# Aprendizado de máquina supervisionado para a previsão de resultados em Counter-Strike: Global Offensive

Matheus Nunes Collacique



Trabalho apresentado para a obtenção do título de especialista em Ciência de Dados & Analytics pelo MBA USP ESALQ.

**Aprendizado de máquina supervisionado para a previsão de resultados em Counter-Strike: Global Offensive**


## Resumo

Com o desenvolvimento de novas tecnologias e a constante popularização dos jogos eletrônicos, o surgimento de competições envolvendo estas modalidades é um evento crescente que abrange diversas comunidades desde o início dos anos 2000 (Scholz, 2019), popularmente conhecido como esports. Atualmente, a presença de patrocinadores, grandes eventos, organizações e premiações, elevou o nível das competições de modo que seja necessário para equipes e jogadores buscarem formas de atingir o máximo potencial para obter sucesso. Neste sentido, em uma das modalidades mais populares atualmente, Counter-Strike: Global Offensive (CS:GO), soluções baseadas na ciência de dados são empregadas de maneira crescente nas comissões técnicas desde a análise de performance das equipes até a análise tática de possíveis adversários. O presente estudo visa desenvolver modelos de aprendizado de máquina supervisionado baseados nas técnicas de Regressão Logística, Árvores de Decisão e Random Forests para a predição de vitórias em CS:GO. Nos modelos desenvolvidos, a capacidade preditiva dos modelos varia entre 83% a 99%. Além disso, empregaram-se técnicas de otimização como Validação-Cruzada Estratificada e métodos de regularização.


**Palavras-chave:** Counter-Strike, classificação, aprendizado de máquina.


**Supervised machine learning for result prediction in Counter-Strike: Global Offensive**


## Abstract

Along with the development of new technologies and the constant popularization of electronic games, the rise of competitions involving such games is an ever-growing event reaching several communities since the early 2000's (Scholz, 2019), commonly referred to as esports. Nowadays, due to the presence of sponsors, big events, organizations and prize pools, the level of competition reaches higher standards, in such a way that teams and players must seek ways to reach their maximum performance. In this context, in one of the most popular modalities nowadays, Counter-Strike: Global Offensive (CS:GO), data science based solutions are employed by the coaching staff of teams, ranging from performance analyses of the team to tactical analyses of possible opponents. This study aims the development of supervised machine learning models employing Logistic Regression, Decision Trees and Random Forests for the prediction of round wins in CS:GO. The predictive capabilities of the developed models are contained in the interval of 83% to 99%. Moreover, several optimization techniques were employed such as Stratified Cross-Validation and regularization methods.


**Keywords:** Counter-Strike, classification, machine learning.

**Introdução**

Historicamente, a prática esportiva se mostra presente como parte integral da sociedade não somente como forma de lazer e desenvolvimento físico, mas como um grande fenômeno cultural e uma forma de expressão (MCCOMB, 2004). Concomitantemente ao avanço da tecnologia, o processo de modernização, globalização e o posterior advento dos jogos eletrônicos, o surgimento de competições voltadas a estes jogos está intrinsecamente ligado aos eventos previamente mencionados (Scholz, 2019).

Com a popularização da modalidade e suas comunidades, diversos segmentos industriais pioneiros e organizações iniciaram investimentos na área, moldando o recém-criado ecossistema e ocasionando a criação dos esportes eletrônicos ou, em outras palavras, *esports*. O avanço da popularidade dos *esports* acompanhado da constante evolução da área de tecnologia da informação resultou em um acelerado crescimento da área, com o ingresso de equipes, jogadores, organizações, competições e, posteriormente, a formalização dos profissionais da área. (Collis, 2020)

Atualmente, entre as modalidades mais populares nos *esports* encontra-se o jogo *Counter-Strike: Global Offensive* (CS:GO). CS:GO é um jogo no estilo *First Person Shooter* desenvolvido pelas empresa Valve Corporation e Hidden Path Entertainment. No jogo, duas equipes compostas por 5 jogadores disputam em tempo real onde, de maneira similar ao vôlei, o jogo é dividido em rodadas, de modo em que cada rodada deve haver um vencedor (sem possibilidade de empate). Cada rodada é disputada em tempo real por todos os jogadores onde diversos aspectos estratégicos, posicionais, econômicos e parâmetros físicos como tempo de reação devem ser levados em conta para obter a vitória (Makarov, 2017).

Dessa forma, devido à complexidade advinda da própria natureza da modalidade, inevitavelmente as organizações e equipes profissionais começaram a optar pelo uso de técnicas baseadas em dados para auxiliar na tomada de decisão durante o jogo, assim como no reconhecimento de padrões, estudo de estratégias viáveis e verificação de desempenho dos atletas, incorporando analistas e cientistas de dados em suas comissões técnicas. (KHROMOV et al., 2019)

Nos esportes tradicionais, a ciência de dados é aplicada não somente nas decisões tomadas durante a prática esportiva, envolvendo também decisões gerenciais e

comportamentais sobre jogadores e treinadores assim como em decisões voltadas a negócios por parte de diretores e gerentes de equipes (MORGULEV; AZAR; LIDOR, 2018).

Ainda assim, diversas dificuldades ainda se mostram presentes para o emprego de métodos baseados em dados (principalmente no contexto de aprendizado de máquina) no estudo de esportes tradicionais, por exemplo, a coleta de dados e, principalmente, o tratamento de dados e o próprio processo de seleção de variáveis explicativas e o treinamento de modelos (RICHTER; O'REILLY; DELAHUNT, 2021).

Atualmente, observa-se um crescimento na execução de estudos e pesquisas acadêmicas dentro do contexto dos *esports*, produzindo conhecimento multidisciplinar em múltiplas áreas incluindo psicologia e ciências cognitivas, tecnologia da informação, economia e direito, por exemplo (REITMAN et al., 2019). Estudos na área de ciência de dados, por sua vez, variam nas suas aplicações e apresentam-se desde o desenvolvimento de inteligência artificial para o reconhecimento de padrões de jogo à previsão de vitória de uma determinada equipe em uma partida competitiva (Makarov et al., 2017).

Neste último tópico, observa-se uma similaridade com os modelos de previsão desenvolvidos para esportes tradicionais, isto é, os modelos baseiam-se majoritariamente no desempenho passado de equipes e seus jogadores e estimam o resultado da partida atual baseado em métricas e suas comparações com a equipe rival, não levando em conta aspectos da partida em andamento – seja pela dificuldade da coleta e processamento dos dados ou pela própria natureza do problema. (Makarov et al., 2017) (Wang, 2022).

Neste sentido, ainda que exista uma constante popularização e profissionalização dos *esports,* a aplicação de conceitos e a atuação de profissionais envolvidos na área de ciência de dados em *esports* ainda está em desenvolvimento, refletindo na carência de literatura especializada disponível sobre o tema e, principalmente, na disponibilidade de ferramentas e técnicas acessíveis baseadas em dados para o processo de aprimoramento do entendimento e estudo das diversas modalidades disponíveis.

Por fim, o presente estudo possui como objetivo o desenvolvimento de modelos de aprendizado de máquina supervisionados para a previsão de resultados em partidas profissionais de Counter-Strike: Global Offensive, abrangendo desde o processo de extração, tratamento e modelagem de dados até a avaliação da performance dos modelos

construídos, visando ampliar a geração de conhecimento no que tange a ciência de dados e suas aplicações em competições e modalidades esportivas.

## Material e Métodos

### Counter-Strike: Global Offensive

Conforme observado na seção introdutória deste estudo, o jogo Counter-Strike: Global Offensive (CS:GO) foi empregado como a modalidade escolhida para a previsão de resultados. Partidas competitivas de CS:GO são disputadas em uma melhor de 30 rodadas de 1 minuto e 55 segundos cada, de modo que a primeira equipe a atingir 16 pontos vence o jogo, com possibilidade de prorrogação caso o jogo atinja o placar de 15-15.

De maneira geral, para fins demonstrativos, define-se o jogo como um “simulador militar”, baseado em duas equipes opostas: uma equipe atacante (denominada com a sigla T), e uma equipe defensora (denominada com a sigla CT). No contexto do jogo, o objetivo da equipe T é plantar um dispositivo explosivo em uma de duas áreas que devem ser defendidas pela equipe CT.

No início de cada rodada, os jogadores possuem uma quantia inicial de dinheiro (no contexto do jogo, não com valores reais) em que podem comprar equipamentos para cada rodada. O sucesso da rodada define a quantidade de dinheiro que uma equipe terá para comprar novos equipamentos em uma rodada seguinte, de modo que a derrota em uma rodada implique em menos dinheiro obtido em uma rodada seguinte. Caso algum jogador seja eliminado durante uma rodada, o jogador perde seus equipamentos e fica inativo até o início da próxima rodada.

Caso o dispositivo explosivo seja plantado pela equipe T em uma das áreas que devem ser defendidas pela equipe CT, a equipe T deve defender o dispositivo por 40 segundos e evitar que a equipe CT desarme o mesmo.

A Figura abaixo apresenta um fluxograma geral de uma partida de CS:GO:

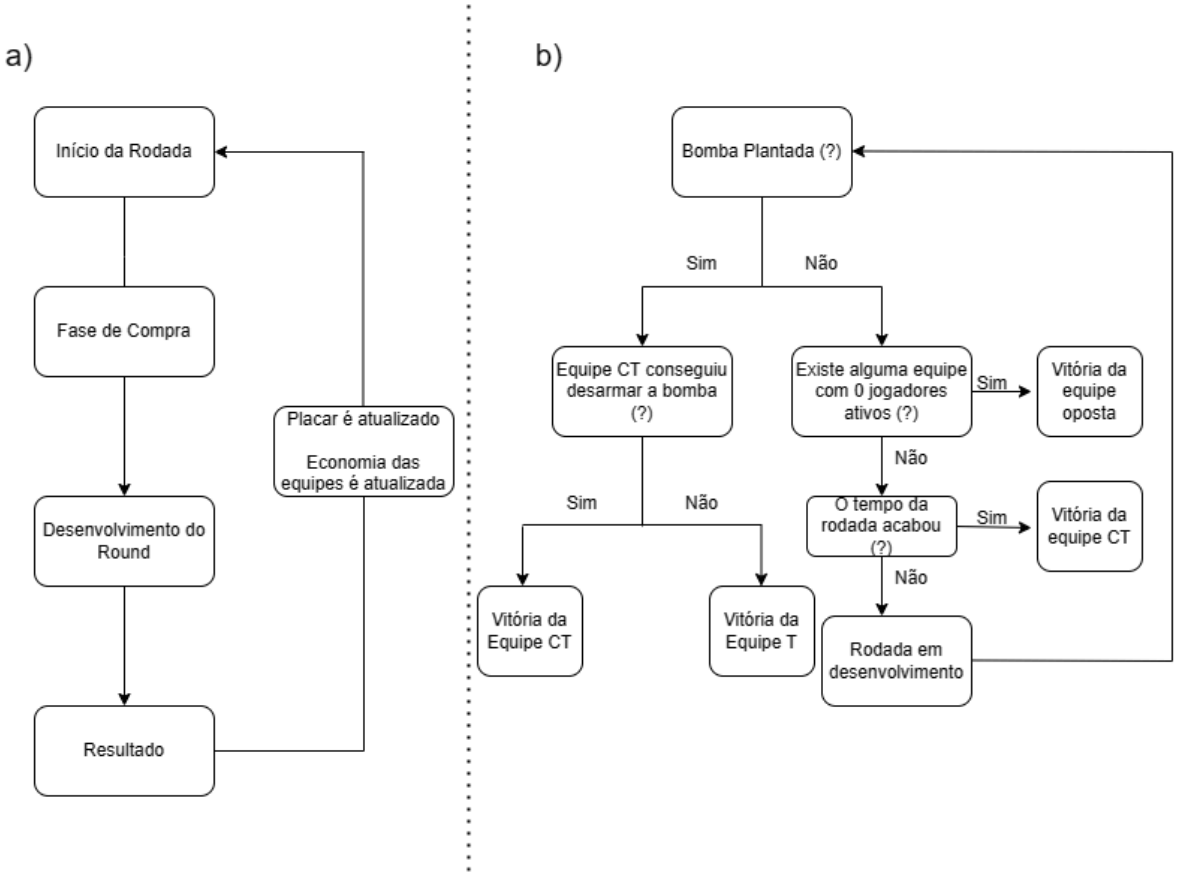


Figura 1. Fluxograma geral de uma rodada de CS:GO: a) Visão Geral e b) Desenvolvimento de uma rodada e critérios de vitória.

Fonte: Dados originais da pesquisa.

Dada a complexidade do jogo, torna-se mais evidente a necessidade de abordagens baseadas em dados para a obtenção de sucesso de equipes profissionais.

**Extração, tratamento e análise exploratória de dados**

A Figura 2 ilustra o esquema geral do fluxo de trabalho empregado para a extração e tratamento dos dados, criação, treinamento e validação do modelo.

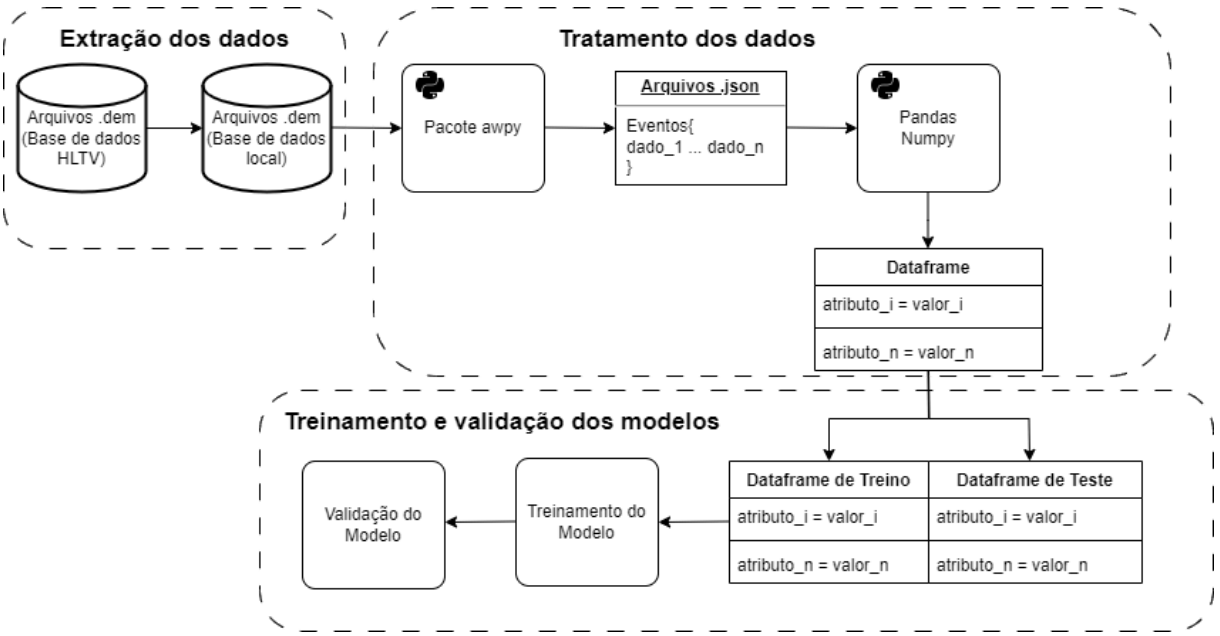


Figura 2. *Pipeline* de dados para extração, tratamento e treinamento/validação dos dados.

Fonte: Dados originais da pesquisa.

Os arquivos de dados referentes à partidas disputadas de CS:GO foram obtidos a partir da plataforma HLTV (“HLTV.org - The home of competitive Counter-Strike”, 2023). No encerramento de uma partida profissional em uma competição, a partida é disponibilizada para acesso público pela plataforma e a organizadora contendo dados relativos a todos os eventos que ocorreram no jogo, a fim de reprodução posterior por parte de um usuário final.

As partidas foram selecionadas com base nos seguintes critérios:

- A partida ocorreu no ano de 2023;
- A partida é acessível ao público;
- Os times participantes são times profissionais registrados na plataforma HLTV (“HLTV.org - The home of competitive Counter-Strike”, 2023);

Inicialmente, em uma primeira avaliação da metodologia, 28 partidas distintas foram selecionadas para compor os dados de treino e teste dos modelos posteriores.

A partir da obtenção dos arquivos .dem contendo os eventos que ocorreram durante uma partida profissional de CS:GO, utilizou-se a biblioteca AWPY (Xenopoulos, 2022) para a extração dos eventos e a linguagem Python para a conversão de todos os eventos presentes em um jogo para em arquivos do tipo *JavaScript Object Notation* (JSON). Dessa

forma, foi construído um *dataframe* com dados pertinentes à ambas equipes registrados periodicamente dentro do servidor.

Primeiramente, optou-se pela remoção dos itens presentes no conjunto de dados que não agregassem valor para o futuro modelo preditivo, isto é, dados que representassem instantes de uma rodada onde não houve evolução significativa da rodada e nenhum jogador de ambas as equipes houvesse sido eliminado. Neste sentido, realizou uma contagem inicial das classes (variáveis dependentes) extraídas do conjunto de dados após o tratamento inicial do conjunto de dados. Os dados foram compilados e exibidos na Figura 3.

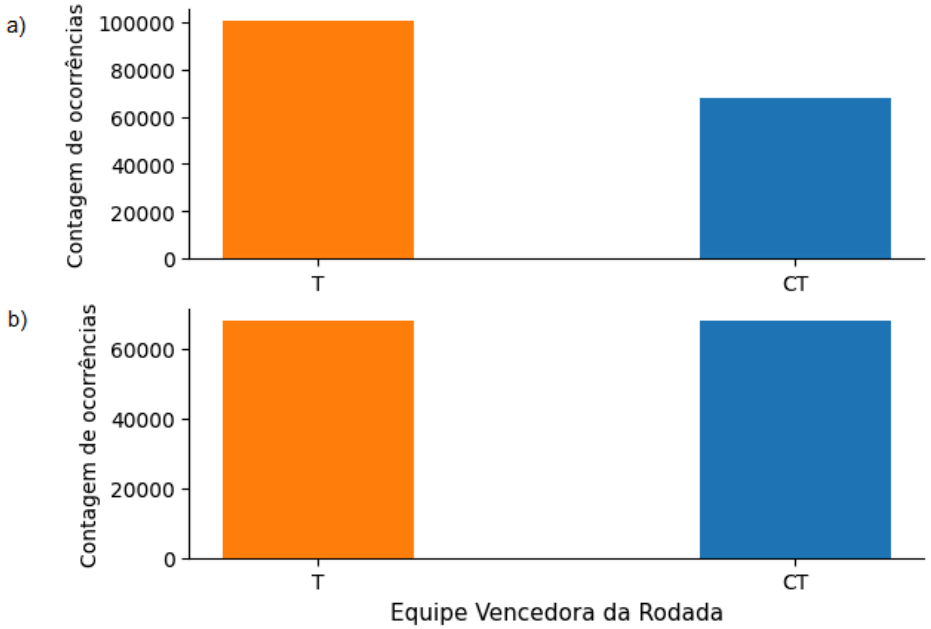


Figura 3. Contagem inicial de classes presentes no conjunto de dados: a) antes do balanceamento de classes por *Random Undersampling* e b) depois do balanceamento de classes por *Random Undersampling*.

Fonte: Dados originais da pesquisa.

Verificou-se a existência de um desbalanceamento de classes do conjunto de dados de modo onde a classe referente ao valor 0 (indicando que a equipe T foi a vencedora de uma rodada específica) possuía 32477 ocorrências a mais que a classe 1 (onde a equipe CT foi a vencedora), compondo cerca de 59,6% do conjunto de dados. Dessa maneira, foi realizado um balanceamento do conjunto de dados através do método de *Random Undersampling* (RUS), onde dados referentes à classe majoritária foram removidos aleatoriamente até que a igualdade na distribuição de classes foi alcançada (Figura 2b). O

conjunto de dados foi posteriormente randomizado e foram reservados 33% dos dados obtidos para o teste do modelo treinado.

Inicialmente, as classes escolhidas para a construção dos modelos estão mostradas na tabela a seguir:

Tabela 1. Variáveis selecionadas inicialmente para a construção dos modelos. A variável dependente é denotada pelo símbolo (*) na Tabela abaixo.

| Variável | Descrição | Tipo de Dado |
|---|---|---|
| aliveT | Número de jogadores ativos da equipe T | Quantitativo discreto |
| aliveCT | Número de jogadores ativos da equipe CT | Quantitativo discreto |
| eqValT | Valor de equipamento da equipe T | Quantitativo discreto |
| eqValCT | Valor de equipamento da equipe CT | Quantitativo discreto |
| totalUtilityT | Quantidade de itens da equipe T | Quantitativo discreto |
| totalUtilityCT | Quantidade de itens da equipe CT | Quantitativo discreto |
| totalTHP | Soma de pontos de vida da equipe T | Quantitativo discreto |
| totalCTHP | Soma de pontos de vida da equipe CT | Quantitativo discreto |
| clockTime | Tempo restante da rodada em segundos | Quantitativo discreto |
| bombPlanted | Bomba plantada | Categórico dicotômico (0, 1) |
| roundWinner* | Vencedor da rodada | Categórico dicotômico (0, 1) |

Fonte: Dados originais da pesquisa.

A variável dependente, consequentemente, apresenta-se na forma binária, onde um valor verdadeiro refere-se à vitória da equipe CT enquanto um valor falso refere-se à vitória da equipe T.

**Validação-Cruzada**

Para evitar o *overfitting* no treinamento dos modelos, empregou-se a técnica de validação-cruzada *k-fold* estratificada durante a etapa de treinamento para cada classificador utilizado, fazendo-se uso também das métricas de acurácia média e desvio padrão para cada *fold* criado no dataset.

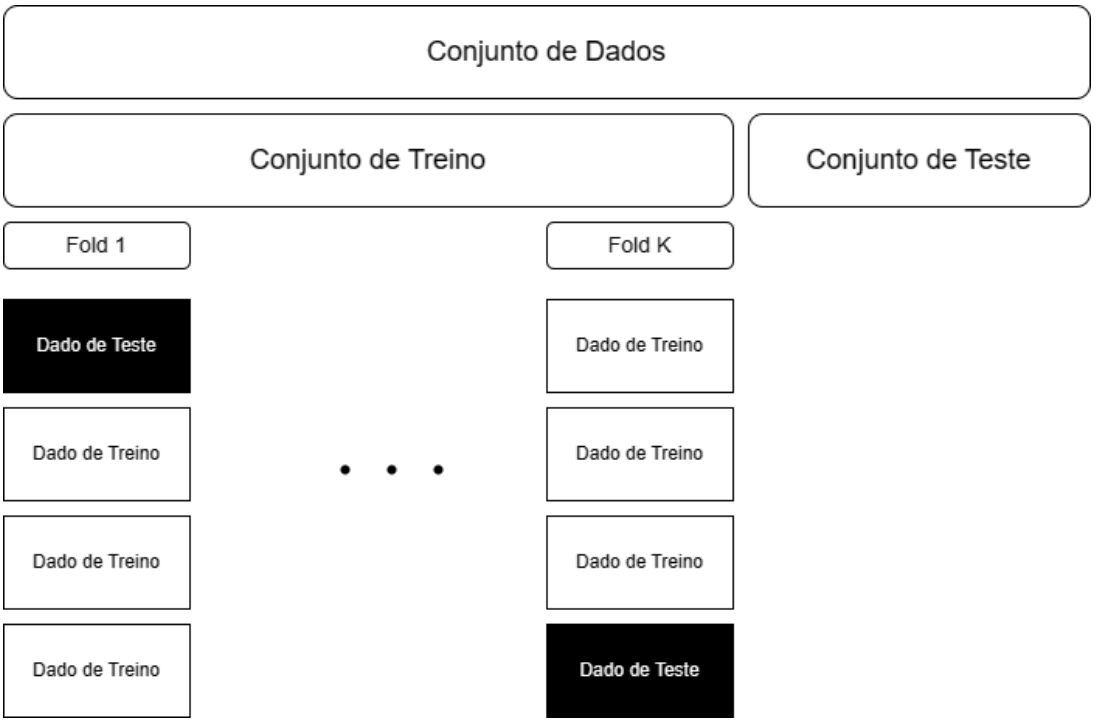


Figura 4. Processo de Validação-Cruzada *K-Fold*.

Fonte: Dados originais da pesquisa.

Mais especificamente, no processo de validação-cruzada, foram empregados 10 *folds* (isto é, 10 subdivisões) do conjunto de dados de treinamento de modo que cada subdivisão contenha sua própria separação entre amostras de treino e teste. O processo estratificado implica na divisão de amostras balanceada de cada classe para que cada um dos folds possuam a mesma distribuição de classes presentes no conjunto de dados original, evitando o treinamento de modelos com conjunto de dados desbalanceados e, consequentemente, o *overfitting* dos modelos. A validação cruzada *k-fold* estratificada foi escolhida visto que o processo lida bem com conjuntos de dados desbalanceados e garante que não haja um treinamento de modelos a partir de bancos de dados que possam estar enviesados contendo números muito díspares de classes. Além disso, computacionalmente, o processo utilizado se mostra muito menos custoso em conjuntos de dados grandes quando comparado à técnicas similares (e.g. Validação-Cruzada *Leave One Out)* (P. SWARNALATHA; TRIPATHY, 2012).

**Regressão Logística**

No contexto do estudo, a técnica de regressão logística binária pode ser empregada como um modelo de classificação utilizando as variáveis apresentadas na Tabela 1 como variáveis explicativas. O termo binário refere-se aos possíveis valores atribuídos à variável dependente, neste caso, a equipe vencedora de uma determinada rodada. A regressão logística possui como finalidade verificar a probabilidade de ocorrência de um evento (e consequentemente, do correspondente não evento associado) a partir da utilização de variáveis explicativas (ou preditoras). Dessa maneira, a probabilidade de ocorrência do evento ($p_i$) é dada por:

$$p_i = \frac{1}{1+e^{-Zi}} \quad (1)$$

Onde:

$$Z_i = \alpha + \beta_{1i} X_{1i} + ... + \beta_{ni} X_{ni} \quad (2)$$

De tal forma que $p_i$ pertence ao intervalo [0,1], onde um valor de $Z_i = 1$ implica na maior probabilidade possível de ocorrência do evento em questão (Fávero; Belfiore, 2017).

A estimação de cada parâmetro que fazem parte da composição do termo $Z_i$ é dada por diferentes funções, neste estudo aplicam-se os métodos de estimação por máxima verossimilhança (a partir do *solver* “liblinear”) onde itera-se por diferentes valores para os parâmetros e se busca maximizar o valor da função de máxima verossimilhança, dada por (Fávero; Belfiore, 2017):

$$LL = \sum_{i=1}^{n} \left\{ \left[ Y_i ln\left( \frac{e^{Zi}}{1+e^{Zi}} \right) \right] + \left[ (1 - Y_i) ln\left( \frac{1}{1+e^{Zi}} \right) \right] \right\} \quad (3)$$

No mesmo sentido, o processo de *Limited Memory BFGS* é aplicado como função para estimação dos parâmetros da Equação 2 e os resultados são comparados posteriormente com os resultados obtidos pelo método de máxima verossimilhança. O algoritmo em questão emprega um modelo baseado em memória limitada (reduzindo o custo computacional) do método BFGS. Este algoritmo busca a minimização local de uma dada função de custo a partir da estimação iterativa dos parâmetros. Sua implementação é relativamente mais complexa quando comparado ao método de máxima verossimilhança.

De maneira geral, dados valores iniciais para os parâmetros, uma matriz hessiana inicial $H_0$ é formulada e uma direção de busca $p_k$ é iniciada, visando encontrar valores que minimizem o gradiente da função de custo, $\nabla f$:

$$p_k = - H_k \nabla f(x_k) \quad (4)$$

Um valor de $\alpha_k$ é escolhido e computa-se um novo valor para *x*:

$$x_{k+1} = x_k + \alpha_k p_k \quad (5)$$

Uma nova matriz Hessiana é computada e seu valor é atualizado, até que um parâmetro de convergência do cálculo iterativo seja atingido (MORALES, 2002). Esta abordagem é empregada a partir da utilização do solver "lbfgs" da biblioteca SciKit-Learn.

Para as abordagens de regressão logística, foram empregados métodos de regularização nos modelos com o intuito de evitar o *overfitting* e penalizar variáveis com menor importância. Foram aplicadas, então as penalidades de Lasso (L1) e Ridge (L2), de modo que a primeira penaliza parâmetros menos influentes igualando-os à zero e a segunda atribui valores baixos (mas não igual à zero) à estes parâmetros (PEREIRA; BASTO; SILVA, 2016).

**Árvores de Decisão**

Diferentemente dos modelos lineares generalizados como a regressão logística, classificadores baseados em árvores de decisão tem como premissa, de maneira geral, o desenvolvimento de uma árvore composta por decisões com base no conjunto de variáveis explicativas empregadas no conjunto de dados.

A partir de uma condição inicial, a árvore divide-se em dois distintos caminhos baseado na resposta (verdadeira ou falsa), onde o processo é repetido até que se atinja uma divisão satisfatória e o resultado final da classificação é dado em um tipo de nó terminal conhecido como "folha". Assim, a partir das condições dadas, as subdivisões resultantes das possíveis respostas para cada nó formam uma estrutura hierárquica utilizada para novas predições.

Árvores de decisão fornecem modelos bastante interpretáveis e intuitivos, sendo possível compreender com facilidade o caminho seguido para que um determinado resultado seja alcançado (KINGSFORD; SALZBERG, 2008).

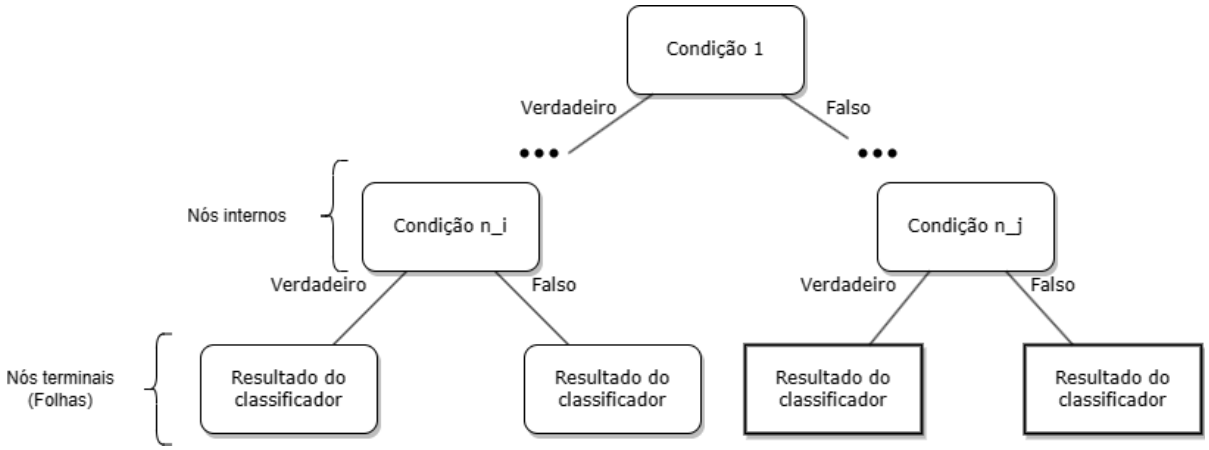


Figura 5. Estrutura genérica para árvores de decisão.

Fonte: Dados originais da pesquisa.

As melhores divisões de cada nó interno das árvores de decisões são calculadas a partir da impureza de Gini, buscando minimizar a impureza (isto é, a heterogeneidade) em cada nó até que a pureza seja atingida e as folhas (nós terminais) sejam atingidos:

$$I_g(p) = 1 - \sum_{i=1}^{j} p_i^2 \quad (6)$$

Neste sentido, existem diversos parâmetros importantes para a operação de uma árvore de decisão, dentre os mais importantes, podem ser citados:

- Profundidade máxima para a árvore;
- Número mínimo de observações presentes em cada nó.

A determinação dos parâmetros e eficiência da árvore são discutidos na seção Resultados e Discussão deste estudo.

**Random Forests**

O método de Random Forests (RF), classificado como um método de *ensemble* e inicialmente introduzido por Breiman (BREIMAN, 2001), emprega um conjunto de árvores de decisão de modo que cada árvore contenha um subconjunto aleatório das variáveis explicativas originais do conjunto de dados (CUTLER; CUTLER; STEVENS, 2012).

Baseado nas árvores que compõem o modelo, espera-se obter uma função para a predição da variável dependente, onde, busca-se minimizar o valor esperado da função de perda. Por meio de um método de votação, as árvores de decisão que compõem o RF

contribuem com sua classificação individual e a classe que conter mais votos como resultado das árvores é escolhida como classificação final do RF.

O método de RF é capaz de lidar com problemas de classificação e regressão, além de possuírem boa performance em problemas com alta dimensionalidade (CUTLER; CUTLER; STEVENS, 2012).

Para o algoritmo de Random Forests, podem ser definidos diversos valores para diferentes hiperparâmetros do método. Dentre os mais importantes, podem ser citados:

- Número mínimo de amostras por divisão;
- Números mínimo de amostras por folha;
- Número de estimadores;
- Critério utilizado para medir a qualidade dos *splits* para cada nó das árvores.

## Resultados e Discussão

### Análise de Correlação

Em uma análise inicial do conjunto de dados, a partir do procedimento de extração e tratamento de dados descrito na seção de Material e Métodos, foi realizada uma matriz de correlação de Pearson para verificar o relacionamento das variáveis explicativas quantitativas presente no conjunto de dados:

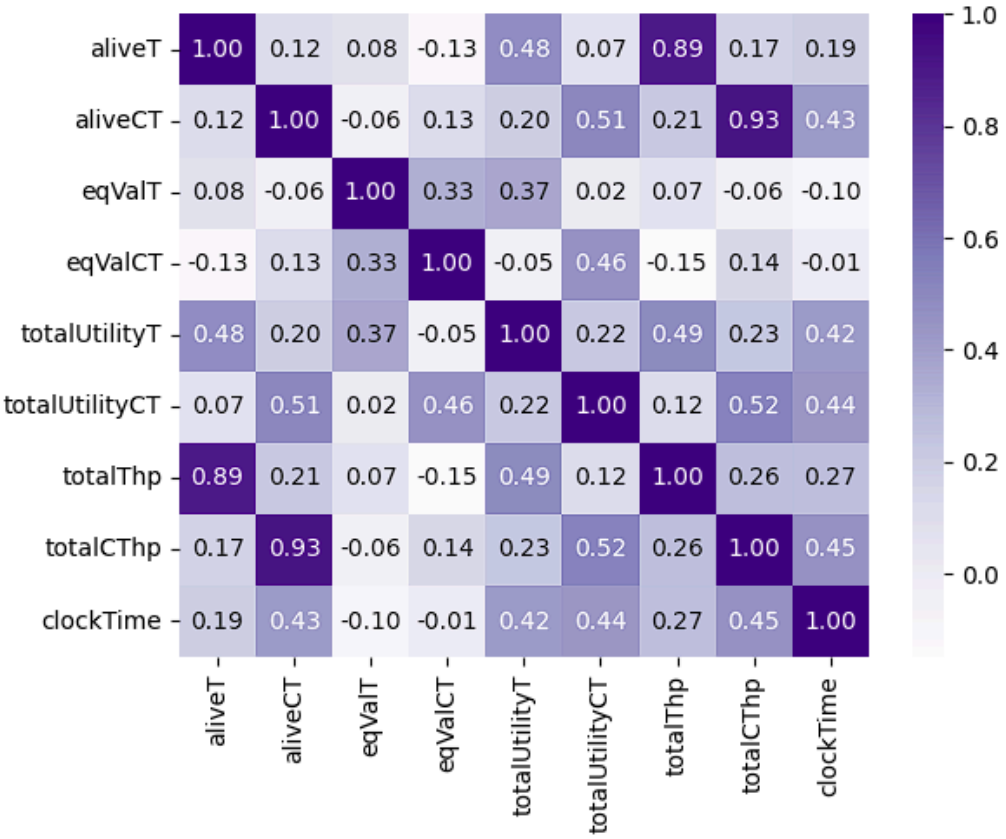


Figura 6. Matriz de correlação de Pearson para o conjunto de dados utilizado.

Fonte: Dados originais da pesquisa.

Na Figura 6, é possível observar uma correlação extremamente forte entre os pares de variáveis “aliveCT” e “totalCTHP”, e “aliveT” e “totalTHP”, que representam, respectivamente, o número de jogadores ativos e a soma dos pontos de vida de cada jogador das equipes CT e T, com valores iguais a 0,93 e 0,89. Isto se deve ao fato em que, caso um jogador seja eliminado, por exemplo, os pontos de vida de um determinado jogador são reduzidos a zero e consequentemente, a variável que representa a soma dos pontos de

vida de uma determinada equipe são reduzidas nas mesma unidades, juntamente com a remoção de uma unidade na quantidade de jogadores ativos.

Além disso, a tendência geral observada na matriz é de uma correlação positiva entre as variáveis que fazem referência à uma mesma equipe, por exemplo, valores iguais a 0,48 e 0,51 são atribuídos aos pares de variáveis referentes aos jogadores ativos e quantidade de itens de uma mesma equipe (para os lados T e CT, respectivamente).

Apesar da existência de alguns valores negativos de correlação, não foi observada nenhuma relação negativa forte entre um par de variáveis.

**Regressão Logística**

Durante o treinamento do modelo, foi empregado o procedimento de *Exhaustive Feature Selection* (EFS) para a seleção do melhor subconjunto de variáveis explicativas para compor o modelo final. O procedimento baseia-se no treinamento recursivo do modelo com todos os subconjuntos possíveis de variáveis explicativas visando obter a melhor acurácia possível.

Definiu-se o número mínimo de variáveis para compor o modelo igual a 5 e o número máximo destas variáveis igual a 9. Apesar do processo ser computacionalmente custoso, a técnica mostrou-se viável para este estudo devido ao baixo número de variáveis e ao tempo de treinamento baixo dos modelos de regressão logística (JOVIC; BRKIC; BOGUNOVIC, 2015).

Após a conclusão do processo, o procedimento excluiu a variável referente ao tempo decorrido da rodada do conjunto de variáveis, entretanto, não ocorreu mudança significativa na acurácia do modelo após o treinamento com o novo conjunto, optando-se por manter o conjunto original de variáveis.

A Tabela a seguir apresenta os dados referentes ao treinamento e validação dos modelos empregando a validação cruzada (10-*fold*) nos modelos de regressão logística para um *cutoff* de 0,5:

Tabela 2. Resultados obtidos a partir do treinamento dos modelos de Regressão Logística.

| Solver | Acurácia Média (*10-fold* CV) | Desvio Padrão | Acurácia (conjunto de teste) | Precisão | Recall |
|---|---|---|---|---|---|
| lbfgs (L2) | 0,835 | 0,003 | 0,832 | 0,819 | 0,850 |
| liblinear (L1) | 0,834 | 0,003 | 0,833 | 0,820 | 0,852 |

Fonte: Dados originais da pesquisa.

Apesar da natureza distinta da obtenção dos parâmetros para cada *solver*, observou-se uma performance muito similar de ambos os classificadores quando aplicados aos mesmo conjuntos de treino e teste, com acurácia próxima a 83%.

A aplicação de ambos os métodos de regularização de Ridge e Lasso, (L2 e L1, respectivamente) apresentaram resultados satisfatórios e similares comparando as métricas de acurácia, precisão e *recall* dos modelos. A matriz de confusão dos modelos apresenta uma visualização dos resultados acima discutidos.

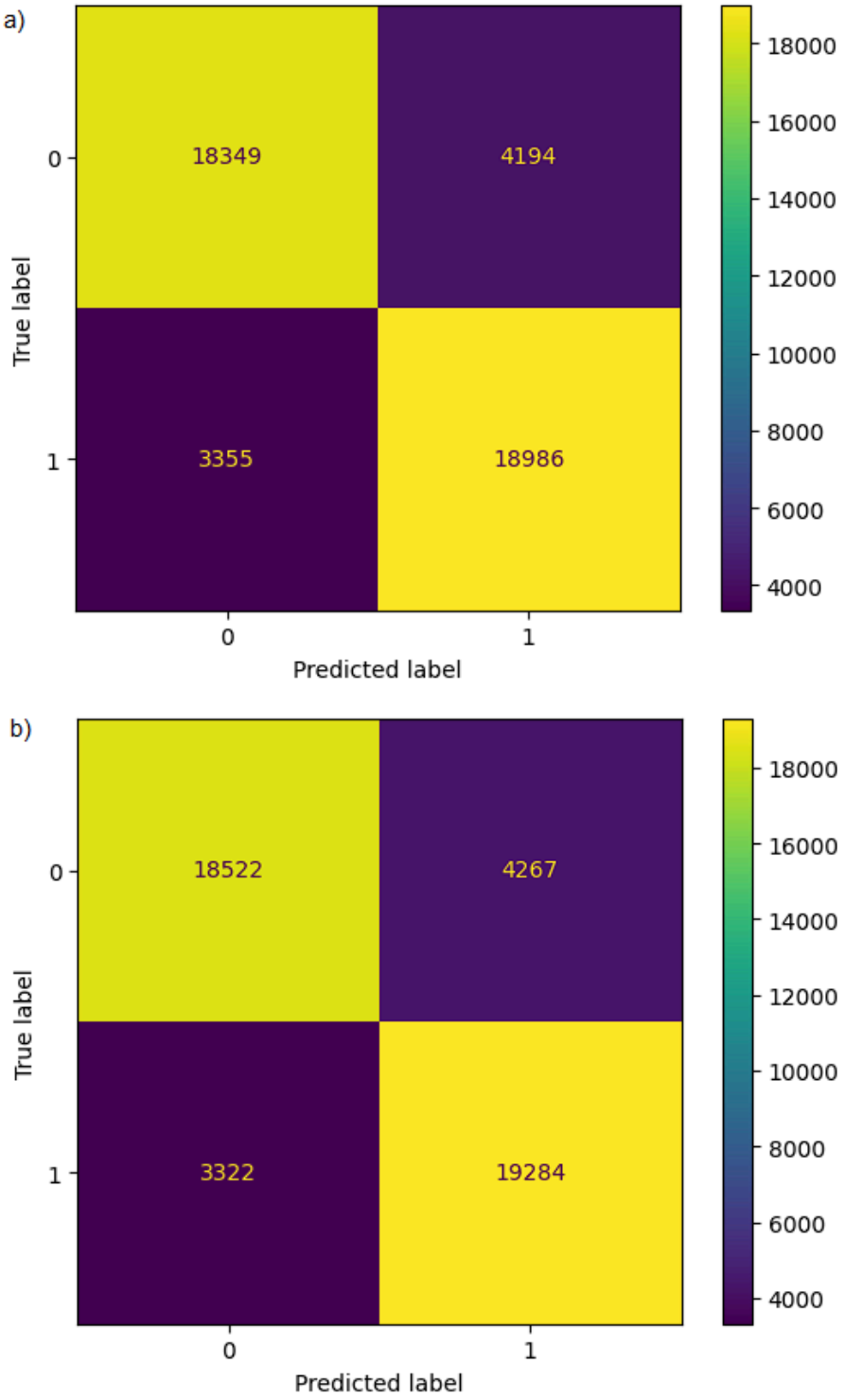


Figura 7. Matriz de confusão para os modelos de regressão logística utilizando os *solvers* a) lbfgs e b) liblinear.

Fonte: Dados originais da pesquisa.

A Figura abaixo apresenta a curva ROC e o cálculo da métrica AUC para cada um dos modelos discutidos até o momento.

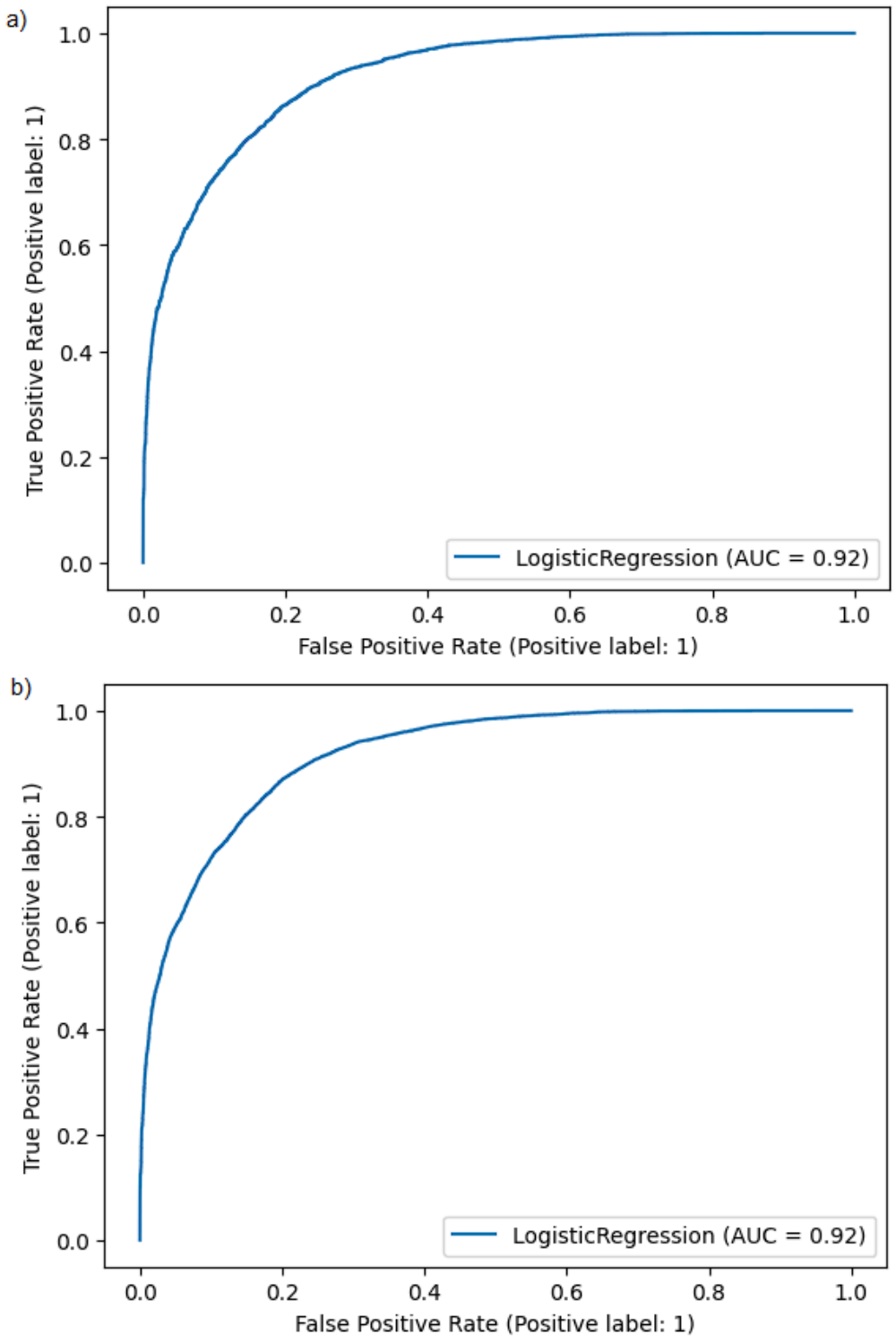


Figura 8. Curvas ROC para os modelos de regressão logística utilizando os *solvers* a) lbfgs e b) liblinear.

Fonte: Dados originais da pesquisa.

Conforme esperado, os modelos em questão apresentaram valores de AUC similares (0.92) e formatos da curva ROC similares, devido à similaridade das métricas apresentadas na Tabela 2. Em ambos os modelos, os valores referentes à média de precisão empregando a validação-cruzada no conjunto de dados de treino e à precisão apresentam valores bastante similares resultando em um indicativo da ausência de *overfitting* dos modelos de regressão logística, devido à técnicas empregadas como o método RUS, à validação-cruzada estratificada e a utilização dos regularizadores de Ridge e Lasso.

**Árvores de Decisão**

Novamente, conforme aplicado para os modelos de regressão logística, empregou-se a técnica de EFS com a finalidade de obtenção de um subconjunto de variáveis explicativas com intuito de encontrar o melhor modelo, utilizando a acurácia deste como métrica para a seleção final. Neste sentido, o resultado final da aplicação do EFS selecionou apenas as seguintes variáveis como o subconjunto de variáveis explicativas que atingem a melhor acurácia:

- Número de jogadores ativos da equipe T (aliveT)
- Valor total de equipamento da equipe T (eqValT)
- Valor total de equipamento da equipe CT (eqValCT)
- Soma de pontos de vida da equipe T (totalThp)
- Soma de pontos de vida da equipe CT(totalCThp)

Entretanto, apesar de encontrado um melhor subconjunto, a acurácia final obtida foi igual à acurácia referente à presença de todas as variáveis explicativas escolhidas inicialmente (mostradas na Tabela 1). Dessa forma, optou-se pela presença do número original de variáveis, já que não houve melhora significativa observada na acurácia do modelo.

Para a seleção do melhor hiperparâmetro relativo à profundidade máxima da árvore, realizou-se cálculos iterativos com o intuito de verificar a acurácia de cada modelo variando a profundidade de cada árvore:

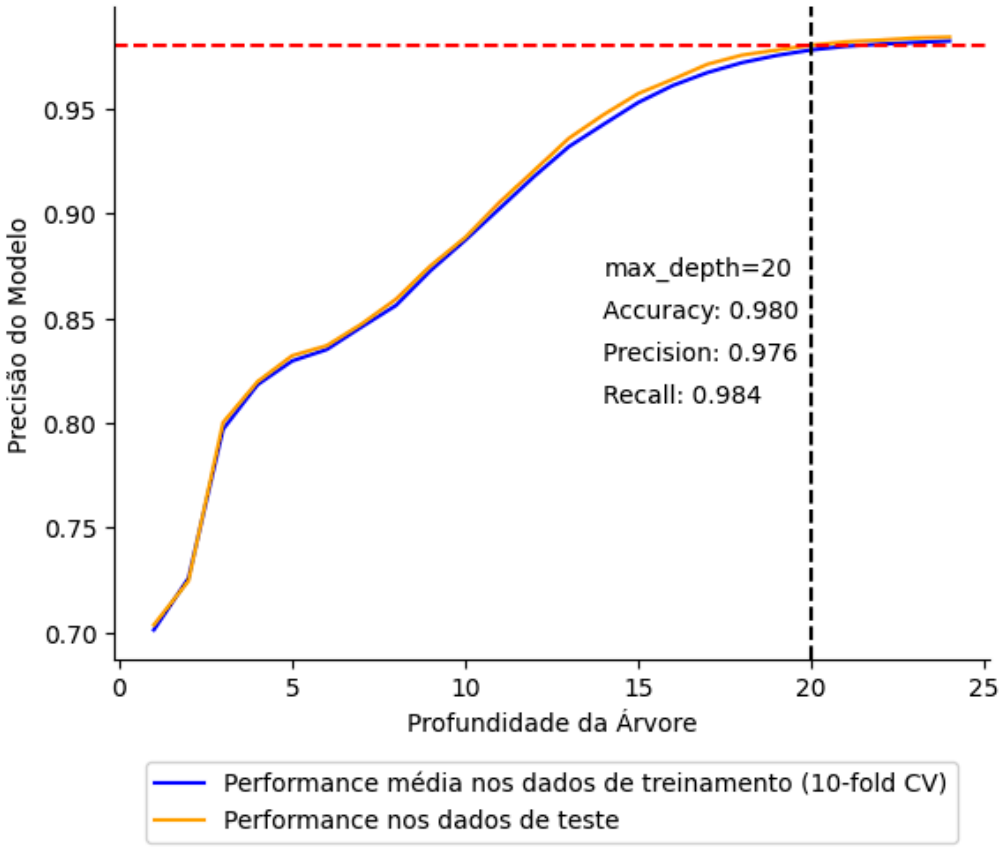


Figura 9. Acurácia do modelo de árvores de decisão em função da profundidade máxima da árvore.

Fonte: Dados originais da pesquisa.

A análise da Figura acima mostra uma evolução clara da acurácia do modelo em função da profundidade máxima da árvore. Nota-se que, ao evoluir o hiperparâmetro entre os valores 1 e 5, a acurácia do modelo é aprimorada em cerca de 13%. Após esse ponto, atinge-se um processo aproximadamente constante de evolução até um número de profundidade máxima igual à 15.

Por fim, selecionou-se o valor de profundidade máxima igual a 20 para o desenvolvimento do modelo final visto que não há uma mudança significativa da mudança do modelo, onde a acurácia do atinge uma performance próxima de 98%. Observa-se também valores similares de acurácia ao longo de toda a curva nos dados de treinamento empregando a validação-cruzada (curva azul) e no conjunto de dados de teste (curva laranja), o que demonstra uma ausência de *overfitting* em todos os modelos gerados.

Ressalta-se ainda, que performances satisfatórias são atingidas para modelos menos robustos com profundidades menores, por exemplo, uma performance de aproximadamente 88% foi atingida para uma profundidade máxima de árvore igual a 10.

A tabela abaixo apresenta os dados obtidos referente ao treinamento do modelo final (profundidade máxima igual a 20), incluindo dados referentes à validação cruzada:

Tabela 3. Resultados obtidos a partir do treinamento dos modelos de Árvores de Decisão.

| Modelo | Acurácia Média (*10-fold* CV) | Desvio Padrão | Acurácia (conjunto de teste) | Precisão | Recall |
|---|---|---|---|---|---|
| Árvore de Decisão | 0,978 | 0,002 | 0,980 | 0,976 | 0,985 |

Fonte: Dados originais da pesquisa.

Os resultados da validação-cruzada apresentam um desvio padrão de 2%, indicando uma baixa variação dos resultados ao longo de cada iteração do processo. O modelo também mostra um bom balanceamento entre os valores de precisão e *recall*. Abaixo, apresentam-se a matriz de confusão e a curva ROC do modelo final no conjunto de dados de treino.

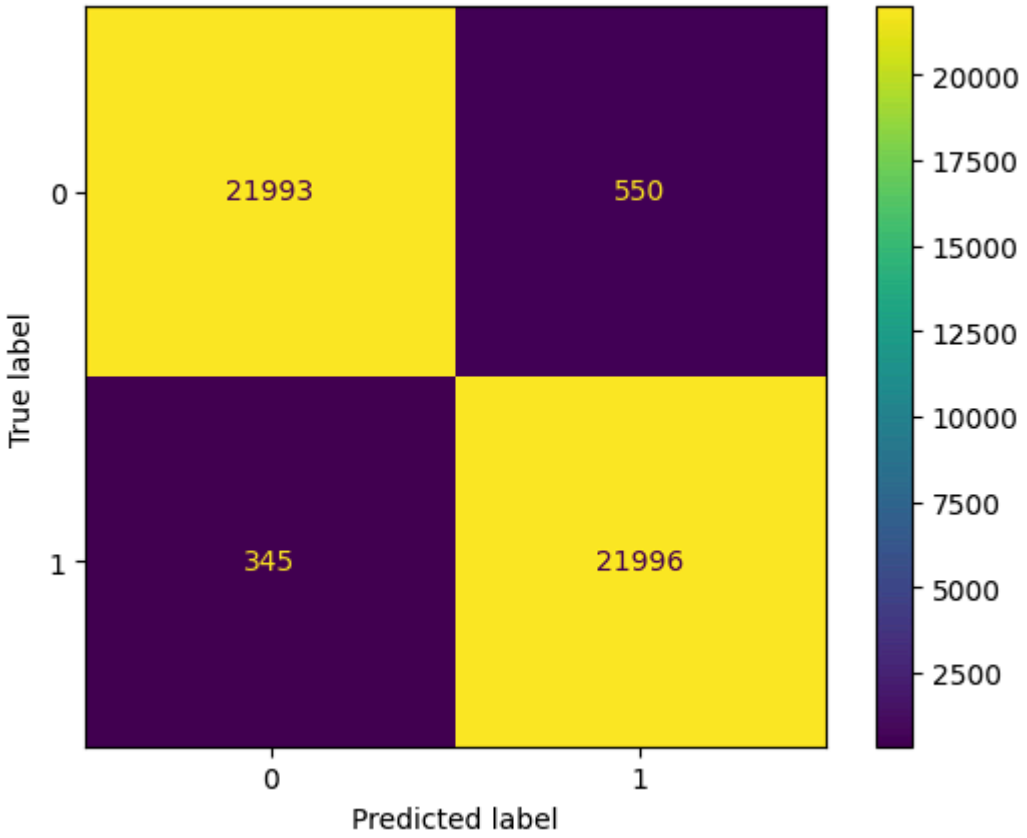


Figura 10. Matriz de confusão do modelo final baseado em árvores de decisão.

Fonte: Dados originais da pesquisa.

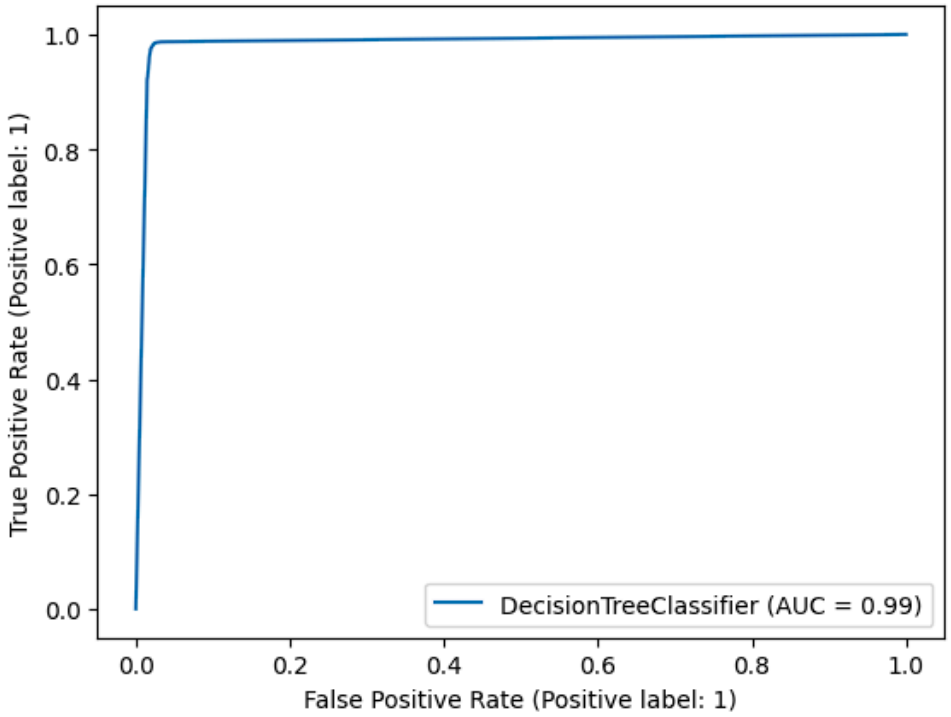

Figura 11.Curva ROC do modelo final baseado em árvores de decisão.

Fonte: Dados originais da pesquisa.

Na matriz de confusão, observa-se uma taxa de erro muito menor quando comparado aos modelos de regressão logística, enquanto observa-se uma distribuição praticamente igualitária entre os acertos relativos à equipe vencedora da rodada.

A métrica AUC, por sua vez, indica uma excelente performance geral do modelo dada a relação entre a taxa de verdadeiros e falsos positivos, com um valor de AUC igual a 0,99. A performance do modelo em questão superou a acurácia dos modelos de regressão logística em aproximadamente 10% mostrando ainda um bom balanço entre as classificações corretas e os erros cometidos pelo modelo.

**Random Forests**

Para o classificador baseado no método de random forests, de maneira semelhante ao que foi realizado com o algoritmo de árvore de decisão, empregou novamente a visualização baseada na profundidade máxima das árvores que compõem o classificador.

Primeiramente, definiram-se os seguintes hiperparâmetros para a realização do estudo iterativo:

- Número mínimo de amostras por divisão: 2
- Números mínimo de amostras por folha: 1
- Número de estimadores: 100
- Critério: Gini

Dessa maneira, os resultados da técnica discutida acima são apresentados abaixo:

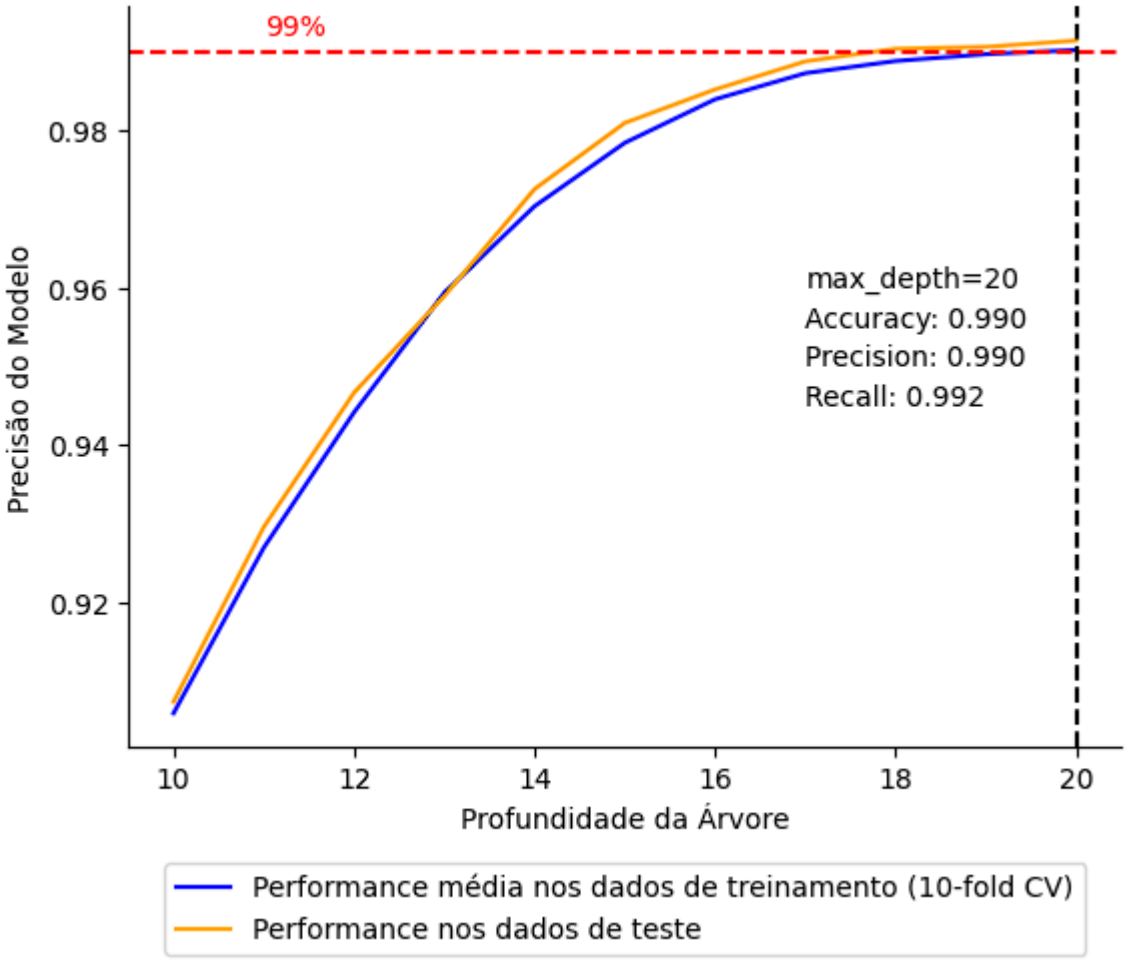


Figura 12. Acurácia do modelo de árvores de decisão em função da profundidade máxima da árvore.

Fonte: Dados originais da pesquisa.

De maneira similar ao observado com o modelo anterior, o algoritmo de Random Forests apresentou uma crescente evolução na acurácia do modelo conforme o número de profundidade máxima permitida para as árvores. Para o menor valor configurado (profundidade máxima igual a 10), observou-se uma performance satisfatória, atingindo uma acurácia de aproximadamente 90%.

A análise dos dados permite inferir que a acurácia de cada modelo era aprimorada em pelo menos 2% a cada 2 unidades acrescentadas no valor do hiperparâmetro até um valor de profundidade igual a 16. Após esse ponto, a taxa em que a acurácia era aprimorada foi reduzida até que se atingisse a acurácia máxima obtida de 99% para uma profundidade

máxima igual  a 20. Neste sentido, selecionou-se este valor para compor o modelo final de modo que os resultados são apresentados na Tabela a seguir:

Tabela 4. Resultados obtidos a partir do treinamento dos modelos de Random Forests.

| Modelo | Acurácia Média (*10-fold* CV) | Desvio Padrão | Acurácia (conjunto de teste) | Precisão | Recall |
|---|---|---|---|---|---|
| Árvore de Decisão | 0,990 | 0,001 | 0,991 | 0,990 | 0,992 |

Fonte: Dados originais da pesquisa.

Durante a validação-cruzada, um desvio padrão igual a 1% foi apresentado indicando uma constância nas decisões do algoritmo para cada *fold* executado. Os melhores valores de acurácia, precisão e *recall* foram obtidos em comparação às técnicas empregadas anteriormente.

O método *built-in "Feature Importances"* da biblioteca foi empregada para avaliar quais variáveis são mais importantes (relativamente) para a decisão final do modelo em questão:

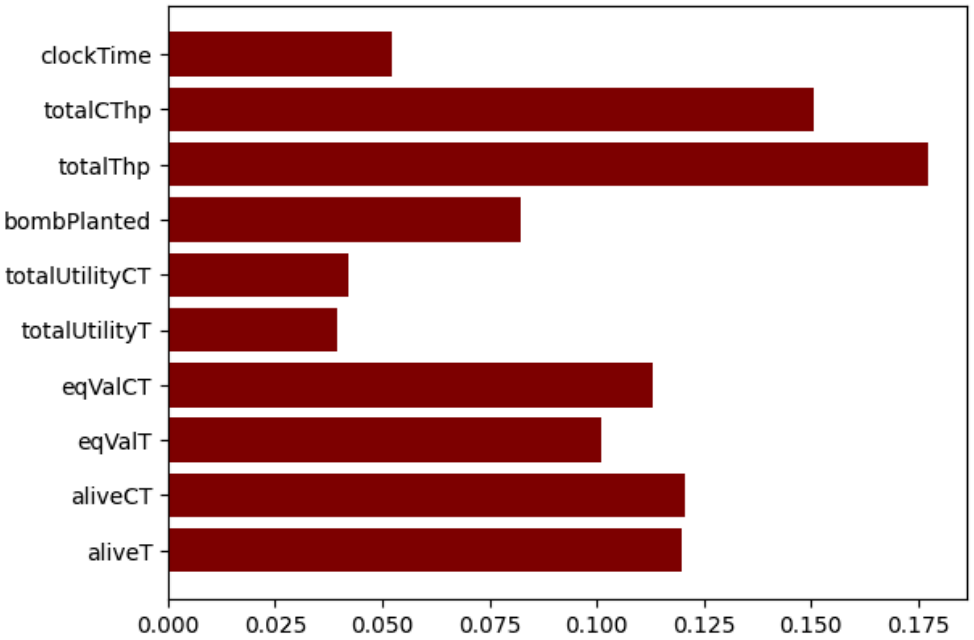


Figura 13. Métrica de *Feature Importances* para o modelo de Random Forests.

Fonte: Dados originais da pesquisa.

Observa-se que a soma total dos pontos de vida dos jogadores e a quantidade total de jogadores ativos são as variáveis mais importantes na tomada de decisão das árvores que compõem o modelo e, em contrapartida, a soma do valor de equipamentos dos jogadores em ambas equipes apresentam-se como variáveis que resultam em uma menor redução da impureza para o classificador.

Abaixo, apresentam-se a matriz de confusão e  a curva ROC do modelo em questão:

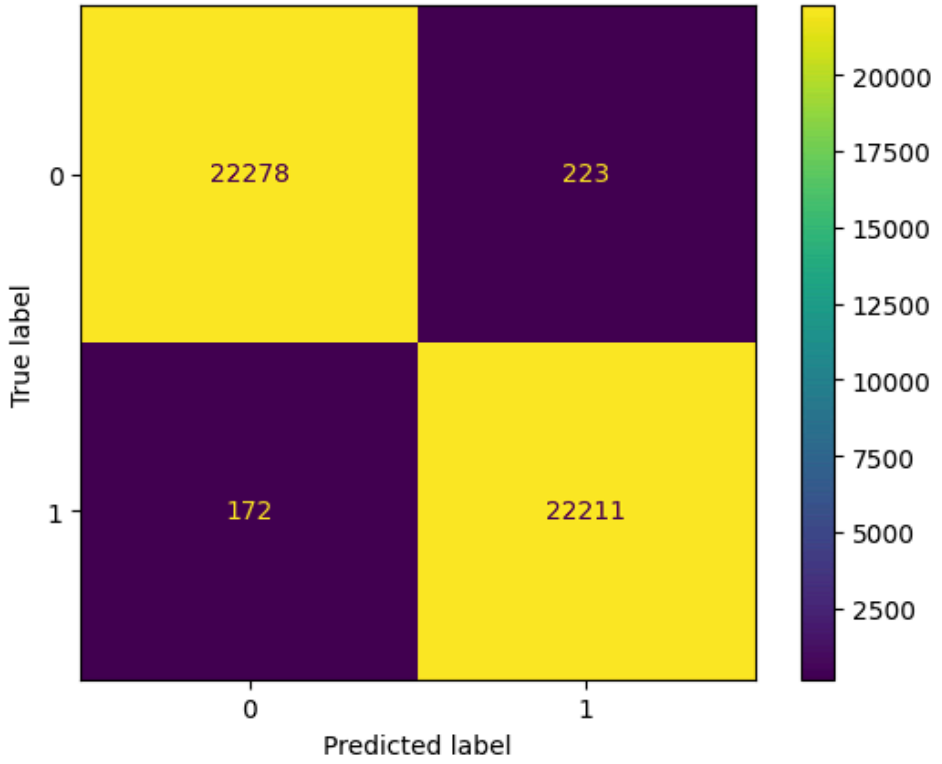


Figura 14. Matriz de confusão do modelo final baseado em Random Forests.

Fonte: Dados originais da pesquisa.

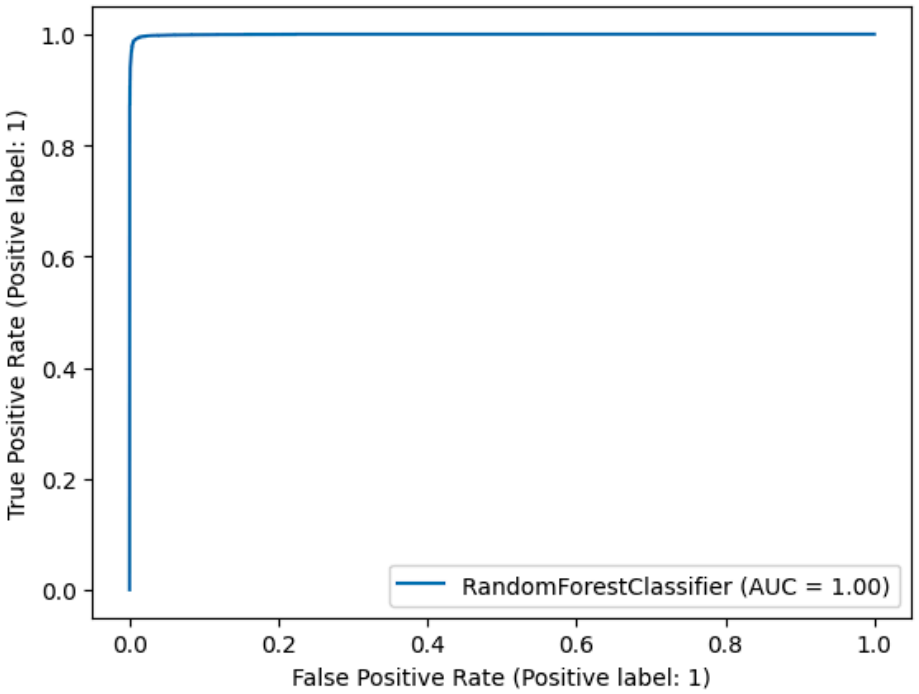


Figura 15. Curva ROC do modelo final baseado em árvores de decisão.

Fonte: Dados originais da pesquisa.

A análise das figuras acima apresenta uma excelente performance do modelo, quando comparado aos modelos prévios. Neste sentido, a matriz de confusão indica um ótimo balanceamento entre os acertos e erros do modelo, sendo que estes compõem menos de 1% do conjunto de treino, totalizando um número bruto de 395 classificações errôneas contra 44489 classificações corretas.

Vale notar, a partir da análise da métrica AUC, que uma valor igual a 1 não indica um classificador perfeito, visto que a métrica apresenta uma relação entre a sensibilidade e especificidade (1 - especificidade) do modelo.

Finalmente, a escolha dos melhores hiperparâmetros juntamente com as técnicas empregadas de limpeza de dados, balanceamento do conjunto de dados, validação-cruzada e o modelo de random forests permitiram o desenvolvimento de um robusto classificador com a maior performance entre os modelo observados, em troca de um tempo de treinamento mais elevado.

## Conclusões

A partir da realização deste estudo, foi possível realizar a construção de diferentes classificadores para a predição de equipes vencedoras de rodadas distintas no jogo eletrônico Counter-Strike: Global Offensive. Todos os classificadores treinados apresentaram acurácia maior que 80% em seus respectivos conjuntos de dados de treino e teste. Neste contexto, apesar de modelos baseados em Random Forests apresentarem maior acurácia quando comparados aos modelos de regressão logística e árvores de decisão, estes apresentaram boa performance e, principalmente, um tempo de treinamento consideravelmente menor levando em conta o grande volume de dados do conjunto de dados.

Conforme o esperado, os modelos apresentados apresentam eficiência maior em momentos mais tardios de uma rodada devido à disparidade nas variáveis apresentadas, facilitando a conclusão da equipe que possui maior vantagem em um determinado momento. O balanceamento inicial das classes durante a etapa de análise exploratória de dados no conjunto de dados, assim como o emprego de validação-cruzada estratificada, mostraram-se eficientes como medidas para a prevenção de *overfitting* nos modelos.

Como perspectivas futuras, a eficiência dos modelos pode ser aprimorada a partir da utilização de dados espaciais relativos à posição de cada jogador em determinado instante de uma rodada, embora estudos adicionais sejam necessários para avaliar a viabilidade da introdução destas novas variáveis aos modelos.

Ademais, ressalta-se que a utilização de dados referentes à partidas passadas de equipes profissionais específicas e seus jogadores, se incorporadas aos modelos, podem beneficiar a performance dos algoritmos, embora o escopo do presente trabalho seja propor uma análise indiscriminante quantos à equipes profissionais e jogadores específicos, restringindo-se apenas a eventos que ocorram durante uma partida, não levando em conta históricos de jogos disputados no passado.

O presente estudo visou também contribuir com a escassa literatura e discussão (principalmente em língua portuguesa) quanto ao uso de técnicas estatísticas e computacionais no contexto de competições, mais especificamente em esports,

esperando-se que as discussões realizadas agreguem como base para novos recursos e métodos aplicados para esta finalidade.

**Referências**


BREIMAN, L. Random Forests. Machine Learning, v. 45, n. 1, p. 5–32, 2001.

COLLIS, W. The Book of Esports. [s.l.] Rosetta Books, 2020.

CUTLER, A.; CUTLER, D. R.; STEVENS, J. R. Random Forests. Ensemble Machine Learning, p. 157–175, 2012.

Fávero, L.P.; Belfiore, P. 2017. Manual de Análise de Dados - Estatística e Modelagem Multivariada com Excel®, SPSS® e Stata®. 1ed. Elsevier Editora, Rio de Janeiro, RJ, Brasil. Disponível em: <https://integrada.minhabiblioteca.com.br/books/9788595155602>. Acesso em: 22 nov. 2021.

HLTV.org - The home of competitive Counter-Strike, 2023. Disponível em: <https://hltv.org>. Acesso em: 26 de outubro de 2023.

JOVIC, A.; BRKIC, K.; BOGUNOVIC, N. A review of feature selection methods with applications. 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), maio 2015.

KHROMOV, N. et al. Esports Athletes and Players: A Comparative Study. IEEE Pervasive Computing, v. 18, n. 3, p. 31–39, 1 jul. 2019.

KINGSFORD, C.; SALZBERG, S. L. What are decision trees? Nature Biotechnology, v. 26, n. 9, p. 1011–1013, set. 2008.

MAKAROV, I. et al. Predicting Winning Team and Probabilistic Ratings in “Dota 2” and “Counter-Strike: Global Offensive” Video Games. Lecture Notes in Computer Science, p. 183–196, 21 dez. 2017.

MCCOMB, D. G. Sports in World History. [s.l.] Psychology Press, 2004.

MCDONALD, G. C. Ridge regression. Wiley Interdisciplinary Reviews: Computational Statistics, v. 1, n. 1, p. 93–100, jul. 2009.

MORALES, J. L. A numerical study of limited memory BFGS methods. Applied Mathematics Letters, v. 15, n. 4, p. 481–487, maio 2002.

MORGULEV, E.; AZAR, O. H.; LIDOR, R. Sports analytics and the big-data era. International Journal of Data Science and Analytics, v. 5, n. 4, p. 213–222, 9 jan. 2018.

PEREIRA, J. M.; BASTO, M.; SILVA, A. F. DA. The Logistic Lasso and Ridge Regression in Predicting Corporate Failure. Procedia Economics and Finance, v. 39, p. 634–641, 2016.
P. SWARNALATHA; TRIPATHY, B. K. Evaluation of Classifier Models Using Stratified Tenfold Cross Validation Techniques. p. 680–690, 1 jan. 2012.

REITMAN, J. G. et al. Esports Research: A Literature Review. Games and Culture, v. 15, n. 1, p. 155541201984089, 15 abr. 2019.

RICHTER, C.; O'REILLY, M.; DELAHUNT, E. Machine learning in sports science: challenges and opportunities. Sports Biomechanics, p. 1–7, 20 abr. 2021.

SARANAVAN, R., POTHUILA S. "A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification". 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, p. 945–49. IEEE Xplore, doi.org/10.1109/ICCONS.2018.8663155.

SCHOLZ, T. M. A Short History of eSports and Management. eSports is Business, p. 17–41, 2019.

SCIKIT-LEARN, 2023. Disponível em: <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>. Acesso em 06 de junho de 2023.

WANG, W. Returning for skill or popularity? The demand for esports match replay. International Journal of Sports Marketing and Sponsorship, 2 dez. 2022.

XENOPOULOS, P.; FREEMAN, W. R.; SILVA, C. Analyzing the Differences between Professional and Amateur Esports through Win Probability. Proceedings of the ACM Web Conference 2022, 25 abr. 2022.