do visitante, o que caracteriza um cenário moderadamente desbalanceado e, principalmente, desafiador pela natureza da classe intermediária. Apesar da inclusão de atributos históricos e contextuais (forma recente, desempenho como mandante/visitante, confronto direto, confronto regional e momento do campeonato), os algoritmos apresentaram limitações de desempenho, com acurácias na faixa de aproximadamente 35% a 47%. As métricas macro reforçaram a dificuldade de tratar todas as classes de forma equilibrada, indicando maior confusão sobretudo na classe empate.

A tabela abaixo apresenta os resultados observados para os modelos avaliados no cenário multiclasses (valores conforme execução reportada).

Observa-se que, embora alguns modelos atinjam acurácia próxima de 47%, os valores de F1-Score macro permanecem baixos, indicando dificuldade de generalização equilibrada entre as três classes. As métricas ponderadas

Modelo	Acurácia	Prec. (macro)	Rec. (macro)	F1 (macro)	Prec. (weighted)	Rec. (weighted)	F1 (weigh
Naive Bayes	0.4109	0.3816	0.3929	0.3670	0.4108	0.4109	0.3943
XGBoost	0.4485	0.3823	0.3661	0.3386	0.4022	0.4485	0.3894
KNN (K=20)	0.4513	0.3716	0.3658	0.3310	0.3932	0.4513	0.3837
Regressão Logística	0.4684	0.4120	0.3706	0.3232	0.4258	0.4684	0.3806
Gradient Boosting	0.4508	0.3820	0.3615	0.3251	0.4004	0.4508	0.3790
Random Forest	0.4730	0.4239	0.3642	0.2953	0.4339	0.4730	0.3591
SVM (Kernel RBF)	0.4724	0.4093	0.3565	0.2704	0.4206	0.4724	0.3394
AdaBoost	0.4616	0.4380	0.3424	0.2427	0.4441	0.4616	0.3158

Tabela 4.5: Desempenho dos Modelos no Cenário Multiclasse (com empates)

(weighted) apresentam desempenho ligeiramente superior, refletindo o maior peso da classe "vitória do mandante" no conjunto de dados. O Naive Bayes destacou-se no F1 ponderado (0.394), enquanto o Random Forest e o SVM obtiveram as maiores acurácias, porém com baixo F1 macro, evidenciando limitações na identificação da classe empate.

4.2.2 Cenário sem empates

Devido ao desempenho limitado no cenário multiclasse, foi decidido testar uma formulação alternativa do problema: classificação binária, removendo a classe "Empate". A hipótese é que as estatísticas de desempenho e forma recente são mais informativas para indicar superioridade técnica (quando há vencedor) do que para modelar a igualdade de placar.

Após a remoção dos empates, a distribuição de classes passou a apresentar maior desbalanceamento, com 820 vitórias do mandante e 473 vitórias do visitante no conjunto de teste. Nesse cenário, observou-se aumento de desempenho em métricas globais, com acurácia na faixa de aproximadamente 64%–65% e F1-Score (classe positiva: vitória do mandante) em torno de 0.76–0.78.

A tabela abaixo apresenta os resultados obtidos no cenário binário.

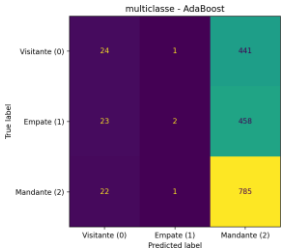
Apesar do aumento do F1-Score e da acurácia, a avaliação por classe revelou um efeito relevante: o modelo tende a privilegiar a previsão de vitórias do mandante. No melhor resultado observado (AdaBoost), o recall para a classe "Vitória do Visitante" foi de apenas 0.10, enquanto o recall para "Vitória do Mandante" atingiu 0.97. Isso indica que parte do ganho observado decorre do desbalanceamento de classes e do viés histórico favorável ao mandante, reduzindo a capacidade do classificador em identificar corretamente vitórias do visitante.

Tabela 4.6: Desempenho dos Modelos no Cenário Binário (sem empates)

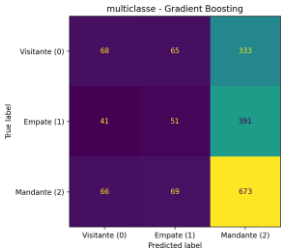
Modelo	**Acurácia**	**Precisão**	**Recall**	**F1-Score**
AdaBoost	0.6527	0.6522	0.9695	0.7798
SVM (Kernel RBF)	0.6473	0.6475	0.9744	0.7780
Random Forest	0.6466	0.6514	0.9524	0.7737
KNN (K=20)	0.6466	0.6616	0.9061	0.7648
Regressão Logística	0.6404	0.6548	0.9159	0.7636
XGBoost	0.6404	0.6567	0.9073	0.7619
Gradient Boosting	0.6342	0.6553	0.8927	0.7558
MLP (Rede Neural)	0.5770	0.6639	0.6744	0.6691
Naive Bayes	0.5770	0.7191	0.5463	0.6209

Dessa forma, embora a remoção dos empates torne o problema mais previsível ao eliminar a classe mais instável, o cenário binário exige métricas que penalizem desequilíbrios entre classes (por exemplo, F1 macro ou acurácia balanceada), além de estratégias de compensação como ponderação de classes (class_weight='balanced') e ajuste de limiar de decisão, para que a melhoria não seja apenas aparente.

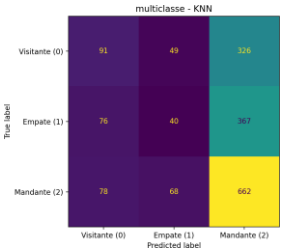
4.2.3
Matriz de Confusão Multiclasse



(a) AdaBoost

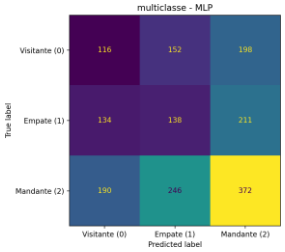


(b) Gradient Boosting

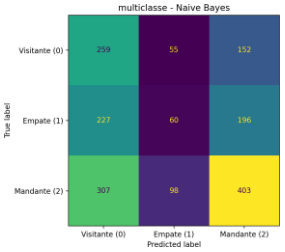


(c) KNN

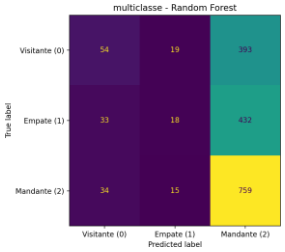
(d) Regressão Logística



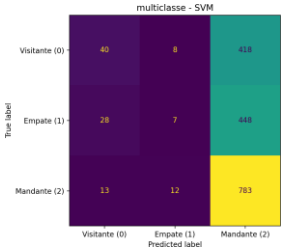
(e) MLP



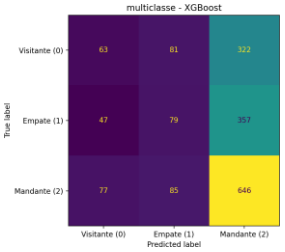
(f) Naive Bayes



(g) Random Forest



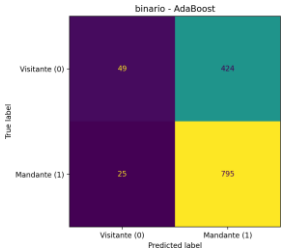
(h) SVM



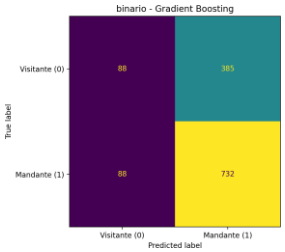
(i) XGBoost

Figura 4.1: Matrizes de confusão dos modelos avaliados no cenário multiclasse

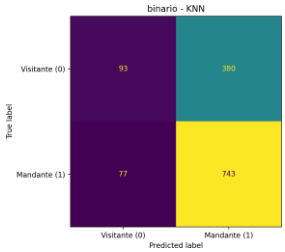
4.2.4
Matriz de Confusão Binária



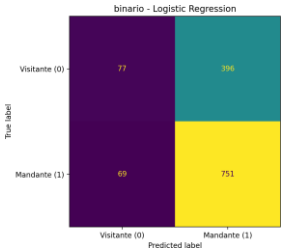
(a) AdaBoost



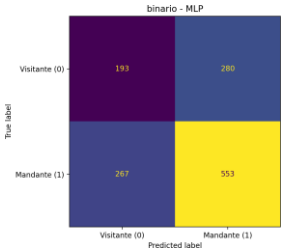
(b) Gradient Boosting



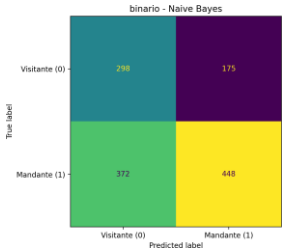
(c) KNN



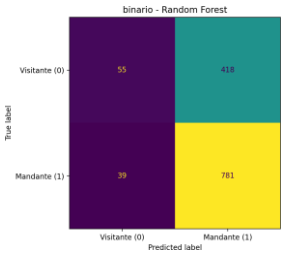
(d) Regressão Logística



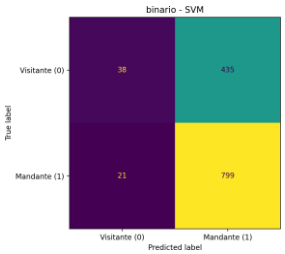
(e) MLP



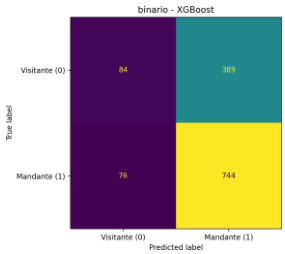
(f) Naive Bayes



(g) Random Forest



(h) SVM



(i) XGBoost

Figura 4.2: Matrizes de confusão dos modelos avaliados no cenário binário

4.2.5
Análise das Matrizes de Confusão: Cenário Binário

No cenário binário, o problema de classificação foi reformulado para distinguir exclusivamente entre *vitória do mandante* e *vitória do visitante*, com a exclusão deliberada dos empates. Essa reformulação teve como objetivo avaliar a capacidade dos modelos em identificar a superioridade técnica relativa entre as equipes, eliminando o ruído introduzido pela classe empate.

A análise das matrizes de confusão revela um padrão consistente entre os diferentes algoritmos: observa-se um forte viés estrutural em favor da classe "Mandante vence". Esse comportamento é coerente tanto com a distribuição histórica dos resultados do Campeonato Brasileiro quanto com a própria dinâmica do futebol nacional, em que o fator casa exerce influência significativa sobre o desfecho das partidas.

De modo geral, a maior parte das predições corretas concentra-se no quadrante correspondente aos Verdadeiros Positivos, isto é, partidas em que a vitória do mandante foi corretamente prevista. O erro mais recorrente consiste na classificação de vitórias do visitante como vitórias do mandante (Falsos Positivos). Por outro lado, os casos em que o modelo prevê vitória do visitante quando o mandante vence (Falsos Negativos) são relativamente raros, indicando que os algoritmos dificilmente subestimam o efeito do mando de campo.

Os modelos baseados em ensembles, como XGBoost, Random Forest, Gradient Boosting e AdaBoost, apresentaram matrizes de confusão fortemente concentradas na diagonal principal, especialmente no acerto da classe mandante. Esses algoritmos demonstraram elevada capacidade de capturar padrões associados ao desempenho em casa, com baixa taxa de Falsos Negativos. Entretanto, persistiu a dificuldade em identificar vitórias do visitante, refletida no número expressivo de Falsos Positivos.

Entre os modelos lineares e baseados em distância, a Regressão Logística apresentou um comportamento relativamente mais equilibrado, ainda que inclinada à classe majoritária. O SVM exibiu o viés mais acentuado, com raríssimas predições de vitória do visitante, caracterizando um classificador conservador que privilegia fortemente o mandante. O KNN, embora apresente maior recuperação de vitórias visitantes em comparação ao SVM, mantém uma assimetria clara nas decisões.

A Rede Neural Multicamadas (MLP) mostrou uma distribuição mais equilibrada entre erros e acertos das duas classes, porém com um número total de erros superior aos modelos ensemble, sugerindo que a maior flexibilidade do modelo não foi suficiente para explorar plenamente os padrões temporais e contextuais presentes nos dados.

Em síntese, o cenário binário evidenciou que as estatísticas técnicas e contextuais derivadas da engenharia de atributos são fortes preditores da vitória do mandante. A previsão de vitórias do visitante permanece um desafio, possivelmente por depender de fatores não capturados pelos dados disponíveis, como aspectos táticos específicos, decisões de arbitragem ou variáveis psicológicas. Ainda assim, o desempenho observado representa um ganho substancial em relação ao cenário multiclasse, confirmando a adequação metodológica da exclusão dos empates.

4.2.6 Análise das Matrizes de Confusão: Cenário Multiclasse

No cenário multiclasse, os modelos foram treinados para classificar os resultados das partidas em três categorias: *vitória do visitante*, *empate* e *vitória do mandante*. A análise das matrizes de confusão revela, de forma consistente entre os diferentes algoritmos, uma dificuldade estrutural em modelar a classe empate.

Três padrões principais emergem das matrizes analisadas. O primeiro é a elevada taxa de acerto da classe ”Mandante vence”, com forte concentração de observações corretamente classificadas. O segundo é a baixa recuperação da classe empate, que frequentemente é confundida com vitórias do mandante. O terceiro padrão consiste na confusão cruzada entre empate e vitória do visitante, especialmente nos modelos probabilísticos.

Os modelos baseados em boosting, como XGBoost e Gradient Boosting, apresentaram o melhor equilíbrio geral no cenário multiclasse. Ainda assim, a maior parte dos empates reais foi classificada como vitória do mandante, resultando em baixo *recall* para a classe empate. Isso indica dificuldade dos modelos em distinguir empates de vitórias apertadas em casa, mesmo quando informações de desempenho recente e confronto direto são consideradas.

Os algoritmos Random Forest e AdaBoost exibiram comportamento fortemente enviesado. Em particular, o AdaBoost praticamente ignorou a classe empate, operando na prática como um classificador quase binário. O SVM apresentou o caso mais extremo: a esmagadora maioria das observações, independentemente da classe real, foi classificada como vitória do mandante, evidenciando incapacidade de aprender fronteiras de decisão adequadas para três classes nesse contexto.

O Naive Bayes foi o único modelo a apresentar uma recuperação relativamente maior da classe visitante, porém às custas de elevada confusão entre visitante e empate e de uma acurácia global inferior. A Rede Neural Multicamadas apresentou maior dispersão nas predições, distribuindo melhor os erros entre as classes, mas sem conseguir formar regiões de decisão bem definidas para o empate.

A análise das matrizes de confusão reforça a hipótese central deste trabalho: o empate não constitui uma classe estatisticamente estável sob o ponto de vista das variáveis técnicas de desempenho utilizadas. Empates podem ocorrer tanto em partidas de baixo volume ofensivo quanto em jogos amplamente dominados por uma equipe, mas com baixa eficiência de finalização. Essa heterogeneidade dificulta a identificação de padrões consistentes, levando os modelos a absorverem a classe empate pela classe majoritária.

De forma comparativa, o cenário multiclasse apresentou desempenho próximo ao baseline ingênuo, enquanto o cenário binário permitiu uma modelagem mais eficaz do fenômeno analisado. As matrizes de confusão demonstram que os erros cometidos pelos modelos não são aleatórios, mas sistemáticos, o que fortalece a validade das conclusões obtidas e justifica a adoção da abordagem binária como principal estratégia de modelagem neste estudo.

4.2.7 Motivação para Expansão da Base de Dados

Embora os resultados obtidos até este ponto sejam considerados satisfatórios do ponto de vista metodológico, especialmente no cenário binário, a análise crítica dos modelos e das matrizes de confusão evidenciou limitações relevantes que restringem o potencial preditivo da abordagem adotada. Em particular, observou-se que, mesmo com uma engenharia de atributos robusta, os modelos apresentaram dificuldade consistente em prever vitórias do visitante e, no cenário multiclasse, em capturar padrões associados ao empate.

Essas limitações não decorrem exclusivamente da escolha dos algoritmos ou da parametrização adotada, mas estão fortemente relacionadas à natureza dos dados utilizados. As variáveis disponíveis concentram-se majoritariamente em estatísticas técnicas de desempenho durante a partida (como posse de bola, finalizações e passes), bem como em indicadores derivados do retrospecto recente das equipes. Embora essas informações sejam eficazes para representar superioridade técnica média, elas se mostram insuficientes para capturar fatores latentes que influenciam o resultado final, especialmente em jogos equilibrados ou fora de casa.

Diante desse cenário, optou-se por avançar para uma nova etapa do estudo, buscando uma base de dados mais atualizada e semanticamente mais rica. Foi identificado um dataset complementar, contendo informações oriundas do mercado de apostas esportivas, incluindo odds pré-jogo, probabilidades implícitas e variações de mercado ao longo do tempo. Esse tipo de informação sintetiza, de forma indireta, o conhecimento coletivo de analistas, casas de apostas e do próprio mercado, incorporando expectativas sobre forma das equipes, desfalques, mando de campo, importância da partida e outros fatores contextuais que não estão explicitamente presentes nas estatísticas tradicionais.

A inclusão de dados de betting representa uma ampliação conceitual do problema, pois adiciona ao modelo uma dimensão de *expectativa prévia* sobre o resultado da partida, complementando as métricas objetivas de desempenho histórico. Estudos recentes na literatura indicam que odds de apostas fun-

cionam como fortes preditores agregados, uma vez que refletem informações públicas e privadas processadas de maneira eficiente pelo mercado.

Assim, embora os resultados obtidos com a base inicial validem a eficácia da engenharia de atributos e da modelagem proposta, a busca por um novo dataset mais completo justifica-se como uma tentativa de superar as limitações observadas, elevar o desempenho preditivo dos modelos e aprofundar a análise sobre os fatores determinantes dos resultados no futebol brasileiro. A próxima seção apresenta a descrição dessa nova base de dados e as adaptações metodológicas realizadas para sua incorporação ao pipeline de modelagem.

4.2.8 Nova Base de Dados: FootyStats (Dataset com Odds de Apostas)

Para essa nova etapa, foi adotado um dataset extraído do portal FootyStats (*footystats.org*), contendo partidas do Campeonato Brasileiro (Série A) organizadas por temporada. A base recebida cobre o intervalo de 2013 a 2024, totalizando 4560 partidas (380 jogos por temporada), com variáveis que incluem tanto estatísticas de jogo quanto informações típicas do mercado de apostas (*betting odds*). Esse tipo de variável é especialmente relevante, pois sintetiza a expectativa pré-jogo do mercado e agrega informações contextuais que dificilmente aparecem em estatísticas tradicionais, como força relativa percebida, influência do mando de campo, momento do campeonato, desfalques e percepção coletiva sobre o confronto.

Entre os principais campos disponíveis, destacam-se: identificação temporal da partida (data/hora), nomes das equipes mandante e visitante, gols (tempo normal e primeiro tempo), informações de contexto (rodada), indicadores pré-jogo (por exemplo, PPG pré-partida) e, principalmente, odds para diferentes mercados, como resultado final (mandante/empate/visitante), over/under de gols e *both teams to score* (BTTS). Essas odds podem ser transformadas em probabilidades implícitas e utilizadas como atributos numéricos adicionais, permitindo avaliar empiricamente o quanto a informação do mercado contribui para elevar o desempenho preditivo dos modelos.

Cabe ressaltar que, durante a inspeção inicial, observou-se a presença de valores iguais a zero em algumas colunas numéricas (incluindo odds e variáveis de xG), o que não é interpretável como valor real nessas métricas e, portanto, foi tratado como ausência de dado (NaN) para posterior imputação no pipeline de modelagem. A próxima seção descreve o procedimento de carregamento, padronização e integração dessas temporadas em uma única tabela analítica, além das novas features derivadas especificamente das odds de apostas.

Coluna	**Descrição**
id	Identificador único da partida.
date_GMT	Data e horário da partida (GMT).
season	Temporada do campeonato.
game_week	Número da rodada (Game Week) da temporada.
status	Status da partida (ex.: finalizada, adiada).
stadium_name	Nome do estádio onde a partida foi realizada.
attendance	Público presente no estádio.
referee	Árbitro responsável pela partida.
home_team_name	Nome da equipe mandante.
away_team_name	Nome da equipe visitante.
home_goal_count	Gols marcados pelo mandante no tempo normal.
away_goal_count	Gols marcados pelo visitante no tempo normal.
total_goal_count	Total de gols da partida.
home_goal_count_half_time	Gols do mandante no primeiro tempo.
away_goal_count_half_time	Gols do visitante no primeiro tempo.
total_goals_at_half_time	Total de gols no primeiro tempo.
home_shots	Total de finalizações do mandante.
away_shots	Total de finalizações do visitante.
home_shots_on_target	Finalizações no alvo do mandante.
away_shots_on_target	Finalizações no alvo do visitante.
home_shots_off_target	Finalizações para fora do mandante.
away_shots_off_target	Finalizações para fora do visitante.
home_possession	Percentual de posse de bola do mandante.
away_possession	Percentual de posse de bola do visitante.
home_corner_count	Total de escanteios do mandante.
away_corner_count	Total de escanteios do visitante.

Continua na próxima página

Coluna	Descrição
home_yellow_cards	Total de cartões amarelos do mandante.
away_yellow_cards	Total de cartões amarelos do visitante.
home_red_cards	Total de cartões vermelhos do mandante.
away_red_cards	Total de cartões vermelhos do visitante.
home_fouls	Total de faltas cometidas pelo mandante.
away_fouls	Total de faltas cometidas pelo visitante.
home_team_xg	Expected Goals (xG) do mandante na partida.
away_team_xg	Expected Goals (xG) do visitante na partida.
home_pre_match_xg	Expected Goals (xG) estimado para o mandante antes da partida.
away_pre_match_xg	Expected Goals (xG) estimado para o visitante antes da partida.
home_ppg	Pontos por jogo (PPG) do mandante na temporada.
away_ppg	Pontos por jogo (PPG) do visitante na temporada.
home_pre_match_ppg	Pontos por jogo (PPG) do mandante antes da partida.
away_pre_match_ppg	Pontos por jogo (PPG) do visitante antes da partida.
odds_ft_home_team_win	Odds pré-jogo para vitória do mandante (mercado 1X2).
odds_ft_draw	Odds pré-jogo para empate (mercado 1X2).
odds_ft_away_team_win	Odds pré-jogo para vitória do visitante (mercado 1X2).
odds_btts_yes	Odds pré-jogo para o mercado BTTS (ambas marcam – sim).

Continua na próxima página

Coluna	Descrição
odds_btts_no	Odds pré-jogo para o mercado BTTS (ambas marcam – não).
odds_over15	Odds pré-jogo para Over 1.5 gols.
odds_over25	Odds pré-jogo para Over 2.5 gols.
odds_over35	Odds pré-jogo para Over 3.5 gols.
odds_over45	Odds pré-jogo para Over 4.5 gols.
over15_percentage_pre_match	Percentual pré-jogo estimado para Over 1.5 gols.
over25_percentage_pre_match	Percentual pré-jogo estimado para Over 2.5 gols.
over35_percentage_pre_match	Percentual pré-jogo estimado para Over 3.5 gols.
over45_percentage_pre_match	Percentual pré-jogo estimado para Over 4.5 gols.
btts_percentage_pre_match	Percentual pré-jogo estimado para o mercado BTTS.
average_goals_per_match_pre_match	Média de gols por partida estimada antes do jogo.
average_corners_per_match_pre_match	Média de escanteios por partida estimada antes do jogo.
average_cards_per_match_pre_match	Média de cartões por partida estimada antes do jogo.

Tabela 4.7: Estrutura do Dataset FootyStats – Campeonato Brasileiro (2013–2024)

4.3 Análise Exploratória dos Dados

Antes da aplicação dos modelos de aprendizado de máquina, foi realizada uma análise exploratória dos dados com o objetivo de compreender a estrutura do dataset, identificar padrões estatísticos relevantes e avaliar a presença de outliers. Essa etapa é fundamental para garantir que as decisões de engenharia de atributos e modelagem sejam fundamentadas em evidências empíricas extraídas dos dados.

O conjunto de dados analisado contempla estatísticas detalhadas de partidas do Campeonato Brasileiro Série A, incluindo informações ofensivas,

defensivas e disciplinares, tanto para o time mandante quanto para o visitante. Todas as variáveis analisadas apresentaram taxa de valores ausentes igual a zero, o que garante consistência e confiabilidade para as análises subsequentes.

4.3.1 Análise Descritiva das Variáveis

Tabela 4.8: Estatísticas descritivas das principais variáveis do dataset FootyStats

Variável	Média	Mediana	Desv. Pad.	Mín.	Máx.
home_team_goal_count	1.41	1.00	1.09	0	5
away_team_goal_count	1.03	1.00	0.99	0	6
home_team_shots	11.20	11.00	3.77	2	25
away_team_shots	8.88	9.00	3.52	1	22
home_team_shots_on_target	5.10	5.00	2.37	0	14
away_team_shots_on_target	4.06	4.00	2.30	0	18
home_team_corner_count	5.68	5.00	2.74	0	14
away_team_corner_count	4.49	4.00	2.57	0	12
home_team_yellow_cards	2.45	2.00	1.51	0	8
away_team_yellow_cards	2.69	3.00	1.54	0	9
home_team_fouls	12.83	13.00	3.72	4	26
away_team_fouls	12.45	12.00	3.80	3	24
home_team_possession (%)	51.54	52.50	10.73	22	78
away_team_possession (%)	48.46	47.50	10.73	22	78
home_ppg	1.69	1.74	0.41	0.74	2.47
away_ppg	1.05	0.89	0.38	0.53	2.00

A Tabela 4.8 apresenta as principais estatísticas descritivas das variáveis selecionadas, incluindo média, mediana, desvio padrão, valores mínimo e máximo.

De forma geral, observa-se uma vantagem consistente do time mandante em métricas ofensivas. A média de gols do mandante foi de 1,41, enquanto a do visitante foi de 1,03, evidenciando o impacto do fator casa. Esse comportamento também se reflete no número médio de finalizações (11,20 contra 8,88) e chutes no alvo (5,10 contra 4,05).

A posse de bola apresentou distribuição relativamente equilibrada, com média de 51,5% para o mandante e 48,5% para o visitante. A proximidade entre média e mediana indica distribuições aproximadamente simétricas, sugerindo estabilidade nessa métrica ao longo das partidas.

Em relação às variáveis disciplinares, os cartões vermelhos apresentaram média inferior a 0,2 por partida para ambos os lados, caracterizando-se como eventos raros. Já os cartões amarelos e o número de faltas cometidas exibiram maior variabilidade, especialmente para os times visitantes, indicando maior propensão a comportamentos defensivos sob pressão.

4.3.2
Identificação de Outliers

A análise por meio de boxplots revelou a presença de valores extremos em variáveis como número de gols, finalizações, chutes no alvo e faltas cometidas. Esses outliers correspondem a partidas atípicas, como jogos com placares elásticos, alto volume ofensivo ou elevada intensidade física.

Optou-se por não remover tais observações, uma vez que elas representam eventos reais e relevantes do ponto de vista esportivo. Além disso, modelos baseados em árvores de decisão e métodos de ensemble, amplamente utilizados neste trabalho, são robustos à presença de outliers e capazes de capturar padrões não lineares associados a esse tipo de ocorrência.

4.3.3
Análise de Correlação

A matriz de correlação entre as principais variáveis evidencia relações coerentes do ponto de vista futebolístico. Observa-se forte correlação positiva entre o número de chutes e chutes no alvo, bem como entre chutes no alvo e gols marcados, validando a consistência estatística do dataset.

Por outro lado, a posse de bola apresentou correlação fraca com o número de gols, indicando que o controle da posse, por si só, não garante maior efetividade ofensiva. Esse resultado reforça a importância de métricas relacionadas à criação de oportunidades e eficiência nas finalizações.

A posse de bola do mandante e do visitante apresentou correlação negativa próxima de −1, o que era esperado, uma vez que essa variável é naturalmente complementar entre os dois times.

4.3.4
Considerações da Análise Exploratória

Os resultados da análise exploratória forneceram subsídios importantes para a etapa de engenharia de atributos e modelagem. A identificação da vantagem do mando de campo, a relevância das métricas ofensivas e a baixa incidência de eventos extremos reforçaram a escolha de variáveis e justificaram o uso de modelos capazes de lidar com distribuições assimétricas e relações não lineares.

A partir dessas observações, procedeu-se à construção de atributos derivados e à aplicação dos modelos de classificação apresentados na seção seguinte.

4.3.5
Análise dos Atributos

Após a análise exploratória dos dados, foi realizada uma avaliação qualitativa e quantitativa dos atributos utilizados na modelagem preditiva. O objetivo dessa etapa foi compreender a relevância estatística e semântica das variáveis, bem como seu potencial impacto na predição do resultado das partidas.

Os atributos disponíveis no conjunto de dados podem ser agrupados em quatro grandes categorias: desempenho ofensivo, desempenho defensivo, controle de jogo e histórico pré-partida.

4.3.5.1
Desempenho Ofensivo

Os atributos relacionados ao desempenho ofensivo incluem o número de gols marcados, finalizações totais, finalizações no alvo e escanteios. Observa-se que o número médio de gols do mandante (1.41) é superior ao do visitante (1.03), evidenciando o efeito do mando de campo. Esse padrão também se repete nas métricas de finalizações e escanteios, nas quais os times mandantes apresentam médias consistentemente superiores.

Essas variáveis são diretamente associadas à criação de oportunidades de gol e, consequentemente, possuem forte correlação com o desfecho da partida. Modelos de aprendizado de máquina tendem a explorar essas diferenças para identificar padrões que favorecem a vitória do mandante ou do visitante.

4.3.5.2
Desempenho Defensivo e Disciplina

As métricas defensivas e disciplinares incluem faltas cometidas, cartões amarelos e cartões vermelhos. Os valores médios de faltas cometidas são semelhantes entre mandantes e visitantes, com ligeira vantagem para os mandantes. Já a média de cartões amarelos é maior para os visitantes, sugerindo maior pressão defensiva ou dificuldade em conter o adversário fora de casa.

Os cartões vermelhos apresentam baixa frequência média, porém seu impacto em partidas individuais é significativo. Embora raros, esses eventos extremos introduzem alta variabilidade e podem alterar drasticamente o resultado da partida, o que contribui para a presença de outliers observados nos boxplots dessas variáveis.

4.3.5.3
Controle de Jogo

A posse de bola apresenta distribuição complementar entre mandante e visitante, com média de aproximadamente 51.5% para o mandante e 48.5% para o visitante. Essa variável apresenta maior dispersão, indicando diferentes estilos de jogo entre as equipes e estratégias específicas para determinadas partidas.

Embora posse de bola não implique diretamente em vitória, sua combinação com métricas ofensivas, como finalizações e chutes no alvo, fornece informações relevantes sobre a dominância do time ao longo da partida.

4.3.5.4
Histórico Pré-Partida

Os atributos de desempenho acumulado, como *points per game* (PPG) pré-jogo, representam o momento das equipes antes da partida. Observa-se que o PPG médio do mandante é consideravelmente superior ao do visitante, tanto no histórico geral quanto nas métricas pré-jogo.

Esses atributos incorporam informações temporais e contextuais do campeonato, permitindo que os modelos capturem tendências de forma mais robusta do que métricas isoladas de uma única partida. Em particular, o PPG pré-jogo mostrou-se um dos atributos mais informativos para diferenciar partidas equilibradas de confrontos com favoritismo claro.

4.3.5.5
Outliers e Variabilidade

A análise por meio de boxplots revelou a presença de outliers em diversas variáveis, especialmente em número de gols, finalizações e cartões. Esses valores extremos refletem a natureza imprevisível do futebol, com partidas atípicas influenciadas por expulsões, goleadas ou estratégias específicas.

Em vez de serem removidos, esses outliers foram mantidos, uma vez que representam eventos reais e relevantes para o domínio do problema. A exclusão desses casos poderia reduzir a capacidade dos modelos de generalizar para situações menos frequentes, porém decisivas.

4.4
Paraâmetros de entrada dos modelos

No experimento de classificação binária com a base FootyStats, cujo objetivo foi prever a vitória do time mandante em oposição aos demais desfechos (empate ou vitória do visitante), o vetor de entrada dos modelos

foi composto exclusivamente por variáveis disponíveis no período pré-jogo. Essas variáveis foram organizadas em grupos conceituais que visam capturar três dimensões fundamentais do desempenho esportivo: a força estrutural das equipes, o momento recente de forma e o efeito do contexto competitivo e do mando de campo.

Inicialmente, foram consideradas variáveis de identificação e contexto da partida. Os atributos *home_code* e *away_code* correspondem à codificação numérica das equipes mandante e visitante, permitindo ao modelo aprender padrões históricos associados a cada clube, mesmo sem utilizar diretamente seus nomes. A variável *rodada* representa a posição temporal da partida dentro do campeonato, enquanto *fase_campeonato* segmenta a temporada em macroperíodos (início, meio e final), possibilitando a captura de mudanças de comportamento tático e estratégico ao longo do torneio. O atributo binário *reta_final* indica se o jogo ocorre nas rodadas finais, período caracterizado por maior pressão competitiva, disputas por títulos, vagas em competições internacionais ou luta contra o rebaixamento. Por fim, *confronto_regional* identifica clássicos ou jogos entre equipes geograficamente próximas, que tendem a apresentar padrões distintos de intensidade, motivação e imprevisibilidade.

Em seguida, foram incorporadas variáveis de confronto direto (*head-to-head*), construídas por meio de engenharia de atributos a partir do histórico de partidas anteriores entre as duas equipes. Essas variáveis incluem o número total de confrontos prévios (*h2h_games*), o número de vitórias do mandante (*h2h_home_wins*), vitórias do visitante (*h2h_away_wins*), empates (*h2h_draws*) e o saldo agregado de gols do confronto (*h2h_goal_diff*). Esse conjunto de atributos busca capturar padrões específicos de dominância histórica ou equilíbrio entre determinados pares de equipes, fenômeno amplamente observado no futebol profissional.

Outro grupo relevante corresponde às probabilidades implícitas de mercado, extraídas das odds pré-jogo. As variáveis *p_home*, *p_draw* e *p_away* representam, respectivamente, as probabilidades de vitória do mandante, empate e vitória do visitante. Adicionalmente, *p_gap_home_away* expressa a diferença entre as probabilidades de vitória do mandante e do visitante, funcionando como uma medida direta de favoritismo relativo, enquanto *odds_gap_home_away* representa a diferença equivalente no espaço das odds. Essas variáveis sintetizam a informação agregada de analistas, apostadores e modelos especializados, incorporando fatores como elenco, lesões, momento, mando de campo e percepção pública.

A forma recente das equipes foi modelada por meio de médias móveis calculadas sobre os cinco jogos imediatamente anteriores a cada partida. Para o

mandante, foram computadas médias de gols marcados e sofridos, pontos conquistados, número de chutes e chutes no alvo, posse de bola, escanteios, faltas, cartões amarelos e vermelhos, além do valor médio de *expected goals (xG)*. Também foram incluídas proporções de vitórias, empates e derrotas no período, bem como um índice sintético de forma (*home_media_forma1_ult_5*), que resume o desempenho recente em um único escalar.

De forma análoga, foram construídas variáveis equivalentes para o visitante, com prefixo *away_media_*, representando seu desempenho médio nos últimos cinco jogos, e variáveis condicionadas ao mando, com prefixo *home_media_side_* e *away_media_side_*, que consideram apenas partidas em que o time atuou, respectivamente, como mandante ou como visitante. Essas chamadas *side features* são fundamentais para capturar o efeito do mando de campo.

No experimento de classificação multiclasse com a base FootyStats, cujo objetivo foi prever simultaneamente vitória do mandante, empate ou vitória do visitante, o conjunto de atributos de entrada permaneceu idêntico ao descrito anteriormente, sendo alterada apenas a variável alvo.

Nos experimentos baseados na base Transfermarkt, tanto no cenário binário quanto no multiclasse, a estrutura conceitual do vetor de entrada manteve-se semelhante, porém sem a inclusão de variáveis de mercado, uma vez que a base não disponibiliza odds de apostas. Assim, foram utilizados atributos de identificação das equipes, contexto competitivo, histórico de confrontos diretos e forma recente. As variáveis de *head-to-head* incluem o número de jogos anteriores entre as equipes, vitórias do mandante, vitórias do visitante, empates e saldo de gols no histórico recente.

A forma recente foi representada por médias móveis dos últimos cinco jogos, contemplando gols marcados e sofridos, pontos, chutes no alvo, posse de bola, escanteios, faltas, estatísticas relacionadas à construção de jogo, tais como número médio de passes e precisão de passes. As proporções de vitórias, empates e derrotas e um índice agregado de forma também foram incluídos. De maneira análoga ao FootyStats, foram calculadas versões condicionadas ao mando (*home_media_side_* e *away_media_side_*), capturando diferenças sistemáticas de desempenho em casa e fora.

É importante ressaltar que as variáveis com prefixos *home_media_*, *away_media_*, *side_* e *h2h_* não fazem parte das bases originais. Elas foram obtidas pelo processo de engenharia de atributos temporal, onde, para cada partida, apenas informações provenientes de jogos anteriores foram utilizadas, garantindo a inexistência de vazamento de informação e assegurando que o problema tratado pelos modelos reflita um cenário real de previsão pré-jogo.

Grupo	Conjunto de Variáveis	Descrição
Identificação	*home_code*, *away_code*	Codificação numérica das equipes
Contexto da competição	*rodada*, *fase_campeonato*, *reta_final*, *confronto_regional*	Estágio do campeonato e rivalidade
Confronto direto (H2H)	*h2h_games*, *h2h_home_wins*, *h2h_away_wins*, *h2h_draws*, *h2h_goal_diff*	Histórico recente entre as equipes
Mercado (odds)	*p_home*, *p_draw*, *p_away*, *p_gap_home_away*, *odds_gap_home_away*	Probabilidades implícitas e favoritismo
Forma recente – Mandante	*home_media_*_ult_5*	Desempenho médio nos últimos 5 jogos
Forma recente – Visitante	*away_media_*_ult_5*	Desempenho médio nos últimos 5 jogos
Mando de campo (side)	*home_media_side_ult_5*, *away_media_side_ult_5*	Desempenho específico em casa/fora

Tabela 4.9: Grupos de atributos utilizados (FootyStats – Binário e Multiclasse)

Métrica	Variável
Gols marcados	**_media_gols_pro_ult_5*
Gols sofridos	**_media_gols_sofridos_ult_5*
Pontos	**_media_pontos_ult_5*
Chutes	**_media_chutes_ult_5*
Chutes no alvo	**_media_chutes_no_alvo_ult_5*
Posse de bola	**_media_posse_de_bola_ult_5*
Escanteios	**_media_escanteios_ult_5*
Faltas	**_media_faltas_ult_5*
Cartões amarelos	**_media_cartao_amarelo_ult_5*
Cartões vermelhos	**_media_cartao_vermelho_ult_5*
Expected Goals (xG)	**_media_xg_ult_5*
Vitórias	**_media_win_ult_5*
Empates	**_media_draw_ult_5*
Derrotas	**_media_loss_ult_5*
Índice de forma	**_media_forma1_ult_5*

Tabela 4.10: Variáveis de Forma Recente (exemplo padrão para cada equipe)

*Obs.: O prefixo * é substituído por home ou away e, no caso de mando, por home_media_side ou away_media_side.*

Grupo	Conjunto de Variáveis	Descrição
Identificação	*mandante_code*, *visitante_code*	Codificação das equipes
Contexto	*rodada*, *fase_campeonato*, *reta_final*, *confronto_regional*	Situação competitiva
Confronto direto (H2H)	*h2h_jogos_ult_n*, *h2h_vitorias_mandante_ult_n*, *h2h_vitorias_visitante_ult_n*, *h2h_empates_ult_n*, *h2h_saldo_gols_mandante_ult_n*	Histórico recente do duelo
Forma recente	*home_media_ult_5*, *away_media_ult_5*	Estatísticas médias dos últimos jogos
Construção de jogo	**_media_passes_ult_5*, **_media_precisao_passes_ult_5*	Volume e qualidade de passes
Mando de campo (side)	*home_media_side_ult_5*, *away_media_side_ult_5*	Desempenho específico em casa/fora

Tabela 4.11: Grupos de atributos (Transfermarkt – Binário e Multiclasse)

Métrica	Variável
Passes	**_media_passes_ult_5*
Precisão de passes	**_media_precisao_passes_ult_5*
Posse de bola	**_media_posse_de_bola_ult_5*
Escanteios	**_media_escanteios_ult_5*
Faltas	**_media_faltas_ult_5*

Tabela 4.12: Métricas adicionais exclusivas da base Transfermarkt

4.5 Resultados

Nesta seção são apresentados os resultados obtidos a partir da aplicação dos modelos de aprendizado de máquina sobre o dataset do *FootyStats*, considerando dois cenários distintos: (i) classificação multiclasse, contemplando vitória do mandante, empate e vitória do visitante; e (ii) classificação binária, na qual os empates foram removidos e o problema foi reformulado como vitória do mandante versus vitória do visitante.

Todos os experimentos utilizaram uma divisão temporal dos dados, respeitando a ordem cronológica das partidas, com aproximadamente 80% das observações destinadas ao conjunto de treino e 20% ao conjunto de teste. As métricas adotadas para avaliação foram acurácia, precisão, *recall* e F1-score, considerando médias macro e ponderadas quando aplicável.

4.5.1 Resultados no Cenário Multiclasse

No cenário multiclasse, o objetivo foi prever o desfecho completo da partida, incluindo a possibilidade de empate. A Tabela 4.13 apresenta o desempenho dos modelos avaliados, ordenados de acordo com o F1-score ponderado.

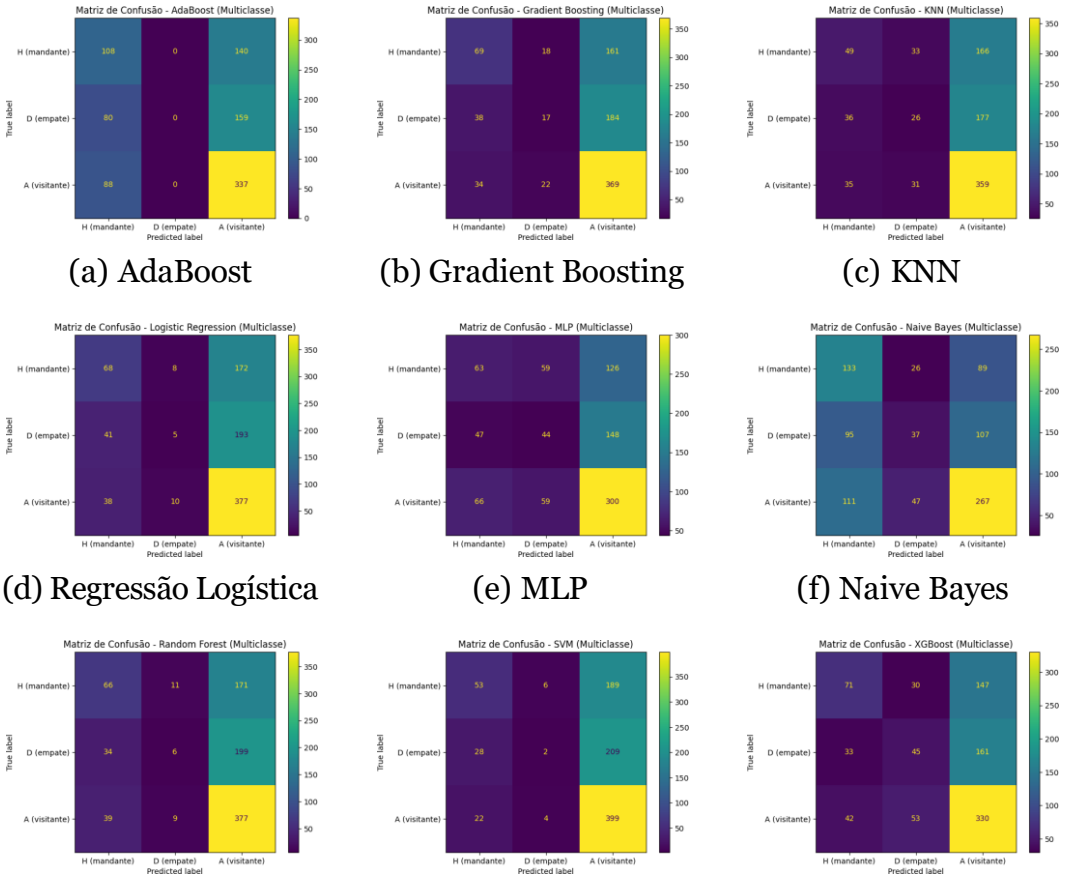
De maneira geral, observa-se que os resultados obtidos são superiores aos alcançados com a base anterior, evidenciando o impacto positivo da utilização de um dataset mais recente e enriquecido com estatísticas detalhadas de jogo. O modelo *Naive Bayes* apresentou o melhor desempenho global neste cenário, alcançando um F1-score ponderado de aproximadamente 0.46, seguido por XGBoost e Gradient Boosting.

Tabela 4.13: Desempenho dos modelos no cenário multiclasse (vitória do mandante, empate e vitória do visitante)

Modelo	Acurácia	Precisão (macro)	Recall (macro)	F1 (weighted)
Naive Bayes	0.479	0.435	0.440	**0.459**
XGBoost	0.489	0.452	0.417	0.452
Gradient Boosting	0.502	0.443	0.409	0.433
Random Forest	0.496	0.424	0.398	0.412
KNN	0.476	0.403	0.384	0.411
AdaBoost	0.488	0.307	0.409	0.408
Logistic Regression	0.493	0.396	0.394	0.405
MLP	0.406	0.377	0.376	0.403
SVM	0.498	0.394	0.387	0.391

A análise do relatório de classificação do melhor modelo indica que a classe “vitória do mandante” é a mais bem prevista, refletindo tanto o viés histórico de mando de campo quanto o desbalanceamento natural das classes. Em contrapartida, a classe “empate” permanece como a mais desafiadora, apresentando baixos valores de *recall*, o que reforça a complexidade inerente à previsão desse tipo de resultado.

4.5.2
Matriz de Confusão Multiclasse - FootyStats



(a) AdaBoost (b) Gradient Boosting (c) KNN

(d) Regressão Logística (e) MLP (f) Naive Bayes

(g) Random Forest (h) SVM (i) XGBoost

Figura 4.3: Matrizes de confusão dos modelos avaliados no cenário multiclasse

4.5.3
Resultados no Cenário Binário

No cenário binário, os empates foram removidos do conjunto de dados, e o problema foi reformulado como uma tarefa de classificação entre vitória do mandante e vitória do visitante. Essa simplificação reduziu a ambiguidade do problema e resultou em ganhos expressivos de desempenho para a maioria dos modelos.

Conforme apresentado na Tabela 4.14, o modelo *SVM* obteve o melhor desempenho, alcançando um F1-score de aproximadamente 0.79, seguido de perto por Random Forest, Gradient Boosting e XGBoost. Observa-se que, nesse cenário, todos os modelos apresentaram métricas significativamente superiores às do caso multiclasse.

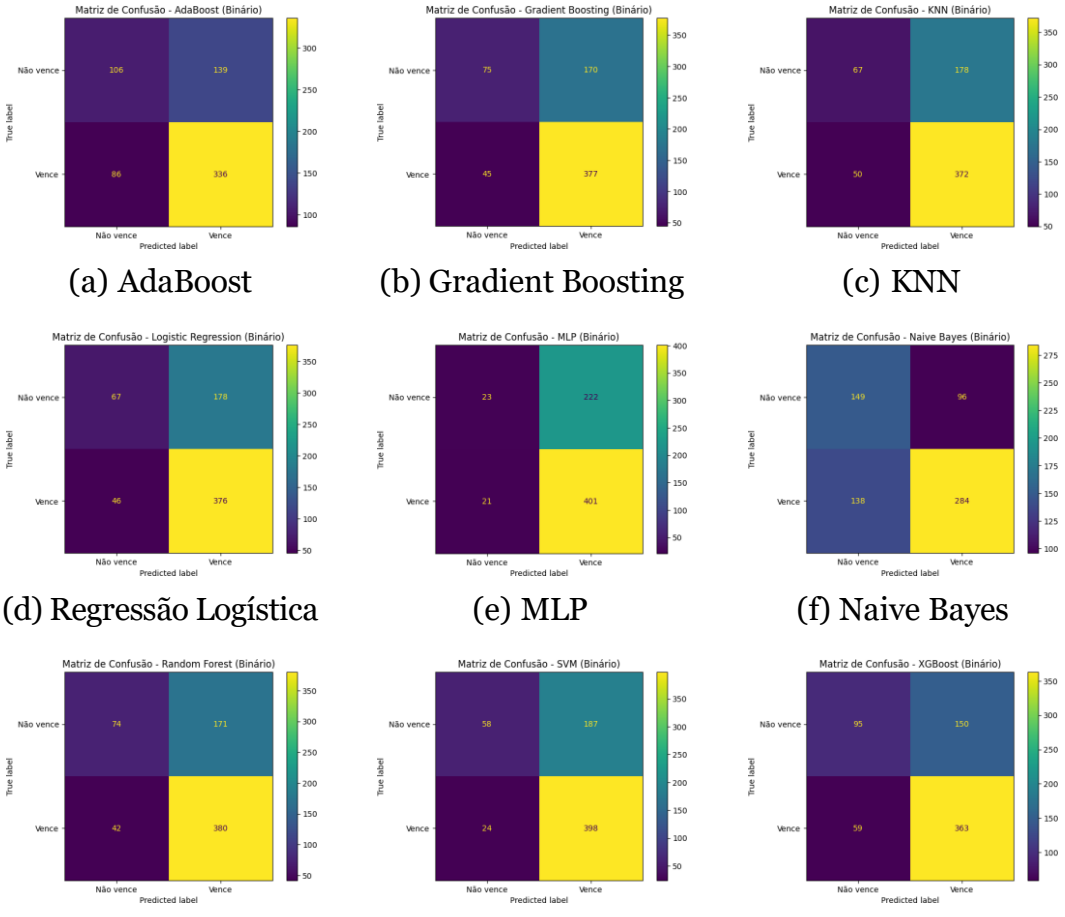
A análise detalhada do relatório de classificação do modelo SVM evidencia um elevado *recall* para a classe "vitória do mandante", indicando que o

Tabela 4.14: Desempenho dos modelos no cenário binário (vitória do mandante vs. vitória do visitante)

Modelo	Acurácia	Precisão	Recall	F1
SVM	0.684	0.680	**0.943**	**0.790**
Random Forest	0.681	0.690	0.900	0.781
Gradient Boosting	0.678	0.689	0.893	0.778
XGBoost	0.687	0.708	0.860	0.776
Logistic Regression	0.664	0.679	0.891	0.770
KNN	0.658	0.676	0.882	0.765
AdaBoost	0.663	0.707	0.796	0.749
Naive Bayes	0.649	0.747	0.673	0.708
MLP	0.604	0.674	0.725	0.699

modelo é particularmente eficiente em identificar esse desfecho. Por outro lado, a previsão de vitórias do visitante ainda apresenta maior dificuldade, com menor sensibilidade, o que pode estar associado ao desbalanceamento residual entre as classes e à menor previsibilidade histórica desses eventos.

4.5.4 Matriz de Confusão Binário - FootyStats



(a) AdaBoost (b) Gradient Boosting (c) KNN

(d) Regressão Logística (e) MLP (f) Naive Bayes

(g) Random Forest (h) SVM (i) XGBoost

Figura 4.4: Matrizes de confusão dos modelos avaliados no cenário multiclasse

4.5.5 Discussão Geral dos Resultados

De forma geral, analisou-se que os dados esportivos vão sempre apresentar uma grande dificuldade para modelar bons resultados, dada a natureza dos dados. Parâmetros que não seguem padrões específicos e estatísticas que não necessáriamente jogam a favor de um vencedor, além dos fatores extra-campo dificultam ainda mais. Por isso, não se pode definir que a riqueza dos dados de aposta da FootyStats definiram um padrão melhor de resultados. Essa comparação só poderia ser feita se o pipeline fosse feito usando a base do FootyStats, sem os dados de apostas e depois com.

Além disso, a comparação entre os cenários evidencia que a presença da classe empate aumenta substancialmente a complexidade do problema, reduzindo as métricas globais de desempenho. Ainda assim, o cenário multiclasse permanece relevante do ponto de vista prático, por representar de forma mais fiel a natureza real do futebol.

Por fim, os resultados obtidos demonstram que, embora haja espaço para melhorias, especialmente na previsão de empates e vitórias do visitante. O modelo binário alcança níveis de desempenho compatíveis com aplicações práticas, enquanto o modelo multiclasse oferece uma base sólida para estudos futuros e refinamentos metodológicos.

5
Conclusão e trabalhos futuros

Este trabalho teve como objetivo investigar a aplicação de técnicas de aprendizado de máquina na predição de resultados de partidas de futebol, utilizando dados históricos do Campeonato Brasileiro da Série A. Para isso, foram explorados dois cenários distintos de modelagem: um problema multiclasse, considerando vitória do mandante, empate e vitória do visitante, e um problema binário, no qual os empates foram removidos.

Inicialmente, foi realizada uma análise exploratória detalhada dos dados, permitindo compreender a distribuição, variabilidade e possíveis outliers das principais métricas de desempenho das equipes. Essa etapa foi fundamental para validar a qualidade do conjunto de dados, identificar padrões relevantes e embasar as decisões tomadas na etapa de engenharia de atributos. Observou-se, por exemplo, a influência do mando de campo em métricas ofensivas, como gols e finalizações, bem como o impacto de atributos históricos, como o desempenho médio por partida (PPG).

A etapa de engenharia de atributos incorporou informações contextuais importantes, como desempenho recente das equipes, histórico de confrontos diretos, mando de campo, momento do campeonato e métricas agregadas de desempenho ofensivo e defensivo. Essa abordagem permitiu enriquecer a representação das partidas, tornando o problema mais informativo para os modelos de aprendizado de máquina.

Os resultados obtidos indicaram que o cenário binário apresentou desempenho significativamente superior ao cenário multiclasse. Enquanto os modelos multiclasse enfrentaram dificuldades em distinguir a classe empate, os modelos binários alcançaram métricas mais elevadas de acurácia e F1-score, com destaque para algoritmos como Support Vector Machines, Random Forest, Gradient Boosting e XGBoost. Esses resultados sugerem que a previsão de empates constitui um desafio intrínseco ao domínio do futebol, devido ao equilíbrio estatístico observado em muitas partidas.

Apesar dos bons resultados alcançados, especialmente no cenário binário, este trabalho apresenta algumas limitações. Entre elas, destaca-se a dificuldade em modelar eventos raros e altamente impactantes, como expulsões e goleadas atípicas, bem como a dependência de atributos estatísticos que, embora informativos, não capturam completamente fatores subjetivos do jogo, como aspectos psicológicos, decisões táticas em tempo real e contexto emocional das equipes.

Com o intuito de superar algumas dessas limitações, este estudo avançou na utilização de um conjunto de dados mais recente e mais completo, proveniente do site *FootyStats*, que inclui informações adicionais relacionadas ao mercado de apostas e métricas avançadas de desempenho. Uma comparação entre as duas bases so daria para ser feita se ambos os datasets fossem exatamente iguais. Somente daria para ter certeza completa de que os parâmetros de apostas evoluiram os resultados se fossem retirados do dataset do Footystats e feitos novamente.

Como trabalhos futuros, sugere-se a exploração de modelos temporais mais sofisticados, como redes neurais recorrentes e modelos baseados em séries temporais, bem como a integração de probabilidades do mercado de apostas como atributos adicionais. Além disso, a avaliação de métodos de balanceamento de classes e abordagens específicas para a classe empate pode contribuir para avanços no cenário multiclasse.

Por fim, conclui-se que o aprendizado de máquina se mostra uma ferramenta promissora para a análise e predição de resultados no futebol, desde que acompanhado de uma engenharia de atributos cuidadosa, análise exploratória aprofundada e compreensão do domínio. Os resultados obtidos reforçam a importância da qualidade dos dados e do contexto na construção de modelos preditivos mais robustos e interpretáveis.

6
Referências bibliográficas

1 MAHER, M. J. A simple model for the distribution of goals in association football. **Journal of the Royal Statistical Society: Series D (The Statistician)**, v. 31, n. 3, p. 241–248, 1982. Disponível em: https://onlinelibrary.wiley.com/doi/epdf/10.1111/j.1467-9574.1982.tb00782.x. Citado na página 13.

2 DIXON, M. J.; COLES, S. G. Modelling association football scores and inefficiencies in the football betting market. **Journal of the Royal Statistical Society: Series C (Applied Statistics)**, v. 46, n. 2, p. 265–280, 1997. Disponível em: https://www.ajbuckeconbikesail.net/wkpapers/Airports/MVPoisson/soccer_betting.pdf. Citado na página 13.

3 KONING, R. Sport and measurement of competition. **De Economist**, v. 157, n. 2, p. 229–249, 2009. Disponível em: https://pure.rug.nl/ws/portalfiles/portal/171911535/Sport_and_measurement_of_competition.pdf. Citado na página 13.

4 HUCALJUK, J.; RAKIPOVIC, A. Predicting football scores using machine learning techniques. In: **2011 Proceedings of the 34th International Convention MIPRO**. [s.n.], 2011. p. 1649–1652. ISBN 978-953-233-059-5. Disponível em: https://ieeexplore.ieee.org/document/5967321. Citado na página 13.

5 BREIMAN, L. Random forests. **Machine Learning**, v. 45, n. 1, p. 5–32, 2001. Disponível em: https://link.springer.com/article/10.1023/A:1010933404324. Citado 2 vezes nas páginas 13 e 21.

6 FRIEDMAN, J. H. Greedy function approximation: A gradient boosting machine. **The Annals of Statistics**, v. 29, n. 5, p. 1189–1232, 2001. Disponível em: https://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-approximation-A-gradient-boosting-machine/10.1214/aos/1013203451.full. Citado 2 vezes nas páginas 13 e 22.

7 BABOOTA, R.; KAUR, H. Predictive analysis and modelling football results using machine learning approach for english premier league. **International Journal of Performance Analysis in Sport**, v. 19, n. 6, p. 865–878, 2019. Acesso em: 14 jun. 2025. Disponível em: https://www.researchgate.net/publication/324072605_Predictive_analysis_and_modelling_football_results_using_machine_learning_approach_for_English_Premier_League. Citado na página 14.

8 GOLEC, J.; TAMARKIN, M. The degree of inefficiency in the football betting market: Statistical tests. **Journal of Financial Economics**, v. 30, n. 2, p. 311–323, 1991. Disponível em: https://www.academia.edu/59679536/The_degree_of_inefficiency_in_the_football_betting_market_Statistical_tests. Citado na página 14.

9 BRUCE, P.; BRUCE, A. **Practical Statistics for Data Scientists**. [S.l.]: O'Reilly Media, 2017. Citado na página 16.

10 SHALEV-SHWARTZ, S.; BEN-DAVID, S. **Understanding Machine Learning: From Theory to Algorithms**. [S.l.]: Cambridge University Press, 2014. Citado na página 20.

11 CHEN, T.; GUESTRIN, C. XGBoost: A scalable tree boosting system. In: **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)**. [S.l.: s.n.], 2016. Citado na página 23.

12 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. Citado na página 23.

13 MITCHELL, T. M. **Machine Learning**. [S.l.]: McGraw-Hill, 1997. Citado na página 24.

14 CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, v. 20, p. 273–297, 1995. Citado na página 25.

15 COVER, T.; HART, P. Nearest neighbor pattern classification. **IEEE Transactions on Information Theory**, v. 13, p. 21–27, 1967. Citado na página 25.

16 RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. **Nature**, v. 323, p. 533–536, 1986. Citado na página 26.

17 Transfermarkt. **Transfermarkt - O Maior Portal de Futebol do Mundo**. 2000. Website. Acesso em: 13 de dezembro de 2025. Disponível em: https://www.transfermarkt.com.br/. Citado na página 31.