Pontifícia Universidade Católica
do Rio de Janeiro

Eduardo Véras Argento

Sistema Inteligente de apoio a técnicos de basquete

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós–graduação em Engenharia Elétrica, do Departamento de Engenharia Elétrica da PUC-Rio.

Orientador : Prof. Marley Maria Bernardes Rebuzzi Vellasco
Coorientador: Prof. José Franco Machado do Amaral
Coorientador: Prof. Karla Tereza Figueiredo Leite

Rio de Janeiro
Setembro de 2023

Pontifícia Universidade Católica
do Rio de Janeiro

Eduardo Véras Argento

Sistema Inteligente de apoio a técnicos de basquete

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós–graduação em Engenharia Elétrica da PUC-Rio. Aprovada pela Comissão Examinadora abaixo:

Prof. Marley Maria Bernardes Rebuzzi Vellasco
Orientador
Departamento de Engenharia Elétrica – PUC-Rio

Prof. José Franco Machado do Amaral
Coorientador
Universidade do Estado do Rio de Janeiro - UERJ

Prof. Karla Tereza Figueiredo Leite
Coorientador
Universidade do Estado do Rio de Janeiro - UERJ

Dr. Meyer Elias Nigri
Pesquisador Autônomo

Prof. Waldemar Celes Filho
Departamento de Informática – PUC-Rio

Prof. Harold Dias de Mello Junior
Universidade do Estado do Rio de Janeiro - UERJ

Rio de Janeiro, 14 de Setembro de 2023

Eduardo Véras Argento

Engenheiro Eletricista com ênfase em Sistemas Eletrônicos graduado na Universidade do Estado do Rio de Janeiro com experiência na área de robótica e automação durante o trabalho de conclusão de curso, trabalho que resultou em três publicações incluindo um congresso e uma revista de automação. Atualmente estudando métodos de apoio à decisão no mestrado da PUC-Rio, tendo trabalhado em projetos envolvendo redes neurais e outros métodos de mineração de dados por meio da linguagem de programação "Python". Em busca de resultados importantes na engenharia.

Ficha Catalográfica

Véras Argento, Eduardo

Sistema Inteligente de apoio a técnicos de basquete / Eduardo Véras Argento; orientador: Marley Maria Bernardes Rebuzzi Vellasco; coorientadores: José Franco Machado do Amaral, Karla Tereza Figueiredo Leite. – 2023.

81 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Elétrica, 2023.

Inclui bibliografia

1. Engenharia Elétrica – Teses. 2. Redes Neurais. 3. K Nearest Neighbours. 4. Basquete. 5. Análise esportiva. 6. Inteligência artificial. 7. Inferência de performance coletiva. 8. Previsão. 9. Apoio à decisão. I. Maria Bernardes Rebuzzi Vellasco, Marley. II. Franco Machado do Amaral, José. III. Tereza Figueiredo Leite, Karla. IV. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Elétrica. V. Título.

CDD: 004

Aos meus pais, pelo apoio
e incentivo.

Agradecimentos

Para meus orientadores Professora Marley Vellasco, Professora Karla Figueiredo e Professor José Franco Amaral pelo estímulo e parceria para realizar este trabalho e pelo auxílio pessoal em diversos aspectos.

Para o Professor Meyer Nigri pelo suporte neste projeto e auxílio pessoal em diversos aspectos.
Para os colaboradores e amigos Paulo Costa, Thiago Toledo, Lívia Silva, Lucas Vital e Waldemar Celes pela parceria neste projeto.
Para minha família e amigos pelo suporte externo.
Para meus colegas da PUC-Rio pela parceria.
Para o CNPq e a PUC-Rio, pelo auxílio fornecido, sem o qual este trabalho não poderia ter sido realizado.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001

Resumo

Véras Argento, Eduardo; Maria Bernardes Rebuzzi Vellasco, Marley; Franco Machado do Amaral, José; Tereza Figueiredo Leite, Karla. **Sistema Inteligente de apoio a técnicos de basquete**. Rio de Janeiro, 2023. 81p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

Em meio ao avanço expressivo da tecnologia e às evoluções contínuas observadas no ramo de inteligência artificial, esta última se mostrou ter potencial para ser aplicada a diferentes setores da sociedade. No contexto de extrema competitividade e relevância crescente nos esportes mais famosos ao redor do mundo, o basquete se apresenta como um esporte interessante para a aplicação de mecanismos de apoio à decisão capazes de aumentar a eficácia e consistência de vitórias dos times nos campeonatos. Diante desse contexto, este estudo propõe o desenvolvimento de sistemas de apoio à decisão baseados em modelos de redes neurais e k-Nearest Neighbors (kNNs). O objetivo é avaliar, para cada substituição durante um jogo de basquete, qual grupo de jogadores em quadra, conhecido por quinteto, apresenta mais chances de ter uma maior vantagem sobre o adversário. Para tal, foram treinados modelos para classificar, ao final de uma sequência de posses de bola, a equipe que conseguiria vantagem, e prever a magnitude dessa vantagem. A base de dados foi obtida de partidas do Novo Basquete Brasil (NBB), envolvendo estatísticas de jogadores, detalhes de jogo e contextos diversos. O modelo apresentou uma acurácia de 76,99% das posses de bola nas projeções de vantagem entre duas equipes em quadra, demonstrando o potencial da utilização de métodos de inteligência computacional na tomada de decisões em esportes profissionais. Por fim, o trabalho ressalta a importância do uso de tais ferramentas em complemento à experiência humana, instigando pesquisas futuras para o desenvolvimento de modelos ainda mais sofisticados e eficazes na tomada de decisões no âmbito esportivo.

Palavras-chave

Redes Neurais; K Nearest Neighbours; Basquete; Análise esportiva; Inteligência artificial; Inferência de performance coletiva; Previsão; Apoio à decisão.

Abstract

Véras Argento, Eduardo; Maria Bernardes Rebuzzi Vellasco, Marley (Advisor); Franco Machado do Amaral, José (Co-Advisor); Tereza Figueiredo Leite, Karla (Co-Advisor). **Intelligent system to support basketball coaches**. Rio de Janeiro, 2023. 81p. Dissertação de Mestrado – Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro.

In light of the recent significant growth in technological capabilities and the observed advancements in the field of computational intelligence, the latter has demonstrated potential for application in various sectors of society. In the context of extreme competitiveness and increasing relevance in the most famous sports around the world, basketball presents itself as an interesting sport for the application of decision-support mechanisms capable of enhancing the efficacy and consistency of team victories in championships. In this context, this study proposes the development of decision-support systems, such as neural networks and k-Nearest Neighbors (kNNs). The goal is to evaluate, for each substitution during a match, which group of players in the field, known as lineup, presents the most probability to be superior to their opponent. For this, models were trained to predict, during a sequence of possessions, the team that would have advantage and the magnitude of this advantage. The database was obtained from "Novo Basquete Brasil" (NBB) matches, involving players statistics, match details and different contexts.. The model achieved an accuracy of 76,99% in projections of superiority between the playing lineups, demonstrating the potential of using computational intelligence methods in decision-making applied to professional sports. Finally, the study highlights the importance of using such tools in conjunction with human experience, encouraging future research for the development of even more sophisticated and effective models for decision-making in the sports field.

Keywords

Neural Networks; K Nearest Neighbours; Basketball; Sports analytics; Basketball analytics; Artificial intelligence; Lineup performance inference; Prediction; Decision support.

Sumário

"O conhecimento é limitado, enquanto a imaginação abraça o mundo inteiro, estimulando o progresso, e dando origem à evolução."...

Albert Einstein, *Regards sur le passé.*

1
Introdução

1.1
Motivação

O esporte profissional, com suas estruturas complexas, alto grau de competitividade e valor comercial, é um campo cada vez mais impulsionado por inovações tecnológicas e metodológicas. A busca incessante por vantagens competitivas levou a uma crescente adoção de ferramentas e técnicas avançadas de análise de dados (LI; XU, 2021), (LI; ZHANG, 2021), (YANG, 2020). O vasto e rico universo de informações geradas durante as observações estatísticas detalhadas dos eventos esportivos fornece uma oportunidade ideal para a aplicação de técnicas de aprendizado de máquina e inteligência computacional.

Dentro desse contexto, o basquete se destaca como uma modalidade esportiva particularmente interessante para a aplicação de tais técnicas (LIU, 2020). Este esporte é caracterizado pela sua complexa dinâmica de jogo, que resulta em uma grande quantidade de dados estatísticos, oferecendo assim um terreno fértil para a análise aprofundada de padrões e tendências. Ressalta-se ainda que, por se tratar de um esporte cuja competitividade apresenta jogos extremamente difíceis de serem previstos (LIU, 2020), mesmo uma pequena vantagem é capaz de mudar completamente o desempenho de um time durante os jogos. Apesar de já se ter registro da aplicação de técnicas avançadas de análise de dados em esporte, a aplicação direta destas para a tomada de decisões em tempo real é uma área emergente de estudo (SINGH, 2020), (KAPADIA et al., 2020).

Neste jogo, cada time consiste de até 15 jogadores inscritos, dos quais apenas 12 podem jogar em uma partida. Em quadra, cada equipe tem 5 jogadores - o chamado quinteto. Pode-se pontuar de três formas diferentes: cestas de 2, quando a pontuação é feita dentro de uma área delimitada chamada de linha de 2 pontos, cestas de 3, quando a pontuação é feita de fora da mesma área considerada anteriormente, e cestas de 1 ponto, quando se realiza a cobrança de uma falta, chamada de arremesso livre. O jogo é dividido em quatro períodos de 10 minutos, conhecidos por “quartos”, e a equipe com mais pontos ao final vence. No caso de um empate pode-se ainda ter prorrogações até que haja um vencedor, as quais tem duração de 5 minutos. As posições em quadra são geralmente divididas em armadores, alas e pivôs, cada um com funções e responsabilidades específicas. Armadores são frequentemente os

responsáveis pela distribuição da bola e pela organização das jogadas; alas são versáteis, capazes de atacar e defender com eficácia; e pivôs são geralmente os jogadores mais altos, posicionados próximo à cesta para efetuar rebotes e bloqueios. A forma de cada jogador exercer sua função e sua capacidade são comumente medidas por estatísticas como rebotes, assistências, faltas, entre outras. Por meio destas estatísticas pode-se entender a importância de determinado jogador para seu time e sua influência no resultado do jogo.

Durante uma partida, não existe um limite para o número de substituições que um técnico pode fazer. E o jogador substituído ainda pode voltar a jogar após, por exemplo, um breve descanso. Por esse motivo, a cada instante um diferente quinteto (resultado de uma ou mais substituições) pode trazer um melhor rendimento, que pode resultar em um maior número de pontos. Deste ponto de vista, observa-se a influência que as substituições causam em uma partida. Neste contexto se encontra o motivo deste trabalho: auxiliar o técnico a definir o melhor quinteto em qualquer instante da partida.

Uma demonstração proeminente da aplicação de sistemas inteligentes no domínio esportivo, particularmente no basquete, é ilustrada pelo estudo de (LOEFFELHOLZ; BEDNAR; BAUER, 2009), em que os autores realizam um estudo aplicando redes neurais em conjunto com fusão para prever o resultado de futuros jogos, e comparam esta previsão à opinião de diversos especialistas do basquete mundial.

Outra aplicação interessante de redes neurais no universo do basquete é vista em (DINIZ, 2023). No mencionado trabalho, uma avançada rede neural de grafos é implementada, a qual é meticulosamente treinada com uma vasta quantidade de dados derivados da National Basketball Association (NBA). O objetivo principal desta implementação é proporcionar uma ferramenta auxiliar para a formação de equipes de basquete, visando uma melhoria substancial no desempenho do time nos jogos.

Este trabalho se insere neste contexto visando contribuir para a aplicação da inteligência artificial no basquete de maneira inovadora. Ao aplicar e otimizar modelos de redes neurais e k-Nearest Neighbors (kNNs) para prever o saldo de pontos em sequências de posses de bola, este estudo se diferencia dos mencionados por focar na análise em tempo real do jogo. Esta metodologia pode proporcionar uma nova perspectiva para treinadores e analistas de desempenho ao ser utilizado como uma ferramenta de apoio para a tomada de decisões durante partidas, complementando as estratégias de jogo e aumentando as chances de sucesso das equipes. O trabalho desenvolvido agrega aos trabalhos citados, que buscam prever ou melhorar o rendimento dos times em larga escala, a capacidade de otimização do elenco dentro da atmosfera de cada

jogo.

1.2 Objetivos

Busca-se nesse trabalho desenvolver e aplicar um modelo preditivo baseado em aprendizado de máquina para melhorar a tomada de decisões em tempo real no basquete profissional. Isto é feito auxiliando, com base nas estatísticas instantâneas e acumuladas da partida até o dado momento, as escolhas do técnico durante um jogo com relação aos jogadores que compõe o time a atuar em quadra, conhecido como quinteto.

Esta aplicação é realizada no contexto da Liga de Basquete Brasileiro, o Novo Basquete Brasil (NBB), que é uma das principais ligas de basquete profissional e fornece um ambiente competitivo e desafiador para tal estudo.

Além disso, espera-se que este estudo contribua para o corpo de pesquisa existente sobre a aplicação de aprendizado de máquina e inteligência artificial no esporte, proporcionando uma maior compreensão dos desafios e oportunidades que essa área apresenta (LOEFFELHOLZ; BEDNAR; BAUER, 2009), (DINIZ, 2023). Por fim, a pesquisa visa destacar a importância do equilíbrio entre o uso de tecnologias de ponta e a experiência e intuição humana na tomada de decisões no esporte profissional (PRETORIUS; PARRY, 2016).

1.3 Descrição da dissertação

O trabalho foi realizado em 4 etapas, são elas: obtenção e avaliação da base de dados, pré-processamento dos dados, treinamento e otimização das redes e a aplicação do sistema.

Para conseguir prever o andamento de uma partida de basquete com a utilização de técnicas de aprendizado de máquina, primeiramente foi necessária a obtenção de dados de diversas naturezas e que fossem relevantes em relação ao comportamento dos dois times dentro da quadra. Os dados utilizados neste trabalho estão concentrados em uma base de dados com mais de 270 mil registros representando diferentes posses de bola em jogos desde 2013 até 2019 e 152 atributos, sendo estes relacionados a fatores desde o de rendimento individual e coletivo dos jogadores, até o tempo que uma posse de bola durou, bem como o resultado desta, ou seja, se resultou em cesta ou não.

Primeiramente se fez um trabalho manual para a seleção das variáveis mais relevantes, bem como a análise de valores impossíveis ou improváveis. Esta análise foi feita em conjunto com especialistas no esporte para que o conhecimento das regras e conceitos comuns pudessem ajudar a observar os

pontos mais importantes. Para auxiliar a visualização dos dados e entender como a base está distribuída foi feita então uma análise sobre as distribuições das variáveis mais importantes observadas, bem como a distribuição das classes de saída. Para este tipo de aplicação, foi necessário um longo processo de trabalho com dados que envolve um pré-processamento, uma mineração de dados e um pós-processamento. Diversos métodos estão envolvidos nesses processos, como seleção de variáveis, normalização e adequação de dados, análise manual de bases de dados, utilização de mecanismos de inteligência computacional como redes neurais, árvores de decisão, entre outros.

Após o pré-processamento dos dados e a seleção de variáveis, bem como a escolha dos métodos utilizados para efetuar a previsão, é feito o treinamento do sistema utilizando a base de dados descrita anteriormente. Em busca do melhor resultado possível para o treinamento é feita uma otimização e uma busca entre os parâmetros mais relevantes do sistema, tais como a taxa de aprendizado, a estrutura de uma rede neural, o número de vizinhos a ser utilizado, o otimizador, o número de épocas do treinamento, o número de paciência, entre outras definições importantes.

Por fim, é feita uma aplicação do sistema paralelamente com todas as opções possíveis de quinteto para enfrentar um determinado quinteto adversário. Dessa forma, o sistema fornece subsídios para o treinador buscar, a cada instante, o melhor quinteto por meio da disponibilização de uma tabela contendo a previsão de desempenho de cada combinação para o dado momento do jogo.

1.4 Contribuições

As contribuições que este trabalho entrega para a sociedade não se limitam a uma utilização de métodos de apoio à decisão para previsão de um resultado, mas representam uma classe relativamente nova de aplicação de inteligência computacional combinada com uma otimização de resultados em esportes. As ferramentas desenvolvidas durante este projeto se mostram valiosas para o crescimento da competitividade das equipes que o utilizarem a seu favor. Portanto, este trabalho representa um grande valor comercial diante de tamanha influência na sociedade observada neste ramo da indústria esportiva.

As principais contribuições desta Dissertação são:

– Classificação

  Utiliza algoritmos de aprendizado de máquina para classificar sequências de posses de bola, com base nas estatísticas instantâneas de jogo, em:

maior saldo de pontos do time de casa, maior saldo de pontos do time de fora ou saldo essencialmente igual para os dois times;

– Previsão numérica de resultado

  Utiliza algoritmos de aprendizado de máquina para prever de forma numérica de quanto será a vantagem de pontos de uma determinada equipe após uma sequência de posses de bola;

– Definição do melhor quinteto

  Reúne as informações da classificação e previsão numérica para fornecer uma previsão completa de vantagem de cada possibilidade de quinteto em relação ao adversário para cada posse de bola em tempo real. Dessa forma, auxilia o técnico na escolha de qual quinteto escalar em cada momento da partida.

1.5 Organização do trabalho

Este trabalho está organizado em mais cinco capítulos. No Capítulo 2 são apresentados os conceitos fundamentais que envolvem os métodos de classificação e previsão utilizados como base do sistema, bem como a teoria por trás do tratamento de dados aplicado. No Capítulo 3 é apresentada a concepção do sistema proposto, introduzindo a base de dados e explorando o desenvolvimento dos estágios do sistema individualmente, bem como a união dos estágios do sistema e a aplicação do modelo desenvolvido para obter o time com maior probabilidade de sucesso. No Capítulo 4 são consolidados os resultados obtidos dentro dos testes realizados em ambiente computacional. Por fim, no Capítulo 5 são apresentadas as conclusões e são feitas as considerações finais sobre o trabalho seguidas pelo capítulo com as referências.

2
Conceitos teóricos

Para compreender as técnicas envolvidas neste projeto se faz necessário o entendimento de alguns conceitos teóricos. Assim, as próximas seções apresentam brevemente alguns dos conceitos fundamentais no desenvolvimento da metodologia proposta nessa dissertação.

2.1
Mineração de dados

Conforme a tecnologia avança, se torna possível capturar, obter, gerar, armazenar e compartilhar uma quantidade de dados inimaginável há pouco tempo. Dado, nessa citação, nada mais é do que informação, bruta ou processada, obtida por alguma forma de observação manual ou mecanizada/eletrônica.

Tais informações são fundamentais na área de inteligência computacional, inferência, inteligência artificial e até mesmo para uma simples análise estatística que pode levar a grandes conclusões.

No entanto, as informações no seu estado bruto não são tão facilmente interpretáveis para as aplicações citadas anteriormente por conta tanto de problemas na aquisição, na transmissão e no armazenamento, mas também da própria natureza do objeto, que pode apresentar ruído, distorções, ou até mesmo fatos quase impossíveis de serem repetidos, que fogem ao padrão natural observado, conhecidos como *outliers*, tendo o potencial de piorar tanto o viés quanto a variância de um modelo.

A mineração de dados é uma área interdisciplinar que se baseia em técnicas e conceitos de estatística, computação e *machine learning* para descobrir padrões, associações, anomalias e regras significativas em grandes conjuntos de dados (HAN; PEI; TONG, 2022). Esta disciplina é parte do campo da Inteligência Artificial e tem como principal objetivo extrair informações úteis e conhecimento a partir de grandes volumes de dados (DIAS, 2007).

Como pode ser observado na Figura 2.1, (REZENDE et al., 2003) demonstra que para ser capaz de utilizar o conhecimento embutido em um dado bruto são necessárias diversas etapas: conhecimento do domínio, no qual é identificado o contexto dos dados e é feita uma primeira análise de como adequar os dados aos objetivos desejados; o pré-processamento de dados, no qual os dados brutos são limpos e transformados em um formato adequado para análise; a extração de padrões, na qual técnicas de *machine*

learning e estatística são aplicadas para descobrir padrões nos dados; e o pós-processamento, no qual os padrões descobertos são avaliados e utilizados para tomar decisões ou fazer previsões.

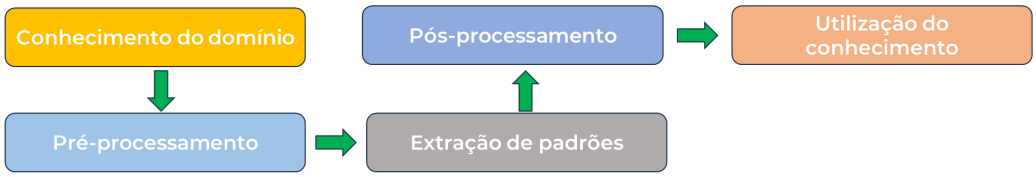

Figura 2.1: Etapas do processo de Mineração de Dados, por (REZENDE et al., 2003).

Esta metodologia tem uma vasta gama de aplicações, desde marketing e negócios até ciências da saúde e esportes. No contexto do esporte, a mineração de dados pode ser usada para analisar o desempenho dos jogadores, identificar estratégias eficazes, prever resultados de jogos, entre outras aplicações (NEVES, 2022). No caso específico desta dissertação, a mineração de dados foi aplicada para tornar uma grande quantidade de dados sobre jogos da NBB aplicável a um modelo de previsão, avaliando, durante a etapa de pré-processamento, as variáveis mais correlacionadas com o alvo previsto e eliminando possíveis variáveis que apresentem redundância ou até mesmo sejam irrelevantes, o que poderia prejudicar o rendimento do sistema.

É importante notar que a mineração de dados, como qualquer metodologia de análise, tem suas limitações. Em particular, os resultados da mineração de dados dependem fortemente da qualidade e quantidade dos dados de entrada. Além disso, a mineração de dados pode descobrir padrões que são estatisticamente significativos, mas não necessariamente úteis ou interessantes do ponto de vista prático (HAN; PEI; TONG, 2022). Por fim, a mineração de dados deve sempre ser usada em conjunto com o conhecimento do domínio e a experiência humana para interpretar corretamente os resultados e tomar decisões adequadas.

Existem diversas etapas de processamento, tratamento e manipulação de dados que buscam avaliar os dados originais e tratá-los das mais diferentes maneiras para extrair da melhor forma possível a informação central do objeto estudado (HAN; PEI; TONG, 2022), (REZENDE et al., 2003).

Entre as etapas empregadas neste trabalho, destacam-se as seguintes:

– **Avaliações de especialistas**

  Consiste em avaliar a base de dados em conjunto com um grupo de especialistas na aplicação para que possam ser observados os atributos que possuem maior ou menor potencial, assim como os que contribuem

negativamente para a construção de uma representação fiel ao padrão da aplicação em questão;

– **Análise de inconsistências**

É necessário avaliar matematicamente se existem inconsistências ou impossibilidades na base de dados observada, dadas as regras da aplicação, conceitos e características do objeto observado. Dessa forma, podem ser reduzidas informações incorretas ou absurdas que podem afetar negativamente a análise futura;

– **Detecção de *outliers***

Para realizar essa tarefa são aplicadas funções que podem observar a distância matemática entre os dados ruidosos (*outliers*) e o padrão esperado, considerando um limiar de tolerância;

– **Seleção de variáveis**

Dependendo do objetivo do processamento dos dados, alguns dos atributos informados podem ser irrelevantes ou até mesmo prejudiciais para o processo seguinte. Portanto, além da avaliação com especialistas já citada, existem alguns métodos que podem avaliar anteriormente ou posteriormente à aplicação se determinadas variáveis estão de fato sendo úteis ou se estão atrapalhando o processo desejado. Tais métodos são explorados na subseção 3.2.

– **Avaliação dos dados**

Dentro do processo de mineração de dados, pode-se considerar a extração de padrões e a busca por sua previsão, agrupamento (clusterização) ou classificação. Para tal, podem ser utilizados diversos sistemas diferentes, como Redes Neurais, k-Nearest Neighbors (kNN)s, entre outros. Estes dois métodos serão mais explorados na subseção 2.3.

– **Pós-processamento**

Após todo o processo de extração de informações de uma base de dados, é necessário ainda que seja feita uma avaliação nos resultados obtidos, se são viáveis, se estão de acordo com os padrões esperados e, por fim, extrair das informações geradas os padrões em sua forma mais acurada possível.

2.2 Métodos de seleção de variáveis

Conforme mencionado anteriormente, para que seja alta a qualidade do resultado obtido através da aplicação de determinados dados em um modelo de previsão ou detecção de padrões, é necessário que as informações fornecidas para o sistema sejam de fato capazes de influenciar de forma coerente o objeto buscado.

A partir da base de dados disponível pode-se observar suas variáveis e avaliar a relação entre elas tanto de forma analítica, com o auxílio de especialistas que dominam o contexto em questão, avaliando qualitativamente cada variável e suas características, como também de forma numérica, com o auxílio de algoritmos, métodos e estudos estatísticos.

Dentre os métodos computacionais, três categorias mais comuns separam a forma como é feita a avaliação das variáveis: Os métodos "*wrapper*", os métodos "*filter*" e os métodos "*Embedding*". Segundo (GHOSH et al., 2020) e (HAN; PEI; TONG, 2022), tais categorias podem ser definidas da seguinte forma:

– Métodos "*Wrapper*"

São uma classe de algoritmos que buscam identificar, a partir de diferentes conjuntos de variáveis, os melhores atributos de entrada baseando-se em um algoritmo de modelagem que é usado como parte do processo de seleção. Pelo fato de aplicar diretamente o modelo a ser testado, esta classe de algoritmos é capaz de retornar um resultado diretamente condizente com a situação à qual o sistema é submetido. No entanto, como na maioria das vezes os modelos desenvolvidos buscam avaliar bases de dados grandes ou buscam entender uma complexidade superior ou próxima à capacidade humana de inferir, é compreensível que os modelos normalmente necessitem de grande capacidade computacional, e realizem uma grande quantidade de processamento para gerar um único resultado. Portanto, este processo exige uma quantidade de tempo e processamento consideráveis, e para aplicar métodos wrapper pode ser inviável ou muito custoso computacionalmente.

Como forma de reduzir este problema algumas possibilidades distintas podem ser empregadas. Pode ser utilizado um algoritmo genético que auxilia a escolha de atributos testada minimizando bruscamente a quantidade de avaliações necessárias para se chegar muito perto do resultado ótimo. Algumas heurísticas também podem ser empregadas como o "*método de seleção passo-a-passo à frente*" e o "*método de eliminação para*

trás" que respectivamente adiciona e elimina variáveis de entrada até que a acurácia atingida comece a retrair (HAN; PEI; TONG, 2022).

Portanto, utilizar métodos "*Wrapper*" permite alcançar melhores combinações de atributos de entrada, no entanto, com um custo computacional muito maior.

– Métodos "*Filter*"

Para contrapor a questão levantada anteriormente, os métodos "*Filter*" analisam relações lineares ou não lineares entre as variáveis de entrada, entre as variáveis de saída e entre as duas anteriores. O objetivo é identificar, por exemplo, a relação de dependência entre duas variáveis, eventualmente tornando uma delas desnecessária, a relação direta entre variáveis de entrada e saída, observando a relevância de cada um dos atributos em relação ao objetivo observado, entre outros. Dessa forma, é possível avaliar estatisticamente as variáveis independentes, e poder escolhê-las como entrada para o sistema, economizando tempo e processamento, bem como permitindo uma melhor compreensão das informações a serem aplicadas no modelo.

– Métodos"*Embedding´´*

Para combinar as vantagens dos dois métodos anteriores, os métodos "*Embedded´´* realizam o desenvolvimento do modelo de previsão ao mesmo tempo em que seleciona as variáveis. Dessa forma, além de gerar uma seleção de atributos mais condizente com o modelo desenvolvido, sua implementação é menos custosa do que observada nos métodos "*Wrapper´´*.

Em suma, a escolha entre métodos wrapper, métodos *filter* e *embedding* depende de vários fatores, incluindo o tamanho e a dimensão do conjunto de dados, o modelo de predição a ser utilizado, e a necessidade de equilibrar a acurácia e a eficiência. Em muitas aplicações práticas, incluindo a previsão de desempenho no esporte, pode ser útil combinar esses três tipos de métodos para explorar as vantagens de cada um, como foi feito neste trabalho.

Uma boa forma de gerir e aplicar algoritmos de seleção de variáveis é utilizando uma ferramenta chamada *WEKA*, que permite analisar os dados do ponto de vista de inúmeras métricas distintas de forma fácil e objetiva (MARKOV; RUSSELL, 2006).

Neste trabalho, os resultados da análise de variáveis foram baseados no ranqueamento gerado pela aplicação do método filter Relief-F.

Segundo (ROBNIK-ŠIKONJA; KONONENKO, 2003), este método busca atributos que são capazes de separar as classes com mais clareza, isto é, é bem

definido em registros de mesma classe enquanto pode ser bem distinguido em registros de classes distintas. O algoritmo utiliza o valor dos atributos em conjunto com sua posição e vizinhança no espaço de registros e identifica os atributos mais ou menos relevantes. Dessa forma, é possível principalmente identificar quais atributos não são de grande serventia para a base.

O processo utilizado neste projeto é desenvolvido na seção 3.2

2.3 Métodos de apoio à decisão

Os métodos de apoio à decisão se referem a uma ampla variedade de técnicas e ferramentas computacionais que auxiliam indivíduos e organizações a tomar decisões com base em um banco de informações. Esses métodos abrangem ferramentas estatísticas de análise de dados, como a classificação naïve Bayes, que assume que os atributos de entrada são independentes entre si e calcula a probabilidade de uma determinada classe com base nestes atributos. Mas principalmente, há algoritmos de *machine learning*, como máquinas de vetores de suporte, caracterizadas por buscar um hiperplano que melhor se adequa a um conjunto de dados, Redes Neurais, que utilizam um conjunto de neurônios artificiais com conexões ponderadas para processar dados de entrada e os K-Nearest Neighbors (kNNs), que buscam a previsão com base no cálculo da distância entre a entrada e os dados mantidos como referência (HAN; PEI; TONG, 2022).

Mais especificamente, tanto as Redes Neurais quanto as kNNs são métodos de apoio à decisão bastante utilizados que podem ser usados para extrair conhecimento a partir de dados e ajudar a tomar decisões informadas em diversos domínios, incluindo o esporte (HAN; PEI; TONG, 2022)M

2.3.1 Redes Neurais

As Redes Neurais são modelos de aprendizado de máquina inspirados na forma como o cérebro humano funciona, sendo compostas por nós, ou "neurônios", organizados em uma ou mais camadas ocultas e uma camada de saída (LAWRENCE, 1993). Ilustrado pela Figura 2.2 e apresentado na Equação 2-1, um neurônio em redes neurais artificiais é caracterizado por um elemento que recebe informações (equiparadas a sinapses no cérebro), pondera a relevância das informações por meio de pesos, considera um viés nesta soma, processa soma por meio de uma função de ativação, e então emite uma informação de saída. Sua função de ativação é tipicamente não-linear para que a rede possa resolver problemas de natureza não-linear. Por fim, conforme

representado na Figura 2.3 uma rede completa é formada por um conjunto de neurônios organizados em camadas que compartilham tais informações ao longo de suas camadas ocultas, resultando em uma ou mais camadas de saída, fornecendo uma conclusão sobre os dados injetados.(HAN; PEI; TONG, 2022).

$$\sum_{i=1}^{n} X_i w_i + b \tag{2-1}$$

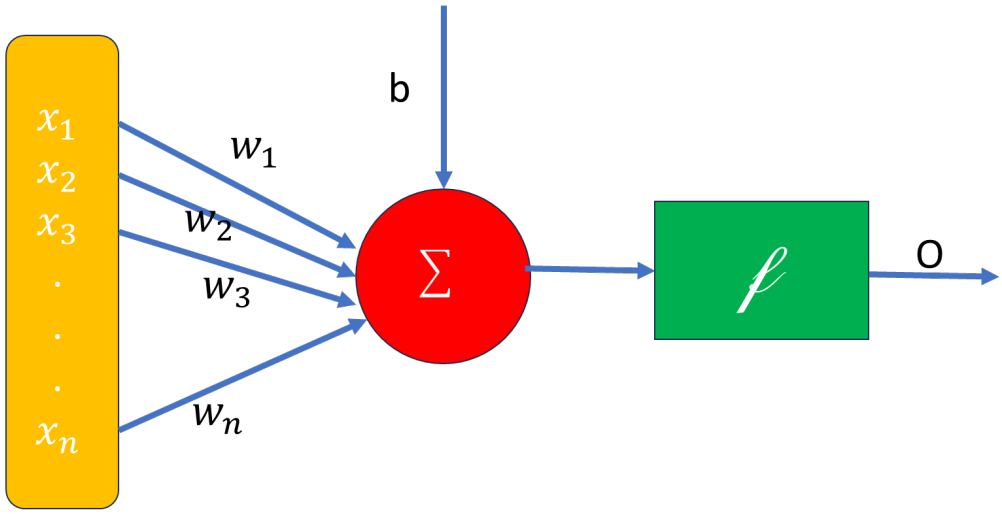

Figura 2.2: Ilustração de neurônio recebendo as entradas x_1, x_2 até x_n, somando uma combinação linear através dos pesos w_1, w_2 até w_n e uma variável que introduz o viés, b, e processando as informações através da função de ativação f.Ilustração por (HAN; PEI; TONG, 2022).

São caracterizadas por aprender a partir de dados de entrada e melhorar seu desempenho ao longo de um processo de treinamento. O aprendizado em uma rede neural acontece ao ajustar os pesos de suas conexões. Durante o processo de treinamento, a rede neural recebe dados de entrada, os processa ao longo das camadas de acordo com atributos que controlam a evolução e o comportamento do aprendizado dos modelos, denominados hiperparâmetros, e apresenta o resultado na saída. A saída é então comparada com a saída esperada e a diferença (ou erro) é usada para ajustar os pesos. Esse processo é repetido várias vezes com diferentes conjuntos de dados de entrada até ter o treinamento interrompido pelo modelo quando o erro do conjunto de validação aumenta consistentemente por um determinado número de vezes de forma sequencial, determinado pelo hiperparâmetro "*paciência*".

Redes neurais são especialmente úteis em tarefas de classificação e regressão, tarefas que envolvem a detecção de padrões complexos ou não lineares nos dados. Elas tem sido empregadas em uma ampla gama de aplicações, desde reconhecimento de voz e imagem, tradução automática, diagnóstico médico, até previsão de séries temporais. No contexto esportivo, as redes neurais podem

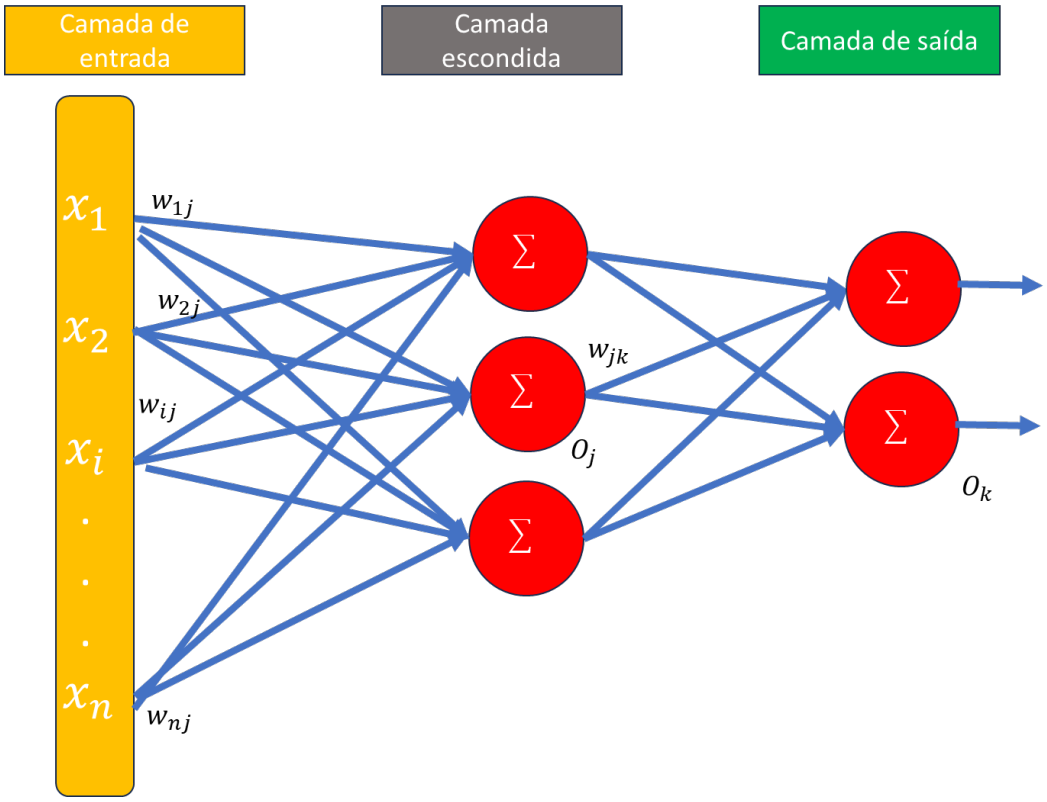

Figura 2.3: Ilustração de rede neural artificial e detalhamento da combinações de neurônios para sua composição. Exemplo com uma camada oculta. Ilustração por (HAN; PEI; TONG, 2022).

ser utilizadas para prever o desempenho dos jogadores ou times baseado em dados históricos (DINIZ, 2023), (LOEFFELHOLZ; BEDNAR; BAUER, 2009).

2.3.2 k-Nearest Neighbors (kNN)s

O k-Nearest Neighbors (kNN) é um algoritmo de aprendizado de máquina baseado em instâncias. Isso significa que ele não cria explicitamente um modelo a partir dos dados de treinamento, mas usa diretamente os dados de treinamento para fazer previsões (GUO et al., 2003). Segundo (SONG et al., 2017) é um método não paramétrico usado para classificação e regressão. Em ambas as tarefas, a entrada consiste nos 'k' exemplos de treinamento mais próximos no espaço de características. Em tarefas de classificação, a saída é uma classe de associação: um objeto é classificado pela classe majoritária entre seus vizinhos. Em tarefas de regressão, a saída é a média dos valores (ou mediana, ou algum outro resumo estatístico) dos 'k' vizinhos mais próximos.

A semelhança entre as instâncias é normalmente medida usando alguma forma de distância, como a distância euclidiana ou de Manhattan. A escolha do valor de 'k' e da medida de distância podem ter um impacto significativo

no desempenho do kNN (SONG et al., 2017). No contexto esportivo, o kNN pode ser usado para prever o desempenho de jogadores ou times com base em desempenhos anteriores semelhantes (WOIDA, 2021).

Este é um algoritmo intuitivo e fácil de entender, mas pode ser computacionalmente intensivo para conjuntos de dados grandes, pois requer o cálculo da distância entre a nova instância e todas as instâncias no conjunto de treinamento. Além disso, o kNN pode ser sensível a instâncias de treinamento ruidosas ou irrelevantes.

3
Concepção do sistema proposto

3.1
A base de dados

3.1.1
Descrição das variáveis

A base de dados utilizada neste trabalho é fornecida pela liga Brasileira de Basquete, o Novo Basquete Brasil (NBB). A base é estruturada com as informações instantâneas e acumuladas de cada posse de bola de cada jogo ocorrido na liga desde 2013 até 2019. Há inúmeras variáveis como a identificação da temporada, ano, times que estão se enfrentando, tempo de jogo, identificação da posse de bola, até informações estatísticas instantâneas, como por exemplo, quantos pontos foram feitos na posse de bola, quem os fez, quais jogadores estavam em quadra e também acumuladas, como por exemplo a porcentagem de cestas bem sucedidas da equipe em quadra, o acúmulo de faltas até o momento, entre outros.

Uma das informações mais importantes fornecidas pela base de dados é a representação de cada um dos jogadores presentes em cada posse de bola. Além das informações individuais, técnicas, instantâneas e acumuladas dos jogos para cada jogador, é informado o número do grupo que representa as principais características físicas, técnicas e comportamentais dos jogadores em comparação com os outros. Este dado é chamado de *cluster*, pois é fruto de um agrupamento (clusterização) feito previamente, baseado nas estatísticas que definem a influência de cada jogador sobre um momento da partida. Este agrupamento objetivou unir um grupo jogadores que possuem fundamentos semelhantes e separar aqueles que apresentam características distintas, dessa forma, gerando uma variável representativa para cada categoria de jogadores com atributos similares.

Assim, os jogadores foram divididos em 16 *clusters*, com base na avaliação de suas principais características, representadas na Tabela 3.1. Estes grupos funcionam neste projeto como principal atributo de escolha de jogadores pelos modelos desenvolvidos e, na aplicação, como parâmetro de informação entregue ao técnico.

Tabela 3.1: Descrição das variáveis utilizadas para realizar o agrupamento dos jogadores.

Variável	Descrição
Total rebounds	Total de rebotes
Offensive rebounds	Rebotes ofensivos
Defensive rebounds	Rebotes defensivos
Steals	Roubadas de bola
Total assists	Total de assistências
Total of dunks	Total de enterradas
Correct dunks	Enterradas corretas
Offensive fouls	Faltas ofensivas
Disqualifying fouls	Faltas desqualificantes
Technical fouls	Faltas técnicas
Unsportsmanlike fouls	Faltas anti-esportivas
Commited fouls	Faltas cometidas
Received fouls	Faltas recebidas
Total of violations	Total de violações
Five seconds violations	Violações de 5 segundos
Three seconds violations	Violações de 3 segundos
Field back violations	Violação de volta à quadra
Out field violations	Violação de bola fora de quadra
Walk violations	Violação de condução
Total of blocks	Total de bloqueios
Total of errors	Total de erros
Double double total	Duplos duplos (quando um jogador atinge dois dígitos em duas estatísticas)
Triple double total	Triplo duplo (quando um jogador atinge dois dígitos em três estatísticas)
% Two points	Porcentagem de cestas de dois pontos
% Three points	Porcentagem de cestas de três pontos
% Free throw points	Porcentagem de arremessos livres

Todas as variáveis presentes na base de dados podem ser observadas na Tabela 3.2, seguidas de sua explicação.

Tabela 3.2: Descrição das variáveis disponíveis na base de dados.

Variável	**Descrição**
season_id, season_year	Id da temporada e ano de início
match_id	Id da partida
home_team_id, home_team_name	Id e nome do time da casa
away_team_id, away_team_name	Id e nome do time visitante
possession_id	Id que representa posse de bola no jogo (único na partida)
period	Quarto em que a posse de bola aconteceu
remaining_minutes	Minutos restantes para o fim do jogo no começo da posse de bola
elapsed_minutes	Tempo decorrido no jogo no começo da posse de bola
home_score, away_score	Placar do time da casa e do time visitante no começo da posse de bola
home_score_difference	Diferença de placar para o time de casa
away_score_difference	Diferença de placar para o time de fora
delta_score_3pos	Média da diferença do placar nas últimas 3 posses de bola
delta_score_10pos	Média da diferença do placar nas últimas 10 posses de bola
delta_score_10s	Média da diferença do placar nos últimos 3 segundos
delta_score_30s	Média da diferença do placar nos últimos 10 segundos
home_period_fouls, away_period_fouls	Faltas acumuladas no quarto para os times de casa e visitante
home_cluster_1..16	Quantos jogadores de cada cluster estão em quadra
home_quintet_id	Identificador do quinteto do time da casa
home_p1_id, home_p1_nickname	Id e nome do primeiro jogador do quinteto do time da casa
home_p1_points	Pontos feitos pelo jogador até o momento do início da posse de bola
home_p1_fouls	Faltas acumuladas na partida cometidas pelo jogador
home_p1_eff_per_min	Razão entre arremessos convertidos e tentados por minuto no início da posse de bola
home_p1_minutes	Minutos que o jogador esteve em quadra, medido no começo da posse de bola
home_p{2,3,4,5}_id, home_p{2,3,4,5}_nickname, home_p{2,3,4,5}_points, home_p{2,3,4,5}_fouls, home_p{2,3,4,5}_eff_per_min, home_p{2,3,4,5}_minutes	Mesmos dados para os demais jogadores do quinteto
home_quintet_points	Pontos acumulados com os 5 jogadores do quinteto em quadra, até o início da posse
home_quintet_assists	Assistências feitas enquanto os 5 jogadores do quinteto estavam em quadra
home_quintet_rebounds	Rebotes coletados enquanto os 5 jogadores do quinteto estavam em quadra
home_quintet_steals	Roubadas de bola feitas enquanto os 5 jogadores do quinteto estavam em quadra
home_quintet_blocks	Bloqueios feitos enquanto os 5 jogadores do quinteto estavam em quadra
home_quintet_missed_throws	Arremessos perdidos (excluindo lance livre) com os 5 jogadores do quinteto em quadra
home_quintet_missed_free_throws	Lances livres não convertidos com os 5 jogadores do quinteto em quadra
home_quintet_fouls	Faltas cometidas com os 5 jogadores do quinteto em quadra
home_quintet_turnovers	Perdas de posse de bola com os 5 jogadores do quinteto em quadra
home_quintet_eff_per_min	Razão entre arremessos convertidos e tentados por minuto, considerando o quinteto
home_quintet_minutes	Minutos que o quinteto esteve junto em quadra até o início da posse de bola
away_cluster_1..16	Quantos jogadores de cada cluster estão em quadra pelo time visitante
away_quintet_id, away_p{1,2,3,4,5}_id, away_p{1,2,3,4,5}_nickname, away_p{1,2,3,4,5}_points, away_p{1,2,3,4,5}_fouls, away_p{1,2,3,4,5}_eff_per_min, away_p{1,2,3,4,5}_minutes	Mesmos dados para os jogadores do time visitante
away_quintet_points, away_quintet_assists, away_quintet_rebounds, away_quintet_steals, away_quintet_blocks, away_quintet_missed_throws, away_quintet_missed_free_throws, away_quintet_fouls, away_quintet_turnovers, away_quintet_eff_per_min, away_quintet_minutes	Mesmos dados para o quinteto do time visitante
home_has_possession, away_has_possession	Booleanos indicando qual time tem a posse de bola
offense_success	Indica se o time com a posse de bola a converteu em pontos
defense_success	Indica se o time sem a posse de bola a converteu em pontos
home_success	Indica se o time da casa converteu a posse de bola em pontos
away_success	Indica se o time visitante converteu a posse de bola em pontos
result	Pontos resultantes para o time com a posse de bola
home_result, away_result	Resultado da posse de bola em função de cada time
elapsed_minutes	Minutos que decorreram na posse de bola
home_wins, away_wins	Indica o time que ganhou a partida (considerando o resultado final)
home_final_score, away_final_score	Resultado final da partida

3.1.2 Estruturação dos dados para aprendizado de máquina

Uma etapa fundamental no desenvolvimento de qualquer modelo de aprendizado de máquina é a preparação e estruturação dos dados. Neste trabalho, os dados coletados provêm de jogos de basquete e são organizados na

forma de posses de bola individuais. No entanto, uma característica importante do basquete é que ele é um jogo extremamente equilibrado, no qual a vantagem técnica de um time não necessariamente se converte imediatamente em uma cesta ou em uma defesa bem-sucedida. Isso torna a previsão do resultado de uma única posse de bola um desafio considerável.

Para mitigar essa complexidade e melhorar a capacidade preditiva do modelo, adotou-se uma estratégia de aglomeração de dados. Em vez de tratar cada posse de bola como uma instância de dados independente, agrupou-se sequências de posses de bola durante as quais os jogadores em quadra não são substituídos em nenhum dos times. Isso permite observar se os times em quadra são realmente equilibrados, ou se um dos lados apresenta uma consistência técnica superior ao outro, tendendo a levar a um impacto no resultado após uma sequência de posses de bola. E dessa forma, tentando explorar a robustez que esse formato traz, o sistema pode observar melhor a consistência da vantagem de determinado quinteto sobre o outro.

O saldo de pontos obtido durante essa sequência de posses de bola é então usado como o objetivo de previsão para os modelos de aprendizado de máquina, e as entradas da rede são demonstradas posteriormente. Este enfoque permitiu analisar as tendências a médio e longo prazo do jogo, em vez de concentrar nos resultados altamente voláteis de uma única posse de bola.

Para abordar de forma mais concisa os dados e conseguir obter melhores resultados diante de tamanha complexidade intrínseca do esporte, os modelos desenvolvidos neste trabalho são divididos em dois estágios. Os estágios são separados de forma a primeiramente categorizar uma sequência de posses de bola e então, posteriormente, prever o saldo ao final desta.

No primeiro estágio, os dados são interpretados como pertencentes a uma das três classes: “vantagem”, “desvantagem” ou “igualdade”, em função do saldo de pontos, do ponto de vista de um dos times. Esta classificação trinária serve como o objetivo de previsão para o modelo de classificação inicial. No segundo estágio, por outro lado, utilizou-se o valor real do saldo de pontos para treinar os modelos, buscando uma regressão (previsão de valores) mais acurada. Uma exemplificação do funcionamento dos estágios em conjunto pode ser observada na Figura 3.1. Esta abordagem multiestágio permitiu construir um sistema de previsão robusto e versátil que é capaz de lidar com a complexidade e a imprevisibilidade do jogo de basquete, bem como oferecer dois níveis de previsão de resultado no duelo entre dois quintetos: se um deles vai obter uma vantagem, desvantagem ou igualdade e por quantos pontos se espera que isso ocorra.

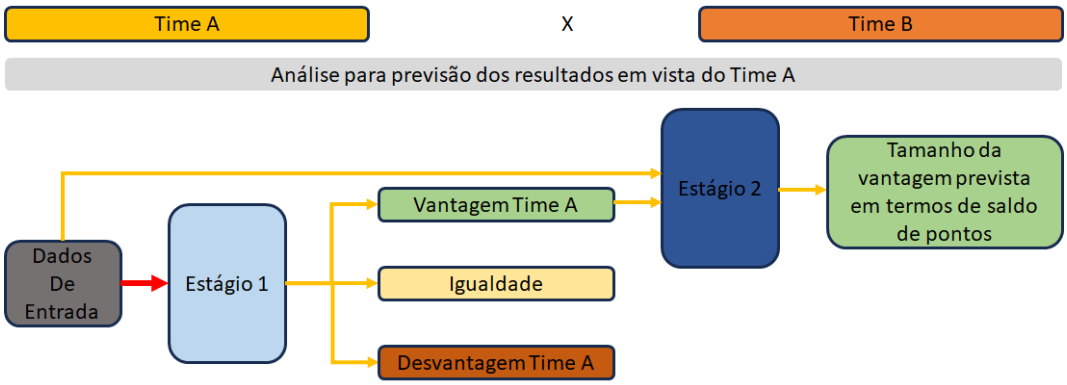

Figura 3.1: Exemplo do funcionamento dos estágios em conjunto no sistema.

Ao separar os dados por sequências de posse de bola sem alteração do time em quadra, foi obtido um dado novo. Cada observação desta nova abordagem é caracterizada por apresentar os atributos referentes à primeira posse de bola da sequência em conjunto com o saldo de pontos obtido ao longo dela, e sua contagem é feita através do número de posses de bola. No entanto, pode-se perceber ainda que contido em uma sequência de 10 posses de bolas seguidas se tem diversas sequências menores de posse de bola sem alteração de jogadores. Portanto, para maximizar os dados existentes foram considerados diversos níveis de granularidade nestas sequências, separando cada uma das janelas menores em um dado novo. Dessa forma, foram obtidas diferentes observações de diferentes tamanhos de janela agregadas em uma grande base que contém cada observação de sequências desde duas posses de bola até 30 posses de bola sequenciais. Apesar deste dado conter a informação de qual tamanho de janela se trata, este não foi considerado como entrada para a rede no primeiro estágio, visto que o algoritmo busca prever a vantagem de um time com relação ao outro, e a diferença de janelas apenas modifica, nesta interpretação, o quanto esta vantagem ficará visível. Já no segundo estágio, esta entrada é necessária como parâmetro para estimar a saída do “saldo”.

Na Figura 3.2 é exemplificado como a nova base de dados é formada. Neste exemplo, se tem em um jogo fictício com 10 posses de bola, no qual os times em quadra foram modificados duas vezes: uma por uma modificação no time de casa e outra por uma modificação no time de fora. Cada uma destas modificações delimitam uma sequência de posses de bola na qual não houve alteração de nenhum time, o que é conhecido como "*run*". Como pode ser observado na figura, duas observações podem ser formadas agregando uma sequência de 4 posses de bola em duas sequências de 3 posses de bola. Dessa forma, é criado um dado para cada tamanho de sequência em termos de posses de bola. No fim, estes dados são unificados gerando um grande arquivo com todas as observações feitas para cada sequência de posses de bola e cada

sequência menor inserida nas maiores. Para limitar o tamanho do arquivo, no entanto, o tamanho mínimo observado para as sequências é de 5 posses de bola e o tamanho máximo é de 15 posses. Assume-se que sequências menores do que esta não fornecem observação suficiente para se tornarem relevantes para a previsão, e sequências maiores são improváveis de ocorrer.

Posse de bola	Jogadores em campo (por cluster)		Atributos de entrada	Pontos ao final da posse de bola
	Time de casa	Time de fora		
Início de jogo				
p_0	1, 2, 3, 4, 5	3,4,4,5,7	Variáveis dependentes no início da posse p_0	3
p_1	1, 2, 3, 4, 5	3,4,4,5,7	Variáveis dependentes no início da posse p_1	2
p_2	1, 2, 3, 4, 5	3,4,4,5,7	Variáveis dependentes no início da posse p_2	-2
p_3	1, 2, 3, 4, 5	3,4,4,5,7	Variáveis dependentes no início da posse p_3	-1
Substituição do jogador do cluster 5 do time de fora por um do cluster 10				
p_4	1, 2, 3, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_4	-3
p_5	1, 2, 3, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_5	3
p_6	1, 2, 3, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_6	1
p_7	1, 2, 3, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_7	-2
Substituição do jogador do cluster 3 do time de casa por um do cluster 15				
p_8	1, 2, 15, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_8	-3
p_9	1, 2, 15, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_9	2
p_10	1, 2, 15, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_10	2
Fim de jogo				

Base agregando sequências de posse de bola

Posse de bola	Posses de bola	Jogadores em campo (por cluster)		Atributos de entrada	Saldo de pontos
		Time de casa	Time de fora		
p_0 - p_3	4	1, 2, 3, 4, 5	3,4,4,5,7	Variáveis dependentes no início da posse p_0	2
p_4 - p_7	4	1, 2, 3, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_4	-1
p_8 - p_10	3	1, 2, 15, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_8	1

Base agregando sequências de 3 posses de bola considerando cada possibilidade dentro de um "run"

Posse de bola	Posses de bola	Jogadores em campo (por cluster)		Atributos de entrada	Saldo de pontos
		Time de casa	Time de fora		
p_0 - p_2	3	1, 2, 3, 4, 5	3,4,4,5,7	Variáveis dependentes no início da posse p_0	3
p_1 - p_3	3	1, 2, 3, 4, 5	3,4,4,5,7	Variáveis dependentes no início da posse p_1	-1
p_4 - p_6	3	1, 2, 15, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_4	1
p_5 - p_7	3	1, 2, 15, 4, 5	3,4,4,10,7	Variáveis dependentes no início da posse p_5	2
p_8 - p_10	3	1, 2, 15, 4, 10	3,4,4,10,7	Variáveis dependentes no início da posse p_8	1

Figura 3.2: Exemplo da criação do arquivo de sequências de posses de bola com resultados fictícios para compreensão do sistema.

3.2 Pré-processamento

Na etapa de análise exploratória e manipulação manual da base de dados, foi observada a presença de valores acima do valor comum para um resultado de posse de bola, isto é, uma cesta de valor máximo igual a 3. Isto se deve ao fato de que após uma cesta que agrega 2 ou 3 pontos de vantagem, o time atacante pode ainda sofrer uma falta sem perder a posse de bola. Ao executar o lance livre, é possível haver um rebote, mantendo a posse de bola com o mesmo time, propiciando a possibilidade de obter mais uma cesta, mais faltas e assim por diante. Apesar desta possibilidade ser real, é extremamente improvável que este tipo de ocorrência se repita muitas vezes. Assim, valores acima de 4 pontos por posse de bola são extremamente raros e, portanto, considerados "*outliers*". A contagem de valores de resultados da base de dados pode ser observada na Tabela 3.3. Dessa forma, foi escolhido retirar os valores improváveis da base.

Tabela 3.3: Contagem dos resultados de saída

Resultado	Contagem	Porcentagem
0	134334	51,779
2	78917	30,418
3	30117	11,608
1	14113	5,4398
4	1512	0,5828
5	383	0,1476
6	46	0,0177
7	10	0,0038
8	3	0,0012
9	3	0,0012
11	1	0,0004

Os valores indicados como raros, ou seja, os resultados acima de 3 pontos, foram unificados, sendo todos representados por um valor de resultado igual a 4.

Ainda deve-se observar que existe uma diferença na frequência dos valores possíveis para a variável de resultado de posse de bola. Portanto, ao inseri-la no treinamento da rede é feita uma equalização em sua frequência de acordo com o modelo aplicado. Tal implementação é explicada de forma detalhada junto à definição de cada modelo nas seções 3.3 e 3.4.

Nesta etapa também foi observado que diversos atributos seriam irrelevantes para melhorar a capacidade de generalização dos modelos, como as variáveis puramente identificadoras, tais como nome e id do time, número da temporada e número da posse de bola, entre outros. Portanto, estes atributos foram previamente retirados.

3.2.1 Seleção de variáveis

Para maximizar o rendimento dos modelos, excluindo atributos possivelmente irrelevantes e para identificar os atributos mais significativos, foi feita uma seleção de variáveis.

Primeiramente foi feita uma análise qualitativa dos dados de entrada do modelo com o auxílio de especialistas em basquete, e com amplo conhecimento da base de dados. Dessa forma foi possível ter uma melhor compreensão das variáveis utilizadas, e uma pré-seleção fora montada.

Foi escolhido um método "*filter*" principalmente por conta da falta de definição dos modelos que seriam utilizados no início do projeto, impossibilitando a utilização de um modelo "*wrapper*" ou "*embedding*".

Há que se considerar ainda que algumas variáveis não podem ser retiradas, tais como os *clusters* dos jogadores em quadra. Isto porque estas variáveis são as mais significativas em termos práticos de jogo, pois agregam a capacidade de aplicação dos resultados à situação real de jogo.

Para a avaliação das demais variáveis, foi utilizado o método Relief-F, introduzido no Capítulo 2. Os resultados podem ser observados na Tabela 3.4

Tabela 3.4: Tabela de avaliação das variáveis pelo método Relief-F

Avaliação	N°/Atributo	Avaliação	N°/Atributo
0,00116809	2 match_elapsed_minutes	0,00014175	7 away_period_fouls
0,00030892	35 away_fouls_players	0,00014172	27 away_quintet_turnovers
0,00025906	23 away_quintet_blocks	0,0001363	10 home_quintet_rebounds
0,00024413	26 away_quintet_fouls	0,00013364	29 away_quintet_minutes
0,00024037	5 home_score_difference	0,00013351	21 away_quintet_rebounds
0,00022499	6 home_period_fouls	0,00012758	20 away_quintet_assists
0,00022024	9 home_quintet_assists	0,00012703	4 away_score
0,00020095	22 away_quintet_steals	0,00012381	3 home_score
0,00019774	34 home_fouls_players	0,00008988	12 home_quintet_blocks
0,00019605	24 away_quintet_missed_throws	0,0000847	13 home_quintet_missed_throws
0,00019266	36 home_minutes_players	0,00005402	1 period
0,0001892	16 home_quintet_turnovers	0,00005311	19 away_quintet_points
0,00018386	8 home_quintet_points	0,00002301	33 away_eff_per_min_players
0,00018359	18 home_quintet_minutes	0,00001433	17 home_quintet_eff_per_min
0,0001759	31 away_points_players	0,00000945	28 away_quintet_eff_per_min
0,00016439	37 away_minutes_players	0,00000574	32 home_eff_per_min_players
0,00015587	30 home_points_players	0,00000224	14 home_quintet_missed_free_throws
0,00015463	11 home_quintet_steals	-0,00004588	25 away_quintet_missed_free_throws
0,00014462	15 home_quintet_fouls		

Dessa forma, pode-se observar que o método indica que alguns atributos apresentam uma boa relevância, tal como "match_elapsed_minutes". Já para outros, como "home_quintet_missed_free_throws" e "away_quintet_missed_free_throws", pode-se observar que o método indica que possuem baixa relevância em relação à variável de saída.

Por fim, foi priorizado manter variáveis do jogo, como "match_elapsed_minutes" e "home_score_difference", entre outras variáveis que representam o andamento do jogo. Também foram utilizadas as variáveis relacionadas aos quintetos, dado que com a representação do jogador por meio de *clusters*, nem sempre os dados individuais serão representativos para um grupo visto que vários jogadores diferentes podem ser representados por ele. Esta análise permite avaliar posteriormente a retirada de variáveis como "period", "home_score" e "away_score", visto a pouca relevância em

relação ao “home_score_difference”. Esta diferença pode ser explicada pela possível influência que, não o placar em si, mas a diferença no placar pode causar nos jogadores. Pode ser observado na Tabela 3.5 as variáveis de entrada para os modelos desenvolvidos neste projeto.

Tabela 3.5: Variáveis de entrada para os modelos

Variáveis
“period”
“match_elapsed_minutes”
“home_score”
“away_score”
“home_score_difference”
“home_period_fouls”
“away_period_fouls”
“home_quintet_points”
“home_quintet_assists”
“home_quintet_rebounds”
“home_quintet_steals”
“home_quintet_blocks”
“home_quintet_missed_throws”
“home_quintet_missed_free_throws”
“home_quintet_fouls”
“home_quintet_turnovers”
“home_quintet_eff_per_min”
“home_quintet_minutes”
“away_quintet_points”
“away_quintet_assists”
“away_quintet_rebounds”
“away_quintet_steals”
“away_quintet_blocks”
“away_quintet_missed_throws”
“away_quintet_missed_free_throws”
“away_quintet_fouls”
“away_quintet_turnovers”
“away_quintet_eff_per_min”
“away_quintet_minutes”
“possessions”
“elapsed_minutes”
Variáveis dos clusters

3.3 Implementação do 1° estágio do sistema (classificação)

Diferentemente de um modelo tradicional que busca prever o resultado de cada posse de bola individualmente, neste estágio a ênfase está em prever o saldo de pontos resultante de sequências específicas de posses de bola, com foco nas disposições em que não houve substituição de jogadores. O basquete é um jogo dinâmico e altamente equilibrado, no qual a vantagem técnica de um dos times nem sempre corresponde a um resultado imediato, seja em forma de cesta ou de defesa bem-sucedida. Portanto, após a realização de alguns testes, o desenvolvimento deste estágio se baseia na ideia central de avaliar o impacto conjunto de uma série de posses de bola na dinâmica do jogo a fim de obter consistentemente a vantagem técnica de um time sobre outro e, portanto, a previsão do resultado para seu enfrentamento. Pelo mesmo motivo, foi adotada uma heurística que considera haver um empate técnico, quando há uma pequena diferença de pontos entre os times. Ou seja, ao utilizar a rede para prever os resultados, foi adotado que uma cesta de dois pontos, devido à frequência de ocorrência durante os jogos, não seria considerada vantagem, portanto, os modelos interpretam o saldo a partir de 3 pontos de diferença.

Para isso, foi desenvolvido um algoritmo de classificação que busca prever o resultado de um saldo de pontos ocorrido durante uma determinada sequência de posses de bola. A classificação é feita em três classes: vantagem (3 ou mais pontos de diferença), desvantagem (3 ou mais pontos de diferença para o adversário) ou empate (até 2 pontos a mais ou a menos). Este é um cenário de classificação multi-classe, no qual o algoritmo aprende a associar as características iniciais da primeira posse de bola da sequência, considerando tanto os jogadores que estão em quadra quanto as estatísticas de início da sequência à classe correspondente ao saldo de pontos.

Conforme mencionado anteriormente, a representatividade das classes nos dados não é equitativa, exigindo uma estratégia para balancear a distribuição. Dada a magnitude do conjunto de dados após os processos executados previamente, optou-se por uma técnica de subamostragem para equalizar a representatividade das classes. Isso envolve a eliminação de algumas entradas das classes super-representadas, promovendo assim uma distribuição mais uniforme para aplicação nos modelos.

A Figura 3.3 apresenta a distribuição original das classes observadas na base de dados, enquanto as Figuras 3.4, 3.5 e 3.6 ilustram a distribuição dos dados de treinamento, validação e teste, respectivamente, antes e após a implementação desta estratégia de balanceamento. Pode-se observar na Figura 3.5 que o conjunto de validação é aproximadamente 10% dos dados

de treinamento e observa-se que no grupo de teste (Figura 3.6) não é feito balanceamento, visando representar devidamente o universo testado.

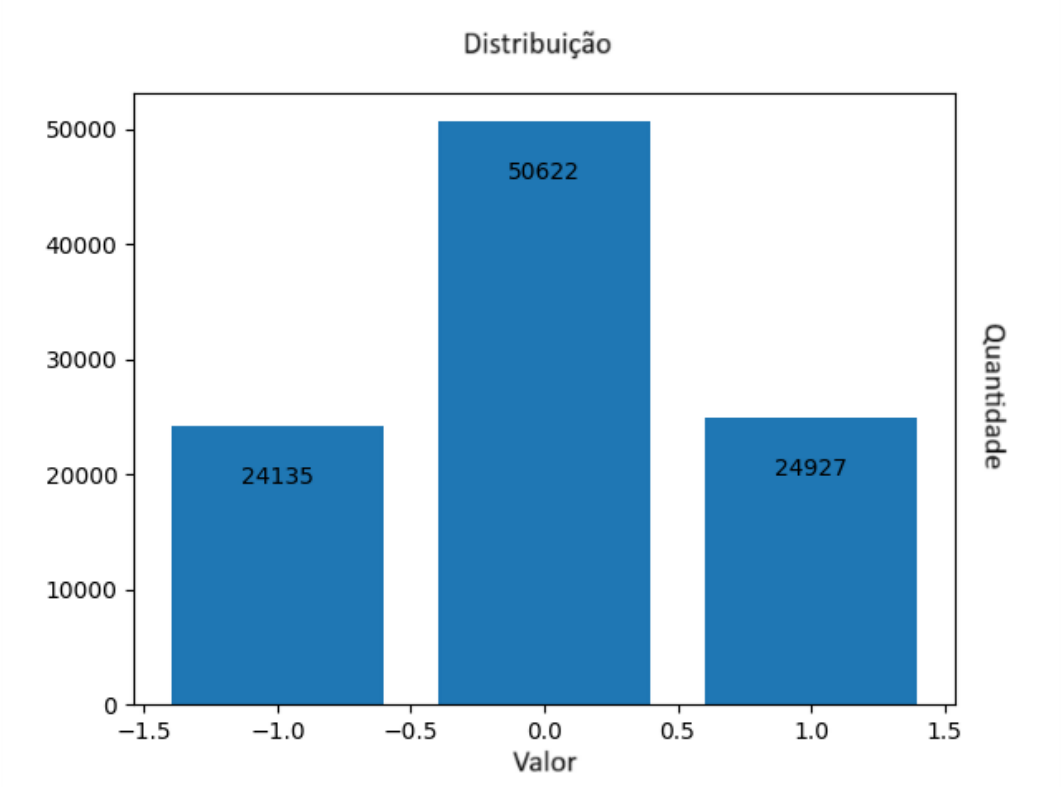

Figura 3.3: Distribuição dos dados antes do balanceamento entre as classes.

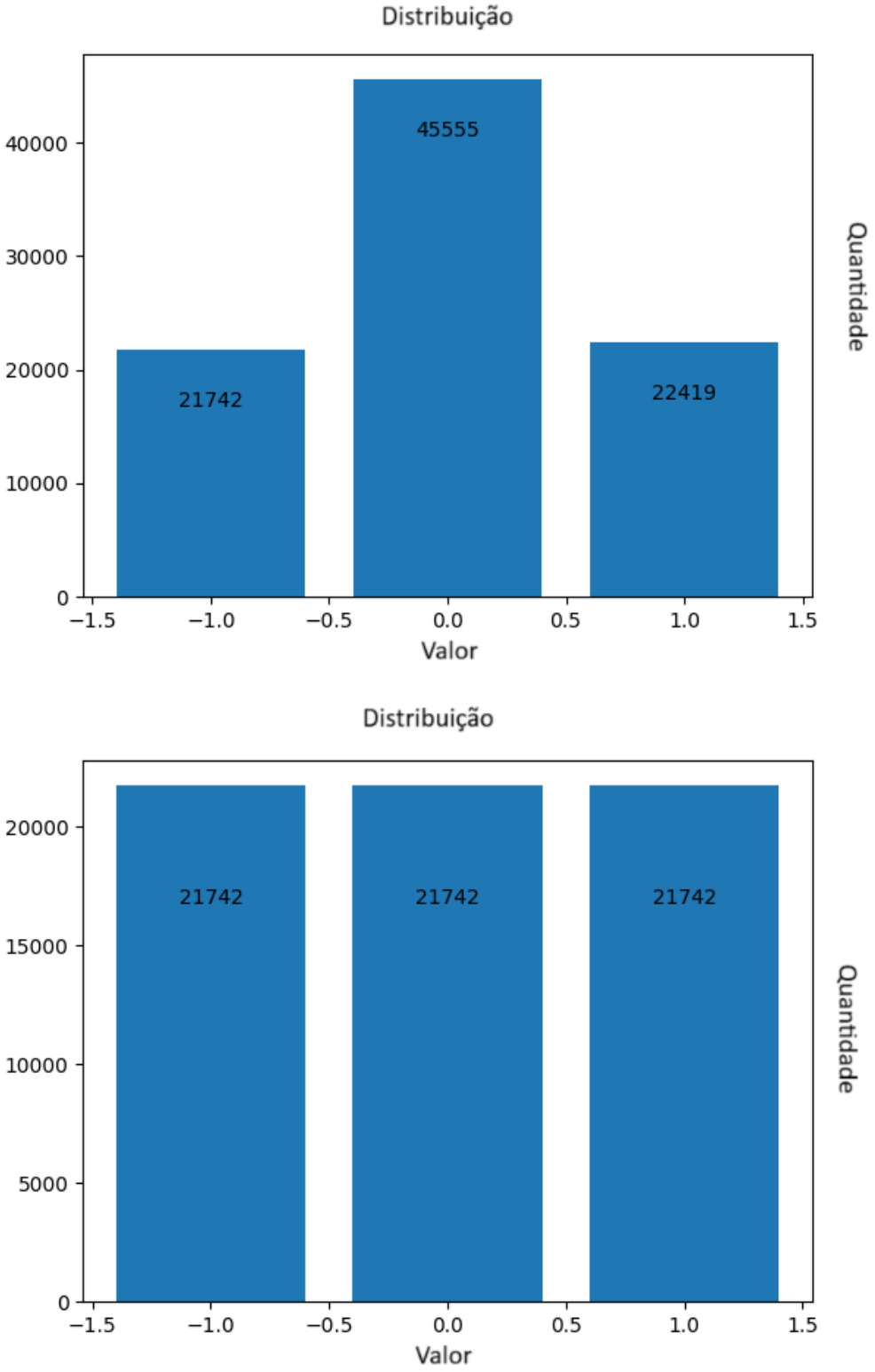

Figura 3.4: Comparação da distribuição de dados de treinamento antes e após o balanceamento entre as classes, respectivamente.

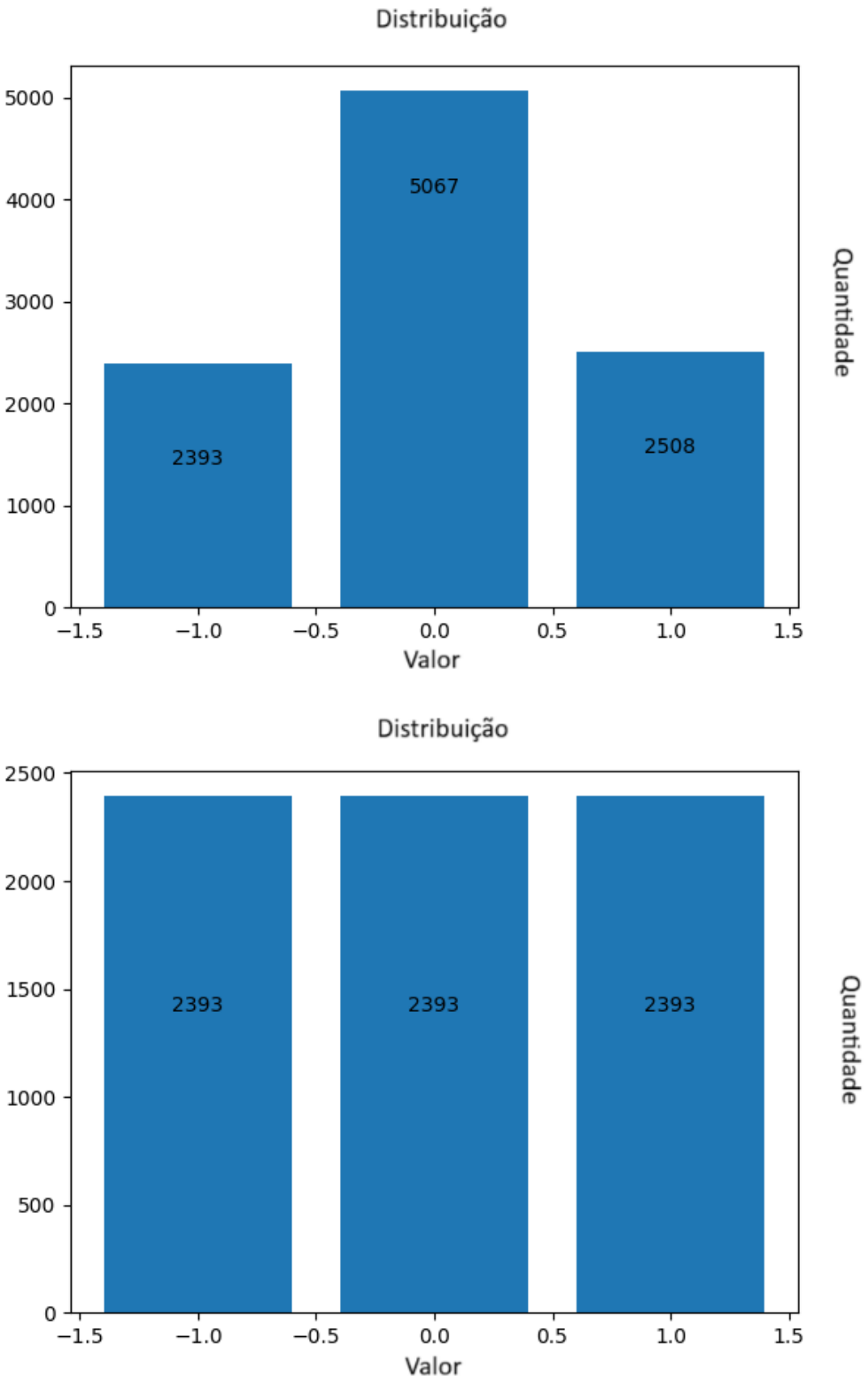

Figura 3.5: Comparação da distribuição de dados de validação antes e após o balanceamento entre as classes, respectivamente.

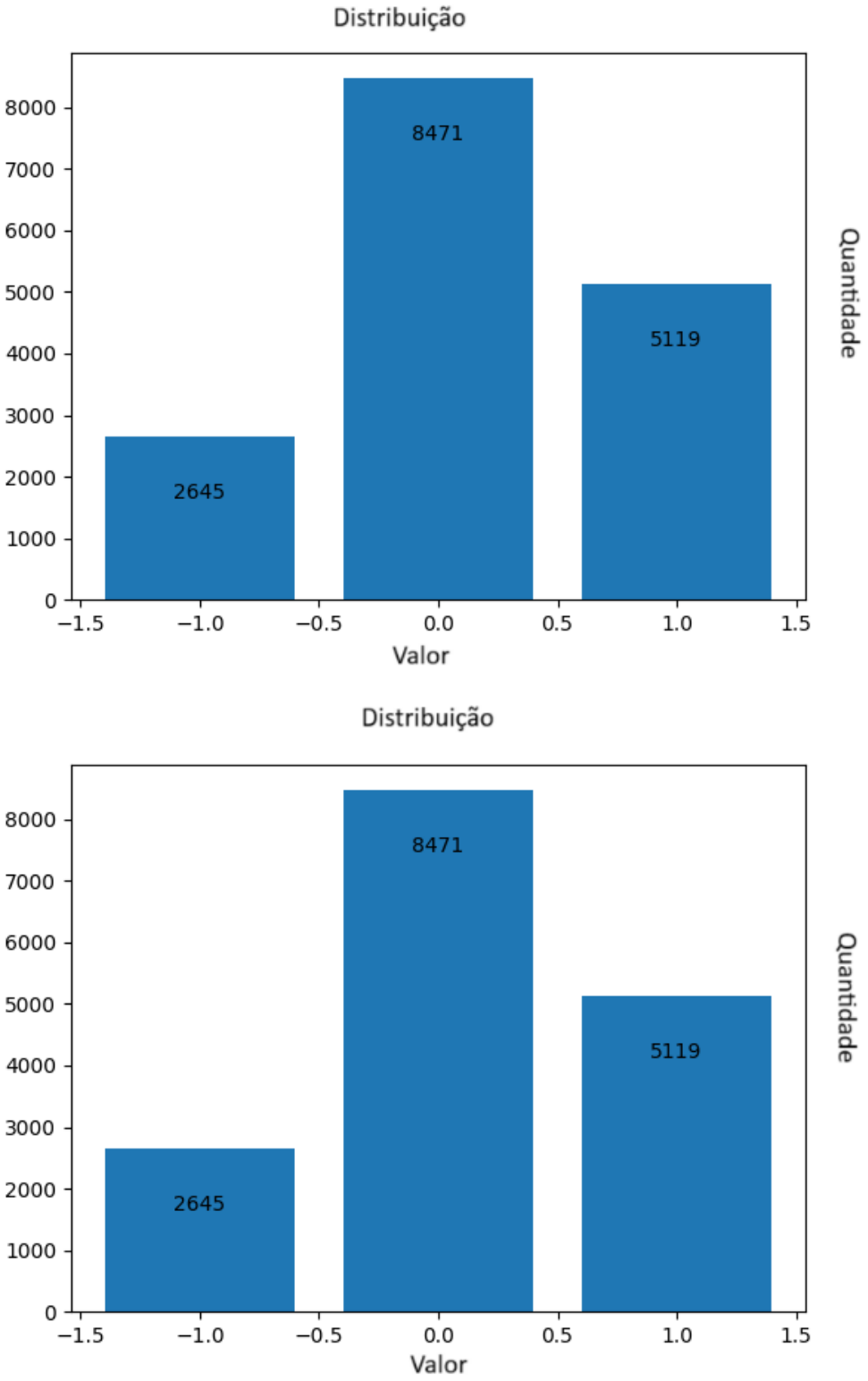

Figura 3.6: Comparação da distribuição de dados de teste antes e após o balanceamento entre as classes, respectivamente.

A implementação deste algoritmo de classificação foi baseada na utilização de dois métodos robustos de aprendizado de máquina: as Redes Neurais Artificiais e o algoritmo kNN. A escolha desses métodos se deu por sua capaci-

dade de lidar com problemas de classificação multi-classe e por sua flexibilidade para se adaptar a diferentes estruturas de dados (HAN; PEI; TONG, 2022).

A rede neural foi treinada com diferentes combinações de camadas ocultas e com função de ativação ReLU, enquanto o kNN foi implementado com a otimização do número de vizinhos por meio de uma busca exaustiva, baseada em validação cruzada.

A avaliação do desempenho desses modelos foi feita através principalmente da métrica acurácia. Para obter o melhor desempenho na aplicação da Rede Neural, as camadas e seus neurônios foram escolhidos utilizando um algoritmo de busca exaustiva para varrer as principais opções de combinação utilizando o próprio resultado da acurácia da validação como avaliação.

Como método secundário de avaliação, para efeito visual, foram utilizadas matrizes de confusão.

A matriz de confusão é uma poderosa ferramenta para a avaliação do desempenho de modelos de aprendizado de máquina, particularmente útil para a análise mais aprofundada de modelos de classificação. Ela permite visualizar o desempenho do modelo em suas tarefas de classificação, mostrando claramente a relação de acertos e erros entre as classes verdadeiras e as classes previstas pelo modelo (MONARD; BARANAUSKAS, 2003).

A matriz de confusão pode ser interpretada da seguinte forma:

Verdadeiros Positivos (TP): estas são as situações em que a rede neural previu corretamente a vitória. Em outras palavras, a previsão do modelo estava correta e a verdadeira classe dos dados era de vitória.

Verdadeiros Negativos (TN): nestes casos, o modelo previu corretamente a derrota. Isso significa que a previsão do modelo estava correta e a verdadeira classe dos dados era de derrota.

Falsos Positivos (FP): aqui, a rede neural previu uma vitória, quando na realidade, a classe verdadeira dos dados era de derrota. Estas são situações em que o modelo errou a previsão.

Falsos Negativos (FN): nestes casos, o modelo previu uma derrota, quando a verdadeira classe dos dados era de vitória. Assim como os falsos positivos, estas são situações em que a previsão do modelo estava errada.

Através da matriz de confusão, é possível também calcular métricas adicionais que auxiliam na avaliação do desempenho do modelo, como a precisão, o recall, o F1-score, entre outras. Essas métricas oferecem uma visão mais ampla e completa do desempenho do modelo, complementando a análise e interpretação da matriz de confusão.

Analisar a matriz de confusão para a melhor configuração da rede neural permite entender melhor onde e como o modelo está acertando e errando em

suas previsões, oferecendo uma boa visibilidade para aprimoramentos e ajustes futuros.

Em resumo, a implementação deste algoritmo de classificação representou um avanço significativo em relação aos modelos tradicionais de previsão no basquete, ao considerar a interação entre sequências consecutivas de posses de bola e o impacto dessas sequências no saldo de pontos. Este método permitiu uma melhor compreensão da dinâmica do jogo e uma previsão mais acurada do desempenho das equipes em diferentes configurações de jogo.

Os resultados deste estágio são apresentados e discutidos em 4.1.

3.4 Implementação do 2° estágio do sistema (previsão)

Na continuidade do processo de construção do sistema preditivo, o segundo estágio da rede foi projetado para agregar a previsão obtida no primeiro estágio. Enquanto a primeira fase foca na determinação do resultado (vantagem, desvantagem ou igualdade) das posses de bola, a segunda etapa busca quantificar a magnitude da classe prevista.

Essa fase emprega a mesma estrutura de Rede Neural e o algoritmo kNN usados no primeiro estágio, porém com ajustes adequados para atender a natureza numérica do problema. O objetivo agora é, partindo da informação entregue pelo estágio 1 de quem terá a vantagem, prever qual será a intensidade dessa vantagem, traduzida pela diferença de pontos.

A principal distinção na implementação do segundo estágio é a inclusão da variável que denota o número de posses de bola em cada sequência. Essa variável é crítica, pois permite ao modelo capturar a tendência de um maior diferencial de pontos em sequências mais longas comparado a sequências mais curtas. Assim, o modelo aprende a prever uma vantagem calibrada de acordo com o comprimento da sequência de posses de bola esperado.

Com base na classificação do estágio anterior, o segundo estágio fornece uma estimativa da magnitude da classe prevista. A segunda fase do sistema acrescenta uma camada adicional de informação às previsões, permitindo uma melhor avaliação da confiabilidade das previsões do primeiro estágio. Em resumo, essa implementação em duas etapas permite uma análise mais detalhada e refinada do desempenho da equipe, oferecendo não apenas uma previsão do vencedor de um conjunto de posses, mas também uma estimativa de quão grande essa vitória pode ser. Esta abordagem aprimorada pode ter um impacto significativo nas estratégias de equipe e na tomada de decisões durante os jogos de basquete.

É observado e constatado que na distribuição dos valores dos pontos

há uma grande porcentagem dos dados com valores pequenos de saldo de pontos, o que faz sentido para o basquete por ser equilibrado. Para adequar a distribuição dos resultados presentes em um universo numérico é utilizada uma normalização linear por partes, considerando para cada valor ou cada pequeno grupo de valores na reta convertida a mesma proporção que representa na base de dados.

O processo de normalização linear por partes consiste em dividir a distribuição dos dados em segmentos (ou "partes") e aplicar uma transformação linear distinta a cada segmento. Isso permite que a distribuição dos dados seja preservada ao máximo, mantendo a proporção de cada valor ou pequeno grupo de valores na reta convertida. Como exemplo pode-se observar a Figura 3.7, na qual existe uma base de dados cuja maior quantidade de informações se concentram em valores baixos. Neste exemplo percebe-se que ao ser aplicada a normalização linear por partes, os dados de menor valor são representados por uma faixa maior de valores no espaço normalizado enquanto os dados de maior valor, no entanto menos presentes na base, se encontram representados por um espaço mais estreito no espaço normalizado. Isso permite que a quantidade de informações concentrada nos valores baixos possam ser representadas com mais detalhes enquanto os dados mais escassos, que não necessitam de tamanho detalhamento, passam a ser representados por um espaço menor. Desta forma a representação do dado original se torna mais equilibrada.

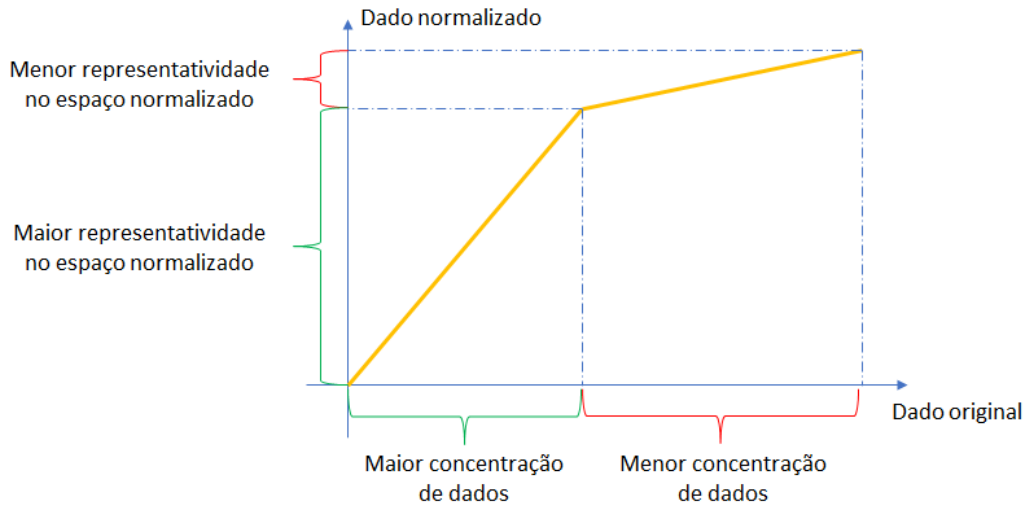

Figura 3.7: Exemplificação do funcionamento da normalização linear por partes.

Para implementar essa técnica, iniciou-se observando a distribuição do saldo de pontos, identificando onde a maior parte dos dados está concentrada. Como observado, a grande maioria dos dados encontra-se em regiões de saldo

de pontos baixos. Portanto, estes dados foram reajustados para se relacionar à maior porção na representação do espaço de valores normalizado.

A utilização da normalização linear por partes resulta em um melhor ajuste da distribuição dos dados, permitindo que a Rede Neural e o algoritmo kNN possam interpretar mais efetivamente a variação dos dados. Além disso, também permite que a normalização seja inversamente aplicada aos resultados previstos pelo modelo, permitindo uma interpretação mais intuitiva e direta dessas previsões no contexto do basquete.

As Figuras 3.8 e 3.9 ilustram a distribuição dos dados e a diferença no espaço de representação dos valores na aplicação da normalização linear por partes, demonstrando a eficácia desta técnica na equalização da distribuição do saldo de pontos. Pode ser observado que, devido à distribuição original dos dados, os valores [0, 0.8] das pontuações 0 e 10 têm maior espaço para representar a distribuição dos valores originais e entre 10 e 30 uma escala mais estreita para representar esses valores maiores que 10. Ao lidar com a distribuição assimétrica dos dados de maneira eficaz, a normalização linear por partes contribui para o melhor desempenho dos modelos de aprendizado de máquina implementados.

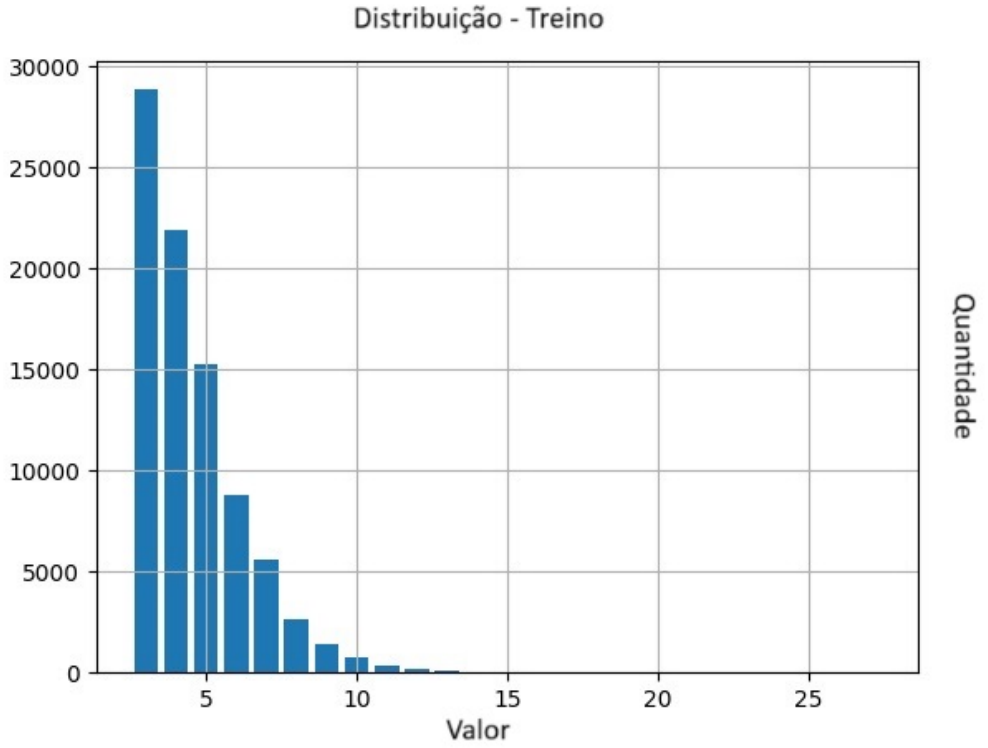

Figura 3.8: Distribuição numérica.

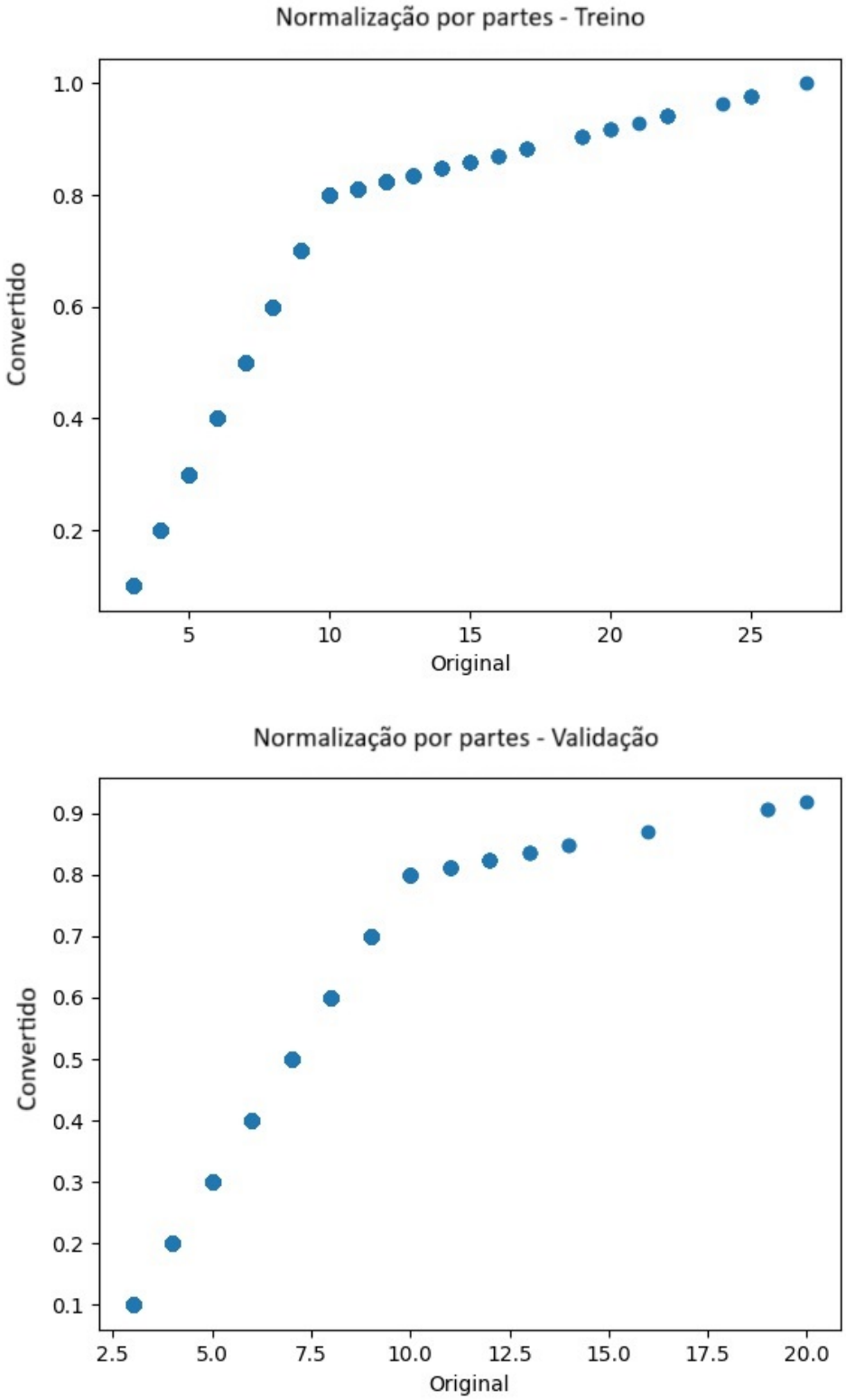

Figura 3.9: Gráfico de mapeamento de normalização por partes para o grupo de treino e validação, respectivamente.

Para realizar a avaliação da qualidade do modelo, bem como sua evolução, foi utilizada uma métrica essencial para validação de rendimento de algoritmos de previsão numérico, o MAPE.

Segundo (MYTTENAERE et al., 2016), o MAPE é uma medida de erro usada para determinar a acurácia das previsões numéricas. Como pode ser observado na Equação 3-1, o MAPE é calculado como a média das diferenças percentuais absolutas entre a previsão e o valor real, proporcionando uma ideia da magnitude do erro, sem considerar a direção. Um valor menor de MAPE indica maior acurácia na previsão do modelo. Como desvantagem essa métrica não é "simétrica" para valores de previsão.

$$MAPE = \left(\frac{1}{n}\right) \cdot \sum_{i=1}^{n} \left(\frac{|verdadeiro - previsto|}{|verdadeiro|}\right) \cdot 100 \qquad (3\text{-}1)$$

Para melhorar a visibilidade das previsões realizadas e compreender como as previsões estão alinhadas com os valores reais, foram utilizados diagramas de dispersão. Os diagramas de dispersão oferecem uma representação visual clara da relação entre as previsões do modelo e os resultados reais.

Em um diagrama de dispersão, cada ponto representa uma previsão feita pelo modelo. O eixo X representa o valor previsto, enquanto o eixo Y representa o valor real. Se todas as previsões estivessem perfeitamente alinhadas com os valores reais, todos os pontos estariam sobre uma linha diagonal, indo do canto inferior esquerdo ao canto superior direito. Desvios dessa linha diagonal representam erros nas previsões.

Isso permite identificar rapidamente onde o modelo está acertando e errando, bem como qualquer tendência sistemática nos erros do modelo. Por exemplo, se os pontos estão principalmente abaixo da linha diagonal, isso significa que o modelo tem uma tendência a subestimar os resultados. Por outro lado, se os pontos estão principalmente acima da linha, isso significa que o modelo tem uma tendência a superestimar os resultados.

O uso de diagramas de dispersão também permite verificar se o modelo é mais acurado para certos valores do que para outros. Por exemplo, o modelo pode ser muito acurado ao prever resultados próximos da média, mas menos acurado ao prever resultados muito altos ou muito baixos. Este tipo de informação é extremamente útil para melhorar e refinar o modelo.

Os resultados deste estágio são apresentados e discutidos em 4.2.

3.5 Aplicação do sistema

Considerando um jogo de basquete, como proposta de uso da metodologia, ao se inserir os dados iniciais de uma determinada posse de bola no sistema, é feita uma experimentação por meio de busca exaustiva combinando todas as possibilidades de jogadores disponíveis em quadra e no banco do time desejado

contra o time adversário em quadra. Para cada escalação possível observada, o primeiro estágio apresenta aquelas em que que o algoritmo prevê vantagem nas próximas posses de bola. Em seguida, o segundo estágio do sistema prevê o resultado de uma sequência de posses de bola para cada combinação classificada como vantagem pelo estágio anterior. Desta forma, pode-se formar uma tabela apresentando todas as combinações de escalação favoráveis de acordo com o primeiro estágio da rede, ordenadas por maior diferença de pontos esperada pelo segundo estágio.

Os resultados desta aplicação são apresentados e discutidos em 4.3.

4
Estudo de caso

Como visto anteriormente, os resultados obtidos a partir da união dos dados em sequências segundo o processo explicado na Figura 3.2 formaram uma base de dados separada para cada tamanho de sequência, em termos de posses de bola, cujo exemplo pode ser observado na Figura 4.1. Um exemplo da estrutura dos dados após a unificação destas bases de dados, chamada por "Agrupamento", é mostrado na Figura 4.2. Como pode ser observado, foram escolhidas sequências desde 8 até 15 posses de bola para formar a base completa, visto que sequências muito pequenas não representam bem a consistência de uma superioridade técnica, enquanto sequências muito longas são improváveis em um jogo.

Base de janelas de 5 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
5	X no Início da 1ª posse	9
5	X no Início da 1ª posse	3
...	...	...

Base de janelas de 6 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
6	X no Início da 1ª posse	4
6	X no Início da 1ª posse	10
...	...	...

Base de janelas de 7 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
7	X no Início da 1ª posse	10
7	X no Início da 1ª posse	8
...	...	...

Base de janelas de 8 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
8	X no Início da 1ª posse	7
8	X no Início da 1ª posse	16
...	...	...

Base de janelas de 9 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
9	X no Início da 1ª posse	8
9	X no Início da 1ª posse	17
...	...	...

Base de janelas de 10 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
10	X no Início da 1ª posse	13
10	X no Início da 1ª posse	10
...	...	...

Base de janelas de 11 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
11	X no Início da 1ª posse	13
11	X no Início da 1ª posse	20
...	...	...

Base de janelas de 12 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
12	X no Início da 1ª posse	17
12	X no Início da 1ª posse	15
...	...	...

Base de janelas de 13 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
13	X no Início da 1ª posse	21
13	X no Início da 1ª posse	16
...	...	...

Base de janelas de 14 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
14	X no Início da 1ª posse	25
14	X no Início da 1ª posse	19
...	...	...

Base de janelas de 15 posses de bola

Posses de bola	Atributos de entrada	Saldo de pontos
15	X no Início da 1ª posse	20
15	X no Início da 1ª posse	29
...	...	...

Figura 4.1: Exemplificação da formação das bases de dados que contém sequências de posses de bola de mesma quantidade de posses de bola por sequência.

Base de janelas de 8 a 15 posses de bola		
Posses de bola	Atributos de entrada	Saldo de pontos
8	X no Início da 1ª posse	7
8	X no Início da 1ª posse	14
9	X no Início da 1ª posse	14
9	X no Início da 1ª posse	8
10	X no Início da 1ª posse	13
10	X no Início da 1ª posse	17
11	X no Início da 1ª posse	14
11	X no Início da 1ª posse	16
12	X no Início da 1ª posse	16
12	X no Início da 1ª posse	15
13	X no Início da 1ª posse	19
13	X no Início da 1ª posse	24
14	X no Início da 1ª posse	19
14	X no Início da 1ª posse	23
15	X no Início da 1ª posse	27
15	X no Início da 1ª posse	22
...	...	...

Figura 4.2: Exemplificação da unificação das bases de dados que contém sequências de posses de bola de mesma quantidade de posses de bola por sequência, chamada de "Agrupamento".

Para executar o treinamento dos modelos e as previsões desejadas da melhor forma possível, em cada etapa foram testadas variações dentro destas estruturas de dados, cuja evolução e resultados são mostrados a seguir.

4.1 Resultados do 1° estágio do sistema

Para buscar arquiteturas de modelos baseados em rede neural, recorreu-se a um algoritmo de busca exaustiva para encontrar a melhor combinação de camadas ocultas. Isso foi feito variando de 5 em 5 neurônios por camada, indo de 10 a 80 neurônios para uma única camada oculta, e de 10 a 60 neurônios para duas camadas ocultas, sendo que, por heurística, a segunda camada foi limitada ao tamanho da primeira (HEATON, 2008).

Durante este processo, foi adotada uma paciência de 500 e um máximo de 20000 épocas. Esses parâmetros foram escolhidos devido a uma peculiaridade da biblioteca do "*Keras*", que não retorna o melhor conjunto de pesos se a execução for interrompida pelo limite máximo de épocas, e não pela condição de paciência. Portanto, esses parâmetros foram ajustados para garantir que o algoritmo tivesse tempo suficiente para encontrar os melhores pesos antes de atingir o limite de épocas.

Para garantir a consistência dos resultados, cada teste de janela foi repetido 10 vezes. A acurácia média dessas 10 execuções é então calculada e

registrada na Tabela 4.1. Isso oferece uma visão mais robusta do desempenho da estrutura de rede e do tamanho da janela escolhida.

Tabela 4.1: Relação de acurácia média e melhor acurácia obtida com a otimização da rede neural para cada dado de entrada desde a sequência com 5 posses de bola até a sequência de 15 posses e o agrupamento,

Melhores camadas	Melhor acurácia média	Melhor acurácia	Tamanho da sequência (posses)	Entradas	Amostras
[50, 30]	0,398	0,440	5,0	61,0	25686,0
[50, 40]	0,444	0,474	6,0	61,0	20162,7
[70]	0,471	0,517	8,0	61,0	13547,7
[55, 40]	0,678	0,716	10,0	61,0	5974,2
[60, 40]	0,681	0,712	11,0	61,0	5223,6
[55, 30]	0,691	0,724	13,0	61,0	2130,3
[50, 40]	0,709	0,756	15,0	61,0	1548,0
[50, 40]	0,716	0,755	14,0	61,0	1802,7
[80]	0,724	0,75	12,0	61,0	4383,9
[60, 40]	0,753	0,762	Agrupamento	61,0	40794,3

Observou-se uma tendência de acurácia crescente com a expansão da janela de análise. Ou seja, aumentar o número de posses de bola na janela resultava em uma previsão mais acurada. Contudo, a acurácia máxima foi atingida com o chamado "Agrupamento", que agrupa todas as janelas de análise. Isso indica que uma análise mais extensa do desempenho de uma equipe fornece uma previsão mais acurada do saldo de pontos.

Nesta etapa, a melhor configuração da rede neural foi obtida com a combinação de camadas ocultas [60, 40], conforme apresentado na Tabela 4.1. A tabela apresenta a acurácia de validação, com a qual a arquitetura de rede é escolhida. A acurácia obtida para o grupo de teste na dada configuração foi de 71,82%. A matriz de confusão nesta configuração é apresentada na Figura 4.3.

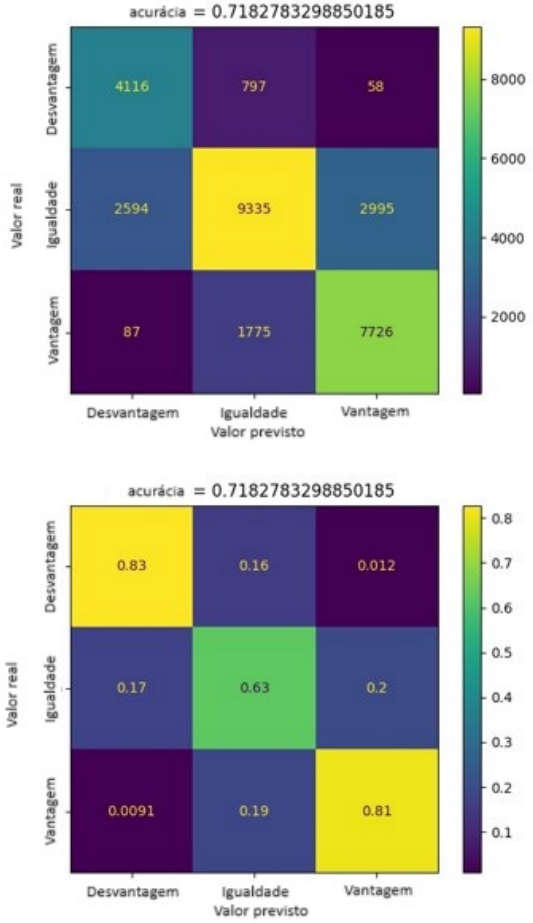

Figura 4.3: Matriz de confusão para a previsão do grupo de teste obtido a partir da melhor configuração da rede neural de acordo com a otimização.

Esses resultados podem ser explicados pela suposição de que o desempenho de uma equipe torna-se mais aparente após um período mais longo em quadra. No entanto, o "Agrupamento" apresenta não apenas os dados de maiores posses de bola para análise, mas também um conjunto de dados significativamente maior, o que provavelmente contribui para a acurácia do modelo.

Apesar do bom desempenho de redes de uma camada oculta em dois casos na otimização, no uso de uma quantidade de dados bem maior, como é o caso do agrupamento, que fornece a maior acurácia obtida, a rede de duas camadas ocultas, com 60 e 40 neurônios respectivamente, se mostra superior e, portanto, é escolhida para comparação com os resultados da kNN neste estágio.

A abordagem escolhida para otimizar o kNN foi a utilização do método *Grid Search Cross-Validation*, uma técnica comum e eficaz para ajuste de hiperparâmetros (ADNAN et al., 2022). *Grid Search* é um método de otimização que busca a melhor combinação de parâmetros dentro de um conjunto pré-definido de opções. Aplica-se, por exemplo, para encontrar o número ideal de vizinhos no algoritmo kNN, bem como outras possíveis configurações.

Em ambos os processos foi utilizada a técnica de *Cross-Validation*, ou validação cruzada, caracterizada por subdividir o conjunto de dados em subconjuntos menores, ou "folds". O modelo é então treinado em k-1 desses *folds* e validado no k *fold* restante. Este processo é repetido várias vezes, cada vez com um conjunto diferente de *folds* para treinamento e validação. A validação cruzada oferece uma visão mais robusta do desempenho do modelo, já que avalia a capacidade do modelo de generalizar a partir de diferentes subconjuntos do conjunto de dados.

Nesse caso, utilizou-se uma validação cruzada com 10 *folds*. Assim, a base de dados foi dividida em 10 subconjuntos, e o modelo foi treinado e validado 10 vezes, usando um subconjunto diferente para validação a cada vez que um novo modelo é criado a partir dos outros 9 *folds*.

Desta forma, ao realizar a otimização do kNN através do *Grid Search Cross-Validation*, foi possível identificar a melhor configuração de parâmetros para este modelo específico, considerando a base de dados. Isso permitiu ajustar o modelo para maximizar a sua acurácia, adequando-o melhor às características e tendências dos dados.

Para apresentar os resultados obtidos das otimizações realizadas acima, é feito o uso de matrizes de confusão conforme citado anteriormente.

O kNN, por sua vez, apresentou um desempenho levemente superior. A acurácia obtida para o mesmo grupo de teste na dada configuração foi de 73,92%. A matriz de confusão para esta configuração é apresentada na Figura 4.4.

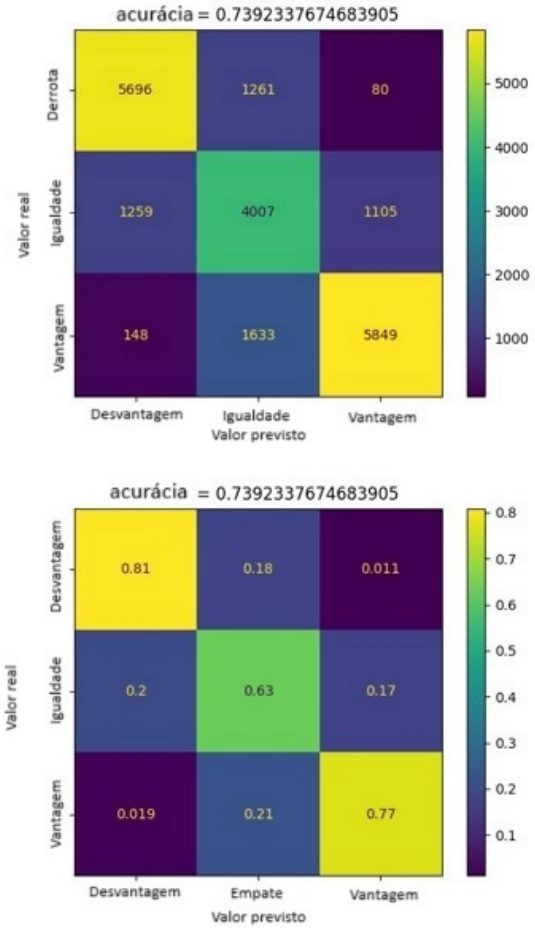

Figura 4.4: Matriz de confusão para a previsão do grupo de validação obtido a partir da otimização do kNN.

Após a otimização, foi feita ainda uma variação do grupo de dados utilizado para compor a entrada da rede. Este teste buscou abranger as janelas mais importantes segundo a distribuição observada anteriormente, portanto continha as janelas de 5 posses de bola até 15 posses de bola. Os resultados observados ao longo do processo, mostram a tendência de melhores resultados para as janelas com maior quantidade de posses de bola do que previsão com janelas de menor quantidade de posses de bola. No entanto, considerar apenas uma por vez limitava demais a quantidade de dados disponível, portanto, foram feitos novos testes considerando agrupamentos de janelas com mais posses de bola para avaliar uma possível melhora da acurácia do modelo.

Como pode ser observado nos resultados apresentados nas Figuras 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, houve uma melhora significativa na utilização de janelas de maior quantidade de posse de bola, chegando a um auge de quase 77% de acurácia.

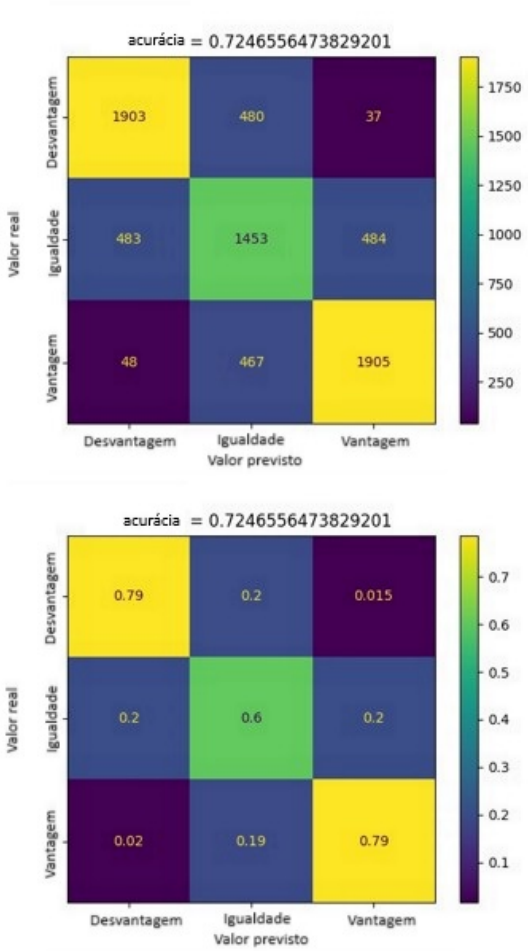

Figura 4.5: Matriz de confusão para a previsão do grupo de validação obtido a partir da melhor configuração da rede neural de acordo com a otimização, utilizando o agrupamento de janelas de 8 a 15.

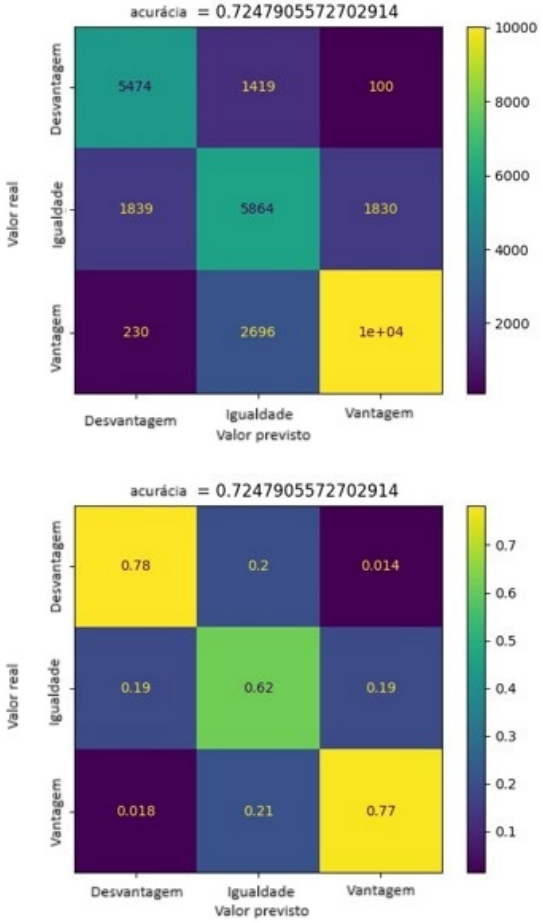

Figura 4.6: Matriz de confusão para a previsão do grupo de teste obtido a partir da melhor configuração da rede neural de acordo com a otimização, utilizando o agrupamento de janelas de 8 a 15.

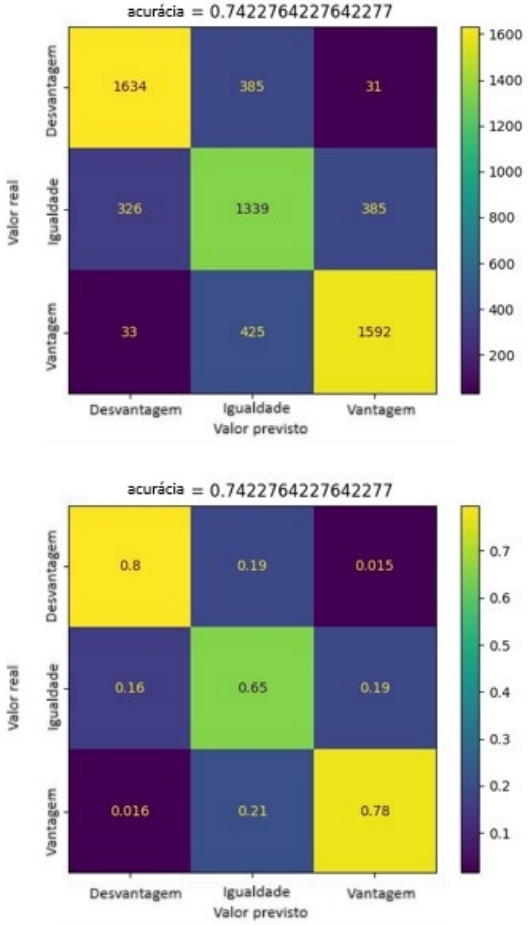

Figura 4.7: Matriz de confusão para a previsão do grupo de validação obtido a partir da melhor configuração da rede neural de acordo com a otimização, utilizando o agrupamento de janelas de 9 a 15.

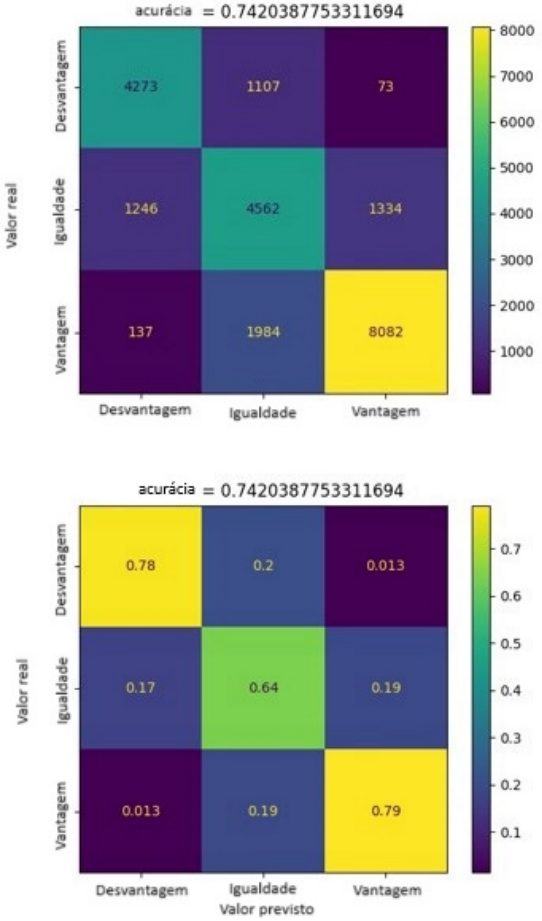

Figura 4.8: Matriz de confusão para a previsão do grupo de teste obtido a partir da melhor configuração da rede neural de acordo com a otimização, utilizando o agrupamento de janelas de 9 a 15.

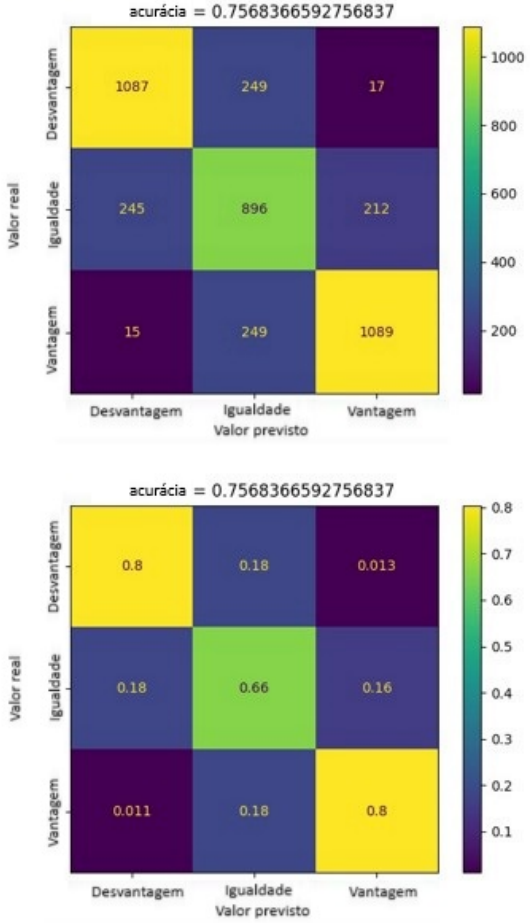

Figura 4.9: Matriz de confusão para a previsão do grupo de validação obtido a partir da melhor configuração da rede neural de acordo com a otimização, utilizando o agrupamento de janelas de 10 a 15.

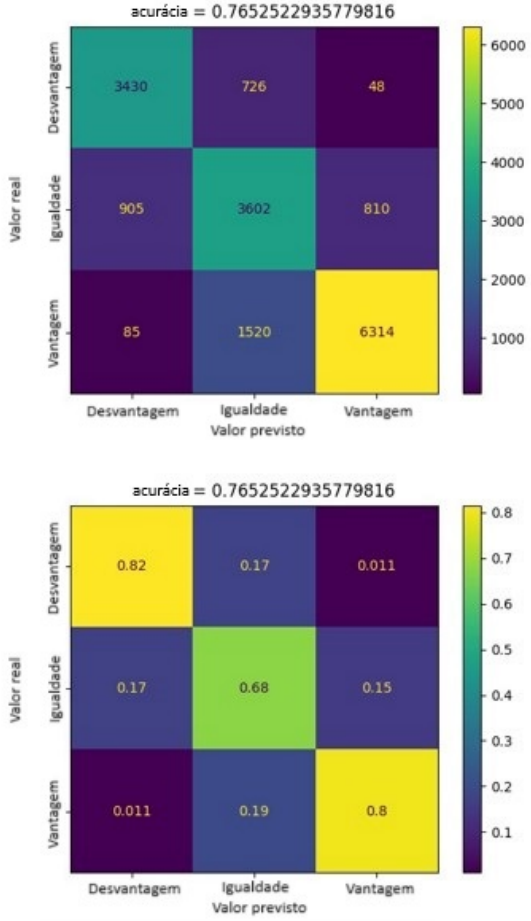

Figura 4.10: Matriz de confusão para a previsão do grupo de teste obtido a partir da melhor configuração da rede neural de acordo com a otimização, utilizando o agrupamento de janelas de 10 a 15.

Para a kNN, como já esperado após os resultados anteriores, de forma geral, os resultados também foram levemente acima dos obtidos pela rede neural. Como é exibido nas Figuras 4.11, 4.6, 4.12, 4.8, 4.13, 4.10, houve uma melhora significativa na utilização de janelas de maior quantidade de posse de bola, chegando a um auge de quase 77% de acurácia.

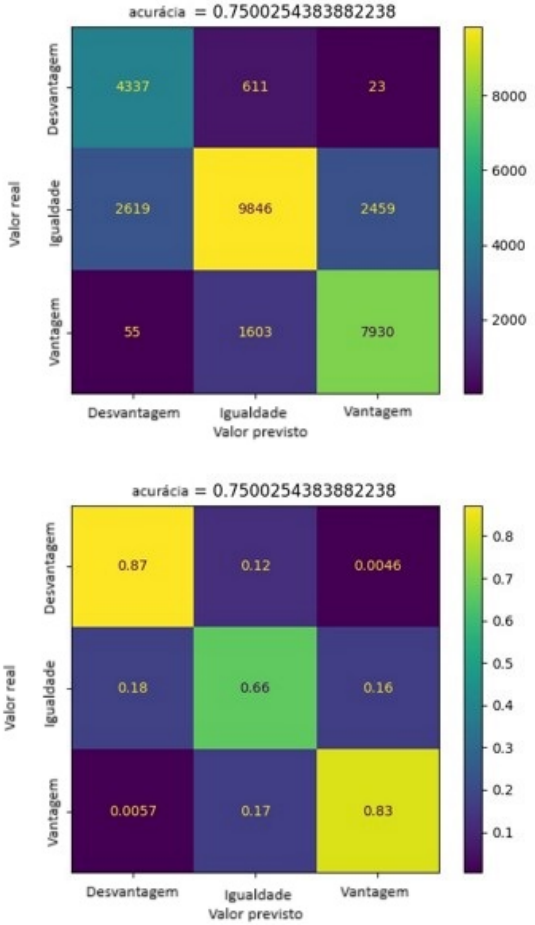

Figura 4.11: Matriz de confusão para a previsão do grupo de validação obtido a partir da melhor configuração da kNN de acordo com a otimização, utilizando o agrupamento de janelas de 8 a 15.

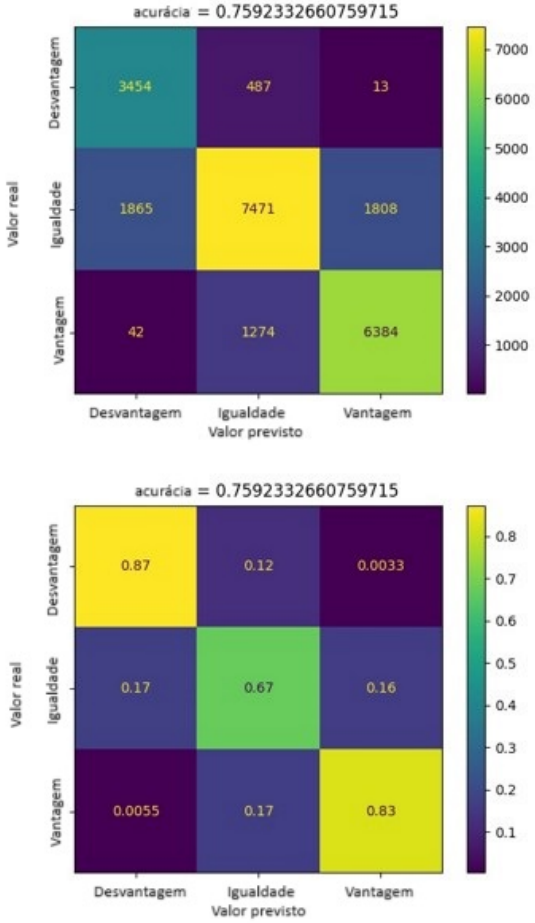

Figura 4.12: Matriz de confusão para a previsão do grupo de validação obtido a partir da melhor configuração da kNN de acordo com a otimização, utilizando o agrupamento de janelas de 9 a 15.

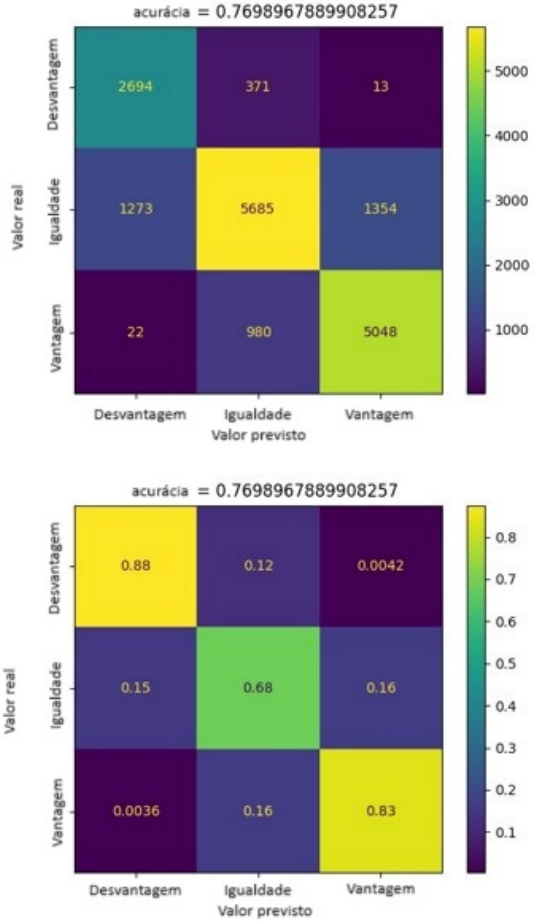

Figura 4.13: Matriz de confusão para a previsão do grupo de validação obtido a partir da melhor configuração da kNN de acordo com a otimização, utilizando o agrupamento de janelas de 10 a 15.

4.2 Resultados do 2° estágio do sistema

Na sequência do trabalho, o foco foi a implementação do segundo estágio do sistema. O procedimento seguido foi similar ao do primeiro estágio, contudo, um componente importante diferenciava este estágio: a utilização da métrica *Mean Absolute Percentage Error* (MAPE) para avaliação dos resultados.

Partindo da lógica do primeiro estágio, foi utilizada a mesma estrutura de otimização para as camadas ocultas da rede neural, com exceção do uso do MAPE como parâmetro de avaliação. Na Tabela 4.2 são apresentados os resultados da otimização considerando desde os segmentos com 10 posses de bola até os de 15 posses, e ainda o "Agrupamento". Segmentos com menos posses de bola são desconsiderados por apresentarem consistentemente resultados muito abaixo do viável. Com base nisso, observou-se que os menores erros estavam associados a uma expansão da janela de análise, indicando que

uma visão mais abrangente do desempenho de uma equipe possibilita uma previsão mais acurada da margem de pontos. O "Agrupamento", englobando todas as janelas de análise, continuou a mostrar o erro mínimo, sugerindo que um conjunto maior de dados contribui significativamente para o rendimento do modelo.

Tabela 4.2: Relação de MAPE médio e melhor MAPE obtido com a otimização da rede neural para cada dado de entrada desde a sequência com 10 posses de bola até a sequência de 15 posses e o agrupamento,

Melhores camadas	Tamanho da sequência (posses)	Menor MAPE médio	Menor MAPE
[65]	10	0,288	0,267
[50, 40]	11	0,310	0,289
[50, 40]	12	0,288	0,259
[65]	13	0,286	0,254
[45, 30]	14	0,272	0,249
[60, 40]	15	0,292	0,266
[60, 40]	Agrupamento	0,241	0,235

Após a otimização, os resultados obtidos através da aplicação da rede neural considerando a métrica MAPE podem ser observados na Tabela 4.2. Tais resultados levam em consideração o uso de camadas ocultas na configuração [60, 40] para padronizar de acordo com o melhor resultado obtido na otimização, taxa de aprendizado de 0.001, e paciência de 500.

Tabela 4.3: Relação do MAPE obtidos com a otimização da rede neural para cada dado de entrada desde a janela de 5 até a janela de 15 e o agrupamento das janelas de 8 a 15.

Tamanho da sequência (posses)	MAPE_Train (%)	MAPE_Val (%)	MAPE_Test (%)
8	25,990	24,268	25,804
9	29,025	27,114	28,633
10	20,572	25,481	26,701
11	22,540	28,268	30,443
12	14,221	23,583	28,125
13	17,459	25,315	46,335
14	18,183	24,503	43,082
15	17,130	26,942	40,371
Agrupamento	21,735	23,391	22,912

Esta tabela fornece visibilidade sobre como a quantidade de posses de bola na janela de análise afetam o cálculo do MAPE na previsão do modelo. A informação adquirida deste processo é essencial para a otimização contínua do sistema de previsão.

A visualização oferecida pelos diagramas de dispersão para cada combinação de janelas observada na Tabela 4.2 pode ser observada nas Figuras

4.14, 4.15, 4.16, 4.17, 4.18, 4.19, 4.20, 4.21 e 4.22. Destaca-se que nos gráficos a seguir, são apresentados os resultados de validação e os resultados de teste, respectivamente.

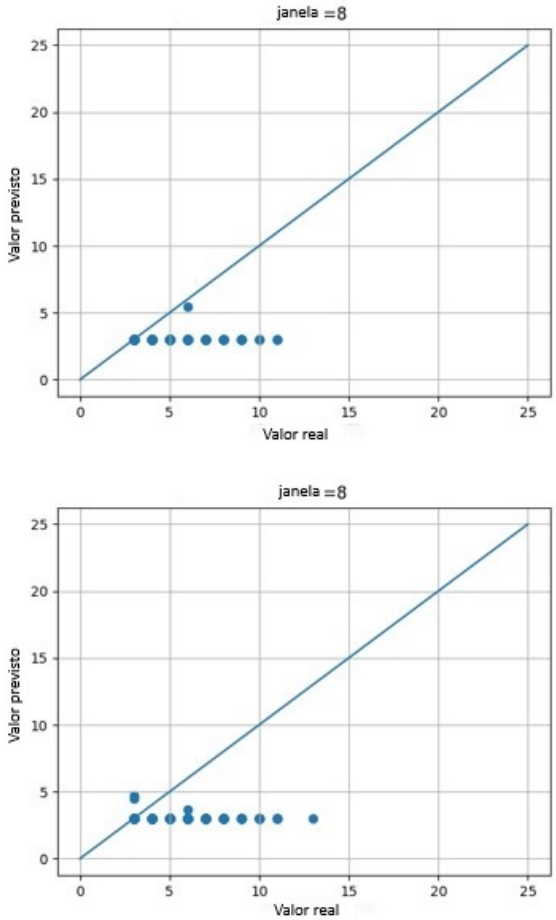

Figura 4.14: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 8 posses de bola.

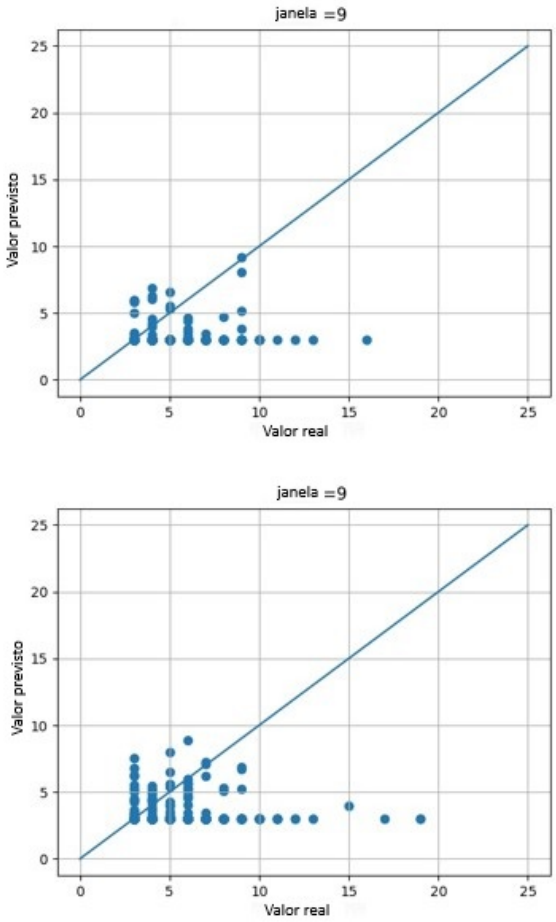

Figura 4.15: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 9 posses de bola.

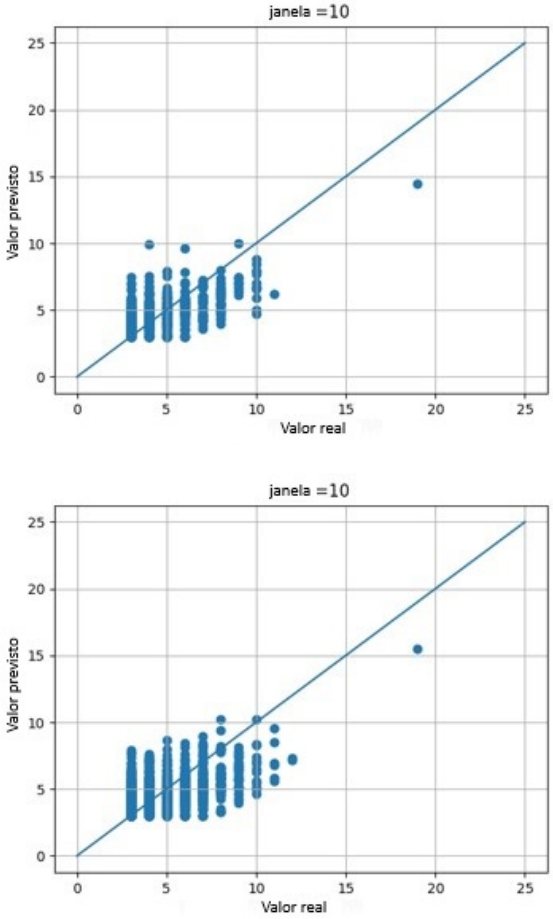

Figura 4.16: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 10 posses de bola.

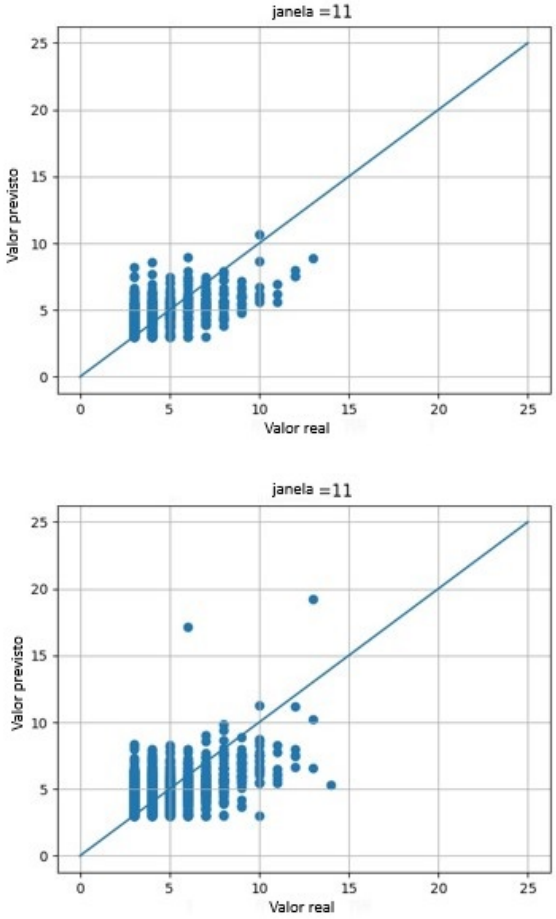

Figura 4.17: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 11 posses de bola.

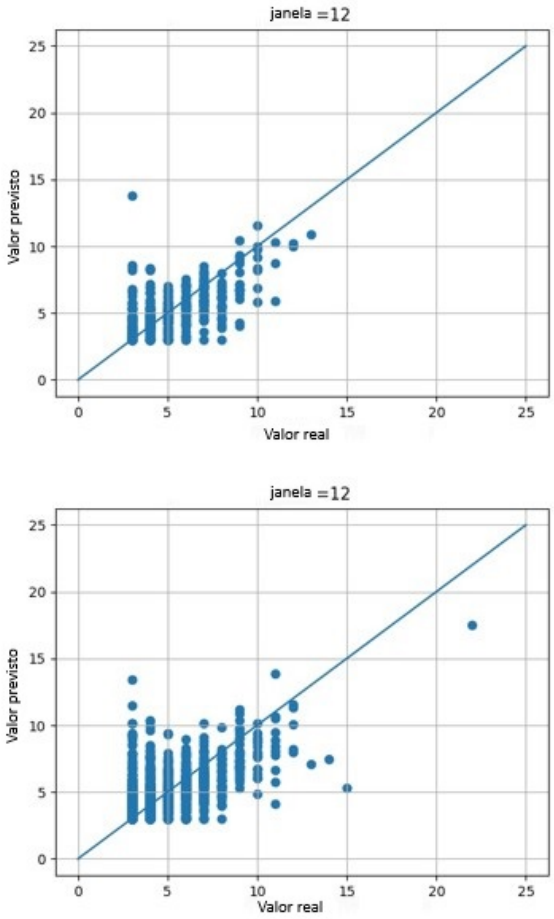

Figura 4.18: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 12 posses de bola.

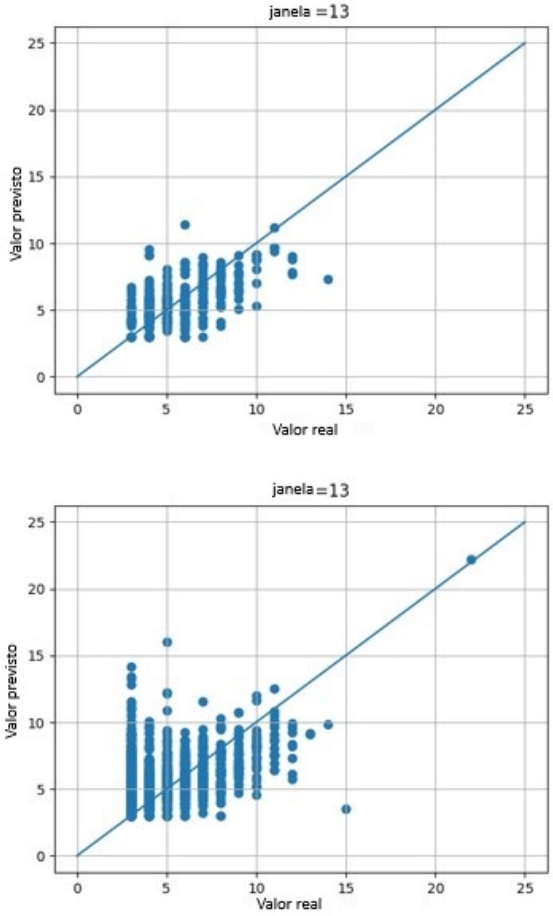

Figura 4.19: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 13 posses de bola.

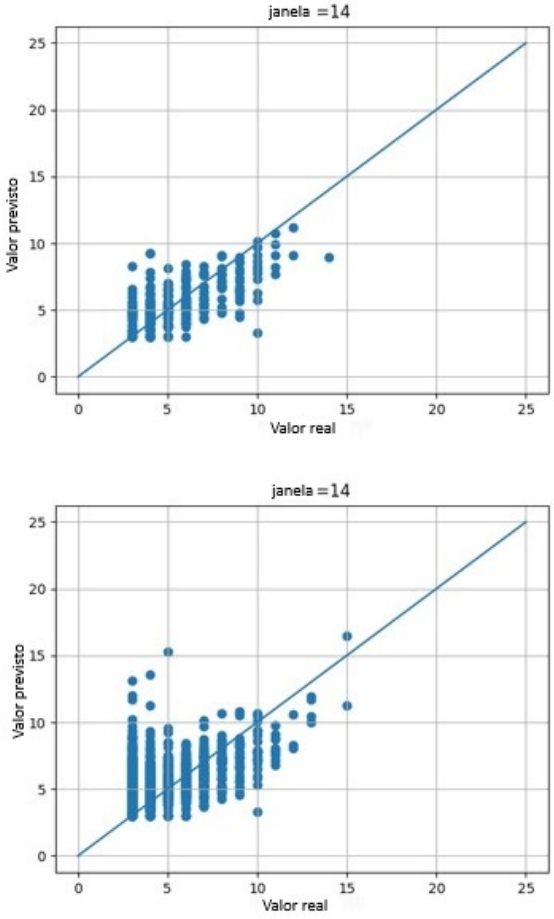

Figura 4.20: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 14 posses de bola.

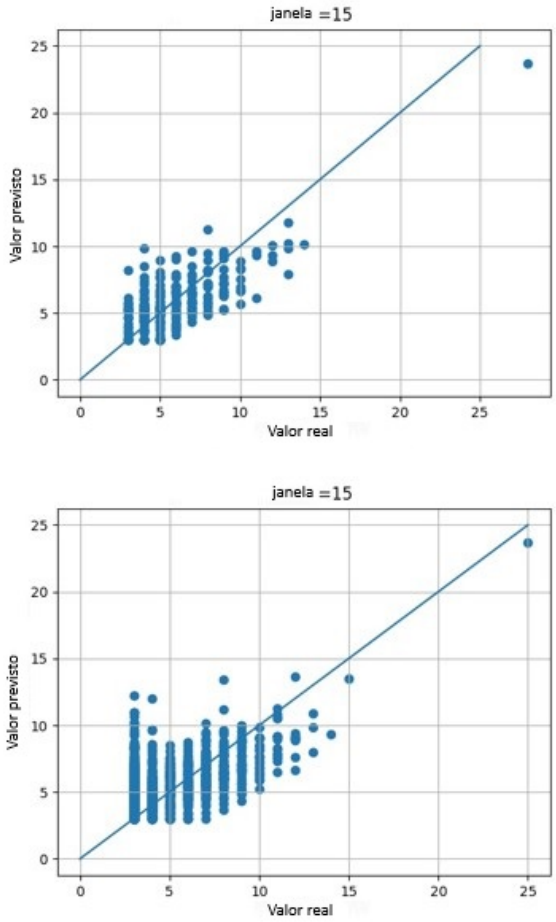

Figura 4.21: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando sequências de 15 posses de bola.

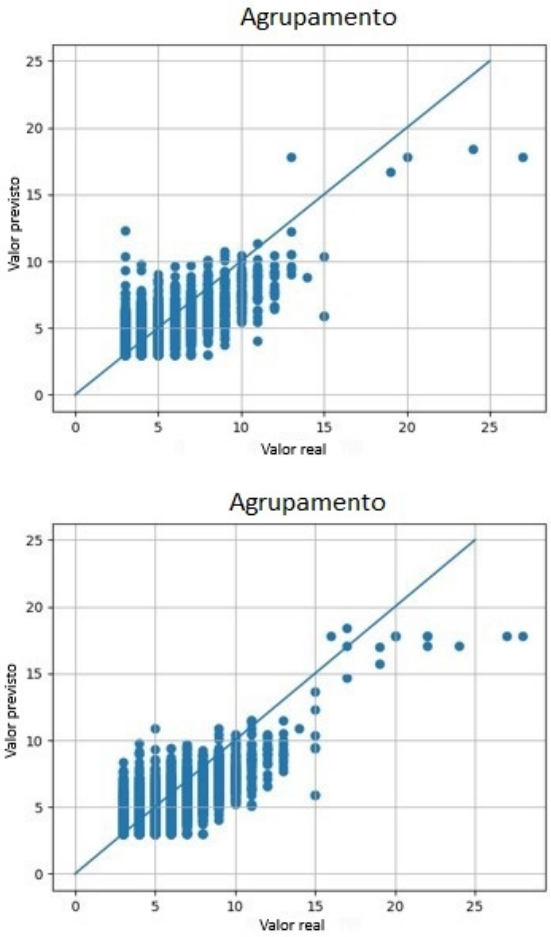

Figura 4.22: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da rede neural de acordo com a otimização, utilizando o agrupamento de janelas de 8 a 15 posses de bola.

No final, a melhor configuração para este segundo estágio do sistema foi obtida com a combinação de janelas de 8 a 15, resultando em um MAPE de 23,39%. A melhor configuração também é escolhida com base nos diagramas de dispersão. A Figura 4.22 apresenta o modelo que conseguiu prever de forma mais adequada tanto valores pequenos quanto valores maiores.

O kNN, por sua vez, apresentou um desempenho levemente inferior. Ao aplicar o algoritmo de *gridsearch*, buscou-se otimizar os hiperparâmetros "*número de vizinhos a serem considerados*" (*n_neighbors*) e "*tipo de peso*" (*weights*). O primeiro é otimizado entre 1 e 10, e informa quantos vizinhos próximos ao valor testado devem ser considerados. O segundo é otimizado entre "*uniforme*" e "*distância*", e decide se devem ser aplicados pesos maiores aos vizinhos mais próximos ou se isso deve ser feito de forma uniforme. O valor de MAPE obtido para o mesmo grupo de teste na dada configuração foi de 26,59%, utilizando os 9 vizinhos mais próximos e pesos ponderados pela distância. Os resultados de "MAPE" em função das janelas são apresentados

na Tabela 4.4 e os respectivos diagramas de dispersão são apresentados nas Figuras 4.23, 4.24, 4.25, 4.26, 4.27, 4.28, 4.29, 4.30 e 4.31.

Tabela 4.4: Relação de "MAPE" obtidos com a otimização da kNN para cada dado de entrada desde a janela de 8 até a janela de 15 e o agrupamento das janelas de 8 a 15.

Tamanho da sequência (posses)	MAPE_Test (%)
8	26,59
9	26,37
10	26,01
11	55,84
12	37,74
13	54,03
14	51,1
15	35,51
Agrupamento	28,67

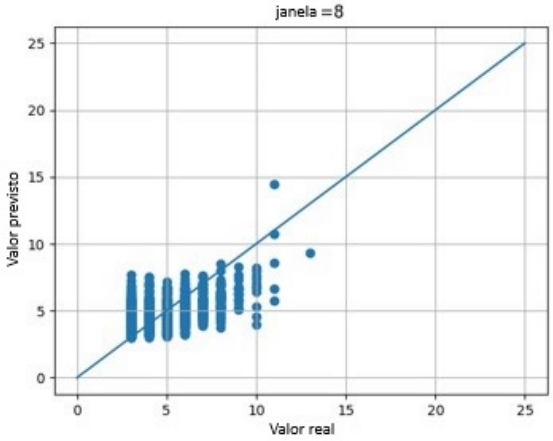

Figura 4.23: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 8 posses de bola.

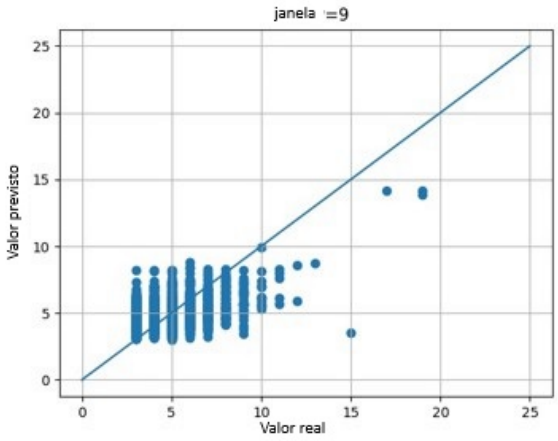

Figura 4.24: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 9 posses de bola.

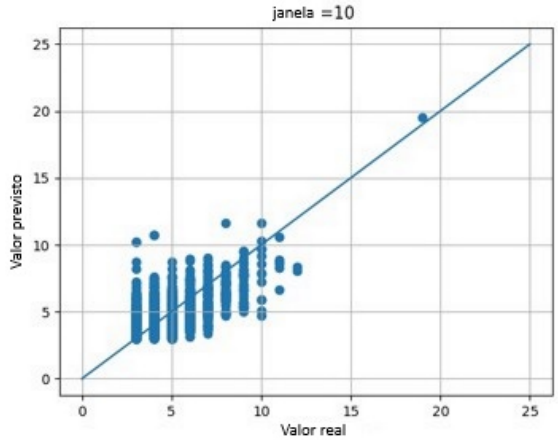

Figura 4.25: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 10 posses de bola.

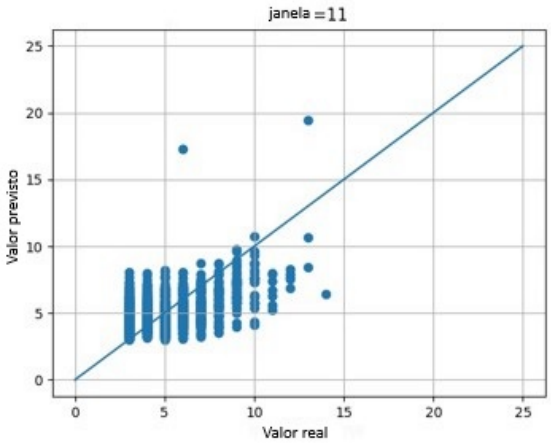

Figura 4.26: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 11 posses de bola.

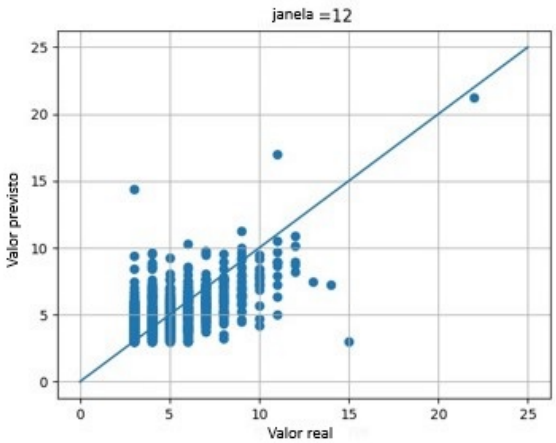

Figura 4.27: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 12 posses de bola.

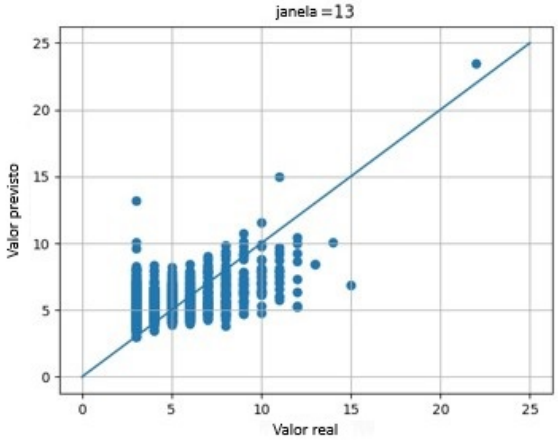

Figura 4.28: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 13 posses de bola.

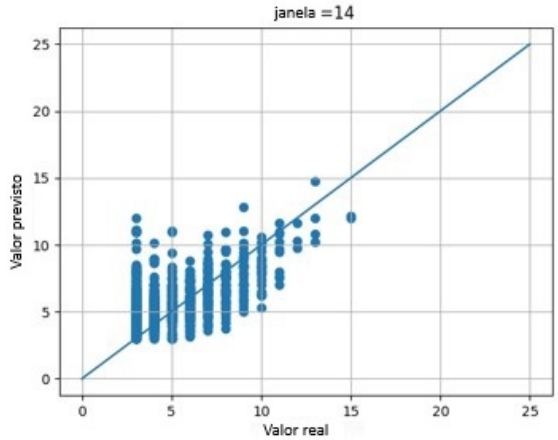

Figura 4.29: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 14 posses de bola.

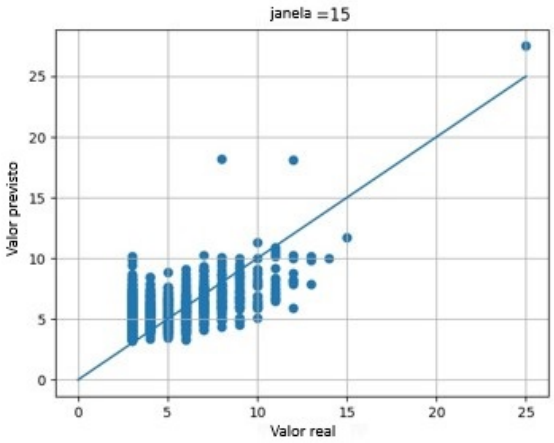

Figura 4.30: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando sequências de 15 posses de bola.

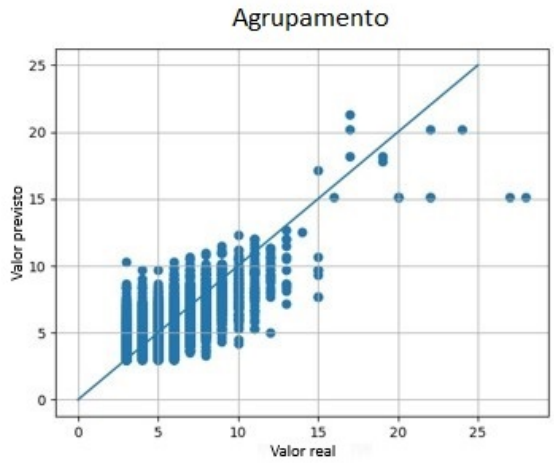

Figura 4.31: Diagramas de dispersão para a previsão dos grupos de validação e teste obtidos a partir da melhor configuração da kNN de acordo com a otimização, utilizando o agrupamento de sequências de posses de bola.

4.3 Resultados da aplicação do sistema em conjunto

Após a otimização dos dois estágios do sistema e a consequente capacidade de prever com alguma confiança não apenas a equipe superior, mas também a magnitude de sua vantagem, buscou-se uma abordagem combinada. Essa aplicação conjunta permite analisar um dado momento do jogo e elaborar um ranking das melhores combinações de quintetos para o time da casa jogar contra o quinteto atual do time visitante.

No ranking proposto, ao considerar uma posse de bola existente na base de dados, são consideradas todas as combinações em que o primeiro

estágio previu vantagem para a equipe da casa, dada a configuração do time oponente naquele instante. A ordenação dessas combinações é feita com base na magnitude da vitória, conforme previsto pelo segundo estágio. Este conjunto de informações pode ser interpretado como uma lista de sugestões de times possivelmente vitoriosos na situação de jogo atual, ranqueados pela confiabilidade da vitória projetada.

Dessa forma, para uma situação de jogo específica entre o Rio Claro e o Corinthians, cujas estatísticas principais podem ser observadas na Tabela 4.5, é apresentado na Tabela 4.6 o ranking discutido anteriormente.

Deve-se ressaltar que a indicação numérica, associada a cada um dos cinco jogadores, indica o grupo ao qual o jogador pertence (lembrando que os jogadores foram agrupados considerando as variáveis descritas na Tabela 3.1).

Tabela 4.5: Dados iniciais da posse de bola número 1 do jogo 23265, de Rio Claro contra Corinthians,

Variável	Valor	Variável	Valor
season_year	2019	home_quintet_turnovers	0
home_team_name	Rio Claro	home_quintet_eff_per_min	0,0
away_team_name	Corinthians	home_quintet_minutes	0,0
possession_id	1	away_cluster_1	1,0
period	1	away_cluster_2	1,0
match_elapsed_minutes	0,0	away_cluster_3	0,0
home_score	0	away_cluster_4	0,0
away_score	0	away_cluster_5	0,0
home_cluster_1	1,0	away_cluster_6	1,0
home_cluster_2	1,0	away_cluster_7	1,0
home_cluster_3	0,0	away_cluster_8	1,0
home_cluster_4	1,0	away_cluster_9	0,0
home_cluster_5	0,0	away_cluster_10	0,0
home_cluster_6	0,0	away_cluster_11	0,0
home_cluster_7	1,0	away_cluster_12	0,0
home_cluster_8	0,0	away_cluster_13	0,0
home_cluster_9	0,0	away_cluster_14	0,0
home_cluster_10	0,0	away_cluster_15	0,0
home_cluster_11	0,0	away_cluster_16	0,0
home_cluster_12	0,0	away_quintet_points	0
home_cluster_13	0,0	away_quintet_assists	0
home_cluster_14	1,0	away_quintet_rebounds	0
home_cluster_15	0,0	away_quintet_steals	0
home_cluster_16	0,0	away_quintet_blocks	0
home_quintet_points	0	away_quintet_missed_throws	0
home_quintet_assists	0	away_quintet_missed_free_throws	0
home_quintet_rebounds	0	away_quintet_fouls	0
home_quintet_steals	0	away_quintet_turnovers	0
home_quintet_blocks	0	away_quintet_eff_per_min	0,0
home_quintet_missed_throws	0	away_quintet_minutes	0,0
home_quintet_missed_free_throws	0	home_has_possession	True
home_quintet_fouls	0		

Tabela 4.6: Seleção de jogadores pelo estágio 1 ranqueadas pelo resultado normalizado do estágio 2 indicando as melhores opções de jogadores para a situação mostrada na Tabela 4.5 em termos de clusters,

jogador_1	jogador_2	jogador_3	jogador_4	jogador_5	Saldo de pontos
4,0	4,0	11,0	15,0	15,0	0,10871218
7,0	7,0	7,0	7,0	15,0	0,09978989
7,0	7,0	7,0	13,0	15,0	0,099072106
5,0	7,0	7,0	7,0	7,0	0,09839028
3,0	7,0	7,0	7,0	7,0	0,0980178
7,0	7,0	7,0	9,0	15,0	0,09563801
4,0	8,0	12,0	15,0	15,0	0,095072016
7,0	7,0	7,0	11,0	15,0	0,09382296
5,0	7,0	7,0	7,0	13,0	0,09348687
4,0	10,0	11,0	15,0	15,0	0,09318312
...	...	...	...	...	...

Esta ferramenta representa um recurso poderoso para os técnicos e estrategistas, que podem usá-la para informar suas decisões e possivelmente maximizar as chances de vitória de seu time.

5
Conclusões e trabalhos futuros

Neste trabalho, aborda-se o desenvolvimento e a implementação de um sistema inovador de suporte à decisão no âmbito do basquete profissional. Técnicas avançadas de aprendizado de máquina, incluindo Redes Neurais e algoritmos k-Nearest Neighbors (kNN), foram empregadas para prever os resultados das sequências de posse de bola durante partidas de basquete.

O sistema de previsão opera em dois estágios: o estágio categórico, que prevê o time superior, e o estágio numérico, que prevê a magnitude da vantagem. Ao implementar esses dois estágios em conjunto, foi possível ranquear as melhores combinações de quintetos para um dado momento do jogo. Isso possibilita a recomendação de equipes possivelmente superiores a seus adversários na situação atual de jogo, classificadas de acordo com a probabilidade de sucesso.

Análises meticulosas da estrutura da rede neural e da janela de análise de posses de bola resultaram em uma otimização substancial da acurácia do modelo preditivo. Verificou-se que uma análise mais extensa do desempenho da equipe, considerando dados com maior número de posses de bola observadas durante o enfrentamento dos quintetos, proporcionou previsões mais acuradas do saldo de pontos.

Adicionalmente à abordagem de segmentos com mais posses de bola citada anteriormente, a otimização do algoritmo kNN, utilizando o método *Grid Search Cross-Validation*, contribuiu para a maximização da acurácia do modelo. O método permitiu identificar a melhor configuração de parâmetros para o modelo, ajustando-o de acordo com as características e tendências do conjunto de dados.

As matrizes de confusão, os diagramas de dispersão e a tabela de ranking fornecem uma visualização clara e intuitiva dos resultados das previsões, facilitando sua interpretação e uso na tomada de decisões.

A implementação bem-sucedida do sistema reforça a importância e o potencial da inteligência artificial no mundo dos esportes profissionais. A capacidade de prever resultados de jogo com uma acurácia relativamente alta em tempo real tem implicações significativas para o planejamento de estratégias e a dinâmica do jogo.

Contudo, reconhece-se que ainda há espaço para aprimoramento. Melhorias futuras do sistema poderiam incluir o ajuste de parâmetros adicionais, a incorporação de outros tipos de dados e a experimentação com outros mode-

los de aprendizado de máquina. Espera-se realizar um modelo de previsão de escolhas do técnico adversário para compor a previsão atual e permitir uma simulação de passos futuros do jogo, intensificando o valor da informação que o algoritmo fornece ao técnico. Ainda como forma de aprimorar os modelos desenvolvidos neste trabalho, se estuda a possibilidade de modificar a abordagem de agrupamentos realizada sobre o perfil dos jogadores, visando uma abordagem mais direta, representando cada jogador por um conjunto de características físicas, comportamentais e técnicas. Em complemento se estuda a possibilidade de buscar como resultado para o segundo estágio a média de pontos por posse de bola esperada para a sequência, ao invés de seu valor absoluto, com o objetivo de melhorar a aplicação real e excluir a necessidade de entrada da variável "tamanho da sequência em termos de posses de bola".

Apesar das limitações, este trabalho fornece uma base sólida para o desenvolvimento contínuo de sistemas preditivos na área do esporte. O conhecimento adquirido e as técnicas empregadas neste projeto poderão ser aplicados em outras ligas de basquete e, possivelmente, em outros esportes, abrindo novas oportunidades para a aplicação da inteligência artificial no domínio desportivo.

6
Referências bibliográficas

ADNAN, M. et al. Utilizing grid search cross-validation with adaptive boosting for augmenting performance of machine learning models. **PeerJ Computer Science**, PeerJ Inc., v. 8, p. e803, 2022.

DIAS, C. A. **Descoberta de conhecimento em banco de dados para apoio a tomada de decisão**. [S.l.], 2007.

DINIZ, E. M. F. **Diamond Hoop: uma abordagem para otimização na composição de equipes no basquetebol com o uso de inteligência artificial sobre estatísticas pregressas**. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2023.

GHOSH, M. et al. A wrapper-filter feature selection technique based on ant colony optimization. **Neural Computing and Applications**, Springer, v. 32, p. 7839–7857, 2020.

GUO, G. et al. Knn model-based approach in classification. In: SPRINGER. **On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings**. [S.l.], 2003. p. 986–996.

HAN, J.; PEI, J.; TONG, H. **Data mining: concepts and techniques**. [S.l.]: Morgan kaufmann, 2022.

HEATON, J. **Introduction to neural networks with Java**. [S.l.]: Heaton Research, Inc., 2008.

KAPADIA, K. et al. Sport analytics for cricket game results using machine learning: An experimental study. **Applied Computing and Informatics**, Emerald Publishing Limited, v. 18, n. 3/4, p. 256–266, 2020.

LAWRENCE, J. **Introduction to neural networks**. [S.l.]: California Scientific Software, 1993.

LI, B.; XU, X. Application of artificial intelligence in basketball sport. **Journal of Education, Health and Sport**, v. 11, n. 7, p. 54–67, 2021.

LI, H.; ZHANG, M. Artificial intelligence and neural network-based shooting accuracy prediction analysis in basketball. **Mobile Information Systems**, Hindawi Limited, v. 2021, p. 1–11, 2021.

LIU, Z. Application of artificial intelligence technology in basketball games. In: IOP PUBLISHING. **IOP Conference Series: Materials Science and Engineering**. [S.l.], 2020. v. 750, n. 1, p. 012093.

LOEFFELHOLZ, B.; BEDNAR, E.; BAUER, K. W. Predicting nba games using neural networks. **Journal of Quantitative Analysis in Sports**, De Gruyter, v. 5, n. 1, 2009.

MARKOV, Z.; RUSSELL, I. An introduction to the weka data mining system. **ACM SIGCSE Bulletin**, ACM New York, NY, USA, v. 38, n. 3, p. 367–368, 2006.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos sobre aprendizado de máquina. **Sistemas inteligentes-Fundamentos e aplicações**, v. 1, n. 1, p. 32, 2003.

MYTTENAERE, A. D. et al. Mean absolute percentage error for regression models. **Neurocomputing**, Elsevier, v. 192, p. 38–48, 2016.

NEVES, B. O. Mineração de dados aplicada à predição de resultados de jogos de basquete. 2022.

PRETORIUS, A.; PARRY, D. A. Human decision making and artificial intelligence: a comparison in the domain of sports prediction. In: **Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists**. [S.l.: s.n.], 2016. p. 1–10.

REZENDE, S. O. et al. Mineração de dados. **Sistemas inteligentes: fundamentos e aplicações**, v. 1, p. 307–335, 2003.

ROBNIK-ŠIKONJA, M.; KONONENKO, I. Theoretical and empirical analysis of relieff and rrelieff. **Machine learning**, Springer, v. 53, p. 23–69, 2003.

SINGH, N. Sport analytics: a review. **learning**, v. 9, p. 11, 2020.

SONG, Y. et al. An efficient instance selection algorithm for k nearest neighbor regression. **Neurocomputing**, Elsevier, v. 251, p. 26–34, 2017.

WOIDA, D. J. **Classificação dos atletas da NCAA para o draft da NBA por meio de técnicas de mineração de dados**. Dissertação (B.S. thesis) — Universidade Tecnológica Federal do Paraná, 2021.

YANG, Z. Research on basketball players' training strategy based on artificial intelligence technology. In: IOP PUBLISHING. **Journal of Physics: Conference Series**. [S.l.], 2020. v. 1648, n. 4, p. 042057.