Análise de Atributos e Previsão de Desempenho de Jogadores Novatos na NBA Utilizando Métodos de Boosting

Alexandre Santos Marques

CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

João Pessoa, PB

2023

Análise de Atributos e Previsão de Desempenho de Jogadores Novatos na NBA Utilizando Métodos de Boosting

Monografia apresentada ao curso Engenharia de Computação
do Centro de Informática, da Universidade Federal da Paraíba,
como requisito para a obtenção do grau de Bacharel em Engenharia de Computação

Orientador: Yuri de Almeida Malheiros Barbosa

Junho de 2023

Catalogação na publicação
Seção de Catalogação e Classificação

M357a Marques, Alexandre Santos.
Análise de atributos e previsão de desempenho de jogadores novatos na NBA utilizando métodos de boosting / Alexandre Santos Marques. - João Pessoa, 2023.
44 f. : il.

Orientação: Yuri de Almeida Malheiros Barbosa.
Coorientação: Thaís Gaudencio do Rêgo.
TCC (Graduação) - UFPB/CI.

1. NBA. 2. Métodos de boosting. 3. Aprendizado de máquina. 4. Draft. I. Barbosa, Yuri de Almeida Malheiros. II. Rêgo, Thaís Gaudencio do. III. Título.

UFPB/CI CDU 004.832

Elaborado por Michelle de Kássia Fonseca Barbosa - CRB-738

CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

Trabalho de Conclusão de Curso de Engenharia de Computação intitulado ***Análise de Atributos e Previsão de Desempenho de Jogadores Novatos na NBA Utilizando Métodos de Boosting*** de autoria de Alexandre Santos Marques, aprovada pela banca examinadora constituída pelos seguintes professores:

Yuri de Almeida Malheiros Barbosa

Prof. Dr. Yuri de Almeida Malheiros Barbosa
Universidade Federal da Paraíba

Thaís Gaudencio do Rêgo

Prof. Dr. Thaís Gaudencio do Rêgo
Universidade Federal da Paraíba

Telmo de Menezes e Silva Filho

Prof. Dr. Telmo De Menezes E Silva Filho
Universidade Federal da Paraíba

João Pessoa, 27 de junho de 2023

Centro de Informática, Universidade Federal da Paraíba
Rua dos Escoteiros, Mangabeira VII, João Pessoa, Paraíba, Brasil CEP: 58058-600
Fone: +55 (83) 3216 7093 / Fax: +55 (83) 3216 7117

AGRADECIMENTOS

Primeiramente, gostaria de expressar minha profunda gratidão aos meus pais, Telca e Irenaldo, por terem proporcionado uma educação de qualidade e por seu apoio incondicional ao longo de minha jornada para alcançar meus objetivos. Também gostaria de agradecer ao meu irmão, Adonildo, e à minha irmã gêmea, Amanda, por estarem sempre ao meu lado, me apoiando incondicionalmente.

Agradeço à minha noiva e futura esposa, Diane, por estar ao meu lado nos últimos 9 anos, fornecendo todo o suporte, carinho e amor durante os momentos bons e difíceis da minha vida. Que me motivou e me cobrou ao longo dos últimos três anos para que eu finalizasse este TCC.

Agradeço aos amigos que fiz durante a graduação, José Olívio, Ícaro e Bruno. E em especial os amigos que levo até hoje e levarei para a vida toda, Adriano, Arthur, José Eugênio, Juan, Suanny e Kalil. Vocês fizeram minha vida na universidade ser muito mais fácil.

Um agradecimento especial para Thaís Gaudêncio, por ter aceitado participar da orientação dessa pesquisa, por mais deslizes que eu tenha dado durante todo esse tempo. Se não fosse ela, esse TCC não teria sido concluído. Agradeço também aos professores Telmo Filho e Yuri Malheiros, por toda ajuda para que pudesse chegar finalmente na conclusão deste trabalho.

RESUMO

O uso de técnicas de aprendizado de máquina no setor esportivo traz diversos benefícios para o esporte, um deles é o auxílio em identificar e encontrar os melhores atributos para determinar bons jogadores. Este trabalho aborda o uso de técnicas de aprendizado de máquina no contexto esportivo, mais especificamente na NBA, com o objetivo de identificar jogadores com *Win Shares per* 48 acima da média e os atributos mais importantes. A justificativa para este estudo é a importância do desempenho dos jogadores na NBA e a necessidade de identificar quais características são mais relevantes para o sucesso dos jogadores que participaram do *draft*. Utilizando dados do basquete universitário e da NBA, foram testados cinco métodos de classificação do tipo *boosting*, sendo o CatBoost o que obteve melhores resultados, alcançando 92% de acurácia. Os atributos mais relevantes para a classificação foram pontos por jogo, rebotes por jogo e assistências por jogo. O estudo contribui para a compreensão dos fatores que influenciam o desempenho dos jogadores na NBA, fornecendo outra perspectiva valiosa para equipes e treinadores na seleção e desenvolvimento de jogadores. Trabalhos futuros podem explorar o uso exclusivo de dados universitários para prever o desempenho dos jogadores antes de sua seleção.

Palavras-chave: <NBA>, <Métodos *Boosting*>, <Aprendizado de máquina>, <*Draft*>.

ABSTRACT

The use of machine learning techniques in the sports industry brings several benefits to the sport, one of which is aiding in the identification and discovery of the best attributes to determine good players. This study addresses the use of machine learning techniques in the sports context, specifically in the NBA, with the goal of identifying players with above-average Win Shares per 48 (WS/48) and determining the most important attributes. The justification for this study lies in the importance of player performance in the NBA and the need to identify which characteristics are more relevant for the success of players who participated in the draft. Utilizing data from college basketball and the NBA, five boosting-based classification methods were tested, with CatBoost achieving the best results, reaching 92% accuracy. The most relevant attributes for classification were points per game, rebounds per game, and assists per game. This study contributes to understanding the factors that influence player performance in the NBA, providing valuable insights for teams and coaches in player selection and development. Future work may explore the exclusive use of college data to predict player performance prior to their selection.

Key-words: <NBA>, <Boosting classification methods>, <Machine learning>, <*Draft*>.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS

ABL - Liga Americana de Basquete (do inglês, *American Basketball League*)

AM - Aprendizado de Máquina

BAA - Associação de Basquetebol da América (do inglês, *Basketball Association of America*)

EUA - Estados Unidos da América

EFB - Pacote de Recursos Exclusivos (do inglês, *Exclusive Feature Bundling*)

FN - Falso Negativo

FP - Falso Positivo

GBM - Máquinas de Aumento de Gradiente (do inglês, *Gradient Boosting Machines*)

GOSS - Amostragem Unilateral Baseada em Gradiente (do inglês, *Gradient-based One-Side Sampling*)

GPU - Unidade de processamento gráfico (do inglês, *Graphics Processing Unit*)

NBA - Associação Nacional de Basquetebol (do inglês, *National Basketball Association*)

NBL - Liga Nacional de Basquete (do inglês, *National Basketball League*)

NCAA - Associação Atlética Universitária Nacional (do inglês, *National Collegiate Athletic Association*)

PAC - Provavelmente Aproximadamente Correto (do inglês, *Probably Approximately Correct*)

PCA - Análise de Componentes Principais (do inglês, *Principal Component Analysis*)

PER - Classificação de Eficiência do Jogador (do inglês, *Player efficiency rating*)

VN - Verdadeiro Negativo

VP - Verdadeiro Positivo

WS - Participação em Vitórias (do inglês, *Win Shares*)

WS/48 - Participação em Vitórias por 48 minutos (do inglês, *Win Shares per 48*)

Sumário

1 INTRODUÇÃO

Nos últimos anos, presenciamos uma significativa revolução tecnológica impulsionada pela criação e disseminação de aplicações baseadas em técnicas de Aprendizado de Máquina (AM). Essa revolução tem transformado o cotidiano das pessoas, alterando a forma como trabalham e se comunicam. À medida que mais pessoas se conectam digitalmente, essas tecnologias avançam rapidamente, impulsionadas pela constante geração e armazenamento de dados. Esses dados constituem a base para o desenvolvimento de soluções que empregam o AM, permitindo resolver uma ampla gama de problemas.

Uma das áreas que se beneficia significativamente dessas técnicas é o setor esportivo, que possui uma enorme quantidade de dados. Diariamente, esses dados são gerados em grande escala. Somente na Associação Nacional de Basquetebol (do inglês, *National Basketball Association* - NBA), por exemplo, são disputados mais de 1230 jogos por ano [31]. A NBA é uma liga renomada mundialmente, onde jogadores competem em um alto nível de desempenho. Essa intensidade e competitividade geram uma grande quantidade de dados estatísticos e de desempenho, incluindo estatísticas de jogadores, resultados de partidas, informações de desempenho individual e coletivo, entre outros.

O AM oferece métodos capazes de interpretar esses dados complexos da NBA, auxiliando desde analistas esportivos na previsão de resultados de jogos e identificação de jogadores com potencial para se tornarem estrelas, até aplicações na medicina esportiva, por meio de análise de dados clínicos. Com o avanço das técnicas de AM, é possível extrair informações valiosas dos dados da NBA, possibilitando uma compreensão mais profunda do jogo, identificação de padrões e tomada de decisões estratégicas mais embasadas.

O *Win Shares per* 48 (WS/48) é uma medida de desempenho amplamente utilizada na NBA para avaliar o rendimento dos jogadores. A retenção de jogadores de alto nível é essencial para construir equipes competitivas. Jogadores com um histórico consistente de alto desempenho têm maiores chances de se destacar, e certas características são indicativas de sucesso. Por meio da aplicação de técnicas de AM, é possível prever jogadores que provavelmente terão um WS/48 acima da média.

1.1 Definição do Problema

Todo ano na NBA, os times enfrentam um período crucial conhecido como *draft*, no qual têm a oportunidade de selecionar jogadores novatos para seus times. A escolha feita pelas equipes nesse momento é de extrema importância para garantir a manutenção de uma equipe competitiva no futuro. No entanto, essa decisão não é trivial, uma vez que requer uma análise criteriosa de diversos fatores, como as habilidades individuais dos jogadores, seu desempenho anterior, estatísticas relevantes e outros aspectos relacionados

ao seu potencial de sucesso na liga.

Diante do exposto, os seguintes questionamentos foram levantados:

- Como identificar quais jogadores novatos têm maior probabilidade de ter um WS/48 acima da média?
- Quais características devem ser consideradas para avaliar o potencial dos jogadores?

1.2 Objetivo geral

Neste trabalho é proposto um método de previsão capaz de determinar as características mais importantes para identificar jogadores com WS/48 acima da média na NBA, a partir de dados referentes ao tempo de atuação na faculdade e NBA. Para este problema, são utilizadas técnicas de Aprendizado de Máquina, especificamente, algoritmos de impulsionamento (do inglês, *boosting*).

1.3 Objetivos específicos

- Desenvolver uma base de dados com estatísticas da faculdade e NBA.
- Treinar e avaliar classificadores para prever se um jogador terá um WS/48 acima da média na NBA.
- Analisar quais os atributos mais importantes no conjunto de dados para a classificação.
- Usar os métodos de *boosting* GBM, XGBoost, LightGBM, CatBoost e AdaBoost

1.4 Estrutura da monografia

O trabalho está organizado da seguinte forma:

(i) Capítulo 2: São apresentados os conceitos teóricos necessários para o entendimento do trabalho.

(ii) Capítulo 3: É realizada uma breve revisão da literatura.

(iii) Capítulo 4: Neste capítulo são abordados os passos para chegar na solução proposta, explicitando os métodos e parâmetros necessários para a replicação dos resultados.

(iv) Capítulo 5: São expostos e discutidos os resultados a partir dos métodos utilizados.

(v) Capítulo 6: Conclusões obtidas a partir da solução e resultados obtidos são apresentadas, levando em conta o problema abordado. São mostrados as limitações e finalizado com a discussão sobre trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão abordados os conceitos teóricos necessários para o desenvolvimento deste trabalho. O capítulo inicia apresentando o cenário esportivo da NBA, abordando como é feita a seleção de novos jogadores para o time e por onde eles surgem. Em seguida, são apresentados os métodos de aprendizado de máquina do tipo *Boosting*, mostrando conceitos e debatendo sobre suas arquiteturas. Por fim, é apresentado o conceito de engenharia de atributos.

2.1 *National Basketball Association*

A Associação Nacional de Basquetebol (do inglês, *National Basketball Association* - NBA), teve seu início em 1946, mas com o nome de Associação de Basquetebol da América (do inglês, *Basketball Association of America* - BAA), e foi organizada e fundada por empresários do ramo do hóquei. A BAA não foi a primeira liga de basquete profissional nos Estados Unidos da América (EUA), já existindo a Liga Nacional de Basquete (do inglês, *National Basketball League* - NBL) e a Liga Americana de Basquete (do inglês, *American Basketball League* - ABL) [10].

Camargo (2019) afirma que a ABL e NBL eram ligas sem muita visibilidade, que faziam seus jogos em pequenas arenas, até mesmo em ginásios de escolas e universidades. Por outro lado, a BAA possuía grandes arenas oriundas dos times de hóquei. No primeiro ano da liga, os times não conseguiram um impulso significante comercialmente, apesar disso, os times profissionais das outras ligas se sentiram atraídos pelo potencial que os grandes ginásios poderiam trazer. Segundo Camargo (2019), nos anos seguintes os grandes times da NBL e ABL migraram para a BAA e, com isso, em 1949 houve a fusão da BAA com NBL, surgindo, assim, a NBA como é conhecida hoje.

A NBA é a terceira liga esportiva mais rica do mundo, com um faturamento de 8 bilhões de dólares, ficando atrás apenas da Liga Nacional de Futebol (do inglês, *National Football League* - NFL) e da Liga Principal de Beisebol (do inglês, *Major League Baseball* - MBL) [11]. Mesmo com a pandemia da COVID-19 impactando financeiramente os times, reduzindo as receitas que viriam de bilheteria, a NBA, com seus 30 times, continuaram muito atrativos para investidores, fazendo com que os valores dos times, em média, tivessem um aumento de 4% em 2021 [12]. Além de ser a terceira liga esportiva mais rica do mundo, a NBA também valoriza o desempenho individual dos jogadores.

Existem diversas métricas de avaliar o desempenho de um jogador na NBA, como afirmado pelo Fromal [13]. O *Win Shares* (WS) é uma estatística de desempenho que visa dividir o peso da vitória do time entre os atletas da equipe, onde uma vitória equivale a um WS [14]. O *Win Shares per* 48 (WS/48) visa tirar a vantagem dos jogadores que passam

mais tempo em quadra. Seu cálculo pode ser visto na Equação (1) [13], onde 48 é o tempo de duração, em minutos, de uma partida de basquete na NBA. Neste trabalho é utilizado o WS/48, já que, segundo Parker [1], o WS/48 é a métrica mais promissora devido a sua capacidade de comparar a contribuição dos atletas. Segundo Doucette (2021), 0,100 é o valor médio do WS/48, e será utilizado como ponto de corte para definir um jogador acima da média.

$$\text{Win Shares per } 48 = 48 * \frac{\text{Win Shares}}{\text{Minutos Jogados}} \tag{1}$$

Camargo (2019) afirma que a NBA é uma liga que tem como seu pilar central a igualdade de oportunidades para seus times, como por exemplo a distribuição de receitas. Outro exemplo é a manutenção dos times, através da seleção de novos talentos, e é de extrema importância para definir o sucesso e o fracasso dos times nos anos seguintes [10]. Essa seleção de novos jogadores acontece através do *draft*, onde mais detalhes serão descritos na próxima seção.

2.1.1 *Draft* da NBA

O *draft* é um evento que acontece todos anos, e é responsável pelo recrutamento de novos jogadores pelos times da NBA [16]. Segundo Camargo (2019), o primeiro *draft* da NBA (na época era BAA) aconteceu em 1947, através de um acordo com a NBL afim de evitar o leilão entre times e ligas, que ocasionava em altos salários dos calouros. Camargo (2019) também afirma que hoje o *draft* tem como maior objetivo a manutenção da paridade da liga, onde os times com os piores resultados no ano tenham as melhores escolhas.

Todos os 30 times da NBA possuem duas escolhas no *draft*, totalizando duas rodadas. Os 14 piores times possuem as primeiras escolhas, e os outros 16 times ficam com as escolhas seguintes, baseado em suas campanhas na temporada regular [17]. Segundo Camargo (2019), alguns times viram no *draft* a oportunidade de evoluir através da derrota, tendo melhores escolhas. Essa estratégia foi conhecida como *tanking*. Por conta disso, a NBA implantou a "loteria", que consiste em os piores times serem sorteados para saber quais teriam as três primeiras escolhas, os demais escolhas continuariam seguindo a classificação na temporada regular [10].

Segundo Martinelli [16], os jogadores que forem selecionados na primeira rodada do *draft* tem garantido os dois primeiros anos de contrato, podendo ser estendido por mais dois anos, se o time optar por isso. Já os jogadores escolhidos na segunda rodada não são garantidos.

Os atletas elegíveis para o *draft* são chamados de prospectos [16]. Para se eleger

ao *draft* os prospectos precisam seguir alguns critérios, que são: ter no mínimo 19 anos de idade no ano do *draft*; ter no mínimo um ano de afastamento, a partir do término do ensino médio; e atuar no basquete universitário [17]. O foco deste trabalho é em jogadores que atuaram no basquete universitário.

2.1.2 Basquete universitário

Em 1950, o basquete universitário possuía um apelo maior junto ao público do que o profissional, pois estava em atividade há mais tempo, desde que o basquete surgiu nas universidades [10]. O basquete se tornou instantaneamente popular nas faculdades, tendo vários jogadores universitários atuando na equipe olímpica dos EUA na disputa da primeira medalha de ouro do basquete nas olimpíadas [18].

Segundo Schwartz (2011), alguns torneios nacionais começaram a surgir, popularizando cada vez mais o basquete universitário em todo os EUA. Foi então que a maior entidade dos esportes universitários dos EUA, chamada de Associação Atlética Universitária Nacional (do inglês, National Collegiate Athletic Association - NCAA), criou o maior torneio de basquete entre faculdades [19].

A NCAA divide o basquete universitário em três divisões, sendo a primeira divisão a mais forte delas. A primeira divisão inclui as faculdades mais tradicionais dos EUA, totalizando 351 universidades distribuídas em 32 conferências. Essas conferências são ligas menores regionalizadas. Durante o ano, as faculdades disputam vários jogos entre si, sendo os jogos dos torneios da conferência os mais importantes, pois é através dela que as universidades conseguem disputar o torneio nacional da NCAA [19].

O basquete universitário revelou grandes lendas do passado da NBA, como Magic Johnson e Larry Bird, cuja rivalidade teve início na grande final do torneio da NCAA [10]. E também de grandes nomes da NBA atual, como Draymond Green multi campeão pelo *Golden State Warriors*, e Anthony Davis campeão pelo Lakers em 2020 [20].

A utilização do aprendizado de máquina na análise esportiva tem como objetivo obter conclusões úteis para jogadores e times. Uma dessas conclusões é a previsão de talentos emergentes [7]. Na próxima seção, serão apresentados os conceitos de alguns modelos de aprendizado de máquina, todos derivados da metodologia *Boosting*.

2.2 Algoritmos de *Boosting*

Os algoritmos de impulsionamento (do inglês, *Boosting*) consistem em um conjunto de técnicas de aprendizado de máquina que combinam vários modelos de aprendizado fracos para criar um modelo mais forte. Seu objetivo principal é melhorar a precisão preditiva, reduzindo o erro de classificação. O conceito de *Boosting* tem sua origem através

do modelo teórico chamado Provavelmente Aproximadamente Correto (do inglês, *Probably Approximately Correct* - PAC), que foi utilizado para validar a ideia do impulsionamento em direção a um modelo mais forte [21].

Nas seções seguintes, serão apresentados os detalhes de cinco algoritmos do tipo *Boosting* utilizados neste trabalho. Cada algoritmo será abordado nas respectivas seções, fornecendo uma análise mais aprofundada de suas características e funcionalidades.

2.2.1 *Gradient Boosting Machines*

O *Gradient Boosting Machines* (GBM) é uma técnica de aprendizado de máquina que utiliza os conceitos da metodologia *Boosting*. Geralmente, são utilizadas árvores de decisão como modelos fracos para construir um modelo mais forte. O GBM funciona adicionando modelos sequencialmente, em que cada novo modelo é treinado para corrigir os erros do modelo anterior. Esse processo é repetido até que não seja possível reduzir mais o erro ou até que seja atingido o limite máximo de modelos [22].

Árvores de decisão são uma técnica popular de aprendizado de máquina que pode ser usada para classificação ou regressão. A ideia por trás de uma árvore de decisão é particionar o espaço das variáveis de entrada em regiões retangulares homogêneas usando um sistema baseado em regras. Essas regras são representadas por nós na árvore, em que cada nó corresponde a uma condição "se-então"sobre uma variável de entrada específica. Através dos ramos, que conectam os nós, a estrutura da árvore codifica e modela naturalmente as interações entre as variáveis preditoras [23]. Na Figura 1 é possível ver um exemplo de uma árvore de decisão, e como é sua estrutura.

O GBM utiliza a atribuição de pesos diferentes aos exemplos de treinamento, os pesos são valores numéricos que indicam a importância relativa de cada exemplo de treinamento no processo de treinamento do GBM. Eles são usados para ajustar a contribuição de cada exemplo e melhorar o desempenho do modelo. Os pesos são atribuídos a cada exemplo de treinamento com base em sua dificuldade de classificação. Os exemplos que foram classificados incorretamente pelo modelo anterior recebem um peso maior, enquanto os exemplos que foram classificados corretamente recebem um peso menor. Dessa forma, dá-se mais importância aos exemplos que foram classificados incorretamente pelo modelo anterior. Essa abordagem auxilia na garantia de que o modelo final seja capaz de generalizar bem para novos dados, evitando a simples memorização dos dados de treinamento [23].

2.2.2 *Extreme Gradient Boosting*

O *Extreme Gradient Boosting* também conhecido como XGBoost, é uma implementação do algoritmo do GBM, baseado no uso de árvores de decisão. Comparado ao

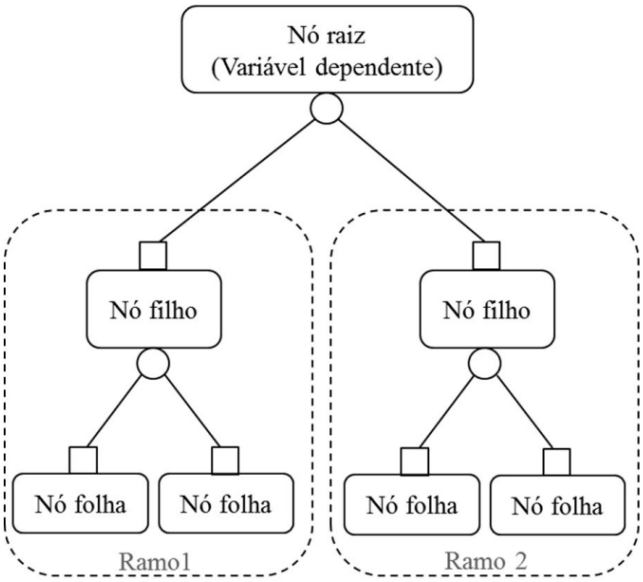


Figura 1: Arquitetura da árvore de decisão. Os quadrados representam os nós de uma árvore de decisão genérica, cada nó possui uma condição e gera dois caminhos baseados na resposta.

Fonte: [32]

GBM, o XGBoost oferece melhorias significativas, como a regularização L1 e L2, um esquema de amostragem estocástica, suporte para processamento distribuído em *clusters* de computadores e capacidade de lidar com dados esparsos e faltantes [24].

Para evitar sobreajuste (do inglês, *overfitting*), é utilizada a técnica de regularização. O XGBoost emprega as regularizações L1 e L2. A regularização L1 promove pesos menores e mais esparsos, ajudando a evitar o *overfitting*, através da adição de uma penalidade proporcional à soma dos valores absolutos dos pesos do modelo. Por outro lado, a regularização L2 adiciona uma penalidade proporcional à soma dos quadrados do peso, incentivando o modelo a ter pesos menores, embora não necessariamente mais esparsos como no caso da L1. Ambas as regularizações contribuem para controlar a complexidade do modelo e reduzir o *overfitting* [24]. Na Figura 2 é possível ver um exemplo do crescimento da árvore de decisão em um modelo XGBoost, onde esses pesos são usados para implementar as técnicas de regularização L1 e L2.

O esquema de amostragem estocástica é uma técnica de amostragem aleatória, aplicada durante a fase de treinamento do modelo, buscando evitar *overfitting* e melhorar a generalização. Em cada etapa, o XGBoost seleciona aleatoriamente um subconjunto das instâncias. Isso ajuda a reduzir a correlação entre as árvores de decisão individuais e aumenta a diversidade do conjunto final de modelos de árvore, o que melhora a robustez e a precisão do modelo final [24].

O XGBoost oferece suporte para processamento distribuído em *clusters* de com-

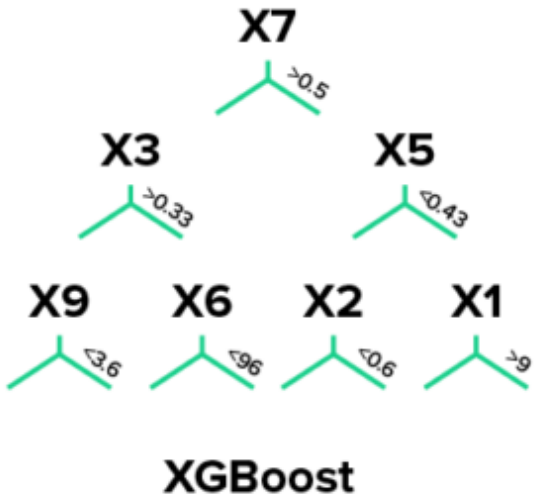


Figura 2: Exemplo de crescimento da árvore de decisão no XGBoost. Os nós são representados pela letra X e um número e, a partir deles, é representada a distribuição dos pesos em um modelo XGBoost. É possível observar ainda que o crescimento da árvore segue por nível.

Fonte: https://www.riskified.com/resources/article/boosting-comparison/

putadores, permitindo que grandes conjuntos de dados sejam rapidamente processados e tenha mais eficiência, pois o algoritmo pode ser executado em vários nós. É utilizada uma arquitetura distribuída baseada em troca de mensagens para coordenar a comunicação entre os nós do *cluster*, garantindo que as tarefas sejam executadas de forma sincronizada e eficiente. O XGBoost também possui armazenamento em cache, permitindo o compartilhamento entre os nós. Todo esse suporte reduz o *overfitting* e acelera o processo de treinamento [24].

2.2.3 *Light Gradient Boosting Machine*

O *Light Gradient Boosting Machine*, também conhecido como LightGBM, é uma outra implementação do algoritmo do GBM baseado no uso de árvores de decisão, proposta pela *Microsoft Research*. Para poder lidar com grandes conjuntos de dados e várias características, ele utiliza duas técnicas: Amostragem Unilateral Baseada em Gradiente (do inglês, *Gradient-based One-Side Sampling* - GOSS) e Pacote de Recursos Exclusivos (do inglês, *Exclusive Feature Bundling* - EFB) [25].

Para lidar com grandes conjuntos de dados em árvores, foi proposta a utilização da técnica GOSS. O GOSS consiste em descartar as instâncias de dados que possuem gradientes pequenos, ou seja, aquelas que já foram bem treinadas e contribuem menos para a melhoria do modelo. Essa abordagem ajuda a otimizar o desempenho e a eficiência do treinamento do modelo, garantindo que a precisão não seja comprometida. Para preservar

a distribuição dos dados e evitar a perda de informações importantes, o GOSS utiliza um método de amostragem unilateral [25]. Na Figura 3 é possível ver um exemplo do crescimento da árvore de decisão em um modelo LightGBM, onde o crescimento não é por nível, mas em folha.

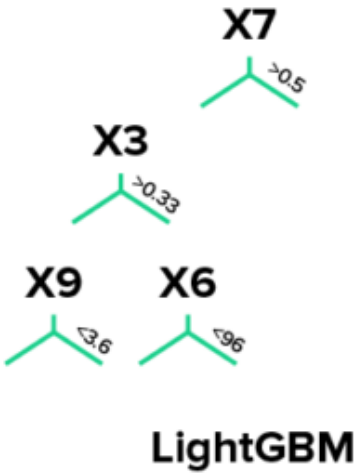


Figura 3: Exemplo de crescimento da árvore de decisão no LightGBM. Os nós são representados pela letra X e um número e, a partir deles, é representada a distribuição dos pesos em um modelo LightGBM. É possível observar ainda que o crescimento da árvore acontece em profundidade a partir de um único nó.
Fonte: https://www.riskified.com/resources/article/boosting-comparison/

Outra técnica utilizada para lidar com grandes conjuntos de dados é o EFB. O EFB consiste em agrupar características exclusivas, ou seja, aquelas que raramente possuem valores não nulos simultaneamente, em um único recurso. Essa abordagem reduz a dimensionalidade dos dados e ajuda a melhorar a eficiência do modelo, ao eliminar características que possuem pouca variabilidade ou relevância para a tarefa de aprendizado [25].

O LightGBM é uma implementação eficiente e escalável, permitindo o treinamento de modelos mais rápidos e com menor consumo de memória em comparação com outros algoritmos GBM. Além disso, ele alcança resultados competitivos em várias tarefas de aprendizado de máquina, como classificação, regressão e ranking. Quando comparado ao XGBoost, o LightGBM se destaca por sua velocidade e eficiência de consumo de memória durante o treinamento. É capaz de lidar com grandes conjuntos de dados e um grande número de características de forma mais eficiente do que o XGBoost [25].

2.2.4 *Categorical Boosting*

O *Categorical Boosting*, também conhecido como CatBoost, é uma outra implementação do algoritmo do GBM. No entanto, as árvores de decisão utilizadas são chamadas de árvores de decisão desatentas (do inglês, *oblivious decision trees*) ou tabelas de decisão (do inglês, *decision tables*), que são menos propensas ao *overfitting*. Isso ocorre porque é utilizado o mesmo critério de divisão em todos os níveis da árvore. Cada nó da árvore é dividido usando o mesmo critério, independente do nó anterior [26].

Uma das técnicas de treinamento utilizada no CatBoost para melhorar a precisão dos modelos é o *ordered boosting*. Nessa técnica, os dados são ordenados com base na sua importância para o aprendizado, e cada árvore é treinada para corrigir os erros cometidos pelas árvores anteriores. A primeira árvore é treinada com o conjunto de dados completo. Em seguida, as instâncias são ordenadas com base nos seus resíduos e a segunda árvore é treinada apenas com os exemplos que possuem maiores resíduos. O processo é repetido para cada árvore seguinte, sendo treinada apenas com as instâncias que as árvores anteriores não conseguiram prever. Por isso essa técnica ajuda a evitar *overfitting* [26].

Para lidar com o problema de vazamento de informações comumente encontrado em muitas implementações de algoritmos de *boosting*, o CatBoost introduziu um algoritmo para processamento de características categóricas. Ele é baseado na técnica conhecida como Codificação de Alvo (do inglês, *Target Encoding*), que ao invés de substituir as categorias por números, utiliza informações do alvo para criar novas características. Essas novas características capturam a relação entre o alvo e as categorias, e são utilizadas pelo modelo para fazer previsões [26].

2.2.5 *Adaptive Boosting*

O *Adaptive Boosting*, mais popularmente conhecido como AdaBoost, é um algoritmo de *boosting* adaptativo que utiliza classificadores fracos simples. São utilizadas árvores de decisão simples ou pilares de decisão (do inglês, *stumps*) como modelos de classificação fracos, com o objetivo de criar um modelo mais robusto e preciso [27].

O AdaBoost ajusta os pesos das instâncias de treinamento em cada iteração com base no desempenho do classificador atual. Na primeira iteração, todas as instâncias possuem o mesmo peso, e nas iterações seguintes, o AdaBoost aumenta o peso das instâncias classificadas incorretamente e diminui o peso das instâncias classificadas corretamente. Isso faz com que o algoritmo leve mais em consideração as instâncias mais difíceis de classificar, permitindo que o modelo final seja mais focado nas amostras que representam um desafio maior [27].

Uma das diferenças do AdaBoost em relação a outros algoritmos de *boosting* é

que ele é adaptativo, o que significa que ajusta automaticamente seus parâmetros com base no desempenho real na iteração atual. Isso permite que o AdaBoost se adapte aos dados e melhore continuamente o seu desempenho ao longo das iterações. Além disso, o AdaBoost é mais sensível a valores discrepantes (do inglês, *outliers*) em comparação com outros algoritmos, pois atribui pesos maiores às instâncias classificadas incorretamente. Isso significa que os *outliers* podem ter uma influência maior no modelo final gerado pelo AdaBoost [27].

O AdaBoost possui algumas limitações, como a sensibilidade a *outliers* e a tendência a ocorrer *overfitting*, quando usado com muitos classificadores fracos, ou quando os dados são muito complexos. No entanto, apesar dessas limitações, o AdaBoost continua sendo um dos algoritmos de aprendizado de máquina mais populares e eficazes para problemas de classificação binária [27].

Para avaliar o desempenho de modelos de classificação, são utilizadas métricas de avaliação específicas. Na próxima seção, serão apresentadas as métricas de avaliação para classificação binária.

2.3 Métricas de Avaliação

As métricas de avaliação em AM são usadas para medir o desempenho dos modelos preditivos, fornecendo uma medida objetiva de quão bem o modelo está funcionando. Essas métricas são essenciais para comparar diferentes modelos e determinar qual é o mais adequado para resolver um determinado problema [34]. A seguir serão abordadas as métricas utilizadas neste trabalho.

A matriz de confusão é uma maneira simples de representar os resultados do modelo através de uma tabela, auxiliando na avaliação da performance. A tabela mostra a quantidade de ocorrências que o classificador teve para cada uma das quatro categorias, que são [34]:

- Verdadeiro Positivo (VP): Quando a classe positiva foi classificada corretamente como positiva.
- Verdadeiro Negativo (VN): Quando a classe negativa foi classificada corretamente como negativa.
- Falso Positivo (FP): Quando a classe negativa foi classificada como positiva.
- Falso Negativo (FN): Quando a classe positiva foi classificada como negativa.

Através da matriz de confusão, é possível calcular quatro métricas adicionais: acurácia, sensibilidade, especificidade e precisão. A acurácia é uma métrica amplamente

utilizada para medir a eficácia geral de um modelo de classificação, indicando a proporção de previsões corretas em relação ao número total de previsões, como pode ser visto na Equação (2). No entanto, é importante ressaltar que a acurácia por si só pode ser enganosa, pois não considera desequilíbrios nas classes ou erros específicos que podem ser mais relevantes para um determinado problema. Portanto, é necessário complementar a avaliação do modelo com outras métricas que levem esses aspectos em consideração [34].

$$\text{Acurácia} = \frac{Acertos}{Total} \tag{2}$$

Na Equação (3) temos a sensibilidade, que avalia a capacidade do modelo de identificar corretamente os resultados classificados como positivos. Já a especificidade, vista na Equação (4), mede a capacidade do modelo de detectar corretamente resultados negativos. E por fim, na Equação (5) temos a precisão que avalia a quantidade de VP sobre a soma de todos os valores preditos [34].

$$Sensibilidade = \frac{VP}{VP + FN} \tag{3}$$

$$Especificidade = \frac{VN}{VN + FP} \tag{4}$$

$$\text{Precisão} = \frac{VP}{VP + FP} \tag{5}$$

A medida F1 (ou *F-score*) é uma métrica que combina a precisão e a sensibilidade em uma única medida. Ela serve quando se deseja ter um equilíbrio entre as duas métricas. Na Equação (6) vemos que a medida F1 é a média harmônica entre a precisão e sensibilidade [34].

$$F1 = 2 * \frac{\text{precisão} * \text{sensibilidade}}{\text{precisão} + \text{sensibilidade}} \tag{6}$$

Neste trabalho, foram utilizadas as métricas de acurácia, especificidade, medida F1 e matriz de confusão como avaliações. Uma maneira de aprimorar os resultados dessas métricas é por meio da engenharia de atributos.

2.4 Engenharia de atributos

Engenharia de atributos é o processo de criar ou selecionar características criando um novo conjunto a partir dos dados brutos, afim de melhorar o desempenho dos modelos de aprendizado de máquina. Esse novo conjunto pode ser criado manualmente ou

automaticamente através de algoritmos específicos [28].

A seleção de características (do inglês, *feature selection*) é uma técnica importante no pré-processamento de dados, visando melhorar o desempenho do aprendizado de máquina, podendo construir modelos mais simples e compreensíveis [29]. O objetivo da *feature selection* é encontrar um subconjunto de características que possa descrever os dados para uma tarefa de aprendizado de máquina melhor do que o conjunto original [30].

Os algoritmos de *Boosting* que utilizam como modelo fraco árvores de decisão, fornece uma medida de importância das características (do inglês, *feature importance*) que ajuda a identificar as variáveis mais importantes para o modelo. A importância é calculada com base na frequência com que cada variável é selecionada como ponto de divisão na árvore e na redução média do ganho obtida ao dividir os dados com base nessa variável. O XGBoost fornece uma função de plotagem (*plot_importance*) que permite a visualização da importância relativa das variáveis em um gráfico de barras [24].

Outra abordagem para identificar características relevantes é por meio da utilização da Análise de Componentes Principais (do inglês, *Principal Component Analysis* - PCA). O PCA é uma técnica amplamente utilizada em estatística multivariada e possui longa trajetória na área científica. Seu objetivo principal é extrair informações de conjuntos de dados multivariados, representando-as por meio de um conjunto de novas variáveis ortogonais chamadas de componentes principais [33].

Os componentes principais são calculados para explicar a maior parte da variação entre as variáveis originais, podendo ser utilizado para redução de dimensionalidade dos dados, identificar padrões e a relação entre as variáveis. O primeiro componente principal apresenta a maior variância possível dos dados, e os demais componentes são calculados sob a restrição de serem ortogonais aos anteriores. As variáveis originais possuem pesos que compõe o componente principal [33].

Os pesos das variáveis são importantes na PCA, pois eles indicam a contribuição de cada variável para os componentes principais. Os pesos são calculados como os coeficientes da combinação linear das variáveis que formam cada componente principal. As variáveis com maiores pesos em uma componente principal são aquelas que mais contribuem para a variação total explicada por essa componente. A interpretação dos pesos é fundamental para entender quais variáveis estão mais relacionadas entre si e como elas influenciam a estrutura dos dados [33].

3 TRABALHOS RELACIONADOS

O uso de técnicas de aprendizado de máquina como forma de analisar os jogadores nos esportes é uma linha de pesquisa que sempre foi bem explorada, como podemos ver no trabalho de Chmait e Westerbeek (2021) e de Apostolou e Tjortjis (2019). O uso da inteligência artificial e análise de dados para detectar jogadores novos que serão bem sucedidos é constantemente utilizado por analistas dos times.

Este capítulo tem como objetivo contextualizar e discutir o estado da arte do uso de técnicas de aprendizado de máquina aplicadas a detecção dos melhores prospectos do *Draft* da NBA. A pesquisa dos trabalhos analisados foi feita através do *Google Scholar*[1] e *Research Gate*[2]. A busca foi direcionada a publicações que utilizassem alguma métrica de avaliação de desempenho utilizada no basquete, e que utilizavam técnicas de aprendizado de máquina.

Foram selecionados três artigos que mais se relacionam com este trabalho. Outros artigos estudados não foram selecionados para este capítulo pois tratam de análises em outros esportes como o Beisebol, como visto no trabalho de Gow (2019), ou davam pouca atenção à análise de desempenho, e avaliavam outras características, como possível salário dos jogares, exemplificado no trabalho de Papadaki e Tsagris (2022).

Kannan et al (2018) utilizaram três modelos de classificação para determinar sucesso na NBA dos jogadores que vão participar do *draft*. Os modelos utilizados foram Regressão Logística, Floresta Aleatória e Máquina de Vetor de Suporte (do inglês, *Suport Vector Machine* - SVM). O conjunto de dados foi composto por dados biométricos, estatísticas da faculdade e ordem de escolha no *draft*, sendo os dados de 2009 a 2014. O que determinava o sucesso ou não do jogador é se o mesmo fez mais de 174 jogos durante o período de interesse. Para cada modelo de classificação, foram feitos dois treinamentos, o primeiro sendo chamado de "Modelo Reduzido", usando apenas os dados biométricos, e o segundo, o "Modelo Completo", usando todo o conjunto de dados. Concluiu-se que Floresta aleatória obteve os melhores resultados, juntamente com o "Modelo Completo". O trabalho de Kannan et al (2018) demonstrou que utilizar dados referentes ao período de faculdade é de grande importância para determinar o sucesso de um jogador na NBA.

Liu, Schulte e Li (2019) (2019) propôs em seu trabalho as melhores escolhas nos *drafts* da *National Hockey League* (NHL) e NBA. Trataremos aqui apenas do que foi desenvolvido para NBA. Foram utilizados dados demográficos e dados da universidade e a Classificação de Eficiência do Jogador (do inglês, *Player efficiency rating* - PER) como atributo de saída. O autor utilizou o modelo preditivo de Regressão Linear (do inglês, *Linear Regression*) e Árvores de modelo (do inglês, *Model Tree*) para encontrar o PER.

[1] *Google Scholar* - https://scholar.google.com.br/
[2] Research Gate - https://www.researchgate.net/

Por fim, Liu (2019) observou que, com seu modelo, obteve melhores escolhas do que as que foram feitas nos *drafts*.

O trabalho de Nguyen (2022) teve como proposta prever a probabilidade de um jogador da NBA ser selecionado para o *All-Star Game* da próxima temporada, que é um evento onde junta os melhores jogadores da temporada, usando a Participação em Vitórias (do inglês, *Win Shares* - WS) como indicador de desempenho. Foram utilizados modelos de regressão e classificação para avaliar o desempenho dos jogadores. As métricas utilizadas para avaliar os modelos foram raiz quadrada do erro médio (do inglês, *Root Mean Squared Error*), erro médio absoluto (do inglês, *Mean Absolute Error*), acurácia, precisão, *Recall* e medida F1. A base de dados utilizada incluiu estatísticas dos jogadores da NBA desde 1979.

A Tabela 1 apresenta um resumo dos trabalhos discutidos neste capítulo, incluindo este trabalho.

No geral, os três artigos mencionados objetivam analisar e detectar os melhores jogadores da NBA. No entanto, alguns trabalhos não adotaram uma métrica de saída confiável que iguale todos os tipos de atletas (ofensivos e defensivos), como o WS/48 [1]. Dessa forma, a proposta deste trabalho é desenvolver um sistema de predição de jogadores utilizando uma nova abordagem, combinando dados da faculdade e NBA, usando a medida WS/48 para prever o sucesso de um jogador na NBA.

Tabela 1: Sumário dos trabalhos relacionados

Trabalho	**Objetivo**	**Métricas de Avaliação**	**Base de Dados**	**Modelos**
Kannan et al. (2018) [2]	Identificar quais métricas são mais importantes para prever o sucesso dos jogadores e avaliar a eficácia dos modelos criados	Precisão média, *eecall* médio e medida F1 média	*Basketball Reference*	Regressão Logística, Floresta Aleatória e SVM
Liu, Schulte e Li (2019) [3]	Descreve uma abordagem baseada em dados para avaliar prospectos na NBA, utilizando a aprendizagem de árvores de modelo para identificar jogadores excepcionais no *draft*	Raiz quadrada do erro médio	*Basketball Reference* e NBA	Árvores de Modelo e Regressão Linear
Nguyen et al. (2022) [4]	Prever a probabilidade de um jogador da NBA ser selecionado para o All-Star Game da próxima temporada, usando WS como indicador de desempenho	Raiz quadrada do erro médio, erro médio absoluto, acurácia, precisão, *recall* e medida F1	*Basketball Reference* e NBA	Rede Neural, Regressão Linear, Regressão Logística, Árvore de Decisão, Floresta Aleatória e *Gradient Boosting*
Este trabalho	Identificar quais características são mais importantes para prever o sucesso de um jogador na NBA, utilizando WS/48	Acurácia, Medida F1 e Especificidade	*Basketball Reference* e *Sports Reference - College Basketbal*	GBM, XGBoost, LightGBM, CatBoost e AdaBoost

4 METODOLOGIA

Este capítulo tem como objetivo apresentar todo o processo feito ao longo do desenvolvimento deste trabalho. Nas seções seguintes serão apresentadas informações sobre a coleta e criação da base de dados, seu pré-processamento e as bibliotecas e parâmetros dos métodos utilizados para previsão e análise dos atributos mais importantes para isso.

4.1 Materiais

Foi utilizada a linguagem de programação *Python* (Versão 3.6) no desenvolvimento deste trabalho. Algumas bibliotecas foram importantes para isso, sendo elas: *BeautifulSoup*, na raspagem dos dados, *Pandas*, na manipulação dos dados, *Matplotlib*, na construção de gráficos e por fim, *Scikit-learn*, no carregamento dos modelos de predição e uso de métricas. Além das bibliotecas próprias dos modelos XGBoost, LightGBM e CatBoost que não estão no *Scikit-learn*.

Foi utilizada a ferramenta gratuita da Google, *Google Colaboraty - Colab*[3] para realização do treino, além de raspagem e pré-processamento dos dados. O Colab é uma plataforma onde qualquer usuário pode escrever e executar um código em *Python* usando o navegador, além de fornecer o acesso a recursos de processamento gráfico (GPU).

4.2 Formação da Base de Dados

A base de dados utilizada foi construída a partir da extração das informações do site *Sports Reference*[4], que disponibiliza estatísticas de diversos esportes americanos, incluindo NBA e basquete universitário. Para a criação da base de dados, foram levados em consideração jogadores que foram selecionados no *draft* da NBA e entraram nas temporadas 2008-2009 à 2020-2021. Os dados extraídos tratam de dados biométricos como: peso, altura e idade; Estatísticas básicas do basquete: número de jogos, minutos jogados, pontos, índices de arremessos de 2 ou 3 pontos, lances livres, assistências, rebotes, bloqueio e roubadas de bola; e estatísticas avançadas como: *Win Shares* (WS) e estimativas em porcentagem das estatísticas básicas. A base teve um total de 1168 instâncias com 102 colunas.

4.3 Pré-processamento dos Dados

Com a base criada, foi realizado o pré-processamento fazendo com que todos os campos não possuíssem inconsistências e estejam preenchidos, garantindo que nenhum

[3]Google Colab - https://colab.research.google.com
[4]Sports Reference - https://www.sports-reference.com/

dado inconsistente afete o treinamento dos modelos. Foram descartadas informações ainda de todos os jogadores que não jogaram na universidade, pois eles não possuem dados referentes a NCAA, utilizados neste trabalho. Os dados referentes a altura dos jogadores estavam em pés e foram convertidos para centímetros. Foi feita a discretização da saída *Win Share per 48* (WS/48) , onde valores maiores que 0,100 foram considerados como 1, representando jogadores acima da média, e valores menores foram considerados 0, que representa jogadores abaixo da média.

Foram removidos ainda atributos que estão relacionadas ao desempenho do jogador que são: PER, WS, *Offensive* WS, *Defensive* WS e WS *per* 40. Como essas características estão relacionadas diretamente com o atributo de saída, elas naturalmente refletiriam o desempenho do jogador.

4.3.1 Categorização de textos

Todos os textos presentes nas colunas *college*, *position* e *NCAA_position* precisaram ser tratados. Todo o texto foi convertido para minúsculo, foram removidos os espaços e os caracteres especiais. Após ter os textos com as mesmas características, foi feita a categorização dessas colunas através da função *Categorical* da biblioteca Pandas.

4.3.2 Dados faltantes

Foram removidos todas as instâncias ou atributos que possuíam dados faltantes. A remoção do atributo foi realizada pois não seria possível preencher através de heurísticas, como por exemplo a porcentagem de rebotes ofensivos, visto que para calcular esse dado seria necessário ter o número de tentativas e o número de acertos, mas a base não possui esses dados separados. Além disso, como aproximadamente 20% da base não possui esses dados, utilizar uma média, mediana ou moda nesses atributos teria um impacto significativo nos resultados.

4.3.3 Atributos com alta correlação

Por último, foram verificados quais atributos possuem uma alta correlação com a saída e entre eles, o método de correlação foi do tipo *pearson* utilizada na função *corr* do pandas. Foram removidos os atributos de entrada que possuíam mais de 70% de correlação entre eles, deixando apenas um. Após o pré-processamento, a base que possuía 1168 linhas e 102 colunas passou a ter 991 linhas e 36 colunas. Apesar de poucas instâncias terem sido removidas, houve a remoção de mais da metade dos atributos. Essa perda de atributos não é considerada como perda de informação, visto que os dados eram altamente correlacionados com os que foram mantidos.

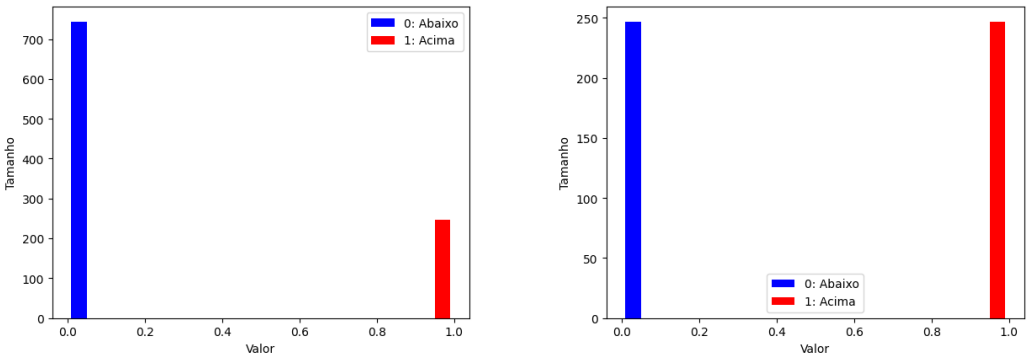
4.3.4 Redução de dimensionalidade

Após a remoção dos atributos altamente correlacionados, foi feita a redução da dimensionalidade dos atributos utilizando PCA, para que possamos avaliar modelos mais simples. Primeiramente os dados foram normalizados, utilizando o método *StandardScaler* do *Scikit-learn preprocessing*. Depois de ter os dados normalizados, foi verificado se o resultado da média está próxima de 0 e o desvio padrão próximo de 1, assim garantimos que os dados estão em uma escala adequada.

Com os dados normalizados, foi possível realizar a redução de dimensionalidade, utilizando o método *PCA* do *Scikit-learn decomposition*. Primeiramente foi analisada a variância explicada cumulativa para cada componente principal, para isso foram utilizados 18 componentes (metade do número de atributos).

4.4 Treinamento

O trabalho foi desenvolvido utilizando cinco métodos diferentes de classificação, sendo eles: GBM, XGBoost, LightGBM, CatBoost e AdaBoost. Foram utilizados métodos do *Scikit-learn* para o GBM e AdaBoost. Para o XGBoost[5], LightGBM[6] e CatBoost[7] foram utilizadas as bibliotecas próprias de cada modelo.



(a) Sem *downsampling* **(b) Com *downsampling***

Figura 4: Quantidade de amostras para treinamento.
Fonte: Autoria própria

Para realizar o treinamento, foi necessário dividir o conjunto de dados de forma que 85% será destinado para treino e validação, e 15% destinado para teste. Antes de dividir os subconjuntos, foi necessário realizar um *downsampling* (redução artificial da taxa de amostragem) de forma aleatória utilizando o método *resample* do *Scikit-learn*, pois como pode ser visto na Figura 4a a quantidade de dados com a classe de valor "Abaixo"está

[5]XGBoost - https://xgboost.readthedocs.io/en/stable/
[6]LightGBM - https://lightgbm.readthedocs.io/en/v3.3.2/
[7]CatBoost - https://catboost.ai/

consideravelmente maior do que o valor "Acima". Após utilizar o método *downsampling*, as saídas passaram a possuir a mesma quantidade de instâncias, como pode ser visto na Figura 4b.

Após separados os subconjuntos de treinamento e teste, foi realizado a validação cruzada com 10 partes. No trabalho foi utilizado os hiperparâmetros padrões de cada modelo, sem nenhuma variação de hiperpâmetros. O modelo foi avaliado com o subconjunto de teste, para geração das métricas de avaliação acurácia, medida F1, especificidade e matriz de confusão, para cada modelo de classificação utilizado nesse trabalho.

5 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

Neste capítulo serão descritos e analisados os resultados obtidos no desenvolvimento deste trabalho. O capítulo foi dividido em três partes: a primeira contém os resultados dos métodos de aprendizado de máquina para classificar os jogadores da NBA, a segunda apresenta as características mais importantes encontradas através do PCA, e por fim, a terceira apresenta a análise dos atributos mais relevantes descobertos pelos modelos de classificação.

5.1 Resultados do treinamento dos modelos

Os resultados do treinamento dos modelos são obtidos através do teste. Esta etapa consiste em executar os modelos já treinados, utilizando um conjunto de dados que não participaram do treinamento. Dessa forma, é feita a previsão através dos cinco algoritmos do tipo *Boost*, e avaliado o melhor classificador. A classificação é binária, sendo os jogadores que estão abaixo da média identificados com 0, e os jogadores acima da média identificados com 1.

A Tabela 2, apresenta os resultados para cada modelo de classificação, através das medidas de acurácia, medida F1 e a especificidade para os testes, com os valores arredondados para duas casas decimais. O modelo destacado em negrito indica o melhor resultado obtido entre os métodos de classificação. O método LightGBM foi o que obteve os melhores resultados em todas as medidas.

Tabela 2: Resultados dos modelos de classificação. Destaca-se em negrito o método que obteve o modelo com maior acurácia, medida F1 e especificidade.

Modelo	Acurácia	Medida F1	Especificidade
XGBoost	0,88	0,89	0,85
LightGBM	**0,91**	**0,91**	**0,91**
CatBoost	0,87	0,88	0,88
AdaBoost	0,84	0,85	0,88
GBM	0,89	0,90	0,91

Além das medidas mostradas acima, é possível avaliar a classificação dos testes através da matriz de confusão. Ela indica visualmente o desempenho de quantas instâncias foram classificadas de forma correta. A matriz mostra a classe verdadeira na vertical e a classe prevista na horizontal, sendo assim é possível saber a quantidade de acertos, quanto menos classes previstas erradas próximo a zero melhor. A Figura 5 apresenta a matriz de confusão do melhor modelo encontrado, como visto na Tabela 2. É possível ver que o modelo erra pouco, condizendo com sua acurácia.

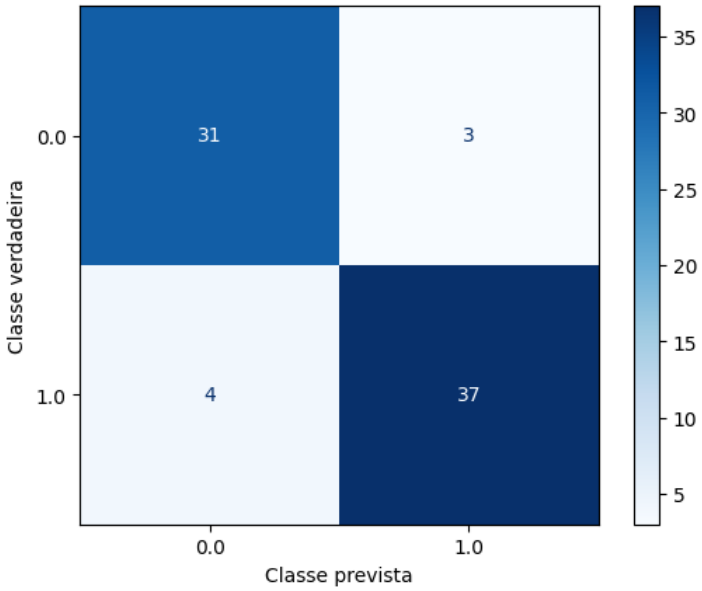


Figura 5: Matriz de confusão do método LightGBM. Ao todo foram utilizadas 75 instâncias de teste. A Classe 0 trata de jogadores abaixo da média e a Classe 1, de jogadores acima da média.

Fonte: Autoria própria

Analisando os resultados apresentados no modelo LightGBM, é possível afirmar que o classificador foi capaz de determinar se os jogadores estão acima ou abaixo da média. No melhor resultado de teste, dos 75 jogadores apenas 7 foram previsto de forma errada, mostrando que o método conseguiu entender as características determinantes para classificar os jogadores. A acurácia de 91% desse método superou a performance prevista na análise feita no trabalho de Kannan (2018).

5.2 Visualização dos atributos mais importantes

Nesta seção, serão apresentados os resultados das análises dos atributos mais importantes utilizando duas estratégias: PCA e a função *feature_importance* (importância das características) presente nos modelos de classificação. Com essa análise será possível identificar quais atributos tem maior influência no modelo preditivo, ajudando a compreender a importância dessas características para classificar os jogadores acima da média.

5.2.1 PCA

O primeiro método utilizado foi o PCA, onde o conjunto das características são transformadas em componentes principais. Conforme visto na Figura 6 é possível ver que a derivada começa a se estabilizar a partir do Componente 7. Sendo assim, a dimensionalidade da base de dados foi reduzida de 36 colunas para 7 componentes principais. O

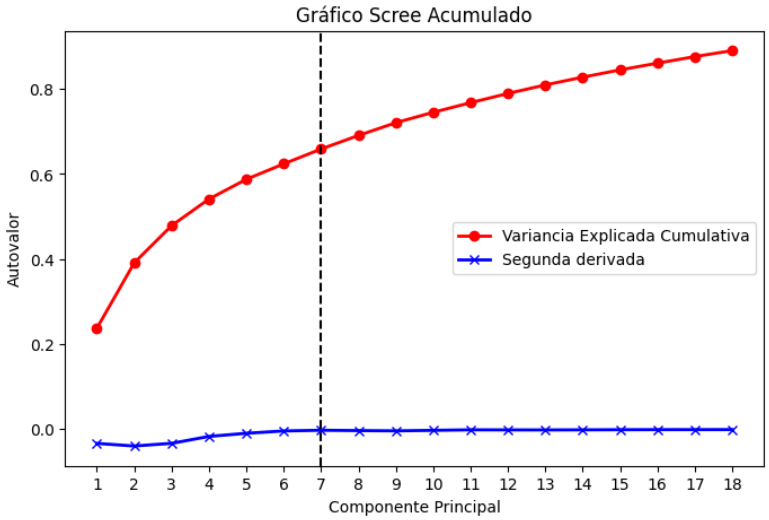


Figura 6: Autovalores do PCA com Derivadas e Variação Explicada Cumulativa para diferentes valores de componentes principais.
Fonte: Autoria própria

primeiro componente é o que apresenta a maior variância explicada das características, cerca de 24%.

No gráfico da Figura 7, é possível ver as contribuições das características no primeiro componente. Essa análise evidencia a importância das estatísticas dos jogadores durante sua passagem pela universidade, enquanto atuavam na NCAA, para a classificação de jogadores acima da média. Esses achados são consistentes com o trabalho de Kannan (2018), que também destacou a relevância dos dados da época universitária

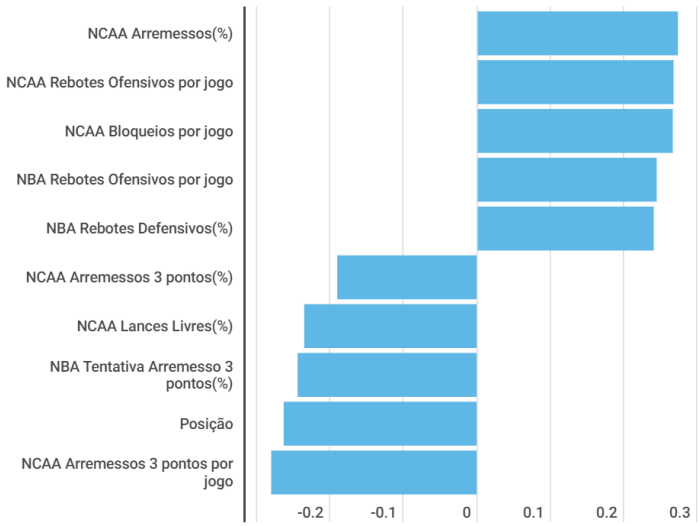


Figura 7: Contribuições das características para o primeiro componente principal. O gráfico apresenta as 10 características com maiores pesos, segundo o método de PCA, sendo 5 positivos e 5 negativos.
Fonte: Autoria própria

A Tabela 3 apresenta os resultados dos métodos de classificação utilizando os com-

ponentes principais gerados através do PCA. Ao contrário dos resultados obtidos utilizando todas as características, o modelo que obteve o melhor resultado foi o CatBoost tendo 84% de acurácia. O método utilizando todas as características obteve melhores resultados, porém, o modelo utilizando os componentes principais teve resultados melhores que os analisados no trabalho de Kannan (2018).

Tabela 3: Resultados do modelos de classificação utilizando os melhores atributos indicados pelo PCA. Em negrito destaca-se o método que apresentou o melhor modelo em relação a acurácia, medida F1 e especificidade.

Modelo	Acurácia	Medida F1	Especificidade
XGBoost	0,81	0,83	0,76
LightGBM	0,83	0,84	0,79
CatBoost	**0,84**	**0,86**	**0,79**
AdaBoost	0,81	0,82	0,85
GBM	0,81	0,83	0,79

5.2.2 Atributos mais importantes

Os métodos de classificação possuem uma função chamada *feature_importance*. Com ela é possível obter as características mais relevantes que cada modelo encontrou. A Tabela 4, apresenta o resultado do retreinamento dos modelos preditivos utilizando os 10 melhores atributos selecionados por cada modelo. É possível ver que o modelo LightGBM teve uma piora em todas as suas medidas, no entanto, o CatBoost obteve o melhor resultado entre todos os modelos deste trabalho.

Tabela 4: Resultados dos modelos de classificação utilizando os 10 melhores atributos indicados por cada um dos métodos. Em negrito destaca-se o modelo com maior acurácia, medida F1 e especificidade.

Modelo	Acurácia	Medida F1	Especificidade
XGBoost	0,81	0,82	0,85
LightGBM	0,88	0,89	0,88
CatBoost	**0,92**	**0,92**	**0,94**
AdaBoost	0,88	0,89	0,88
GBM	0,89	0,90	0,88

Analisando a matriz de confusão presente na Figura 8, é possível afirmar que o modelo obteve uma melhora na classificação de jogadores abaixo da média, sendo um ponto positivo, já que ele erra menos ao classificar um jogador ruim como um jogador bom. Além disso, conseguiu manter a taxa de acertos dos jogadores acima da média.

Ao analisar as razões pelas quais o modelo cometeu erros em 6 dos 75 casos, foram identificados padrões significativos. Dos 4 jogadores erroneamente previstos como abaixo

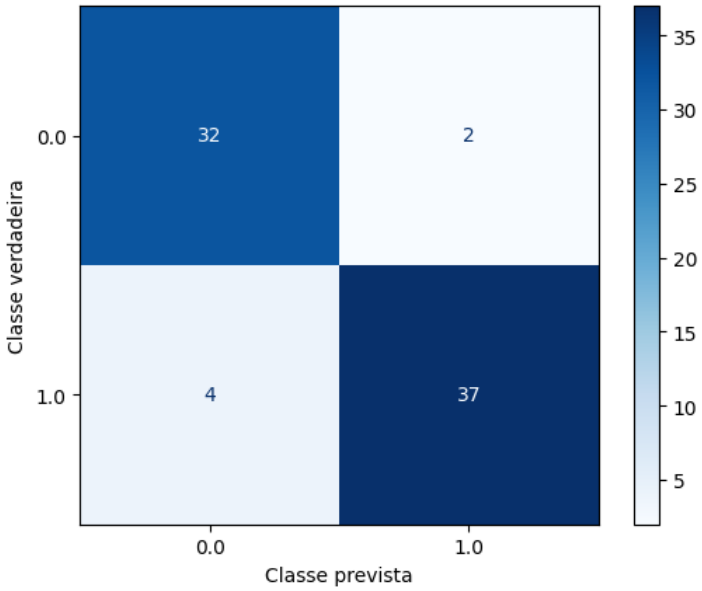


Figura 8: Matriz de confusão do método CatBoost utilizando as 10 atributos mais importantes. Ao todo foram utilizadas 75 instâncias de teste. A Classe 0 trata de jogadores abaixo da média, e Classe 1, de jogadores acima da média.
Fonte: Autoria própria

da média, constatou-se que eles apresentavam um baixo acerto de arremessos ou uma baixa porcentagem de rebotes ofensivos, duas estatísticas consideradas relevantes pelo modelo. Por outro lado, dos 2 jogadores previstos erroneamente como acima da média, um se destacava por ter uma alta porcentagem de rebotes ofensivos, enquanto o outro se destacava por cometer um número extremamente baixo de erros em quadra.

O gráfico da Figura 9, apresenta as 10 características mais importantes selecionadas pelo modelo CatBoost. Ao comparar essas características com aquelas definidas como mais relevantes pelo PCA, observa-se uma diferença significativa. Enquanto o PCA destaca as estatísticas da NCAA como mais relevantes, o modelo CatBoost valoriza mais as estatísticas da NBA. Essa discrepância entre os resultados sugere abordagens distintas na análise dos dados.

É interessante notar que, em comparação com o trabalho de Kannan (2018), algumas características se sobressaíram e apareceram entre as 10 primeiras. Por exemplo, a porcentagem de lances livres e os arremessos de 3 pontos demonstraram ser atributos relevantes em ambos os estudos.

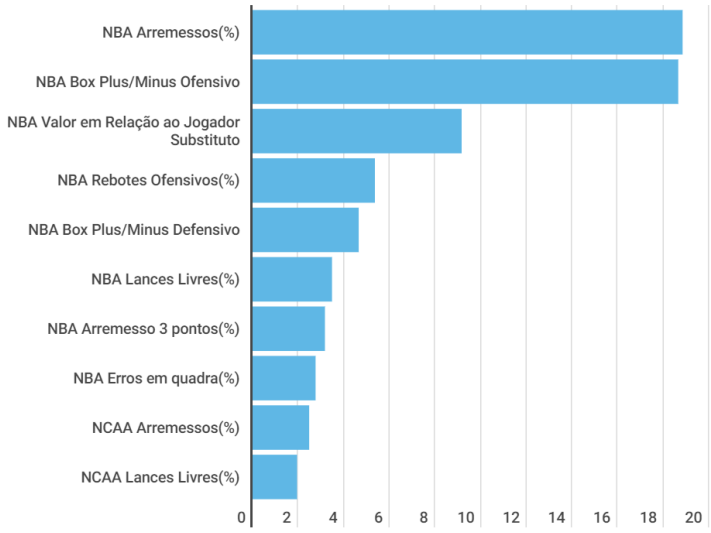


Figura 9: Atributos mais importantes encontrado no modelo CatBoost. O gráfico apresenta as 10 características selecionadas com maiores pesos no modelo de classificação.

Fonte: Autoria própria

6 CONCLUSÕES E TRABALHOS FUTUROS

O uso de métodos de Inteligência Artificial deve se tornar um padrão no progresso do descobrimento de novos talentos no meio esportivo. Os métodos possuem capacidade em extrair dos dados as informações relevantes, e desenvolver modelos preditivos capazes de classificar os jogadores, auxiliando os analistas esportivos dos times a escolherem os atletas.

Este trabalho teve como propósito o desenvolvimento de um modelo preditivo que utiliza técnicas de AM para analisar as estatísticas dos atletas, afim de identificar as características mais importantes para escolher os jogadores com *Win Share per* 48 maior que 0,100, o que é considerado jogadores acima da média na NBA. Analisando os resultados obtidos no teste, podemos ver que este trabalho obteve êxito em desenvolver um modelo capaz de prever e determinar quais atributos são mais relevantes para classificar um bom jogador. Principalmente quando comparamos o seu desempenho com o de outros trabalhos na literatura.

A principal dificuldade deste trabalho foi a construção da base de dados, visto que a maioria das bases disponíveis são pagas, ou não possuíam dados completos referentes a NCAA e NBA. Uma das formas de lidar com esse problema foi a extração dos dados presentes em sites de estatísticas de basquete, feito a partir do mapeamento da requisição que vinha em HTML para o banco de dados.

Considerando que o modelo faz uso de estatísticas referentes a NBA e NCAA, não é possível avaliar seu uso em escolhas pré *draft*, visto que para o sistema é de grande importância os dados referentes a liga principal. Logo, este trabalho é de suma importância para a manutenção de contrato dos novos jogadores, visto que entre os 2 a 4 primeiros anos é feito um pré contrato entre os times e jogadores. A partir disso, é possível ter dados suficientes para definir se renova ou não os contratos. Uma solução, e sugestão de trabalho futuro, seria criar um modelo usando apenas os dados da NCAA, utilizando os mesmos melhores atributos da NBA, podendo prever os jogadores selecionados para o *draft*.

Outra limitação deste trabalho foi a utilização de apenas 7 componentes principais no PCA. Seria mais adequado utilizar um número maior de componentes para obter uma maior variância explicada cumulativa. Além disso, foi identificado um problema relacionado à normalização dos dados. Idealmente, a normalização deveria ter sido realizada antes da verificação da alta correlação.

Neste trabalho, foram demonstradas as melhores características para classificar um jogador independentemente de sua posição. Uma possibilidade de trabalho futuro é identificar os melhores atributos para cada posição no basquete. Logo, um analista

esportivo poderá encontrar o melhor jogador para a posição específica que ele deseja. Os resultados obtidos através da análise de atributos mais relevantes podem servir como base para esses novos modelos.

REFERÊNCIAS

[1] PARKER, C. **NBA Draft Pick Valuation**. Summer Program for Undergraduate Research (SPUR), 2018. Disponível em: <https://repository.upenn.edu/spur/25>. Acesso em: 20 abr 2023.

[2] KANNAN, Adarsh; KOLOVICH, Brian; LAWRENCE, Brandon; e RAFIQI, Sohail. **Predicting National Basketball Association Success: A Machine Learning Approach**. **SMU Data Science Review**, v. 1, n. 3, 2018. Disponível em: <https://scholar.smu.edu/datasciencereview/vol1/iss3/7>. Acesso em: 20 abr. 2023.

[3] LIU, Yejia; SCHULTE, Oliver; e LI, Chao. **Model Trees for Identifying Exceptional Players in the NHL and NBA Drafts**. In: BREFELD, Ulf; DAVIS, Jesse; VAN HAAREN, Jan; ZIMMERMANN, Albrecht (org.). **Machine Learning and Data Mining for Sports Analytics**. Cham: Springer International Publishing, 2019, p. 93-105. ISBN: 978-3-030-17274-9.

[4] NGUYEN, Nguyen H. et al. The application of machine learning and deep learning in sport: predicting NBA players' performance and popularity. **Journal of Information and Telecommunication**, 6, 2, 217-235, 2022.

[5] KUMAR, Gunjan. **Machine learning for soccer analytics**. **University of Leuven**, 2013.

[6] CHMAIT, Nader; WESTERBEEK, Hans. **Artificial Intelligence and Machine Learning in Sport Research: An Introduction for Non-data Scientists**. **Frontiers in Sports and Active Living**, 3:682287, 2021. Disponível em: <https://www.frontiersin.org/articles/10.3389/fspor.2021.682287>. Acesso em: 25 abr 2023.

[7] APOSTOLOU, Konstantinos; TJORTJIS, Christos. **Sports Analytics algorithms for performance prediction**. **2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)**, pages 1-4, 2019.

[8] GOW, Alexander. **Using Machine Learning to predict MLB success Based on MILB performance**. IPHS 300: Artificial Intelligence for the Humanities: Text, Image, and Sound. Paper 18, 2019. Disponível em: <https://digital.kenyon.edu/dh_iphs_ai/18>. Acesso em: 25 abr 2023.

[9] PAPADAKI, Ioanna; TSAGRIS, Michail. **Are NBA Players' Salaries in Accordance with Their Performance on Court?** In: **Contributions to Economics**. Springer International Publishing, 2022. p. 405-428. Disponível em: <https://doi.org/10.1007%2F978-3-030-85254-2_25>. Acesso em: 25 abr 2023.

[10] CAMARGO, Vitor. **Era de Gigantes: A História do Basquete Profissional Norte-Americano no Século XX.**. Amazon, 2019.

[11] BETWAY INSIDER. NFL é a liga esportiva mais rica do mundo; veja o ranking e valores. 2022. Disponível em: <http://https://blog.betway.com/pt/outros-esportes/nfl-é-a-liga-esportiva-mais-rica-do-mundo-veja-o-ranking-e-valores/>. Acesso em: 15 mai 2023.

[12] FORBES. NBA Team Values 2021: Knicks Keep Top Spot At $5 Billion, While Warriors Seize No. 2 From Lakers. Forbes, 2021. Disponível em: <https://www.forbes.com/sites/kurtbadenhausen/2021/02/10/nba-team-values-2021-knicks-keep-top-spot-at-5-billion-warriors-bump-lakers-for-second-place/?sh=463f93f9645b>. Acesso em: 15 mai 2023.

[13] FROMAL, Adam. Understanding the NBA: Explaining Advanced Comprehensive Stats and Metrics. Bleacher Report, 2012. Disponível em: <https://bleacherreport.com/articles/1040320-understanding-the-nba-explaining-advanced-comprehensive-stats-and-metrics>. Acesso em: 15 mai 2023.

[14] BASKETBALL REFERENCE. NBA Win Shares. Disponível em: <https://www.basketball-reference.com/about/ws.html>. Acesso em: 15 mai 2023.

[15] DOUCETTE, Fox. In Search of the "League Average" NBA Player (Part 2: WS/48). Pace and Space, 2021. Disponível em: <https://paceandspacehoops.com/in-search-of-the-league-average-nba-player-part-2-ws-48/>. Acesso em: 15 mai 2023.

[16] MARTINELLI, Luís. Confira como funciona o Draft da NBA. Torcedores, 2022. Disponível em: <https://www.torcedores.com/noticias/2022/06/como-funciona-o-draft-da-nba/>. Acesso em: 15 mai 2023.

[17] VERONESI, Gabriel; SASSO, Leonardo; SUAIDE, Pedro. Draft da NBA para iniciantes: em 10 perguntas, respondemos tudo o que você precisa entender sobre a 'noite dos sonhos'. ESPN, 2019. Disponível em: <https://www.espn.com.br/nba/artigo/_/id/5746647/draft-da-nba-para-iniciantes-em-10-perguntas-respondemos-tudo-o-que-voce-precisa-entender-sobre-a-noite-dos-sonhos>. Acesso em: 15 mai 2023.

[18] SCHWARTZ, Jay. The History of College Basketball. SportsRec, 2019. Disponível em: <https://www.sportsrec.com/378124-the-history-of-college-basketball.html>. Acesso em: 15 mai 2023.

[19] LAZARINI, Rodrigo. Descobrindo o College: as regras e estrutura do basquete universitário dos EUA. LIVE BASKETBALL BR, 2020. Disponível em: <https://livebasketballbr.com/descobrindo-o-college-as-regras-e-estrutura-do-basquete-universitario-dos-eua/>. Acesso em: 15 mai 2023.

[20] DONNELL, Ricky O. The 30 best men's college basketball players of the decade, ranked. SBNATION, 2020. Disponível em: <https://www.sbnation.com/college-basketball/2020/1/1/20856865/college-basketball-players-of-the-decade-2010s-zion-williamson-anthony-davis-draymond-green>. Acesso em: 30 mai 2023.

[21] FREUND, Yoav; SCHAPIRE, Robert E. A Short Introduction to Boosting. **Journal of Japanese Society for Artificial Intelligence**, 14(5):771-780, setembro, 1999.

[22] FRIEDMAN, Jerome H. Greedy Function Approximation: A Gradient Boosting Machine. **Annals of Statistics**, 29, 5, 1189-1232, 2001.

[23] NATEKIN, Alexey; KNOLL, Alois. Gradient boosting machines, a tutorial. **Frontiers in Neurorobotics** 7, dezembro, 2013.

[24] CHEN, Tianqi; GUESTRIN, Carlos. **XGBoost: A Scalable Tree Boosting System**. Washington: University of Washington, 2016. Disponível em: <https://arxiv.org/pdf/1603.02754.pdf>. Acesso em: 15 mai 2023.

[25] KE, Guolin et al. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. **Advances in Neural Information Processing Systems**, 30, 2017.

[26] PROKHORENKOVA, Liudmila et al. CatBoost: unbiased boosting with categorical features. **Advances in Neural Information Processing Systems**, 31, 2018.

[27] BINDER, H. et al. The Evolution of Boosting Algorithms. **Methods of Information in Medicine**, 53, 6, 419-427, 2014. Disponível em: <https://www.thieme-connect.com/products/ejournals/abstract/10.3414/ME13-01-0122>. Acesso em: 15 mai 2023.

[28] HEATON, Jeff. An Empirical Analysis of Feature Engineering for Predictive Modeling. **SoutheastCon**, 2016. Disponível em: <https://ieeexplore.ieee.org/document/7506650>. Acesso em: 15 mai 2023.

[29] LI, Jundong et al. Feature Selection: A Data Perspective. **ACM Comput. Surv.**, 50, 6, 2017. Disponível em: <https://dl.acm.org/doi/10.1145/3136625>. Acesso em: 15 mai 2023.

[30] PHYU, Thu et al. Performance Comparison of Feature Selection Methods. **MATEC Web of Conferences**, 42, janeiro, 2016.

[31] WILSON, Josh. How many games are in an NBA season?. Fansided, 2022. Disponível em: <https://fansided.com/2022/10/28/how-many-games-nba-season/>. Acesso em: 15 mai 2023.

[32] PIANUCCI, M.N et al. Uso de árvore de decisão para previsão de geração de viagens como alternativa ao método de classificação cruzada. **Revista de Engenharia Civil**, 56, 5-13, 2019.

[33] ABDI, Hervé; WILLIAMS, Lynne J. Principal component analysis. **Wiley interdisciplinary reviews: computational statistics**, 2, 4, 433-459, 2010.

[34] MARIANO, Diego. Métricas de avaliação em machine learning: acurácia, sensibilidade, precisão, especificidade e F-score. **BIOINFO – Revista Brasileira de Bioinformática**, 01, jun, 2021.