LUCIO VARGAS DE ALBUQUERQUE NUNES

OTIMIZAÇÃO DA MONTAGEM DE ELENCO NO FUTEBOL: APRENDIZADO DE MÁQUINA PARA REDUÇÃO DA IMPREVISIBILIDADE EM CONTRATAÇÕES

LAVRAS – MG

2026

Ficha Catalográfica elaborada pelo Sistema de Geração de Ficha Catalográfica da Biblioteca Universitária da UFLA, com dados informados pelo(a) próprio(a) autor(a).

Nunes, Lucio Albuquerque.
Otimização da Montagem de Elenco no Futebol : Aprendizado de Máquina para Redução da Imprevisibilidade em Contratações / Lucio Albuquerque Nunes. - 2026.
87 p. : il.

Orientador: Dilson Lucas Pereira

Dissertação (Mestrado Acadêmico) - Universidade Federal de Lavras, 2026.
Bibliografia.

1. Futebol. 2. Aprendizado de Máquina. 3. Pesquisa Operacional. 4. Análise de Esportes. 5. Montagem de Elenco. I. Pereira, Dilson Lucas. II. Universidade Federal de Lavras. III. Título.

LUCIO VARGAS DE ALBUQUERQUE NUNES

OTIMIZAÇÃO DA MONTAGEM DE ELENCO NO FUTEBOL: APRENDIZADO DE MÁQUINA PARA REDUÇÃO DA IMPREVISIBILIDADE EM CONTRATAÇÕES

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, para a obtenção do título de Mestre.

Prof. Dr. Dilson Lucas Pereira
Orientador

LAVRAS – MG
2026

LUCIO VARGAS DE ALBUQUERQUE NUNES

OTIMIZAÇÃO DA MONTAGEM DE ELENCO NO FUTEBOL: APRENDIZADO DE MÁQUINA PARA REDUÇÃO DA IMPREVISIBILIDADE EM CONTRATAÇÕES

OPTIMIZATION OF SQUAD BUILDING IN FOOTBALL: MACHINE LEARNING FOR REDUCING UNCERTAINTY IN PLAYER RECRUITMENT

Dissertação apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Pós-Graduação em Ciência da Computação, para a obtenção do título de Mestre.

APROVADA em 23 de março de 2026.

Prof. Dr. Mayron César de Oliveira Moreira	UFLA
Prof. Dr. Cristiano Arbex Valle	UFMG

gov.br Documento assinado digitalmente
DILSON LUCAS PEREIRA
Data: 08/04/2026 10:02:34-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. Dilson Lucas Pereira
Orientador

gov.br Documento assinado digitalmente
CRISTIANO ARBEX VALLE
Data: 07/04/2026 10:32:57-0300
Verifique em https://validar.iti.gov.br

gov.br Documento assinado digitalmente
MAYRON CESAR DE OLIVEIRA MOREIRA
Data: 06/04/2026 21:36:36-0300
Verifique em https://validar.iti.gov.br

LAVRAS – MG
2026

Aos meus pais, pelo apoio incondicional e por tornar cada jornada minha possível.

AGRADECIMENTOS

Agradeço, em primeiro lugar, ao meu orientador, Prof. Dr. Dilson Lucas Pereira, pela excelente orientação, pela paciência ao longo de todo o processo e pela confiança depositada no desenvolvimento deste trabalho.

Ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Lavras (UFLA), bem como à própria UFLA, que desde a graduação até o mestrado me proporcionou a estrutura acadêmica necessária para minha formação.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), pelo apoio e fomento que contribuíram para a realização deste trabalho.

Aos meus pais, Mario e Gilda, por toda a base, incentivo e apoio incondicional ao longo da minha trajetória.

À minha namorada, Ana Paula, pelo apoio constante e pela compreensão durante toda a jornada do mestrado.

Ao meu primo Júlio, por acreditar com entusiasmo no meu projeto e sempre me incentivar a seguir em frente.

À professora Evelise, minha orientadora na graduação, que teve papel fundamental no início da minha jornada na programação e na formação que tornou possível chegar até aqui.

Aos professores Luiz Merschmann, Paulo Afonso e Mayron, cujos ensinamentos e contribuições foram fundamentais ao longo do mestrado.

À Josiane, pela dedicação e apoio na resolução de questões burocráticas relacionadas à defesa.

Por fim, agradeço aos colegas Danilo e João, pela companhia durante as aulas da pós-graduação.

"Quanto maior o artista, maior a dúvida; uma confiança perfeita é concedida aos menos talentosos como prêmio de consolação."

(Robert Hughes)

RESUMO

O futebol, o esporte mais popular do mundo, é marcado por sua alta imprevisibilidade, o que dificulta a montagem de elencos, as escolhas táticas e a seleção de jogadores para cada partida. Este estudo explora o uso de técnicas de Aprendizado de Máquina (AM) e Pesquisa Operacional (PO) como ferramentas de apoio à tomada de decisão no contexto esportivo. O objetivo é desenvolver um modelo preditivo capaz de estimar o desempenho dos clubes com base em dados estatísticos individuais dos jogadores. A partir dessas previsões, são aplicados algoritmos de otimização para orientar estratégias de contratações, permitindo decisões mais eficientes e fundamentadas em dados. Essa abordagem visa reduzir a subjetividade nos processos de gestão esportiva, promovendo maior assertividade nas decisões de mercado por parte dos clubes.

Palavras-chave: futebol; aprendizado de máquina; pesquisa operacional; análise de esportes; montagem de elenco.

ABSTRACT

Football, the most popular sport in the world, is marked by a high degree of unpredictability, which makes squad building, tactical planning, and player selection particularly challenging. This study explores the use of Machine Learning (ML) and Operations Research (OR) techniques as decision-support tools in the sports context. The objective is to develop a predictive model capable of estimating club performance based on individual player statistics. These predictions are then used as inputs for optimization algorithms aimed at guiding player recruitment strategies, enabling more efficient and data-driven decisions. This approach seeks to reduce subjectivity in sports management processes, promoting greater accuracy in market decisions and contributing to a more strategic, evidence-based decision-making framework within football clubs.

Keywords: football; machine learning; operations research; sports analytics; squad building.

INDICADORES DE IMPACTO

O presente trabalho apresenta impactos predominantemente tecnológicos e econômicos, com desdobramentos sociais em potencial, ao propor uma abordagem baseada em aprendizado de máquina e otimização para apoio à tomada de decisão na montagem de elencos no futebol. A metodologia desenvolvida integra técnicas de redução de dimensionalidade, redes neurais artificiais e heurísticas de busca local em um pipeline capaz de simular cenários de contratação sob restrições orçamentárias, contribuindo para a redução da imprevisibilidade associada a decisões tradicionalmente baseadas em critérios subjetivos. Do ponto de vista tecnológico, o estudo fortalece a aplicação da ciência de dados no contexto esportivo, oferecendo uma ferramenta replicável por clubes, analistas de desempenho e departamentos de scouting. Sob a perspectiva econômica, os resultados indicam potencial para otimizar a alocação de recursos financeiros, mitigando riscos em contratações e promovendo maior eficiência na gestão de elencos, especialmente em organizações com limitações orçamentárias. Os impactos sociais são considerados indiretos, uma vez que a melhoria na gestão esportiva pode influenciar positivamente a competitividade das equipes, a valorização de atletas e o engajamento de torcedores, além de contribuir para a formação de profissionais qualificados na interface entre tecnologia e esporte. O trabalho não apresenta caráter extensionista direto, mas possui potencial de aplicação prática em instituições esportivas e empresas do setor, com abrangência nacional e internacional. Em termos de classificação temática, enquadra-se principalmente na área de Tecnologia e Produção, com interfaces nas áreas de Educação e Trabalho. Ademais, a pesquisa apresenta alinhamento com os Objetivos de Desenvolvimento Sustentável da Organização das Nações Unidas, especialmente o ODS 8 (Trabalho Decente e Crescimento Econômico) e o ODS 9 (Indústria, Inovação e Infraestrutura), ao promover inovação tecnológica e maior eficiência econômica no contexto esportivo.

IMPACT INDICATORS

This work presents predominantly technological and economic impacts, with potential social implications, by proposing an approach based on machine learning and optimization to support decision-making in football squad building. The developed methodology integrates dimensionality reduction techniques, artificial neural networks, and local search heuristics into a unified pipeline capable of simulating transfer scenarios under budget constraints, contributing to reducing the uncertainty traditionally associated with subjective decision-making processes. From a technological perspective, the study advances the application of data science in the sports domain, providing a framework that can be adopted by clubs, performance analysts, and scouting departments. From an economic standpoint, the results indicate potential to optimize financial resource allocation, mitigating risks in player recruitment and promoting more efficient squad management, particularly in organizations with limited budgets. The social impacts are considered indirect, as improvements in sports management may positively influence team competitiveness, player valuation, and fan engagement, in addition to contributing to the training of professionals at the intersection of technology and sports. Although the work does not present a direct extension component, it has practical application potential in sports institutions and industry-related organizations, with both national and international scope. In terms of thematic classification, it is primarily aligned with the area of Technology and Production, with secondary connections to Education and Work. Furthermore, the research aligns with the United Nations Sustainable Development Goals, particularly SDG 8 (Decent Work and Economic Growth) and SDG 9 (Industry, Innovation and Infrastructure), by fostering technological innovation and improving economic efficiency in the sports context.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Futebol é o esporte mais popular do mundo (Almulla *et al.*, 2023). Nas últimas décadas, sua evolução tecnológica e analítica tem sido notável, acompanhada por um crescimento expressivo nos lucros gerados pelas atividades esportivas (Rodrigues; Pinto, 2023). No entanto, o futebol é um esporte altamente imprevisível. Um exemplo notável dessa imprevisibilidade foi a surpreendente conquista da English Premier League (PL) pelo Leicester City em 2016, um feito que surpreendeu especialistas e torcedores em todo o mundo (Baboota; Kaur, 2019).

Dada a relevância econômica e o impacto do futebol como forma de entretenimento, pesquisadores da academia e profissionais da indústria têm se dedicado ao desafio de prever seus resultados (Baboota; Kaur, 2019). O crescimento do mercado de apostas reflete essa busca, impulsionando também os lucros da indústria de prognósticos esportivos (Rodrigues; Pinto, 2023). Para se ter uma noção da magnitude econômica do futebol, entre 1º de junho e 1º de setembro de 2023, as transferências de jogadores movimentaram R$ 36,6 bilhões, conforme relatório divulgado pela Federação Internacional de Futebol (FIFA) (GLOBO Esporte, 2023). No Brasil, em 2025, a janela de transferências da Série A brasileira bateu a marca de R$ 1 bilhão em movimentações financeiras (GLOBO Esporte, 2025).

No campo da Ciência da Computação, diversos estudos têm explorado técnicas de análise de dados e estatística com o objetivo de modelar padrões e realizar predições em ambientes complexos. Um exemplo relevante é o estudo de Joseph, Fenton e Neil (2006), que investigou o uso de redes Bayesianas e outras técnicas de Aprendizado de Máquinas (AM) para prever resultados do Tottenham Hotspur entre 1995 e 1997. Outro caso notável de aplicação de dados no futebol envolve o Brighton & Hove Albion, clube inglês que, sob a gestão de Tony Bloom, implementou uma estratégia fortemente orientada por análises estatísticas para recrutamento e tomada de decisões. Essa filosofia refletiu-se diretamente na política de transferências do clube, com a identificação sistemática de atletas subvalorizados no mercado. Entre os exemplos mais emblemáticos está Moisés Caicedo, contratado junto ao Independiente Del Valle em 2021 por cerca de £4,5 milhões e negociado com o Chelsea em 2023 por valores próximos a 115 milhões €, tornando-se uma das maiores vendas da história da PL (BBC Sport, 2023; CNN Brasil, 2023). O mesmo caso de altos lucros em revendas se aplica a outros jogadores, como Alexis Mac Allister e Marc Cucurella, abordagem que permitiu que o Brighton se consolidasse como

um dos clubes mais eficientes do mercado de transferências da PL, reinvestindo os recursos obtidos e mantendo a competitividade sustentável na elite do futebol inglês (Olley, 2023).

Na literatura acadêmica, observa-se que a maioria dos trabalhos envolvendo AM no futebol concentra-se na previsão de resultados de partidas, probabilidades de vitória ou otimização de ganhos no contexto de apostas esportivas. Em contraste, menos estudos direcionam esforços à construção e otimização de elencos com foco na avaliação e seleção de jogadores (Abidin, 2021). Essa lacuna é particularmente relevante, uma vez que a escolha de um atleta para compor um elenco envolve a análise simultânea de múltiplos parâmetros técnicos, físicos e mentais, além de fatores contextuais como estilo de jogo da equipe, liga disputada e momento da carreira do jogador (Abidin, 2021). A natureza não determinística inerente ao futebol torna esse processo ainda mais desafiador, pois mesmo contratações que aparentam ser ideais sob múltiplos critérios podem não atingir o desempenho esperado.

Segundo Ian Graham (2024), uma transferência pode ser considerada um sucesso se o jogador iniciar 50% das partidas nas primeiras duas temporadas ("The 50% rule"). Considerando todas as transferências de mais de € 10 milhões que ocorreram na PL entre os anos de 1992 e 2021, perto da metade (46%) não atingiram o critério estabelecido e, portanto, podem ser classificadas como falhas (Graham, 2024). Entre os principais fatores associados a esses insucessos, destacam-se:

1. O jogador atual é melhor que o novo jogador;
2. O jogador não é tão bom quanto se pensava inicialmente;
3. O jogador não se encaixa no estilo do time;
4. O jogador joga fora de posição;
5. O técnico não aprova o jogador;
6. O jogador tem problemas de condicionamento físico ou pessoais.

Nos últimos anos, o AM tem ampliado significativamente a capacidade de análise de dados esportivos, permitindo a identificação de padrões antes imperceptíveis em grandes volumes de informação. Técnicas avançadas vêm sendo aplicadas para estimar desde a probabilidade de vitória em uma partida até o impacto de decisões táticas ao longo de uma temporada. Entretanto, tais modelos não eliminam a incerteza inerente ao futebol, atuando como ferramentas

de apoio à tomada de decisão humana, com o objetivo de reduzir riscos e embasar escolhas estratégicas.

Neste contexto, o presente trabalho investiga o uso de modelos preditivos para converter estatísticas históricas de desempenho individual em estimativas de desempenho esperado de jogadores quando inseridos em contextos específicos de equipe e temporada. Diferentemente de abordagens voltadas à previsão direta de resultados de partidas ou à avaliação de talento de forma isolada, a proposta concentra-se na análise contextualizada do jogador como parte de um sistema coletivo. As previsões geradas são utilizadas como subsídio em um processo de simulação e avaliação de cenários de contratação, visando apoiar decisões de mercado no futebol profissional de forma orientada por dados e alinhada a critérios de eficiência esportiva e financeira.

2 OBJETIVOS

2.1 Objetivo Geral

O presente estudo tem como objetivo principal desenvolver e validar um modelo de AM capaz de prever o desempenho de equipes de futebol a partir de dados estatísticos de desempenho individual de jogadores. Neste trabalho, entende-se por desempenho a pontuação esperada de uma equipe ao longo de uma temporada de uma competição esportiva.

Entretanto, o objetivo central não se limita ao desenvolvimento do modelo preditivo em si. Pretende-se que tal modelo seja utilizado como componente de algoritmos de otimização voltados ao apoio à tomada de decisão na montagem de elencos. Em particular, seja f o modelo de AM obtido e seja X um arranjo contendo as *features* correspondentes ao elenco de um determinado time. Embora o modelo possa ser utilizado para prever o desempenho de equipes existentes, uma vez treinado ele também pode ser empregado para estimar o desempenho de equipes hipotéticas.

Dessa forma, considere P o conjunto de todos os elencos X' que podem ser obtidos a partir do elenco atual X por meio da contratação de novos jogadores, respeitando um determinado orçamento disponível. O objetivo passa então a ser identificar, dentre todas as combinações possíveis de jogadores em P, aquela que maximiza o desempenho previsto pelo modelo de AM. Formalmente, busca-se resolver o problema definido pela Equação 2.1:

$$\max_{X' \in P} f(X') \tag{2.1}$$

Observa-se, portanto, que o problema estudado configura um problema de otimização combinatória, no qual o modelo de AM desempenha o papel de função objetivo.

2.2 Objetivos Específicos

Com o intuito de atingir o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

1. Coletar, estruturar e processar dados históricos de jogadores e partidas, abrangendo múltiplas temporadas, garantindo a qualidade e a integridade das informações utilizadas no desenvolvimento dos modelos preditivos.

2. Desenvolver uma metodologia para quantificar o impacto individual dos jogadores, combinando estatísticas históricas ao longo das temporadas por meio de um sistema de ponderação temporal, priorizando desempenhos mais recentes.

3. Construir modelos de AM para prever o desempenho agregado de uma equipe em uma determinada temporada, considerando a composição do elenco e os dados históricos dos jogadores.

4. Validar a eficácia dos modelos preditivos, comparando as estimativas com os resultados reais de temporadas subsequentes, utilizando métricas estatísticas como erro médio absoluto (MAE) e coeficiente de determinação (R^2).

5. Desenvolver modelos simples de otimização baseados no modelo de AM selecionado, visando a identificação de combinações de jogadores que maximizem o desempenho esperado da equipe sob restrições de mercado.

Ao integrar técnicas de AM e otimização, este estudo busca fornecer uma ferramenta analítica de apoio à decisão para a montagem de elencos no futebol profissional, contribuindo para práticas de gestão esportiva orientadas por dados.

3 REFERENCIAL TEÓRICO

Este capítulo apresenta os principais conceitos teóricos que fundamentam o desenvolvimento desta pesquisa. Inicialmente, são introduzidos os fundamentos de AM, destacando suas principais características e aplicações no contexto da modelagem preditiva. Na sequência, são abordados conceitos de PO, com ênfase em problemas de otimização combinatória e em métodos heurísticos que servem de base à abordagem de otimização da composição de elencos proposta neste trabalho.

3.1 Aprendizado de Máquina (AM)

Aprendizado de Máquina (AM) é o estudo de algoritmos e modelos estatísticos empregados por sistemas computacionais para realizar tarefas específicas de forma autônoma. Essa abordagem permite que as máquinas aprendam a partir de dados e melhorem seu desempenho sem necessidade de reprogramação explícita (Mahesh, 2020). Segundo Geron (2019), AM pode ser descrito como a ciência (ou arte) de programar computadores para que aprendam com os dados.

Um exemplo clássico de aplicação de AM é o filtro de *spam*, que permite que o sistema identifique automaticamente emails indesejados, diferenciando-os de mensagens legítimas com base em padrões aprendidos (Géron, 2019).

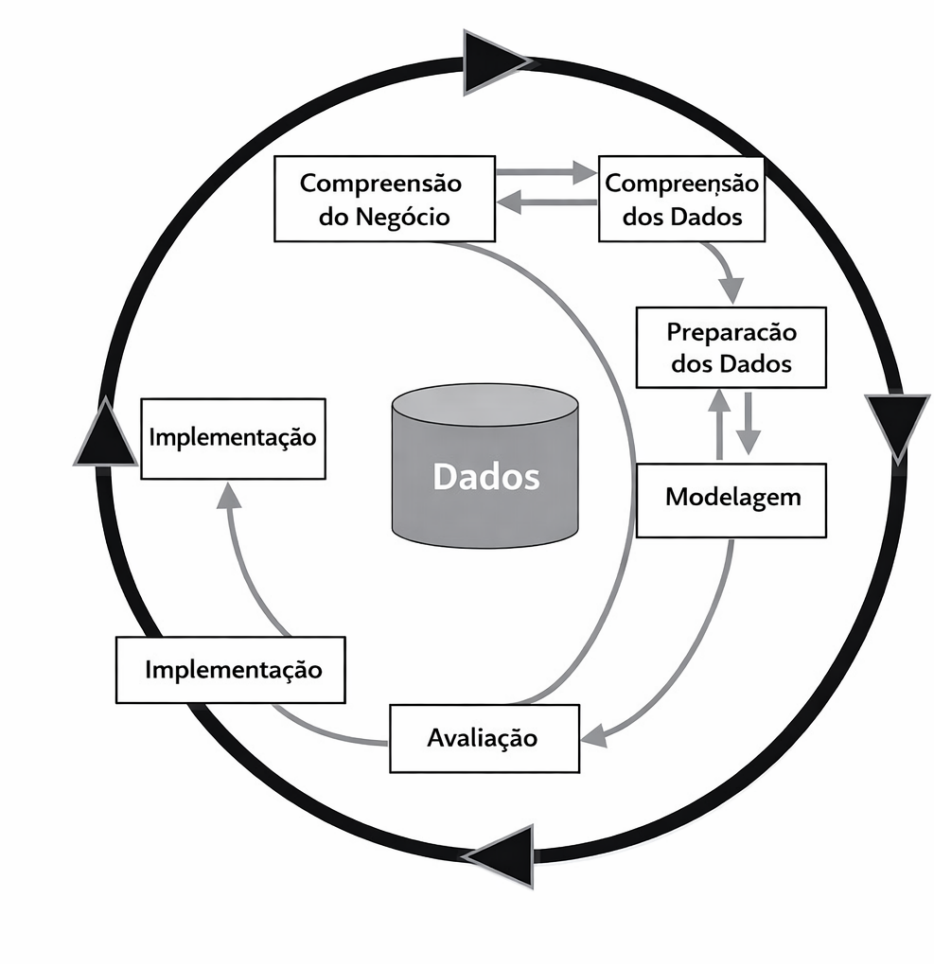
Com a crescente disponibilidade de *datasets*, o uso de AM tem apresentado um notável crescimento (Mahesh, 2020). Esse avanço pode ser atribuído à maior disponibilidade de dados e ao aumento da capacidade computacional. A constante conexão das pessoas a dispositivos digitais gera volumes massivos de dados, enquanto tecnologias modernas permitem o processamento eficiente dessas grandes quantidades de informação (Jordan; Mitchell, 2015).

Muitas indústrias têm adotado o AM com o objetivo de extrair insights relevantes de *databases*, facilitando o aprendizado a partir dos dados (Mahesh, 2020). Além disso, o AM tem sido amplamente aplicado em diversas áreas, incluindo reconhecimento facial, segmentação de imagens, detecção de câncer e previsão de classes e resultados, destacando-se também no contexto esportivo (Almulla *et al.*, 2023).

No desenvolvimento de sistemas de AM, a construção de modelos geralmente ocorre dentro de um pipeline composto por múltiplas etapas de processamento de dados. Em um

projeto típico de AM, os dados passam inicialmente por fases de coleta e preparação, seguidas por etapas de exploração e preparação dos dados, engenharia de atributos, seleção e treinamento de modelos, além de procedimentos de avaliação e ajuste de hiperparâmetros (Géron, 2019). Essas etapas são frequentemente organizadas em forma de pipeline, no qual cada componente executa uma transformação específica sobre os dados e produz uma saída que será utilizada pela etapa subsequente do processo (Géron, 2019). Essa estrutura modular permite organizar o fluxo de processamento e facilita a reprodução e manutenção dos modelos desenvolvidos. Um exemplo de organização dessas etapas no ciclo de vida de um projeto de análise de dados é apresentado na Figura 3.1, baseada no processo CRISP-DM, amplamente utilizado em projetos de mineração de dados e AM (Kelleher; Namee; D'Arcy, 2015).

Figura 3.1 – Pipeline CRISP-DM



Fonte: Adaptado de (Kelleher; Namee; D'Arcy, 2015)

Dentro desse pipeline, a preparação e transformação dos dados desempenham papel fundamental, uma vez que os algoritmos de aprendizado frequentemente exigem representações específicas dos atributos para funcionar adequadamente.

3.1.1 Generalização, Viés e Variância

Um dos principais desafios em AM é garantir que o modelo apresente boa capacidade de generalização, isto é, desempenho satisfatório em dados não utilizados durante o treinamento (Goodfellow; Bengio; Courville, 2016).

Modelos excessivamente complexos podem se ajustar não apenas aos padrões relevantes, mas também ao ruído presente no conjunto de treinamento, fenômeno conhecido como *overfitting*. Em contrapartida, modelos demasiadamente simples podem não capturar relações importantes nos dados, resultando em *underfitting*.

Esse comportamento está diretamente relacionado ao equilíbrio entre viés e variância. Modelos com alto viés tendem a apresentar subajuste, enquanto modelos com alta variância tornam-se excessivamente sensíveis às particularidades do conjunto de treinamento. Técnicas como validação cruzada, regularização e controle da complexidade do modelo são estratégias amplamente utilizadas para mitigar esses problemas (Géron, 2019).

3.1.2 Seleção de Atributos e Redução de Dimensionalidade

Em problemas com elevado número de variáveis explicativas, a presença de atributos redundantes ou irrelevantes pode impactar negativamente o desempenho dos modelos preditivos. A seleção de atributos busca identificar subconjuntos de variáveis que contribuam de forma significativa para a tarefa de aprendizagem (Liu *et al.*, 2010).

Os métodos de seleção podem ser classificados em três categorias principais: filtros, que utilizam medidas estatísticas independentes do modelo; wrappers, que avaliam subconjuntos de atributos com base no desempenho de um algoritmo específico; e métodos embutidos, nos quais a seleção ocorre durante o próprio processo de treinamento (Guyon *et al.*, 2006).

Além da seleção direta de atributos, técnicas de redução de dimensionalidade, como a Análise de Componentes Principais (PCA), realizam uma transformação linear das variáveis originais em um novo conjunto de componentes ortogonais que preservam a maior parte da

variância dos dados. Essa abordagem pode reduzir complexidade computacional e mitigar problemas de multicolinearidade. Cabe ressaltar que a PCA não consiste em um método de seleção de atributos, mas sim em uma técnica de extração de atributos, pois cria novas variáveis a partir de combinações lineares das variáveis originais (Guyon *et al.*, 2006).

Além dos aspectos relacionados à preparação e transformação dos dados, também é importante distinguir os principais paradigmas de aprendizagem utilizados na construção de modelos preditivos. Em AM, os algoritmos podem ser classificados em diferentes categorias de acordo com a forma como aprendem a partir dos dados. Uma das distinções mais comuns ocorre entre aprendizado supervisionado e não supervisionado. No aprendizado supervisionado, o modelo é treinado a partir de um conjunto de dados rotulados, no qual cada instância possui uma variável alvo associada. O objetivo do algoritmo é aprender uma função que relacione os atributos de entrada com essa variável alvo, permitindo realizar previsões para novas instâncias (Géron, 2019). Por outro lado, no aprendizado não supervisionado os dados não possuem rótulos explícitos, e o objetivo do algoritmo consiste em identificar padrões, estruturas ou agrupamentos presentes nos dados (Géron, 2019). Técnicas como *clustering* e redução de dimensionalidade, incluindo a Análise de Componentes Principais (PCA), são exemplos de métodos amplamente utilizados nesse contexto.

Em problemas de aprendizado supervisionado, as tarefas podem ser geralmente divididas em classificação e regressão. Na classificação, o objetivo é atribuir cada instância a uma classe discreta previamente definida, como por exemplo identificar se um e-mail é *spam* ou não (Géron, 2019). Já na regressão, busca-se prever valores numéricos contínuos a partir de um conjunto de variáveis explicativas, como a previsão de preços, temperaturas ou outras quantidades mensuráveis (Géron, 2019). No contexto deste trabalho, o problema é formulado como uma tarefa de regressão, uma vez que o objetivo consiste em estimar a pontuação final de equipes em um campeonato a partir das características individuais dos jogadores que compõem o elenco.

Após as etapas de preparação, transformação e possível redução de dimensionalidade dos dados, diferentes algoritmos de AM podem ser empregados para construir modelos preditivos. A seguir, são apresentados alguns dos algoritmos mais comumente utilizados para a resolução de problemas de regressão, abordagem adotada no modelo proposto neste trabalho.

Entre os diversos algoritmos disponíveis na literatura, foram selecionados neste trabalho métodos amplamente utilizados em aprendizado supervisionado para tarefas de regressão,

incluindo modelos lineares, métodos baseados em árvores, métodos de *boosting* e redes neurais artificiais.

3.1.3 Regressão Linear (RL)

A regressão linear é uma técnica estatística amplamente utilizada para investigar e modelar a relação entre variáveis, sendo aplicada em diversas áreas da ciência, engenharia e ciências sociais (Montgomery; Peck; Vining, 2012). Nesse método, assume-se que a variável dependente pode ser expressa como uma combinação linear das variáveis explicativas.

Na forma mais geral, a RL pode ser expressa conforme a Equação 3.1:

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p \tag{3.1}$$

em que $\hat{y}$ representa o valor previsto da variável resposta, β_0 é o intercepto do modelo, $\beta_1, \beta_2, \ldots, \beta_p$ são os coeficientes associados às variáveis explicativas e $x_1, x_2, \ldots, x_p$ representam os atributos de entrada.

O processo de treinamento da RL consiste em estimar os coeficientes do modelo de forma a minimizar o erro entre os valores previstos e os valores observados no conjunto de treinamento. Uma abordagem amplamente utilizada para esse ajuste é o método dos mínimos quadrados ordinários, que busca minimizar a soma dos quadrados dos resíduos (Géron, 2019). Dessa forma, o modelo encontra a combinação linear dos atributos que melhor se ajusta aos dados segundo esse critério.

A RL apresenta como principais vantagens a simplicidade, a interpretabilidade e o baixo custo computacional. Por se tratar de um modelo linear, seus coeficientes podem ser interpretados como a contribuição de cada variável explicativa para a previsão da variável alvo, mantidas constantes as demais variáveis. Além disso, a RL costuma servir como modelo de referência (*baseline*) em tarefas preditivas, permitindo comparar o desempenho de modelos mais complexos (Géron, 2019).

Entretanto, a RL possui limitações importantes, especialmente quando a relação entre as variáveis explicativas e a variável resposta não pode ser adequadamente representada por uma função linear. Nesses casos, o modelo tende a apresentar desempenho inferior ao de métodos capazes de capturar padrões não lineares mais complexos. Ainda assim, sua utilização perma-

nece relevante tanto pela simplicidade quanto pelo seu papel como ponto de comparação em estudos de modelagem preditiva (Géron, 2019).

3.1.4 Árvore de Decisão (AD)

As Árvores de Decisão são algoritmos versáteis de AM que podem ser utilizados tanto para tarefas de classificação, quanto para tarefas de regressão (Géron, 2019). Esses algoritmos recebem dados de treinamento e retornam como saída uma estrutura em árvore capaz de realizar previsões para novas instâncias com base nos atributos de entrada (Géron, 2019).

A construção de uma árvore de decisão ocorre por meio de um processo recursivo de particionamento do espaço de atributos. Em cada nó interno da árvore, o algoritmo seleciona um atributo e um ponto de divisão que melhor separam os dados de acordo com um critério de qualidade da partição. Em tarefas de classificação, critérios como índice de Gini e entropia são amplamente utilizados, enquanto em problemas de regressão são comuns medidas baseadas na redução da variância ou do erro quadrático (Géron, 2019).

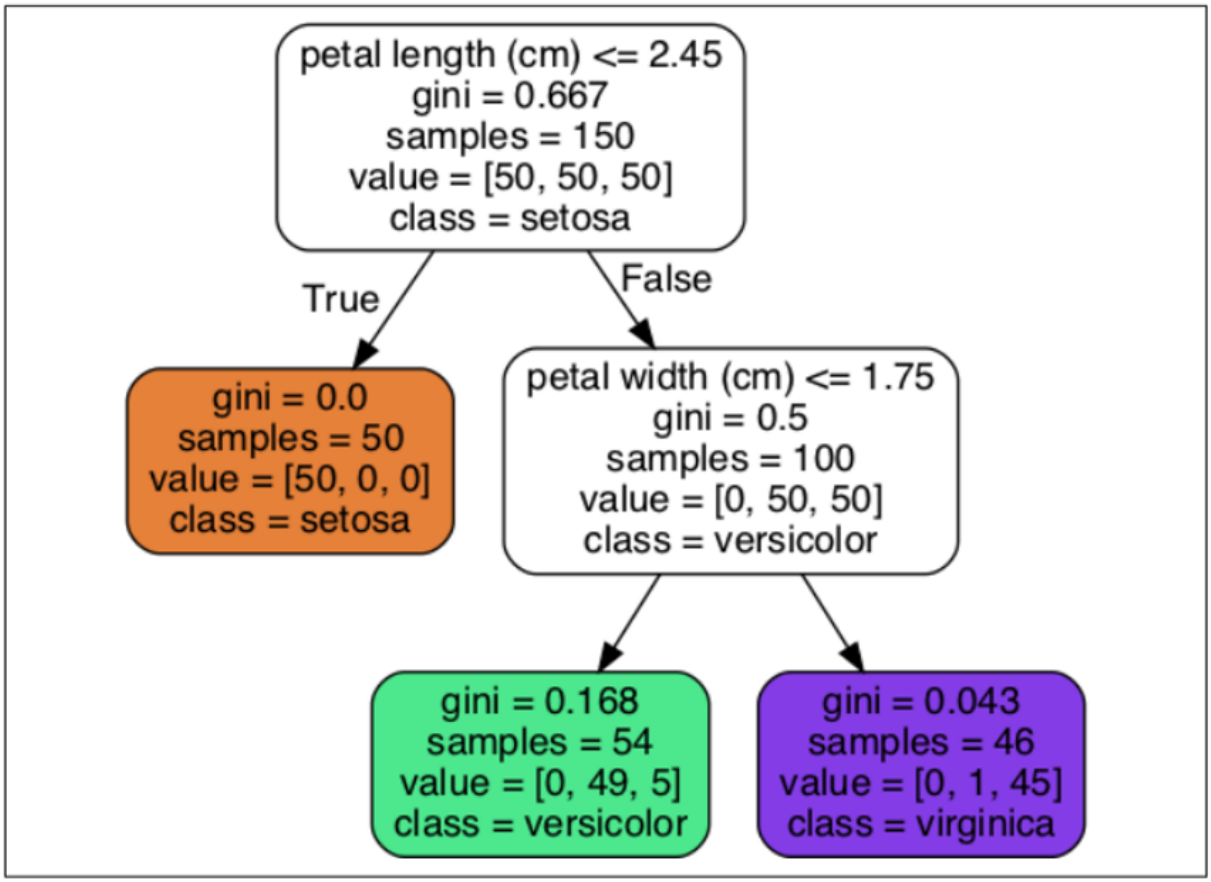
Além do processo de divisão recursiva, a geração de uma árvore de decisão envolve a escolha sucessiva dos melhores atributos e pontos de corte que maximizam a qualidade das partições geradas. Durante o treinamento, o algoritmo avalia diferentes possíveis divisões nos atributos disponíveis e seleciona aquela que produz o maior ganho de informação, a maior redução do índice de Gini ou a maior redução da variância/erro quadrático, dependendo da natureza do problema e do critério adotado. Esse procedimento é repetido de forma recursiva até que uma condição de parada seja satisfeita, como a profundidade máxima da árvore, o número mínimo de amostras em um nó ou a ausência de melhoria significativa na qualidade dos subconjuntos (Géron, 2019).

Em comparação com outras técnicas de aprendizado supervisionado, as árvores de decisão apresentam algumas características importantes. Uma de suas principais vantagens é a interpretabilidade, uma vez que a estrutura da árvore permite interpretar de forma transparente como as decisões são tomadas a partir dos atributos de entrada. Além disso, esses modelos requerem pouca preparação dos dados, pois não exigem normalização ou padronização das variáveis. Por outro lado, árvores de decisão individuais tendem a apresentar alta variância, sendo sensíveis a pequenas variações nos dados de treinamento. Essa característica pode levar ao fenômeno de *overfitting*, motivo pelo qual métodos baseados em conjuntos de árvores, como o

Random Forest, são frequentemente utilizados para melhorar a capacidade de generalização do modelo (Géron, 2019).

Um exemplo de AD pode ser visto na Figura 3.2.

Figura 3.2 – AD para Iris



Fonte: (Géron, 2019))

Na Figura 3.2, é possível observar uma árvore de decisão utilizada para classificar flores do conjunto de dados *Iris*. O nó raiz divide as amostras com base no comprimento da pétala (*petal length* $\leq$ 2.45), separando a classe *setosa* (pura, Gini = 0). As amostras restantes são subdivididas pelo critério de largura da pétala (*petal width* $\leq$ 1.75), onde as classes *versicolor* e *virginica* são diferenciadas com alta precisão (Gini próximo de 0 nos nós finais). A árvore otimiza a classificação minimizando o índice de Gini em cada divisão (Géron, 2019).

3.1.5 Random Forest (RF)

O RF consiste em uma combinação de AD, geralmente treinadas por meio do método *bagging* (Géron, 2019). Diferente das ADs, RF utiliza uma randomização extra no crescimento

das árvores. Ao invés de buscar a melhor *feature* para dividir o nó, utiliza-se da melhor dentro de um *subset* randômico. Nesse caso, busca-se reduzir a variância do modelo por meio da agregação de múltiplas árvores, mantendo um viés relativamente baixo quando comparado a modelos excessivamente regularizados.

A motivação para essa estratégia está relacionada à redução da variância do modelo. ADs individuais são modelos de alta variância, ou seja, pequenas variações no conjunto de treinamento podem resultar em estruturas de árvore significativamente diferentes. Ao treinar múltiplas árvores em subconjuntos distintos dos dados e introduzir aleatoriedade na seleção de atributos, o RF reduz a correlação entre os modelos individuais. Como consequência, ao combinar as previsões das árvores, os erros individuais tendem a se compensar, produzindo um modelo mais estável e com melhor capacidade de generalização quando comparado a uma única AD (Géron, 2019).

Além da seleção aleatória de subconjuntos de atributos em cada divisão, o RF também utiliza amostragem com reposição (*bootstrap sampling*) para gerar diferentes subconjuntos de dados de treinamento para cada árvore. Esse procedimento, conhecido como *bootstrap aggregating* ou *bagging*, contribui para aumentar a diversidade entre os modelos individuais da floresta (Géron, 2019).

Durante a fase de predição, as saídas das árvores são agregadas para produzir o resultado final do modelo. Em tarefas de classificação, essa agregação ocorre por meio de votação majoritária entre as árvores, enquanto em problemas de regressão o resultado é geralmente obtido pela média das previsões individuais (Géron, 2019). Essa estratégia reduz a variância do modelo e tende a melhorar a capacidade de generalização quando comparada ao uso de uma única AD.

3.1.6 Gradient Boosting (XGBoost)

O *Gradient Boosting* é uma técnica de aprendizado baseada na combinação sequencial de modelos fracos, geralmente árvores de decisão rasas, com o objetivo de minimizar uma função de perda diferenciável (Géron, 2019). Diferentemente do *bagging*, no qual os modelos são treinados de forma independente, o *boosting* constrói cada novo preditor de maneira iterativa, ajustando-o aos resíduos (erros) cometidos pelo modelo anterior. Esse processo pode ser interpretado como uma otimização por gradiente no espaço funcional, na qual cada novo modelo aproxima a direção de maior redução da função de custo.

Formalmente, a cada iteração m, busca-se adicionar um novo modelo $h_m(x)$ de forma que o modelo final seja definido conforme a Equação 3.2:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \tag{3.2}$$

em que γ_m representa o passo ótimo na direção do novo preditor.

O *XGBoost* (*Extreme Gradient Boosting*) constitui uma implementação otimizada do *Gradient Boosting*, incorporando técnicas de regularização, paralelização e controle eficiente de memória (Géron, 2019). Além disso, sua função objetivo inclui termos de penalização que auxiliam na prevenção de *overfitting*, tornando-o especialmente eficaz em problemas de alta dimensionalidade e grandes volumes de dados.

3.1.7 Multilayer Perceptron (MLP)

O *Multilayer Perceptron* (MLP) é uma função matemática parametrizada que aprende a melhor forma de transformar um vetor de entrada **x** em uma saída **y**. Assim, o MLP pode ser entendido como uma composição de várias funções aplicadas em sequência, formando um modelo de múltiplas camadas (Goodfellow; Bengio; Courville, 2016).

A arquitetura do MLP segue uma estrutura do tipo entrada $\rightarrow$ camadas ocultas $\rightarrow$ saída, utilizando uma função de ativação $g(\cdot)$ para introduzir não linearidade em cada camada. Essa relação pode ser expressa conforme a Equação 3.3:

$$h^{(l)} = g\left(W^{(l)} h^{(l-1)} + b^{(l)}\right) \tag{3.3}$$

em que $W^{(l)}$ e $b^{(l)}$ representam, respectivamente, a matriz de pesos e o vetor de vieses da camada l (Goodfellow; Bengio; Courville, 2016).

A introdução da função não linear $g(\cdot)$ é essencial, pois permite que o MLP aprenda relações complexas entre as variáveis de entrada e saída. Na ausência dessa não linearidade, o modelo se reduz a uma combinação linear das entradas, tornando-se incapaz de representar funções não lineares (Goodfellow; Bengio; Courville, 2016).

De forma geral, um MLP é composto por uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída, com conexões totalmente conectadas entre camadas consecutivas. Nesse tipo de rede, o fluxo de informação ocorre em apenas uma direção, da entrada para

a saída, caracterizando uma arquitetura do tipo *feedforward* (Géron, 2019). Cada neurônio de uma camada recebe como entrada uma combinação linear das ativações da camada anterior, adiciona um termo de viés e aplica uma função de ativação não linear, produzindo uma nova representação intermediária dos dados.

A presença de múltiplas camadas ocultas permite que o modelo aprenda representações internas cada vez mais abstratas dos dados de entrada. Esse processo de transformação sucessiva possibilita que redes neurais representem funções altamente complexas, o que torna os MLPs capazes de resolver problemas que modelos lineares não conseguem modelar adequadamente (Goodfellow; Bengio; Courville, 2016; Géron, 2019).

O treinamento de um MLP é realizado por meio do algoritmo de retropropagação do erro (*backpropagation*), que calcula o gradiente da função de perda em relação aos parâmetros da rede. Esse gradiente é utilizado por métodos de otimização baseados em descida do gradiente para atualizar iterativamente os pesos e vieses da rede, minimizando o erro entre as previsões do modelo e os valores observados (Goodfellow; Bengio; Courville, 2016). De acordo com (Géron, 2019), o processo de treinamento envolve a propagação direta dos dados pela rede (*forward pass*), seguida do cálculo do erro e da propagação desse erro no sentido inverso para ajustar os parâmetros do modelo.

Os MLPs podem ser utilizados tanto em tarefas de regressão quanto de classificação. Em problemas de regressão, a camada de saída geralmente possui um único neurônio com ativação linear, enquanto em tarefas de classificação podem ser utilizadas funções de ativação como a logística ou *softmax*, dependendo da natureza do problema (Géron, 2019). Essa flexibilidade faz com que o MLP seja uma das arquiteturas mais amplamente utilizadas no contexto de aprendizado profundo.

Além disso, o desempenho do modelo depende fortemente da escolha de hiperparâmetros como o número de camadas ocultas, o número de neurônios por camada, a função de ativação e a taxa de aprendizado. A definição adequada desses parâmetros é fundamental para garantir a capacidade de generalização do modelo e evitar problemas como *overfitting* ou *underfitting* (Géron, 2019).

3.2 Pesquisa Operacional (PO)

A Pesquisa Operacional (PO) constitui um campo multidisciplinar dedicado ao desenvolvimento e aplicação de modelos matemáticos, estatísticos e computacionais com o objetivo de apoiar processos de tomada de decisão. De forma geral, a PO busca representar problemas reais por meio de formulações matemáticas, permitindo a identificação de soluções que maximizem ou minimizem determinados objetivos sob um conjunto de restrições (Hillier; Lieberman, 2001).

Historicamente, técnicas de PO têm sido aplicadas em uma ampla variedade de contextos organizacionais, incluindo manufatura, transportes, construção civil, telecomunicações, planejamento financeiro, sistemas de saúde e gestão pública (Hillier; Lieberman, 2001). Em muitos desses cenários, os problemas de decisão envolvem a seleção de recursos limitados ou a alocação eficiente de elementos dentro de determinadas restrições estruturais.

No contexto esportivo, problemas como a montagem de elencos e o planejamento competitivo podem ser formalizados como problemas de otimização matemática sob restrições estruturais e orçamentárias. Em tais situações, busca-se selecionar ou alocar recursos — como jogadores — de forma a maximizar algum critério de desempenho coletivo, respeitando limitações impostas por fatores como orçamento, composição do elenco ou regras da competição.

Dependendo da natureza do problema, tais formulações podem recorrer a abordagens heurísticas voltadas à resolução de problemas de otimização combinatória, nos quais o espaço de soluções possíveis cresce exponencialmente com o número de variáveis de decisão. Nesses casos, métodos heurísticos tornam-se particularmente úteis para explorar o espaço de soluções e identificar configurações de boa qualidade em tempo computacional viável.

Entre os problemas clássicos da otimização combinatória destacam-se modelos que envolvem decisões de seleção sob restrições de capacidade ou orçamento. Um dos exemplos mais conhecidos dessa classe de problemas é o Problema da Mochila, apresentado a seguir.

3.2.1 Problema da Mochila

Um dos problemas clássicos da otimização combinatória é o Problema da Mochila (*Knapsack Problem*), amplamente utilizado como modelo abstrato para representar situações de seleção de recursos sob restrições limitadas (Cormen *et al.*, 2009). Nesse problema, considera-

se um conjunto de n itens, no qual cada item i possui um valor associado v_i e um peso w_i. O objetivo consiste em selecionar um subconjunto desses itens de modo a maximizar o valor total transportado, sem que o peso total ultrapasse a capacidade máxima W da mochila.

Na formulação conhecida como *0-1 Knapsack Problem*, cada item pode ser selecionado integralmente ou não selecionado, caracterizando uma decisão binária para cada elemento do conjunto. Dessa forma, o problema pode ser formulado matematicamente conforme as Equações 3.4–3.6:

$$\max \sum_{i=1}^{n} v_i x_i \tag{3.4}$$

sujeito a

$$\sum_{i=1}^{n} w_i x_i \leq W \tag{3.5}$$

$$x_i \in \{0,1\}, \quad i = 1,2,\ldots,n \tag{3.6}$$

em que x_i representa a decisão de selecionar ($x_i = 1$) ou não ($x_i = 0$) o item i.

De acordo com (Cormen *et al.*, 2009), o problema da mochila apresenta a propriedade de subestrutura ótima, característica comum em diversos problemas de otimização combinatória. No entanto, diferentemente da variante fracionária do problema, na qual é permitido selecionar frações de itens, a versão 0-1 não admite, em geral, soluções ótimas por meio de estratégias puramente gulosas. Nesse caso, a decisão de incluir ou excluir um determinado item depende da comparação entre diferentes combinações possíveis de itens, o que torna o problema computacionalmente mais desafiador.

Problemas de seleção sob restrições orçamentárias frequentemente apresentam estrutura semelhante à do Problema da Mochila. No contexto deste trabalho, a tarefa de definir quais jogadores devem compor um elenco sob um limite financeiro pode ser interpretada como um problema de seleção de elementos cujo custo total não deve exceder um orçamento disponível. De maneira análoga ao problema clássico, cada jogador pode ser associado a um custo de contratação e a um valor estimado de contribuição para o desempenho da equipe, avaliado por meio do modelo de AM desenvolvido neste estudo. Sob essa perspectiva, o problema de montagem de elencos pode ser interpretado como uma variação do Problema da Mochila, no qual se busca

selecionar um conjunto de jogadores capaz de maximizar o desempenho previsto do time sem ultrapassar o orçamento disponível.

Entretanto, diferentemente da formulação clássica do problema, a seleção de jogadores envolve também restrições estruturais adicionais. Em um elenco de futebol, por exemplo, o número total de jogadores é limitado e a composição da equipe deve respeitar determinadas necessidades posicionais e funcionais. Essas características aproximam o problema de variantes mais gerais do Problema da Mochila, nas quais múltiplas restrições ou categorias de seleção são consideradas.

Dessa forma, embora o problema abordado neste trabalho não corresponda exatamente à formulação clássica do Problema da Mochila, sua estrutura de decisão apresenta forte analogia com esse modelo. Essa relação fornece uma base conceitual importante para compreender a natureza combinatória da montagem de elencos e justifica a adoção de heurísticas de otimização para a obtenção de soluções de boa qualidade em tempo computacional viável.

A partir dessa fundamentação, apresentam-se a seguir dois métodos heurísticos amplamente utilizados em problemas de otimização combinatória: os algoritmos gulosos e a busca local.

3.2.2 Algoritmo Guloso

Em problemas de otimização, um algoritmo é dito guloso quando realiza uma escolha localmente ótima, na esperança de que essa decisão conduza a uma solução globalmente ótima (Cormen *et al.*, 2009). Diferentemente de abordagens exaustivas, algoritmos gulosos tomam decisões irrevogáveis a cada etapa, baseando-se apenas na informação disponível no momento da escolha. É importante destacar que, conforme ressaltado por (Cormen *et al.*, 2009), algoritmos gulosos não garantem, em geral, a obtenção de soluções globalmente ótimas, sendo necessários argumentos específicos de corretude para justificar quando a estratégia gulosa é válida.

Conforme discutido por (Cormen *et al.*, 2009), algoritmos gulosos podem ser formulados de maneira iterativa ou recursiva. Na forma recursiva, o método costuma seguir um padrão *top-down*: realiza-se uma escolha gulosa, reduz-se o problema a um subproblema estritamente menor e repete-se o processo até a construção completa de uma solução. A justificativa de corretude, quando possível, normalmente se apoia em dois ingredientes: (i) a *propriedade da escolha gulosa*, isto é, a existência de uma solução ótima que contém a decisão local tomada; e (ii) a

subestrutura ótima, segundo a qual, após fixar a decisão gulosa, o restante do problema preserva a mesma estrutura do original, de modo que uma solução ótima do subproblema completa uma solução ótima do problema inicial.

Para ilustrar o funcionamento dessa estratégia, considere o problema do *Traveling Salesman Problem* (TSP), no qual se deseja encontrar um circuito Hamiltoniano de custo mínimo que visite exatamente uma vez cada cidade de um conjunto dado e retorne ao ponto inicial. Nesse contexto, uma heurística gulosa construtiva bastante conhecida é o método do *vizinho mais próximo* (*nearest neighbor*). O algoritmo inicia em uma cidade arbitrária e, a cada etapa, seleciona a cidade ainda não visitada que possui a menor distância em relação à cidade atual, repetindo o processo até que todas as cidades tenham sido visitadas e o ciclo seja fechado retornando à cidade inicial (Gendreau; Potvin, 2010).

Esse procedimento exemplifica claramente a lógica gulosa: em cada passo é realizada a escolha localmente mais vantajosa — visitar a cidade mais próxima — sem reconsiderar decisões anteriores. Embora tal estratégia permita construir rapidamente uma solução viável para o problema, não há garantia de que o circuito obtido seja globalmente ótimo, uma vez que decisões tomadas nas primeiras etapas podem restringir significativamente as escolhas disponíveis nas etapas subsequentes (Gendreau; Potvin, 2010).

O Algoritmo 1 apresenta o pseudocódigo dessa heurística gulosa aplicada ao TSP.

Algoritmo 1 Heurística Gulosa para o TSP (Vizinho Mais Próximo)

Require: Conjunto de cidades V, matriz de distâncias $d(i, j)$
Ensure: Ciclo Hamiltoniano T
  Escolher cidade inicial v_0
  Inicializar T com v_0
  Inicializar conjunto de cidades não visitadas $U \leftarrow V \setminus \{v_0\}$
  Definir cidade atual $v \leftarrow v_0$
  while existirem cidades não visitadas **do**
    Escolher $u \in U$ que minimize $d(v, u)$
    Adicionar u ao final de T
    Remover u de U
    Atualizar cidade atual $v \leftarrow u$
  end while
  Adicionar v_0 ao final de T ▷ retorno à cidade inicial
  return T

No contexto deste trabalho, um algoritmo guloso é utilizado de forma recursiva com o objetivo de gerar uma solução inicial para o problema de otimização da montagem de elencos. Dado um time e um orçamento, a cada iteração, o jogador com menor minutagem do elenco é

removido e substituído por candidatos externos. Para cada substituição possível, o modelo de AM é empregado como função objetivo, estimando a pontuação esperada do time resultante. A escolha gulosa consiste, portanto, em selecionar o(s) jogador(es) que cabem no orçamento do time e cuja inclusão maximiza a pontuação prevista, mantendo-se fixas as demais variáveis do elenco. Assim como no exemplo discutido anteriormente, a solução é construída incrementalmente por decisões locais irrevogáveis, produzindo rapidamente uma solução factível que servirá como ponto de partida para os procedimentos de refinamento realizados por busca local.

3.2.3 Algoritmo de Busca Local

Enquanto algoritmos gulosos constroem soluções de forma incremental a partir de decisões locais, métodos de busca local partem de uma solução inicial já factível e buscam refiná-la por meio da exploração iterativa de sua vizinhança. A busca local é uma técnica heurística amplamente utilizada em problemas de otimização combinatória, cujo objetivo é aprimorar uma solução inicial factível por meio da exploração iterativa de soluções vizinhas (Gendreau; Potvin, 2010). Diferentemente de métodos exatos, a busca local não realiza uma exploração global do espaço de soluções, concentrando-se na melhoria incremental da solução corrente a partir de movimentos locais.

Conforme descrito por (Gendreau; Potvin, 2010), um algoritmo de busca local é composto essencialmente por quatro elementos: uma solução inicial, a definição de uma vizinhança, um critério de aceitação de movimentos e uma condição de parada. A vizinhança determina o conjunto de soluções que podem ser alcançadas a partir da solução atual por meio de pequenas modificações estruturais, enquanto o critério de aceitação estabelece se uma solução vizinha deve ou não substituir a solução corrente. O processo é repetido iterativamente até que nenhuma melhoria adicional seja encontrada ou que um critério de parada seja satisfeito.

Um exemplo clássico da aplicação de busca local ocorre no Problema do Caixeiro Viajante (*Traveling Salesman Problem* – TSP). Croes (Croes, 1958) propôs um método de melhoria que parte de uma solução inicial e aplica transformações denominadas *inversions*, nas quais a ordem de uma subsequência de cidades do tour é invertida sempre que essa operação reduz o comprimento total do percurso. Essas transformações geram soluções vizinhas a partir da solução corrente e são aceitas apenas quando produzem melhoria na função objetivo. Esse

procedimento caracteriza uma estratégia típica de busca local, na qual soluções vizinhas são exploradas iterativamente até que nenhuma melhoria adicional possa ser obtida.

No contexto do TSP, uma operação de inversão consiste em selecionar duas posições do tour e inverter a ordem das cidades contidas no segmento compreendido entre essas posições. Essa modificação gera uma nova rota candidata que pode apresentar menor comprimento total quando comparada à solução original. Caso a nova solução produza melhoria na função objetivo, ela passa a substituir a solução corrente, e o processo de exploração da vizinhança continua até que nenhuma inversão adicional resulte em redução do custo do percurso.

Um pseudocódigo simplificado desse procedimento de busca local baseado em inversões é apresentado no Algoritmo 2.

Algoritmo 2 Busca local baseada em inversões para o TSP

Require: tour inicial T
Ensure: tour melhorado T^*
melhoria $\leftarrow$ verdadeiro
while melhoria **do**
    melhoria $\leftarrow$ falso
    for $i = 1$ até $n-1$ **do**
        for $j = i+1$ até n **do**
            $T' \leftarrow$ tour obtido invertendo o segmento entre i e j
            if $dist(T') < dist(T)$ **then**
                $T \leftarrow T'$
                melhoria $\leftarrow$ verdadeiro
            end if
        end for
    end for
end while
return T

No problema abordado neste trabalho, a busca local é empregada para avaliar diferentes conjuntos de jogadores candidatos a contratação, respeitando uma restrição de orçamento. Cada solução é representada por um conjunto de jogadores de cardinalidade fixa, sendo a vizinhança definida por operações de substituição unitária (*swap neighborhood*). Em cada iteração, jogadores do conjunto atual são considerados candidatos à remoção e substituídos por jogadores externos ainda não selecionados, produzindo soluções vizinhas que preservam o tamanho do elenco. Cada solução vizinha gerada é avaliada por meio de uma função objetivo implementada a partir do modelo de AM, responsável por estimar a pontuação esperada do time na temporada considerada.

A restrição orçamentária é tratada como condição de factibilidade: soluções que excedem o orçamento estabelecido são descartadas durante o processo de busca. O critério de aceitação adotado é de melhoria estrita, isto é, uma solução vizinha é aceita apenas se apresentar valor superior da função objetivo em relação à solução corrente. O algoritmo é encerrado quando nenhuma substituição viável resulta em melhoria adicional, caracterizando a convergência para um ótimo local, conforme descrito em (Gendreau; Potvin, 2010).

4 REVISÃO DE LITERATURA

Após a apresentação dos principais fundamentos teóricos relacionados ao AM e aos métodos de otimização, torna-se importante analisar como essas abordagens têm sido aplicadas na literatura científica. Assim, este capítulo apresenta uma revisão de estudos que utilizam técnicas de AM e otimização no contexto esportivo, com ênfase em aplicações relacionadas ao futebol.

4.1 Aprendizado de Máquina (AM) no contexto esportivo

Em 2020, *Horvat & Job* apresentaram uma revisão abrangente sobre o uso de AM na predição de resultados esportivos, com foco principal em esportes coletivos, especialmente o futebol. Os autores destacam que o AM é particularmente adequado a essa tarefa, uma vez que modalidades esportivas geram grandes volumes de dados estruturados e não estruturados, provenientes de eventos de jogo, sistemas de *tracking*, probabilidades associadas às partidas e bases históricas consolidadas (Horvat; Job, 2020).

Conforme apontado por *Horvat & Job* (2020), a maior parte dos estudos que empregam métodos preditivos no contexto esportivo concentra-se na predição de resultados de partidas individuais, especialmente na classificação de vitórias, empates e derrotas. Essa predominância pode ser explicada tanto pelo interesse mercadológico associado às apostas esportivas, que incentiva o desenvolvimento de modelos voltados à antecipação de resultados (Abidin, 2021), quanto pela própria natureza técnica do problema, que se ajusta adequadamente aos algoritmos supervisionados mais difundidos e à disponibilidade de bases históricas amplamente registradas no futebol (Horvat; Job, 2020). Nesse contexto, observa-se menor atenção a problemas relacionados ao desempenho agregado ao longo de uma temporada, especialmente quando este é modelado de forma prospectiva como função da composição individual do elenco.

Em contraste com essa predominância de estudos voltados à predição de partidas isoladas, a revisão conduzida por (Abidin, 2021) identifica, ainda que em menor número, trabalhos que abordam diretamente o problema de seleção e formação de equipes sob uma perspectiva multicritério e de otimização. Entre os estudos mencionados, destacam-se (Tavana *et al.*, 2013), que propõem um sistema de inferência fuzzy em duas fases para seleção de jogadores e posterior arranjo tático da equipe, e (Qader *et al.*, 2017), que estruturam o problema de escolha

de atletas como uma análise multicritério baseada em múltiplas métricas de desempenho. Tais abordagens representam esforços relevantes no sentido de formalizar o processo de composição de elencos, ainda que se apoiem predominantemente em modelos fuzzy ou técnicas de agregação determinística, sem incorporar mecanismos de aprendizado supervisionado voltados à predição prospectiva de desempenho coletivo.

Os estudos voltados à predição de resultados, por sua vez, identificam o uso recorrente de algoritmos como regressão logística, *Support Vector Machines* (SVM), *Random Forest* e redes neurais, destacando que modelos que integram múltiplas fontes de informação — incluindo estatísticas tradicionais, indicadores derivados e variáveis contextuais — tendem a apresentar melhor desempenho preditivo (Horvat; Job, 2020). Ainda assim, a tarefa permanece desafiadora devido à natureza estocástica do futebol e à ausência de padronização entre bases de dados utilizadas em diferentes estudos. Cabe ressaltar que, nos últimos anos, a geração de dados aumentou significativamente com a maior disponibilidade de dados esportivos em larga escala, impulsionada pela expansão de sistemas de *tracking* e por bancos estruturados como *Opta* e *StatsBomb* (Horvat; Job, 2020).

4.1.1 Previsão de resultados de partidas

Conforme ressaltado anteriormente, os modelos de previsão de resultados de partidas constituem uma das aplicações mais tradicionais do AM no futebol, com foco na classificação de vitórias, empates e derrotas. A seguir, apresentam-se alguns estudos representativos dessa linha de pesquisa, com o objetivo de evidenciar sua evolução metodológica.

Um dos trabalhos pioneiros é o de *Joseph, Fenton & Neil* (2006) (Joseph; Fenton; Neil, 2006), que empregaram redes Bayesianas para prever resultados a partir de um conjunto reduzido de dados do *Tottenham Hotspur Football Club* (1995–1997). A base analisada compreendeu 76 partidas e 30 atributos, sendo 28 jogadores representados por variáveis binárias (presença ou ausência), além do local da partida e da qualidade do adversário. Os próprios autores reconhecem tratar-se de um cenário com disponibilidade limitada de dados, o que impõe desafios adicionais aos métodos de aprendizado supervisionado. Ainda assim, os resultados indicaram que a rede Bayesiana construída com conhecimento especializado apresentou melhor desempenho relativo nos experimentos realizados, mesmo diante da limitação amostral. Esse resultado reforça que, em contextos com bases reduzidas e forte especificidade estrutural, a mo-

delagem explícita das relações entre variáveis pode mitigar parcialmente os efeitos da escassez de dados. Entretanto, o estudo permanece circunscrito à predição de partidas individuais e a uma equipe específica.

Em contexto mais recente, *Baboota & Kaur* (2019) (Baboota; Kaur, 2019) investigaram a predição multiclasse de resultados na Premier League utilizando dados de duas temporadas (2014–2015 e 2015–2016). O estudo incorporou variáveis relacionadas ao desempenho recente das equipes, estatísticas técnicas de jogo (finalizações, posse de bola, escanteios), *ratings* derivados do FIFA Index e fator casa/fora. Foram avaliados algoritmos como *Random Forest*, SVM, Naive Bayes e XGBoost, sendo o *Gradient Boosting* o modelo de melhor desempenho. Ainda assim, os autores observam que os modelos não superaram as previsões das casas de apostas, que incorporam variáveis não observáveis pelos modelos estatísticos. Embora represente uma ampliação substancial em relação a trabalhos anteriores — tanto em volume de dados quanto na diversidade de algoritmos avaliados — o problema permanece centrado na predição do desfecho de partidas individuais.

No âmbito do Aprendizado Profundo, *Awadallah & Khandelwal* (2020) (Awadallah; Khandelwal, 2020) compararam regressão logística, *Random Forest*, redes neurais densas (DNN) e LSTM na predição de resultados da PL ao longo de dez temporadas (2008–2019), utilizando aproximadamente 4.180 partidas e um conjunto reduzido de 24 *features*. O modelo LSTM apresentou o melhor desempenho, alcançando cerca de 58% de acurácia em teste. Apesar do avanço metodológico representado pelo uso de redes neurais profundas, os autores destacam limitações relacionadas à generalização e à natureza estocástica do futebol.

De modo geral, observa-se uma evolução metodológica significativa, desde bases reduzidas com forte dependência de conhecimento especializado até modelos supervisionados e redes neurais aplicados a bases mais amplas. Ainda assim, a unidade analítica predominante permanece sendo a partida individual, com ênfase na classificação do resultado do jogo.

4.1.2 Aplicações recentes do Aprendizado de Máquina (AM) no futebol

Para além da predição de partidas isoladas, a literatura recente evidencia uma ampliação do escopo de aplicação do AM no futebol. *Rico-González et al.* (2023) demonstram que o AM tem sido empregado em diversas frentes, como análise de desempenho físico, detecção de padrões táticos, avaliação de risco de lesões e modelagem espacial com dados de *tracking*

(Rico-González *et al.*, 2023). A crescente disponibilidade de dados tem impulsionado abordagens metodológicas cada vez mais complexas, incluindo redes neurais profundas, modelos probabilísticos e métodos baseados em séries temporais.

Além das aplicações voltadas à predição de partidas, estudos recentes evidenciam a incorporação crescente de dados espaciais e fisiológicos na modelagem do desempenho esportivo. O volume *Machine Learning and Data Mining for Sports Analytics* (MLSA, 2023) destaca a integração entre dados de eventos e informações de *tracking*, ressaltando que modelos exclusivamente baseados em eventos enfrentam limitações estruturais pela ausência de contexto espacial e pressão defensiva (Brefeld *et al.*, 2024). De forma complementar, pesquisas como a de Majumdar et al. (2022) exploram a aplicação de AM na predição de lesões a partir de variáveis relacionadas à carga de treino e competição, utilizando desde regressão logística até redes neurais artificiais. Essas abordagens evidenciam a ampliação temática e metodológica do campo, mas permanecem majoritariamente voltadas à modelagem de situações específicas de jogo ou risco individual, não se direcionando à modelagem prospectiva do desempenho agregado de temporada como função da composição do elenco.

No contexto nacional, observa-se também o crescimento da área de *sports analytics*, com iniciativas acadêmicas como o Sports Analytics Lab (SALab) da UFMG, voltadas à aplicação de métodos quantitativos no futebol.

Apesar dessa expansão metodológica e temática, observa-se que a maior parte dos estudos ainda se concentra na modelagem de eventos específicos ou na predição de partidas isoladas, conforme destacado por *Horvat & Job* (2020). Em contraste, verifica-se menor atenção a abordagens voltadas à modelagem do desempenho agregado das equipes ao longo de uma temporada, especialmente quando esse desempenho é tratado como função da composição individual do elenco. Essa constatação reforça a necessidade de examinar de forma mais sistemática como a literatura tem abordado o desempenho agregado sob perspectivas estatísticas e econômicas.

4.1.3 Desempenho agregado e determinantes de temporada

Além da predição de partidas individuais, parte da literatura tem investigado determinantes do desempenho agregado ao longo de uma temporada. Lago-Ballesteros e Lago-Peñas (2010), ao analisarem todas as 380 partidas da temporada 2008–2009 da La Liga espanhola, identificaram diferenças estatisticamente significativas entre equipes do topo e da parte inferior

da tabela em variáveis como gols marcados, total de finalizações, finalizações no alvo, assistências e posse de bola (Lago-Ballesteros; Lago-Peñas, 2010). Os resultados indicam que o sucesso competitivo agregado pode ser parcialmente explicado por padrões estatísticos observáveis ao longo da temporada.

Sob a ótica econômica, estudos como os de Carmichael, McHale e Thomas (2011) analisam o desempenho agregado das equipes a partir de variáveis estruturais, como gastos salariais e receitas, evidenciando associação entre investimento financeiro e posição final na liga (Carmichael; McHale; Thomas, 2011). De forma complementar, Totty e Owens (2011) discutem como mecanismos institucionais, como tetos salariais, influenciam o equilíbrio competitivo das ligas profissionais (Totty; Owens, 2011). Essas abordagens indicam que o desempenho ao longo de uma temporada pode ser parcialmente explicado por fatores estruturais e de alocação de recursos, reforçando a ideia de que o sucesso competitivo agregado não é puramente aleatório.

Herold, Memmert e Perl (2019) investigaram a aplicação de AM no futebol profissional masculino, utilizando variáveis como passes, finalizações, dribles e métricas derivadas, como Gols Esperados (xG). O estudo demonstra que padrões técnicos individuais podem ser representados matematicamente e modelados por meio de algoritmos supervisionados e não supervisionados, incluindo redes neurais, árvores de decisão e regressão logística (Herold; Memmert; Perl, 2019). Tais evidências reforçam que estatísticas de desempenho individual constituem insumos mensuráveis e potencialmente preditivos do sucesso competitivo.

Complementarmente, Kite e Nevill (2017) investigaram preditores de sucesso ao longo de temporadas consecutivas de uma mesma equipe, demonstrando que determinados indicadores de desempenho apresentam capacidade explicativa sobre a posição final e os resultados obtidos (Kite; Nevill, 2017). Tais resultados evidenciam que métricas de jogo podem ser utilizadas para modelar o sucesso competitivo.

Embora o estudo de *Kite e Nevill* (2017) represente um avanço ao empregar modelos estatísticos com finalidade preditiva, sua modelagem permanece centrada no nível da partida individual. O sucesso é operacionalizado a partir do desfecho dos jogos (vitória, empate ou derrota), e as variáveis explicativas derivam predominantemente de indicadores agregados por partida (Kite; Nevill, 2017). Dessa forma, ainda que tais contribuições ampliem a compreensão dos determinantes estatísticos do desempenho competitivo, não há uma modelagem explícita do desempenho agregado da temporada como função da composição estrutural do elenco. Essa lacuna torna-se particularmente relevante quando se considera que decisões estratégicas no fu-

tebol profissional — especialmente no mercado de transferências — são tomadas com base na expectativa de desempenho ao longo de toda a competição, e não apenas em resultados isolados.

De modo semelhante, embora essas contribuições identifiquem determinantes técnicos e econômicos do sucesso competitivo, elas concentram-se majoritariamente na análise de resultados já observados ou em relações estruturais agregadas. Não modelam explicitamente, de forma prospectiva, a pontuação esperada de uma equipe como função direta da composição individual do elenco, tampouco integram tais estimativas a problemas formais de decisão.

4.2 Pesquisa Operacional (PO) no contexto esportivo

A Pesquisa Operacional (PO) constitui um conjunto de métodos analíticos voltados à modelagem e otimização de processos decisórios (Medri; Yotsumoto, 2009; Wright, 2009). No contexto esportivo, sua aplicação abrange desde problemas estratégicos, como o gerenciamento de equipes, até decisões táticas e logísticas (Gerchak, 1994).

Técnicas como Programação Linear, Programação Dinâmica e Processos de Markov têm sido empregadas na análise de substituições, estratégias de jogo e planejamento competitivo (Clarke; Norman, 1998; Hirotsu; Wright, 2002). Em muitos casos, tais problemas apresentam natureza combinatória e elevada complexidade computacional.

Conforme discutido por (Kendall; Parkes; Spoerer, 2010), diversos problemas de organização competitiva e escalonamento esportivo (*sports scheduling*) apresentam elevada complexidade combinatória, sendo muitos deles classificados como NP-difíceis ou NP-completos em suas versões formais. Entre os exemplos destacados estão o Traveling Tournament Problem (TTP), que busca minimizar as distâncias percorridas pelas equipes sob restrições estruturais de calendário; o problema de minimização de *breaks* em torneios round-robin, cuja versão com calendário parcial é NP-difícil; e variantes do problema de alocação de árbitros, cuja versão de decisão é NP-completa. Diante dessa complexidade, especialmente em instâncias de grande porte e com múltiplas restrições, a literatura recorre amplamente a heurísticas e metaheurísticas, como busca local, *simulated annealing*, *tabu search* e métodos híbridos, visando à obtenção de soluções de alta qualidade em tempo computacional viável.

Nesse contexto, destaca-se o trabalho de Ribeiro e Urrutia (Ribeiro; Urrutia, 2007), que aborda a variante *mirrored traveling tournament problem* (mTTP), uma extensão do TTP em que o calendário da segunda metade da competição replica a primeira com inversão de mandos

de campo. Os autores propõem heurísticas construtivas e metaheurísticas híbridas baseadas em GRASP e *iterated local search*, demonstrando a eficácia dessas abordagens na obtenção de soluções de alta qualidade para problemas de grande porte. Esse trabalho evidencia tanto a complexidade inerente ao escalonamento esportivo quanto a relevância de métodos heurísticos sofisticados para sua resolução.

Além da dificuldade computacional inerente a esses problemas, observa-se também uma limitação recorrente na modelagem adotada. Na maior parte dos trabalhos, a função objetivo é definida por métricas simplificadas ou parâmetros fixos, não incorporando modelos preditivos baseados no desempenho individual de atletas como componente central do problema decisório. Assim, permanece inexplorada a integração entre modelos preditivos de desempenho agregado e técnicas de otimização aplicadas à montagem de elenco.

Problemas de montagem de elenco também têm sido abordados sob a ótica da otimização matemática. Pantuso (Pantuso, 2017) propõe o *Football Team Composition Problem* por meio de um modelo de programação estocástica multietapa, no qual a incerteza sobre o desenvolvimento futuro dos atletas é incorporada explicitamente à formulação. De forma complementar, Boon e Sierksma (Boon; Sierksma, 2003) modelam a formação de equipes como um problema de alocação ótima jogador–posição, formulado via programação linear inteira e equivalente a um problema de *matching* máximo ponderado. Já Becker e Sun (Becker; Sun, 2015) aplicam uma estrutura análoga ao problema da mochila para otimizar decisões de seleção em *fantasy football* sob restrições orçamentárias.

Embora esses trabalhos avancem na formalização matemática da montagem de equipes, observa-se que a função objetivo permanece baseada em avaliações previamente definidas, pontuações atribuídas a priori ou estimativas paramétricas fixadas antes do processo de otimização. Não há, nesses estudos, a incorporação explícita de modelos preditivos de desempenho agregado aprendidos diretamente a partir de dados históricos como componente central da função objetivo. Dessa forma, permanece uma lacuna na literatura quanto à integração estruturada entre modelos de AM e problemas combinatórios de montagem de elenco.

4.2.1 Otimização Prescritiva e Problemas Combinatórios no Esporte

A transição de *predictive analytics* para *prescriptive analytics*, conforme discutido por (Bertsimas; Kallus, 2019), consiste na integração entre modelos preditivos e problemas de oti-

mização, permitindo que estimativas estatísticas sejam incorporadas diretamente na definição de políticas ótimas de decisão. Nesse paradigma, modelos de AM deixam de atuar apenas como ferramentas de previsão e passam a compor a função objetivo de problemas decisórios.

Além das abordagens clássicas de otimização, a literatura recente evidencia uma aproximação crescente entre PO e métodos de Inteligência Artificial (IA). Conforme discutido por (Chmait; Westerbeek, 2021), técnicas como aprendizado por reforço e algoritmos evolutivos vêm sendo empregadas em problemas decisórios no esporte, aproximando modelos de aprendizado e processos de otimização. Esse movimento reforça a convergência entre predição estatística e decisão prescritiva, abrindo espaço para estruturas integradas nas quais modelos de AM podem ser incorporados diretamente à função objetivo de problemas combinatórios.

No contexto esportivo, decisões estratégicas como montagem de elenco e planejamento competitivo podem ser formalizadas como problemas de otimização sob restrições orçamentárias e estruturais. Tais problemas frequentemente assumem estrutura combinatória análoga àquela observada em problemas clássicos de escalonamento esportivo (Kendall; Parkes; Spoerer, 2010; Kendall *et al.*, 2010), os quais envolvem múltiplos agentes, restrições e objetivos, sendo tradicionalmente abordados por meio de técnicas como programação inteira, programação por restrições e metaheurísticas. Tal complexidade justifica o uso de heurísticas e metaheurísticas para obtenção de soluções viáveis.

No futebol, essa formulação tem sido investigada sob diferentes perspectivas, sendo frequentemente referida como *Football Team Composition Problem* (FTCP), que consiste em determinar a melhor composição de elenco considerando simultaneamente restrições financeiras, regulatórias e táticas. Trabalhos recentes, como o de (Dantas; Ritt, 2025), propõem abordagens baseadas em programação inteira para maximizar métricas de habilidade dos jogadores, incorporando requisitos táticos e estruturais na modelagem do problema. Embora Dantas (2025) proponha um objetivo mais alinhado ao desempenho esportivo, sua abordagem ainda se fundamenta predominantemente em métricas estáticas de desempenho individual, sem modelar explicitamente o impacto prospectivo da composição do elenco sobre o desempenho competitivo coletivo da equipe. Nesse sentido, a metodologia aqui apresentada diferencia-se ao incorporar modelos preditivos na definição da função objetivo, permitindo capturar de forma mais direta os efeitos futuros das decisões de montagem de elenco.

Nesse contexto, o presente estudo insere-se nessa interseção entre AM e PO ao propor um modelo capaz de estimar a pontuação esperada de uma equipe em uma temporada a partir

do desempenho individual de seus jogadores na temporada anterior. Diferentemente das abordagens focadas na previsão de partidas isoladas ou na análise retrospectiva de determinantes agregados, o modelo desenvolvido busca capturar prospectivamente o impacto da composição do elenco sobre o desempenho competitivo. Posteriormente, essa estimativa é incorporada como função objetivo em um problema combinatório de montagem de elenco sob restrição orçamentária, aproximando predição estatística e decisão prescritiva no contexto do futebol profissional.

5 MODELAGEM PREDITIVA POR APRENDIZADO DE MÁQUINA (AM)

Este capítulo apresenta a metodologia de modelagem preditiva baseada em AM utilizada neste trabalho. Inicialmente, descreve-se o processo de coleta e organização dos dados utilizados no estudo. Em seguida, são apresentados os procedimentos de pré-processamento e preparação das variáveis, bem como os modelos de AM empregados para estimar o desempenho das equipes a partir das estatísticas individuais dos jogadores.

5.1 Coleta dos Dados

Com o intuito de coletar os dados, foi desenvolvido um algoritmo de *web scraping*, utilizando a biblioteca *Beautiful Soup* da linguagem *Python*. Essa biblioteca é amplamente empregada para extração e manipulação de informações a partir de páginas HTML e XML, permitindo a coleta automatizada e estruturada de grandes volumes de dados provenientes da web (Beautiful Soup Documentation, 2024). O algoritmo implementado automatiza o acesso às páginas de estatísticas, realiza a extração das tabelas correspondentes a cada clube e temporada e armazena os dados de forma padronizada para posterior análise.

O site utilizado para a extração foi o FBref, uma ampla base de dados de estatísticas de futebol que disponibiliza dados detalhados de jogadores e equipes de diversas competições ao redor do mundo. O FBref se destaca por fornecer estatísticas avançadas, como gols esperados (xG), ações de pressão, passes progressivos e estatísticas defensivas, tornando-se uma fonte relevante para análises preditivas e modelagem estatística no contexto do futebol.

Com o objetivo de utilizar um volume de dados superior ao empregado por *Baboota & Kaur* (2018) e *Joseph, Fenton & Neil* (2006), bem como reduzir o risco de *overfitting*, foram coletados os dados de todos os jogadores da *Serie A* italiana e da *Premier League* que atuaram ao longo de cinco temporadas (2019-2020, 2020-2021, 2021-2022, 2022-2023 e 2023-2024), além das tabelas de resultados de todos os jogos disputados nesse mesmo período.

A utilização de duas ligas distintas neste estudo possui o papel metodológico deevitar a ocorrência de *data leakage* entre os conjuntos de treinamento, teste e validação dos modelos de AM. Para isso, os dados de uma liga são utilizados exclusivamente na etapa de treinamento, enquanto os dados da outra liga são reservados para teste e validação. Essa estratégia assegura que o modelo seja avaliado em um contexto completamente distinto daquele empregado no apren-

dizado, eliminando dependências estruturais entre amostras e proporcionando uma avaliação mais realista de sua capacidade de generalização.

Os dados obtidos foram armazenados no formato CSV, organizados em arquivos distintos para cada clube e temporada. Dessa forma, cada arquivo contém as estatísticas individuais dos jogadores de um determinado time em uma temporada específica, facilitando a organização, o pré-processamento e as etapas subsequentes da análise. Ao todo, foram coletados dados de 2900 jogadores pertencentes a 30 equipes da Serie A italiana e de 2399 jogadores pertencentes a 28 equipes da PL. Assim, o conjunto de dados utilizado neste estudo totaliza 5299 registros de jogadores, distribuídos entre 58 equipes, que serviram de base para o desenvolvimento dos modelos preditivos.

5.2 Pré-processamento dos Dados

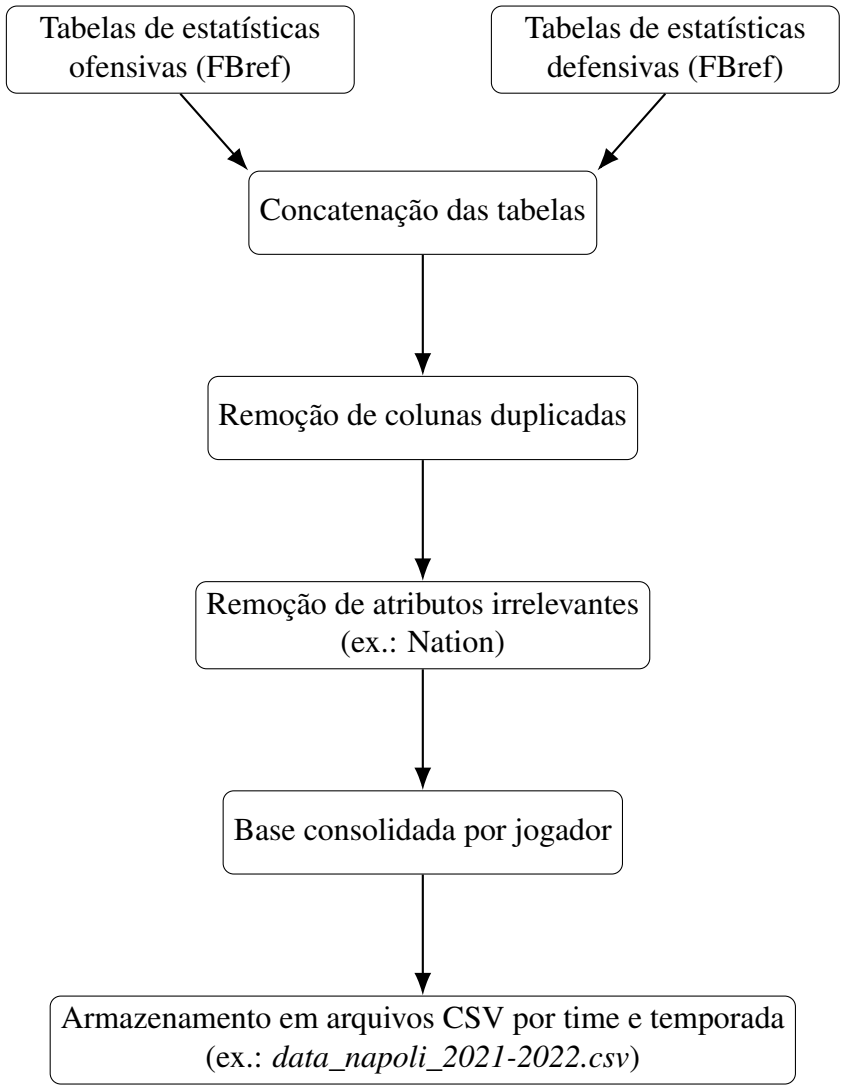
5.2.1 Base de Dados dos Jogadores

As estatísticas dos jogadores foram extraídas do site FBref (FBref, 2026) para todos os times e temporadas consideradas no estudo. No entanto, o FBref disponibiliza separadamente as estatísticas ofensivas e defensivas dos jogadores em tabelas distintas. Dessa forma, foi necessário realizar uma etapa de organização e integração dessas informações para construir uma base única de atributos por jogador.

Inicialmente, foram removidos atributos que não possuem influência direta no desempenho esportivo, como variáveis meramente descritivas (por exemplo, nacionalidade). Em seguida, as tabelas contendo estatísticas ofensivas e defensivas foram concatenadas, permitindo que cada jogador passasse a possuir, em um único registro, todas as métricas de desempenho consideradas no estudo. Durante esse processo, eventuais colunas duplicadas geradas pela concatenação foram identificadas e removidas. Por fim, os dados consolidados foram armazenados em arquivos CSV individuais para cada clube e temporada, seguindo o padrão *data_time_temporada.csv* (por exemplo, *data_napoli_2021-2022.csv*). A Figura 5.1 apresenta o fluxo de organização dos dados a partir das tabelas originais do FBref.

As tabelas resultantes desse processo possuem 47 atributos relacionados à idade, posição, minutos jogados, bem como estatísticas ofensivas (gols, assistências, passes decisivos, entre outras) e defensivas (bloqueios, desarmes, erros graves, entre outras) de cada jogador.

Figura 5.1 – Fluxo de organização das tabelas ofensivas e defensivas extraídas do FBref para a construção da base final de atributos dos jogadores.



Fonte: Elaborado pelo autor.

Esses dados foram carregados em diferentes *dataframes*, um para cada liga considerada, e posteriormente organizados por jogador, incluindo informações de clube, temporada e respectivos indicadores de desempenho. Ressalta-se que um mesmo atleta pode aparecer em diferentes temporadas e clubes dentro da organização adotada.

A Tabela 5.1 apresenta a descrição completa das variáveis utilizadas no estudo.

Tabela 5.1 – Descrição das variáveis utilizadas no modelo

Variável	**Descrição**
Identificação e contexto	
Player	Nome do jogador.

Variável	Descrição
Season	Temporada à qual os dados se referem.
Team	Clube pelo qual o jogador atuou na temporada.
Pos.	Posição principal do jogador em campo.
Idade	Idade do jogador durante a temporada.
Participação e minutagem	
MP	Número de partidas disputadas.
Inícios	Número de partidas iniciadas como titular.
Min.	Total de minutos jogados.
90s	Minutos jogados convertidos em equivalentes de partidas completas (90 minutos).
Produção ofensiva (totais)	
Gols	Número total de gols marcados.
Assis.	Número total de assistências realizadas.
G+A	Soma de gols e assistências.
G-PB	Gols marcados excluindo penalidades.
PB	Número de pênaltis convertidos.
PT	Número de pênaltis cobrados.
Disciplina	
CrtsA	Cartões amarelos recebidos.
CrtV	Cartões vermelhos recebidos.
Métricas avançadas (totais)	
xG	*Expected Goals* (gols esperados), métrica que estima a probabilidade de finalizações resultarem em gol.
npxG	*Expected Goals* excluindo penalidades.
xAG	*Expected Assists* (assistências esperadas).
npxG+xAG	Soma de gols esperados sem pênaltis e assistências esperadas.
Progressão	
PrgC	Corridas progressivas com a bola.
PrgP	Passes progressivos realizados.
PrgR	Recebimentos progressivos de passe.

Variável	Descrição
Produção ofensiva (por 90 minutos)[a]	
Gols.1	Gols por 90 minutos[a].
Assis..1	Assistências por 90 minutos[a].
G+A.1	Gols mais assistências por 90 minutos[a].
G-PB.1	Gols sem pênaltis por 90 minutos[a].
G+A-PB	Gols mais assistências excluindo pênaltis por 90 minutos[a].
xG.1	*Expected Goals* por 90 minutos[a].
xAG.1	*Expected Assists* por 90 minutos[a].
xG+xAG	Soma de xG e xAG por 90 minutos[a].
npxG.1	*Expected Goals* sem pênaltis por 90 minutos[a].
npxG+xAG.1	Soma de npxG e xAG por 90 minutos[a].
Ações defensivas	
Div	Desarmes (*tackles*) realizados.
TklW	Desarmes vencidos.
Terço Def	Desarmes realizados no terço defensivo do campo.
Terço Central	Desarmes realizados no terço central do campo.
Terço de Ataque	Desarmes realizados no terço ofensivo.
Div.1	Tentativas de desarme.
Tent	Tentativas totais de desarme contra adversários.
Tkl%	Percentual de sucesso nos desarmes.
Perdido	Número de vezes em que o jogador foi driblado.
Bloqueios e interceptações	
Bloqueios	Número de ações de bloqueio (chutes ou passes bloqueados).
TC	Chutes bloqueados.
Passe	Passes bloqueados.
Crts	Interceptações realizadas.
Tkl+Int	Soma de desarmes e interceptações.
Def	Ações defensivas realizadas na área defensiva.
Erros	Erros que resultaram em finalizações adversárias.

[a] Variáveis normalizadas por 90 minutos (equivalente a uma partida completa).

Variável	Descrição

Fonte: FBref (2026), adaptado pelo autor.

Após a etapa de organização, os dados estruturados foram armazenados em um repositório público na plataforma GitHub, com o objetivo de garantir reprodutibilidade, transparência metodológica e facilidade de acesso para futuras análises. O repositório utilizado neste estudo está disponível em https://github.com/LunusMax/epl-seriea-data.

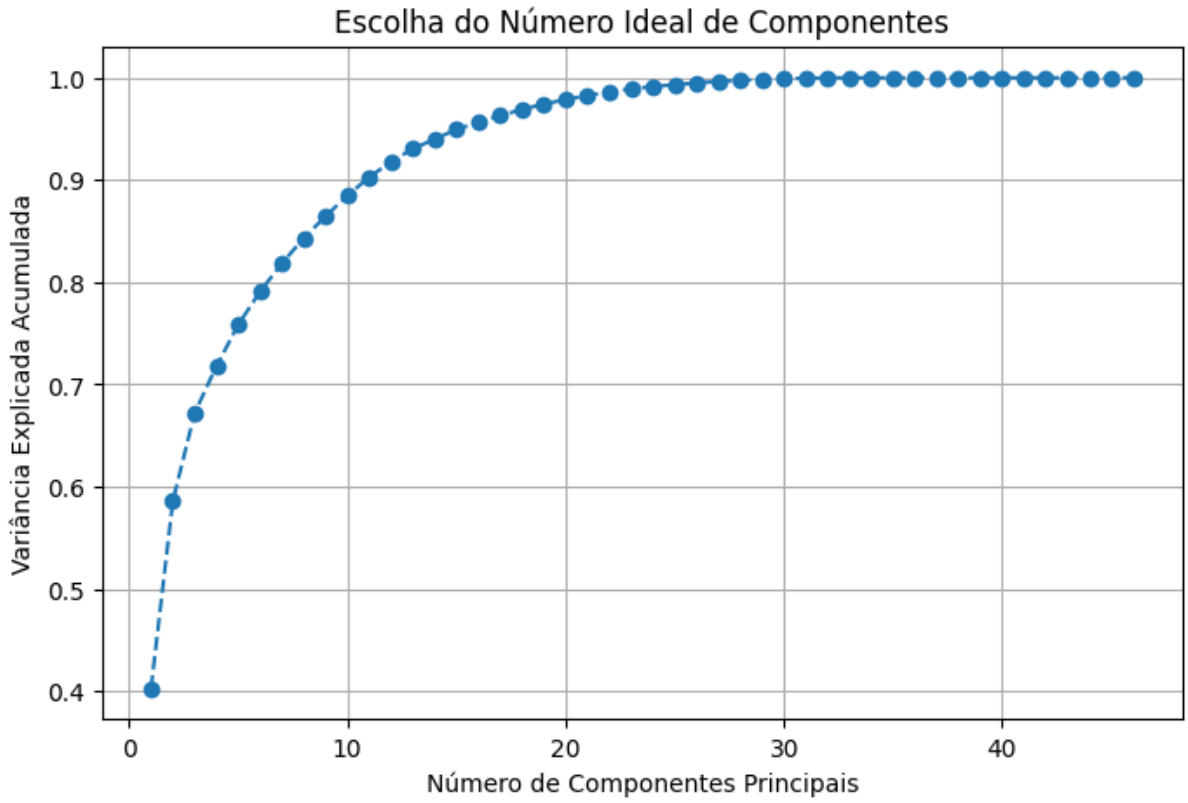
Em seguida, os arquivos CSV disponibilizados no repositório GitHub foram carregados no ambiente de execução do Google Colab. A partir desses arquivos, os dados dos jogadores foram lidos e organizados em *dataframes*, sendo realizada a concatenação dos registros pertencentes a cada liga. Cada registro da base de dados representa o conjunto de estatísticas produzidas por um jogador atuando por um determinado clube em uma temporada específica. Como resultado, foram construídos dois conjuntos de dados distintos, um correspondente à Premier League e outro à Serie A italiana, cada um contendo todos os jogadores das respectivas ligas ao longo das temporadas consideradas no estudo. Esses conjuntos foram utilizados nas etapas subsequentes da análise.

Considerando o elevado número de atributos e a possível correlação entre variáveis, foi aplicada uma técnica de redução de dimensionalidade com o objetivo de mitigar *overfitting* e melhorar a eficiência computacional dos modelos de AM. Para isso, utilizou-se o Principal Component Analysis (PCA), conforme descrito na subseção a seguir.

5.2.1.1 Principal Component Analysis (PCA)

Para determinar o número ideal de componentes principais que a serem utilizados no PCA, foi gerado um *Scree Plot*, conforme ilustrado na Figura 5.2.

Figura 5.2 – Escolha do número ideal de componentes do PCA



Fonte: Elaborado pelo autor.

O gráfico apresenta a variância explicada acumulada em função do número de componentes principais, permitindo identificar o ponto de inflexão, ou seja, o momento em que a adição de novos componentes passa a contribuir de forma marginal para a variância total dos dados. O eixo x representa o número de componentes principais, enquanto o eixo y exibe a variância explicada acumulada.

A partir da Figura 5.2, observa-se que, acima de 20 componentes principais, o ganho de informação adicional se torna insignificante. Com base nessa análise, foram realizados testes com diferentes quantidades de componentes, garantindo que sempre eram utilizados no mínimo 20 componentes, de forma a preservar a maior quantidade possível de variância dos dados sem comprometer a eficiência do modelo. Optou-se, por fim, pela utilização de 20 componentes principais, uma vez que esse valor já era suficiente para explicar 97,66% da variância total dos dados, indicando que praticamente toda a informação relevante do conjunto original estava preservada nessa representação reduzida.

Originalmente, cada jogador era descrito por 47 atributos estatísticos. Após a aplicação do PCA, esse conjunto foi reduzido para 20 atributos por jogador. Além disso, para padronizar

a representação das equipes, cada elenco foi truncado para conter apenas os 20 jogadores com maior minutagem na temporada, removendo aqueles que tiveram menor participação. Essa decisão foi motivada pelo fato de que alguns times apresentavam um número elevado de jogadores com baixa minutagem e, consequentemente, menor contribuição efetiva. Alternativamente, foram realizados testes preenchendo equipes com menor número de jogadores por meio de valores nulos, porém essa abordagem mostrou-se inadequada por dificultar o aprendizado do modelo.

Dessa forma, cada equipe passou a ser representada por 20 jogadores, cada um descrito por 20 atributos derivados do PCA, resultando em um total de 400 atributos relacionados ao desempenho individual dos jogadores. Adicionalmente, foi incorporada como variável explicativa a pontuação obtida pelo clube na temporada anterior. Assim, cada equipe passou a ser descrita por um vetor de 401 atributos.

A Tabela 5.2 apresenta uma representação simplificada da estrutura dos dados utilizados como entrada do modelo. Cada equipe é representada por 20 jogadores, cada um descrito por 20 componentes principais, além da pontuação obtida pelo clube na temporada anterior.

Tabela 5.2 – Representação simplificada da estrutura dos dados de entrada do modelo

Jogador	PC1	PC2	…	PC20	Pontuação $t-1$
J_1	$x_{1,1}$	$x_{1,2}$	…	$x_{1,20}$	p_{t-1}
J_2	$x_{2,1}$	$x_{2,2}$	…	$x_{2,20}$	p_{t-1}
J_3	$x_{3,1}$	$x_{3,2}$	…	$x_{3,20}$	p_{t-1}
⋮	⋮	⋮	⋱	⋮	⋮
J_{20}	$x_{20,1}$	$x_{20,2}$	…	$x_{20,20}$	p_{t-1}

Fonte: Elaborado pelo autor.

Observa-se que, nessa representação, cada equipe é descrita pelo conjunto de jogadores que compõem seu elenco, sendo cada atleta caracterizado por seus componentes principais. Como a ordem em que os jogadores aparecem na estrutura de dados não possui significado intrínseco para o problema estudado, diferentes ordenações do mesmo elenco representam essencialmente a mesma equipe. Além disso, considerando que o número de observações disponíveis ainda é relativamente limitado para o treinamento de modelos de AM, optou-se pela utilização de uma estratégia de *data augmentation*, descrita a seguir, com o objetivo de gerar múltiplas representações equivalentes dos elencos, ampliando o conjunto de treinamento e contribuindo para reduzir o risco de *overfitting*.

5.2.2 Data Augmentation

Embora o conjunto de dados utilizado neste estudo contemple múltiplas temporadas e duas ligas distintas, o número total de observações ainda pode ser considerado relativamente limitado para o treinamento de modelos de AM. Em cenários desse tipo, técnicas de *data augmentation* podem ser empregadas para ampliar artificialmente o conjunto de treinamento, contribuindo para reduzir o risco de *overfitting* e melhorar a capacidade de generalização dos modelos.

No presente trabalho, foi adotada uma estratégia de aumento de dados baseada na geração de múltiplas representações equivalentes de cada equipe em uma determinada temporada. Inicialmente, cada observação recebeu um identificador auxiliar (*copy*) com valor igual a zero, representando a base original. Em seguida, foram geradas k cópias adicionais do conjunto de dados contendo os componentes principais obtidos por meio do PCA.

Para cada cópia gerada, as linhas do *dataframe* foram embaralhadas aleatoriamente utilizando diferentes sementes de aleatoriedade. Esse procedimento não altera os valores das variáveis associadas a cada jogador, mas modifica a ordem em que os jogadores aparecem no conjunto de dados. Como o processo de construção das variáveis de entrada posteriormente reorganiza os dados por meio de uma operação de *pivot*, na qual os jogadores de cada equipe passam a ocupar posições indexadas no vetor de atributos, diferentes ordenações produzem diferentes representações vetoriais de um mesmo elenco.

Após o embaralhamento, as cópias geradas foram concatenadas ao conjunto original, formando um novo conjunto de treinamento ampliado. Neste estudo, foram geradas $k = 200$ cópias adicionais da base, de modo que cada equipe e temporada passou a possuir múltiplas representações no conjunto de dados, diferenciadas apenas pela ordenação dos jogadores no vetor de atributos.

Essa estratégia permite expor o modelo a diversas permutações equivalentes de um mesmo elenco durante o treinamento, reduzindo a dependência da predição em relação à ordem arbitrária dos jogadores no vetor de entrada. Como não há modificação nos valores das variáveis originais, a distribuição estatística dos dados é preservada, ao mesmo tempo em que se aumenta a diversidade das amostras utilizadas no processo de aprendizado.

5.2.3 Base de Dados dos Jogos

Os dados dos jogos realizados entre as temporadas 2019-2020 e 2023-2024, obtidos na etapa de coleta, estavam organizados em 14 colunas. No entanto, grande parte dessas informações era irrelevante para a análise (por exemplo, horário e data das partidas, público, local e árbitro). Assim, para otimizar o conjunto de dados, foram removidos os atributos considerados ruído, mantendo-se apenas três elementos essenciais: time mandante, time visitante e resultado da partida.

A coluna de resultados foi transformada em valores numéricos por meio de um algoritmo em Python, em processo análogo a um *encoder*: vitória do mandante foi codificada como '1', vitória do visitante como '2' e empate como '0'.

Com base nessa codificação, percorreu-se a base atribuindo três pontos a cada vitória e um ponto a cada empate. O resultado desse processamento foi uma tabela organizada por temporada, contendo cada clube e a respectiva pontuação acumulada.

5.3 Configuração Experimental

Os experimentos computacionais foram realizados na plataforma *Google Colab*, ambiente de computação em nuvem que disponibiliza recursos de hardware, incluindo suporte a GPU. A implementação foi desenvolvida em Python, com uso das bibliotecas *Pandas*, *NumPy*, *Scikit-learn* e *TensorFlow*.

A utilização desse ambiente permitiu a execução eficiente dos modelos e facilitou a reprodutibilidade dos experimentos, eliminando dependências de infraestrutura local.

Para garantir a reprodutibilidade dos resultados, foi fixada uma semente aleatória (*seed* = 532) nas bibliotecas utilizadas. Essa configuração reduz a variabilidade decorrente de processos estocásticos presentes tanto no treinamento do modelo quanto nos algoritmos de otimização.

5.4 Treinamento dos Modelos de Aprendizado de Máquina (AM)

Os dados de pontuação dos clubes foram organizados por temporada, enquanto as estatísticas individuais dos jogadores corresponderam sempre à temporada anterior. Assim, para cada equipe e temporada considerada, os modelos utilizam informações de desempenho indi-

vidual observadas na temporada $t-1$ para estimar a pontuação final do clube na temporada subsequente t. Essa estratégia garante que todas as variáveis utilizadas como entrada estejam disponíveis antes do período que se pretende prever, evitando a utilização de informações futuras no processo de modelagem.

Além das estatísticas individuais dos jogadores, a pontuação do clube na temporada anterior também foi incorporada como variável explicativa, permitindo capturar o histórico recente de desempenho da equipe. A inclusão dessa variável busca representar fatores estruturais do clube, como estabilidade tática, qualidade do elenco e consistência competitiva ao longo do tempo.

Diferentes modelos de AM foram treinados com o objetivo de prever a pontuação final dos clubes ao longo das temporadas, utilizando como base o desempenho individual de seus jogadores. Para evitar a ocorrência de *data leakage* entre as etapas de treinamento e avaliação, adotou-se uma estratégia de separação entre ligas. Nesse contexto, os dados de uma das ligas foram utilizados exclusivamente para o treinamento dos modelos e para o ajuste das transformações aplicadas aos dados, incluindo a redução de dimensionalidade por meio do PCA.

Posteriormente, os modelos treinados foram aplicados aos dados da outra liga, que foram utilizados apenas para a etapa de teste e validação das previsões. Dessa forma, garante-se que o treinamento e a avaliação ocorram sempre em competições distintas, independentemente de qual liga seja utilizada em cada etapa. Essa estratégia assegura que os modelos sejam avaliados em um conjunto de dados completamente diferente daquele utilizado durante o processo de aprendizado, proporcionando uma avaliação mais realista de sua capacidade de generalização.

5.4.1 Regressão Linear (RL)

Neste trabalho, a Regressão Linear (RL) foi implementada utilizando a biblioteca *Scikit-learn*, com configuração padrão. O modelo foi incluído como uma abordagem de referência (*baseline*), permitindo comparar o desempenho de métodos mais complexos com uma técnica simples, amplamente utilizada em problemas de regressão.

A predição da pontuação final dos clubes foi realizada a partir das variáveis explicativas obtidas após as etapas de preparação e transformação dos dados. Os coeficientes do modelo foram ajustados por mínimos quadrados ordinários, conforme a implementação padrão da biblioteca utilizada.

5.4.2 Random Forest (RF)

O modelo de *Random Forest* (RF) foi implementado utilizando a biblioteca *Scikit-learn*. No presente trabalho, foram adotados os seguintes hiperparâmetros principais:

- Número de árvores: `n_estimators = 500`
- Profundidade máxima: `max_depth = 12`
- Número mínimo de amostras para divisão: `min_samples_split = 5`
- Número mínimo de amostras por folha: `min_samples_leaf = 2`
- Seleção aleatória de atributos: `max_features = sqrt`

A utilização de múltiplas árvores permitiu avaliar a capacidade do modelo em capturar relações não lineares entre as variáveis explicativas e a pontuação final dos clubes. O RF foi incluído entre os modelos testados por sua robustez e ampla utilização em tarefas de regressão com dados estruturados.

5.4.3 Gradient Boosting (XGBoost)

Neste estudo, foi utilizada a implementação do algoritmo *Extreme Gradient Boosting* (XGBoost), por meio da biblioteca *XGBoost*. O modelo foi avaliado como uma alternativa baseada em *boosting* para a tarefa de predição da pontuação final dos clubes.

Foram adotados os seguintes hiperparâmetros principais:

- Número de árvores: `n_estimators = 500`
- Profundidade máxima das árvores: `max_depth = 6`
- Taxa de aprendizado: `learning_rate = 0.05`
- Subamostragem das observações: `subsample = 0.8`
- Subamostragem das variáveis: `colsample_bytree = 0.8`

A escolha desse modelo se deve à sua capacidade de lidar com padrões complexos em dados estruturados, além de seu desempenho recorrente em problemas supervisionados de regressão.

5.4.4 Multilayer Perceptron (MLP)

Para a etapa de previsão da pontuação dos clubes, foi implementado o modelo de MLP utilizando a biblioteca *Keras*. O modelo foi construído no formato *Sequential*, contendo três camadas densamente conectadas. A primeira camada recebe como entrada um vetor com dimensão igual ao número de atributos disponíveis, seguido por uma camada densa com 128 neurônios e função de ativação ReLU. Esta camada foi regularizada por meio de penalização L1, de forma a reduzir o risco de sobreajuste.

Em seguida, aplicou-se normalização em lote (*Batch Normalization*) e uma camada de *Dropout* com taxa de 60%, visando maior robustez e generalização. A segunda camada oculta é composta por 32 neurônios, também com ativação ReLU e regularização L1, seguida novamente por *Batch Normalization* e *Dropout* de 60%. Por fim, a camada de saída é formada por um único neurônio linear, responsável por produzir a previsão da pontuação final do clube na temporada.

A arquitetura do modelo pode ser resumida da seguinte forma:

- Camada de entrada: vetor de dimensão n (atributos do jogador e do clube).
- Primeira camada oculta: `Dense(128)`, ativação ReLU, regularização L1.
- `Batch Normalization`.
- `Dropout(0.6)`.
- Segunda camada oculta: `Dense(32)`, ativação ReLU, regularização L1.
- `Batch Normalization`.
- `Dropout(0.6)`.
- Camada de saída: `Dense(1)`, ativação linear.

O MLP foi escolhido como base para o modelo de previsão de pontuação no contexto do futebol por reunir propriedades teóricas e práticas que o tornam adequado para lidar com a natureza complexa e não linear dos dados esportivos. Entre as principais razões para essa escolha, destacam-se:

1. **Capacidade de representação universal** — Um MLP com pelo menos uma camada oculta e funções de ativação não lineares é capaz de aproximar qualquer função mensurável (Hornik, 1991). No contexto do futebol, isso significa que o modelo pode aprender padrões não lineares entre variáveis de desempenho (como gols, desarmes, progressão). Assim, o MLP não depende de hipóteses rígidas sobre a forma das relações entre variáveis, sendo capaz de aprendê-las diretamente dos dados.

2. **Flexibilidade para entradas de alta dimensão** — De acordo com *Goodfellow et. al., 2016*, cada camada de uma rede *feedforward* aprende uma representação do vetor de entrada que torna mais fácil a predição da saída desejada. Essa característica confere ao MLP flexibilidade para lidar com conjuntos de dados de alta dimensionalidade, nos quais múltiplas variáveis interagem de forma não linear. Mesmo com a aplicação prévia da técnica de *Principal Component Analysis* (PCA), que reduz a dimensionalidade de 48 para 20 componentes principais, o problema ainda envolve um espaço vetorial de considerável complexidade. Nesse contexto, o MLP é capaz de explorar as relações não lineares entre os componentes principais, aprendendo representações mais informativas que auxiliam na previsão de desempenho esportivo.

3. **Treinamento eficiente via retropropagação** — O algoritmo de *backpropagation*, apresentado em (Goodfellow; Bengio; Courville, 2016), permite o cálculo eficiente dos gradientes de erro e a atualização simultânea dos pesos da rede por meio de gradiente descendente. Isso viabiliza o treinamento em grandes bases de dados de futebol, reduz o tempo de convergência e possibilita otimizar métricas contínuas.

Além dessas vantagens teóricas, o MLP oferece um equilíbrio entre capacidade de generalização, interpretabilidade e eficiência computacional, características desejáveis em modelos aplicados ao futebol, onde os dados são heterogêneos, ruidosos e interdependentes.

Com base nos procedimentos descritos neste capítulo, foram desenvolvidos os modelos de AM utilizados para a predição do desempenho das equipes a partir das estatísticas individuais dos jogadores. A partir dessas modelagens, torna-se possível avaliar o desempenho dos modelos e analisar sua capacidade de generalização entre diferentes ligas. Assim, no próximo capítulo são apresentados e discutidos os resultados obtidos a partir da aplicação dos modelos propostos.

6 AVALIAÇÃO DOS MODELOS DE APRENDIZADO DE MÁQUINA (AM)

Este capítulo apresenta os resultados obtidos a partir da aplicação dos modelos de Aprendizado de Máquina desenvolvidos neste trabalho. Inicialmente, são discutidas as métricas de desempenho alcançadas pelos modelos na tarefa de predição da pontuação final das equipes. Em seguida, é realizada uma análise do comportamento das previsões, buscando avaliar a capacidade dos modelos em capturar padrões relevantes a partir das estatísticas individuais dos jogadores.

6.1 Modelos de Aprendizado de Máquina (AM)

Como referência para interpretação dos resultados, considera-se o modelo de baseline implícito associado ao coeficiente de determinação. Em problemas de regressão, um valor de $R^2 = 0$ corresponde a um modelo que simplesmente prevê a média da variável alvo para todas as observações, isto é, a pontuação média dos clubes na liga. Dessa forma, valores positivos de R^2 indicam que o modelo consegue explicar parte da variabilidade observada nos dados, superando a estratégia trivial de prever sempre a média das pontuações.

O melhor modelo de AM desenvolvido neste estudo apresentou um coeficiente de determinação máximo de aproximadamente 0,55 na tarefa de prever a pontuação final de um time em sua liga nacional, utilizando como base os dados estatísticos de desempenho individual de seus jogadores na temporada anterior.

Para avaliar o desempenho dos diferentes algoritmos aplicados à tarefa de previsão, foram calculadas métricas amplamente utilizadas em problemas de regressão, incluindo o coeficiente de determinação (R^2), o erro absoluto médio (MAE), o erro quadrático médio (MSE), a raiz do erro quadrático médio (RMSE), o erro percentual absoluto médio (MAPE) e o tempo de processamento de cada modelo. A Tabela 6.1 apresenta um resumo comparativo dos resultados obtidos, permitindo analisar tanto a qualidade preditiva quanto o custo computacional de cada abordagem.

Tabela 6.1 – Comparação de desempenho entre os modelos de Machine Learning avaliados no conjunto de testes

Modelo	**R^2**	**MAE**	**MSE**	**RMSE**	**MAPE**	**Tempo (s)**
RL	0,41	11,48	191,78	13,85	0,2580	1,72
RF	0,46	10,52	174,98	13,23	0,2390	118,66
XGBoost	0,41	11,23	191,23	13,83	0,2512	99,08
MLP	0,55	11,08	170,42	13,05	0,2424	161,68

Fonte: Elaborado pelo autor.

Os resultados apresentados na Tabela 6.1 evidenciam um desempenho moderado dos algoritmos avaliados na tarefa de previsão da pontuação final dos clubes. O melhor modelo alcançou um valor de R^2 de aproximadamente 0,55, indicando que parte significativa da variabilidade da pontuação pode ser explicada pelas variáveis consideradas, embora ainda exista um grau relevante de incerteza inerente ao problema. A pontuação de um time ao longo de uma temporada é influenciada por diversos fatores não diretamente capturados pelas estatísticas individuais dos jogadores, como aspectos táticos, mudanças de treinador, lesões, dinâmica coletiva e variações contextuais ao longo do campeonato, o que naturalmente impõe um limite ao desempenho preditivo dos modelos.

Entre os métodos avaliados, a Rede Neural Multicamadas (MLP) apresentou o melhor desempenho geral. O modelo obteve o maior valor de R^2 (0,55), além dos menores valores de MSE e RMSE, indicando maior capacidade de capturar a variabilidade presente nos dados. Esses resultados sugerem que a arquitetura neural foi mais eficaz em explorar relações não lineares existentes entre os componentes principais gerados pelo PCA e a pontuação final das equipes.

O modelo de *Random Forest* apresentou o segundo melhor desempenho, alcançando R^2 de 0,46 e os menores valores de MAE e MAPE entre os modelos avaliados. Esse resultado indica que o método baseado em árvores também foi capaz de capturar padrões relevantes nos dados, embora com menor capacidade explicativa global em comparação à MLP. Ainda assim, o custo computacional do RF foi significativamente inferior ao da rede neural, o que pode torná-lo uma alternativa interessante em cenários em que o tempo de treinamento seja um fator crítico.

Por outro lado, os modelos de Regressão Linear (RL) e *XGBoost* apresentaram desempenhos bastante semelhantes entre si, ambos com R^2 em torno de 0,41 e erros superiores aos ob-

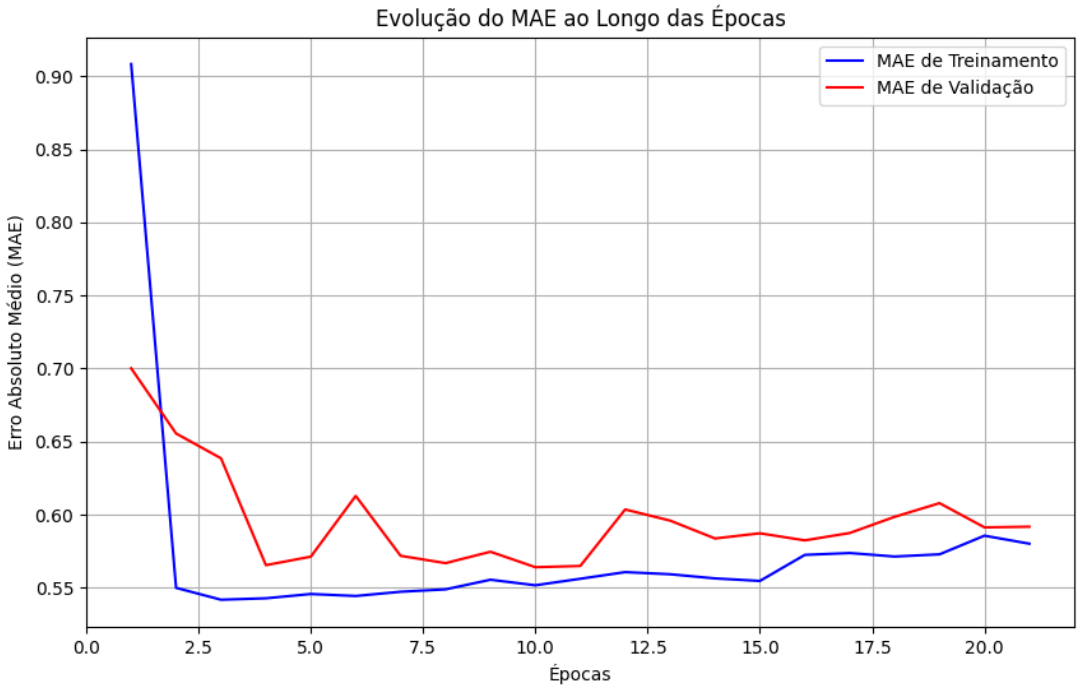
servados nos demais métodos. Esse comportamento sugere que, embora parte da variabilidade da pontuação possa ser explicada por relações aproximadamente lineares entre os componentes principais e a variável alvo, a presença de padrões não lineares limita a capacidade desses modelos de alcançar desempenhos superiores nesse contexto.

De maneira geral, os resultados indicam que modelos capazes de capturar relações não lineares mais complexas tendem a apresentar melhor desempenho na tarefa considerada. Dessa forma, o MLP foi selecionado como modelo preditivo para as etapas subsequentes do estudo.

6.1.1 Predição do Modelo Multilayer Perceptron (MLP)

Antes de analisar as previsões realizadas pelo modelo, foi avaliado o comportamento do processo de treinamento da rede neural ao longo das épocas. A Figura 6.1 apresenta a evolução do erro absoluto médio (MAE) durante o treinamento da rede MLP, considerando tanto o conjunto de treinamento quanto o conjunto de validação.

Figura 6.1 – Evolução do erro absoluto médio (MAE) ao longo das épocas de treinamento da rede MLP para os conjuntos de treinamento e validação.



Fonte: Elaborado pelo autor.

Observa-se uma redução acentuada do erro nas primeiras épocas de treinamento, indicando que o modelo rapidamente aprende padrões relevantes presentes nos dados. Após essa

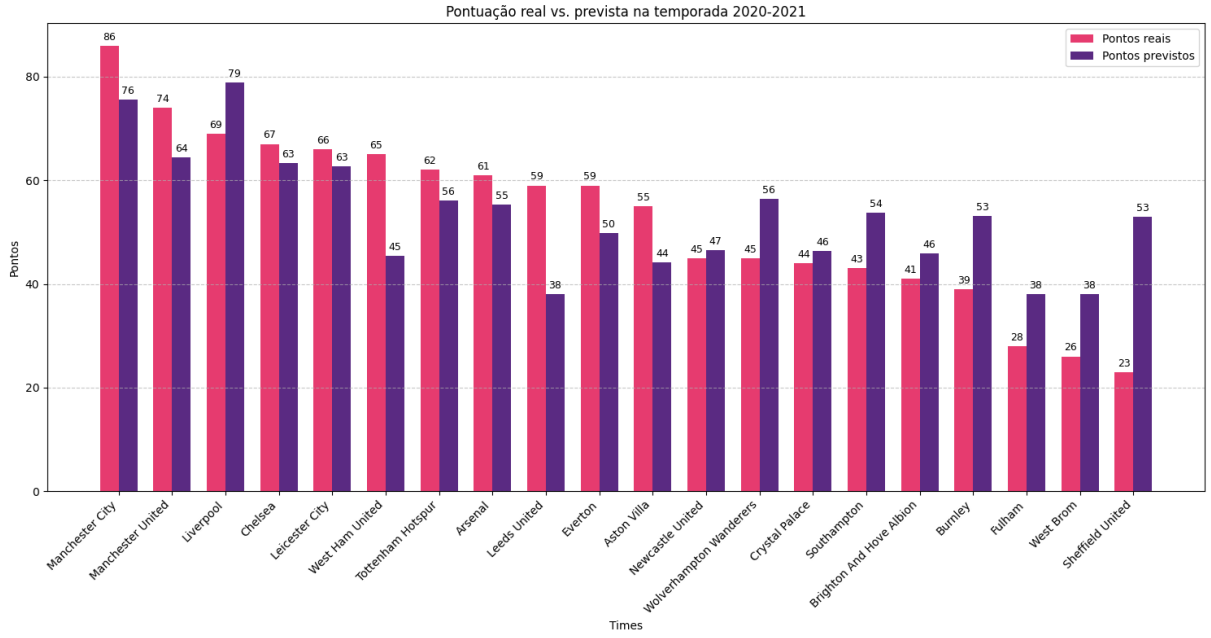
fase inicial, o erro tende a estabilizar, apresentando pequenas oscilações ao longo das épocas subsequentes.

Nota-se também que as curvas de treinamento e validação permanecem relativamente próximas durante todo o processo de treinamento, sugerindo ausência de sobreajuste significativo (*overfitting*). Esse comportamento indica que o modelo manteve boa capacidade de generalização ao longo do treinamento.

Com o intuito de compreender a predição realizada pelo melhor modelo (MLP), foi realizada uma análise visual da relação entre as pontuações reais dos clubes da Premier League e as pontuações previstas pelo modelo ao longo das temporadas analisadas.

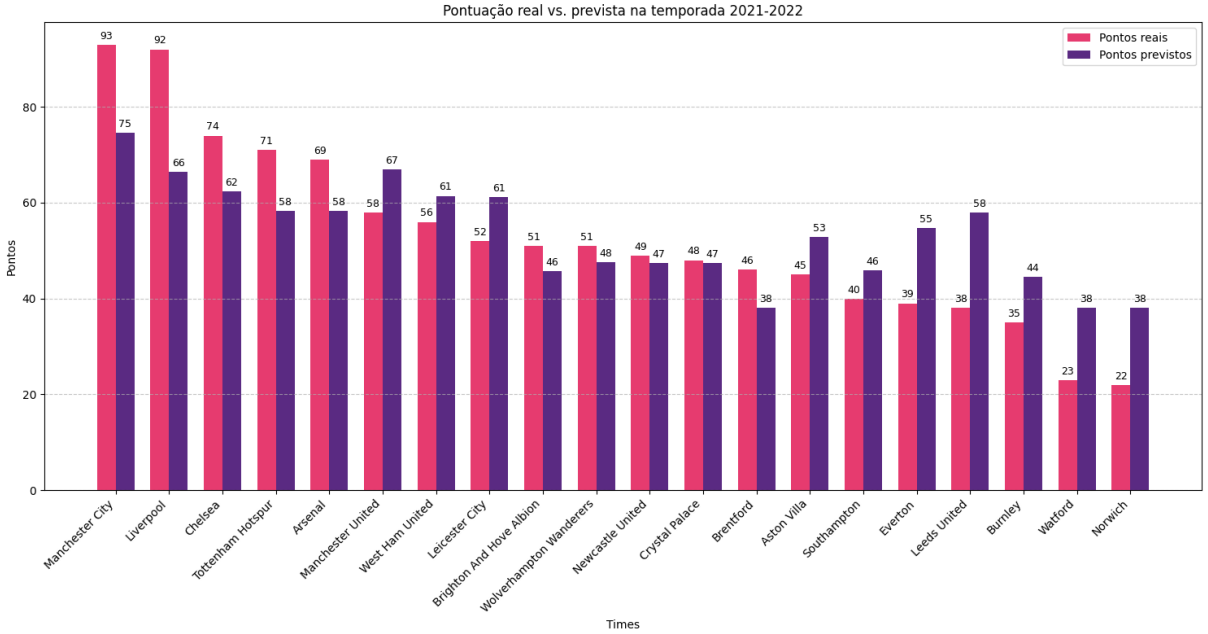
As Figuras 6.2–6.5 apresentam os gráficos de comparação entre as pontuações reais e previstas pelo modelo MLP para as temporadas de 2020–2021 a 2023–2024 da Premier League.

Figura 6.2 – Comparação entre pontuações reais e previstas pelo modelo MLP na temporada 2020–2021 da Premier League.



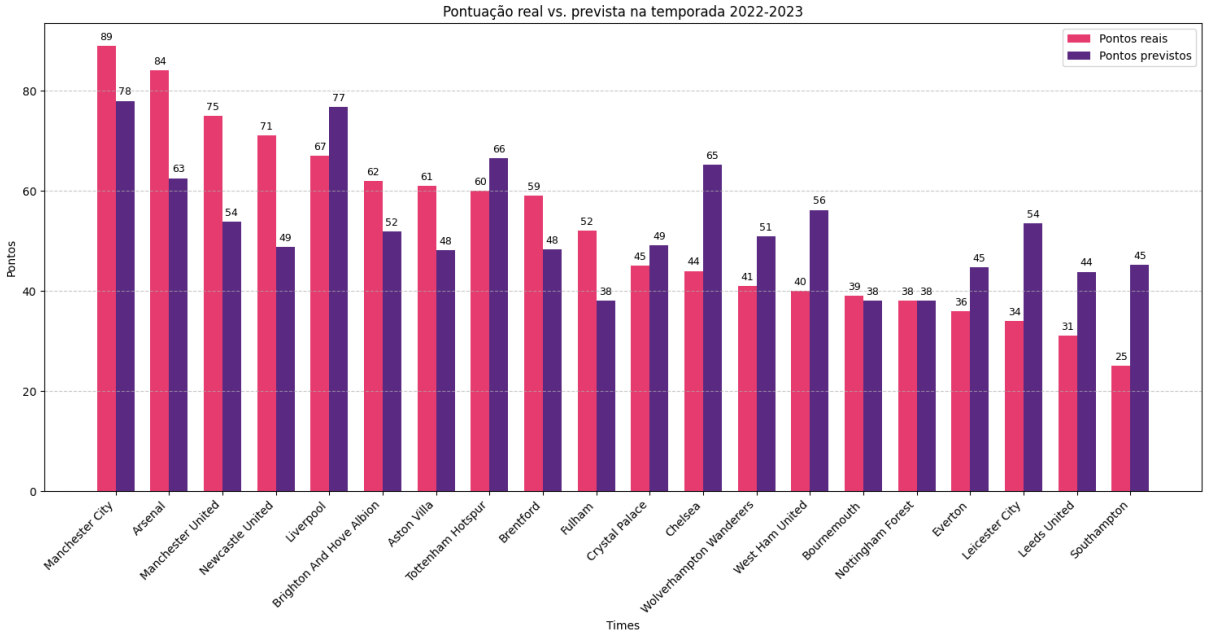
Fonte: Elaborado pelo autor.

Figura 6.3 – Comparação entre pontuações reais e previstas pelo modelo MLP na temporada 2021–2022 da Premier League.



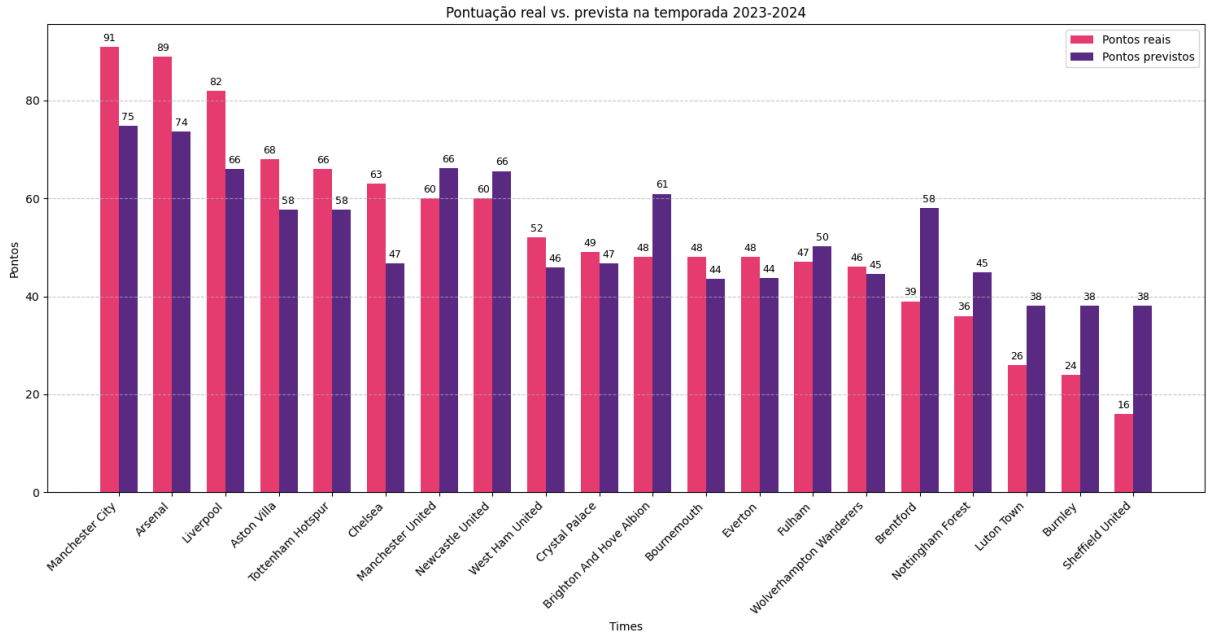
Fonte: Elaborado pelo autor.

Figura 6.4 – Comparação entre pontuações reais e previstas pelo modelo MLP na temporada 2022–2023 da Premier League.



Fonte: Elaborado pelo autor.

Figura 6.5 – Comparação entre pontuações reais e previstas pelo modelo MLP na temporada 2023–2024 da Premier League.



Fonte: Elaborado pelo autor.

Os resultados apresentados nas Figuras 6.2–6.5 indicam que o modelo MLP apresenta maior precisão ao estimar as pontuações de clubes que terminaram as temporadas com valores próximos da média da liga. Clubes posicionados próximos à região central da tabela, tendem a ter suas pontuações previstas com menor erro. Esse comportamento é comum em modelos preditivos sob alta incerteza: quando os atributos disponíveis não capturam toda a variabilidade do fenômeno, as previsões convergem naturalmente para valores intermediários.

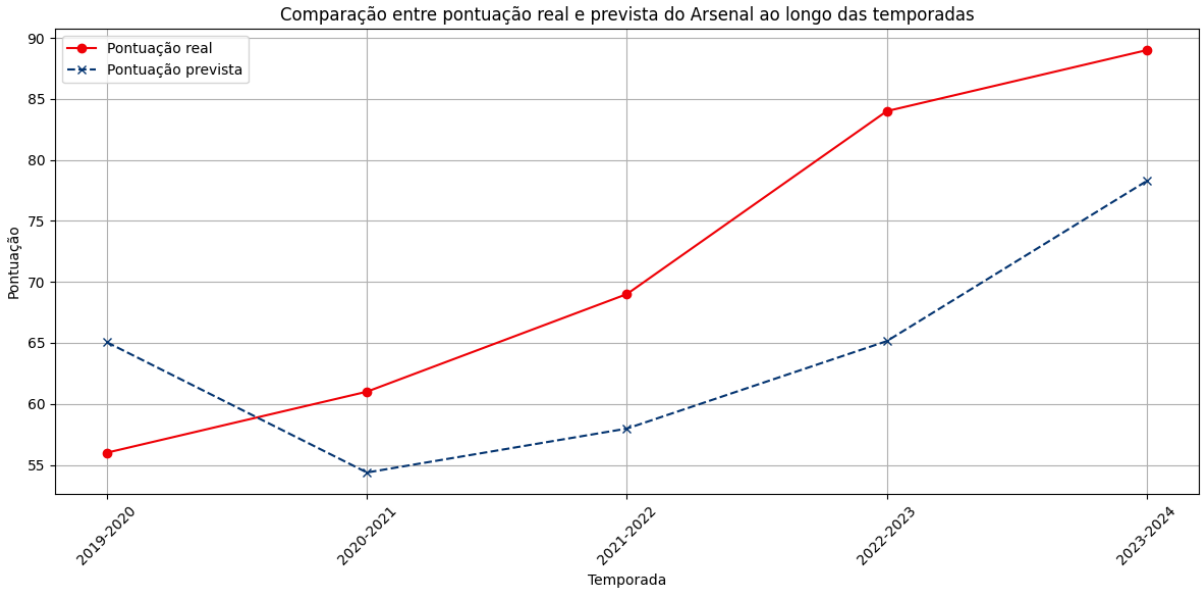
Esse resultado alinha-se com a literatura clássica, segundo a qual modelos preditivos em ambientes de alta variabilidade, como é o caso do futebol, tendem a gerar funções de saída menos extremas e com menor amplitude, aproximando-se de valores centrais, devido a mecanismos de *shrinkage* e ao balanço entre viés e variância discutidos por Bishop (Bishop, 2006). Esse comportamento é análogo ao fenômeno estatístico conhecido como regressão à média.

Dessa forma, observa-se que o modelo MLP é capaz de capturar tendências gerais de desempenho dos clubes com base nas estatísticas individuais de seus jogadores, porém nota-se uma subestimação sistemática das equipes de desempenho elevado, como Manchester City, Arsenal e Liverpool, e uma superestimação das equipes de baixo desempenho, como Sheffield United, Burnley e Luton Town.

Para ilustrar esse comportamento de forma mais detalhada, as Figuras 6.6 e 6.7 apresentam a comparação entre as pontuações reais e previstas pelo modelo MLP para dois clubes com desempenhos contrastantes na liga.

A Figura 6.6 apresenta o caso do Arsenal, uma equipe de alto desempenho ao longo das temporadas analisadas. Observa-se que o modelo é capaz de capturar a tendência geral de crescimento da pontuação do clube, embora apresente uma leve subestimação nas temporadas mais recentes.

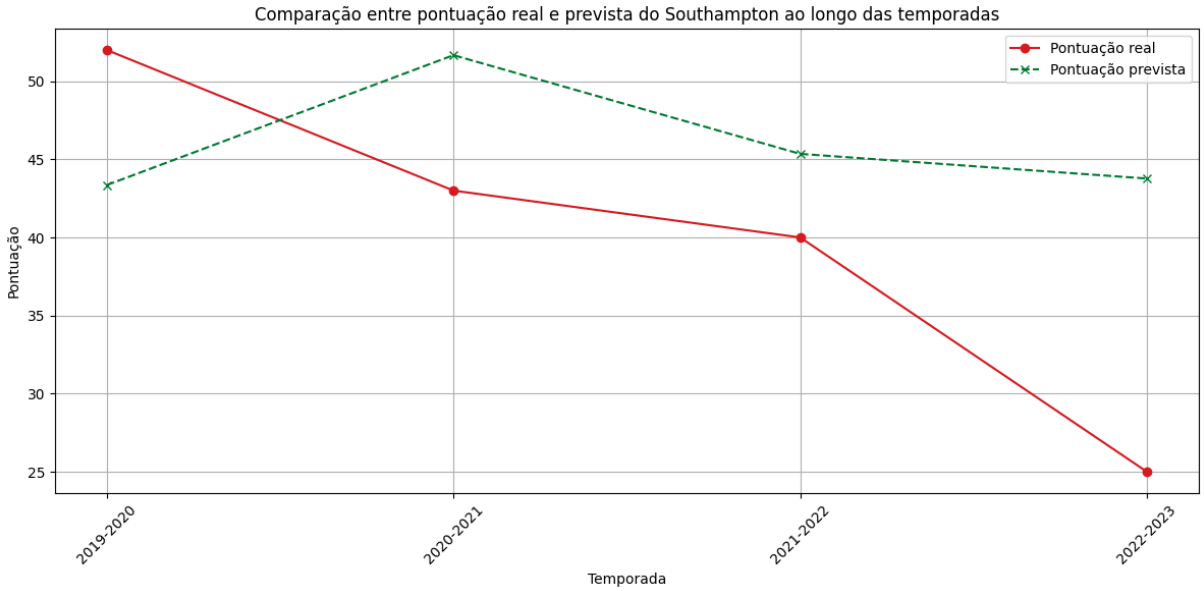
Figura 6.6 – Comparação entre pontuação real e prevista do Arsenal ao longo das temporadas.



Fonte: Elaborado pelo autor.

A Figura 6.7 apresenta o caso do Southampton, equipe que apresentou queda de desempenho ao longo das temporadas analisadas. Nesse caso, observa-se uma tendência de superestimação das pontuações pelo modelo, comportamento consistente com o fenômeno de regressão à média discutido anteriormente.

Figura 6.7 – Comparação entre pontuação real e prevista do Southampton ao longo das temporadas.



Fonte: Elaborado pelo autor.

Com base nos resultados apresentados neste capítulo, foi possível identificar o modelo de AM que apresentou o melhor desempenho na tarefa de predição da pontuação final das equipes. A partir dessa definição, o modelo selecionado (MLP) passa a ser utilizado como função objetivo na etapa de otimização da composição de elencos. Dessa forma, no próximo capítulo são apresentados os procedimentos de PO empregados para simular contratações e avaliar configurações de elenco capazes de maximizar o desempenho previsto das equipes.

7 OTIMIZAÇÃO DA COMPOSIÇÃO DE ELENCOS

Uma vez identificado o modelo de AM com melhor capacidade de predizer a pontuação final dos times em uma liga a partir das informações estatísticas de seus elencos, inicia-se a segunda etapa do projeto, dedicada à busca de soluções capazes de otimizar esse desempenho. Nessa fase, o problema passa a ser tratado como um processo de otimização sob restrições, no qual diferentes composições de elenco são avaliadas de acordo com sua pontuação estimada pelo modelo de AM.

7.1 Cálculo do Valor de Mercado Sintético

Para viabilizar o processo de otimização, torna-se necessário incorporar restrições financeiras que reflitam, ainda que de forma simplificada, as limitações orçamentárias enfrentadas por clubes de futebol em contextos reais. Nesse sentido, foi definida uma métrica sintética de valor de mercado dos jogadores, construída a partir de dois fatores principais: idade e desempenho estatístico agregado.

A Equação 7.1 apresenta formalmente o cálculo adotado para essa métrica.

$$V_i = 1000 \cdot \left(1000 \cdot \max(0, 30 - I_i) + 10 \cdot \sum_{j=1}^{n} x_{ij} \right) \tag{7.1}$$

onde:

- V_i representa o valor de mercado sintético do jogador i;
- I_i corresponde à idade do jogador i;
- x_{ij} representa o valor da j-ésima estatística numérica associada ao jogador i;
- n é o número total de variáveis estatísticas consideradas.

O primeiro termo da equação corresponde a um fator etário, que atribui maior peso a jogadores mais jovens, considerando um limite superior de 30 anos. Jogadores com idade superior a esse limiar não recebem penalização adicional, evitando a atribuição de valores negativos. Essa escolha busca refletir uma característica recorrente do mercado do futebol, no qual atletas mais jovens tendem a apresentar maior potencial de valorização, longevidade esportiva e retorno econômico futuro.

O segundo termo da equação agrega o desempenho estatístico do jogador por meio da soma de suas variáveis numéricas, escalonada por um fator constante. Essa agregação tem como objetivo capturar, de forma simplificada, o impacto global do desempenho esportivo individual, sem privilegiar métricas específicas ou posições em campo.

Por fim, o fator multiplicativo externo é aplicado exclusivamente para reescalonamento da variável, de modo a produzir valores em uma ordem de grandeza mais próxima às estimativas observadas no mercado real. Esse reescalonamento facilita tanto a interpretação dos valores obtidos quanto a utilização da métrica como restrição orçamentária nos modelos de otimização empregados na etapa subsequente.

Dessa forma, jogadores que combinam alto desempenho estatístico com menor idade tendem a apresentar valores de mercado sintético mais elevados, enquanto atletas mais velhos ou com desempenho inferior são associados a custos reduzidos, em consonância com evidências empíricas da literatura que apontam idade e desempenho como fatores centrais na formação do valor econômico de jogadores profissionais. (Frick, 2007; Müller; Simons; Weinmann, 2017) Ressalta-se que essa métrica não tem como objetivo reproduzir valores reais de mercado, mas sim fornecer uma função custo coerente e monotônica, adequada ao processo de otimização proposto.

A partir da definição dessa função de custo, torna-se possível formular o problema de seleção de jogadores como um problema de otimização combinatória, no qual se busca identificar configurações de elenco que maximizem o desempenho previsto da equipe sob restrições orçamentárias. Nesse contexto, diferentes estratégias heurísticas podem ser empregadas para explorar o espaço de soluções. Como tais métodos frequentemente apresentam componentes estocásticos e podem produzir resultados distintos entre execuções, sua avaliação requer múltiplas repetições experimentais e análise estatística apropriada. Nesse sentido, conforme discutido por (Derrac *et al.*, 2011), testes estatísticos não paramétricos são amplamente recomendados na comparação de algoritmos heurísticos, pois não pressupõem normalidade na distribuição dos resultados obtidos. A seguir, descreve-se os métodos de otimização utilizados neste trabalho.

7.2 Algoritmos Gulosos para Seleção Iterativa de Jogadores

Com o objetivo de explorar possíveis melhorias na composição do elenco, foram implementados algoritmos gulosos responsáveis por selecionar iterativamente jogadores a serem contratados.

O método parte do elenco atual da equipe e constrói gradualmente um conjunto de reforços. Em cada iteração, o algoritmo percorre a base de dados de jogadores e considera aqueles que atuaram na temporada imediatamente anterior à temporada analisada. Durante essa varredura, são desconsiderados jogadores que já pertencem ao elenco atual da equipe ou que já foram previamente selecionados em iterações anteriores do próprio algoritmo.

Além disso, cada possível contratação é submetida à restrição orçamentária. O valor de mercado do atleta, registrado na temporada anterior, é utilizado para verificar se a contratação pode ser realizada dentro do orçamento disponível. Apenas jogadores que respeitam essa restrição são avaliados.

A qualidade de cada possível contratação é avaliada por meio da função objetivo definida neste trabalho, implementada como o modelo de AM responsável por estimar a pontuação esperada da equipe na temporada considerada. Para manter constante o tamanho do elenco avaliado, a inclusão de novos jogadores é simulada removendo-se, da formação original, a mesma quantidade de atletas com menor minutagem na temporada analisada.

Cabe destacar que, durante a execução dos algoritmos de otimização, a função objetivo é avaliada de forma iterativa por meio de chamadas sucessivas ao modelo de AM desenvolvido neste trabalho.

Mais especificamente, a cada candidato à contratação considerado — seja na adição de um jogador nos algoritmos gulosos ou na substituição de jogadores nos métodos de busca local — é construída uma nova representação do elenco e submetida ao modelo preditivo, que retorna a pontuação esperada da equipe.

Dessa forma, o processo de otimização está diretamente acoplado ao modelo de AM, sendo a qualidade de cada solução inteiramente determinada pelas previsões geradas pelo modelo para cada configuração de elenco avaliada.

O processo é repetido até que não existam mais jogadores viáveis dentro da restrição orçamentária. Sempre que uma contratação é aceita, o valor de mercado do jogador selecionado é subtraído do orçamento disponível, atualizando a restrição orçamentária para as iterações

seguintes. Como característica fundamental da estratégia gulosa, as decisões tomadas em cada iteração são definitivas: uma vez que um jogador é selecionado e adicionado ao conjunto de reforços, essa escolha não é revista nas iterações subsequentes.

Foram implementadas duas variações desse algoritmo. Ambas possuem exatamente a mesma estrutura e seguem o mesmo procedimento descrito anteriormente, diferindo apenas na forma como a melhoria da função objetivo é aceita em cada iteração. Na versão *first-improvement*, os jogadores são avaliados sequencialmente durante a varredura da base e a primeira contratação que resulta em melhoria da função objetivo é imediatamente aceita. Já na versão *best-improvement*, todos os jogadores viáveis são avaliados em cada iteração e seleciona-se aquele cuja inclusão produz o maior aumento na função objetivo.

Os pseudocódigos dos algoritmos gulosos utilizados são apresentados nos Algoritmos 3 e 4.

Algoritmo 3 Algoritmo Guloso com estratégia First Improvement

```
Obter lista de jogadores candidatos
Inicializar conjunto de contratações como vazio
Calcular pontuação atual com conjunto vazio
Inicializar custo atual igual a zero
while verdadeiro do
    Definir variável de melhora como falsa
    for cada jogador candidato p do
        if p ainda não foi selecionado e cabe no orçamento then
            Calcular a nova pontuação prevista ao adicionar p
            if a nova pontuação for maior que a pontuação atual then
                Adicionar p ao conjunto de contratações
                Atualizar a pontuação atual
                Atualizar o custo atual
                Definir variável de melhora como verdadeira
                break                                   ▷ first improvement
            end if
        end if
    end for
    if nenhuma melhora for encontrada then
        break
    end if
end while
return conjunto de contratações, custo atual e pontuação prevista
```

Algoritmo 4 Algoritmo Guloso com estratégia Best Improvement

```
Obter lista de jogadores candidatos
Inicializar conjunto de contratações como vazio
Calcular pontuação atual com conjunto vazio
Inicializar custo atual igual a zero
while verdadeiro do
    Inicializar melhor jogador como vazio
    Inicializar melhor pontuação como vazia
    for cada jogador candidato p do
        if p ainda não foi selecionado e cabe no orçamento then
            Calcular a nova pontuação prevista ao adicionar p
            if não existir melhor jogador ou a nova pontuação for maior que a melhor pontuação then
                Armazenar p como melhor jogador da iteração
                Armazenar a nova pontuação como melhor pontuação da iteração
            end if
        end if
    end for
    if existe melhor jogador e a melhor pontuação for maior que a pontuação atual then
        Adicionar o melhor jogador ao conjunto de contratações
        Atualizar a pontuação atual
        Atualizar o custo atual
    else
        break
    end if
end while
return conjunto de contratações, custo atual e pontuação prevista
```

7.3 Algoritmos de Busca Local para Otimização do Elenco

Além das heurísticas gulosas descritas na subseção anterior, foram implementados algoritmos baseados em busca local com o objetivo de explorar de forma mais ampla o espaço de soluções possíveis para o problema de montagem de elenco.

Enquanto os algoritmos gulosos constroem o conjunto de contratações de forma incremental, selecionando jogadores individualmente a cada iteração, a estratégia de busca local parte de uma solução inicial completa e passa a realizar modificações estruturais nessa solução ao longo do processo de otimização.

Assim como na abordagem gulosa, a inclusão de jogadores no elenco é simulada mantendo constante o tamanho da formação avaliada. Para isso, sempre que um conjunto de con-

tratações é considerado, remove-se da formação original a mesma quantidade de atletas com menor minutagem na temporada analisada.

A solução inicial é construída selecionando aleatoriamente jogadores da temporada imediatamente anterior à temporada analisada e respeitando a restrição orçamentária estabelecida. Durante essa etapa, são desconsiderados jogadores que já pertencem ao elenco atual da equipe. O processo de seleção inicial é repetido até que o orçamento disponível não permita novas adições.

Para tornar o experimento computacionalmente viável durante a execução da busca local, foi adotada uma estratégia de amostragem do espaço de candidatos. Em vez de considerar todos os jogadores disponíveis na base de dados, foi definido um *pool* de 100 jogadores selecionados aleatoriamente entre os atletas elegíveis da temporada anterior. Esse subconjunto é utilizado como universo de candidatos para as operações de substituição ao longo do processo de busca. Embora essa abordagem reduza significativamente o custo computacional do algoritmo, ela também limita o espaço de soluções explorado, podendo impactar negativamente a qualidade da solução final encontrada. Esse tipo de restrição é comum em problemas de otimização combinatória de grande escala, nos quais a redução controlada do espaço de busca é empregada para viabilizar a aplicação prática dos algoritmos.

Uma vez obtida a solução inicial viável, o algoritmo passa a explorar sua vizinhança por meio de operações de substituição unitária (*swap*). Em cada vizinho gerado, um jogador do conjunto atual de contratações é substituído por outro jogador elegível ainda não selecionado. A substituição somente é considerada factível quando o valor de mercado do novo jogador, descontado o valor do atleta removido, respeita a restrição orçamentária estabelecida.

Uma vantagem importante da abordagem de busca local em relação às heurísticas gulosas é que a avaliação da função objetivo passa a considerar diretamente o conjunto completo de contratações selecionadas. Enquanto os algoritmos gulosos analisam o impacto marginal da adição de um único jogador por vez, a busca local avalia modificações em uma solução já formada, permitindo que possíveis interações entre os jogadores selecionados sejam levadas em conta pelo modelo de avaliação. Dessa forma, correlações positivas ou negativas entre atletas — capturadas implicitamente pelo modelo de AM utilizado como função objetivo — podem influenciar a escolha das substituições realizadas durante o processo de otimização.

De maneira análoga ao caso dos algoritmos gulosos, duas variações da busca local foram implementadas, diferenciando-se exclusivamente pelo critério de aceitação das melhorias

encontradas. Na versão *first-improvement*, as substituições são avaliadas sequencialmente e a primeira troca que produz melhoria na função objetivo é imediatamente aceita, reiniciando-se a exploração da vizinhança a partir da nova solução. Já na versão *best-improvement*, todas as substituições possíveis na vizinhança são avaliadas antes da atualização da solução corrente, sendo então aplicada a troca que produz o maior aumento na função objetivo.

O processo iterativo é encerrado quando nenhuma substituição viável resulta em melhoria adicional na função objetivo, caracterizando a convergência para um ótimo local no espaço de soluções explorado.

Os pseudocódigos dos algoritmos de busca local utilizados são apresentados nos Algoritmos 5 e 6.

Algoritmo 5 Busca Local com estratégia First Improvement

```
Obter uma amostra aleatória de jogadores candidatos
Obter o custo de cada jogador
Gerar solução inicial usando o algoritmo guloso
Armazenar conjunto atual de contratações, custo atual e pontuação atual
while houver melhora do
    Definir variável de melhora como falsa
    for cada posição i no conjunto atual de contratações do
        Definir o jogador da posição i como jogador de saída
        for cada jogador candidato p do
            if p não pertence ao conjunto atual e a troca cabe no orçamento then
                Criar uma cópia da solução atual
                Substituir o jogador da posição i por p
                Calcular a nova pontuação prevista
                if a nova pontuação for maior que a pontuação atual then
                    Atualizar a solução atual
                    Atualizar a pontuação atual
                    Recalcular o custo atual
                    Definir variável de melhora como verdadeira
                    break                                  ▷ first improvement
                end if
            end if
        end for
        if uma melhora for encontrada then
            break
        end if
    end for
end while
return conjunto de contratações, custo atual e pontuação prevista
```

Algoritmo 6 Busca Local com estratégia Best Improvement

```
Obter uma amostra aleatória de jogadores candidatos
Obter o custo de cada jogador
Gerar solução inicial usando o algoritmo guloso
Armazenar conjunto atual de contratações, custo atual e pontuação atual
while houver melhora do
    Definir variável de melhora como falsa
    for cada posição i no conjunto atual de contratações do
        Definir o jogador da posição i como jogador de saída
        Inicializar melhor substituição como vazia
        Inicializar melhor pontuação como vazia
        for cada jogador candidato p do
            if p não pertence ao conjunto atual e a troca cabe no orçamento then
                Criar uma cópia da solução atual
                Substituir o jogador da posição i por p
                Calcular a nova pontuação prevista
                if não existir melhor substituição ou a nova pontuação for maior que a melhor
pontuação then
                    Armazenar a nova solução como melhor substituição
                    Armazenar a nova pontuação como melhor pontuação
                end if
            end if
        end for
        if existe melhor substituição e a melhor pontuação for maior que a pontuação atual
then
            Atualizar a solução atual
            Atualizar a pontuação atual
            Recalcular o custo atual
            Definir variável de melhora como verdadeira
        end if
        if uma melhora for encontrada then
            break
        end if
    end for
end while
return conjunto de contratações, custo atual e pontuação prevista
```

Uma vez definidos os algoritmos de otimização e suas respectivas estratégias de exploração da vizinhança, torna-se possível avaliar empiricamente o comportamento dessas abordagens no problema de montagem de elenco considerado neste trabalho. No capítulo seguinte, são apresentados os resultados experimentais obtidos com a aplicação das heurísticas gulosas e dos métodos de busca local, analisando-se tanto a qualidade das soluções encontradas quanto o custo computacional associado a cada estratégia de busca.

8 RESULTADOS DA OTIMIZAÇÃO DE MONTAGEM DE ELENCO (PO)

Esta seção apresenta os resultados experimentais obtidos com a aplicação das heurísticas propostas para o problema de otimização da composição de elencos. Conforme discutido na seção anterior, o objetivo do problema consiste em identificar combinações de jogadores que maximizem o desempenho esperado de uma equipe ao longo da temporada.

Uma vez que os algoritmos de otimização utilizam o modelo de AM como função objetivo, a qualidade final das soluções depende diretamente das previsões produzidas por esse modelo. Dessa forma, tanto o algoritmo guloso quanto o método de busca local operam sobre a mesma função objetivo, diferenciando-se apenas pela estratégia utilizada para explorar o espaço de soluções. Assim, a comparação entre os métodos concentra-se principalmente na qualidade das soluções encontradas e no custo computacional necessário para alcançá-las.

A qualidade de cada solução gerada pelos algoritmos é medida pela pontuação prevista do time resultante, estimada pelo modelo de AM desenvolvido anteriormente neste trabalho. Esse modelo recebe como entrada as características agregadas do elenco e retorna a quantidade esperada de pontos que a equipe obteria na competição. Dessa forma, quanto maior a pontuação prevista, melhor é considerada a solução encontrada pelo algoritmo.

Como os elencos gerados pelos métodos de otimização representam cenários hipotéticos de contratação, não existe um resultado real observado para essas configurações. Assim, a avaliação da qualidade das soluções é realizada por meio da função objetivo definida pelo modelo preditivo, que estima o desempenho esperado da equipe para cada configuração de elenco simulada.

Antes da análise estatística dos experimentos, foi realizado um *benchmark* inicial com uma única execução de cada algoritmo. Esse experimento preliminar tem como objetivo ilustrar o comportamento dos métodos e destacar diferenças iniciais entre as estratégias de exploração do espaço de soluções.

Nesse *benchmark* inicial foram registrados o tempo de execução, o número de iterações, o número de avaliações da função objetivo e a pontuação final obtida pela solução encontrada. Esses resultados permitem uma primeira comparação entre os algoritmos, especialmente em relação ao custo computacional necessário para alcançar soluções de alta qualidade.

8.1 Experimento Preliminar (*Benchmark* inicial)

Para ilustrar o comportamento dos algoritmos de otimização, foi realizado inicialmente um experimento preliminar (*benchmark*) com uma única execução de cada método. Como cenário de teste, foi utilizado o elenco do Napoli na temporada 2021–2022, considerando um orçamento disponível de 120.000.000 € para contratações.

Antes da aplicação dos algoritmos de otimização, a pontuação prevista base do Napoli para essa temporada, estimada pelo modelo de aprendizado de máquina, foi de 71,83 pontos. O objetivo dos algoritmos consiste em selecionar jogadores adicionais que maximizem essa pontuação prevista dentro da restrição orçamentária estabelecida.

A Tabela 8.1 apresenta os resultados obtidos após uma única execução de cada algoritmo, incluindo o tempo de execução, o número de iterações realizadas, o número de avaliações da função objetivo e a pontuação final prevista para o elenco resultante.

Tabela 8.1 – Resultados do experimento preliminar (*benchmark*)

Algoritmo	Tempo (s)	Iterações	Avaliações	Melhoras	Score final
Guloso First-Improvement	255,08	4	1571	–	72,55
Guloso Best-Improvement	475,66	3	2809	–	72,53
Busca Local First-Improvement	1989,71	14	10538	13	74,66
Busca Local Best-Improvement	1637,75	3	8334	2	74,66

Fonte: Elaborado pelo autor.

Observa-se que todos os algoritmos foram capazes de melhorar a pontuação prevista da equipe em relação ao valor base de 71,83 pontos. Entre os métodos analisados, os algoritmos de busca local apresentaram a maior pontuação final, alcançando 74,66 pontos previstos. No entanto, esse desempenho foi obtido ao custo de um tempo de execução significativamente superior aos demais métodos, decorrente do grande número de avaliações da função objetivo realizadas durante o processo de busca.

Em contraste, o algoritmo Guloso First-Improvement apresentou o menor tempo de execução, porém produziu uma solução de menor qualidade quando comparado às demais abordagens. Já os métodos baseados em busca local produziram soluções melhores em termos de

pontuação, com destaque para o algoritmo Busca Local Best-Improvement, que apresentou menor tempo de execução e menor número de avaliações da função objetivo em comparação à variante Busca Local First-Improvement.

A solução com maior pontuação prevista foi obtida pelos algoritmos de Busca Local, que sugeriram a contratação dos jogadores Luis Muriel, Romelu Lukaku e Zlatan Ibrahimović.

A Tabela 8.2 apresenta algumas estatísticas ofensivas desses jogadores, obtidas a partir dos dados utilizados neste trabalho. Observa-se que todos apresentam forte produção ofensiva, com destaque para o elevado número de gols, assistências e valores de *expected goals* (xG).

Tabela 8.2 – Estatísticas ofensivas dos jogadores sugeridos pelos algoritmos de otimização

Jogador	Idade	Gols	Assistências	G+A	xG	xAG
Luis Muriel	29	22	7	29	15,5	4,5
Romelu Lukaku	27	24	11	35	23,0	7,0
Zlatan Ibrahimović	38	15	2	17	16,2	1,8

Fonte: Elaborado pelo autor.

Observa-se que os três jogadores sugeridos pelo algoritmo atuam predominantemente em posições ofensivas. Esse resultado pode estar relacionado à própria função objetivo utilizada no processo de otimização, uma vez que o modelo de AM pode atribuir maior peso a métricas associadas à produção ofensiva na previsão da pontuação final das equipes. Dessa forma, o algoritmo tende a priorizar jogadores com forte impacto em ações de ataque.

Esse comportamento ilustra como a natureza da função objetivo influencia diretamente as soluções encontradas pelos algoritmos de otimização. Assim, melhorias futuras poderiam considerar a inclusão de restrições adicionais relacionadas à estrutura posicional do elenco ou a incorporação de métricas defensivas mais detalhadas no modelo preditivo.

Embora esse experimento inicial permita observar diferenças claras no comportamento dos algoritmos, a presença de componentes aleatórias no processo de busca pode influenciar os resultados obtidos em uma única execução. Por esse motivo, na próxima seção são apresentados os resultados de múltiplas execuções independentes de cada algoritmo, permitindo uma análise estatística mais robusta do desempenho dos métodos.

8.2 Avaliação Experimental com Múltiplas Execuções

Com o objetivo de obter uma análise mais robusta do desempenho dos algoritmos, foram realizadas seis execuções independentes para cada método considerado. Em cada execução, foram registradas métricas relacionadas à qualidade da solução e ao custo computacional do processo de busca, incluindo tempo de execução, número de iterações, número de avaliações da função objetivo e pontuação final da solução obtida.

A Tabela 8.3 apresenta um resumo estatístico dos resultados, incluindo média e desvio padrão das métricas analisadas para cada algoritmo.

Tabela 8.3 – Resumo estatístico dos resultados após 6 execuções

Algoritmo	Tempo (s)	σ_T	Iterações	Avaliações	Score	σ_S
Guloso First-Improvement	257,48	6,35	4,0	1571,0	72,55	0,00
Guloso Best-Improvement	479,23	15,87	3,0	2809,0	72,53	0,000
Busca Local First-Improvement	2016,45	46,79	14,0	10538,0	74,66	0,00
Busca Local Best-Improvement	1657,46	53,45	3,0	8334,0	74,66	0,00

Fonte: Elaborado pelo autor.

Na comparação entre as heurísticas gulosas, observa-se um *trade-off* bastante claro entre qualidade da solução e custo computacional. O algoritmo Guloso Best-Improvement foi ligeiramente inferior na pontuação média em relação ao Guloso First-Improvement, obtendo 72,53 pontos previstos, enquanto o Guloso First-Improvement alcançou média de 72,55 pontos. Essa pequena diferença na qualidade da solução foi acompanhada por um custo computacional substancialmente superior no método Best-Improvement.

Enquanto o Guloso First-Improvement apresentou tempo médio de execução de 257,48 segundos e média de 1571 avaliações da função objetivo, o Guloso Best-Improvement demandou, em média, 479,23 segundos e 2809 avaliações. Como ambos apresentaram média de aproximadamente 3 a 4 iterações, a diferença de desempenho computacional está diretamente associada à estratégia de exploração adotada. No caso do método Best-Improvement, a busca exaustiva pela melhor alternativa local em cada etapa elevou significativamente o número de avaliações realizadas.

Observa-se ainda que as heurísticas gulosas apresentaram desvio padrão nulo na pontuação final, indicando comportamento determinístico nas execuções realizadas. Isso ocorre porque, dado o mesmo conjunto de candidatos e a mesma função objetivo, o processo de construção da solução tende a conduzir sempre ao mesmo resultado no cenário experimental considerado. De forma semelhante, os algoritmos de busca local também não apresentaram variabilidade na pontuação final, indicando convergência consistente para a mesma solução nas diferentes execuções.

No caso dos algoritmos de busca local, observa-se comportamento distinto em relação às heurísticas gulosas. O método Busca Local Best-Improvement apresentou menor tempo médio de execução (1657,46 s) quando comparado ao Busca Local First-Improvement (2016,45 s), além de exigir menor número médio de avaliações da função objetivo (8334 contra 10538).

Esse resultado sugere que a estratégia Best-Improvement foi capaz de identificar movimentos de melhoria mais eficazes em cada etapa da busca, reduzindo o número total de iterações necessárias até a convergência. Em termos de qualidade da solução, ambos os métodos apresentaram desempenho idêntico, com pontuação média de 74,66 pontos.

De forma geral, os resultados indicam que o impacto da estratégia de exploração (first-improvement ou best-improvement) depende do tipo de heurística em que ela é empregada. Nas heurísticas gulosas, a estratégia Best-Improvement não trouxe ganhos de qualidade e implicou maior custo computacional. Já na busca local, a mesma estratégia apresentou maior eficiência computacional, reduzindo o número de iterações e avaliações da função objetivo, sem diferenças na qualidade das soluções obtidas.

Embora os experimentos apresentados tenham permitido comparar o desempenho dos algoritmos no cenário considerado, o comportamento das heurísticas também pode ser influenciado pela restrição orçamentária disponível para contratações. Em problemas de otimização aplicados ao contexto esportivo, o orçamento representa uma das principais limitações práticas enfrentadas pelos clubes, afetando diretamente o conjunto de jogadores candidatos e, consequentemente, o espaço de soluções explorado pelos algoritmos. Dessa forma, torna-se relevante investigar como diferentes níveis de orçamento impactam tanto a qualidade das soluções encontradas quanto o desempenho computacional dos métodos propostos. Na próxima seção, é realizada uma análise de sensibilidade em relação à restrição orçamentária, avaliando o comportamento dos algoritmos sob diferentes limites financeiros disponíveis para montagem do elenco.

8.3 Análise de Sensibilidade à Restrição Orçamentária

Para analisar o impacto da restrição orçamentária no comportamento dos algoritmos, foi realizado um experimento adicional considerando um cenário com orçamento significativamente menor. Nesse caso, utilizou-se o elenco do Cagliari na mesma temporada analisada anteriormente, considerando um orçamento disponível de 60.000.000 € para contratações.

Esse cenário representa uma situação mais restritiva do ponto de vista financeiro, típica de clubes com menor capacidade de investimento. Dessa forma, o conjunto de jogadores candidatos torna-se mais limitado, o que pode alterar a dinâmica de exploração do espaço de soluções pelos algoritmos.

Antes da aplicação dos algoritmos de otimização, a pontuação prevista base do Cagliari para essa temporada, estimada pelo modelo de AM, foi de 41,17 pontos. O objetivo dos algoritmos, nesse contexto, consiste em selecionar jogadores adicionais que maximizem essa pontuação prevista dentro da restrição orçamentária estabelecida.

A Tabela 8.4 apresenta os resultados obtidos após uma execução de cada algoritmo nesse cenário.

Tabela 8.4 – Resultados dos algoritmos para o Cagliari com orçamento de 60.000.000 €

Algoritmo	Tempo (s)	Iterações	Avaliações	Melhoras	Score final
Guloso First-Improvement	3,11	5	21	–	41,32
Guloso Best-Improvement	302,47	2	1838	–	41,33
Busca Local First-Improvement	822,43	12	4876	11	41,98
Busca Local Best-Improvement	639,45	5	3741	4	42,05

Fonte: Elaborado pelo autor.

Observa-se que todos os algoritmos foram capazes de melhorar a pontuação prevista da equipe em relação ao valor base de 41,17 pontos. Nesse cenário mais restritivo, os algoritmos de busca local apresentaram melhor desempenho em termos de qualidade da solução. O método Busca Local Best-Improvement obteve a maior pontuação final, alcançando 42,05 pontos previstos.

Esse resultado corresponde a um ganho aproximado de 0,88 ponto em relação à configuração original do elenco, indicando que, mesmo sob restrições financeiras mais severas, a otimização ainda é capaz de identificar combinações de jogadores capazes de melhorar o desempenho esperado da equipe.

Esse resultado é consistente com o experimento anterior realizado com o Napoli, no qual os algoritmos de Busca Local também apresentaram o melhor desempenho. Uma possível explicação para esse comportamento está relacionada ao tamanho e à estrutura do espaço de soluções. Em cenários com maior disponibilidade financeira e maior número de jogadores candidatos, heurísticas construtivas podem ser suficientes para identificar soluções de alta qualidade. Entretanto, quando o orçamento é mais restrito, a exploração iterativa do espaço de soluções realizada pelos métodos de busca local tende a se tornar ainda mais eficaz para identificar combinações vantajosas de jogadores dentro das limitações impostas.

Os resultados apresentados neste capítulo demonstram o potencial da abordagem proposta para a otimização da composição de elencos a partir das previsões obtidas por modelos de AM. A análise realizada permitiu avaliar o comportamento dos algoritmos de otimização e os ganhos potenciais associados às contratações sugeridas. No capítulo seguinte, são apresentadas as conclusões deste trabalho, bem como possíveis direções para pesquisas futuras.

9 CONCLUSÕES

Este trabalho apresentou uma abordagem integrada de AM e Pesquisa Operacional PO para apoiar o processo de montagem de elencos no futebol profissional. Inicialmente, foi desenvolvido um modelo preditivo capaz de estimar o desempenho esperado de equipes a partir das características individuais dos jogadores. Os resultados obtidos indicam que o modelo proposto foi capaz de explicar aproximadamente 55% da variabilidade da pontuação final dos clubes, evidenciando sua capacidade de capturar padrões relevantes nos dados e produzir estimativas consistentes de desempenho coletivo.

A partir dessas previsões, o modelo foi utilizado como função objetivo em um problema de otimização voltado à seleção de jogadores sob restrições orçamentárias. Para explorar o espaço de soluções, foram implementadas duas heurísticas: um algoritmo guloso baseado em substituições iterativas e um método de busca local.

Do ponto de vista prático, os resultados obtidos indicam que a abordagem proposta pode contribuir para tornar o processo de tomada de decisão na montagem de elencos mais estruturado e orientado por dados. No contexto do futebol brasileiro, onde decisões de contratação frequentemente são influenciadas por fatores subjetivos ou avaliações intuitivas, ferramentas baseadas em análise quantitativa podem oferecer suporte adicional à identificação de jogadores que se encaixem de forma mais eficiente em determinadas estruturas de equipe, respeitando restrições financeiras.

Assim, este trabalho demonstra que a integração entre técnicas de AM e métodos de PO representa um caminho promissor para apoiar processos de recrutamento e planejamento esportivo. Ao transformar dados de desempenho em estimativas de impacto coletivo e utilizá-las em modelos de decisão, torna-se possível reduzir a dependência exclusiva da intuição na escolha de atletas e avançar em direção a abordagens mais analíticas na gestão do futebol. Nesse sentido, os resultados apresentados reforçam o potencial da integração entre Aprendizado de Máquina e Pesquisa Operacional como uma abordagem promissora para transformar dados de desempenho esportivo em ferramentas quantitativas de apoio à tomada de decisão, contribuindo para o avanço de métodos analíticos aplicados à gestão e ao planejamento no futebol profissional.

9.1 Trabalhos Futuros

Como continuidade desta pesquisa, algumas direções podem ser exploradas em trabalhos futuros. Uma possibilidade consiste no aprimoramento dos modelos de AM, buscando capturar de forma ainda mais precisa a relação entre o desempenho individual dos jogadores e o desempenho coletivo das equipes. A incorporação de novas variáveis explicativas, como características táticas das equipes, métricas de interação entre jogadores ou informações contextuais das partidas, pode contribuir para aumentar a capacidade preditiva dos modelos.

Outra linha de investigação envolve o desenvolvimento de métodos de otimização mais avançados para o problema de montagem de elencos, incluindo metaheurísticas ou abordagens híbridas capazes de explorar o espaço de soluções de maneira mais eficiente. Além disso, futuras pesquisas podem considerar restrições adicionais relevantes ao contexto real do futebol profissional, como limitações contratuais, regras de competições e requisitos específicos de formação de elenco.

Por fim, a ampliação da base de dados utilizada, incorporando informações de outras ligas e temporadas, pode contribuir para tornar os modelos mais robustos e generalizáveis, ampliando o potencial de aplicação prática da abordagem proposta.

REFERÊNCIAS

ABIDIN, D. A case study on player selection and team formation in football with machine learning. **Turkish Journal of Electrical Engineering and Computer Sciences**, v. 29, n. 3, 2021. Disponível em: https://journals.tubitak.gov.tr/elektrik/vol29/iss3/23.

ALMULLA, J. *et al.* Soccernet: A gated recurrent unit-based model to predict soccer match winners. **PLoS ONE**, ago. 2023. Disponível em: https://doi.org/10.1371/journal.pone.0288933.

AWADALLAH, A. A.; KHANDELWAL, R. Football match prediction using deep learning (recurrent neural network). 2020. CS230 Deep Learning Project. Disponível em: https://github.com/raghav68/CS230DLProject.

BABOOTA, R.; KAUR, H. Predictive analysis and modelling football results using machine learning approach for english premier league. **International Journal of Forecasting**, v. 35, n. 2, p. 741–755, 2019. Disponível em: https://doi.org/10.1016/j.ijforecast.2018.01.003.

BBC Sport. **Moises Caicedo transfer news: Chelsea agree £115m deal for Brighton midfielder**. 2023. Accessed: 28 Jan. 2026. Disponível em: https://www.bbc.com/sport/football/66491106.

Beautiful Soup Documentation. **Beautiful Soup Documentation**. 2024. Acessado em: 10 fev. 2025. Disponível em: https://beautiful-soup-4.readthedocs.io/en/latest/.

BECKER, A.; SUN, X. A. An analytical approach for fantasy football draft and lineup management. **Journal of Quantitative Analysis in Sports**, De Gruyter, v. 11, n. 1, p. 13–27, 2015.

BERTSIMAS, D.; KALLUS, N. From predictive to prescriptive analytics. **Management Science**, INFORMS, v. 65, n. 3, p. 1025–1044, 2019.

BISHOP, C. M. **Pattern Recognition and Machine Learning**. New York: Springer, 2006. (Information Science and Statistics). ISBN 978-0-387-31073-2.

BOON, B. H.; SIERKSMA, G. Team formation: Matching quality supply and quality demand. **European Journal of Operational Research**, Elsevier, v. 148, n. 1, p. 71–82, 2003.

BREFELD, U. *et al.* (Ed.). **Machine Learning and Data Mining for Sports Analytics**: 10th international workshop, mlsa 2023, turin, italy, september 18, 2023, revised selected papers. [*S.l.*]: Springer Nature Switzerland, 2024. v. 2035. (Communications in Computer and Information Science, v. 2035). ISBN 978-3-031-53832-2.

CARMICHAEL, F.; MCHALE, I.; THOMAS, D. Maintaining market position: Team performance, revenue and wage expenditure in the english premier league. **Bulletin of Economic Research**, v. 63, n. 4, 2011.

CHMAIT, N.; WESTERBEEK, H. Artificial intelligence and machine learning in sport research: An introduction for non-data scientists. **Frontiers in Sports and Active Living**, v. 3, p. 682287, 2021. Disponível em: https://doi.org/10.3389/fspor.2021.682287.

CLARKE, S. R.; NORMAN, J. M. When to rush a behind in australian rules football: A dynamic programming approach. **Journal of the Operational Research Society**, Palgrave Macmillan, v. 49, n. 5, p. 530–536, 1998.

CNN Brasil. **Moisés Caicedo no Chelsea: de desconhecido a jogador mais caro do futebol inglês**. 2023. Accessed: 28 Jan. 2026. Disponível em: https://www.cnnbrasil.com.br/esportes/futebol/futebol-internacional/moises-caicedo-no-chelsea-de-desconhecido-a-jogador-mais-caro-do-futebol-ingles/.

CORMEN, T. H. *et al.* **Introduction to Algorithms**. 3rd. ed. Cambridge, MA: The MIT Press, 2009. ISBN 978-0262033848.

CROES, G. A. A method for solving traveling-salesman problems. **Operations Research**, v. 6, n. 6, p. 791–812, 1958.

DANTAS, G. P.; RITT, M. Squad optimizing: Integrating tactics into football team composition problem via integer programming. **Environment Systems and Decisions**, Springer, v. 45, p. 36, 2025.

DERRAC, J. *et al.* A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. **Swarm and Evolutionary Computation**, Elsevier, v. 1, n. 1, p. 3–18, 2011. ISSN 2210-6502.

FBref. **FBref: Football Statistics and History**. 2026. Acesso em: 2024. Disponível em: https://fbref.com/.

FRICK, B. The football players' labor market: Empirical evidence from the major european leagues. **Scottish Journal of Political Economy**, Blackwell Publishing, v. 54, n. 3, p. 422–446, 2007.

GENDREAU, M.; POTVIN, J.-Y. (Ed.). **Handbook of Metaheuristics**. 2. ed. New York, NY: Springer, 2010. (International Series in Operations Research & Management Science).

GERCHAK, Y. Operations research in sports. In: **Handbooks in Operations Research & Management Science**. [*S.l.*]: Elsevier Science B.V., 1994. v. 6, p. 507–512.

GLOBO Esporte. **Mercado de transferências bate recorde em 2023: R$ 36,6 bilhões**. 2023. Acesso em: 11 fev. 2025. Disponível em: https://ge.globo.com/futebol/futebol-internacional/noticia/2023/09/08/mercado-de-transferencias-bate-recorde-em-2023-r-366-bilhoes.ghtml.

GLOBO Esporte. **Janela da Série A bate marca de R$ 1 bilhão em reforços; veja quem mais gastou**. 2025. Acesso em: 24 fev. 2025. Disponível em: https://ge.globo.com/espiao-estatistico/noticia/2025/02/06/janela-da-serie-a-bate-marca-de-r-1-bilhao-em-reforcos-veja-quem-mais-gastou.ghtml.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. MIT Press, 2016. ISBN 978-0-262-03561-3. Disponível em: https://www.deeplearningbook.org.

GRAHAM, I. **How to Win the Premier League**. [*S.l.*]: Century, 2024.

GUYON, I. *et al.* (Ed.). **Feature Extraction: Foundations and Applications**. Berlin Heidelberg: Springer, 2006. v. 207. (Studies in Fuzziness and Soft Computing, v. 207). ISBN 978-3-540-35487-1.

GéRON, A. **Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow**. 2nd. ed. [S.l.]: O'Reilly Media, 2019.

HEROLD, M.; MEMMERT, D.; PERL, J. Machine learning in men's professional football: Current applications and future directions for improving attacking play. **International Journal of Sports Science & Coaching**, SAGE Publications, v. 14, n. 6, p. 798–817, 2019.

HILLIER, F. S.; LIEBERMAN, G. J. **Introduction to Operations Research**. 7. ed. New York: McGraw-Hill, 2001. ISBN 978-0072416186.

HIROTSU, N.; WRIGHT, M. Using a markov process model of an association football match to determine the optimal timing of substitution and tactical decisions. **Journal of the Operational Research Society**, v. 53, p. 88–96, 2002.

HORNIK, K. Approximation capabilities of multilayer feedforward networks. **Neural Networks**, v. 4, p. 251–257, 1991.

HORVAT, T.; JOB, J. The use of machine learning in sport outcome prediction: A review. **WIREs Data Mining and Knowledge Discovery**, Wiley, v. 10, n. 6, p. e1382, 2020.

JORDAN, M. I.; MITCHELL, T. M. Machine learning: Trends, perspectives, and prospects. **Science**, v. 349, n. 6245, p. 255–260, 2015.

JOSEPH, A.; FENTON, N. E.; NEIL, M. Predicting football results using bayesian nets and other machine learning techniques. **Knowledge-Based Systems**, v. 19, 2006.

KELLEHER, J. D.; NAMEE, B. M.; D'ARCY, A. **Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies**. Cambridge, MA: MIT Press, 2015. ISBN 978-0262029445.

KENDALL, G. *et al.* Scheduling in sports: An annotated bibliography. **Computers & Operations Research**, Elsevier, v. 37, n. 1, p. 1–19, 2010.

KENDALL, G.; PARKES, A.; SPOERER, K. A survey of np-complete scheduling problems. **Journal of Scheduling**, Springer, v. 13, n. 3, p. 257–276, 2010.

KITE, C. S.; NEVILL, A. The predictors and determinants of inter-seasonal success in a professional soccer team. **Journal of Human Kinetics**, v. 58, p. 157–167, 2017.

LAGO-BALLESTEROS, J.; LAGO-PEñAS, C. Performance in team sports: Identifying the keys to success in soccer. **Journal of Human Kinetics**, v. 25, p. 85–91, 2010.

LIU, H. *et al.* Feature selection: An ever evolving frontier in data mining. In: LAWRENCE, N. (Ed.). **JMLR: Workshop and Conference Proceedings - The Fourth Workshop on Feature Selection in Data Mining (FSDM 2010)**. [*S.l.*]: JMLR Workshop and Conference Proceedings, 2010. v. 10, p. 4–13.

MAHESH, B. Machine learning algorithms - a review. **International Journal of Science and Research (IJSR)**, v. 9, p. 381–386, jan. 2020. Disponível em: https://doi.org/10.21275/ART20203995.

MEDRI, W.; YOTSUMOTO, A. **Pesquisa Operacional na Tomada de Decisão**. [*S.l.: s.n.*], 2009.

MONTGOMERY, D. C.; PECK, E. A.; VINING, G. G. **Introduction to Linear Regression Analysis**. 5. ed. [*S.l.*]: Wiley, 2012.

MÜLLER, O.; SIMONS, A.; WEINMANN, M. Beyond crowd judgments: Data-driven estimation of market value in association football. **European Journal of Operational Research**, Elsevier, v. 263, n. 1, p. 273–287, 2017.

OLLEY, J. **How Brighton's transfer mastery broke Premier League profit record**. ESPN, 2023. Acesso em: 11 fev. 2025. Disponível em: https://www.espn.com/soccer/story/_/id/40069726/how-brightons-transfer-mastery-broke-premier-league-profit-record.

PANTUSO, G. The football team composition problem: A stochastic programming approach. **Journal of Quantitative Analysis in Sports**, 2017.

QADER, M. A. *et al.* A methodology for football players selection problem based on multi-measurements criteria analysis. **Measurement**, v. 111, p. 38–50, 2017.

RIBEIRO, C. C.; URRUTIA, S. Heuristics for the mirrored traveling tournament problem. **European Journal of Operational Research**, Elsevier, v. 179, n. 3, p. 775–787, 2007.

RICO-GONZÁLEZ, M. *et al.* Machine learning application in soccer: a systematic review. **Biology of Sport**, Institute of Sport, v. 40, n. 1, p. 249–263, 2023.

RODRIGUES, F.; PINTO, A. Prediction of football match results with machine learning. **Procedia Computer Science**, v. 204, p. 463–470, jan. 2023. Disponível em: https://doi.org/10.1016/j.procs.2022.08.057.

TAVANA, M. *et al.* A fuzzy inference system with application to player selection and team formation in multi-player sports. **Sport Management Review**, v. 16, p. 97–110, 2013.

TOTTY, E. S.; OWENS, M. F. Salary caps and competitive balance in professional sports leagues. **Journal for Economic Educators**, v. 11, n. 2, p. 46–56, 2011.

WRIGHT, M. B. 50 years of or in sport. **Journal of the Operational Research Society**, Palgrave Macmillan, v. 60, n. S1, p. S161–S168, 2009.