UNIVERSIDADE FEDERAL DE ALAGOAS
Instituto de Computação

José Rubens da Silva Brito

Uma Abordagem de Aprendizado de Máquina para Identificação de Tendências de Sucesso de Jogadores de Basquete Universitário Americano para a Liga Profissional: Conciliando Predição e Explicabilidade

Maceió - AL
2024

José Rubens da Silva Brito

Uma Abordagem de Aprendizado de Máquina para Identificação de Tendências de Sucesso de Jogadores de Basquete Universitário Americano para a Liga Profissional: Conciliando Predição e Explicabilidade

Dissertação apresentada ao Programa de Pós-graduação em Informática do Instituto de Computação da Universidade Federal de Alagoas como requisito parcial para obtenção do título de Mestre em Informática.

Orientador: Prof. Dr. Evandro de Barros Costa
Coorientador: Profa. Dra. Roberta Vilhena Vieira Lopes

Maceió - AL
2024

Catalogação na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico
Bibliotecária: Helena Cristina Pimentel do Vale – CRB4 - 661

B862u Brito, José Rubens da Silva.
Uma abordagem de aprendizado de máquina para identificação de tendências de sucesso de jogadores de basquete universitário americano para a liga profissional : conciliando predição e explicabilidade / José Rubens da Silva Brito. – 2024.
119 f.: il.

Orientador: Evandro de Barros Costa.
Coorientador: Roberta Vilhena Vieira Lopes.
Dissertação (mestrado em Informática) – Universidade Federal de Alagoas, Instituto de Computação. Maceió, 2024.

Bibliografia: f. 74-78.
Apêndices: 79-119.

1. Aprendizado de máquina. 2. Esporte universitário americano. 3. Jogadores de Basquetebol. 4. Performance. I. Título.

CDU: 004:796.32

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE ALAGOAS
INSTITUTO DE COMPUTAÇÃO
Av. Lourival Melo Mota, S/N, Tabuleiro do Martins, Maceió - AL, 57.072-970
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO (PROPEP)
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

Folha de Aprovação

JOSÉ RUBENS DA SILVA BRITO

UMA ABORDAGEM DE APRENDIZADO DE MÁQUINA PARA IDENTIFICAÇÃO DE TENDÊNCIAS DE SUCESSO DE JOGADORES DE BASQUETE UNIVERSITÁRIO AMERICANO PARA A LIGA PROFISSIONAL: CONCILIANDO PREDIÇÃO E EXPLICABILIDADE

A MACHINE LEARNING APPROACH TO IDENTIFY SUCCESS TRENDS IN NCAA PLAYERS WITH GOOD CHANCE TO REACH NBA: A BALANCE BETWEEN PREDICTION PERFORMANCE AND EXPLAINABILITY

Dissertação submetida ao corpo docente do Programa de Pós-Graduação em Informática da Universidade Federal de Alagoas e aprovada em 29 de outubro de 2024.

Banca Examinadora:

gov.br Documento assinado digitalmente
EVANDRO DE BARROS COSTA
Data: 19/11/2024 14:09:53-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. EVANDRO DE BARROS COSTA
UFAL – Instituto de Computação
Orientador

gov.br Documento assinado digitalmente
BRUNO ALMEIDA PIMENTEL
Data: 27/11/2024 10:21:13-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. BRUNO ALMEIDA PIMENTEL
UFAL – Instituto de Computação
Examinador Interno

gov.br Documento assinado digitalmente
THALES MIRANDA DE ALMEIDA VIEIRA
Data: 27/11/2024 10:11:53-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. THALES MIRANDA DE ALMEIDA VIEIRA
UFAL – Instituto de Computação
Examinador Interno

gov.br Documento assinado digitalmente
JOSEANA MACEDO FECHINE REGIS DE ARAUJO
Data: 27/11/2024 09:56:15-0300
Verifique em https://validar.iti.gov.br

Profa. Dra. JOSEANA MACEDO FECHINE
UFCG – Universidade Federal de Campina Grande
Examinadora Externa

gov.br Documento assinado digitalmente
JOSE ANTAO BELTRAO MOURA
Data: 26/11/2024 11:40:47-0300
Verifique em https://validar.iti.gov.br

Prof. Dr. JOSÉ ANTÃO BELTRÃO MOURA
UFCG – Universidade Federal de Campina Grande
Examinador Externo

Dedico este trabalho a meus pais, Tereza Cristina e Rubens Amaro, e à minha esposa, Alice Maria, por estarem sempre ao meu lado e pelo apoio incondicional.

Se pude enxergar mais longe, foi porque me apoiei em ombros de gigantes

(Isaac Newton)

Agradecimentos

Primeiramente, agradeço a Deus pela vida e pela saúde que me permitiram concluir esta importante etapa. Aos meus pais, Tereza Cristina e Rubens Amaro, por serem meu pilar inabalável, sempre me incentivando nos momentos difíceis e compreendendo minha ausência durante a dedicação a este trabalho. Ao meu irmão, José Rafael, por sempre torcer pelas minhas vitórias.

Agradeço à minha esposa, Alice Maria, que esteve ao meu lado nas horas mais desafiadoras, demonstrando compreensão, paciência e afeto.

Sou grato à UFAL e ao corpo docente do Programa de Pós-Graduação, que me proporcionaram oportunidades únicas de crescimento profissional. Em especial, agradeço ao meu orientador, Professor Evandro de Barros Costa, e à Professora Roberta Vilhena Vieira Lopes, pelo tempo dedicado e pelo apoio que me fizeram acreditar na minha capacidade.

Agradeço ao corpo técnico do Programa de Pós-Graduação em Informática pelo suporte nas demandas burocráticas.

Aos meus amigos e colegas de curso, com quem compartilhei experiências intensas nos últimos anos, meu agradecimento pelo companheirismo que me fez crescer tanto pessoal quanto academicamente. Um agradecimento especial a Matheus Gomes, Ramon Basto, João Victo, Wagner Silva e Lucas Lisboa.

Agradeço também aos professores que aceitaram compor a banca examinadora.

Por fim, agradeço a todos que cruzaram meu caminho durante esses anos, cujo incentivo impactou minha formação. E, claro, agradeço a você, leitor, por fazer parte dessa jornada.

Resumo

Este estudo investiga a aplicação de técnicas de aprendizado de máquina em dados históricos do basquete universitário americano (NCAA), com o objetivo de prever quais jogadores têm maior potencial de chegar à NBA. Nos últimos anos, a análise de dados no basquete ganhou destaque, indo além das estatísticas convencionais e incorporando dados de sensores e câmeras, a fim de identificar padrões que possam aprimorar o desempenho de jogadores e equipes. Contudo, embora esse crescente complexidade aumente o valor preditivo, ela também gera um conjunto de dados que pode conter características redundantes, ruidosas e irrelevantes, as quais podem impactar negativamente a atividade preditiva. Para mitigar esse problema e buscar uma solução que concilie predição e explicabilidade dos modelos preditivos, propôs-se, neste trabalho, uma abordagem composta por três etapas principais: **(I)** seleção dos atributos mais relevantes para auxiliar na tomada de decisão; **(II)** utilização de algoritmos de aprendizado de máquina; **(III)** análise dos resultados preditivos por meio da explicabilidade de cada modelo. Na etapa **(I)**, foi feita a seleção de dados que, quando combinados, formam um conjunto de atributos do jogador, influenciando direta ou indiretamente sua contratação por equipes da NBA, considerando a configuração atual do time. Utilizaram-se técnicas consolidadas na literatura, como Wrapper, Filter, Embedding, além do Algoritmo Genético, com o objetivo de melhorar a precisão preditiva e reduzir o número de características. Na etapa **(II)**, buscou-se equilibrar interpretabilidade e precisão preditiva, empregando métodos de classificação transparente, como Árvores de Decisão, Regressão Logística e o Algoritmo de Regras (PRISM). Como referência de modelo de opaco, utilizou-se a Máquina de Vetores de Suporte (SVM). Já na etapa **(III)**, analisou-se a explicabilidade de cada modelo. Isso foi feito de duas maneiras: pela própria construção dos algoritmos, como os de indução via Árvores de Decisão e via regras, e por meio da ferramenta de explicabilidade SHAP. Para a validação da abordagem, utilizou-se a base de dados mencionada, e os resultados indicaram um impacto positivo da seleção de atributos nos modelos preditivos, com destaque para a influência benéfica do Algoritmo Genético na etapa de seleção. Essa abordagem contribuiu para a identificação de um conjunto mínimo de atributos e para a melhoria das métricas de predição dos classificadores. Especificamente, a combinação do Algoritmo Genético com SVM na função de aptidão, na etapa de seleção gerou um conjunto de atributos que, ao ser utilizado na Árvore de Decisão (CART), alcançou uma acurácia de 80%. Por fim, foi realizada uma análise da interpretabilidade dos modelos Árvore de Decisão CART e PRISM, destacando a clareza fornecida por cada um: o primeiro baseado em estrutura de Árvore de Decisão e o segundo em algoritmo de regras. Adicionalmente, utilizamos a ferramenta SHAP para analisar as saídas geradas pelos algoritmos de aprendizado de máquina, permitindo uma interpretação mais clara dos resultados, para assim, auxiliar na tomada de decisão. Espere-se com esses resultados poder contribuir para a aplicação eficiente de técnicas de aprendizado de máquina no basquete, sobretudo na predição de comportamentos futuros de jogadores com base em variáveis explicáveis e selecionadas de forma criteriosa.

Abstract

This study investigates the application of machine learning techniques on historical data from American college basketball (NCAA), with the aim of predicting which players have the highest potential to make it to the NBA. In recent years, data analysis in basketball has gained prominence, going beyond conventional statistics and incorporating data from sensors and cameras to identify patterns that can enhance player and team performance. However, although this growing complexity increases the predictive value, it also generates a dataset that may contain redundant, noisy, and irrelevant features, which can negatively impact predictive accuracy. To mitigate this problem and seek a solution that balances prediction and explainability of predictive models, this work proposes an approach composed of three main steps: **(I)** selection of the most relevant features to aid decision-making; **(II)** use of machine learning algorithms; **(III)** analysis of predictive results through the explainability of each model. In step **(I)**, data selection was performed, and when combined, these features form a set of player attributes that directly or indirectly influence their hiring by NBA teams, considering the team's current configuration. Established techniques from the literature, such as Wrapper, Filter, Embedding, as well as the Genetic Algorithm, were used to improve predictive accuracy and reduce the number of features. In step **(II)**, the goal was to balance interpretability and predictive accuracy by employing transparent classification methods such as Decision Trees, Logistic Regression, and the Rule-based Algorithm (PRISM). The Support Vector Machine (SVM) was used as a reference for opaque models. In step **(III)**, the explainability of each model was analyzed. This was done in two ways: through constructing the algorithms themselves, such as decision tree induction and rule-based algorithms, and through the SHAP explainability tool. For validation, the mentioned dataset was used, and the results indicated a positive impact of feature selection on the predictive models, with particular emphasis on the beneficial influence of the Genetic Algorithm in the selection phase. This approach contributed to identifying a minimal set of features and improving the prediction metrics of the classifiers. Specifically, combining the Genetic Algorithm with SVM in the fitness function during the selection phase produced a set of features that, when used in the Decision Tree (CART), achieved an accuracy of 80%. Finally, an analysis of the interpretability of the CART Decision Tree and PRISM models was carried out, highlighting the clarity provided by each: the first based on a decision tree structure and the second on a rule-based algorithm. Additionally, the SHAP tool was used to analyze the outputs generated by the machine learning algorithms, allowing for a clearer interpretation of the results, and thus aiding decision-making. Hopefully, these results will contribute to the efficient application of machine learning techniques in basketball, particularly in predicting future player behaviors based on explainable variables selected in a rigorous manner.

Lista de Figuras

1.1 Arquitetura Proposta (Adaptado de (LUNDBERG, S. M.; LEE, S.-I., 2017a) . . . 16

2.1 Abordagem *Filter* (Autor, 2023) . . . 20
2.2 Arquitetura Algoritmo Genético (KATO; PAIVA; IZIDORO, 2021) . . . 22

3.1 Etapas do protocolo PRISMA (Autor, 2023) . . . 29
3.2 Países que mais produzem na área (Autor, 2023) . . . 32
3.3 Mapa com as produções cientificas por pais (Autor, 2023) . . . 32
3.4 Principais temas (Autor, 2023) . . . 34
3.5 Abordagens Utilizadas (Autor, 2023) . . . 35

4.1 Arquitetura Proposta (Adaptado de (LUNDBERG, S. M.; LEE, S.-I., 2017a)) . . . 38
4.2 Cruzamento 1 Ponto de Corte (Adaptado de (KATO; PAIVA; IZIDORO, 2021)) . 48
4.3 Mutação por Inversão (Adaptado de (KATO; PAIVA; IZIDORO, 2021)) . . . 48

5.1 Regra gerada via PRISM . . . 62
5.2 Árvore de Decisão - CART . . . 64
5.3 Gráfico *Beeswarm* gerado com a ferramenta SHAP, utilizando o modelo CART e os atributos selecionados pelo método *Filter* (Autor, 2024). . . . 67
5.4 Gráfico *Beeswarm* gerado com a ferramenta SHAP, utilizando o modelo CART e os atributos selecionados pelo método *AG - SVM* (Autor, 2024). . . . 68
5.5 Distribuição dos Scores de Teste por Subconjunto de Features e Modelo (Autor, 2024) . . . 69
5.6 Teste de Nemenyi (Autor, 2024) . . . 70

B.1 Matriz de Confusão - C5.0 - Features *Embedded* . . . 105
B.2 Matriz de Confusão - C5.0 - Features *Filter* . . . 106
B.3 Matriz de Confusão - C5.0 - Features GA CART . . . 106
B.4 Matriz de Confusão - C5.0 - Features GA Entropy . . . 107
B.5 Matriz de Confusão - C5.0 - Features GA SVM . . . 107
B.6 Matriz de Confusão - C5.0 - Features *Wrapper* . . . 108
B.7 Matriz de Confusão - C5.0 - Features *All Features* . . . 108
B.8 Matriz de Confusão - Cart - Features *Embedded* . . . 109
B.9 Matriz de Confusão - Cart - Features *Filter* . . . 109
B.10 Matriz de Confusão - Cart - Features GA CART . . . 110
B.11 Matriz de Confusão - Cart - Features GA Entropy . . . 110
B.12 Matriz de Confusão - Cart - Features GA SVM . . . 111
B.13 Matriz de Confusão - Cart - Features *Wrapper* . . . 111
B.14 Matriz de Confusão - Cart - Features *All Features* . . . 112
B.15 Matriz de Confusão - RL - Features *Embedded* . . . 112

B.16 Matriz de Confusão - RL - Features *Filter* 113
B.17 Matriz de Confusão - RL - Features GA CART 113
B.18 Matriz de Confusão - RL - Features GA Entropy 114
B.19 Matriz de Confusão - RL - Features GA SVM 114
B.20 Matriz de Confusão - RL - Features *Wrapper* 115
B.21 Matriz de Confusão - RL - Features *All Features* 115
B.22 Matriz de Confusão - SVM - Features *Embedded* 116
B.23 Matriz de Confusão - SVM - Features *Filter* 116
B.24 Matriz de Confusão - SVM - Features GA CART 117
B.25 Matriz de Confusão - SVM - Features GA Entropy 117
B.26 Matriz de Confusão - SVM - Features GA SVM 118
B.27 Matriz de Confusão - SVM - Features *Wrapper* 118
B.28 Matriz de Confusão - SVM - Features *All Features* 119

Lista de Tabelas

3.1 Questões Sobre o Escopo da Pesquisa 27
3.2 *Strings* de Busca 28
3.3 Autores mais relevantes 31
3.4 Afiliações mais relevantes 31
3.5 Fontes mais relevantes 33
3.6 Artigos e Número de Citações 37

4.1 Interpretação dos índices de correlação, (SCHOBER; BOER; SCHWARTE, 2018) 45
4.2 Matriz de Correlação 52
4.3 Fórmulas das Métricas utilizadas 52

5.1 Método de Seleção de Características pelo Total de Atributos 55
5.2 Características selecionadas pelas técnicas 56
5.3 Resultado da predição sem seleção de atributos (Autor, 2024) 58
5.4 Resultado da predição com métodos de seleção de atributos tradicionais, (Autor, 2024) 59
5.5 Resultado da predição com metodós de seleção de atributos tradicionais, (Autor, 2024) 60

Lista de Abreviaturas e Siglas

DBPM	*Defensive Box Plus Minus*
Google Colab	*Google Computer Engine*
IA	Inteligência Artificial
IEEE	*Institute of Electrical and Electronics Engineers*
KDD	*Knowledge Discovery in Databases*
MEM	*Maximum Entropy Markov Model*
MLP	*Multi Layer Perceptron*
NCAA	*National Collegiate Athletic Association*
OGBPM	*Offensive Game Box Plus Minus*
RNN	*Recurrent Neural Network*
SVM	Support Vector Machines
TREB	*Total Rebound Percent*

Sumário

1 Introdução **14**
1.1 Motivação e Contextualização da Pesquisa . . . 14
1.2 Objetivos da Pesquisa . . . 16
1.2.1 Objetivos Gerais . . . 16
1.2.2 Objetivos Específicos . . . 17
1.3 Questões de Pesquisa . . . 17
1.4 Estrutura do Trabalho . . . 18

2 Fundamentação Teórica **19**
2.1 Métodos de Seleção Atributos . . . 19
2.1.1 *Filter* . . . 19
2.1.2 *Wrapper* . . . 20
2.1.3 *Embedded* . . . 21
2.1.4 Algoritmo Genético . . . 21
2.2 Algoritmos de Aprendizagem de Máquina Supervisionados . . . 22
2.2.1 Indução via Árvores de Decisão . . . 22
2.2.2 Indução via Algoritmo de Regras . . . 23
2.2.3 Regressão Logística . . . 24
2.2.4 Máquina de Vetores de Suporte . . . 24
2.3 Explicabilidade em Modelos de Aprendizagem de Máquina . . . 25
2.4 Considerações Finais do Capítulo . . . 25

3 Revisão da Literatura e Bibliométrica **27**
3.1 Metodologia . . . 28
3.2 Análise Bibliométrica dos Artigos . . . 30
3.2.1 Quais são os principais autores e instituições que publicam na área de estudo? . . . 30
3.2.2 Quais são os países que trabalham no assunto? . . . 31
3.2.3 Quais são as principais revistas e conferências que publicam sobre o assunto ? . . . 33
3.2.4 Quais são os principais temas de pesquisa na área e os métodos utilizados? . . . 34
3.2.5 Quais são os trabalhos mais citados? . . . 36
3.3 Considerações Finais do Capítulo . . . 37

4 Abordagem Proposta **38**
4.1 Visão Geral da Abordagem . . . 38
4.2 Descrição dos Dados . . . 39
4.3 Pré-processamento . . . 44

4.4 Análise Qualitativa dos Dados . . . 45
4.5 Métodos Utilizados para a Extração de Características . . . 46
4.5.1 Método *Wrapper* . . . 46
4.5.2 Método *Filter* . . . 47
4.5.3 Método *Embedded* . . . 47
4.5.4 Algoritmo Genético . . . 47
4.6 Escolha dos Hiperparâmetros . . . 49
4.7 Algoritmos Utilizados para Classificação . . . 50
4.7.1 Árvores de Decisão . . . 50
4.7.2 PRISM . . . 51
4.7.3 Regressão Logística . . . 51
4.7.4 Máquina de Vetores de Suporte . . . 51
4.8 Métricas de Avaliação . . . 52
4.9 Método Aplicado . . . 52
4.10 Ambiente de Teste . . . 53
4.11 Considerações Finais do Capítulo . . . 54

5 Resultados e Discussão da Abordagem **55**
5.1 RQ1 - Como melhorar o processo de otimização de seleção de atributos, identificando os atributos mais relevantes? . . . 55
5.2 RQ2 - Qual o Impacto da etapa de seleção de atributos nos modelos preditivos? 57
5.3 RQ3 - Quais são as vantagens e desvantagens dos modelos com e sem o uso de algoritmos genéticos? . . . 59
5.4 RQ4 - De que forma as explicações geradas por modelos de aprendizado de máquina caixa branca contribuem para entender os fatores determinantes na previsão do sucesso de jogadores universitários no *draft* da NBA? . . . 61
5.4.1 PRISM . . . 62
5.4.2 Árvore de Decisão . . . 62
5.4.3 Análise de Interpretabilidade . . . 64
5.5 RQ5 - Quais são as características fornecidas pela explicabilidade em modelos preditivos caixa branca? . . . 66
5.5.1 Explicabilidade de Modelos - Com atributos selecionados sem Algoritmo Genético . . . 66
5.5.2 Explicabilidade de Modelos - Com atributos selecionados via Algoritmo Genético . . . 67
5.5.3 Análise Estatística . . . 68
5.6 Considerações Finais do Capítulo . . . 70

6 Conclusão **71**
6.1 Conclusões da Pesquisa . . . 71
6.2 Limitações e Trabalhos Futuros . . . 72

Referências bibliográficas **74**

A Primeiro Apêndice **79**

B Segundo Apêndice **105**

Capítulo 1

Introdução

A análise de dados no basquete vem ganhando destaque ao longo dos anos, em que cada vez mais se tenta analisar o jogo em uma maior profundidade, com o intuito de encontrar padrões avançados que possam otimizar o desempenho da equipe e dos jogadores (TICHY, 2016). Tais análises atualmente são realizadas por algoritmos de *Machine Learning* e *Deep Learning*, em que não se analisam apenas os dados estatísticos das partidas, como também, os dados captados por sensores e câmeras.

Considerando a importância financeira do basquete universitário (BORGHESI, 2018), a predição do desempenho, dos jogadores por meio do uso de dados atuais e históricos, tem crescido de forma acentuada, com um foco especial no basquete (KUBATKO et al., 2007). Através desses dados consegue-se obter análises e previsões esportivas, se tornando um campo de estudo atrativo para os profissionais de Inteligência Artificial – IA e Ciência de Dados, por contar com uma grande quantidade de dados gerados pelos jogadores nas partidas.

Neste trabalho, explorou-se um conjuntos de dados históricos do basquete universitário masculino da *National Collegiate Athletic Association* – NCAA, com o objetivo de fornecer informações relevantes aos tomadores de decisão e melhorar seu julgamento ao contratar os profissionais para a NBA.

1.1 Motivação e Contextualização da Pesquisa

Na última década, houve um grande interesse e investimentos em análise esportiva, particularmente na interseção entre Inteligência Artificial e Esportes. Em particular, alguns estudos indicam que técnicas de aprendizado de máquina são uma forma promissora de automatizar

parte dos processos de análise de dados para entender e melhorar o desempenho esportivo (CLACY et al., 2017).

No presente trabalho, focaliza-se o basquete, particularmente a Liga de Basquete Universitário, que é um torneio com 68 equipes, distribuídas em quatro regiões (Leste, Oeste, Sul e Meio-Oeste), levando em consideração tanto o equilíbrio técnico quanto a localização geográfica das universidades (GUMM; BARRETT; HU, 2015). Dentro de cada região, os times são classificados com um número de ranqueamento (*seed*) que varia de 1 a 16, sendo que números menores indicam elencos mais fortes. O chaveamento é estruturado de forma que o time com o menor *seed* enfrente o de maior *seed* em sua região correspondente. Em cada rodada, as equipes vencedoras avançam, enquanto as perdedoras são eliminadas (MELLO, 2024).

Anualmente, 60 jogadores são selecionados sequencialmente para se juntarem a uma das 30 equipes da NBA, sendo a maioria proveniente da liga da *National Collegiate Athletic Association*. Para ter sucesso na NBA, uma equipe precisa selecionar os jogadores de forma eficiente. No entanto, mesmo equipes com as melhores oportunidades de recrutamento frequentemente fazem escolhas ruins. Às vezes, um jogador tratado como prioridade máxima não tem um bom desempenho, enquanto outro jogador selecionado em uma posição de menor prioridade acaba se saindo melhor. Isso se torna desafio em saber qual jogador selecionar com base nas possibilidades e na ordem de recrutamento da equipe (ALAMAR, 2013).

No contexto mencionado acima, obter informações de qualidade, a partir de dados estatísticos coletados em partidas da NCAA, é menos custoso computacionalmente em comparação com as tecnologias sofisticadas relacionadas à visão computacional ou até mesmo uma análise humana, além de ser possível cobrir mais equipes e atletas por meio desse tipo de abordagem, uma vez que é apenas necessário processar os dados coletados, que são facilmente acessíveis.

Este trabalho se propõe a explorar dados do basquete, fazendo o uso da arquitetura apresentada na Figura 1.1, que ilustra os principais elementos abordados neste estudo, destacando dois pontos fundamentais explorados em maior profundidade (representados pelas engrenagens amarelas). Cada componente desempenha um papel crucial no processo de predição, em que a qualidade dos dados e a seleção de atributos (uma das etapas do pré-processamento) impactam diretamente nas previsões e, consequentemente, na explicabilidade dos resultados.

No contexto esportivo, há uma abundância de dados disponíveis, tanto em termos de instâncias quanto de atributos. No entanto, nem todos esses dados são de qualidade suficiente

para garantir uma predição satisfatória. Em outras palavras, uma grande quantidade de dados não necessariamente se traduz em boa qualidade.

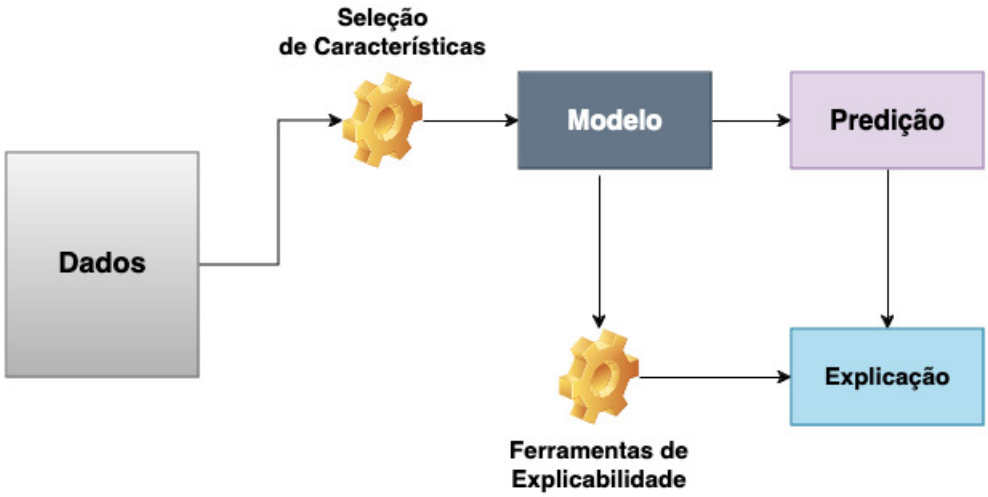

Figura 1.1: Arquitetura Proposta (Adaptado de (LUNDBERG, S. M.; LEE, S.-I., 2017a)

As engrenagens amarelas na Figura 1.1 são o foco desta pesquisa. Inicialmente, o estudo concentra-se na seleção de atributos, visando reduzir a quantidade de dados e melhorar as predições. Isso é realizado através de técnicas tradicionais da literatura, como também através da utilização de Algoritmo Genético. Os atributos selecionados por cada técnica são então usados para a predição em Algoritmos de Classificação caixa branca, com o objetivo de manter a interpretabilidade e a explicabilidade dos modelos. Para fins de comparação, um modelo caixa preta também é utilizado. Na etapa final do trabalho, o objetivo é utilizar ferramentas de explicabilidade, visando tornar os resultados das predições interpretáveis para os humanos, independentemente do tipo de classificador, seja ele de caixa branca ou caixa preta.

1.2 Objetivos da Pesquisa

Os objetivos gerais e específicos do presente estudo, são descritos a seguir:

1.2.1 Objetivos Gerais

Este trabalho tem como objetivo propor uma abordagem de aprendizado de máquina que concilie acurácia e explicabilidade em modelos preditivos. Para isso, utilizamos dados históricos de jogadores de basquete universitário americano, a fim de identificar tendências de sucesso na transição para a liga profissional.

1.2.2 Objetivos Específicos

Para alcançar nosso objetivo geral, foram definidos os seguintes objetivos específicos.

1. Definir uma arquitetura conceitual.
2. Identificar quais características do banco de dados da NCAA são preditores relevantes para determinar a tendência de um jogador chegar à NBA.
3. Aplicar técnicas de seleção de atributos para determinar quais são os mais relevantes para os modelos de aprendizado de máquina.
4. Analisar os modelos de seleção de características que apresentam os melhores resultados utilizando os dados dos jogadores de basquete da NCAA para prever sua tendência de chegar à NBA.
5. Definir o melhor método de seleção de características e o modelo de aprendizado de máquina com melhor desempenho.
6. Avaliar a eficácia do método proposto comparando-o com métodos clássicos.
7. Explorar aspectos de explicabilidade dos modelos, equilibrando com a acurácia desses modelos.

1.3 Questões de Pesquisa

Este trabalho busca responder as seguintes questões de pesquisas:

- **RQ1** - Como melhorar o processo de otimização de seleção de atributos, identificando os atributos mais relevantes?
- **RQ2** - Qual o Impacto da etapa de seleção de atributos nos modelos preditivos?
- **RQ3** - Quais são as vantagens e desvantagens dos modelos com e sem o uso de algoritmos genéticos?
- **RQ4** - De que forma as explicações geradas por modelos de aprendizado de máquina caixa branca contribuem para entender os fatores determinantes na previsão do sucesso de jogadores universitários no *draft* da NBA?

- **RQ5** - Quais são as características fornecidas pela explicabilidade em modelos preditivos caixa branca?

1.4 Estrutura do Trabalho

Este documento está organizado em seis capítulos. Os capítulos subsequentes a esta introdução são descritos a seguir.

- **Capítulo 2:** Apresenta a fundamentação teórica dos temas relevantes para a pesquisa, oferecendo a base conceitual necessária para o entendimento dos tópicos abordados.
- **Capítulo 3:** Apresenta uma revisão de literatura e bibliométrica sobre a área de estudo da presente pesquisa.
- **Capítulo 4:** Apresenta a abordagem proposta nesta pesquisa, incluindo a metodologia adotada e o passo a passo para sua realização.
- **Capítulo 5:** Apresenta e discute os resultados obtidos a partir da aplicação da abordagem proposta, incluindo a análise dos desempenhos dos algoritmos utilizados e a interpretação dos dados gerados.
- **Capítulo 6:** Apresenta as considerações finais do estudo, destacando as principais contribuições da pesquisa, as limitações encontradas, e sugerindo possíveis direções para trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Neste capítulo, apresenta-se a fundamentação teórica que serve de apoio para a proposta deste estudo, abordando os conceitos relacionados aos métodos de seleção de atributos, algoritmos de aprendizado de máquina e à explicabilidade dos modelos.

2.1 Métodos de Seleção Atributos

2.1.1 *Filter*

Considerado como uma das abordagens mais antigas para a seleção de atributos, o método *filter* descarta atributos irrelevantes sem fazer referência a uma técnica de mineração de dados. Eles aplicam uma busca independente, que é principalmente baseada na avaliação das propriedades intrínsecas dos atributos e sua relação com a classe do conjunto de dados (por exemplo, Relief, Incerteza Simétrica, Correlação de Pearson, etc.) (LIU; MOTODA, 2007). A principal vantagem do método *filter* é a sua reduzida complexidade computacional, que se deve ao critério simples e independente utilizado para a avaliação dos atributos. Na maioria dos casos, o método *filter* fornece um ranking baseado em pontuações que refletem a utilidade dos atributos para a classe (ESSEGHIR, 2010).

Na literatura existem alguns métodos de filtragem, como: ganho de informação, teste qui-quadrado, escore de Fisher, coeficiente de correlação e limiar de variância. Entre esses, a correlação de Pearson é especialmente popular, pois quantifica a dependência linear entre duas variáveis, gerando como resultado a dependência no intervalo [-1, 1]. Dadas duas variáveis x_i e y_i, a correlação de Pearson é definida como (LEE RODGERS; NICEWANDER, 1988):

$$\rho = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}, \tag{2.1}$$

Em que:

- ρ é o coeficiente de correlação de Pearson.
- x_i e y_i são os valores individuais das variáveis X e Y, respectivamente.
- $\bar{x}$ e $\bar{y}$ são as médias das variáveis X e Y, respectivamente.
- n é o número total de observações ou pares de dados.

A Figura 2.1 permite uma análise mais clara do funcionamento desse método, destacando as principais etapas que compõem o processo.

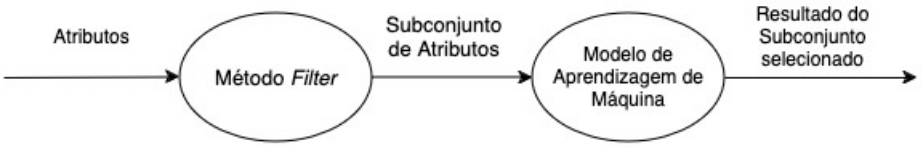

Figura 2.1: Abordagem *Filter* (Autor, 2023)

2.1.2 *Wrapper*

A seleção de atributos baseada no método *wrapper*, tem por característica sua avaliação de maneira simultânea usando um algoritmo de classificação. A exploração dos subconjuntos requer uma estratégia de busca heurística (KOHAVI; JOHN, 1997). Kohavi et al. foram os primeiros autores a defender o *wrapper* como uma estrutura geral para a seleção de atributos em aprendizado de máquina. Muitos estudos utilizaram essa estrutura com diferentes combinações dos componentes de avaliação e busca. As técnicas de busca incluem desde métodos sequenciais gananciosos de seleção de atributos (por exemplo, SFS, SBE, Floating Search (SOMOL et al., 1999)) até métodos randomizados e estocásticos (por exemplo, TABU, BEAM, Algoritmo Genético (GUYON; GUNN et al., 2008)) (ESSEGHIR, 2010).

O método *wrapper* frequentemente oferece resultados melhores do que o método *filter* porque considera apenas um classificador dentro do processo de avaliação. No entanto, é importante notar que os métodos de seleção de atributos baseados em *wrapper* são computacionalmente mais caros em comparação com o método *filter*, devido ao custo do processo iterativo de execução do algoritmo de classificação (GUYON; ELISSEEFF, 2003).

2.1.3 *Embedded*

No método de seleção de atributos *embedded*, a seleção é integrada ou incorporada ao algoritmo do classificador. Durante a etapa de treinamento, o classificador ajusta seus parâmetros internos e determina os pesos/importância apropriados para cada atributo a fim de produzir a melhor precisão de classificação. Portanto, a busca pelo subconjunto ótimo de atributos e a construção do modelo em um método *embedded* são combinadas em uma única etapa (GUYON; ELISSEEFF, 2003). Alguns exemplos de métodos *embedded* incluem algoritmos baseados em Árvores de Decisão (por exemplo, Árvore de Decisão, floresta aleatória, boosting por gradiente) e seleção de atributos usando modelos de regularização (por exemplo, LASSO ou *elastic net*). Métodos de regularização geralmente trabalham com classificadores lineares (por exemplo: SVM, regressão logística), penalizando ou reduzindo o coeficiente de atributos que não contribuem de forma significativa para o modelo (OKSER; PAHIKKALA; AITTOKALLIO, 2013).

A metodologia *embedded* é uma solução intermediária entre métodos *filter* e *wrapper*, pois combinam as qualidades de ambos os métodos (GUO et al., 2019). Especificamente, assim como o método *filter*, os métodos *embedded* são menos exigentes em termos computacionais do que o método *wrapper*. Essa redução na carga computacional ocorre mesmo quando o método *embedded* permite interações com o classificador. Isso significa que o viés do classificador é incorporado na seleção de atributos, o que tende a melhorar o seu desempenho, de maneira semelhante ao que acontece nos métodos *wrapper* (PUDJIHARTONO et al., 2022).

2.1.4 Algoritmo Genético

O Algoritmo Genético (AG) é baseado na teoria da evolução biológica proposta por Darwin (HOLLAND, J. H., 1992), (GOLDBERG; HOLLAND, J., 1988). Essencialmente, o AG simula o princípio da sobrevivência do mais apto: na natureza, organismos que estão melhor adaptados ao seu ambiente têm mais chances de sobreviver e transmitir seus genes para a próxima geração. Com o tempo, os genes que permitem que as espécies se adaptem melhor ao ambiente (evitem predadores e encontrem alimentos) tornam-se predominantes nas gerações subsequentes.

Inspirado pelos cromossomos e genes que encontramos na natureza, o Algoritmo Genético aborda problemas de otimização representando-os como um conjunto de variáveis. Cada so-

lução para um problema é comparada a um cromossomo, enquanto cada gene representa uma variável específica do problema (MIRJALILI et al., 2020).

O funcionamento é da seguinte forma: primeiramente, é criada uma população inicial de soluções. Cada solução representa um conjunto de características que serão utilizadas para treinar o modelo. Cada elemento da solução é uma variável binária que indica se a respectiva característica deve ser incluída ou não no conjunto de atributos (KATO; PAIVA; IZIDORO, 2021).

Em seguida, o algoritmo avalia cada solução na população de acordo com uma função de aptidão que mensura o desempenho do modelo treinado com o conjunto correspondente de características, como pode ser visto na imagem 2.2. Por exemplo, em problemas de classificação, medidas de desempenho como acurácia, *recall*, precisão e *F1-Score* podem ser utilizadas.

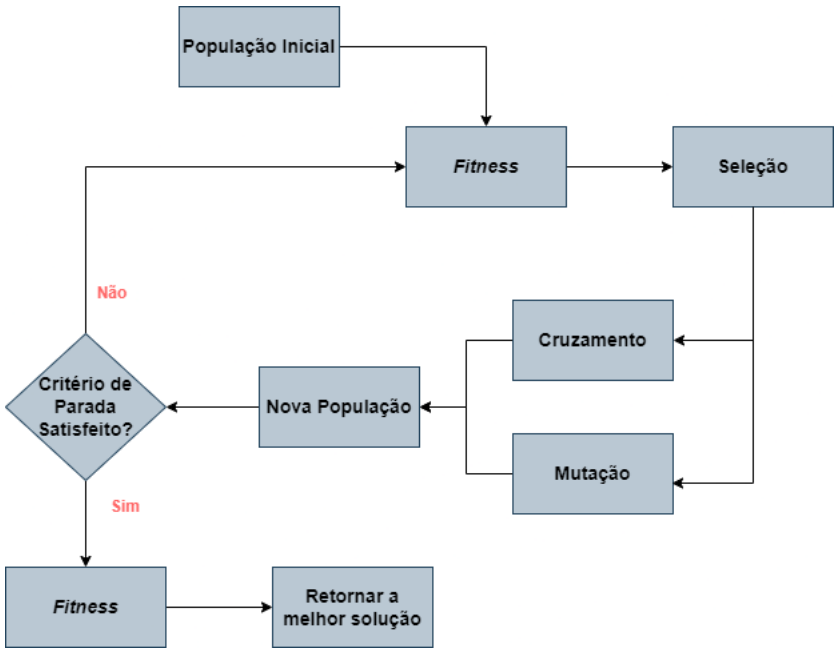

Figura 2.2: Arquitetura Algoritmo Genético (KATO; PAIVA; IZIDORO, 2021)

2.2 Algoritmos de Aprendizagem de Máquina Supervisionados

2.2.1 Indução via Árvores de Decisão

Os algoritmos baseados em Árvores de Decisão são uma poderosa ferramenta para aproximar funções-alvo de valores discretos, de modo que a função aprendida é representada por uma

Árvore de Decisão. Essas árvores podem ser facilmente convertidas em conjuntos de regras *if-then*, o que torna sua interpretação mais intuitiva para os seres humanos. Esses métodos de aprendizado estão entre os mais populares na inferência indutiva e têm sido aplicados com sucesso em diversas áreas, como diagnósticos médicos e avaliação de risco de crédito para candidatos a empréstimos (MITCHELL; MITCHELL, 1997).

Grande parte dos algoritmos desenvolvidos que se baseiam em Árvores de Decisão são variações de um algoritmo central, que utiliza uma busca gulosa, de cima para baixo, no espaço de possíveis Árvores de Decisão.

Os algoritmos funcionam da seguinte forma: dividem recursivamente o conjunto de dados de treinamento em subconjuntos menores com base nos valores de uma variável preditora escolhida, utilizando algum critério, como Entropia ou índice Gini. A divisão é realizada de tal maneira que a variância das classes nos subconjuntos resultantes seja minimizada. Esse processo é repetido em cada subconjunto até que todos os subconjuntos resultantes pertençam a uma única classe (no caso de classificação) ou até que seja alcançado um limite de profundidade da árvore (SINGH; GUPTA, 2014).

2.2.2 Indução via Algoritmo de Regras

A indução por regras é um processo de mineração de dados que deduz regras do tipo *if-then* a partir de um conjunto de dados. Essas regras de decisão simbólicas explicam uma relação inerente entre os atributos e os rótulos de classe no conjunto de dados (ELKAN, 2013). Muitas experiências da vida real são baseadas na indução intuitiva de regras.

Um dos algoritmos de regras utilizados na literatura é o Prism, que foi originalmente desenvolvido para resolver o problema de subárvores replicadas que geralmente ocorre com algoritmos de aprendizado de Árvore de Decisão (CENDROWSKA, 1987). Como um algoritmo de aprendizado de regras, o algoritmo Prism é capaz de selecionar atributos com base em sua importância para uma classe específica. Ou seja, ele seleciona uma classe-alvo e aprende um conjunto de regras que separa a classe-alvo das demais classes; esse processo é repetido selecionando cada classe como a classe-alvo (LIU; COCEA; DING, 2018).

O algoritmo opera da seguinte maneira: começa concentrando-se em uma única classe de interesse, repetindo esse processo para cada classe presente no conjunto de dados, uma de cada vez. Para cada atributo, o algoritmo calcula o valor que melhor separa as instâncias da classe alvo das outras classes. Isso é feito avaliando cada valor do atributo e determinando a

proporção de instâncias da variável dependente que ele cobre. O atributo e valor que melhor discriminam a categoria alvo são selecionados e adicionados à regra. As instâncias que não satisfazem a condição da regra são removidas do conjunto de dados considerado. Esse processo é repetido até que a regra cubra apenas instâncias da variável de interesse ou que nenhuma instância de outras classes seja coberta. (CENDROWSKA, 1987)

2.2.3 Regressão Logística

A Regressão Logística é uma técnica preditiva que visa desenvolver um modelo que permita prever ou explicar os valores assumidos por uma variável dependente qualitativa (muitas vezes binária) a partir de um conjunto de variáveis explicativas quantitativas ou qualitativas (PENG; SO, 2002). A Regressão Logística também pode ser definida como uma técnica para ajustar uma superfície de regressão aos dados quando a variável dependente é dicotômica. Ela é utilizada em estudos que buscam verificar se variáveis independentes podem prever uma variável dependente binária (PENG; LEE, K. L.; INGERSOLL, 2002).

A regressão logística é muito semelhante à da regressão linear. Entretanto, a Regressão Linear é usada para caracterizar as relações entre uma variável quantitativa Y e variáveis explicativas (WEISBERG, 2005). Porém, nesse modelo não se consegue aplicar variáveis qualitativas, especialmente quando Y (variável binária) é expressa em termos de resposta Sim ou Não. Logo, se torna necessário utilizar um modelo adequado para relacionar as variáveis explicativas com a variável qualitativa Y a ser prevista. O truque da regressão logística é não modelar diretamente a variável qualitativa Y, mas sim a probabilidade de que ela se realize (PENG; SO et al., 2002).

2.2.4 Máquina de Vetores de Suporte

O algoritmo de Máquinas de Vetores de Suporte (Support Vector Machines – SVM) surgiu para atender à necessidade de ferramentas de classificação e regressão baseadas em previsões. Introduzido por Vapnik (VAPNIK, 2013), esse algoritmo tem o objetivo de separar os dados em diferentes classes. Onde, dependendo da complexidade dos conjuntos de dados, esses problemas podem ser classificados como lineares ou não lineares. Esse algoritmo atua como uma ferramenta de previsão que busca encontrar uma linha ou uma fronteira de decisão, conhecida como hiperplano, que separa os conjuntos de dados ou classes de forma eficaz, evitando o ajuste excessivo dos dados. Para isso, ele opera em um espaço de hipóteses linear que é

projetado em um espaço de características de alta dimensão. Além disso, é capaz de lidar com dados não lineares através da aplicação de funções de núcleo (SOMVANSHI et al., 2016).

2.3 Explicabilidade em Modelos de Aprendizagem de Máquina

Alguns sistemas de IA são operacionalmente opacos; no entanto, em muitas aplicações, os operadores humanos precisam tomar decisões finais e requerem que as recomendações geradas pela IA possam ser autoexplicativas. Atualmente, até mesmo os especialistas encontram dificuldades para interpretar modelos complexos de IA, como modelos integrados ou modelos de aprendizado profundo. Existe uma grande lacuna entre precisão e interpretabilidade. Diante desse desafio, a XAI (Inteligência Artificial Explicável) é proposta e utilizada para tornar os sistemas de IA mais transparentes (ARRIETA et al., 2020).

Atualmente, existem muitos modelos interpretáveis, a maioria dos quais se enquadra em modelos locais e agnósticos. Os modelos agnósticos comumente usados são principalmente métodos de visualização, extração de conhecimento, métodos de influência e explicações baseadas em exemplos (ZHANG, K.; XU; ZHANG, J., 2020).

Para tornar os modelos de IA mais compreensíveis, uma das ferramentas mais adotadas é o SHAP, que utiliza o valor de Shapley, um conceito originado da teoria dos jogos cooperativos (LUNDBERG, S., 2018). O valor de Shapley avalia a importância de cada característica (LUNDBERG, S. M.; LEE, S.-I., 2017b), ajudando a quantificar a contribuição de cada variável de entrada para o desempenho de um modelo complexo de aprendizado de máquina. Para determinar a importância das características, é calculado o aumento no erro do modelo após a substituição de uma característica. Ou seja, substituir características importantes tende a aumentar o erro do modelo, o que evidencia sua importância para o funcionamento do sistema.

2.4 Considerações Finais do Capítulo

Neste capítulo, apresentamos as técnicas de aprendizado de máquina e metodologias de seleção de características relevantes para prever a tendência de jogadores de basquete universitário ingressarem na NBA. Exploramos as bases teóricas que sustentam o desenvolvimento

dos modelos e o processo de escolha de variáveis, preparando o caminho para uma aplicação prática.

No próximo capítulo, faremos uma revisão da literatura sobre estudos relacionados, abordando abordagens e resultados de trabalhos prévios que investigaram a predição de sucesso no esporte e a aplicabilidade de diferentes técnicas de aprendizado de máquina. Essa revisão servirá como base comparativa e fundamentará a seleção dos métodos utilizados no nosso estudo.

Capítulo 3

Revisão da Literatura e Bibliométrica

A revisão sistemática da literatura tem como intuito auxiliar na compreensão e entendimento de como a área de estudo tem se comportado, quais são os principais algoritmos utilizados, bem como os gargalos encontrados e os problemas já resolvidos. Para analisar o estado da arte, foi realizada uma revisão de literatura junto com uma análise bibliométrica.

Através dessa análise, tem-se o intuito de responder as questões propostas na Tabela 3.1.

Tabela 3.1: Questões Sobre o Escopo da Pesquisa

	Questões
Q1	Quais são os principais autores e instituições que publicam na área de estudo?
Q2	Quais são os países que trabalham no assunto?
Q3	Quais são as principais revistas e conferências que publicam sobre o assunto?
Q4	Quais são os principais temas de pesquisa na área e os métodos utilizados ?
Q5	Quais são os trabalhos mais citados ?

Para responder as questões levantadas na Tabela 3.1, foram exploradas 3 bases de dados: *Scopus*, *Web Of Science* e *IEEE Xplore*. Para realizar as consultas nessas bases de dados, foi necessário desenvolver *strings* de busca para auxiliar na filtragem dos estudos. A *string* de busca foi desenvolvida tendo como base as questões de pesquisas levantadas na Seção 1.3. Diante disso, as *strings* montadas para as buscas, foram:

Tabela 3.2: *Strings* de Busca

RQs	*Strings* de Busca
RQ1	*((sport* analytic* OR machine learning OR artificial intelligence) AND (basketball OR ncaa OR nba) AND (featur* extract* OR featur* selecti*))*
RQ2	*((sport* analytic* OR machine learning OR artificial intelligence) AND (basketball OR ncaa OR nba) AND (predict*)*
RQ3	*((sport* analytic* OR machine learning OR artificial intelligence) AND (basketball OR ncaa OR nba) AND (genetic* algorith*))*
RQ4	*((sport* analytic* OR machine learning OR artificial intelligence) AND (basketball OR ncaa OR nba) AND (predict*) AND (XAI OR SHAP OR Explain* OR Interpret*))*

3.1 Metodologia

A revisão sistemática da literatura é um método de investigação científica utilizado para identificar artigos relevantes por meio de critérios explícitos e reproduzíveis (LINDE; WILLICH, 2003). Essa abordagem tem como objetivo integrar o conhecimento da produção científica sobre um determinado tema, agregando estudos previamente realizados que apresentam diversos tipos de resultados e métodos, com o intuito de identificar e analisar os resultados obtidos por esses estudos, bem como demonstrar os temas pouco explorados na área.

Para este estudo, foi adotado a metodologia PRISMA (MOHER et al., 2015), juntamente com a analise bibliométrica nos artigos selecionados. O processo teve as seguinte etapas:

- Definição do protocolo de pesquisa;
- Análise dos artigos selecionados;
- Síntese dos resultados

Na Figura 3.1, é demonstrado o protocolo PRISMA, explicando passo a passo do processo.

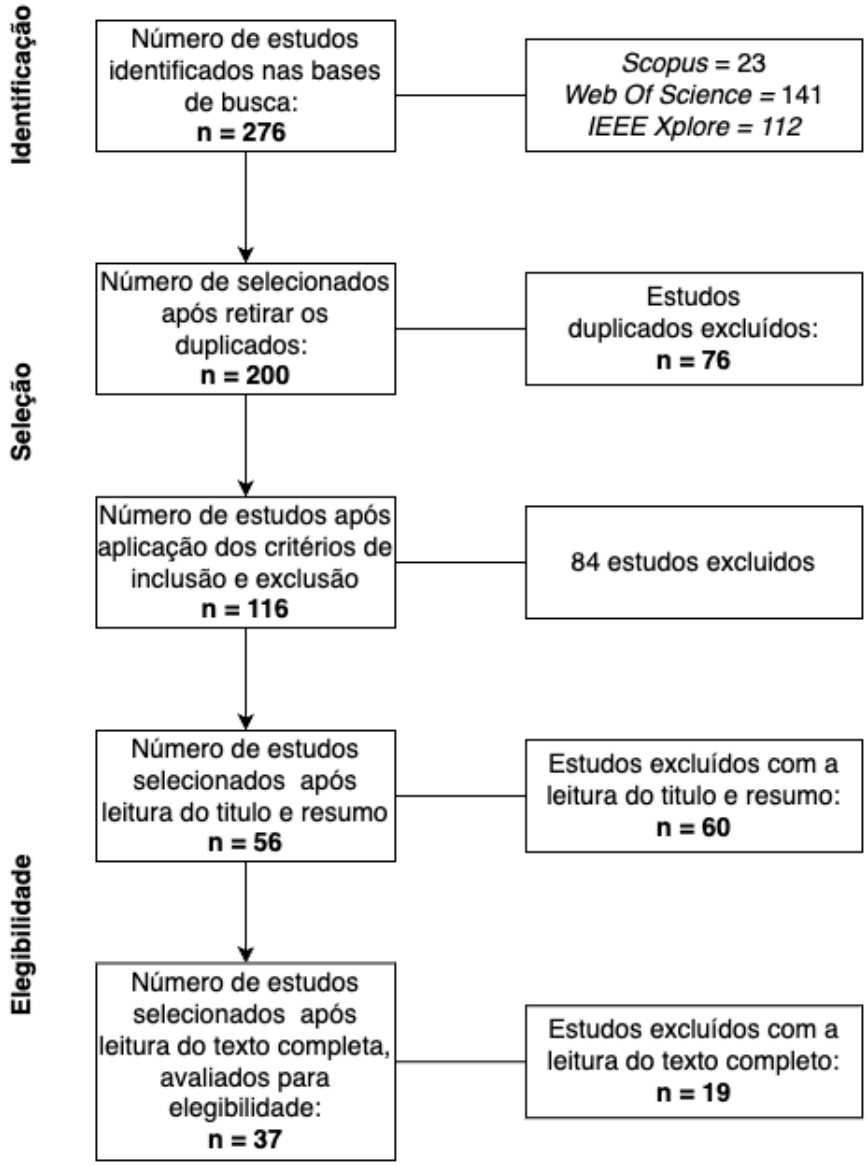

Figura 3.1: Etapas do protocolo PRISMA (Autor, 2023)

Obteve-se um total de 276 artigos como retorno das 3 bases de dados, dos quais 76 eram duplicados, resultando em um total de 200 artigos. Aplicamos os seguintes critérios de inclusão e exclusão aos trabalhos que restaram:

- **Inclusão**
  - Está escrito em Inglês;
  - Está na área de Ciência da Computação ou Ciência Esportiva;
  - O trabalho deve ser Artigo ou Artigo de Conferência;
  - Está no intervalo entre 2017 e 2023

- **Exclusão**
  - *Books* e séries de *book*;
  - Artigo não disponível publicamente ou via *Proxy* da Universidade

Após a aplicação dos critérios, restaram um total de 116 artigos, dos quais deverão ser avaliados com base no título e resumo. Em cada artigo, será analisado se está relacionado ao esporte, e mais especificamente ao basquete.

Ao final, restaram 56 artigos após a exclusão de 60. Os artigos restantes deverão ser analisados por meio da leitura completa do texto, com o objetivo de verificar se a abordagem está voltada para o tema da pesquisa.

Após a leitura completa dos artigos, restaram um total de 37 artigos. Esses serão os artigos que serão submetidos à análise bibliométrica, cujo objetivo será responder as questões levantadas na Tabela 3.1, para o qual foi necessário utilizar a ferramenta Bibliometrix (ARIA; CUCCURULLO, 2017a).

3.2 Análise Bibliométrica dos Artigos

Esta seção está dívida de acordo com as perguntas geradas na Tabela 3.1.

3.2.1 Quais são os principais autores e instituições que publicam na área de estudo?

Na Tabela 3.3, são apresentados os autores mais relevantes na área de estudo, ou seja, aqueles que possuem mais publicações dentro da área de pesquisa. Já na Tabela 3.4, é possível obter um panorama sobre as instituições que possuem maior quantidade de artigos publicados na área. É notório que a maioria das instituições é norte-americana.

Autor	Número de Documentos
Aradhya R	2
Arpitha TC	2
Kiran Kumar HK	2
Maymin P	2
Prithvi BS	2
Sanjay HS	2
Abbasi RA	1
Albert AA	1
Allbright K	1
Annapureddy P	1

Tabela 3.3: Autores mais relevantes

Afiliações	Artigos
Escole Nationale Supérieure En Informatique	4
Sacred Heart University	4
School of Integrated Technology (SIT)	4
Texas Tech University	3
Ahmedabad University	2
Brunel University London	2
Fairfield University Dolan School of Business	2
International Hellenic University	2
International Islamic University	2
Muğla Sıtkı Koçman University	2

Tabela 3.4: Afiliações mais relevantes

3.2.2 Quais são os países que trabalham no assunto?

Na Figura 3.2, é possível analisar os países que mais publicam dentro da área, bem como a quantidade de artigos que foram publicados apenas de forma *single*, ou seja, sem a co-participação de autores de outros países, assim como de forma *multiple*, na qual houve a co-participação de autores de países distintos em um único artigo.

Observa-se que alguns países, como Grécia, Índia e Coreia, não possuem publicações que envolvam autores de outras nacionalidades. Por outro lado, Austrália e Arábia Saudita possuem publicações exclusivamente com a co-participação de autores de outros países. Já nos Estados Unidos, nota-se uma distribuição equilibrada entre as publicações *single* e *multiple*.

Já na Figura 3.3, é possível observar o mapa-múndi, no qual é possível visualizar diferentes países em tonalidades distintas de azul, representando o número de artigos publicados. As

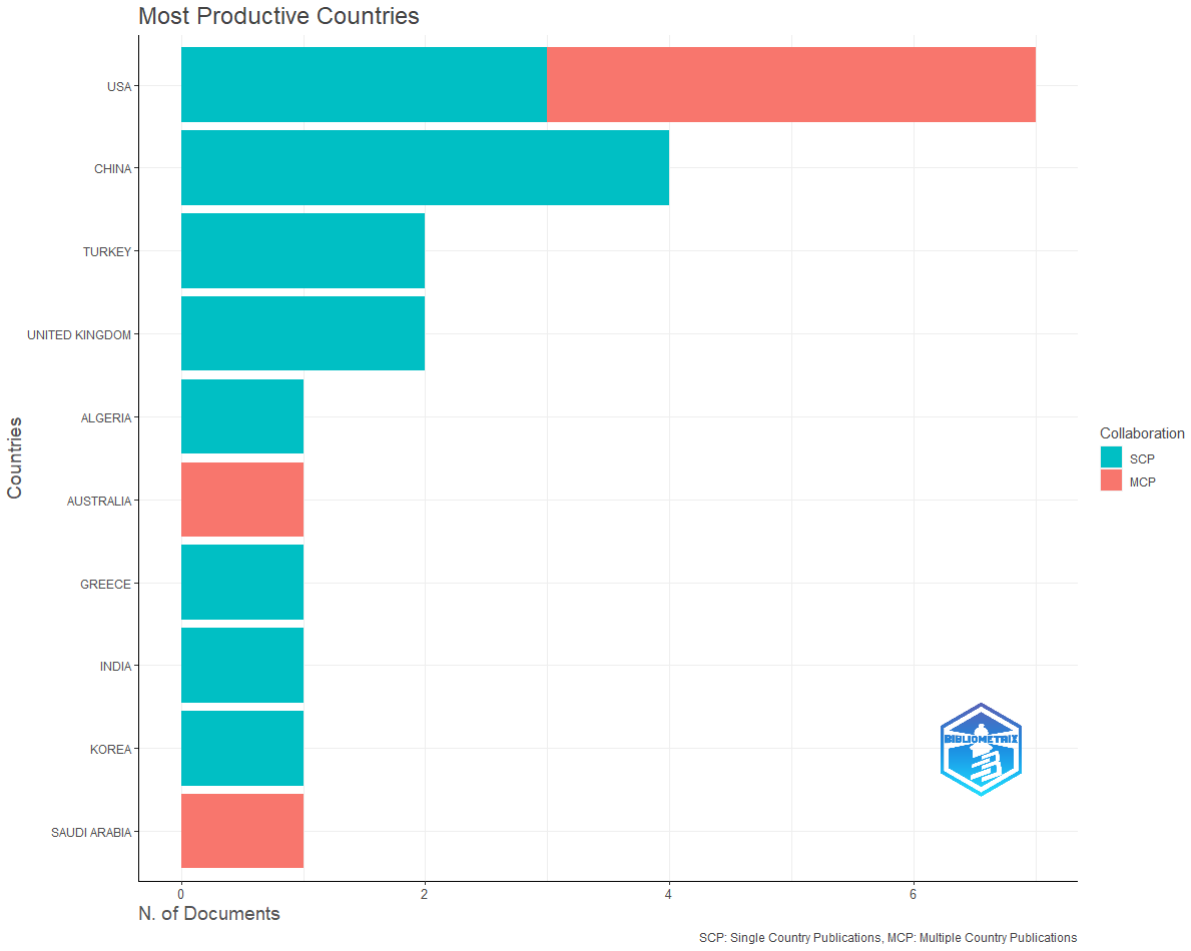

Figura 3.2: Países que mais produzem na área (Autor, 2023)

tonalidades mais intensas indicam um maior número de artigos publicados, enquanto as tonalidades menos intensas correspondem a uma quantidade menor. É perceptível que os Estados Unidos lideram com 33 artigos publicados.

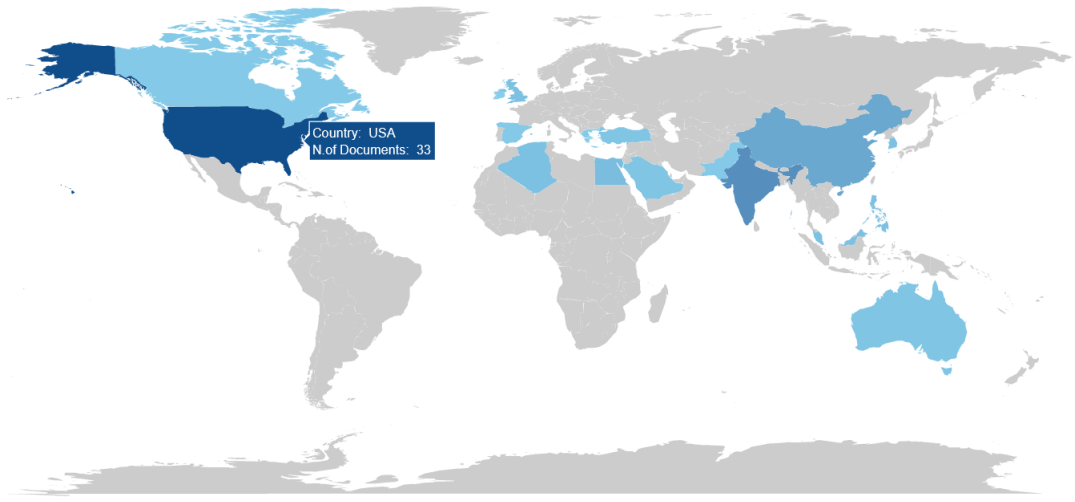

Figura 3.3: Mapa com as produções cientificas por pais (Autor, 2023)

3.2.3 Quais são as principais revistas e conferências que publicam sobre o assunto ?

Na Tabela 3.5, é possível observar as revistas e conferências que possuem mais publicações na área de pesquisa e são consideradas de alta relevância, visto que a maioria está vinculada ao *Institute of Electrical and Electronics Engineers* – IEEE. Além disso, é notório que algumas dessas revistas não tratam apenas de computação, mas também de áreas de esportes. Com isso, observa-se que as quatro primeiras revistas/conferências tiveram cada uma dois artigos publicados, enquanto as demais revistas publicaram apenas um artigo.

Fontes	Número de Documentos
2021 IEEE Mysore Subsection International Conference	2
ACM International Conference Proceedings Series	2
Journal of Business Analytics	2
Proceedings of Spie - The International Society	2
2017 Intelligent Systems Conference	1
2018 International Conference on System Science	1
2019 International Conference on Computer and Information	1
2020 IEEE Congress on Evolutionary Computation	1
2020 International Conference on Advancement	1
2021 IEEE 23RD International Conference	1
6TH International Conference on Electronics	1
Applied Artificial Intelligence	1
Chaos, Solitons and Fractals	1
Electronics (Switzerland)	1
Expert Systems	1
Frontiers in Artificial Intelligence	1
IEEE Access	1
IFIP Advances in Information and Communication	1
Information Systems	1
International Conference on Control	1

Tabela 3.5: Fontes mais relevantes

3.2.4 Quais são os principais temas de pesquisa na área e os métodos utilizados?

Para entender melhor os gráficos a seguir, é importante compreender a estrutura e a representação de cada elemento das figuras (VAN ECK; WALTMAN, 2014), (ARIA; CUCCURULLO, 2017b).

- **Nó**: Cada nó na imagem representa uma unidade de análise — neste caso, palavras-chave nas Figuras 3.4 e 3.5. O tamanho do nó está relacionado à frequência de uso da palavra: quanto maior e mais central o nó, maior a relevância dessa palavra na área de estudo.
- **Arestas**: As arestas conectam os nós e simbolizam a coocorrência de palavras-chave nos artigos. Uma aresta mais espessa entre dois nós indica que essas palavras-chave aparecem juntas com maior frequência, sugerindo uma forte conexão temática entre os tópicos representados.
- ***Clusters***: Grupos de palavras-chave interligadas formam *clusters*, que representam subtemas ou áreas de pesquisa relacionadas. Cada *cluster* pode ser interpretado como um subcampo da pesquisa, mostrando áreas de interesse dos pesquisadores. Isso auxilia na compreensão das divisões temáticas e das tendências do campo, permitindo a visualização de temas emergentes ou já bem estabelecidos.

Na Figura 3.4, é possível identificar algumas áreas de estudo sendo exploradas no contexto do basquete, como predição de estrelas, estrelas em ascensão, análise de fatores, e análises antropométricas, entre outras.

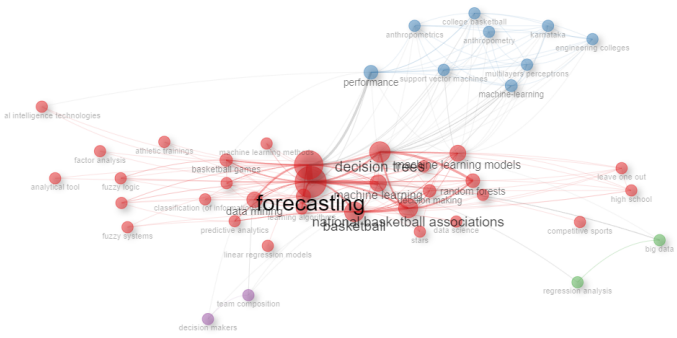

Figura 3.4: Principais temas (Autor, 2023)

Já na Figura 3.5, além de ser possível visualizar toda a gama de áreas de estudo, podem-se notar algumas técnicas utilizadas nesse campo, como regressão logística, SVM, sistemas *Fuzzy*, *Deep Learning*, redes neurais, entre outras técnicas.

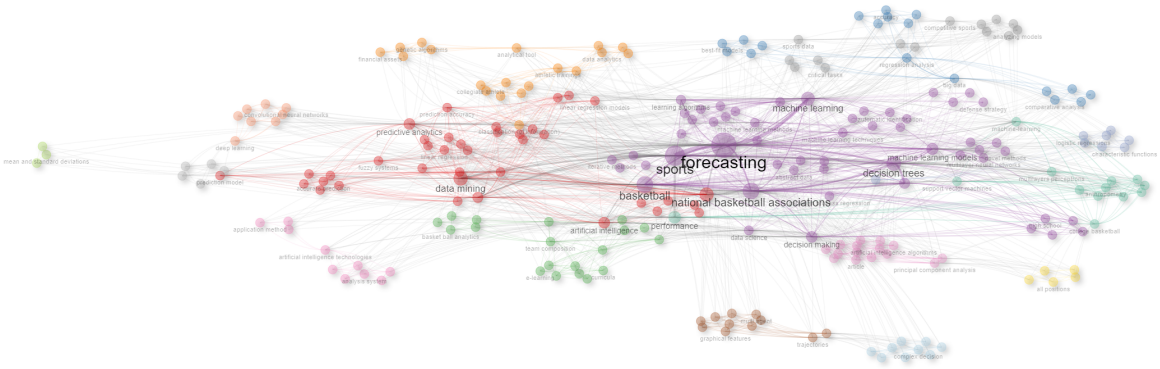

Figura 3.5: Abordagens Utilizadas (Autor, 2023)

No trabalho de (MAHMOOD; DAUD; ABBASI, 2021), é realizada uma análise para determinar se um jogador específico é uma estrela em ascensão na NBA. Para realizar essa análise, foi utilizado um termo denominado *Co-player*, que basicamente se refere a uma pessoa que pertence ao mesmo time ou ao time adversário e que jogou partidas durante algum período comum. O *Co-player*, segundo o autor, é um fator importante para realizar a predição das estrelas em ascensão. Para a predição, foram utilizados os seguintes algoritmos de aprendizado de máquina: SVM, Árvore de Decisão CART, *Maximum Entropy Markov Model* – MEM, Rede Bayesiana e Naive Bayes. No trabalho, foram criados alguns novos atributos, tendo como base os atributos pré-existentes nas bases de dados, como, por exemplo, a média do *Hollinger Score* dos *Co-players* de um jogador P. Ao final do estudo, é demonstrada a importância desses atributos relacionados ao *Co-player* para auxiliar na predição das estrelas.

Na pesquisa realizada por (ALBERT et al., 2021), foi proposta uma abordagem híbrida denominada de ANN (*Adaboost*, *Random Forest* e *Multi Layer Perceptron* – MLP), na qual os modelos utilizam o mesmo conjunto de dados como entrada. Segundo o autor, essa combinação ponderada dos três modelos convencionais não tem sido objeto de pesquisa, tornando-a assim uma abordagem inovadora para o problema de predição de estrelas na NBA. Essa combinação foi obtida a partir dos resultados individuais de uma variedade de modelos de aprendizagem de máquina testados, e os três modelos citados resultaram em melhores métricas em termos de sensibilidade e especificidade. Essa ANN foi construída para ser uma camada oculta de uma *Recurrent Neural Network* – RNN. Ao refazer os testes com o modelo proposto, os autores ob-

tiveram um valor de 90% de especificidade e 80% de sensibilidade, a especificidade diminuiu um pouco em relação aos modelos individuais, entretanto houve um aumento significativo no valor de sensibilidade.

O trabalho de (HSU et al., 2018) tenta prever as dezesseis melhores equipes da NBA por meio da aplicação de algoritmos de aprendizado de máquina, com base nas características dos jogadores. Essas características estão relacionadas às estatísticas de pontos, bloqueios, rebotes ofensivos e defensivos, entre outras métricas de jogo. Os modelos aplicados devem calcular a contribuição vencedora dos jogadores para a equipe. Para obter esse resultado, foram empregados os seguintes modelos: Regressão Polinomial, *Random Forest Regression* e SVM. Para realizar a comparação entre os modelos, foi utilizada uma classificação de eficiência do jogador denominada como PER.

O sistema híbrido inteligente proposto por (OZKAN, 2020) foi uma combinação entre uma rede neural artificial e lógica *fuzzy*. Esse sistema neuro-*fuzzy* concorrente foi estabelecido para determinar qual equipe irá vencer, levando em consideração dados sobre o sucesso geral da equipe, o desempenho nos últimos jogos e a qualidade dos jogadores. A rede neural foi desenvolvida com o intuito de prever qual time (o time da casa ou o time visitante) venceria o jogo com base em alguns parâmetros. O sistema de inferência *fuzzy* desenvolvido foi capaz de prever o favorito do jogo, a fim de aumentar a precisão e sensibilidade geradas pela rede neural.

Através dos trabalhos citados anteriormente, é possível observar um panorama geral de como o problema de pesquisa está sendo abordado e quais algoritmos de aprendizagem de máquina estão sendo utilizados na literatura para tanger o problema. Dito isso, este trabalho foca nos dados dos jogadores de basquete da liga universitária, buscando prever a tendência de sucesso desses jogadores na liga profissional por meio de algoritmos de aprendizado de máquina em conjunto com Algoritmo Genético.

3.2.5 Quais são os trabalhos mais citados?

Na Tabela 3.6, é possível visualizar os trabalhos mais citados na área de estudo, sendo que (SARLIS; TJORTJIS, 2020) é o estudo com o maior número de citações. Esse artigo tem como objetivo realizar uma comparação entre as análises de desempenho existentes na literatura para avaliar equipes e jogadores.

O trabalho de (JAIN; KAUR, 2017), o segundo mais citado de acordo com a Tabela 3.6, tem como objetivo prever os resultados de um jogo de basquete, considerando algumas estatísticas dos jogadores. Este trabalho tentou abordar o problema através do algoritmo SVM. Entretanto, mesmo sendo um algoritmo poderoso para classificação, possui a limitação de não produzir regras que auxiliem na tomada de decisão. Para contornar essa limitação do algoritmo, o trabalho propõe um modelo híbrido *Fuzzy*-SVM para a geração de regras. No qual, em cada ponto de entrada do SVM, é aplicada uma função de pertinência *fuzzy*, possibilitando diferentes contribuições no aprendizado. Esse aprimoramento também reduz o efeito de ruídos e *outliers* nas entradas de dados, diminuindo diretamente o efeito do erro.

Tabela 3.6: Artigos e Número de Citações

Artigos	**Número Citações**
(SARLIS; TJORTJIS, 2020)	84
(JAIN; KAUR, 2017)	38
(OZKAN, 2020)	18
(NGUYEN et al., 2022)	18
(OUGHALI; BAHLOUL; EL RAHMAN, 2019)	17
(CALIWAG et al., 2018)	13
(SOLIMAN; MISBAH; ELDAWLATLY et al., 2017)	10

3.3 Considerações Finais do Capítulo

Neste capítulo, revisamos a literatura relevante sobre a aplicação de aprendizado de máquina para a previsão de sucesso de jogadores de basquete e outras áreas correlatas. Essa revisão nos permitiu fundamentar as decisões metodológicas deste trabalho, oferecendo um embasamento teórico para a implementação prática que será discutida no próximo capítulo.

No Capítulo 4, será descrita a abordagem prática proposta, com foco no processo de predição e explicabilidade dos modelos. Esse processo será estruturado em etapas, conforme o método KDD, incluindo a descrição dos dados, o pré-processamento, a seleção de características e a classificação, além das métricas de avaliação e o ambiente de teste.

Capítulo 4

Abordagem Proposta

Neste capítulo, apresenta-se a abordagem associada ao processo de predição e explicação do modelo. Assim, foi dividido em etapas ligadas ao método *Knowledge Discovery in Databases* – KDD que foram abordadas, como: Descrição dos Dados, Pré-processamento, Algoritmos Utilizados para Extração de Características e Classificação, Métricas de Avaliação, Método Aplicado e Ambiente de Teste.

4.1 Visão Geral da Abordagem

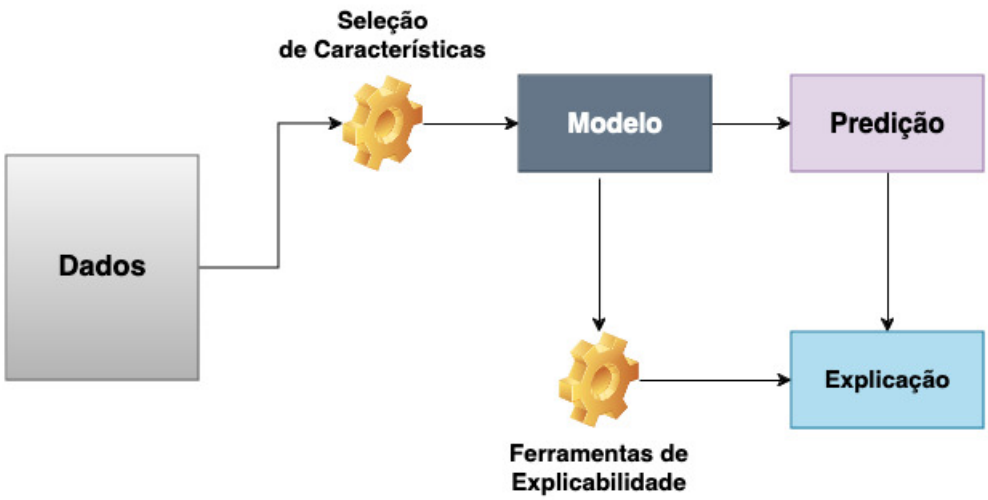

Figura 4.1: Arquitetura Proposta (Adaptado de (LUNDBERG, S. M.; LEE, S.-I., 2017a))

A abordagem proposta neste trabalho segue um fluxo sistemático que visa equilibrar a eficiência preditiva com a explicabilidade dos modelos. Conforme ilustrado na Figura 4.1, o

processo pode ser dividido em três fases principais: Seleção de Características, Predição e Explicabilidade, todas derivadas dos dados brutos fornecidos.

Na primeira fase, os Dados passam por um processo de Seleção de Características, de tal modo que os atributos irrelevantes ou redundantes são filtrados para otimizar a performance do modelo. Essa seleção busca não apenas melhorar a acurácia dos algoritmos de aprendizado de máquina, mas também reduzir a carga computacional, facilitando o entendimento das variáveis mais relevantes para o processo preditivo.

Na segunda fase, o conjunto de dados filtrado é alimentado no Modelo, que realiza o processo de Predição. Essa etapa centraliza a atividade de modelagem preditiva, na qual o desempenho é avaliado com base na métrica de acurácia gerada por cada modelo.

Por fim, na terceira fase, entram em ação as Ferramentas de Explicabilidade, responsáveis por fornecer uma análise interpretativa das predições feitas pelo modelo, por meio da geração de explicações. O uso dessas ferramentas permite que os tomadores de decisão compreendam como os atributos selecionados influenciam as previsões, promovendo uma maior transparência no processo de tomada de decisões. Isso é crucial para garantir que os resultados possam ser interpretados, não apenas com base na acurácia preditiva, mas também em termos de justificativas claras e compreensíveis.

Também é possível notar que algumas predições já podem gerar explicações de forma natural, sem a necessidade de passar pelas ferramentas de explicabilidade. Isso ocorre por meio de predições realizadas por modelos baseados em árvore de decisão, por exemplo, onde uma árvore é gerada de forma transparente, permitindo que os tomadores de decisão compreendam como cada atributo influencia a predição final, de forma intuitiva e de fácil interpretação.

Esse fluxo garante que, além de atingir um bom desempenho preditivo, o modelo também seja capaz de justificar suas decisões, alinhando-se com a crescente demanda por modelos de aprendizado de máquina explicáveis.

4.2 Descrição dos Dados

O conjunto de dados selecionado abrange o período de 2009 a 2021 e envolve atletas universitários americanos que competiram na NCAA. Esse intervalo foi definido em função da disponibilidade de informações na base de dados utilizada, que cobre exclusivamente esses anos, limitando assim a análise a esse período temporal. O banco de dados é composto por 65

características relacionadas aos atletas e às partidas disputadas por eles. Os dados foram coletados da plataforma *Kaggle*, no seguinte endereço: **`https://www.kaggle.com/datasets/adityak2003/college-basketball-players-20092021`**. A variável alvo é o atributo ***ROUND***, que representa a rodada em que um jogador é selecionado. Os demais atributos do *dataset* estão listados abaixo.

1. ***Player Name:*** Nome do jogador.
2. ***Team:*** Time do jogador.
3. ***Conference:*** Nome da Conferência.
4. ***Games Played:*** Total de games jogado pelo jogador.
5. ***Minutes Percentage:*** Minutos jogados pelo jogador dividido pelo tempo total de jogo da equipe na temporada. Mostra a porcentagem total que o jogador esteve em quadra na temporada.
6. ***Offensive Rating (ORtg):*** Mede o desempenho ofensivo de um time ou a eficiência de um jogador individualmente na produção de pontos para o ataque.
7. ***Usage (usg%):*** Mostra a porcentagem estimada de jogadas do time que um jogador utilizou.
8. ***Effective Field Goal Percentage (eFG%):*** Demostrar o porcentagem efetiva dos arremessos em campo, pode ser calculada para cada jogador ou por time.
9. ***True Shooting Percentage (TS%):*** É a porcentagem de arremessos ajustada para arremessos de três pontos e lances livres e mede a eficiência de um jogador no arremesso da bola.
10. ***Offensive Rebound Percentage (ORB%):*** Representa a porcentagem de possíveis rebotes ofensivos que o jogador pode pegar enquanto estava em quadra.
11. ***Defensive Rebound Percentage (DRB%):*** Representa a porcentagem de possíveis rebotes defensivos que o jogador pode pegar enquanto estava em quadra.
12. ***Assist Percentage (AST%):*** Estima a porcentagem de posses de uma equipe que um jogador auxilia enquanto está na quadra.

13. ***Turnover Percentage (TOV%):*** Estimativa de *turnovers* de uma equipe que um jogador tem enquanto está na quadra.

14. ***Free Throws Made (FT):*** Quantidade de lances livres que um jogador fez na temporada.

15. ***Free Throw Attempts (FTA):*** Quantidade de lances livres tentados de um jogador na temporada.

16. ***Free Throw Percentage (FT%):*** Lances livres realizados divididos por tentativas de lance livre.

17. ***Two Points Made (2P):*** Quantidade de arremesso de 2 pontos realizadas por um jogador na temporada.

18. ***Two Point Attempts (2PA):*** Quantidade das tentativas de arremesso de 2 pontos do jogador na temporada.

19. ***Two Point Percentage (2P%):*** Arremesso de 2 pontos realizados divididos pelas tentativas de arremesso de 2 pontos.

20. ***Three Points Made (3P):*** Quantidade de arremesso de 3 pontos realizadas por um jogador na temporada.

21. ***Three Point Attempts (3PA):*** Quantidade das tentativas de arremesso de 3 pontos do jogador na temporada.

22. ***Three Point Percentage (3P%):*** Arremesso de 3 pontos realizados divididos pelas tentativas de arremesso de 3 pontos.

23. ***Block Percentage (BLK%):*** A porcentagem de bloqueios de uma equipe que um jogador realiza enquanto está em campo.

24. ***Steal Percentage (STL%):*** A porcentagem dos roubos de bola de uma equipe que um jogador realiza enquanto está em campo.

25. ***Free Throw Rate (FTR):*** A taxa de lance livre é encontrada dividindo as tentativas de lance livre pelas tentativas de arremesso de campo.

26. ***Year in College:*** É uma variável categórica que pode assumir os valores de Calouro, Segundo ano, Terceiro ano, Quarto ano.

27. ***Height:*** Altura do jogador.

28. ***Points Over Replacement Per Adjusted Game (Porpag):*** É uma métrica que auxilia prever quantos pontos a mais um jogador pode marcar em comparação com um substituto hipotético.

29. ***Adjusted Offensive Efficiency (AdjOE):*** É uma estimativa dos pontos por 100 posses que uma equipe marcaria contra a defesa de uma equipe média da divisão 1.

30. ***Personal Foul Rate (pfr):*** Faltas pessoais a cada 40 minutos.

31. ***Year:*** Ano da temporada.

32. ***Player ID (pid):*** ID dos jogadores.

33. ***Assist / Turnover (ast/tov):*** Assistências do jogador divididas por *turnovers* do jogador.

34. ***Rimmade:*** O número de arremessos convertidos próximos ao aro.

35. ***Rimmade + Rimmiss:*** O número de arremessos convertidos próximos ao aro mais o número de arremessos perdidos próximos ao aro.

36. ***Midmade:*** Número de arremessos convertidos de média distância.

37. ***Midmade + Midmiss:*** Número de arremessos convertidos de média distância mais número de arremessos perdidos de média distância.

38. ***Rimmade / (Rimmade + Rimmiss):*** Obtido dividindo o número de arremessos convertidos próximos ao aro, pelo total de tentativas de arremesso próximo ao aro.

39. ***Midmade / (Midmade + Midmiss):*** Obtido dividindo o número de arremessos convertidos pelo jogador, pelo total de tentativas de arremesso de média distância pelo jogador.

40. ***Dunksmade:*** Quantidade de enterradas realizadas pelo jogador.

41. ***Dunksmiss + Dunksmade:*** Número de enterradas perdidas pelo jogador mais o número de enterradas realizadas pelo jogador.

42. ***Dunksmade / (Dunksmade + Dunkmiss):*** Número de enterradas realizadas pelo jogador dividido pelo total de tentativas de enterradas do jogador.

43. ***Pick:*** Mostra o status do *draft* dos jogadores. 1 para selecionado, 0 para não selecionado.

44. ***Defensive Rating (drtg):*** É uma estatística usada para medir a eficiência de um jogador individual em impedir que o outro time marque pontos.

45. ***Adjusted Defensive Rating (adrtg):*** É uma estatística que mede a eficiência defensiva de uma equipe, levando em consideração vários fatores, como o ritmo de jogo, a força do adversário e outros elementos.

46. ***Stops:*** *Stops* são bloqueios, roubadas de bola, rebotes defensivos, além da probabilidade de *turnovers* e arremessos errados realizados pelo jogador, que não resultaram em roubo de bola ou bloqueio.

47. ***Box Plus/Minus (bpm):*** É uma métrica baseada na estatística de pontuação do basquete que estima a contribuição de um jogador de basquete para a equipe enquanto esse jogador está em quadra.

48. ***Offensive Box Plus/Minus (obpm):*** É uma estatística que mede a contribuição ofensiva de um jogador para sua equipe quando ele está em quadra. Essa métrica estima quantos pontos por 100 posses de bola um jogador contribui para a ofensiva de sua equipe acima ou abaixo de um jogador médio

49. ***Defensive Box plus/minus (dbpm):*** É uma estatística que mede a contribuição defensiva de um jogador para sua equipe quando ele está em quadra. Essa métrica estima quantos pontos por 100 posses de bola um jogador contribui para a defesa de sua equipe acima ou abaixo de um jogador médio

50. ***Game Box Plus/minus (gbpm):*** Semelhante à métrica **BPM**, porém concentra-se exclusivamente em um único jogo.

51. ***Offensive Game Box Plus/Minus (ogbpm):*** Semelhante à métrica **OBPM**, porém concentra-se exclusivamente em um único jogo.

52. ***Defensive Game Box Plus/Minus (dgbpm):*** Semelhante à métrica **DBPM**, porém concentra-se exclusivamente em um único jogo.

53. ***Minutes Played (mp):*** Total de minutos que um jogador passou em quadra.

54. ***Offensive Rebounds (oreb):*** Total de rebotes ofensivos do jogador.

55. ***Defensive Rebounds (dreb):*** Total de rebotes defensivos do jogador.
56. ***Total Rebounds (treb):*** Soma dos rebotes defensivos e ofensivos do jogador.
57. ***Assist (ast):*** Total de assistências que o jogador realizou.
58. ***Steal (stl):*** Total de roubos de bola que o jogador realizou.
59. ***Block (blk):*** Total de bloqueios que o jogador realizou.
60. ***Points (pts):*** Total de pontos que o jogador marcou.
61. ***Position (Pstn):*** Posição do jogador em quadra.
62. ***Threes Per 100 Possessions (TPA / 100):*** Tentativas de arremessos de três pontos do jogador por 100 posses de bola.

O trabalho de (GÜLER, 2022) utiliza a mesma base de dados, entretanto, o foco do autor é abordar sobre o balanceamento, através da utilização de técnicas ja existentes na literatura, demonstrando a importância de possuir dados balanceados no momento da classificação nos algoritmos de aprendizagem de máquina.

4.3 Pré-processamento

Pré-processamento é uma etapa do KDD que várias técnicas são aplicadas aos dados com o objetivo de melhorar as taxas de aprendizado dos modelos. O trabalho de Blum (BLUM; LANGLEY, 1997) mostra a importância da seleção de características para modelos de aprendizado de máquina. Neste trabalho, serão aplicados quatro métodos de seleção: *Embedded*, *Filter*, *Wrapper* e Algoritmo Genético.

Na etapa de pré-processamento dos dados, alguns atributos foram retirados por apresentarem baixa relevância para os resultados do estudo, não contribuindo significativamente para a análise preditiva. A decisão de excluir esses atributos foi baseada na falta de relação direta com o objetivo da pesquisa. Os atributos removidos foram os seguintes: "AFFILIATION", "TEAM", "year", "ROUND.1", "OVERALL","Unnamed: 64", "Unnamed: 65","pick", "Rec Rank","player_name", "team", "conf","yr","ht","type","porpag", "num"

4.4 Análise Qualitativa dos Dados

Na matriz de correlação gerada em `https://qualitative-analysis.vercel.app/`, foram obtidos os valores de p-valor e do coeficiente de Pearson (ρ), que ajudam a identificar a força e significância das relações entre as variáveis de interesse. Para conduzir a análise de correlação, estabelecemos as seguintes hipóteses:

- **Hipótese Nula (H)**: Não existe correlação significativa entre <atributo 1> e <atributo 2> , indicando que a correlação observada é nula ou próxima de zero.
- **Hipótese Alternativa (H)**: Existe uma correlação significativa entre <atributo 1> e <atributo 2>.

O p-valor, neste contexto, representa a probabilidade de observar uma correlação tão extrema (ou mais extrema) quanto a observada, assumindo que a hipótese nula seja verdadeira. Um p-valor menor que 0,05 sugere que a correlação é estatisticamente significativa, ou seja, indica que há menos de 5% de probabilidade de que o resultado obtido ocorra apenas devido ao acaso, o que nos leva a rejeitar a hipótese nula em favor da hipótese alternativa.

Já o coeficiente de Pearson (ρ) mede a força e a direção da associação entre duas variáveis de forma monotônica, independentemente de uma relação estritamente linear. Valores de ρ próximos de +1 indicam uma forte correlação positiva; valores próximos de -1, uma forte correlação negativa; e valores próximos de 0 indicam ausência de correlação. A Tabela 4.1 apresenta uma escala para interpretação dos valores de ρ, facilitando a compreensão do grau de associação entre as variáveis analisadas.

Tabela 4.1: Interpretação dos índices de correlação, (SCHOBER; BOER; SCHWARTE, 2018)

Pearson (ρ)	**Interpretação**
0.9 - 1.0	Correlação Muito Forte
0.7 - 0.89	Correlação forte
0.4 - 0.69	Correlação moderada
0.10 - 0.39	Correlação fraca
0.0 - 0.10	Correlação insignificante

Além da análise geral dos p-valores e dos coeficientes de Pearson, identificamos padrões específicos nas correlações entre variáveis ofensivas, defensivas e avançadas, oferecendo uma visão mais detalhada sobre o desempenho dos jogadores:

- **Fortes Correlações Positivas**
  - Estatísticas Ofensivas: As variáveis relacionadas a pontos, porcentagem de acertos de arremessos, porcentagem de lances livres e eficiência de arremesso estão altamente correlacionadas. Esse padrão sugere que jogadores que se destacam em uma dessas áreas tendem a se sobressair em outras estatísticas ofensivas.
  - Estatísticas Defensivas: Existe uma forte correlação entre roubos de bola, bloqueios e rebotes defensivos, o que indica que jogadores com bom desempenho em uma dessas áreas tendem a ser habilidosos defensivamente de forma geral.
- **Correlações Positivas Moderadas**
  - Relação entre Ataque e Defesa: Rebotes e assistências apresentaram correlação positiva moderada com pontos, sugerindo que, em média, jogadores que pegam mais rebotes ou realizam mais assistências também tendem a marcar mais pontos.
- **Correlações Negativas**
  - *Turnovers* e Pontos: Existe uma correlação negativa entre *turnovers* e pontos, indicando que jogadores que cometem menos erros de posse tendem a pontuar mais.

4.5 Métodos Utilizados para a Extração de Características

4.5.1 Método *Wrapper*

Para este trabalho, foi escolhido o método de *Forward Stepwise*, que funciona da seguinte maneira: primeiramente, é iniciado um conjunto vazio A. Em seguida, é realizado um teste com as características para selecionar qual característica será adicionada fixamente ao conjunto A. Ou seja, de acordo com a métrica escolhida, é feita uma comparação entre as características e aquela com melhor desempenho é selecionada. Com a primeira característica selecionada, realiza-se uma nova busca para verificar qual será a próxima característica fixada juntamente com a primeira, repetindo essas etapas para todas as características do conjunto de dados. No final, o melhor conjunto de características é retornado.

4.5.2 Método *Filter*

Para o método *Filter*, optamos por implementar a correlação de Pearson. Esta escolha se baseia na capacidade do método de quantificar relações lineares entre variáveis, atribuindo coeficientes num intervalo de [-1, 1]. Essa métrica não só evidencia a força da relação, como também a direção, sendo fundamental para identificar variáveis influentes e minimizar redundâncias no modelo.

4.5.3 Método *Embedded*

Neste trabalho, foi utilizada a Árvore de Decisão C5.0 (PANDYA, R.; PANDYA, J., 2015) como o modelo para esta abordagem, mas outros métodos como C4.5 (QUINLAN, 2014), CART (ANDERSON, 1983) e ID3 (ANDERSON, 1983) podem ser encontrados na literatura.

4.5.4 Algoritmo Genético

Para que o Algoritmo Genético pudesse realizar a redução de dimensionalidade no conjunto de dados, foi desenvolvido um algoritmo híbrido no qual a função de aptidão (*fitness*) utilizou algoritmos de classificação para determinar qual deles melhor auxiliaria na resolução do problema. Inicialmente, utilizou-se um algoritmo de caixa branca, como a Árvore de Decisão CART, e em seguida foi utilizado um algoritmo de caixa preta, como o SVM. Também foi utilizado um Algoritmo Genético puro, para assim conseguir realizar uma análise entre os algoritmos genéticos. Todas as combinações utilizaram a métrica de acurácia para avaliar a aptidão dos cromossomos.

Cada cromossomo foi representado como um vetor binário, onde cada posição corresponde a um atributo do conjunto de dados. Um valor ’1’ indica a inclusão do atributo no subconjunto utilizado para treinar o classificador, enquanto ’0’ indica a exclusão. Essa representação binária permite uma exploração eficiente do espaço de busca, possibilitando a avaliação de diversas combinações de atributos.

Com base na função de aptidão, o algoritmo seleciona os melhores indivíduos na população e utiliza operadores genéticos, como cruzamento e mutação, para gerar novas soluções. O operador de cruzamento combina as características de dois indivíduos selecionados aleatoriamente, enquanto o operador de mutação altera aleatoriamente as características de um indivíduo selecionado.

O cruzamento implementado foi do tipo ponto único aleatório no cromossomo, onde ocorre a troca de elementos entre os pais. A Figura 4.2 ilustra o funcionamento deste tipo de cruzamento, onde um ponto é escolhido aleatoriamente para dividir e combinar os cromossomos dos pais (HASSANAT et al., 2019). Ao utilizar apenas o operador de cruzamento para gerar os descendentes faz com que o Algoritmo Genético fique preso em pontos ótimos locais, pois os genes benéficos contidos nos pais sobrevivem em cada geração, resultando na constante ocorrência de ótimos locais (HASSANAT et al., 2019). Para mitigar esse problema, o operador de mutação é utilizado, garantindo que os filhos gerados sejam distintos dos pais e, assim, promover a diversidade na população. (KOREJO et al., 2013).

A mutação por inversão é uma das técnicas utilizadas para diversificar a população. Nesse tipo de mutação, um segmento do cromossomo é selecionado aleatoriamente e tem sua ordem de genes invertida (KATO; PAIVA; IZIDORO, 2021). A Figura 4.3 exemplifica esse processo, onde o segmento escolhido é invertido, criando uma nova combinação genética no cromossomo mutado.

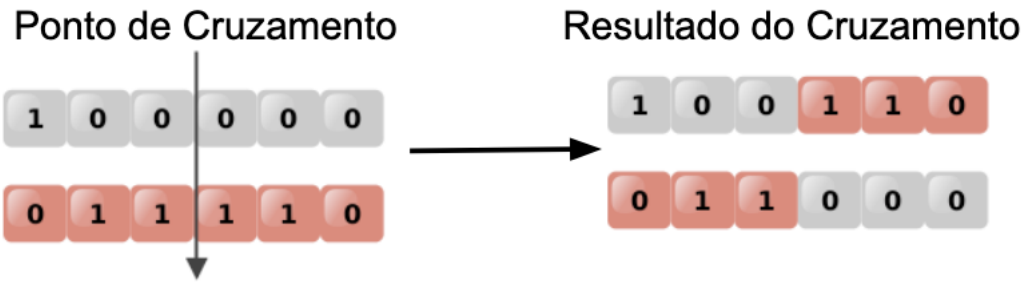

Figura 4.2: Cruzamento 1 Ponto de Corte (Adaptado de (KATO; PAIVA; IZIDORO, 2021))

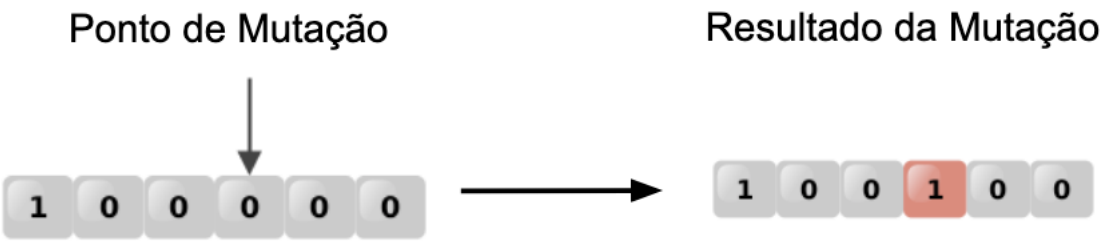

Figura 4.3: Mutação por Inversão (Adaptado de (KATO; PAIVA; IZIDORO, 2021))

As taxas de mutação e cruzamento foram comprovadas como elementos-chave para o sucesso dos algoritmos genéticos (BEASLEY; BULL; MARTIN, 1993). DeJong sugere que os

valores ótimos para o tamanho da população estejam na faixa de [50-100], com uma taxa de mutação baixa, uma vez que altas taxas de mutação podem levar a uma busca aleatória (SCHLIERKAMP-VOOSEN, 1993). Quanto ao cruzamento com um ponto de corte, é aconselhável que esteja próximo de 60%. Esses parâmetros têm sido amplamente utilizados em implementações que empregam algoritmos genéticos (SCHLIERKAMP-VOOSEN, 1993).Na literatura, existem muitos trabalhos que abordam a escolha ótima de parâmetros para o AG, os quais podem auxiliar na configuração adequada do algoritmo.

No caso deste trabalho, as taxas de mutação e cruzamento foram ajustadas conforme essas diretrizes. A taxa de mutação foi definida como 30%, e a taxa de cruzamento foi ajustada para 70%.

Esse processo de avaliação, seleção, mutação e cruzamento é repetido várias vezes até que a condição de parada seja alcançada. Na literatura, existem várias condições de parada (SAFE et al., 2004), tais como:

- Número fixo de gerações alcançado (SAFE et al., 2004).
- A variação da aptidão da população é menor que um pequeno valor predefinida (o nível de aptidão desejado foi atingido) (SAFE et al., 2004).
- Nenhuma melhoria no melhor valor de aptidão (EIBEN; SMITH, 2015).

Neste trabalho, foram utilizadas duas condições de parada: a primeira é quando a população atinge um número máximo de 10.000 gerações; a segunda é se um indivíduo for o melhor durante 10 gerações consecutivas. Quando uma das condições de parada é satisfeita, o algoritmo retorna o subconjunto de características correspondente à melhor solução encontrada durante o processo. Esse subconjunto deve ser capaz de representar com precisão os dados e reduzir a dimensão do problema, resultando em um modelo mais simples e interpretável.

4.6 Escolha dos Hiperparâmetros

Os algoritmos de aprendizado de máquina requerem a configuração de hiperparâmetros e, para identificar os hiperparâmetros que melhor representavam o conjunto de dados, foi necessário realizar o *GridSearch*.

O *GridSearch* é uma abordagem sistemática para automatizar o processo de ajuste de parâmetros de um algoritmo, gerando e avaliando várias combinações de parâmetros. A combina-

ção que melhor representa o conjunto de dados é selecionada como a mais adequada (MISHRA et al., 2019).

Para descobrir os melhores hiperparâmetros para cada classificador, foi necessário realizar o *GridSearch* em cada sub-conjunto de atributos extraidos dos algoritmos de seleção, comentados na Seção 4.5. O algoritmo foi executado com *cross validation* com valor de 5, e a métrica utilizada foi a acurácia. A escolha desse valor foi um compromisso entre a necessidade de obter uma estimativa precisa do desempenho do modelo e a limitação do tempo computacional.

O resultado da execução pode ser encontrada no apêndice A.

4.7 Algoritmos Utilizados para Classificação

Após a etapa de pré-processamento dos dados, foram selecionados alguns algoritmos de classificação com o objetivo de diversificar as abordagens em modelos de aprendizado de máquina, dando preferência aos modelos caixa branca, a fim de manter a interpretabilidade e explicabilidade. Seguindo esse raciocínio, foram incluídas as abordagens baseadas em árvores, como C4.5, C5.0 e CART, além de um modelo baseado em regras, o algoritmo PRISM, e um modelo estatístico, a regressão logística. Além disso, para explorar o potencial dos dados, foi incluído um modelo caixa preta baseado em *kernel*, o SVM, para analisar o comportamento dos dados em um modelo de baixa interpretabilidade. Esses algoritmos foram implementados através das bibliotecas *Scikit-Learn* (KRAMER; KRAMER, 2016) e *ChefBoost* (SERENGIL, 2021).

Para a execução dos algoritmos, foi necessária a divisão dos dados em conjuntos de treino e teste, utilizando o método *holdout* (DEVROYE; WAGNER, 1979), com a seguinte proporção: 70% para treino e 30% para teste. A escolha do método *holdout* se deu pela sua simplicidade e eficiência, uma vez que ele permite uma divisão direta dos dados, sem a necessidade de múltiplas iterações, como ocorre em técnicas mais complexas.Esses algoritmos serviram como base para compreender o comportamento do conjunto de dados em relação às métricas de desempenho de cada abordagem

4.7.1 Árvores de Decisão

Neste trabalho, foram utilizados três tipos de algoritmos baseados em Árvore de Decisão. A seguir, são apresentados os atributos definidos por cada algoritmo.

- **CART** - Para este algoritmo, foram fixados dois atributos e outros quatro foram definidos por meio da técnica de *gridSearch*. Os atributos fixos foram: *Random_state* = 33 e *Criterion* = Gini. Os atributos definidos pelo *gridSearch* foram: *max_depth* = ?[1], *max_features* = ?, *min_samples_leaf* = ?, *min_samples_split* = ?, e *Splitter* = ?.

- **C5.0** - De forma análoga ao algoritmo CART, apenas os atributos *Random_state* = 33 e *Criterion* = Entropy foram fixados, enquanto os demais foram ajustados por meio do *gridSearch*.

- **C4.5** - A biblioteca utilizada para implementar este algoritmo não permitia a realização de testes para encontrar os melhores hiperparâmetros. Assim, apenas o critério de escolha da árvore pôde ser ajustado, definido como *Criterion* = Entropy.

4.7.2 PRISM

Para a aplicação deste algoritmo, o conjunto de dados foi dividido em treinamento e teste. Durante o treinamento, foram calculadas as probabilidades condicionais das variáveis preditoras em relação à variável alvo (*ROUND*), o que resultou em uma tabela de probabilidades. A partir desta tabela, a regra com a maior probabilidade foi selecionada, e o conjunto de dados foi atualizado, removendo os exemplos que se encaixavam nesta regra. Este processo foi repetido até que não restassem mais exemplos, resultando em um conjunto de regras.

4.7.3 Regressão Logística

Para o algoritmo de regressão logística, os seguintes hiperparâmetros foram ajustados por meio da técnica de *gridSearch*: *max_iter* = ?, *penalty* = ? e *solver* = ?.

4.7.4 Máquina de Vetores de Suporte

Para o algoritmo de Máquinas de Vetores de Suporte (SVM), os seguintes hiperparâmetros foram ajustados utilizando a técnica de *gridSearch*: *C* = ?, *gamma* = ? e *kernel* = ?.

[1] Atributos que foram descobertos através do *GridSearch*

4.8 Métricas de Avaliação

A métrica de acurácia foi selecionada como medida de comparação entre os algoritmos, visando avaliar o desempenho de cada um de acordo com os parâmetros de entrada utilizados em cada tipo de seleção de características. Para entender melhor a métrica que será utilizada, deve-se entender primeiro alguns conceitos. Na Tabela 4.2 pode-se observar uma matriz de confusão, na qual é possível indicar os erros e acertos do modelo de aprendizagem de máquina, comparando assim com os resultados esperados.

	Positivo - Detectado	**Negativo - Detectado**
Positivo - Real	Verdadeiro Positivo (VP)	Falso Negativo (FN)
Negativo - Real	Falso Positivo (FP)	Verdadeiro Positivo (VN)

Tabela 4.2: Matriz de Correlação

A partir da matriz na Tabela 4.2, é possível calcular diferentes métricas de avaliação para os modelos de aprendizagem de máquina.

Métrica	**Fórmula**
Acurácia	$\frac{VP+VN}{VP+VN+FP+FN}$
Precisão	$\frac{VP}{VP+FP}$
Recall	$\frac{VP}{VP+FN}$
F1-Score	$\frac{2*precisao*recall}{precisao+recall}$

Tabela 4.3: Fórmulas das Métricas utilizadas

Na Tabela 4.3, observa-se a composição matemática das fórmulas utilizadas para mensurar os modelos de aprendizagem de máquina. Contudo, neste estudo, será considerada apenas a fórmula da acurácia como métrica principal para avaliação dos modelos aplicados ao conjunto de dados. Apesar disso, as matrizes de confusão, que fornecem uma visão mais detalhada do desempenho dos modelos, serão disponibilizadas no apêndice A para consulta.

4.9 Método Aplicado

O método aplicado neste trabalho pode ser dividido em etapas, que consistem em:

- **1ª** - Aplicar os dados pré-processados em modelos de aprendizado de máquina;
- **2ª** - Aplicar os dados pré-processados em técnicas de seleção de atributos;

- **3ª** - Aplicar as características extraídas na 2ª etapa em modelos de aprendizado de máquina;

Após o pré-processamento dos dados, restaram 53 atributos. Em seguida, foi realizado um teste experimental com o intuito de avaliar o desempenho dos modelos de classificação utilizando essa quantidade de atributos. Esse teste também serviu como base comparativa para análises posteriores do desempenho após a aplicação de técnicas de seleção de características. Esse processo corresponde à 1ª etapa.

Na 2ª etapa, os mesmos 53 atributos foram aplicados às técnicas de seleção mencionadas na Seção 4.5, utilizando 4 métodos distintos. O objetivo dessa etapa foi realizar uma análise dos resultados obtidos, tanto em relação à quantidade quanto à qualidade desses atributos, quando utilizados como entrada para os modelos de classificação.

Feita a seleção das características, elas foram utilizadas como entrada para os modelos de classificação mencionados na Seção 4.7. Ao final desta etapa, foram obtidos os resultados para serem utilizados como comparativo com as outras etapas que também envolveram algoritmos de classificação.

4.10 Ambiente de Teste

O ambiente de teste utilizado para fazer as execuções foi o *Google Computer Engine* – Google Colab [2], bem como um *Notebook Dell Inspiron*.

O *Google Colab*, possui as seguintes configurações:

- RAM: aproximadamente 13 GB;
- HD: aproximadamente 108 GB;

Enquanto o *Dell Inspiron*, é composto pelas seguintes configurações:

- Sistema Operacional: Windows 11, 64 bits
- Processador: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz;
- RAM: 16 GB;
- SSD: 1 TB;

[2] `https://colab.research.google.com`

Para construir e executar os algoritmos foi utilizado o *Python* 3, e também algumas bibliotecas, como:

- *Scikit-learn* versão 1.2.2 para construção dos modelos de *machine learning* (KRAMER; KRAMER, 2016);
- *Chefboost* versão 0.0.17 para construção de modelos de *machine learning* (SERENGIL, 2021);
- Pandas versão 2.0.2 para manipulação dos *Data Sets*;
- Numpy versão 1.25.0 para manipular operações em *arrays* multidimensionais.

4.11 Considerações Finais do Capítulo

Neste capítulo, detalhamos a abordagem prática proposta, explicando cada etapa do processo de predição e de explicação do modelo. Desde a descrição dos dados e o pré-processamento até a seleção de características e a aplicação dos modelos, cada fase foi planejada para equilibrar a precisão preditiva com a interpretabilidade dos resultados.

No Capítulo 5, serão apresentados e discutidos os resultados obtidos a partir dessa abordagem. As análises serão organizadas em torno das perguntas de pesquisa, proporcionando uma visão estruturada sobre o impacto de cada método e técnica na qualidade e interpretabilidade das previsões. Essa estrutura permitirá avaliar o desempenho e a relevância das características selecionadas, oferecendo uma compreensão crítica dos resultados alcançados.

Capítulo 5

Resultados e Discussão da Abordagem

Este capítulo será dividido de acordo com as perguntas de pesquisa estabelecidas na Seção 1.3. Assim, será subdividido em subseções para auxiliar na visualização e compreensão dos resultados apresentados.

5.1 RQ1 - Como melhorar o processo de otimização de seleção de atributos, identificando os atributos mais relevantes?

A seguir, será possível visualizar os resultados obtidos com a aplicação das técnicas de extração de características mencionadas na Seção 4.5. Na Tabela 5.1, será possível verificar a técnica utilizada e o número total de características restantes após a aplicação.

Técnica de Seleção	Total de Atributos
AG - Puro	5
AG - CART	6
Wrapper	16
Filter	20
AG - SVM	34
Embedded	46

Tabela 5.1: Método de Seleção de Características pelo Total de Atributos

Na Tabela 5.1, é possível observar que o número de características selecionadas pelos algoritmos resultou em valores distintos. O Algoritmo Genético (puro) obteve o menor número

de características relevantes, representando o conjunto de dados de 65 atributos com apenas 5 características, o que representa uma redução superior à 90% do *dataset* original.

Técnicas	Atributos Selecionados
AG - Puro	TS_per, ORB_per, DRB_per, FTA, FT_per
AG - CART	dbpm, adrtg, ogbpm, TPM, stops, treb
Wrapper	DRB_per, FT_per, twoP_per, TP_per, stl_per, pfr, ast/tov, dporpag, obpm, dbpm, gbpm, mp, ogbpm, dreb, treb, stl
Filter	dporpag, stops, twoPM, twoPA, pts, FTM, FTA, dreb, treb, bpm, mp, Min_per, rimmade, midmade+midmiss, rimmade+rimmiss, midmade, gbpm, ogbpm, adjoe, obpm
AG - SVM	eFG, TPM, DRB_per, TS_per, dunksmade, dbpm, blk_per, pfr, twoPA, TP_per, drtg, usg, dunksmade/(dunksmade+dunksmiss), ORB_per, dunksmiss+dunksmade, Min_per, gbpm, FTM, ast, oreb, obpm, TO_per, twoP_per, midmade+midmiss, dporpag, AST_per, stl_per, rimmade/(rimmade+rimmiss), stl, GP, adrtg, rimmade+rimmiss, pid, midmade/(midmade+midmiss)
Embedded	dporpag, gbpm, adjoe, midmade/(midmade+midmiss), adrtg, AST_per, ast/tov, rimmade+rimmiss, pid, ORB_per, dunksmade, ast, stl_per, pts, midmade, stl, FTA, TP_per, oreb, Ortg, GP, twoPA, dunksmiss+dunksmade, rimmade/(rimmade+rimmiss), dgbpm, Min_per, TPA, twoPM, usg, FTM, mp, treb, stops, twoP_per, drtg, blk, obpm, dbpm, eFG, DRB_per, pfr, TO_per, bpm, TS_per, rimmade, ogbpm

Tabela 5.2: Características selecionadas pelas técnicas

Na Tabela 5.2, é possível visualizar as características que foram selecionadas por cada método, e através disso pode-se observar que cada técnica resultou em uma saída distinta, a qual melhor representa os dados. Com base nisso, é possível concluir que:

Existem algumas características que são comuns entre os métodos, como observado abaixo:

1) Entre os métodos *Wrapper*, *Filter* e *Embedding*, as características que se repetiram nos três foram: dporpag, gbpm, mp, obpm, ogbpm e treb;

2) Entre os métodos Algoritmo Genético com Árvore de Decisão CART e SVM, as características que se cruzaram foram: TPM, adrtg e dbpm;

3) Entre os métodos Algoritmo Genético (puro) e AG com SVM, as características que se repetiram em ambos foram: TS_per, ORB_per, DRB_per;

4) Em relação às características comuns entre os métodos *Wrapper*, *Filter*, *Embedding* e Algoritmo Genético com Árvore de Decisão CART, apenas duas características apareceram em todos os quatro métodos: ogbpm e treb;

5) Nos métodos *Wrapper*, *Embedding*, Algoritmo Genético com Árvore de Decisão CART e SVM, apenas uma característica foi comum: dbpm;

6) Entre os 3 métodos de seleção de atributos que utilizaram o Algoritmo Genético, não houve resultado comum.

7) Entre os seis métodos de seleção de atributos, não houve resultado comum.

Embora nenhuma característica tenha se repetido em todos os modelos, é possível observar que algumas delas são consideradas mais relevantes em quatro dos modelos, incluindo os seguintes atributos:

- *Defensive Box Plus Minus* – DBPM, representa a diferença entre os pontos que a equipe permite com o jogador em quadra e fora dela.
- *Total Rebound Percent* – TREB, estima a porcentagem de rebotes que um jogador teve enquanto estava em quadra.
- *Offensive Game Box Plus Minus* – OGBPM, representa o impacto ofensivo de um atleta com base em uma fórmula derivada de outros atributos: (Pontos + 0,2 x Assistências + 0,7 x Rebotes Ofensivos - 0,7 x Tentativas de Arremesso) / Minutos Jogados.

5.2 RQ2 - Qual o Impacto da etapa de seleção de atributos nos modelos preditivos?

Para responder à essa pergunta de pesquisa, foi considerada a execução dos algoritmos da Seção 4.7 com os dados originais após o pré-processamento, que envolveu a remoção de dados ruidosos. E também, os dados após passarem pelos métodos de seleção de atributos tradicionais. Os algoritmos foram executados com os atributos listados na Tabela 5.2 para cada técnica

utilizada, não considerando ainda a utlização do Algoritmo Genético, ele será explorado em uma outra pergunta de pesquisa.

Resultados com todos os atributos	
Modelos	**Acurácia**
Cart	75,00%
C5.0	74,14%
C4.5	77,65%
PRISM	58,62%
Regressão Logística	73,83%
SVM	58,87%

Tabela 5.3: Resultado da predição sem seleção de atributos (Autor, 2024)

Resultados com todos os métodos tradicionais de seleção de atributos		
Método de Seleção	**Modelos**	**Acurácia**
Wrapper	Cart	70,71%
	C5.0	74,76%
	C4.5	76,90%
	PRISM	56,09%
	Regressão Logística	76,01%
	SVM	74,14%
Filter	Cart	78,81%
	C5.0	77,25%
	C4.5	77,65%
	PRISM	55,66%
	Regressão Logística	76,63%
	SVM	78,19%
Embedded	Cart	73,20%
	C5.0	73,20%
	C4.5	77,65%
	PRISM	58,03%
	Regressão Logística	73,83%
	SVM	58,87%

Tabela 5.4: Resultado da predição com métodos de seleção de atributos tradicionais, (Autor, 2024)

5.3 RQ3 - Quais são as vantagens e desvantagens dos modelos com e sem o uso de algoritmos genéticos?

Esta pergunta de pesquisa busca avaliar as vantagens e desvantagens dos algoritmos de classificação ao serem combinados ou não com algoritmos genéticos para a seleção de atributos. Para isso, foram considerados os algoritmos mencionados na Seção 4.7, utilizando os atributos de saída gerados pelos algoritmos genéticos listados na Tabela 5.2.

Serão examinadas três variações do Algoritmo Genético, cada uma com uma função de aptidão específica. Os resultados dos preditores serão obtidos utilizando os atributos selecionados pelo melhor indivíduo de cada população. Essa abordagem permitirá uma análise comparativa das vantagens e desvantagens dos algoritmos com e sem o uso de algoritmos genéticos na seleção de atributos.

Resultados gerados na seleção de atributos com Algoritmo Genético		
Método de Seleção	**Modelos**	**Acurácia**
Algoritmo Genético com CART	Cart	73,52%
	C5.0	73,20%
	C4.5	75,22%
	PRISM	53,14%
	Regressão Logística	73,20%
	SVM	72,58%
Algoritmo Genético com SVM	Cart	80,00%
	C5.0	72,00%
	C4.5	77,03%
	PRISM	59,65%
	Regressão Logística	76,01%
	SVM	59,87%
Algoritmo Genético	Cart	67,60%
	C5.0	67,60%
	C4.5	XX
	PRISM	60,65%
	Regressão Logística	67,00%
	SVM	67,28%

Tabela 5.5: Resultado da predição com metodós de seleção de atributos tradicionais, (Autor, 2024)

É póssivel observar na Tabela 5.5 que os Algoritmo Genético com a função de aptidão CART e SVM geraram os melhores subconjunto de atributos, apresentando ganhos em relação aos métodos de seleção tradicionais e à ausência de métodos de seleção. Quando se fala em vantagens da utilização do AG, fica evidente que a primeira vantagem está relacionada ao aumento na métrica de acurácia. Ao comparar com as Tabelas 5.4 e 5.3, é notável que houve

um ganho nas métricas. Por exemplo, a Árvore de Decisão do tipo CART apresentou uma acurácia inicial de 75%. Em contraste, nos métodos de seleção tradicionais, a acurácia variou entre 70% e 78%. Com a utilização dos atributos gerados pelo AG, foi possível alcançar uma margem de 80%.

Outra vantagem evidente na utilização do AG está relacionada à geração do conjunto mínimo de atributos. Na Tabela 5.2, pode-se notar que o AG puro e o AG com a Árvore de Decisão do tipo CART, na função de aptidão, geraram os dois menores subconjuntos de atributos. No entanto, esses atributos ao serem submetidos aos algoritmos classificadores, produziram resultados satisfatórios, embora não tenham superado os resultados obtidos pelo AG com SVM na função de aptidão.

A desvantagem encontrada na utilização do AG está relacionada ao custo computacional e ao tempo de resposta. Dependendo da quantidade dos atributos de entrada, o algoritmo pode demandar uma quantidade significativa de recursos computacionais, resultando em tempos de execução mais longos. Esse aumento no tempo de processamento pode tornar o uso de AGs menos viável em situações em que a rapidez de resposta é crucial.

5.4 RQ4 - De que forma as explicações geradas por modelos de aprendizado de máquina caixa branca contribuem para entender os fatores determinantes na previsão do sucesso de jogadores universitários no *draft* da NBA?

A interpretabilidade dos modelos de *machine learning* é essencial em diversas áreas, inclusive na análise esportiva, onde transparência e compreensão das decisões são fundamentais. O uso de modelos interpretáveis, como PRISM e Árvores de Decisão, visa identificar e compreender os fatores mais relevantes para o sucesso de jogadores universitários de basquete na transição para a NBA. A seguir, detalhamos como essas técnicas fornecem *insights* sobre os atributos mais influentes, facilitando a tomada de decisão e aumentando a confiança nas previsões geradas.

- Se **TS_per** = 51.15, então **ROUND** = 1.0
- Se **TS_per** = 51.44, então **ROUND** = 2.0
- Se **ORB_per** = 11.90, então **ROUND** = 0.0
- Se **FTA** = 137.00, então **ROUND** = 2.0
- Se **ORB_per** = 4.9, então **ROUND** = 1.0

Figura 5.1: Regra gerada via PRISM

5.4.1 PRISM

Na Figura 5.1, é possível observar uma das regras geradas pelo algoritmo PRISM. Esta regra estabelece que, se o valor de TS_per de um determinado ponto de dados for exatamente **51,15**, então, com base no conjunto de treinamento, esse ponto de dados pertence à primeira rodada do *draft* (ROUND = 1). Assim, o PRISM permite que identifiquemos condições específicas sob as quais jogadores têm maior probabilidade de serem selecionados em rodadas iniciais, o que contribui para uma análise mais detalhada dos fatores críticos no processo de seleção.

5.4.2 Árvore de Decisão

Na Figura 5.2, apresentamos uma saída gerada pela Árvore de Decisão - CART, usando um conjunto específico de atributos de entrada. Essa árvore utiliza as variáveis **TS_per**, **FTA** e **FT_per** para classificar as amostras, facilitando a compreensão de como cada variável influencia na classificação dos jogadores. A seguir, destacam-se alguns pontos importantes para análise, que indicam como cada divisão reflete a probabilidade de um jogador ser selecionado para a NBA com base em suas características de desempenho:

1. **Primeira Divisão**:
    - **TS_per <= 30.335**: Divide as amostras em dois grupos. Se **TS_per** é menor ou igual a 30.335, todas as 84 amostras são da classe 0 (à esquerda).
    - Se **TS_per** é maior que 30.335, a árvore faz uma nova divisão com base em **FTA**.
2. **Segunda Divisão (Para TS_per > 30.335)**:

- **FTA <= 239.691**: Divide novamente as amostras. Se **FTA** é menor ou igual a 239.691, há outra divisão com base em **FTA**.

3. **Terceira Divisão (Para FTA <= 239.691)**:

   - **FTA <= 100.533**: Se **FTA** é menor ou igual a 100.533, a maioria das amostras são da classe 0.
   - Se **FTA** é maior que 100.533, as amostras se dividem mais entre as classes, com predominância da classe 1.

4. **Divisão Final (Para FTA > 239.691)**:

   - **FT_per <= 0.619**: Se **FT_per** é menor ou igual a 0.619, a maioria das amostras são da classe 1.
   - Se **FT_per** é maior que 0.619, ainda há predominância da classe 1, mas com alguma diversidade.

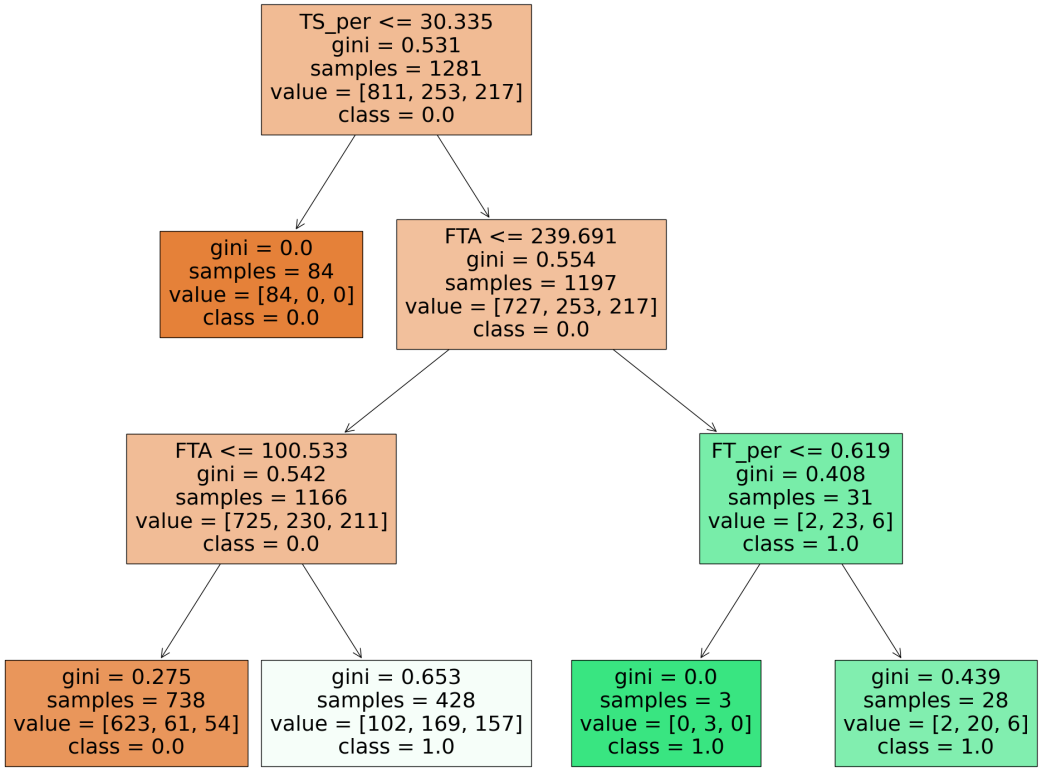

Figura 5.2: Árvore de Decisão - CART

A Árvore de Decisão revela como as variáveis **TS_per**, **FTA** e **FT_per** influenciam na classificação das amostras. Cada nó representa uma decisão baseada em um valor específico de uma variável, e cada folha final (nó terminal) mostra a classe predominante e a distribuição das amostras nesse grupo. Este tipo de visualização é útil para entender a lógica de decisão do modelo e identificar quais atributos são mais importantes para a classificação, auxiliando diretamente no entendimento de quais fatores impactam a transição dos jogadores para a NBA.

5.4.3 Análise de Interpretabilidade

Ao avaliar a interpretabilidade das técnicas empregadas, alguns aspectos se destacam em relação ao contexto deste estudo:

- **Transparência**:
  - **Algoritmos de Regras**: Oferecem transparência com regras simples e diretas, como as que explicam o papel de variáveis como **TS_per** para a seleção dos jo-

gadores. No entanto, podem tornar-se complexos com muitos atributos, o que dificultaria a análise de fatores específicos para o *draft*.

  - **Árvores de Decisão**: Proporcionam uma estrutura hierárquica clara das decisões, facilitando tanto a interpretação global quanto a local, o que é útil para o entendimento da importância de variáveis como **FTA** e **FT_per**.

- **Complexidade**:

  - **Algoritmos de Regras**: Podem gerar muitas regras, especialmente em conjuntos de dados complexos, dificultando a compreensão, sobretudo quando o objetivo é identificar as variáveis-chave que influenciam o sucesso na transição para a liga profissional.
  - **Árvores de Decisão**: A complexidade pode ser controlada por técnicas de poda, mantendo a interpretabilidade, o que torna essa abordagem vantajosa ao estudar tendências de sucesso com um número controlado de variáveis relevantes.

- **Flexibilidade**:

  - **Algoritmos de Regras**: Permitem a adição ou remoção modular de regras, mas a interação entre regras pode ser complexa. Esse aspecto pode limitar sua aplicabilidade no contexto do basquete, onde variáveis inter-relacionadas desempenham papéis significativos.
  - **Árvores de Decisão**: Menos flexíveis na modificação de uma única condição, pois cada nó está interligado hierarquicamente, mas, ao mesmo tempo, essa hierarquia permite uma visualização mais intuitiva dos principais fatores.

Ambos os métodos têm suas vantagens e desvantagens em termos de interpretabilidade. Algoritmos de regras, como o PRISM, são simples e modulares, mas podem tornar-se complexos com muitos dados. Árvores de Decisão são visivelmente claras e hierárquicas, mas podem sofrer de *overfitting* e sensibilidade aos dados.

5.5 RQ5 - Quais são as características fornecidas pela explicabilidade em modelos preditivos caixa branca?

Para facilitar a interpretação global no SHAP, foi selecionado o gráfico *beeswarm*, que permite identificar padrões e tendências nas contribuições das características para as previsões do modelo. Por exemplo, se um ponto aparece consistentemente à direita (ou à esquerda) do gráfico, isso pode sugerir que a característica correspondente tem um impacto positivo (ou negativo) nas previsões. Além disso, a densidade dos pontos em diferentes regiões do gráfico pode revelar a importância relativa das características em várias faixas de valores SHAP.

Duas subseções foram definidas para analisar a interpretabilidade dos modelos. A primeira, referenciada em 5.5.1, foca nos atributos que não utilizaram o Algoritomo Génetico na etapa de seleção. Enquanto que a subseção, 5.5.2 será dedicada à avaliação de modelos que estão utilizando os atributos gerados pelo Algoritmo Genético.

5.5.1 Explicabilidade de Modelos - Com atributos selecionados sem Algoritmo Genético

Nesta subseção, discutiremos o modelo que apresentou os melhores resultados utilizando os atributos gerados por meio de técnicas de seleção de atributos que não utilizaram algoritmos genéticos. O algoritmo a ser analisado é a Árvore de Decisão do tipo CART, que obteve o melhor desempenho em comparação com os demais, alcançando uma acurácia superior a 78%. Esse modelo alcançou tal desempenho com os atributos selecionados pelo método *filter*.

A Figura 5.3 apresentada é um gráfico de dispersão do tipo *beeswarm*, gerado pela ferramenta SHAP, que ilustra a importância dos atributos utilizados no modelo CART. Cada ponto no gráfico representa a influência de um atributo específico na predição do modelo, enquanto a cor indica o valor do atributo (vermelho para valores altos e azul para valores baixos). Podemos observar que os atributos ***dporpag*** e ***bpm*** possuem uma influência significativa, uma vez que apresentam uma distribuição mais dispersa de valores SHAP, indicando que são preditores importantes no modelo. Essa análise reforça a eficácia da seleção de atributos realizada pelo método *filter*, evidenciando os fatores que tendem a influenciar o desempenho do modelo.

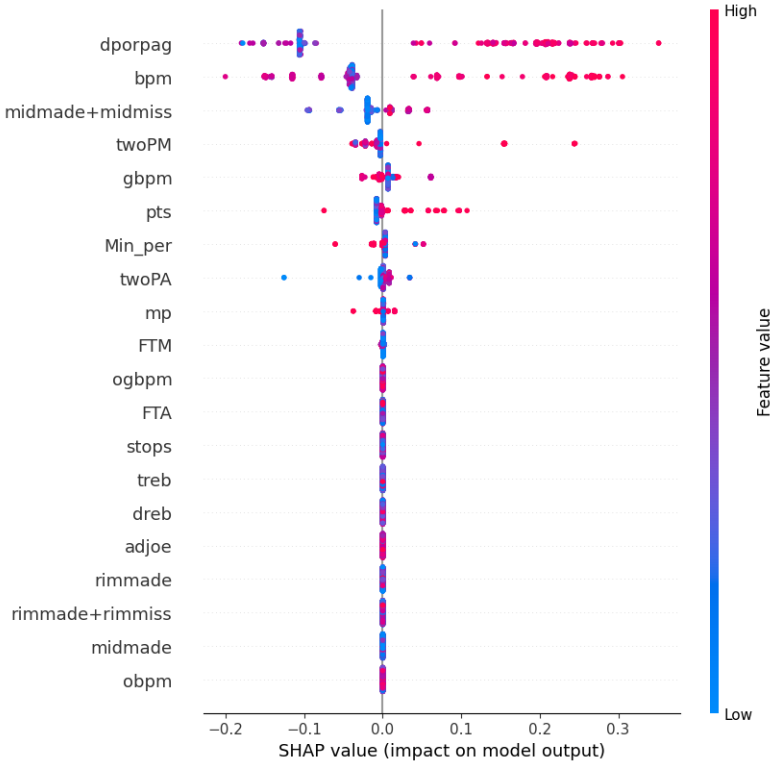

Figura 5.3: Gráfico *Beeswarm* gerado com a ferramenta SHAP, utilizando o modelo CART e os atributos selecionados pelo método *Filter* (Autor, 2024).

5.5.2 Explicabilidade de Modelos - Com atributos selecionados via Algoritmo Genético

O modelo que apresentou o melhor desempenho, alcançando uma acurácia de 80%, foi a Árvore de Decisão do tipo CART. Este resultado foi obtido ao utilizar conjuntos de atributos gerados por meio de um Algoritmo Genético na etapa de seleção. O sucesso desse modelo pode ser atribuído à combinação do SVM com o Algoritmo Genético para a otimização da função de *fitness*.

A Figura 5.4 também mostra um gráfico de dispersão do tipo *beeswarm*. Neste gráfico, observa-se que os atributos com maior influência e dispersão são ***dporparg***, ***FTM*** e ***twoPA***. Em comparação com a Figura 5.3, apenas o atributo *dporparg* aparece em ambos os gráficos. Isso demonstra que, ao utilizar o mesmo classificador com diferentes conjuntos de atributos, os resultados podem variar. Essa variação só pode ser compreendida por meio de ferramentas de explicabilidade.

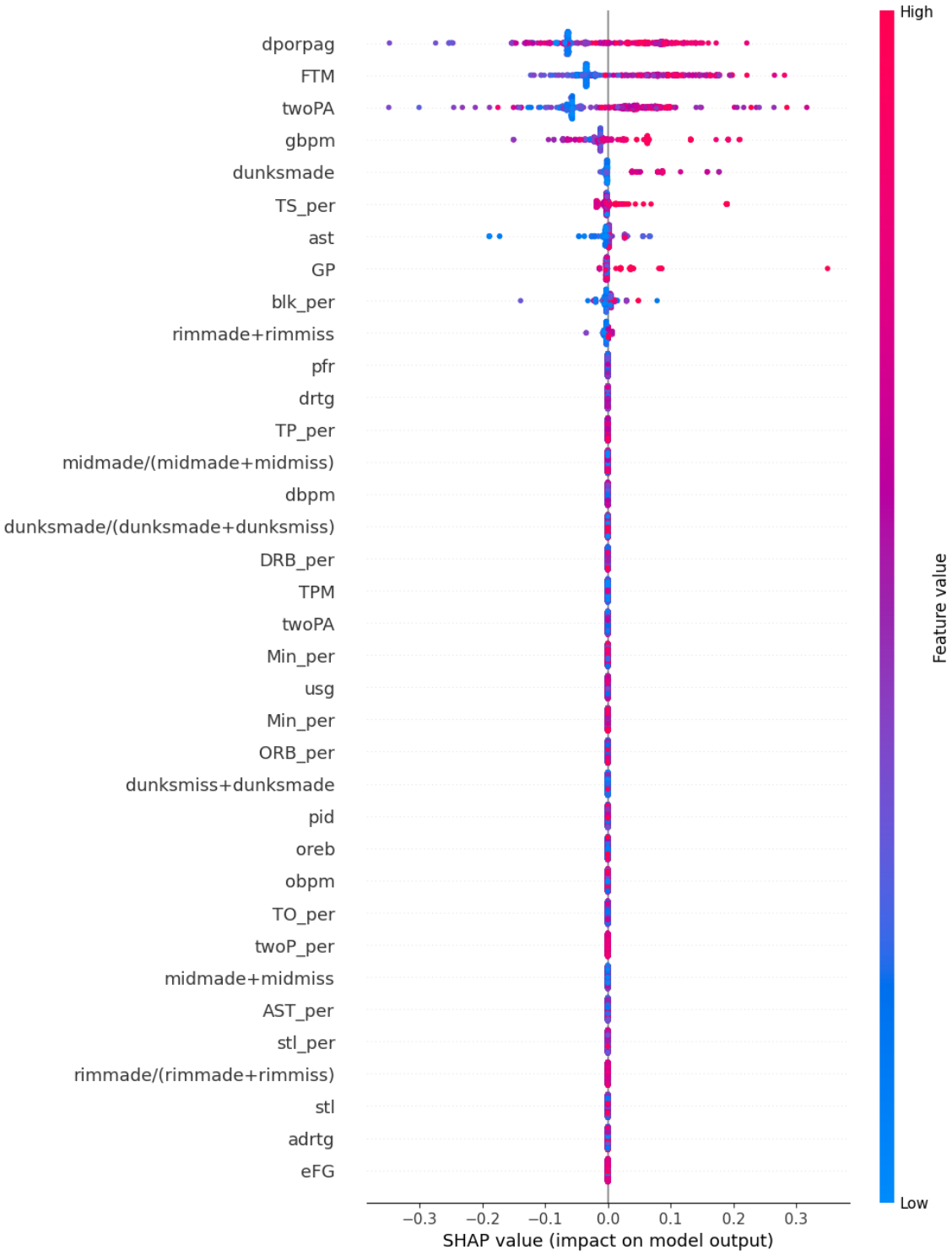

Figura 5.4: Gráfico *Beeswarm* gerado com a ferramenta SHAP, utilizando o modelo CART e os atributos selecionados pelo método *AG - SVM* (Autor, 2024).

5.5.3 Análise Estatística

Para avaliar a significância estatística dos resultados obtidos, foi gerada uma amostra com 30 execuções de cada algoritmo, considerando todas as combinações de atributos discutidas

anteriormente. O objetivo foi verificar se as diferenças observadas entre os métodos de seleção de características e os modelos eram estatisticamente significativas. No entanto, vale destacar que o único algoritmo para o qual não foi possível gerar as 30 amostras foi o C4.5, devido à descontinuidade do suporte da biblioteca utilizada.

Para realizar essa análise, foi executado o teste de Friedman (FRIEDMAN, 1937), um teste não paramétrico que avalia se há diferenças nas medianas das acurácias entre as diferentes combinações de modelos de aprendizagem de máquina e subconjuntos de atributos. Os resultados indicaram diferenças estatisticamente significativas, conforme ilustrado na Figura 5.5. Para uma análise mais detalhada dessas diferenças, foi aplicado o teste post-hoc de Nemenyi (NEMENYI, 1963), que realiza comparações múltiplas entre os pares de métodos, corrigindo o erro tipo I associado às múltiplas comparações, ou seja, controlando o risco de identificar diferenças significativas por acaso.

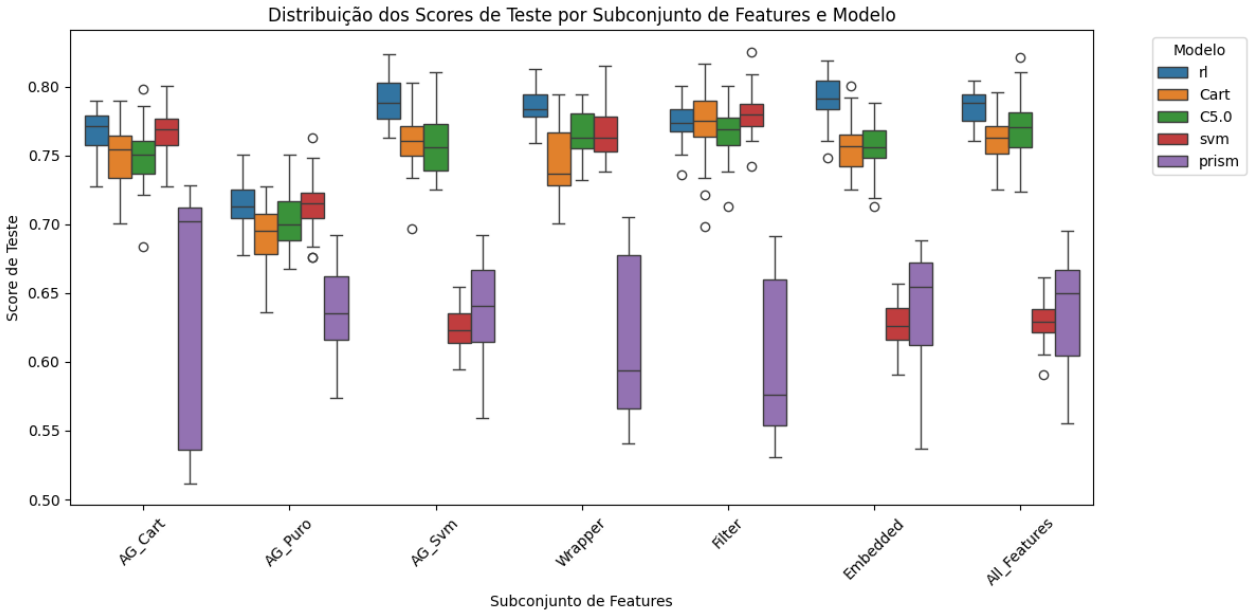

Figura 5.5: Distribuição dos Scores de Teste por Subconjunto de Features e Modelo (Autor, 2024)

O teste de Nemenyi revelou quais combinações apresentaram diferenças significativas entre si, como mostrado na Figura 5.6. Esses resultados forneceram insights valiosos sobre as abordagens que geraram os melhores desempenhos. Por exemplo, observou-se que o subconjunto de atributos gerados pelo método AG_Puro teve um desempenho superior aos outros, enquanto que, entre os subconjuntos de atributos AG_SVM, *Embedded* e *All_Features*, não foram encontradas diferenças significativas.

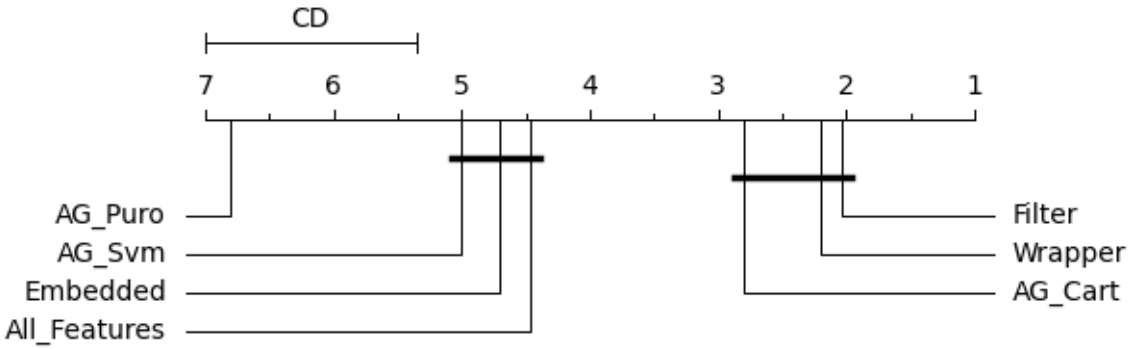

Figura 5.6: Teste de Nemenyi (Autor, 2024)

Esses testes estatísticos confirmaram a existência de variações significativas nos resultados, fornecendo uma base sólida para a discussão sobre as melhores práticas na seleção de características e na aplicação de modelos de aprendizado de máquina, especialmente no contexto do basquete universitário americano.

5.6 Considerações Finais do Capítulo

Neste capítulo, discutimos os resultados obtidos com a aplicação da abordagem proposta, explorando o impacto das diferentes técnicas de seleção de características e o desempenho dos modelos preditivos. A análise permitiu uma compreensão aprofundada de como cada método contribui para a precisão e interpretabilidade das previsões, respondendo às perguntas de pesquisa estabelecidas.

No Capítulo 6, serão apresentadas as conclusões finais do estudo. Esse capítulo oferece uma síntese do trabalho realizado, discute as principais contribuições e limitações da pesquisa, e propõe direções para futuros trabalhos. Essas reflexões fornecem uma visão consolidada dos avanços obtidos e dos caminhos a serem explorados na área.

Capítulo 6

Conclusão

Neste capítulo, estão apresentadas as conclusões desta pesquisa, incluindo uma síntese do que foi realizado e uma sumarização das principais contribuições e limitações, assim como uma sugestão para trabalhos futuros próximos.

6.1 Conclusões da Pesquisa

Neste trabalho, diferentes métodos de seleção de atributos foram executados com o objetivo de otimizar o desempenho dos classificadores. Para tal, utilizamos métodos tradicionais, como *Wrapper*, *Filter* e *Embedded*, bem como uma solução menos convencional adotada pelos pesquisadores: a utilização de Algoritmo Genético. Os resultados demonstraram que o Algoritmo Genético não apenas reduziu o número de atributos necessários, mas também melhorou o desempenho dos classificadores. Observou-se que os modelos que utilizaram os atributos selecionados pelo Algoritmo Genético obtiveram desempenho superior em comparação com os modelos que utilizaram os atributos sem nenhum método de seleção ou com aqueles que empregaram métodos de seleção já consolidados na literatura. Portanto, este trabalho destaca a relevância dessa abordagem para melhorar o desempenho dos modelos de aprendizado de máquina e mitigar o problema da dimensionalidade.

Além disso, conseguimos demonstrar a importância da seleção de características no domínio dos esportes. A partir de um total de 65 variáveis, identificou-se que 5 delas representam adequadamente os dados, resultando em uma redução superior a 90% na dimensionalidade. Essas características são essenciais para prever a tendência de um jogador da liga universitária de basquete para a liga profissional. No contexto dos esportes, em que grandes quantidades

de dados são coletados e analisados, a seleção de características desempenha um papel crucial na eficiência dos modelos preditivos.

Adicionalmente, enfatiza-se a importância da explicabilidade dos modelos de aprendizado de máquina, especialmente em áreas nas quais a transparência e a compreensão das decisões são fundamentais, como medicina, finanças e esportes. Foram implementados modelos que permitem a interpretação dos resultados, utilizando técnicas como PRISM e Árvore de Decisão, que geram regras e estruturas de fácil compreensão.

Outrossim, foram empregadas ferramentas de explicabilidade, como o SHAP, que possibilitaram uma análise detalhada da importância e influência dos atributos selecionados nos modelos preditivos. Os gráficos *beeswarm* gerados pelo SHAP, por exemplo, evidenciaram quais atributos tiveram maior impacto nas previsões, além de mostrar como esses impactos variaram conforme o conjunto de atributos considerado. A inclusão de métodos explicáveis, como PRISM e Árvores de Decisão, juntamente com técnicas de análise, reforça a importância de não apenas buscar alta acurácia, mas também de compreender como e por que os modelos tomam determinadas decisões. Dessa forma, este trabalho contribui para o avanço do conhecimento na intersecção entre seleção de atributos e explicabilidade de modelos, oferecendo *insights* valiosos para a construção de sistemas de aprendizado de máquina mais transparentes e eficazes.

6.2 Limitações e Trabalhos Futuros

Durante o desenvolvimento deste trabalho, enfrentamos algumas dificuldades que impactaram os resultados finais. Uma das principais limitações foi a indisponibilidade do algoritmo C4.5. Como indicado na Tabela 5.5, é possível visualizar a presença de **"XX"** na coluna de resultados, não sendo possível obter os valores esperados, pois a biblioteca correspondente perdeu suporte e não estava mais disponível até a data de conclusão deste estudo. Adicionalmente, encontramos dificuldades na integração da ferramenta SHAP com certas bibliotecas de aprendizado de máquina, uma vez que nem todos os algoritmos possuem suporte completo para essa integração até o presente momento.

Além disso, a ausência de um especialista na área de análise de desempenho esportivo representou um desafio considerável. Essa lacuna dificultou uma avaliação mais detalhada das

variáveis relevantes para a predição de sucesso de jogadores, o que poderia ter agregado maior assertividade na seleção de características e na interpretação dos resultados finais.

Com base nas limitações observadas e nos resultados desta pesquisa, diversas oportunidades para trabalhos futuros podem ser identificadas. Em primeiro lugar, recomenda-se a ampliação do conjunto de dados, incorporando variáveis adicionais ou explorando outras fontes que possam enriquecer a análise e possibilitar uma maior generalização dos resultados. Além disso, sugere-se a extensão da arquitetura proposta a outros domínios, como educação e finanças.

Referências bibliográficas

ALAMAR, B. C. **Sports Analytics: A Guide for Coaches, Managers, and Other Decision Makers**. [S.l.]: Columbia University Press, 2013.

ALBERT, A. A.; MINGO LÓPEZ, L. F. de; ALLBRIGHT, K.; GÓMEZ BLAS, N. A Hybrid Machine Learning Model for Predicting USA NBA All-Stars. **Electronics**, MDPI, v. 11, n. 1, p. 97, 2021.

ANDERSON, J. R. **Machine learning: An artificial intelligence approach**. [S.l.]: Morgan Kaufmann, 1983. v. 3.

ARIA, M.; CUCCURULLO, C. bibliometrix: An R-tool for comprehensive science mapping analysis. **Journal of Informetrics**, Elsevier, v. 11, n. 4, p. 959–975, 2017. Disponível em: <https://doi.org/10.1016/j.joi.2017.08.007>.

ARIA, M.; CUCCURULLO, C. bibliometrix: An R-tool for comprehensive science mapping analysis. **Journal of informetrics**, Elsevier, v. 11, n. 4, p. 959–975, 2017.

ARRIETA, A. B. et al. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. **Information fusion**, Elsevier, v. 58, p. 82–115, 2020.

BEASLEY, D.; BULL, D. R.; MARTIN, R. R. An overview of genetic algorithms: Part 1, fundamentals. **University computing**, v. 15, n. 2, p. 56–69, 1993.

BLUM, A. L.; LANGLEY, P. Selection of relevant features and examples in machine learning. **Artificial intelligence**, Elsevier, v. 97, n. 1-2, p. 245–271, 1997.

BORGHESI, R. The financial and competitive value of NCAA basketball recruits. **Journal of Sports Economics**, Sage Publications Sage CA: Los Angeles, CA, v. 19, n. 1, p. 31–49, 2018.

CALIWAG, J. A.; ARAGON, M. C. R.; CASTILLO, R. E.; COLANTES, E. M. S. Predicting Basketball Results Using Cascading Algorithm. In: PROCEEDINGS of the 1st International Conference on Information Science and Systems. [S.l.: s.n.], 2018. P. 64–68.

CENDROWSKA, J. PRISM: An algorithm for inducing modular rules. **International Journal of Man-Machine Studies**, Elsevier, v. 27, n. 4, p. 349–370, 1987.

CLACY, A. et al. A knock to the system: A new sociotechnical systems approach to sport-related concussion. **Journal of sports sciences**, Taylor & Francis, v. 35, n. 22, p. 2232–2239, 2017.

DEVROYE, L.; WAGNER, T. Distribution-free performance bounds for potential function rules. **IEEE Transactions on Information Theory**, IEEE, v. 25, n. 5, p. 601–604, 1979.

EIBEN, A. E.; SMITH, J. E. **Introduction to evolutionary computing**. [S.l.]: Springer, 2015.

ELKAN, C. **Predictive analytics and data mining**. [S.l.]: University of California San Diego, 2013.

ESSEGHIR, M. A. Effective wrapper-filter hybridization through grasp schemata. In: PMLR. FEATURE selection in data mining. [S.l.: s.n.], 2010. P. 45–54.

FRIEDMAN, M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. **Journal of the american statistical association**, Taylor & Francis, v. 32, n. 200, p. 675–701, 1937.

GOLDBERG, D.; HOLLAND, J. **Genetic algorithms and machine learning. 3 (2): 95-99**. [S.l.]: Kluwer Academic Publishers-Plenum Publishers, 1988.

GÜLER, E. **Imbalanced learning techniques: Experiments on NCAA College Basketball League player statistics dataset**. 2022. Diss. (Mestrado) – Middle East Technical University.

GUMM, J.; BARRETT, A.; HU, G. A machine learning strategy for predicting march madness winners. In: 2015 IEEE/ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). [S.l.: s.n.], 2015. P. 1–6. DOI: 10.1109/SNPD.2015.7176206.

GUO, Y.; CHUNG, F.-L.; LI, G.; ZHANG, L. Multi-label bioinformatics data classification with ensemble embedded feature selection. **IEEE access**, IEEE, v. 7, p. 103863–103875, 2019.

GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. **Journal of machine learning research**, v. 3, Mar, p. 1157–1182, 2003.

GUYON, I.; GUNN, S.; NIKRAVESH, M.; ZADEH, L. A. **Feature extraction: foundations and applications**. [S.l.]: Springer, 2008. v. 207.

HASSANAT, A. et al. Choosing mutation and crossover ratios for genetic algorithms—a review with a new dynamic approach. **Information**, MDPI, v. 10, n. 12, p. 390, 2019.

HOLLAND, J. H. **Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence**. [S.l.]: MIT press, 1992.

HSU, P.-H.; GALSANBADAM, S.; YANG, J.-S.; YANG, C.-Y. Evaluating machine learning varieties for NBA players' winning contribution. In: IEEE. 2018 International Conference on System Science and Engineering (ICSSE). [S.l.: s.n.], 2018. P. 1–6.

JAIN, S.; KAUR, H. Machine learning approaches to predict basketball game outcome. In: IEEE. 2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA)(Fall). [S.l.: s.n.], 2017. P. 1–7.

KATO, R.; PAIVA, V.; IZIDORO, S. Algoritmos Genéticos. In: BIOINFO - Revista Brasileira de Bioinformática e Biologia Computacional. [S.l.]: Alfahelix, jul. 2021.

KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. **Artificial intelligence**, Elsevier, v. 97, n. 1-2, p. 273–324, 1997.

KOREJO, I.; YANG, S.; BROHI, K.; KHUHRO, Z. Multi-population methods with adaptive mutation for multi-modal optimization problems. AirCC Publishing, 2013.

KRAMER, O.; KRAMER, O. Scikit-learn. **Machine learning for evolution strategies**, Springer, p. 45–53, 2016.

KUBATKO, J.; OLIVER, D.; PELTON, K.; ROSENBAUM, D. T. **Journal of Quantitative Analysis in Sports**, v. 3, n. 3, 2007. DOI: doi : 10 . 2202 / 1559 - 0410 . 1070. Disponível em: <https://doi.org/10.2202/1559-0410.1070>.

LEE RODGERS, J.; NICEWANDER, W. A. Thirteen ways to look at the correlation coefficient. **The American Statistician**, Taylor & Francis, v. 42, n. 1, p. 59–66, 1988.

LINDE, K.; WILLICH, S. N. How objective are systematic reviews? Differences between reviews on complementary medicine. **Journal of the royal society of medicine**, SAGE Publications Sage UK: London, England, v. 96, n. 1, p. 17–22, 2003.

LIU, H.; COCEA, M.; DING, W. Multi-task learning for intelligent data processing in granular computing context. **Granular Computing**, Springer, v. 3, p. 257–273, 2018.

LIU, H.; MOTODA, H. **Computational methods of feature selection**. [S.l.]: CRC press, 2007.

LUNDBERG, S. Welcome to the SHAP documentation. **Welcome to the SHAP Documentation-SHAP latest documentation. URL: https://shaplrjball. readthedocs. io/(Accessed: 25/06/2023)**, 2018.

LUNDBERG, S. M.; LEE, S.-I. A Unified Approach to Interpreting Model Predictions. In: GUYON, I. et al. (Ed.). **Advances in Neural Information Processing Systems 30**. [S.l.]: Curran Associates, Inc., 2017. P. 4765–4774. Disponível em: <http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf>.

LUNDBERG, S. M.; LEE, S.-I. A unified approach to interpreting model predictions. **Advances in neural information processing systems**, v. 30, 2017.

MAHMOOD, Z.; DAUD, A.; ABBASI, R. A. Using machine learning techniques for rising star prediction in basketball. **Knowledge-Based Systems**, Elsevier, v. 211, p. 106506, 2021.

MELLO, H. **NCAA March Madness: Tudo o que Precisa Saber sobre o Basquete Universitário**. [S.l.], 2024.

MIRJALILI, S.; SONG DONG, J.; SADIQ, A. S.; FARIS, H. Genetic algorithm: Theory, literature review, and application in image reconstruction. **Nature-inspired optimizers: Theories, literature reviews and applications**, Springer, p. 69–85, 2020.

MISHRA, P. et al. Application of student's t-test, analysis of variance, and covariance. **Annals of cardiac anaesthesia**, Wolters Kluwer–Medknow Publications, v. 22, n. 4, p. 407, 2019.

MITCHELL, T. M.; MITCHELL, T. M. **Machine learning**. [S.l.]: McGraw-hill New York, 1997. v. 1.

MOHER, D. et al. Preferred reporting items for systematic review and meta-analysis protocols (PRISMA-P) 2015 statement. **Systematic reviews**, BioMed Central, v. 4, n. 1, p. 1–9, 2015.

NEMENYI, P. B. **Distribution-free multiple comparisons.** [S.l.]: Princeton University, 1963.

NGUYEN, N. H.; NGUYEN, D. T. A.; MA, B.; HU, J. The application of machine learning and deep learning in sport: predicting NBA players' performance and popularity. **Journal of Information and Telecommunication**, Taylor & Francis, v. 6, n. 2, p. 217–235, 2022.

OKSER, S.; PAHIKKALA, T.; AITTOKALLIO, T. Genetic variants and their interactions in disease risk prediction–machine learning and network perspectives. **BioData mining**, Springer, v. 6, p. 1–16, 2013.

OUGHALI, M. S.; BAHLOUL, M.; EL RAHMAN, S. A. Analysis of NBA players and shot prediction using random forest and XGBoost models. In: IEEE. 2019 international conference on computer and information sciences (ICCIS). [S.l.: s.n.], 2019. P. 1–5.

OZKAN, I. A. A novel basketball result prediction model using a concurrent neuro-fuzzy system. **Applied Artificial Intelligence**, Taylor & Francis, v. 34, n. 13, p. 1038–1054, 2020.

PANDYA, R.; PANDYA, J. C5. 0 algorithm to improved decision tree with feature selection and reduced error pruning. **International Journal of Computer Applications**, Foundation of Computer Science, v. 117, n. 16, p. 18–21, 2015.

PENG, C.-Y. J.; LEE, K. L.; INGERSOLL, G. M. An introduction to logistic regression analysis and reporting. **The journal of educational research**, Taylor & Francis, v. 96, n. 1, p. 3–14, 2002.

PENG, C.-Y. J.; SO, T.-S. H. Logistic regression analysis and reporting: A primer. **Understanding Statistics: Statistical Issues in Psychology, Education, and the Social Sciences**, Taylor & Francis, v. 1, n. 1, p. 31–70, 2002.

PENG, C.-Y. J.; SO, T.-S. H.; STAGE, F. K.; ST. JOHN, E. P. The use and interpretation of logistic regression in higher education journals: 1988–1999. **Research in higher education**, Springer, v. 43, p. 259–293, 2002.

PUDJIHARTONO, N.; FADASON, T.; KEMPA-LIEHR, A. W.; O'SULLIVAN, J. M. A review of feature selection methods for machine learning-based disease risk prediction. **Frontiers in Bioinformatics**, Frontiers Media SA, v. 2, p. 927312, 2022.

QUINLAN, J. R. **C4. 5: programs for machine learning**. [S.l.]: Elsevier, 2014.

SAFE, M.; CARBALLIDO, J.; PONZONI, I.; BRIGNOLE, N. On stopping criteria for genetic algorithms. In: SPRINGER. ADVANCES in Artificial Intelligence–SBIA 2004: 17th Brazilian Symposium on Artificial Intelligence, Sao Luis, Maranhao, Brazil, September 29-Ocotber 1, 2004. Proceedings 17. [S.l.: s.n.], 2004. P. 405–413.

SARLIS, V.; TJORTJIS, C. Sports analytics—Evaluation of basketball players and team performance. **Information Systems**, Elsevier, v. 93, p. 101562, 2020.

SCHLIERKAMP-VOOSEN, D. Optimal interaction of mutation and crossover in the breeder genetic algorithm. In: INTERNATIONAL Conference on Genetic Algorithms; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA. [S.l.: s.n.], 1993.

SCHOBER, P.; BOER, C.; SCHWARTE, L. A. Correlation coefficients: appropriate use and interpretation. **Anesthesia & analgesia**, LWW, v. 126, n. 5, p. 1763–1768, 2018.

SERENGIL, S. I. **ChefBoost: A Lightweight Boosted Decision Tree Framework**. [S.l.]: Zenodo, out. 2021. https://doi.org/10.5281/zenodo.5576203. DOI: 10.5281/zenodo.5576203.

SINGH, S.; GUPTA, P. Comparative study ID3, cart and C4. 5 decision tree algorithm: a survey. **International Journal of Advanced Information Science and Technology (IJAIST)**, Citeseer, v. 27, n. 27, p. 97–103, 2014.

SOLIMAN, G.; MISBAH, A.; ELDAWLATLY, S. et al. Predicting all star player in the national basketball association using random forest. In: IEEE. 2017 Intelligent Systems Conference (IntelliSys). [S.l.: s.n.], 2017. P. 706–713.

SOMOL, P.; PUDIL, P.; NOVOVIČOVÁ, J.; PACLIK, P. Adaptive floating search methods in feature selection. **Pattern recognition letters**, Elsevier, v. 20, n. 11-13, p. 1157–1163, 1999.

SOMVANSHI, M.; CHAVAN, P.; TAMBADE, S.; SHINDE, S. A review of machine learning techniques using decision tree and support vector machine. In: IEEE. 2016 international conference on computing communication control and automation (ICCUBEA). [S.l.: s.n.], 2016. P. 1–7.

TICHY, W. Changing the Game: Dr. Dave Schrader on sports analytics. **Ubiquity**, ACM New York, NY, USA, v. 2016, May, p. 1–10, 2016.

VAN ECK, N. J.; WALTMAN, L. Visualizing bibliometric networks. In: MEASURING scholarly impact: Methods and practice. [S.l.]: Springer, 2014. P. 285–320.

VAPNIK, V. **The nature of statistical learning theory**. [S.l.]: Springer science & business media, 2013.

WEISBERG, S. **Applied linear regression**. [S.l.]: John Wiley & Sons, 2005. v. 528.

ZHANG, K.; XU, P.; ZHANG, J. Explainable AI in deep reinforcement learning models: A shap method applied in power system emergency control. In: IEEE. 2020 IEEE 4th conference on energy internet and energy system integration (EI2). [S.l.: s.n.], 2020. P. 711–716.

Apêndice A

Primeiro Apêndice

Resultado GridSearch


```
[
    {
        "Model": "rl",
        "Features": [
            "dbpm",
            "adrtg",
            "ogbpm",
            "TPM",
            "stops",
            "treb"
        ],
        "Best Parameters": {
            "max_iter": 500,
            "penalty": "l2",
            "solver": "lbfgs"
        },
        "Features SubSet Name": "GA_CART",
        "Best CV Score": 0.9906594197370187,
        "Test Score": 0.9906980319803198
    },
    {
        "Model": "rl",
        "Features": [
            "TS_per",
            "ORB_per",
            "DRB_per",
            "FTA",
            "FT_per"
        ],
        "Best Parameters": {
            "max_iter": 200,
            "penalty": "l2",
            "solver": "lbfgs"
        },
        "Features SubSet Name": "GA_Entropy",
        "Best CV Score": 0.9906594141966089,
        "Test Score": 0.9910055350553506
    },
    {
        "Model": "rl",
        "Features": [
            "eFG",
            "Min_per",
            "twoPA",
```

```
            "TPM",
            "DRB_per",
            "TS_per",
            "dunksmade",
            "dbpm",
            "blk_per",
            "pfr",
            "twoPA",
            "TP_per",
            "drtg",
            "usg",
            "dunksmade/(dunksmade+dunksmiss)",
            "ORB_per",
            "dunksmiss+dunksmade",
            "Min_per",
            "gbpm",
            "FTM",
            "ast",
            "oreb",
            "obpm",
            "TO_per",
            "twoP_per",
            "midmade+midmiss",
            "dporpag",
            "AST_per",
            "stl_per",
            "rimmade/(rimmade+rimmiss)",
            "stl",
            "GP",
            "adrtg",
            "rimmade+rimmiss",
            "pid",
            "midmade/(midmade+midmiss)"
        ],
        "Best Parameters": {
            "max_iter": 100,
            "penalty": "l2",
            "solver": "newton-cholesky"
        },
        "Features SubSet Name": "GA_SVM",
        "Best CV Score": 0.9913705368514713,
        "Test Score": 0.9920049200492005
    },
    {
        "Model": "rl",
        "Features": [
```

```
            "DRB_per",
            "FT_per",
            "twoP_per",
            "TP_per",
            "stl_per",
            "pfr",
            "ast/tov",
            "dporpag",
            "obpm",
            "dbpm",
            "gbpm",
            "mp",
            "ogbpm",
            "dreb",
            "treb",
            "stl"
        ],
        "Best Parameters": {
            "max_iter": 100,
            "penalty": "l1",
            "solver": "liblinear"
        },
        "Features SubSet Name": "Wrapper",
        "Best CV Score": 0.9908131716442353,
        "Test Score": 0.9911592865928659
    },
    {
        "Model": "rl",
        "Features": [
            "dporpag",
            "stops",
            "twoPM",
            "twoPA",
            "pts",
            "FTM",
            "FTA",
            "dreb",
            "treb",
            "bpm",
            "mp",
            "Min_per",
            "rimmade",
            "midmade+midmiss",
            "rimmade+rimmiss",
            "midmade",
            "gbpm",
```

```
            "ogbpm",
            "adjoe",
            "obpm"
        ],
        "Best Parameters": {
            "max_iter": 100,
            "penalty": "l2",
            "solver": "newton-cholesky"
        },
        "Features SubSet Name": "Filter",
        "Best CV Score": 0.9910053610665555,
        "Test Score": 0.9919280442804428
    },
    {
        "Model": "rl",
        "Features": [
            "dporpag",
            "gbpm",
            "adjoe",
            "midmade/(midmade+midmiss)",
            "adrtg",
            "AST_per",
            "ast/tov",
            "rimmade+rimmiss",
            "pid",
            "ORB_per",
            "dunksmade",
            "ast",
            "stl_per",
            "pts",
            "midmade",
            "stl",
            "FTA",
            "TP_per",
            "oreb",
            "Ortg",
            "GP",
            "twoPA",
            "dunksmiss+dunksmade",
            "rimmade/(rimmade+rimmiss)",
            "dgbpm",
            "Min_per",
            "TPA",
            "twoPM",
            "usg",
            "FTM",
```

```
            "mp",
            "treb",
            "stops",
            "twoP_per",
            "drtg",
            "blk",
            "obpm",
            "dbpm",
            "eFG",
            "DRB_per",
            "pfr",
            "TO_per",
            "bpm",
            "TS_per",
            "rimmade",
            "ogbpm"
        ],
        "Best Parameters": {
            "max_iter": 100,
            "penalty": "l2",
            "solver": "newton-cholesky"
        },
        "Features SubSet Name": "Embedded",
        "Best CV Score": 0.9914858530903878,
        "Test Score": 0.9919280442804428
    },
        {
        "Model": "rl",
        "Features": [
            "GP",
            "Min_per",
            "Ortg",
            "usg",
            "eFG",
            "TS_per",
            "ORB_per",
            "DRB_per",
            "AST_per",
            "TO_per",
            "FTM",
            "FTA",
            "FT_per",
            "twoPM",
            "twoPA",
            "twoP_per",
            "TPM",
```

```
            "TPA",
            "TP_per",
            "blk_per",
            "stl_per",
            "ftr",
            "adjoe",
            "pfr",
            "pid",
            "ast/tov",
            "rimmade",
            "rimmade+rimmiss",
            "midmade",
            "midmade+midmiss",
            "rimmade/(rimmade+rimmiss)",
            "midmade/(midmade+midmiss)",
            "dunksmade",
            "dunksmiss+dunksmade",
            "dunksmade/(dunksmade+dunksmiss)",
            "drtg",
            "adrtg",
            "dporpag",
            "stops",
            "bpm",
            "obpm",
            "dbpm",
            "gbpm",
            "mp",
            "ogbpm",
            "dgbpm",
            "oreb",
            "dreb",
            "treb",
            "ast",
            "stl",
            "blk",
            "pts"
        ],
        "Best Parameters": {
            "max_iter": 100,
            "penalty": "l1",
            "solver": "liblinear"
        },
        "Features SubSet Name": "All_Features",
        "Best CV Score": 0.9914089725197714,
        "Test Score": 0.9916974169741697
    },
```

```
{
    "Model": "Cart",
    "Features": [
        "dbpm",
        "adrtg",
        "ogbpm",
        "TPM",
        "stops",
        "treb"
    ],
    "Best Parameters": {
        "max_depth": 5,
        "max_features": null,
        "min_samples_leaf": 1,
        "min_samples_split": 2,
        "splitter": "random"
    },
    "Features SubSet Name": "GA_CART",
    "Best CV Score": 0.9908131790314482,
    "Test Score": 0.9910824108241082
},
{
    "Model": "Cart",
    "Features": [
        "TS_per",
        "ORB_per",
        "DRB_per",
        "FTA",
        "FT_per"
    ],
    "Best Parameters": {
        "max_depth": 3,
        "max_features": null,
        "min_samples_leaf": 1,
        "min_samples_split": 2,
        "splitter": "random"
    },
    "Features SubSet Name": "GA_Entropy",
    "Best CV Score": 0.9906786338775623,
    "Test Score": 0.9910055350553506
},
{
    "Model": "Cart",
    "Features": [
        "eFG",
        "Min_per",
```

```
            "twoPA",
            "TPM",
            "DRB_per",
            "TS_per",
            "dunksmade",
            "dbpm",
            "blk_per",
            "pfr",
            "twoPA",
            "TP_per",
            "drtg",
            "usg",
            "dunksmade/(dunksmade+dunksmiss)",
            "ORB_per",
            "dunksmiss+dunksmade",
            "Min_per",
            "gbpm",
            "FTM",
            "ast",
            "oreb",
            "obpm",
            "TO_per",
            "twoP_per",
            "midmade+midmiss",
            "dporpag",
            "AST_per",
            "stl_per",
            "rimmade/(rimmade+rimmiss)",
            "stl",
            "GP",
            "adrtg",
            "rimmade+rimmiss",
            "pid",
            "midmade/(midmade+midmiss)"
        ],
        "Best Parameters": {
            "max_depth": 5,
            "max_features": null,
            "min_samples_leaf": 1,
            "min_samples_split": 2,
            "splitter": "random"
        },
        "Features SubSet Name": "GA_SVM",
        "Best CV Score": 0.9906594160434121,
        "Test Score": 0.9919280442804428
    },
```

```
{
    "Model": "Cart",
    "Features": [
        "DRB_per",
        "FT_per",
        "twoP_per",
        "TP_per",
        "stl_per",
        "pfr",
        "ast/tov",
        "dporpag",
        "obpm",
        "dbpm",
        "gbpm",
        "mp",
        "ogbpm",
        "dreb",
        "treb",
        "stl"
    ],
    "Best Parameters": {
        "max_depth": 3,
        "max_features": null,
        "min_samples_leaf": 1,
        "min_samples_split": 2,
        "splitter": "random"
    },
    "Features SubSet Name": "Wrapper",
    "Best CV Score": 0.9907362947672252,
    "Test Score": 0.9909286592865929
},
{
    "Model": "Cart",
    "Features": [
        "dporpag",
        "stops",
        "twoPM",
        "twoPA",
        "pts",
        "FTM",
        "FTA",
        "dreb",
        "treb",
        "bpm",
        "mp",
        "Min_per",
```

```
            "rimmade",
            "midmade+midmiss",
            "rimmade+rimmiss",
            "midmade",
            "gbpm",
            "ogbpm",
            "adjoe",
            "obpm"
        ],
        "Best Parameters": {
            "max_depth": 4,
            "max_features": null,
            "min_samples_leaf": 2,
            "min_samples_split": 2,
            "splitter": "best"
        },
        "Features SubSet Name": "Filter",
        "Best CV Score": 0.9906017625409618,
        "Test Score": 0.9906211562115621
    },
    {
        "Model": "Cart",
        "Features": [
            "dporpag",
            "gbpm",
            "adjoe",
            "midmade/(midmade+midmiss)",
            "adrtg",
            "AST_per",
            "ast/tov",
            "rimmade+rimmiss",
            "pid",
            "ORB_per",
            "dunksmade",
            "ast",
            "stl_per",
            "pts",
            "midmade",
            "stl",
            "FTA",
            "TP_per",
            "oreb",
            "Ortg",
            "GP",
            "twoPA",
            "dunksmiss+dunksmade",
```

```
            "rimmade/(rimmade+rimmiss)",
            "dgbpm",
            "Min_per",
            "TPA",
            "twoPM",
            "usg",
            "FTM",
            "mp",
            "treb",
            "stops",
            "twoP_per",
            "drtg",
            "blk",
            "obpm",
            "dbpm",
            "eFG",
            "DRB_per",
            "pfr",
            "TO_per",
            "bpm",
            "TS_per",
            "rimmade",
            "ogbpm"
        ],
        "Best Parameters": {
            "max_depth": 3,
            "max_features": null,
            "min_samples_leaf": 1,
            "min_samples_split": 2,
            "splitter": "random"
        },
        "Features SubSet Name": "Embedded",
        "Best CV Score": 0.9907555126013754,
        "Test Score": 0.9904674046740467
    },
        {
        "Model": "Cart",
        "Features": [
            "GP",
            "Min_per",
            "Ortg",
            "usg",
            "eFG",
            "TS_per",
            "ORB_per",
            "DRB_per",
```

```
        "AST_per",
        "TO_per",
        "FTM",
        "FTA",
        "FT_per",
        "twoPM",
        "twoPA",
        "twoP_per",
        "TPM",
        "TPA",
        "TP_per",
        "blk_per",
        "stl_per",
        "ftr",
        "adjoe",
        "pfr",
        "pid",
        "ast/tov",
        "rimmade",
        "rimmade+rimmiss",
        "midmade",
        "midmade+midmiss",
        "rimmade/(rimmade+rimmiss)",
        "midmade/(midmade+midmiss)",
        "dunksmade",
        "dunksmiss+dunksmade",
        "dunksmade/(dunksmade+dunksmiss)",
        "drtg",
        "adrtg",
        "dporpag",
        "stops",
        "bpm",
        "obpm",
        "dbpm",
        "gbpm",
        "mp",
        "ogbpm",
        "dgbpm",
        "oreb",
        "dreb",
        "treb",
        "ast",
        "stl",
        "blk",
        "pts"
    ],
```

```
    "Best Parameters": {
        "max_depth": 4,
        "max_features": null,
        "min_samples_leaf": 2,
        "min_samples_split": 2,
        "splitter": "random"
    },
    "Features SubSet Name": "All_Features",
    "Best CV Score": 0.9906786357243653,
    "Test Score": 0.991389913899139
},
{
    "Model": "C5.0",
    "Features": [
        "dbpm",
        "adrtg",
        "ogbpm",
        "TPM",
        "stops",
        "treb"
    ],
    "Best Parameters": {
        "max_depth": 5,
        "max_features": null,
        "min_samples_leaf": 1,
        "min_samples_split": 4,
        "splitter": "random"
    },
    "Features SubSet Name": "GA_CART",
    "Best CV Score": 0.9908131716442352,
    "Test Score": 0.9910824108241082
},
{
    "Model": "C5.0",
    "Features": [
        "TS_per",
        "ORB_per",
        "DRB_per",
        "FTA",
        "FT_per"
    ],
    "Best Parameters": {
        "max_depth": 3,
        "max_features": null,
        "min_samples_leaf": 1,
        "min_samples_split": 2,
```

```
            "splitter": "best"
        },
        "Features SubSet Name": "GA_Entropy",
        "Best CV Score": 0.9906786338775623,
        "Test Score": 0.9910055350553506
    },
    {
        "Model": "C5.0",
        "Features": [
            "eFG",
            "Min_per",
            "twoPA",
            "TPM",
            "DRB_per",
            "TS_per",
            "dunksmade",
            "dbpm",
            "blk_per",
            "pfr",
            "twoPA",
            "TP_per",
            "drtg",
            "usg",
            "dunksmade/(dunksmade+dunksmiss)",
            "ORB_per",
            "dunksmiss+dunksmade",
            "Min_per",
            "gbpm",
            "FTM",
            "ast",
            "oreb",
            "obpm",
            "TO_per",
            "twoP_per",
            "midmade+midmiss",
            "dporpag",
            "AST_per",
            "stl_per",
            "rimmade/(rimmade+rimmiss)",
            "stl",
            "GP",
            "adrtg",
            "rimmade+rimmiss",
            "pid",
            "midmade/(midmade+midmiss)"
        ],
```

```
    "Best Parameters": {
        "max_depth": 5,
        "max_features": null,
        "min_samples_leaf": 2,
        "min_samples_split": 2,
        "splitter": "random"
    },
    "Features SubSet Name": "GA_SVM",
    "Best CV Score": 0.9907555126013754,
    "Test Score": 0.9910055350553506
},
{
    "Model": "C5.0",
    "Features": [
        "DRB_per",
        "FT_per",
        "twoP_per",
        "TP_per",
        "stl_per",
        "pfr",
        "ast/tov",
        "dporpag",
        "obpm",
        "dbpm",
        "gbpm",
        "mp",
        "ogbpm",
        "dreb",
        "treb",
        "stl"
    ],
    "Best Parameters": {
        "max_depth": 3,
        "max_features": null,
        "min_samples_leaf": 1,
        "min_samples_split": 2,
        "splitter": "best"
    },
    "Features SubSet Name": "Wrapper",
    "Best CV Score": 0.9906786338775623,
    "Test Score": 0.9910055350553506
},
{
    "Model": "C5.0",
    "Features": [
        "dporpag",
```

```
            "stops",
            "twoPM",
            "twoPA",
            "pts",
            "FTM",
            "FTA",
            "dreb",
            "treb",
            "bpm",
            "mp",
            "Min_per",
            "rimmade",
            "midmade+midmiss",
            "rimmade+rimmiss",
            "midmade",
            "gbpm",
            "ogbpm",
            "adjoe",
            "obpm"
        ],
        "Best Parameters": {
            "max_depth": 5,
            "max_features": null,
            "min_samples_leaf": 2,
            "min_samples_split": 2,
            "splitter": "random"
        },
        "Features SubSet Name": "Filter",
        "Best CV Score": 0.9907362855332092,
        "Test Score": 0.9910055350553506
    },
    {
        "Model": "C5.0",
        "Features": [
            "dporpag",
            "gbpm",
            "adjoe",
            "midmade/(midmade+midmiss)",
            "adrtg",
            "AST_per",
            "ast/tov",
            "rimmade+rimmiss",
            "pid",
            "ORB_per",
            "dunksmade",
            "ast",
```

```
            "stl_per",
            "pts",
            "midmade",
            "stl",
            "FTA",
            "TP_per",
            "oreb",
            "Ortg",
            "GP",
            "twoPA",
            "dunksmiss+dunksmade",
            "rimmade/(rimmade+rimmiss)",
            "dgbpm",
            "Min_per",
            "TPA",
            "twoPM",
            "usg",
            "FTM",
            "mp",
            "treb",
            "stops",
            "twoP_per",
            "drtg",
            "blk",
            "obpm",
            "dbpm",
            "eFG",
            "DRB_per",
            "pfr",
            "TO_per",
            "bpm",
            "TS_per",
            "rimmade",
            "ogbpm"
        ],
        "Best Parameters": {
            "max_depth": 3,
            "max_features": null,
            "min_samples_leaf": 1,
            "min_samples_split": 2,
            "splitter": "random"
        },
        "Features SubSet Name": "Embedded",
        "Best CV Score": 0.9906786338775623,
        "Test Score": 0.9910055350553506
    },
```

```
{
 "Model": "C5.0",
 "Features": [
     "GP",
     "Min_per",
     "Ortg",
     "usg",
     "eFG",
     "TS_per",
     "ORB_per",
     "DRB_per",
     "AST_per",
     "TO_per",
     "FTM",
     "FTA",
     "FT_per",
     "twoPM",
     "twoPA",
     "twoP_per",
     "TPM",
     "TPA",
     "TP_per",
     "blk_per",
     "stl_per",
     "ftr",
     "adjoe",
     "pfr",
     "pid",
     "ast/tov",
     "rimmade",
     "rimmade+rimmiss",
     "midmade",
     "midmade+midmiss",
     "rimmade/(rimmade+rimmiss)",
     "midmade/(midmade+midmiss)",
     "dunksmade",
     "dunksmiss+dunksmade",
     "dunksmade/(dunksmade+dunksmiss)",
     "drtg",
     "adrtg",
     "dporpag",
     "stops",
     "bpm",
     "obpm",
     "dbpm",
     "gbpm",
```

```
            "mp",
            "ogbpm",
            "dgbpm",
            "oreb",
            "dreb",
            "treb",
            "ast",
            "stl",
            "blk",
            "pts"
        ],
        "Best Parameters": {
            "max_depth": 4,
            "max_features": null,
            "min_samples_leaf": 1,
            "min_samples_split": 2,
            "splitter": "best"
        },
        "Features SubSet Name": "All_Features",
        "Best CV Score": 0.9906209859155215,
        "Test Score": 0.9908517835178352
    },
    {
        "Model": "svm",
        "Features": [
            "dbpm",
            "adrtg",
            "ogbpm",
            "TPM",
            "stops",
            "treb"
        ],
        "Best Parameters": {
            "C": 0.1,
            "gamma": "auto",
            "kernel": "linear"
        },
        "Features SubSet Name": "GA_CART",
        "Best CV Score": 0.9906786338775623,
        "Test Score": 0.9910055350553506
    },
    {
        "Model": "svm",
        "Features": [
            "TS_per",
            "ORB_per",
```

```
            "DRB_per",
            "FTA",
            "FT_per"
        ],
        "Best Parameters": {
            "C": 0.1,
            "gamma": "auto",
            "kernel": "linear"
        },
        "Features SubSet Name": "GA_Entropy",
        "Best CV Score": 0.9906786338775623,
        "Test Score": 0.9910055350553506
    },
    {
        "Model": "svm",
        "Features": [
            "eFG",
            "Min_per",
            "twoPA",
            "TPM",
            "DRB_per",
            "TS_per",
            "dunksmade",
            "dbpm",
            "blk_per",
            "pfr",
            "twoPA",
            "TP_per",
            "drtg",
            "usg",
            "dunksmade/(dunksmade+dunksmiss)",
            "ORB_per",
            "dunksmiss+dunksmade",
            "Min_per",
            "gbpm",
            "FTM",
            "ast",
            "oreb",
            "obpm",
            "TO_per",
            "twoP_per",
            "midmade+midmiss",
            "dporpag",
            "AST_per",
            "stl_per",
            "rimmade/(rimmade+rimmiss)",
```

```
            "stl",
            "GP",
            "adrtg",
            "rimmade+rimmiss",
            "pid",
            "midmade/(midmade+midmiss)"
        ],
        "Best Parameters": {
            "C": 0.1,
            "gamma": "auto",
            "kernel": "rbf"
        },
        "Features SubSet Name": "GA_SVM",
        "Best CV Score": 0.9906786338775623,
        "Test Score": 0.9910055350553506
    },
    {
        "Model": "svm",
        "Features": [
            "DRB_per",
            "FT_per",
            "twoP_per",
            "TP_per",
            "stl_per",
            "pfr",
            "ast/tov",
            "dporpag",
            "obpm",
            "dbpm",
            "gbpm",
            "mp",
            "ogbpm",
            "dreb",
            "treb",
            "stl"
        ],
        "Best Parameters": {
            "C": 1,
            "gamma": "auto",
            "kernel": "rbf"
        },
        "Features SubSet Name": "Wrapper",
        "Best CV Score": 0.990870830687095,
        "Test Score": 0.991389913899139
    },
    {
```

```
    "Model": "svm",
    "Features": [
        "dporpag",
        "stops",
        "twoPM",
        "twoPA",
        "pts",
        "FTM",
        "FTA",
        "dreb",
        "treb",
        "bpm",
        "mp",
        "Min_per",
        "rimmade",
        "midmade+midmiss",
        "rimmade+rimmiss",
        "midmade",
        "gbpm",
        "ogbpm",
        "adjoe",
        "obpm"
    ],
    "Best Parameters": {
        "C": 10,
        "gamma": "auto",
        "kernel": "linear"
    },
    "Features SubSet Name": "Filter",
    "Best CV Score": 0.9907362855332092,
    "Test Score": 0.9912361623616236
},
{
    "Model": "svm",
    "Features": [
        "dporpag",
        "gbpm",
        "adjoe",
        "midmade/(midmade+midmiss)",
        "adrtg",
        "AST_per",
        "ast/tov",
        "rimmade+rimmiss",
        "pid",
        "ORB_per",
        "dunksmade",
```

```
            "ast",
            "stl_per",
            "pts",
            "midmade",
            "stl",
            "FTA",
            "TP_per",
            "oreb",
            "Ortg",
            "GP",
            "twoPA",
            "dunksmiss+dunksmade",
            "rimmade/(rimmade+rimmiss)",
            "dgbpm",
            "Min_per",
            "TPA",
            "twoPM",
            "usg",
            "FTM",
            "mp",
            "treb",
            "stops",
            "twoP_per",
            "drtg",
            "blk",
            "obpm",
            "dbpm",
            "eFG",
            "DRB_per",
            "pfr",
            "TO_per",
            "bpm",
            "TS_per",
            "rimmade",
            "ogbpm"
        ],
        "Best Parameters": {
            "C": 0.1,
            "gamma": "auto",
            "kernel": "rbf"
        },
        "Features SubSet Name": "Embedded",
        "Best CV Score": 0.9906786338775623,
        "Test Score": 0.9910055350553506
    },
    {
```

```
"Model": "svm",
"Features": [
    "GP",
    "Min_per",
    "Ortg",
    "usg",
    "eFG",
    "TS_per",
    "ORB_per",
    "DRB_per",
    "AST_per",
    "TO_per",
    "FTM",
    "FTA",
    "FT_per",
    "twoPM",
    "twoPA",
    "twoP_per",
    "TPM",
    "TPA",
    "TP_per",
    "blk_per",
    "stl_per",
    "ftr",
    "adjoe",
    "pfr",
    "pid",
    "ast/tov",
    "rimmade",
    "rimmade+rimmiss",
    "midmade",
    "midmade+midmiss",
    "rimmade/(rimmade+rimmiss)",
    "midmade/(midmade+midmiss)",
    "dunksmade",
    "dunksmiss+dunksmade",
    "dunksmade/(dunksmade+dunksmiss)",
    "drtg",
    "adrtg",
    "dporpag",
    "stops",
    "bpm",
    "obpm",
    "dbpm",
    "gbpm",
    "mp",
```

```
            "ogbpm",
            "dgbpm",
            "oreb",
            "dreb",
            "treb",
            "ast",
            "stl",
            "blk",
            "pts"
        ],
        "Best Parameters": {
            "C": 0.1,
            "gamma": "auto",
            "kernel": "rbf"
        },
        "Features SubSet Name": "All_Features",
        "Best CV Score": 0.9906786338775623,
        "Test Score": 0.9910055350553506
    }
]
```

Apêndice B

Segundo Apêndice

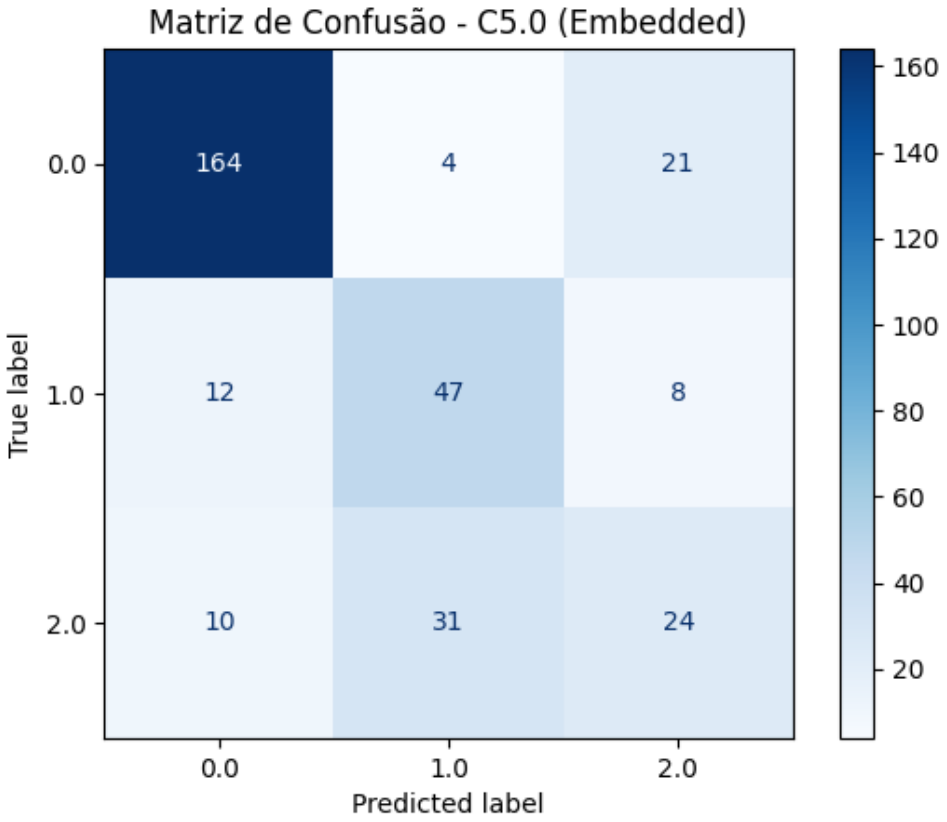

Figura B.1: Matriz de Confusão - C5.0 - Features *Embedded*

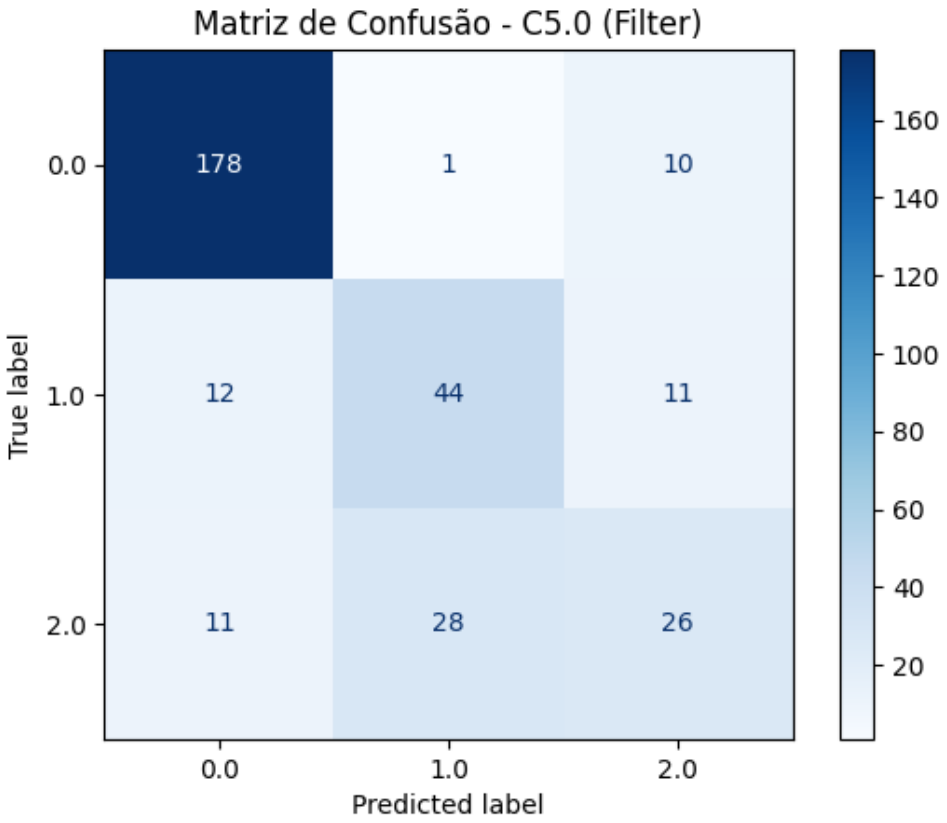

Figura B.2: Matriz de Confusão - C5.0 - Features *Filter*

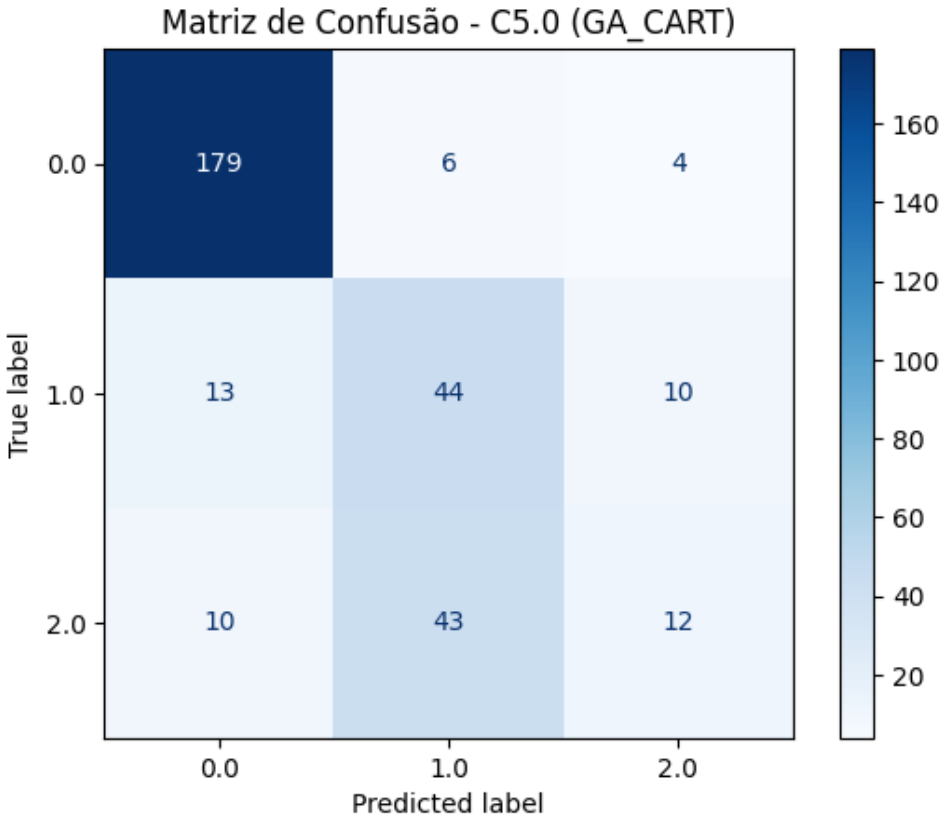

Figura B.3: Matriz de Confusão - C5.0 - Features GA CART

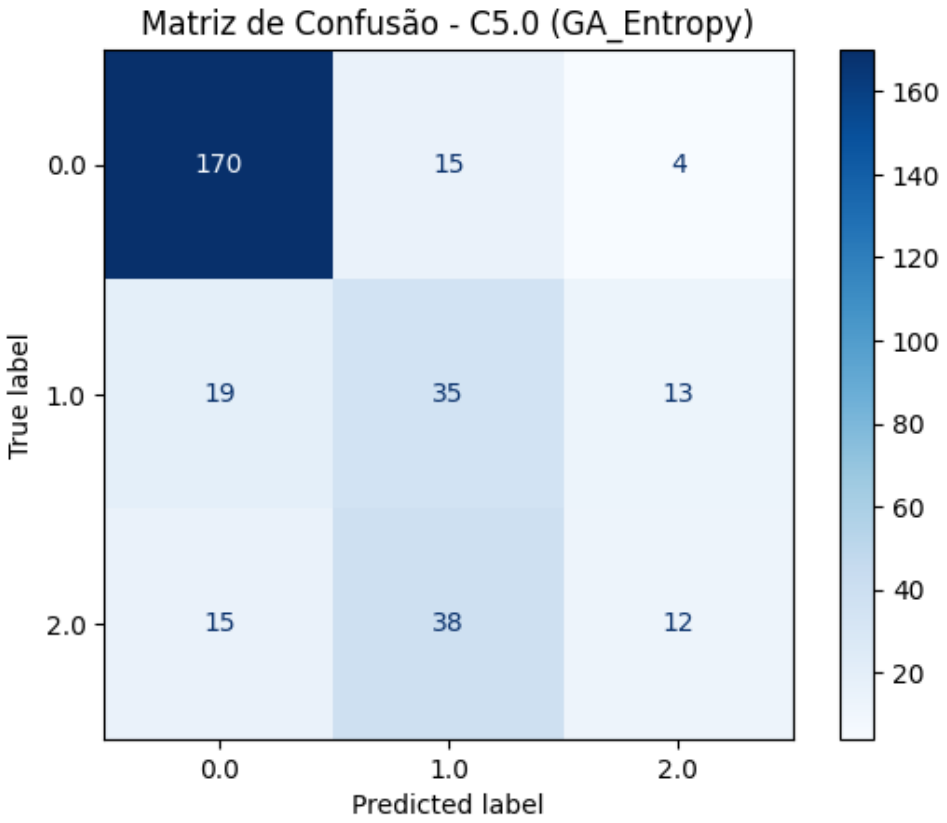

Figura B.4: Matriz de Confusão - C5.0 - Features GA Entropy

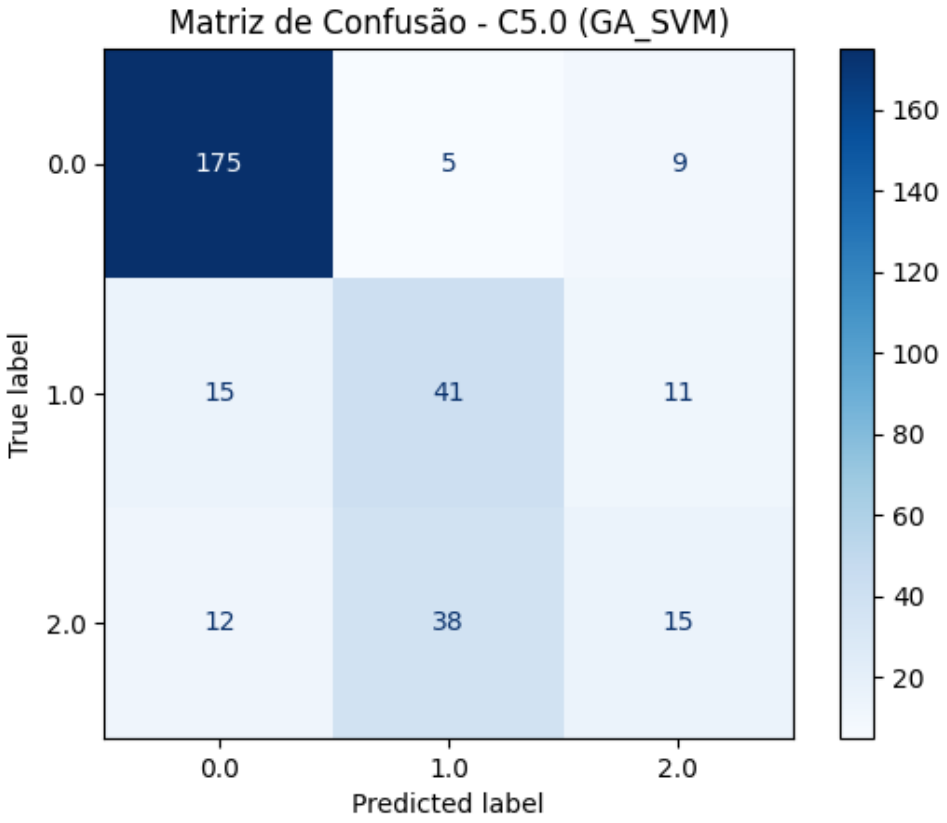

Figura B.5: Matriz de Confusão - C5.0 - Features GA SVM

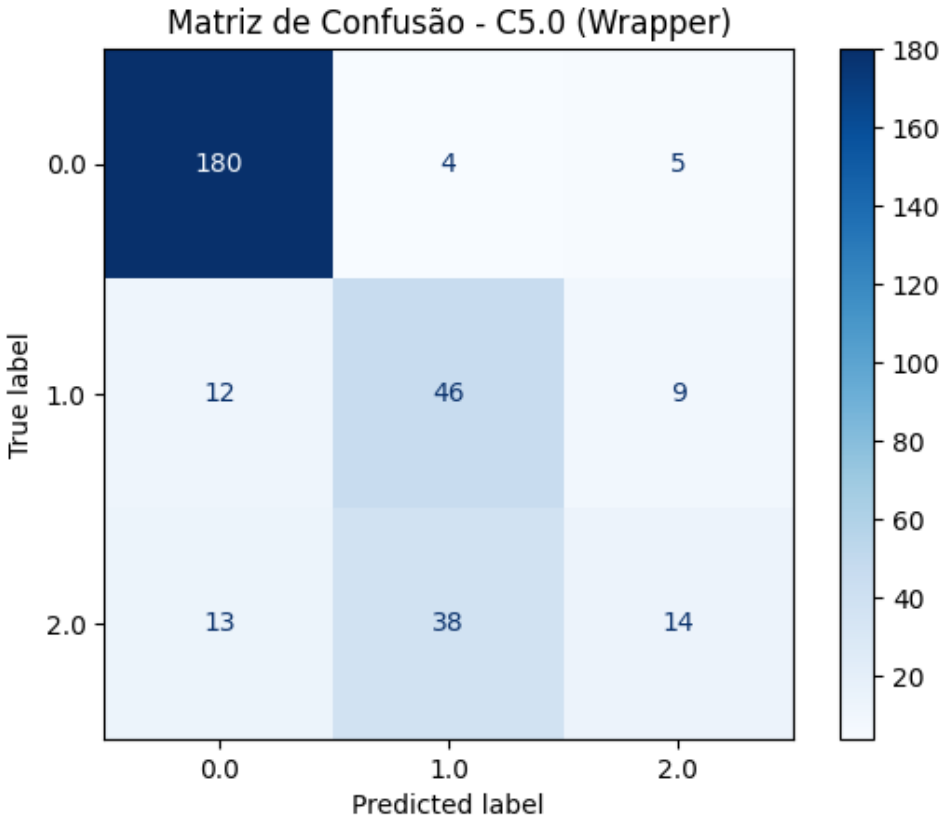

Figura B.6: Matriz de Confusão - C5.0 - Features *Wrapper*

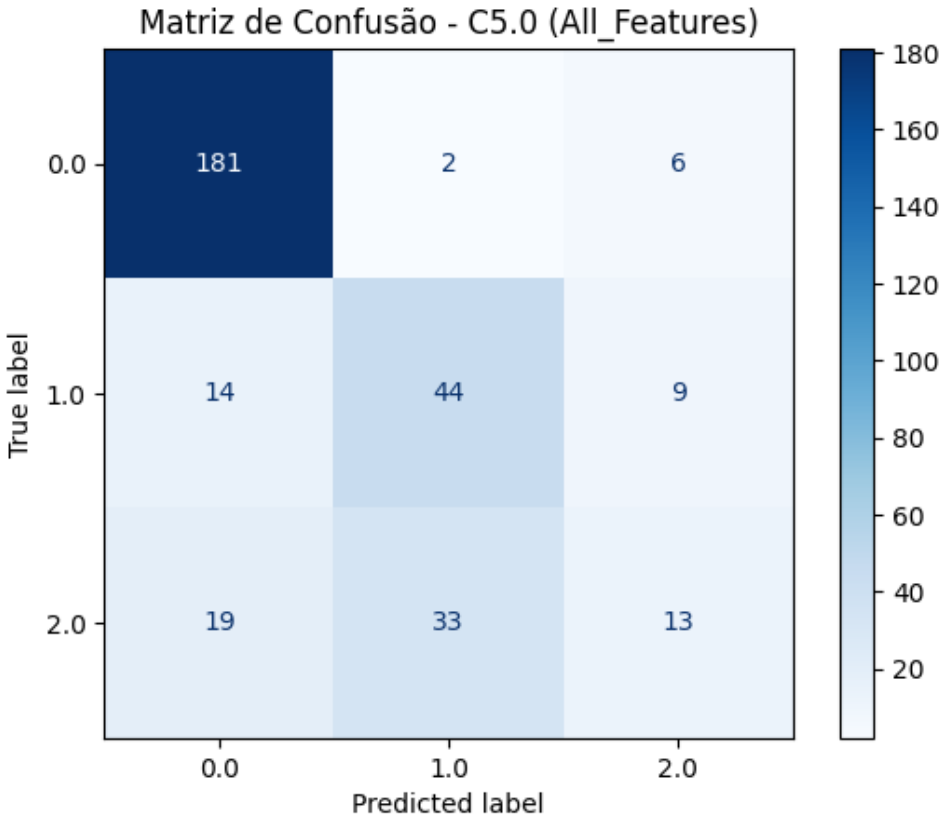

Figura B.7: Matriz de Confusão - C5.0 - Features *All Features*

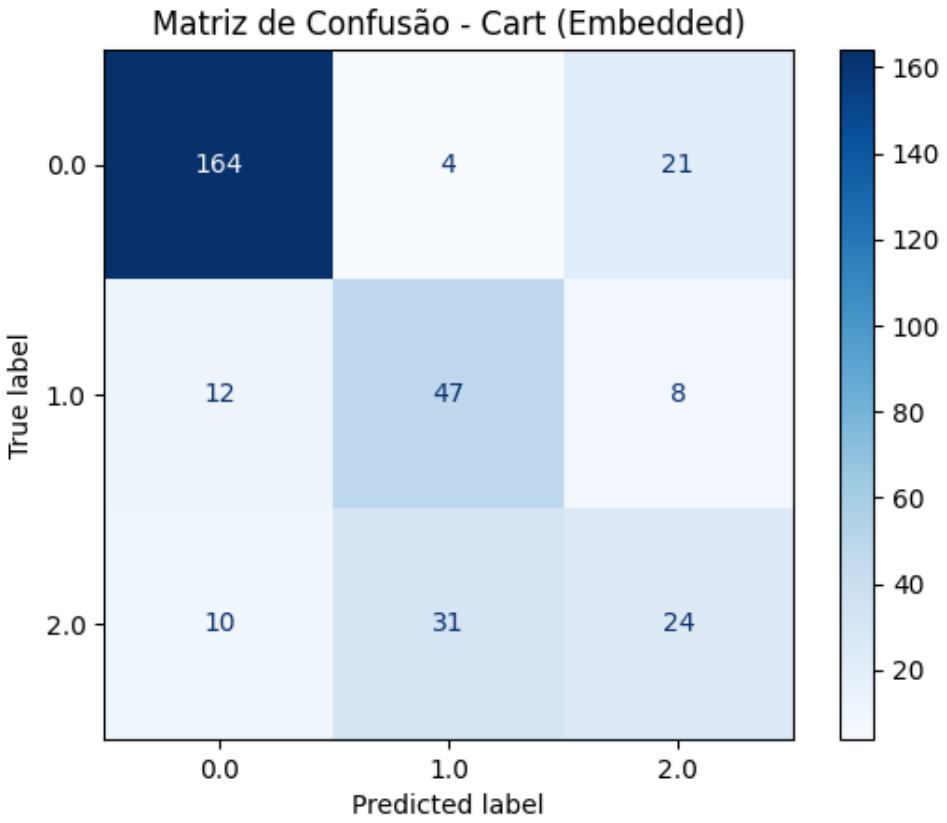

Figura B.8: Matriz de Confusão - Cart - Features *Embedded*

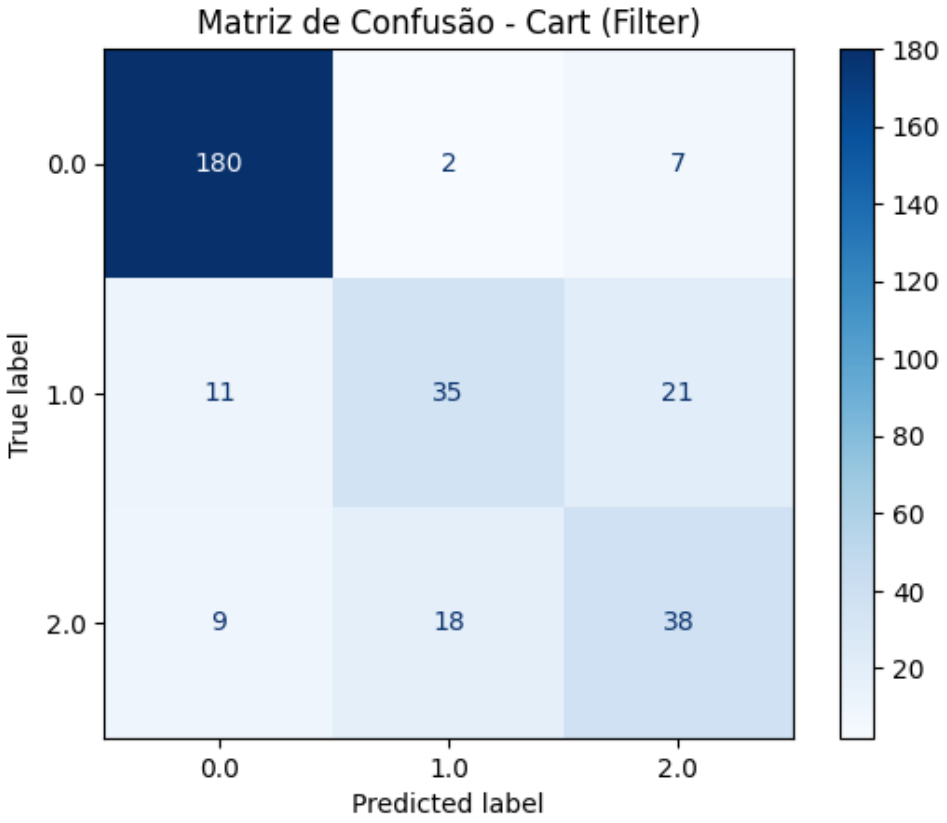

Figura B.9: Matriz de Confusão - Cart - Features *Filter*

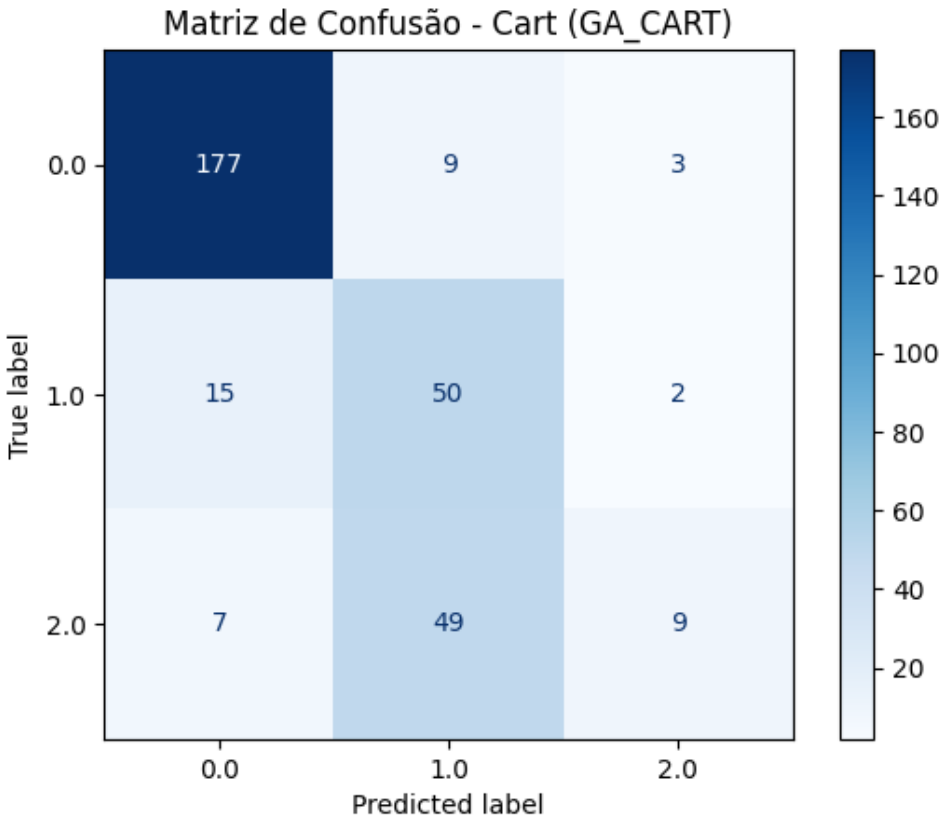

Figura B.10: Matriz de Confusão - Cart - Features GA CART

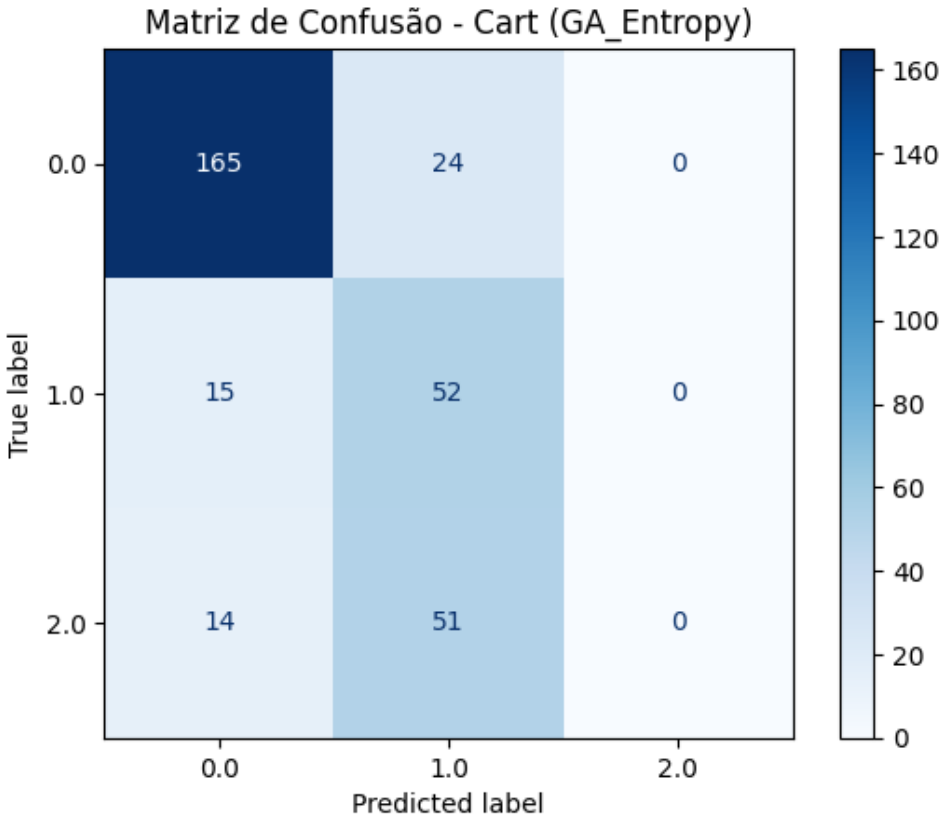

Figura B.11: Matriz de Confusão - Cart - Features GA Entropy

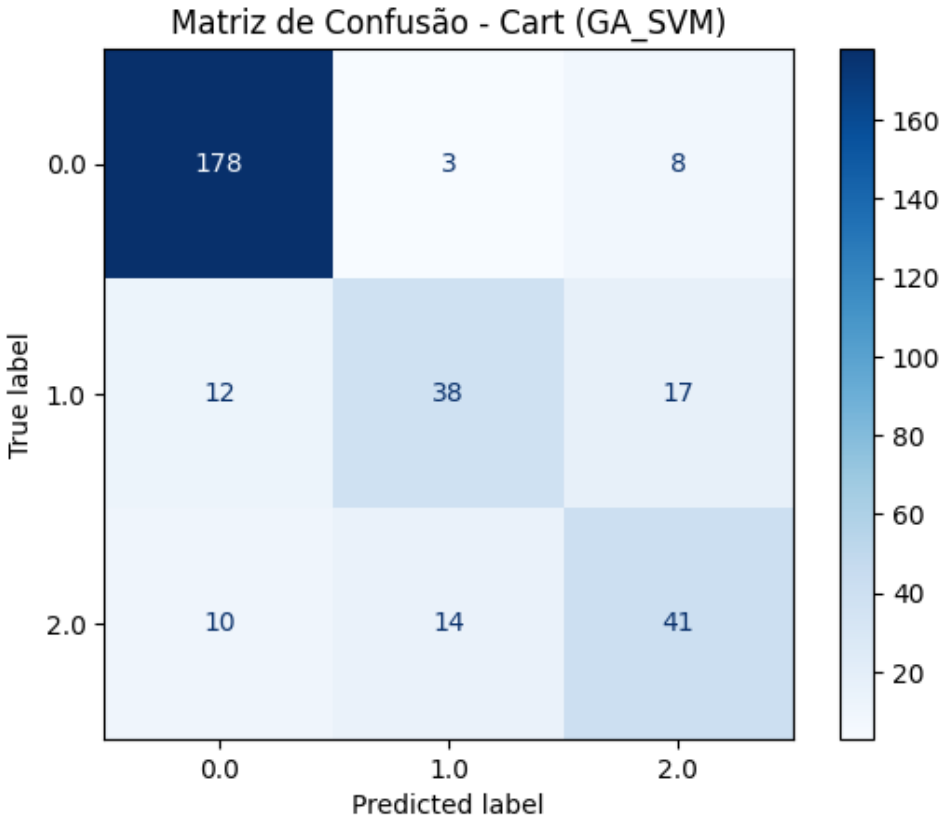

Figura B.12: Matriz de Confusão - Cart - Features GA SVM

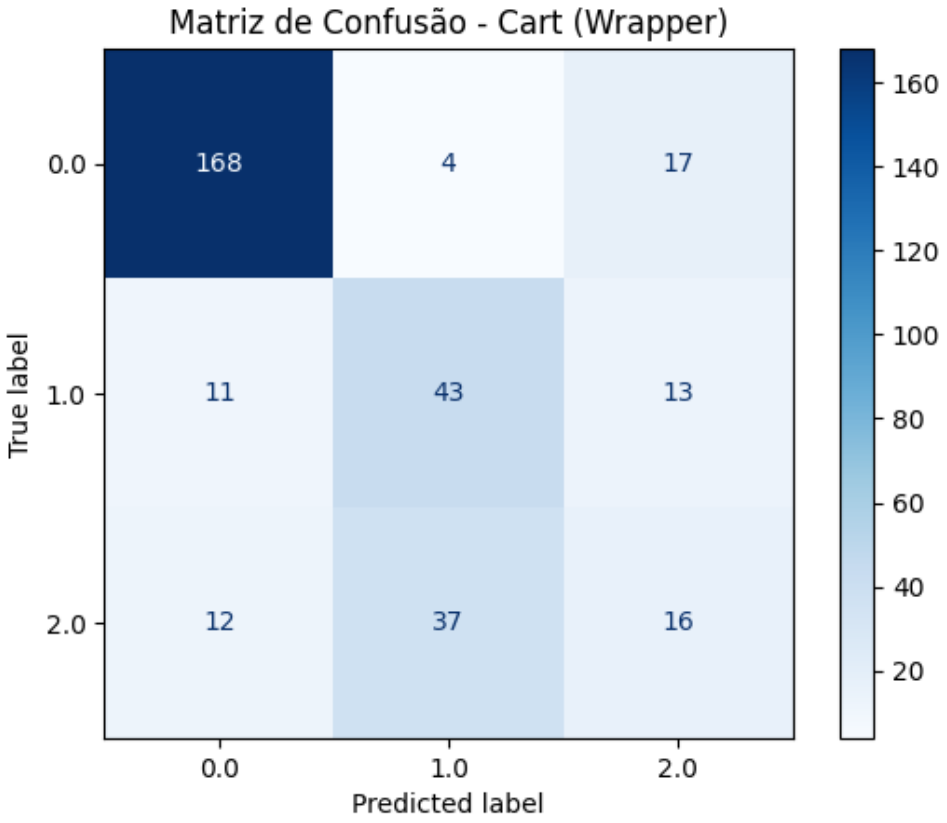

Figura B.13: Matriz de Confusão - Cart - Features *Wrapper*

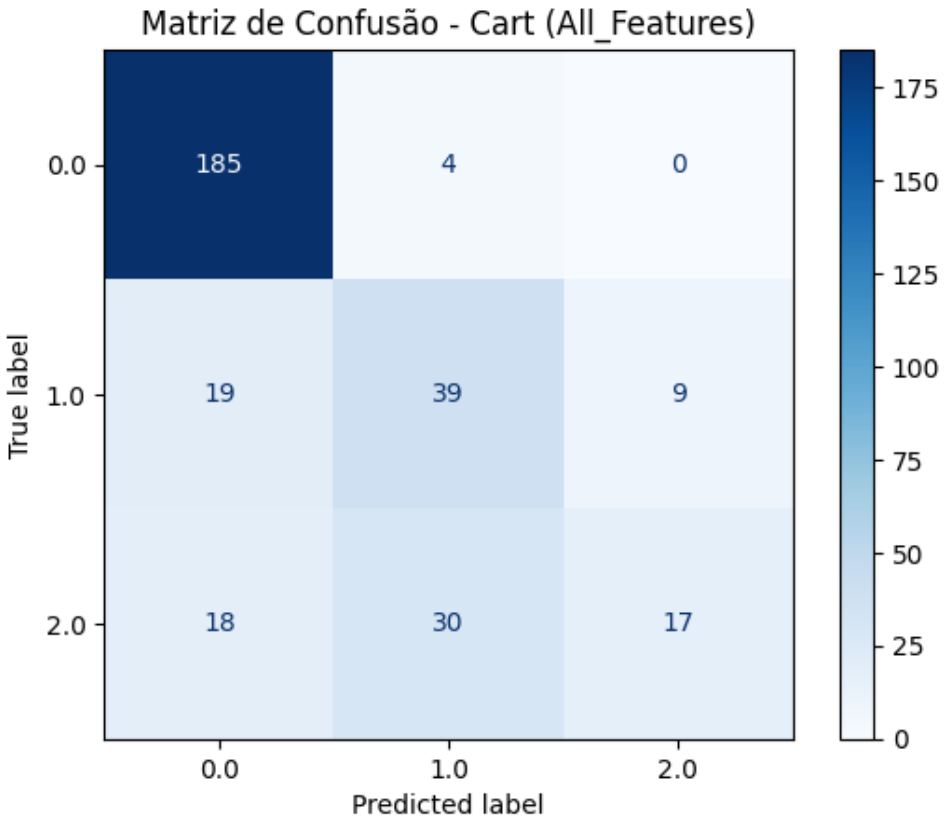

Figura B.14: Matriz de Confusão - Cart - Features *All Features*

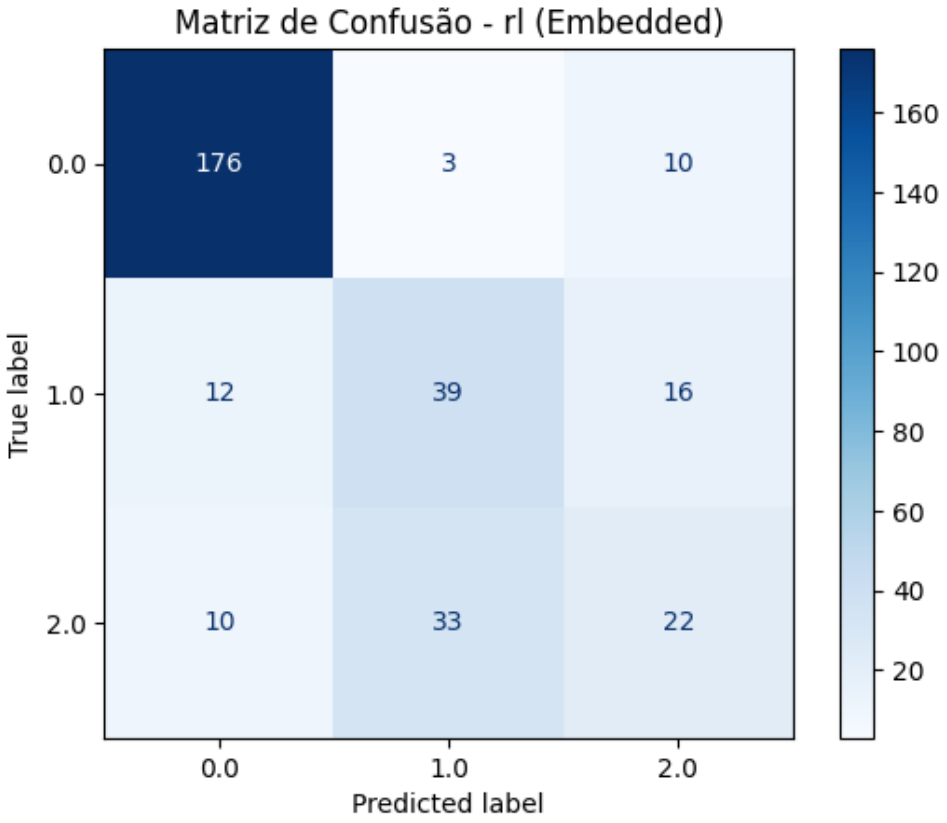

Figura B.15: Matriz de Confusão - RL - Features *Embedded*

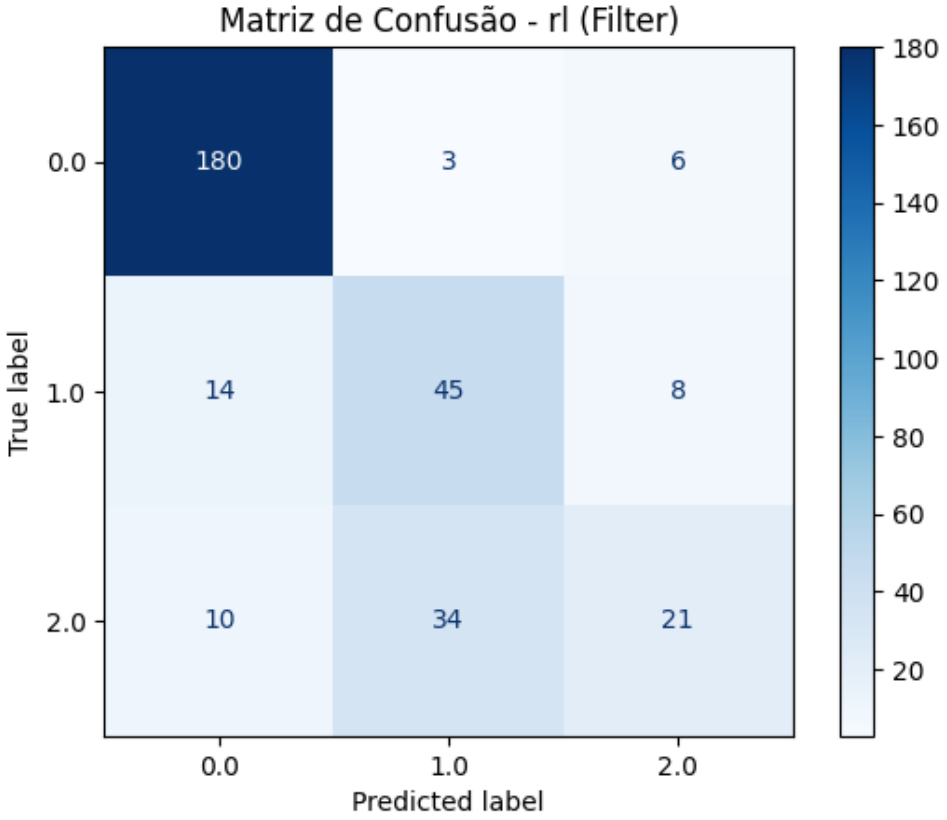

Figura B.16: Matriz de Confusão - RL - Features *Filter*

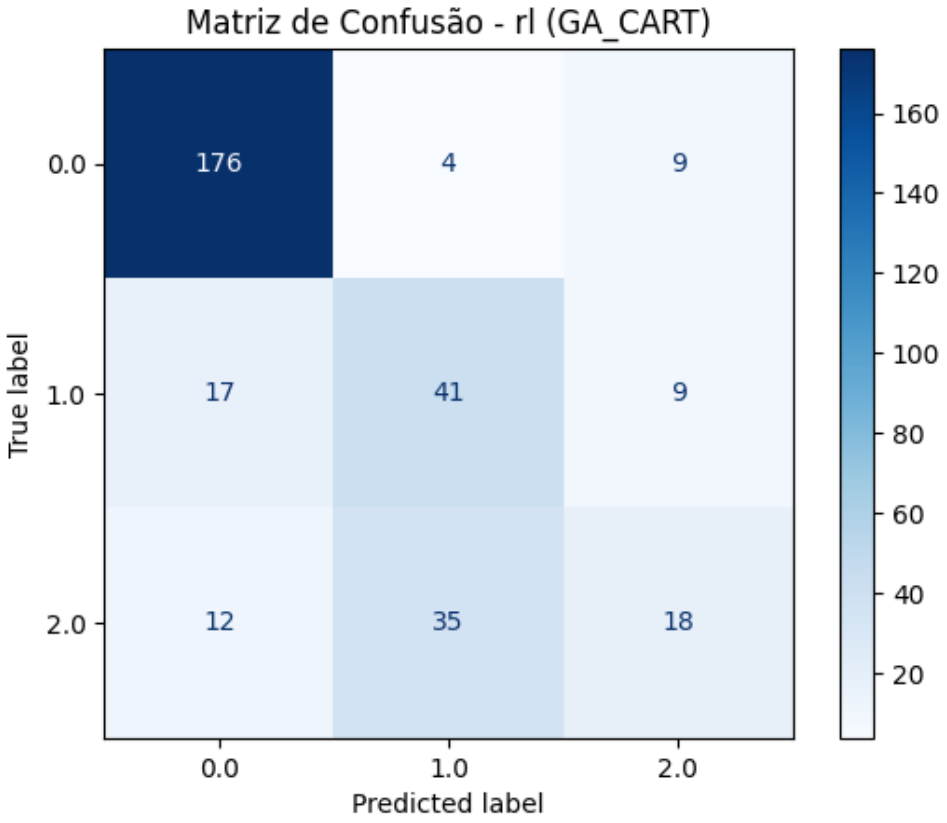

Figura B.17: Matriz de Confusão - RL - Features GA CART

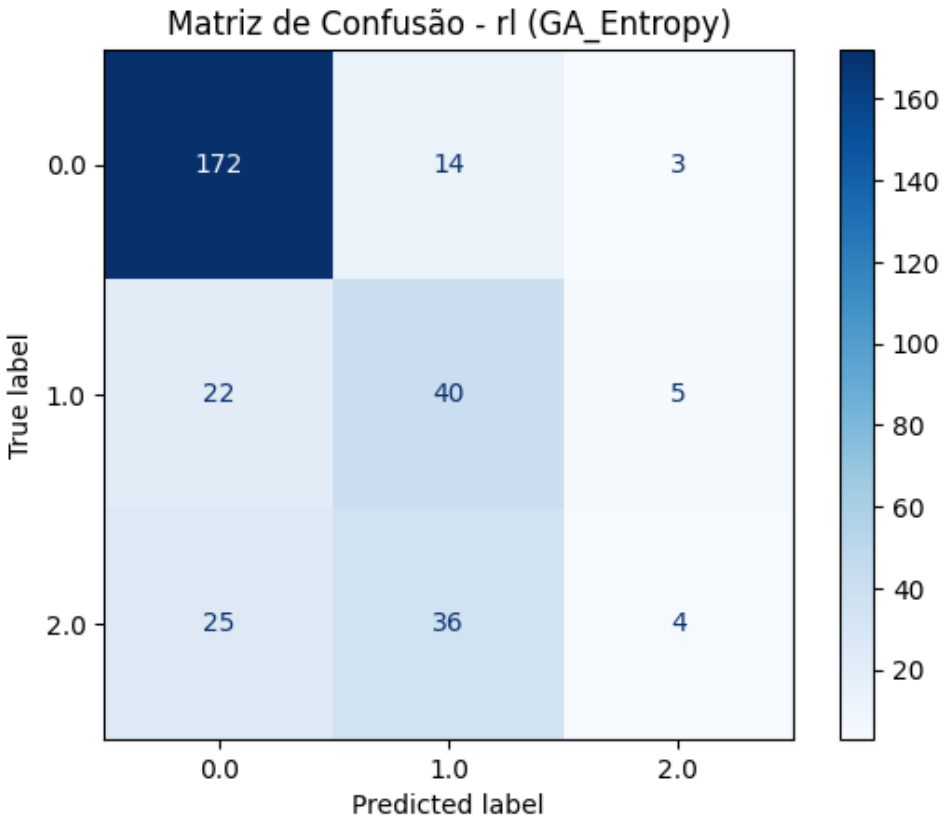

Figura B.18: Matriz de Confusão - RL - Features GA Entropy

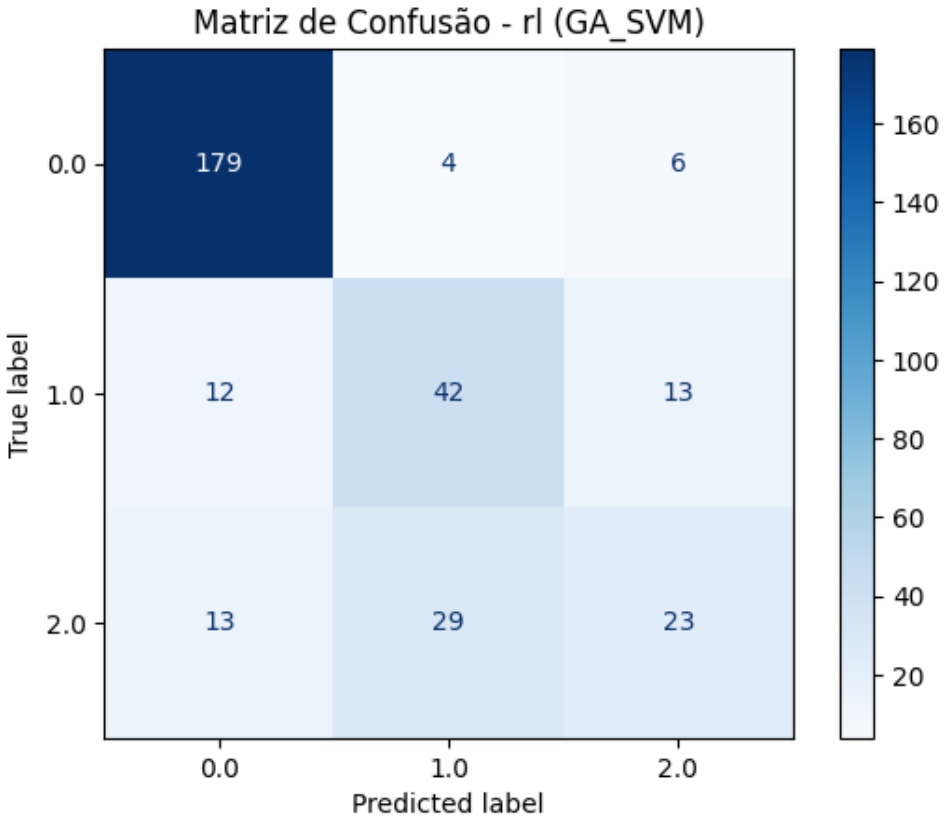

Figura B.19: Matriz de Confusão - RL - Features GA SVM

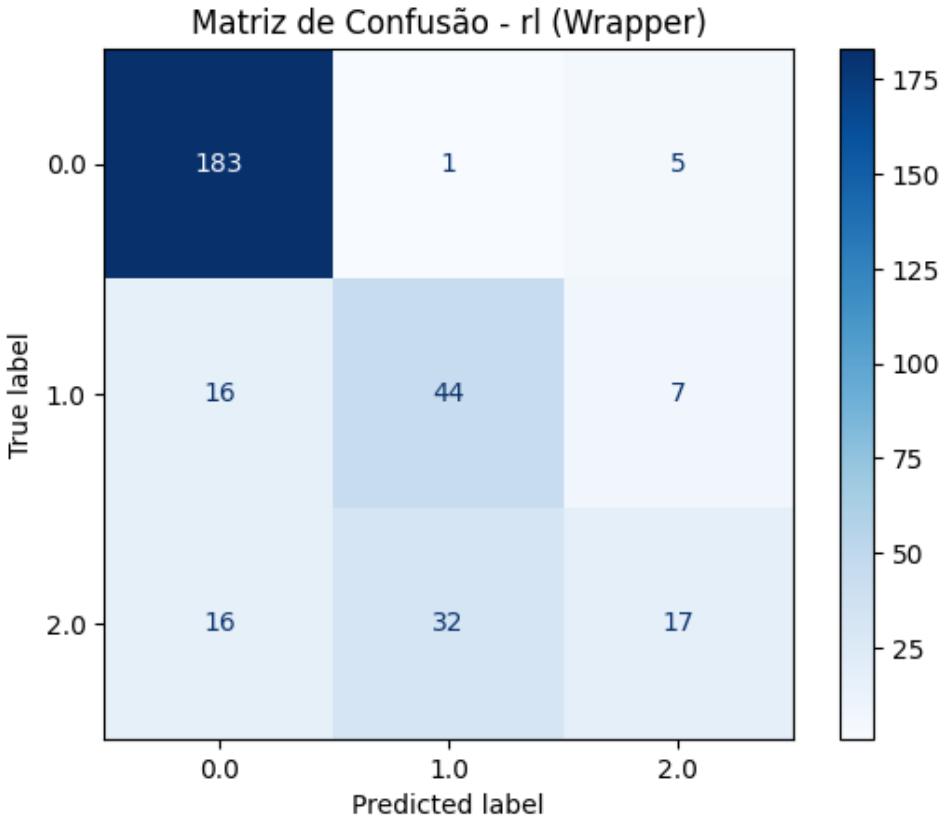

Figura B.20: Matriz de Confusão - RL - Features *Wrapper*

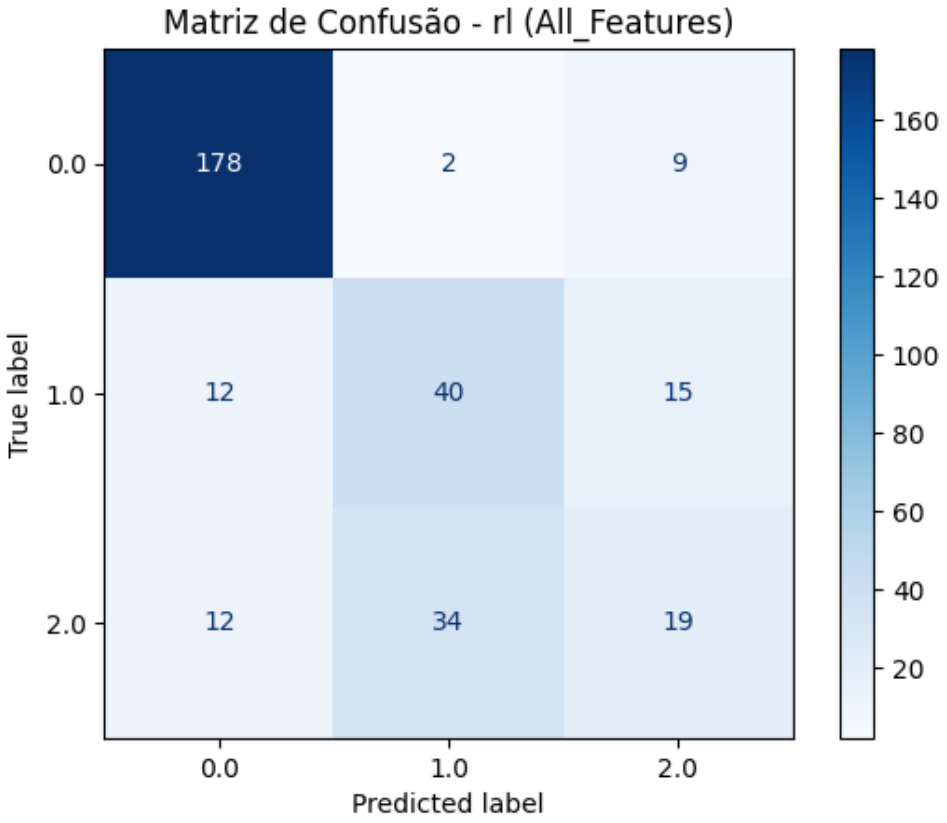

Figura B.21: Matriz de Confusão - RL - Features *All Features*

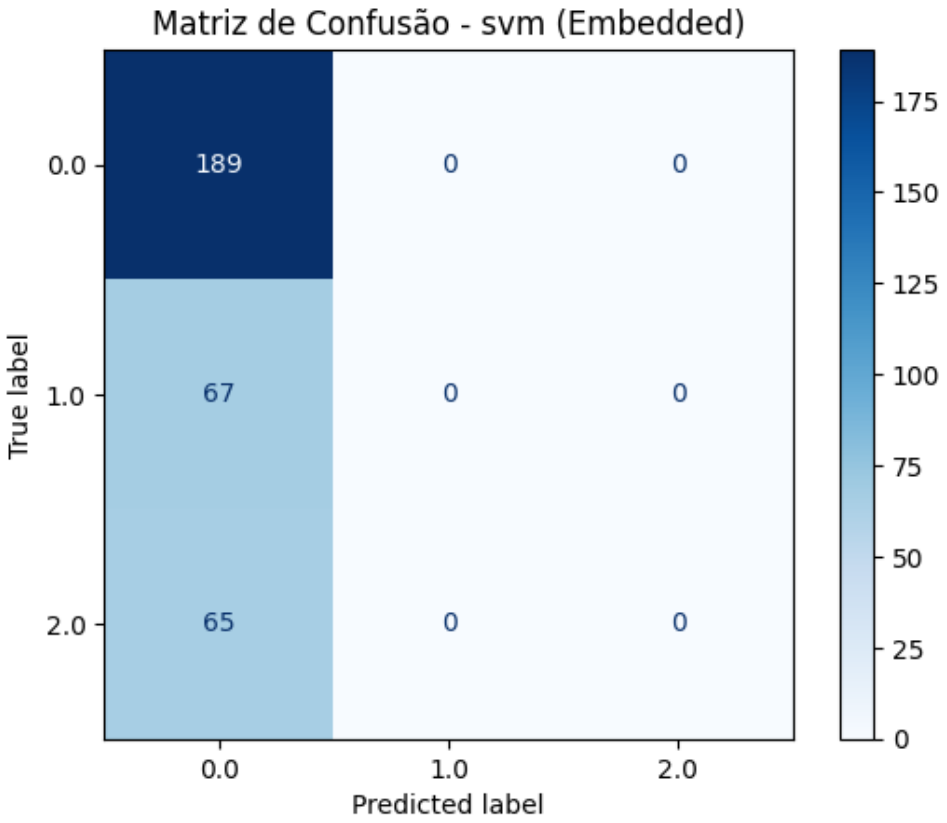

Figura B.22: Matriz de Confusão - SVM - Features *Embedded*

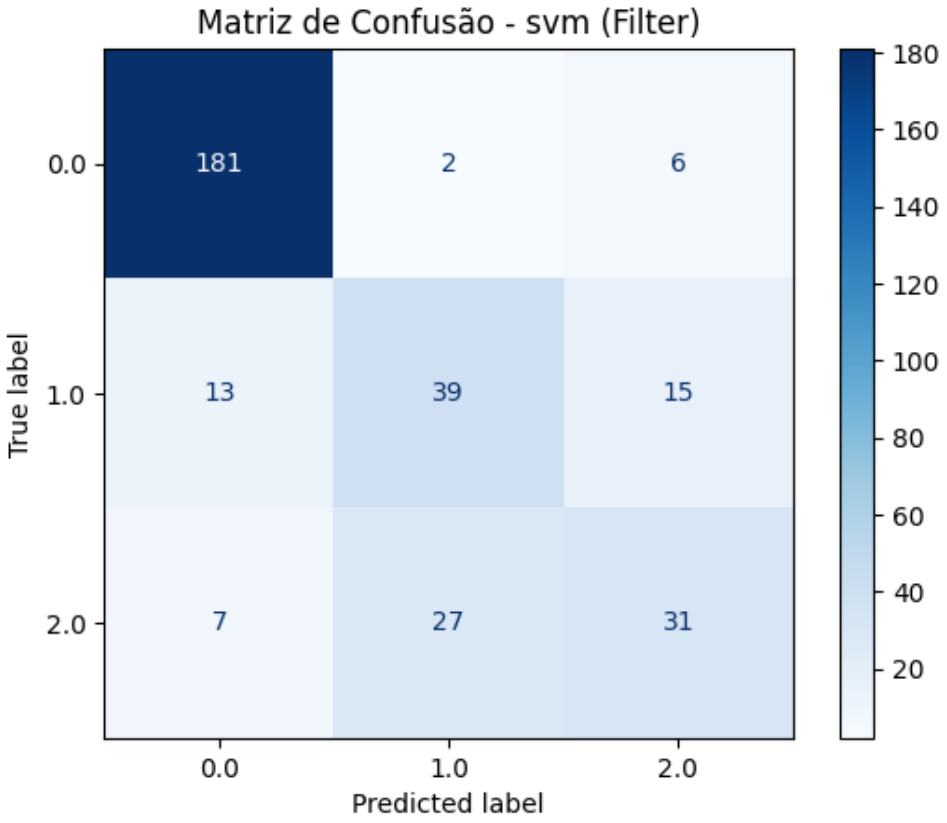

Figura B.23: Matriz de Confusão - SVM - Features *Filter*

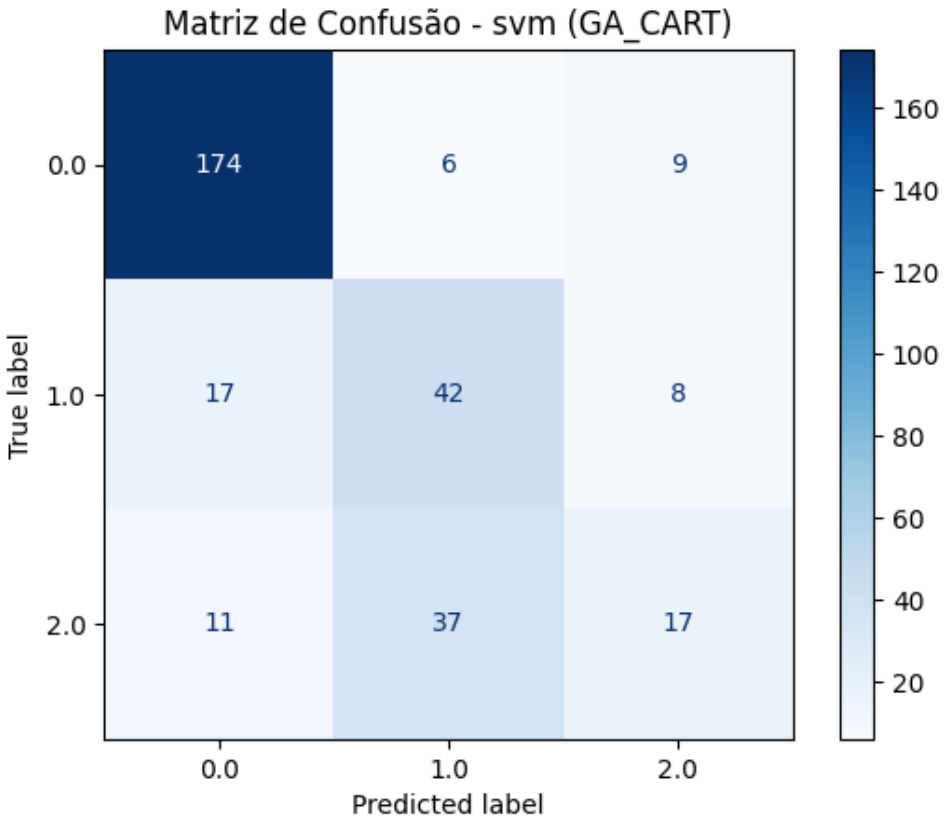

Figura B.24: Matriz de Confusão - SVM - Features GA CART

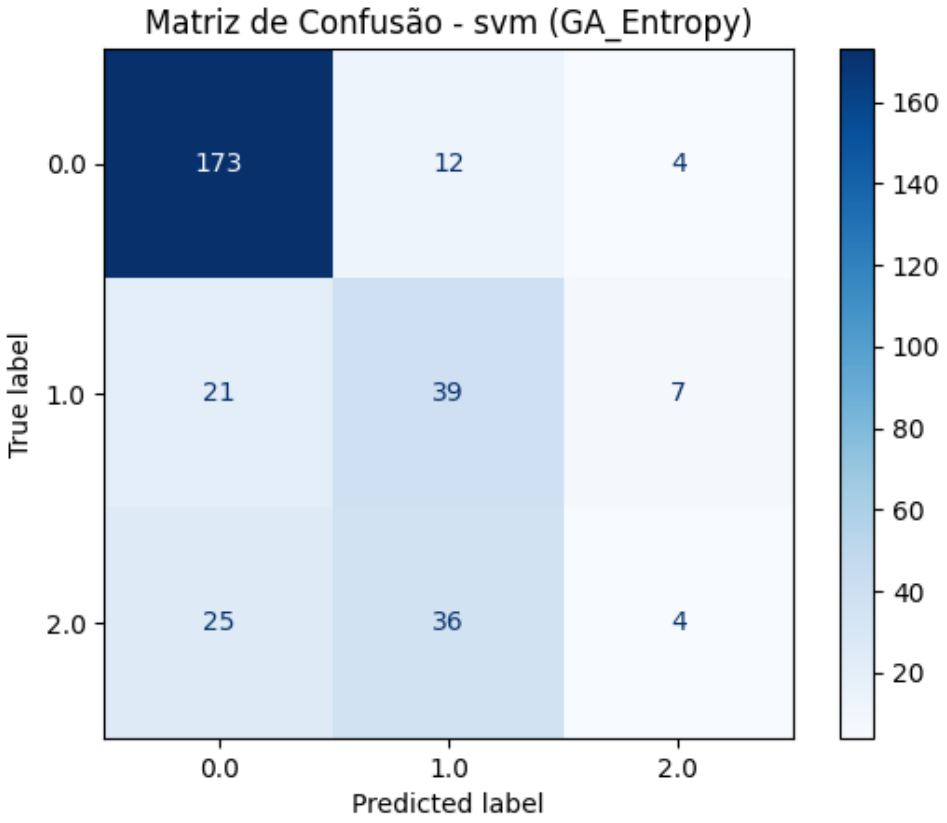

Figura B.25: Matriz de Confusão - SVM - Features GA Entropy

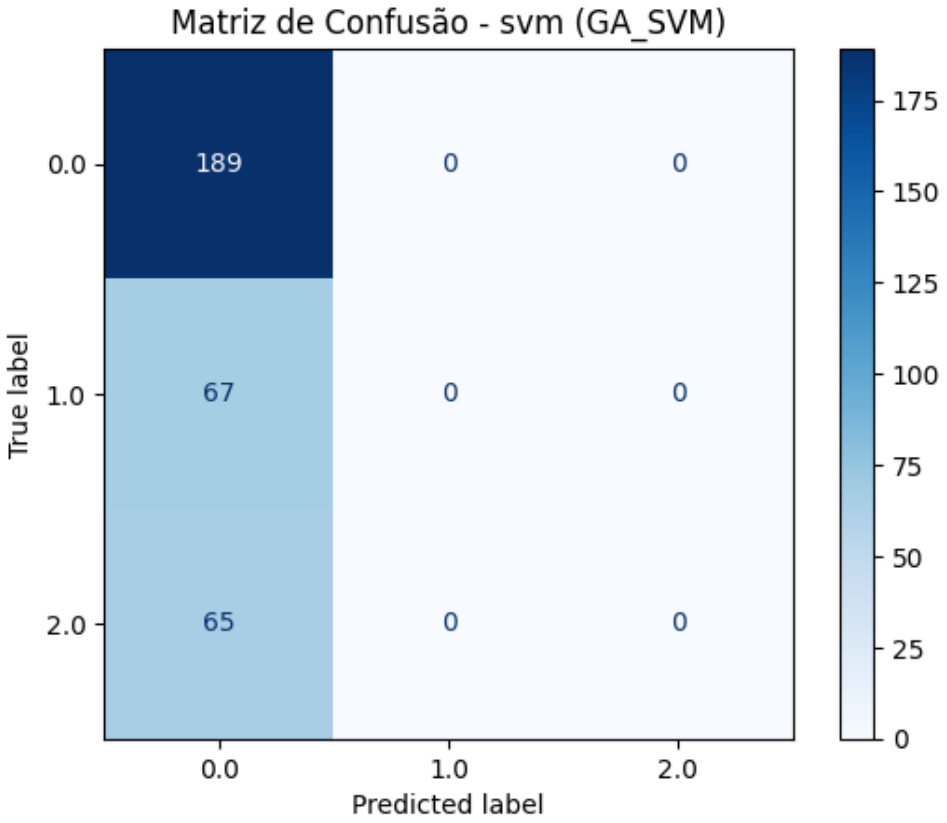

Figura B.26: Matriz de Confusão - SVM - Features GA SVM

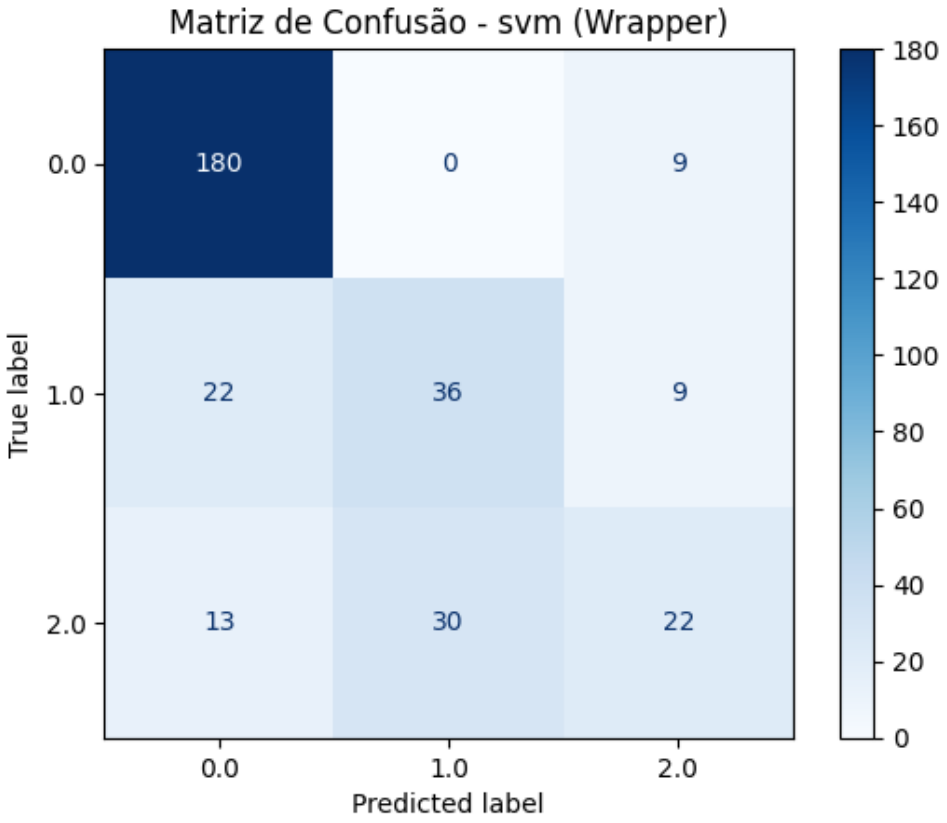

Figura B.27: Matriz de Confusão - SVM - Features *Wrapper*

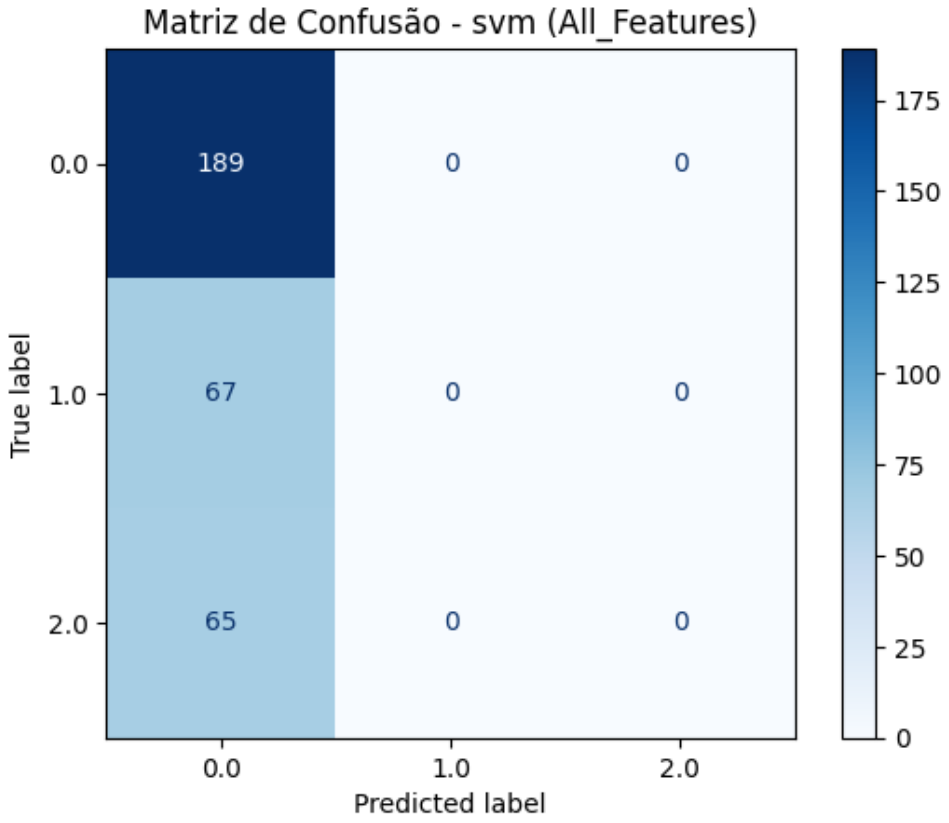

Figura B.28: Matriz de Confusão - SVM - Features *All Features*