INSTITUTO UNIVERSITÁRIO DE LISBOA

PROJETO APLICADO A CIÊNCIA DE DADOS I

Previsão do número de *sets* para conclusão de um jogo de ténis profissional na Bélgica

REALIZADO POR:

Carolina Morgado - 123794

Diogo Nobre - 123441

Francisco Rosa - 123418

Inês Silva - 123407

Marcos Mestre - 123436

Turma CDB2

2024/2025

ÍNDICE

1. Business Understanding

No âmbito da unidade curricular Projeto Aplicado a Ciência de Dados I, foi proposto o desenvolvimento de um modelo capaz de prever o número de *sets* necessários para a conclusão de um jogo de ténis profissional à melhor de 3.

O foco deste trabalho incide somente sobre os jogos realizados na Bélgica, utilizando um conjunto de variáveis relacionadas tanto com os jogadores como com os jogos.

Para garantir uma abordagem estruturada e eficaz, seguimos a metodologia *CRISP-DM*, que orientou todas as fases do projeto.

1.1. ATP, regras do ténis e o ténis na Bélgica

A *Association of Tennis Professionals* (ATP) foi fundada em 1972 para proteger os interesses dos jogadores. A ATP é um órgão que gere o circuito masculino profissional de ténis e que atualmente organiza o ATP *Tour,* onde estão incluídos vários níveis de torneios: *Grand Slams*, ATP *Masters* 1000, ATP 500, ATP 250, ATP *Finals*. O *ranking* ATP reflete a performance dos jogadores nos vários jogos e torneios, e é atualizado semanalmente.

Para ganhar um *set*, um jogador precisa de vencer, pelo menos, seis jogos e de ter uma vantagem de, pelo menos, dois jogos sobre o adversário.

Um jogo (*game*) começa no 0 e à medida que os jogadores fazem pontos aumenta para 15, 30, 40 e por fim para o *gamepoint*. Caso haja um empate (*deuce*), ou seja, ambos os jogadores contenham 40 pontos, é necessário que um deles ganhe dois pontos seguidos para conquistar o jogo.

Os *sets* são compostos por jogos e, por norma, são à melhor de 3. No entanto, para alguns torneios mais prestigiados, nas fases finais os *sets* são à melhor de 5.

Quando o *set* empata em 6-6, temos um *tie-break* onde se joga até aos 7 pontos, com diferença de 2. Em casos específicos como os jogos a pares e em alguns torneios, no terceiro *set,* joga-se um super *tie-break* até aos 10 pontos.

A Bélgica não é um país que costuma receber os mais emblemáticos torneios de ténis, no entanto, existem edições que se destacam a nível internacional, como o *European Open Antwerp* ou o Torneio *Ethias*, em Mons.

1.2. Aplicações práticas do modelo preditivo

O desenvolvimento do modelo preditivo proposto poderá ter relevância em dois contextos de utilização: as apostas desportivas e o apoio estratégico a treinadores e jogadores.

No setor das apostas, a previsão exata do *score* de um jogo pode contribuir para diversas oportunidades de negócio. Por exemplo, os apostadores podem utilizar as previsões do modelo para apostar com mais confiança em mercados de "*set betting*". Além disso, é possível usar as previsões para compor apostas múltiplas, combinando vários resultados numa só aposta.

Em vez de fazer escolhas baseadas apenas em palpites, os apostadores podem concentrar-se nas partidas que apresentam maior previsibilidade, aumentando assim as suas hipóteses de sucesso. Também poderá ser possível identificar padrões através do modelo, como jogadores que se destacam em determinados pisos ou encontros com elevada probabilidade de incluir um *tie-break*, e usar essa informação para orientar os apostadores.

Por outro lado, o modelo pode também ser útil para as próprias casas de apostas. As previsões geradas poderão servir para ajustar as probabilidades em tempo real (durante eventos ao vivo), reduzindo o risco de perdas significativas. Esta capacidade de gestão de risco permite que as casas de apostas se protejam em cenários de maior incerteza, ao mesmo tempo que continuam a oferecer oportunidades interessantes aos apostadores.

Para além do contexto das apostas, este modelo pode assumir um papel relevante na definição da estratégia de jogo por parte de treinadores e jogadores. Assim, antes de cada encontro, as previsões geradas podem apoiar a análise detalhada dos pontos fortes e fracos de cada atleta, permitindo ao treinador ajustar a abordagem tática de forma mais precisa, tendo em conta as características do adversário.

Adicionalmente, ao antecipar o número provável de *sets*, o modelo pode contribuir para uma melhor gestão do esforço físico, uma vez que jogos mais longos tendem a ser mais exigentes do ponto de vista físico, sendo essencial planear em função do desgaste esperado.

2. Data Understanding

A fase da compreensão dos dados tem como objetivo a familiarização com o *dataset*, nomeadamente, com a sua estrutura, qualidade e significado das variáveis. Envolve a identificação dos tipos de dados, a verificação de valores omissos e inconsistentes, a compreensão das relações entre variáveis e a avaliação da relevância global do conjunto de dados para um objetivo específico. Esta análise inicial é crucial para assegurar a qualidade dos dados e fundamentar as etapas seguintes do projeto.

O *dataset* original continha 1.308.835 observações e 16 variáveis. Como o foco do nosso trabalho é exclusivamente sobre os jogos de ténis realizados na Bélgica, apenas 11.005 desses registos foram extraídos e considerados relevantes.

Na tabela abaixo pode ser visualizado o nome, o tipo e a descrição de todas as variáveis presentes no *dataset* e ainda, o número de valores omissos e as estatísticas descritivas (adequadas ao tipo de variável) de cada uma.

Nome	Tipo	Descrição	Nº val. omissos	Estatística Descritivas
_id	—	Identificador único de cada partida	—	—
PlayerName	Categórica	Nome do jogador de referência da partida	0	Valores únicos: 1691
Born	Categórica	Local de nascimento do jogador de referência da partida (pode conter a cidade, o país ou ambos)	2754	Valores únicos: 824
Height	Numérica	Altura do jogador de referência da partida, em centímetros	2681	Média: 181.84 Mediana: 185 Mínimo: 0 Máximo: 211
Hand	Categórica	Mão dominante e estilo de *backhand* do jogador de referência da partida	1727	Valores únicos: 7
LinkPlayer	—	URL do perfil do jogador de referência da partida do site ATP	0	Valores únicos: 1691
Tournament	Categórica	Nome do torneio	0	Valores únicos: 74
Location	Categórica	Localização do torneio (pode conter a cidade, o país ou ambos)	0	Valores únicos: 37
Date	Qualitativa Ordinal	Intervalo de datas do torneio	0	—
Ground	Categórica	Tipo de piso da partida	0	Valores únicos: 3

Prize	Numérica	Valor monetário do prémio e respetiva moeda	263	Média: 100539.35 Mediana: 15000.0 Mínimo: 10000.0 Máximo: 1100000.0
GameRound	Categórica	Fase do torneio em que foi jogada a partida	0	Valores únicos: 10
GameRank	Numérica	Ranking ATP do oponente no momento do jogo	967	Média: 591.89 Mediana: 496 Mínimo: 1 Máximo: 2236
Oponent	Categórica	Nome do adversário da partida	0	Valores únicos: 1969
WL	Categórica	Resultado da partida	68	Valores únicos: 2
Score	Categórica	Resultado da partida, incluindo o resultado registado para cada set	69	Valores únicos: 1623

3. Data Preparation

Nesta fase, o principal objetivo foi construir uma base de dados limpa, consistente e adequada para poder ser utilizada na fase de modelação, visto que a qualidade dos dados tem um impacto direto no desempenho do modelo.

Além disso, foi também nesta fase que criámos novas variáveis que achámos que podiam acrescentar valor à análise. Este processo inclui ainda o tratamento de valores omissos, a normalização dos dados, remoção de duplicados e correção de erros. De seguida são apresentados os diversos passos realizados na limpeza do *dataset*.

3.1. Preparação dos Dados Relativos aos Jogadores

O nosso *dataset* inclui dois tipos de jogadores: o *PlayerName*, que representa o jogador principal da partida, e o *Oponent*, que corresponde ao adversário nesse jogo. As variáveis disponíveis para cada tipo de jogador são diferentes.

No caso dos jogadores presentes no campo *PlayerName*, o *dataset* original disponibiliza informações detalhadas como o local de nascimento (*Born*), altura (*Height*), mão dominante e estilo de *backhand* (*Hand*), bem como o link para o perfil oficial no site da ATP (*LinkPlayer*). Por outro lado, os jogadores registados como *Oponent* possuem apenas o seu ranking ATP na altura da partida (*GameRank*).

Considerando que o nosso objetivo é ter uma base de dados completa e uniforme, decidimos que faria sentido criar novas variáveis de forma a incluir, sempre que possível, estas informações para todos os jogadores, independentemente de surgirem como *PlayerName* ou *Oponent*.

Além disso, concluímos que armazenar separadamente a informação relativa ao *PlayerName* e ao *Oponent* poderia não ser suficiente para o modelo compreender a relação entre ambos. Por esse motivo, decidimos criar também variáveis combinadas que traduzem comparações diretas entre os dois jogadores.

Um exemplo disso é a variável *Age_Gap*, que representa a diferença absoluta de idades entre o jogador principal e o adversário. Por exemplo, se a *Age_Player* for 25 anos e a *Age_Oponent* for 28 anos, então *Age_Gap* será igual a 3 anos. Este tipo de variável comparativa permite ao modelo captar de forma mais eficaz os contrastes entre os jogadores, que podem ser determinantes para o resultado do jogo.

3.1.1. *Born*, *Born_City*, *Born_Country*, *Hand*, *BackHand*, *Hand_vs*, *Height*

Começámos por extrair todos os jogadores (*PlayerName*), sem duplicados, que continham pelo menos um dos seguintes campos vazios: *Born, Hand, Height*.

Realizámos um *web scraping* através do *LinkPlayer* no site ATP para completar os valores em falta, e os resultados são apresentados na tabela seguinte:

	Nº *Players* com campo vazio antes	Nº *Players* com campo vazio depois
Born	496	6
Hand	284	0
Height	516	480

Após a realização do *web scraping* para os *Players* reparámos que a variável *Hand,* apesar de não ter valores omissos, continha 80% das linhas com "*unkown*". Assim, foi feito um segundo *web scraping* no site *Tennis Explorer* que disponibiliza informações complementares sobre a mão dominante dos jogadores.

	Nº *Players* com unknown antes	Nº *Players* com unknown depois
Hand	261	96

Como para os oponentes não tínhamos qualquer informação sobre estas variáveis, incluindo o *link* do perfil ATP, fizemos um *web scraping* no site *Tennis Explorer* através da pesquisa pelo apelido. Os resultados do *web scraping* são apresentados na tabela seguinte:

	Nº *Oponents* com campo vazio antes	Nº *Oponents* com campo vazio depois
Born	318	89
Hand	318	92
Height	318	296

A informação relativa à mão dominante e ao tipo de *backhand* dos jogadores encontrava-se originalmente agregada numa única *string*, separada por vírgulas.

Para facilitar a análise e o tratamento dos dados, optámos por separar essa informação em colunas distintas, criando as variáveis *Hand, Backhand, Hand_Op e Backhand_Op*, correspondentes ao jogador principal e ao oponente.

Adicionalmente, consideramos que a combinação da mão dominante de ambos os jogadores poderia ter impacto no decorrer do jogo, uma vez que determinadas combinações (por exemplo, jogador destro contra esquerdino) podem influenciar o estilo de jogo e os padrões de resposta. Por isso, criámos a variável *Hand_vs*, que representa a combinação das mãos dominantes dos dois jogadores.

Por exemplo, se ***Hand*** = *Right-Handed* e ***Hand_Op*** = *Left-Handed*, então ***Hand_vs*** = *Right-Handed vs Left-Handed.* Esta variável é simétrica, ou seja, se o *Player* for esquerdino e *Oponent* for destro, o valor de ***Hand_vs*** mantém-se igual.

Notámos também que a variável *Born*, responsável por armazenar o local de nascimento dos jogadores, apresentava uma estrutura inconsistente. Em alguns casos, combinava a cidade e o país numa única *string* separada por vírgulas, enquanto noutros continha apenas uma das duas informações, ou apenas a cidade, ou apenas o país.

Para separar esta informação começámos por dividir as *strings* pelas vírgulas em 4 variáveis: *born_1, born_2, born_3, born_4*.

Exemplo:

String original	*born_1*	*born_2*	*born_3*	*born_4*
Lisboa, Portugal	Lisboa	Portugal	*null*	*null*
Manhattan, New York, USA	Manhattan	New York	USA	*null*

Para ajudar na correta identificação de cidades e países, arranjámos um ficheiro *csv* com uma lista de cidades e o respetivo país, que comparava os *born_n* e atribuía-os às colunas objetivo: *Born_City* e *Born_Country* (para os *Players*) e *Born_City_Op* e *Born_Country_Op* (para os Oponentes).

Em situações de conflito, por exemplo na escolha da cidade na string "Manhattan, New York, USA", foi mantida a cidade com maior população (New York).

Relativamente à variável *Height*, verificámos que inicialmente existiam valores de altura apenas para os jogadores identificados como *PlayerName*. Logo, o primeiro passo foi identificar quais desses jogadores atuaram como oponentes também e adicionámos as respetivas alturas a esses oponentes, sempre que disponíveis.

Para os casos em que a altura dos oponentes estava em falta, procedemos ao preenchimento desses dados, através do processo de *web scraping* ao *site TennisExplorer*. Este processo assegura que as variáveis relacionadas com a altura estavam completas para ambos os jogadores envolvidos em cada jogo.

3.1.2. *Date of Birth (DOB), Age, Age_Gap* e *Age_Diff*

De acordo com o enunciado, tínhamos de ter a variável Data de Nascimento (*DOB - Date of Birth*), que não estava presente no *dataset* original. Para a obter, realizámos um *web scraping* que no final nos permitiu ter duas novas variáveis: *DOB* e *DOB_Op*.

Tendo a data de nascimento, criámos duas novas variáveis: *Age* e *Age_Op*, que contêm a idade do jogador ou oponente na data em que se realizou o jogo.

Considerando que a diferença de idades entre os jogadores poderia ser um fator relevante para o desempenho em campo, decidimos criar também a variável *Age_Gap*, que representa a diferença absoluta entre essas idades e a variável *Age_Difference*, que representa a diferença não absoluta.

Exemplo: ***Age*** = 25; ***Age_Op*** = 28, ***Age_Gap*** = 3; ***Age_Difference*** = -3

3.1.3. *GameRank, RankPlayer, Difference_ranks* e *Difference_ranks_Gap*

No *dataset* original, apenas os oponentes tinham associada a variável *GameRank*, que corresponde ao seu *ranking* ATP na data em que o jogo foi disputado.

Considerando a relevância desta informação para ambos os jogadores, decidimos alargar a sua disponibilidade ao jogador principal. Para isso, realizámos um processo de *web scraping* utilizando o *LinkPlayer*, o que nos permitiu recolher o *ranking* ATP do jogador principal na mesma data, guardando essa informação na nova variável *Rank_Player*.

De forma a captar a diferença entre o nível competitivo dos dois jogadores, criámos também a variável *Difference_ranks_Gap*, que representa a diferença absoluta entre os *rankings* de ambos, e a variável *Difference_ranks*, que representa a diferença não absoluta. Estas variáveis comparativas permitem ao modelo identificar com maior precisão eventuais desequilíbrios entre os jogadores, que podem influenciar diretamente o resultado do encontro.

Exemplo: ***GameRank*** = 30; ***Rank_Player*** = 58; ***Difference_ranks_Gap*** = 28; ***Difference_ranks*** = -28

3.1.4. *Weight e IMC*

Pensámos ainda que podia ser interessante ter o peso dos jogadores, pois este pode ser um indicador da forma física, e realizámos um *web scraping* para recolher essa informação para jogadores e oponentes.

Para os jogadores que tínhamos o peso e a altura, criámos também uma variável com o cálculo do IMC ($\frac{peso}{altura^2}$). Além disso, fizemos a diferença absoluta entre o *IMC* e o *IMC_Op*, originando assim a nova variável *IMC_abs*.

Ficámos então com 5 novas variáveis: *Weight, Weight_Op, IMC, IMC_Op e IMC_abs.*

3.1.5. Indicadores de Forma e Histórico

Considerámos que a forma atual dos jogadores e o seu histórico competitivo poderiam ser fatores úteis na previsão do número de *sets*. Por isso, desenvolvemos

um conjunto de variáveis que refletem o desempenho recente e o comportamento típico dos atletas em diferentes contextos.

Para garantir a fiabilidade dessas variáveis, foi necessário organizar o *dataset* por ordem cronológica. Para tal, utilizámos a data de início do torneio (*Start*) e, adicionalmente, recorremos a uma hierarquia definida para as rondas dos torneios (*GameRound*), assumindo que cada ronda ocorre apenas após a conclusão das anteriores.

Desde o início, reconhecemos que estas variáveis poderão apresentar um número significativo de valores omissos, mas a utilidade efetiva de cada uma será avaliada posteriormente.

3.1.5.1. Percentagem de vitórias do jogador por ronda até à partida

A percentagem de vitórias por ronda permite identificar se um jogador tende a ter mais ou menos sucesso em determinadas fases das competições. Para isso, implementámos uma função que calcula, de forma acumulada, a percentagem de vitórias de cada jogador em cada ronda, até à data da partida.

Esta percentagem é armazenada nas colunas *Percentagem_Vitorias_PlayerName* e *Percentagem_Vitorias_Oponent*. Nos casos em que o jogador nunca disputou uma determinada ronda, o valor correspondente fica com valor omisso.

Tal como em variáveis anteriores, tendo as informações de ambos os jogadores que disputam a partida, criamos a variável *Percentagem_Victory_Abs*, que representa a diferença absoluta entre as percentagens dos dois jogadores e a *Percentagem_Victory_diff*, com a diferença não absoluta.

3.1.5.2. Percentagem de vitórias nas últimas 5 partidas do jogador

Com o objetivo de medir a forma recente dos jogadores, calculamos a percentagem de vitórias nas últimas cinco partidas antes de cada jogo. Esta métrica permite perceber em que estado competitivo o jogador se encontrava no momento da partida.

Nos casos em que um jogador ainda não disputou cinco encontros no nosso *dataset*, o valor fica omisso. Os resultados são guardados nas colunas *Recent_Form_Player* e *Recent_Form_Oponent*, e a diferença absoluta e não

absoluta entre ambas é calculada, respetivamente, nas variáveis Abs_Recent_Form e *Diff_Recent_Form*.

3.1.5.3. Percentagem de *sets* vencidos pelo jogador nas últimas 5 partidas

Para analisar mais profundamente o desempenho recente, considerámos também a percentagem de *sets* ganhos nos últimos cinco jogos de cada jogador. A fórmula aplicada foi:

$$\frac{sets\ vencidos}{sets\ vencidos + sets\ perdidos}$$

Dado que todos os jogos são disputados à melhor de 3 *sets*, esta métrica permite uma leitura direta da consistência do jogador em vencer *sets*. Por exemplo, uma vitória por 2-0 indica que o jogador venceu todos os *sets*, enquanto uma derrota por 1-2 indica que venceu apenas um. Os valores calculados são guardados nas variáveis *Per_Win_Sets* e *Per_Win_Sets_Oponent*.

À semelhança de outras métricas do modelo, calculámos ainda a diferença entre os dois jogadores (*Per_Win_Sets_diff*) e a diferença absoluta (*Per_Win_Sets_abs*), permitindo uma melhor comparação entre o desempenho recente de ambos.

3.1.5.4. Percentagem de vitórias do Jogador em cada tipo de *Ground*

A adaptação de cada jogador ao tipo de superfície onde se realiza o torneio pode ter grande impacto no desempenho. Assim, calculamos a percentagem de vitórias por tipo de piso (*hard, clay, grass*, etc.), de forma semelhante ao que foi feito para as rondas.

Se o jogador nunca competiu num determinado tipo de superfície, o valor correspondente é nulo. Os resultados estão armazenados nas colunas *Ground_Wins* e *Ground_Wins_Op*, sendo ainda criadas as variáveis *Ground_Wins_Diff* e *Ground_Wins_Abs* para representar a diferença e diferença absoluta entre os dois jogadores, respetivamente.

3.1.5.5. Confronto direto entre dois jogadores

O histórico de confrontos diretos entre dois jogadores pode revelar padrões que tendem a repetir-se. Considerámos como confronto direto qualquer jogo entre os mesmos dois jogadores, independentemente da ordem (por exemplo, Jogador A vs Jogador B ou Jogador B vs Jogador A).

De forma a tornar o histórico mais realista, inserimos pesos no número de jogos já disputados entre ambos. Se antes da partida, houver apenas um resultado de um jogo, o peso do confronto direto é menor do que se houver 3 jogos entre ambos os jogadores no passado. Assim, atribuiram-se os seguintes pesos:

Número de jogos (confronto direto)	Peso
1	0.25
2	0.25
3	0.35
4	0.65
5	1

As percentagens ponderadas de vitórias são guardadas nas colunas *H2He H2H_Op*. Além disso, criámos as variáveis *H2H_Diff* e *H2H_Abs*, que representam, respetivamente, a diferença e a diferença absoluta entre os desempenhos históricos dos dois jogadores. O número total de confrontos anteriores é registado na coluna *N_Games*.

3.2. Preparação dos Dados Relativos aos Jogos

3.2.1. Partidas não realizadas, duplicadas e à melhor de 5 sets

Durante o processo de preparação dos dados, também achámos que faria sentido eliminar as linhas que correspondiam a partidas não realizadas. Esses casos foram identificados quando a variável *Oponent* era igual a "*bye*", indicando que a partida não ocorreu porque o *Player* avançou automaticamente para a fase seguinte sem disputar o jogo.

Além disso, partidas realizadas entre os mesmos jogadores, na mesma ronda, no mesmo torneio e na mesma data foram classificadas como duplicadas e

foi eliminada uma das cópias, para assegurar que cada partida estava representada uma única vez.

Adicionalmente, foram removidas todas as partidas disputadas à melhor de 5 *sets*, de forma a garantir que o modelo fosse treinado exclusivamente com jogos à melhor de 3 *sets*, assegurando assim a consistência e a aplicabilidade das previsões.

3.2.2. *Date*

A variável original *Date* encontrava-se em formato de *string*, contendo duas datas separadas por um hífen (por exemplo: 1999.07.05 - 1999.07.11), representando o intervalo de realização de cada torneio. Para facilitar o tratamento e a análise temporal dos dados, esta informação foi desdobrada em três colunas distintas: *Start* (data de início do torneio), *End* (data final do torneio) e *Days*, que representa o número de dias do torneio.

Esta reformulação permite uma utilização mais flexível da variável temporal, nomeadamente na análise de padrões ao longo do tempo ou na comparação entre torneios de diferentes durações.

3.2.3. *Prize*

A variável *Prize* também precisou de ser tratada pois apresentava vírgulas entre os algarismos e a moeda juntamente com o número (Exemplo: $125,000), o que impedia que esta variável fosse tratada como valor inteiro.

De forma a podermos comparar os valores do prémio entre torneios com diferentes datas, utilizámos dois ficheiros de texto contendo os valores anuais do CPI (*Consumer Price Index*): um para os Estados Unidos, retirado do *site* oficial do *Bureau of Labor Statistics* (BLS), e outro para a zona euro, feito manualmente copiando os valores da tabela do site www.rateinflation.com.

A função que criámos tem em conta o ano de início de cada torneio (*Start*) e, dependendo da moeda indicada na coluna *Currency*, seleciona o dicionário de CPI correspondente. Para valores em euros, o prémio é simplesmente ajustado para o valor equivalente em 2025 através do fator de inflação calculado com o CPI.

No caso de valores em dólares, o prémio é primeiro ajustado com base no CPI dos EUA e depois convertido para euros usando uma taxa de câmbio fixa de 1$

= 0.80€. O valor final inflacionado é guardado numa nova coluna chamada *Prize_2025*.

Assim, tornamos os valores comparáveis ao longo do tempo, o que permite ao modelo de previsão interpretar corretamente a importância e magnitude real de cada torneio, independentemente do ano em que ocorreu.

3.2.4. *Location*

À semelhança da variável *Born*, a variável *Location* tinha a informação representada de forma não consistente, combinando por vezes a cidade e o país numa única *string* separada por vírgulas, enquanto noutras ocasiões continha apenas uma das duas informações, ou apenas a cidade, ou apenas o país.

Para uniformizar os dados, começámos por segmentar a *string* original com base nas vírgulas, criando três colunas auxiliares: *location_1, location_2, location_3*.

De seguida, utilizámos o mesmo ficheiro *csv* utilizado para tratar a variável *Born*, que contém cidades e países, e fizemos uma correspondência de forma a obtermos a nova variável *Location_City* (o país foi ignorado porque todos os jogos decorrem na Bélgica).

3.2.5. *Score*

Originalmente, a variável *Score* apresentava os resultados das partidas em formato textual, muitas vezes com inconsistências e variações de formatação (**Exemplo**: “63 62 62” **ou “**76, 67, 63” **ou “**36 46”). Para sermos capazes de extrair informação útil, criámos a partir desta variável duas novas colunas: *Sets*, que representa o número de *sets* disputados na partida (e corresponde à nossa variável *target*), e *Games*, que indica o número total de jogos realizados.

A variável *Sets* soma o número de pares de números, separados por espaços. **Exemplo**: “63 62 62” → 3 *sets*; **ou** “36 46” → 2 *sets*.

A variável *Games* soma todos os algarismos que aparecem na variável *Score* original. **Exemplo**: “63 62 62” → 25 jogos; **ou** “36 46” → 19 jogos.

Durante o processo, foram eliminados todos os jogos que continham siglas como RET (*retired* - jogo não terminado), DEF (*default* - jogador desqualificado) e W/O (*walkover* - ausência de jogo), uma vez que estes casos não refletem um resultado competitivo regular e poderiam introduzir ruído nos dados de treino do modelo.

3.3. Escolha das variáveis para a modelação

No final da etapa da limpeza e da criação de novas variáveis, tivemos de selecionar as variáveis que poderiam fazer sentido manter para a etapa seguinte: a criação dos modelos.

Como o objetivo do projeto é prever o número de *sets* de um jogo, segue-se uma breve justificação do uso, ou não, de cada variável.

3.3.1. Variáveis qualitativas com muita diversidade

A utilização de variáveis qualitativas com um número elevado de valores únicos revela-se pouco adequada na implementação de modelos preditivos. Isto porque, ao surgirem com muito pouca frequência (por exemplo, quando um nome aparece apenas algumas vezes em todo o *dataset*), estas variáveis não oferecem padrões consistentes que o modelo possa generalizar de forma eficaz. Além disso, a codificação dessas variáveis, como através da criação de variáveis *dummies*, implicaria um aumento substancial da dimensionalidade dos dados, resultando num custo computacional elevado com benefícios mínimos em termos de *performance*. Este tipo de abordagem poderia ainda levar a problemas como *overfitting*, maior dispersão dos dados e dificuldade de interpretação dos resultados.

Tendo isto em consideração, optámos por não incluir no modelo as seguintes variáveis: *PlayerName, Oponent, Born_Country, Born_City, Born_Country_Op, Born_City_Op, Tournament e Location_City.*

3.3.2. Datas

As variáveis em formato de data não foram incluídas nos modelos preditivos, uma vez que representam apenas características contextuais, como a data de nascimento dos jogadores ou o calendário dos torneios.

Por não oferecerem valor preditivo direto para o número de *sets* e por não serem informativas por si só sem transformação adicional, optámos por excluir as seguintes variáveis: *DOB, DOB_Op, Start e End.*

3.3.3. Variáveis individuais referentes aos jogadores

Como mencionado anteriormente, as variáveis que existem tanto para o *PlayerName* como para o *Oponent* foram agrupadas numa só, à exceção do peso e da altura. As últimas duas variáveis referidas, foram usadas no cálculo do IMC.

Logo, tendo as variáveis conjuntas, não faria sentido utilizarmos também as variáveis individuais e, portanto, não foram utilizadas as seguintes variáveis: *Height, Weight, Hand, Backhand, Age, Rank_Player, Height_Op, Weight_Op, Hand_Op, Backhand_Op, Age_Op, Rank_Op, Percentagem_Vitorias_PlayerName, Percentagem_Vitorias_Oponent, Recent_Form_Player, Recent_Form_Oponent, H2H, H2H_op, N_Games, Per_win_Sets, Per_win_Sets_Oponent, ground_wins, ground_wins_Op*.

3.3.4. Variáveis *diff* (diferença não absoluta)

Durante o processo de preparação dos dados, foram criadas variáveis que representavam a diferença e a diferença absoluta entre atributos dos dois jogadores. No entanto, optámos por utilizar apenas as diferenças absolutas, pois o foco da análise era apenas verificar a desigualdade entre os jogadores e não a direção da diferença, ou seja, se o valor mais alto pertence ao jogador A ou B.

Além disso, o uso de variáveis absolutas contribui para evitar interpretações ambíguas e reduz potenciais efeitos de colinearidade com outras variáveis que já consideram a identidade do jogador.

3.3.5. Variáveis relacionadas com o resultado do jogo

As variáveis *WL* e *Games* não poderão ser utilizadas na modelação pois contêm informação que só se sabe após a partida acontecer, logo não faria sentido incluí-las no contexto do nosso problema.

3.3.6. Variáveis escolhidas para modelação e as suas estatísticas descritivas

Depois da limpeza dos dados, ficámos com 5.647 jogos. Estas são as linhas que passaram por todos os filtros implementados até aqui. As variáveis escolhidas para a modelação foram selecionadas como sendo aquelas que, à partida, poderiam permitir captar os aspetos mais importantes que influenciam o número de *sets*.

Sets - variável *target*

Para este trabalho, escolhemos a variável *Sets* como a *target* dos modelos preditivos, dado que o objetivo principal é prever o número de *sets* jogados em cada partida de ténis. A variável *Sets* assume valores que indicam se o jogo terminou em 2 ou 3 *sets*, refletindo assim a duração e o desfecho do encontro.

Para os modelos, transformámos a variável *Sets* em binária, onde:

- **0**: representam os jogos de 3 *sets,*
- **1**: representam os jogos de 2 *sets,*

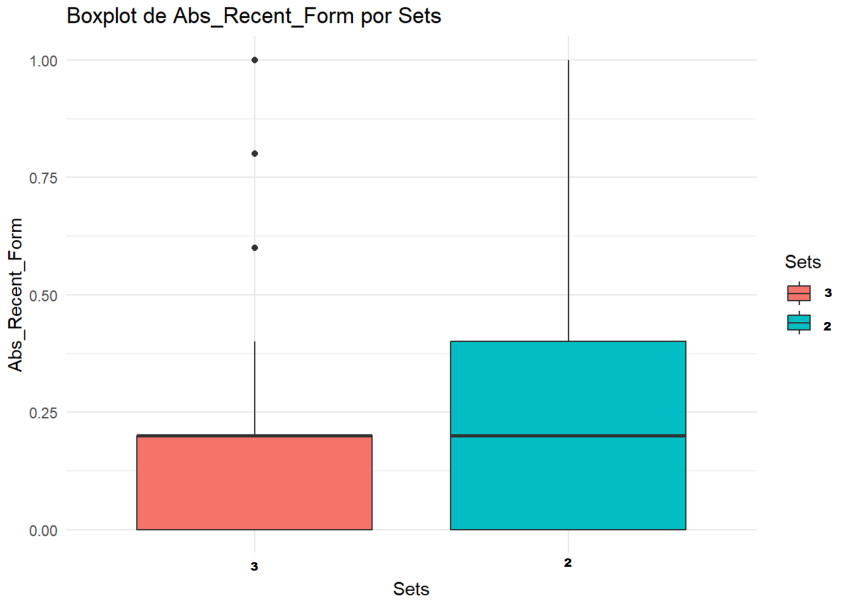
simplificando assim a modelação para um problema de classificação binária. Os valores obtidos revelam que a maioria das partidas (3866 jogos) terminou em 2 *sets*, enquanto uma parcela menor (1781 jogos) terminou em 3 *sets*.

Abs_Recent_Form

A variável *Abs_Recent_Form* foi selecionada para inclusão no modelo devido à sua importância em refletir as diferenças do desempenho atual de cada um dos atletas. O desempenho recente é um fator essencial para entender o estado competitivo dos jogadores, influenciando diretamente a probabilidade de vitória e a dinâmica da partida. Através da tabela abaixo verificamos que existem 4075 valores omissos (cerca de 72% do total de registos).

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	0	0.200	0.213	0.400	1	4075

No *boxplot* abaixo pode observar-se a distribuição da variável dividida por *sets*. Pode observar-se que a distribuição da categoria 2 número de *sets* é mais equilibrada que a de 3 *sets*.

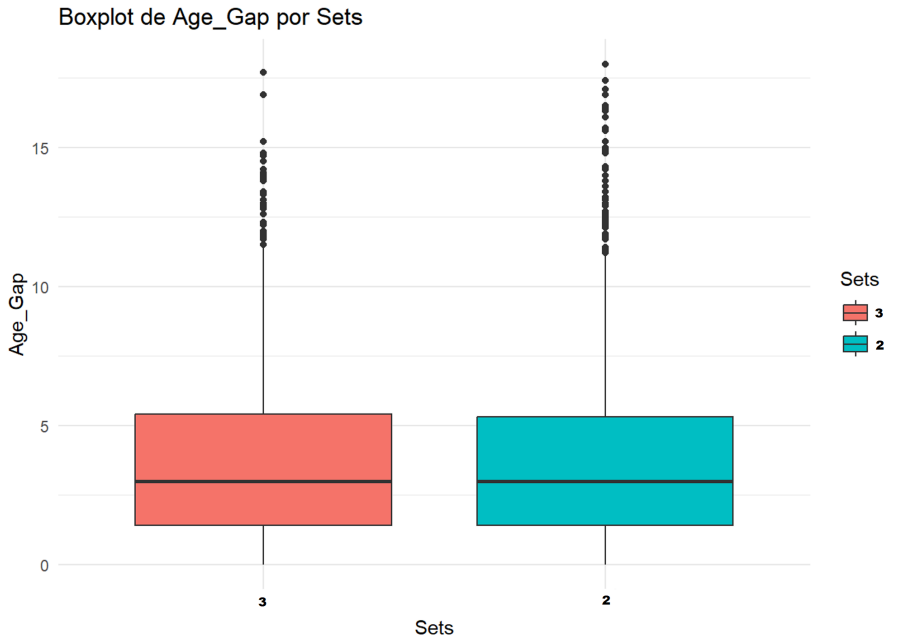


Age_Gap

Decidimos incluir a variável *Age_Gap* no modelo por ser um possível fator de influência no desempenho e resultado do jogo. A diferença de idade pode refletir variações em experiência, resistência física e maturidade tática, elementos que podem impactar no número de *sets* jogados. Através da tabela abaixo, podemos evidenciar o número considerável de valores omissos, correspondente a 35% do *dataset*.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	1.400	3	3.749	5.300	18	1959

No *boxplot* abaixo encontra-se a distribuição do *Age_Gap* por número de *sets*, e pode-se verificar que a distribuição é semelhante entre as categorias.



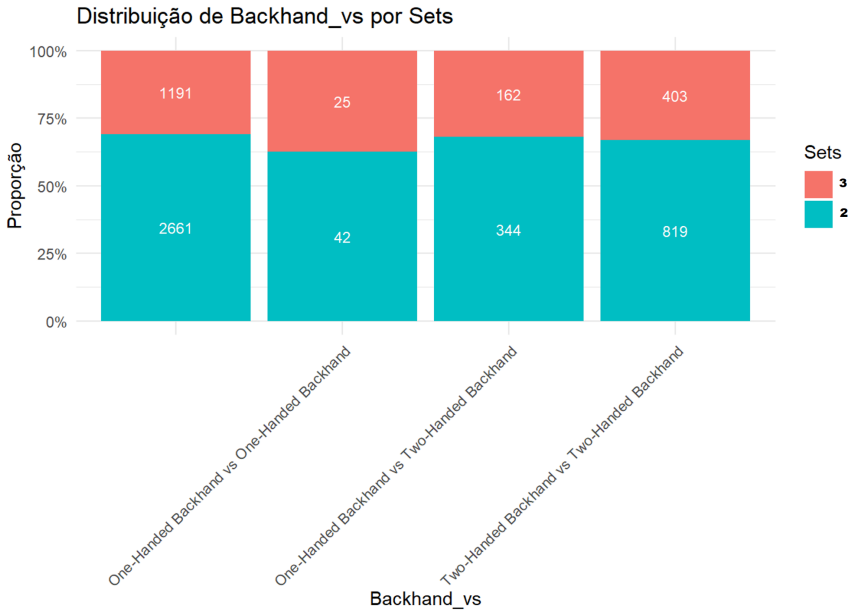
Backhand_vs

A variável *Backhand_vs* foi incluída no modelo pois o estilo de *backhand* pode influenciar o jogo, as estratégias e o desempenho, impactando potencialmente o número de *sets* jogados.

No total, temos:

- *One-Handed Backhand vs One-Handed Backhand*: 67 observações
- *One-Handed Backhand vs Two-Handed Backhand*: 506 observações
- *Two-Handed Backhand vs Two-Handed Backhand*:1222 observações
- Omissos: 3852 observações

A distribuição desta variável por *set* pode ser visualizada no gráfico de barras abaixo. Os valores omissos correspondem a cerca de 68% do total de observações e a sua distribuição corresponde à primeira coluna do gráfico.



Days

A variável *Days* foi incluída no modelo por poder influenciar o desempenho dos atletas, pois é um indicador importante do tempo de descanso ou fadiga acumulada, e consequentemente, o número de *sets* jogados.

Através da tabela abaixo, verificamos que a ausência de valores omissos nesta variável permite que o modelo utilize de forma consistente a informação temporal, sem necessidade de tratamento adicional. No entanto, o facto de o 1º e o 3º quartis serem iguais indica que esta variável apresenta pouca dispersão, refletindo uma concentração dos valores numa faixa muito restrita.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
5	6	6	6.074	6	13	-

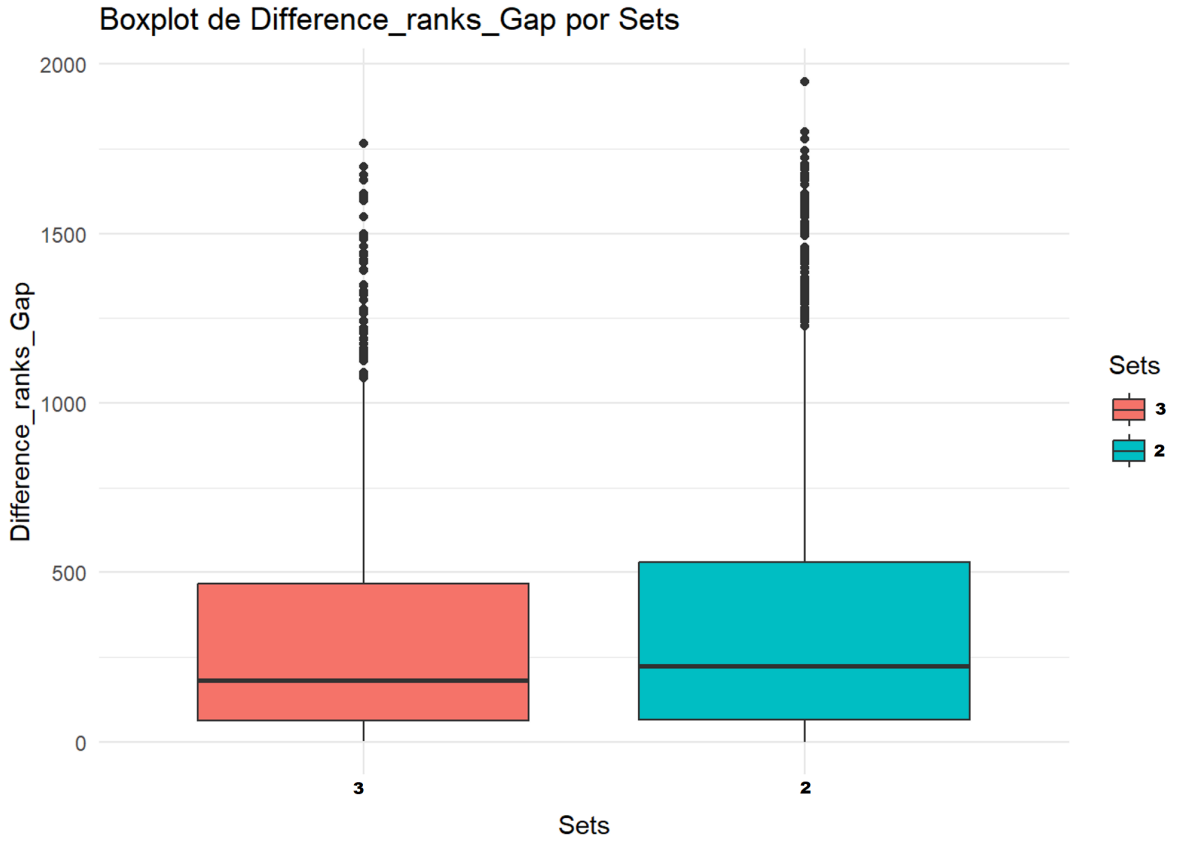
Difference_Ranks_Gap

A variável *Difference_Ranks_Gap* foi incluída no modelo por refletir o nível relativo de experiência e desempenho dos atletas, indicando a disparidade competitiva entre estes, o que pode influenciar o desfecho do jogo e o número de *sets* jogados.

Através da tabela abaixo, verificamos a existência de 861 valores omissos, cerca de 15% do *dataset*.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	65	210	340.2	513	1949	861

No *boxplot* abaixo pode ser verificado que as distribuições por categoria (2 ou 3 *sets*) são semelhantes.



GameRound

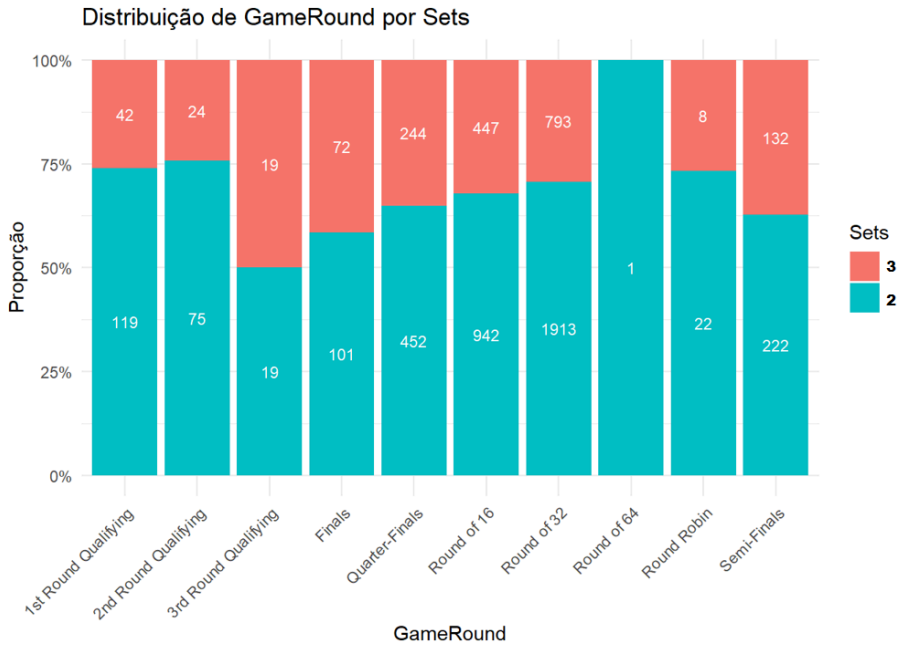
A variável *GameRound* é relevante para o modelo, pois a importância e a pressão competitiva tendem a aumentar conforme o torneio avança, o que pode influenciar o desempenho dos jogadores e o número de *sets* jogados.

No total, temos:

- *Round Robin :* 30 observação
- *Round of 64 :* 1 observação
- *Round of 32 :*2706 observações
- *Round of 16 :*1389 observações
- *Quarter-Finals :* 696 observações
- *Semi-Finals* : 354 observações
- *Finals* : 173 observações

- *1st Round Qualifying* : 161 observações
- *2nd Round Qualifying* : 99 observações
- *3rd Round Qualifying* : 38 observações

Há categorias com pouca expressão na *dataset*, sendo as mais expressivas a Round of 32 e Round of 16. Nestas duas categorias a percentagem de observações na categoria 2 *sets* é cerca de 75%.

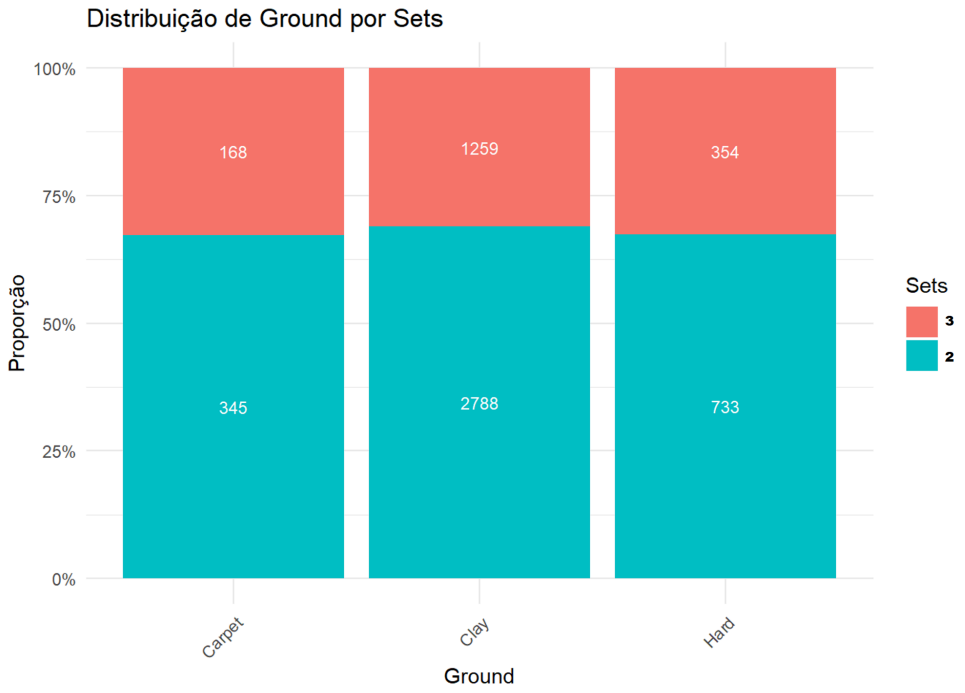


Ground

Pensámos que incluir a variável *Ground* poderia influenciar o número de *sets*, uma vez que o tipo de piso pode afetar o ritmo e o equilíbrio dos jogos.
No total, temos:

- *Clay*: 4047 observações
- *Hard*: 1087 observações
- *Carpet*: 513 observações
- Omissos: 0 observações

No nosso *dataset* a maior parte dos jogos são em piso *Clay* e no gráfico de barras abaixo, pode notar-se que a distribuição por número de *sets* (2 ou 3) é semelhante entre os diferentes tipos de piso.

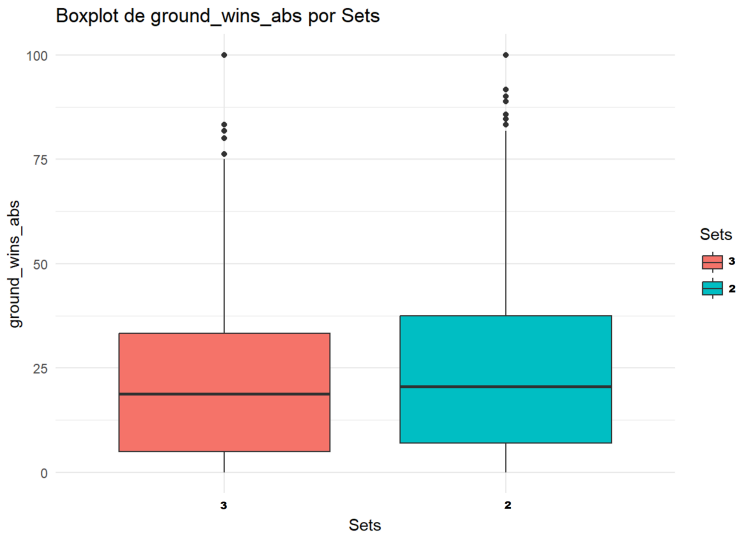


Ground_wins_abs

Pensámos que incluir a variável Ground_Wins_Abs poderia influenciar o número de *sets*, já que a diferença da adaptação de cada jogador ao tipo de superfície do torneio pode ter um grande impacto no seu desempenho. No total, temos 1823 observações com valor omisso (cerca de 32% do *dataset*), e estes são casos em que pelo menos um dos jogadores nunca competiu nesse tipo de piso. Na tabela seguinte podem ser visualizadas as estatísticas descritivas para esta variável:

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	6.475	20	24.028	36.4	100	1823

Abaixo apresentamos um *boxplot* com a distribuição da variável Ground_Wins_Abs por número de *sets*, permitindo observar que não existe muita diferença entre ambas as categorias.



H2H_abs

Decidimos incluir a variável *H2H_abs* pois pensámos que poderia influenciar o número de *sets*, uma vez que representa a diferença absoluta no confronto direto entre os dois jogadores. Esta variável permite capturar possíveis desequilíbrios históricos, por exemplo, se um dos jogadores costuma ganhar claramente ao outro, é mais provável que o jogo atual seja menos equilibrado e, por isso, resolvido em menos *sets*. Como pode ser visto na tabela abaixo, o maior problema desta variável é a grande quantidade de valores omissos (cerca de 94% de linhas omissas), o que já era previsto quando a variável foi criada.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	25	25	24.084	35	65	5303

Hand_vs

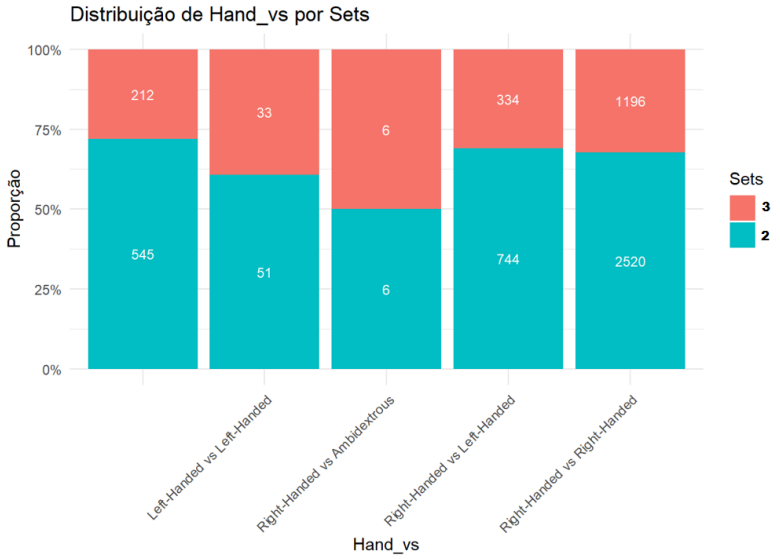
A variável *Hand_vs* reflete a diferença na mão dominante entre os jogadores em cada partida, podendo ser relevante para a previsão do número de *sets*, uma vez que confrontos entre jogadores com mãos dominantes diferentes podem influenciar o estilo de jogo e a dinâmica do encontro.

A distribuição por categoria é a seguinte:

- *Left-Handed vs Left-Handed*: 84
- *Right-Handed vs Ambidextrous:* 12
- *Right-Handed vs Left-Handed*: 1078
- *Right-Handed vs Right-Handed*: 3716

- Nulo: 757

No gráfico de barras abaixo pode ser visualizada a distribuição por número de *sets* e podemos verificar que é relativamente semelhante nas categorias com mais elementos. A primeira coluna corresponde ainda à distribuição dos valores omissos por *set*.

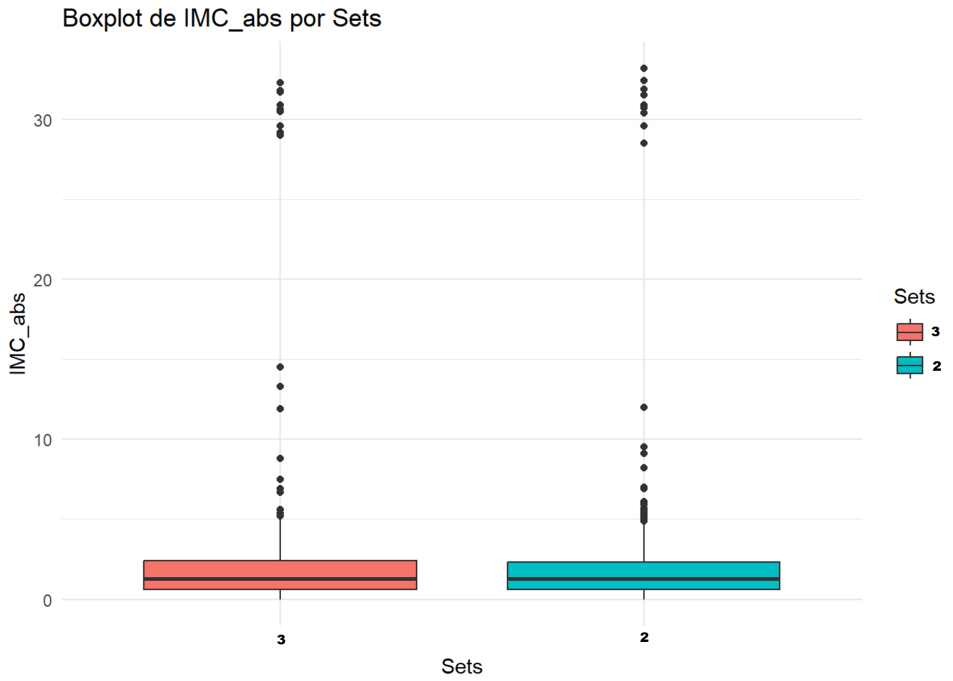


IMC_abs

A variável *IMC_abs* representa a diferença absoluta do Índice de Massa Corporal (IMC) entre os jogadores de cada partida. Esta variável pode ser relevante para a previsão do número de *sets*, pois diferenças físicas significativas podem influenciar o desempenho e a resistência dos atletas durante o jogo. Na tabela abaixo podemos verificar que em média esta diferença é de 1.3 e que temos 3270 valores omissos (cerca de 58% do *dataset*), o que é bastante e poderá ser problemático.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	0.6	1.3	1.84	2.3	33.2	3270

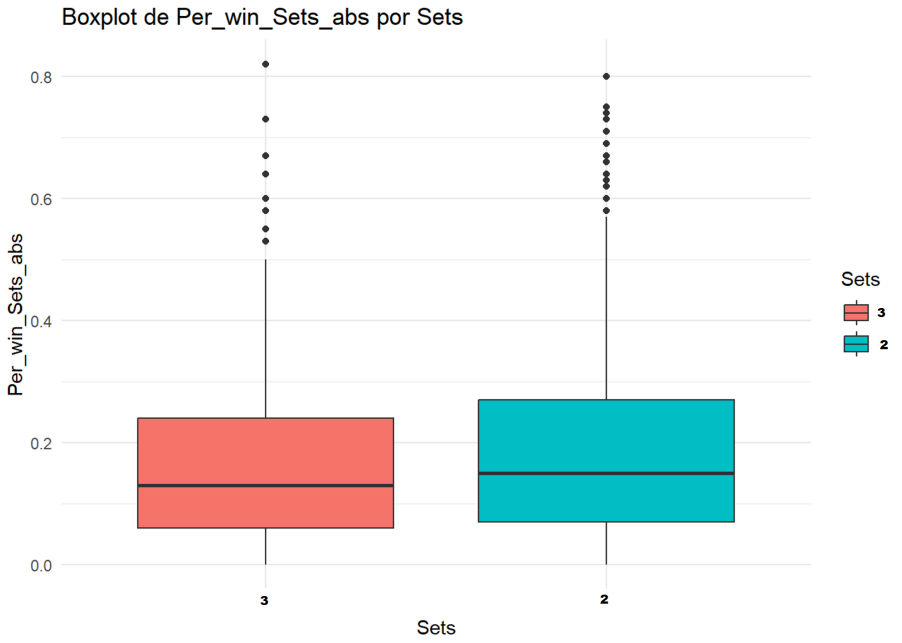
No *boxplot* abaixo pode-se notar que a dispersão desta variável é muito semelhante para os jogos com 2 ou 3 *sets*.



Per_win_Sets_abs

Tal como as restantes variáveis de forma e histórico, esta variável apresentou potencial teórico por espelhar a diferença competitiva atual entre ambos os jogadores relativamente ao número de *sets* ganhos. É também uma variável potencialmente problemática pois apresenta 72% de valores omissos. As estatísticas descritivas podem ser visualizadas na tabela abaixo, tal como o *boxplot* com as distribuições por categoria.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	0.07	0.150	0.181	0.26	0.82	4075

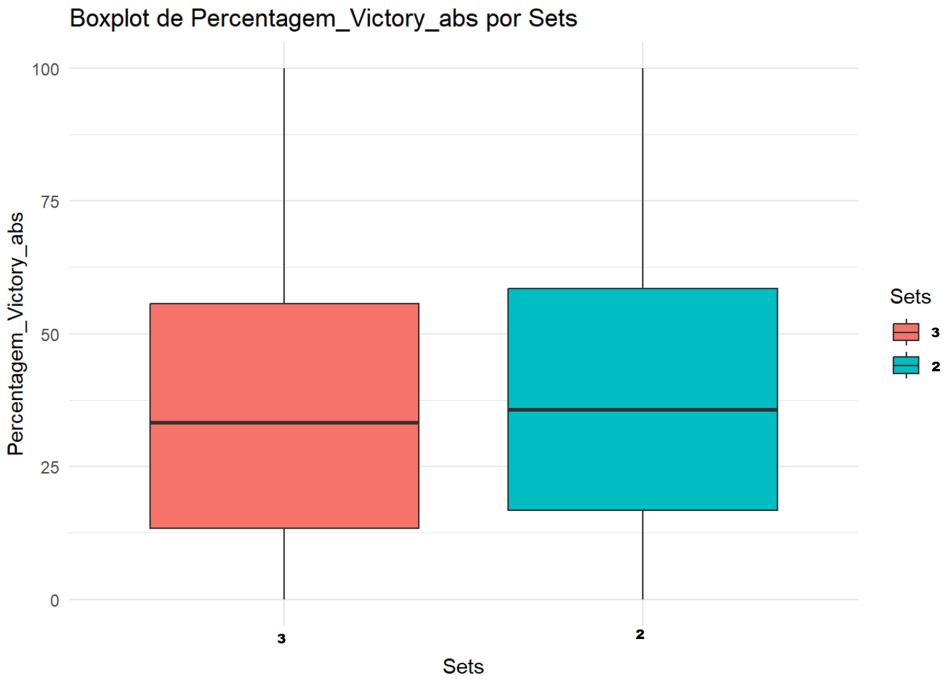


Percentagem_Victory_Abs

A diferença absoluta na percentagem de vitórias por ronda entre os dois jogadores poderá ser útil pois quanto maior a diferença, maior a probabilidade de um jogador dominar a partida, resultando num jogo mais rápido e com menos *sets* disputados. Por outro lado, uma diferença pequena sugere que os jogadores têm desempenhos semelhantes naquela fase, o que pode indicar um confronto mais equilibrado e, consequentemente, uma maior chance do jogo se prolongar por mais *sets*. As estatísticas descritivas desta variável estão na tabela abaixo. É também uma variável com muitos valores omissos, que correspondem a 63% do *dataset*.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	15.4	34.5	39.41	57.1	100	3530

No *boxplot* abaixo pode-se observar que a distribuição é semelhante para os jogos com 2 e 3 *sets*.



Prize_2025

O valor do prémio do torneio onde a partida está a ser disputada pode ser uma variável importante para prever o número de *sets*, pois torneios com prémios mais elevados costumam atrair jogadores de topo, o que pode resultar em jogos mais equilibrados e competitivos, frequentemente com mais *sets* disputados.

Por outro lado, torneios com prémios mais baixos podem apresentar uma maior variação no nível dos jogadores, levando a partidas mais rápidas e com menos *sets*. Assim, incluir o prémio do torneio poderá ajudar o modelo a captar o contexto competitivo e a pressão associada, que são fatores que influenciam a duração e intensidade do jogo. De forma a reduzir a elevada dispersão de valores, aplicámos o logaritmo à variável.

Na tabela abaixo, após a transformação, apresentam-se as estatísticas descritivas desta variável.

Mínimo	1ºQuartil	Mediana	Média	3ºQuartil	Máximo	Omissos
0	9.406	9.746	10.502	11.802	14.556	-

3.4. Correlações entre as variáveis escolhidas

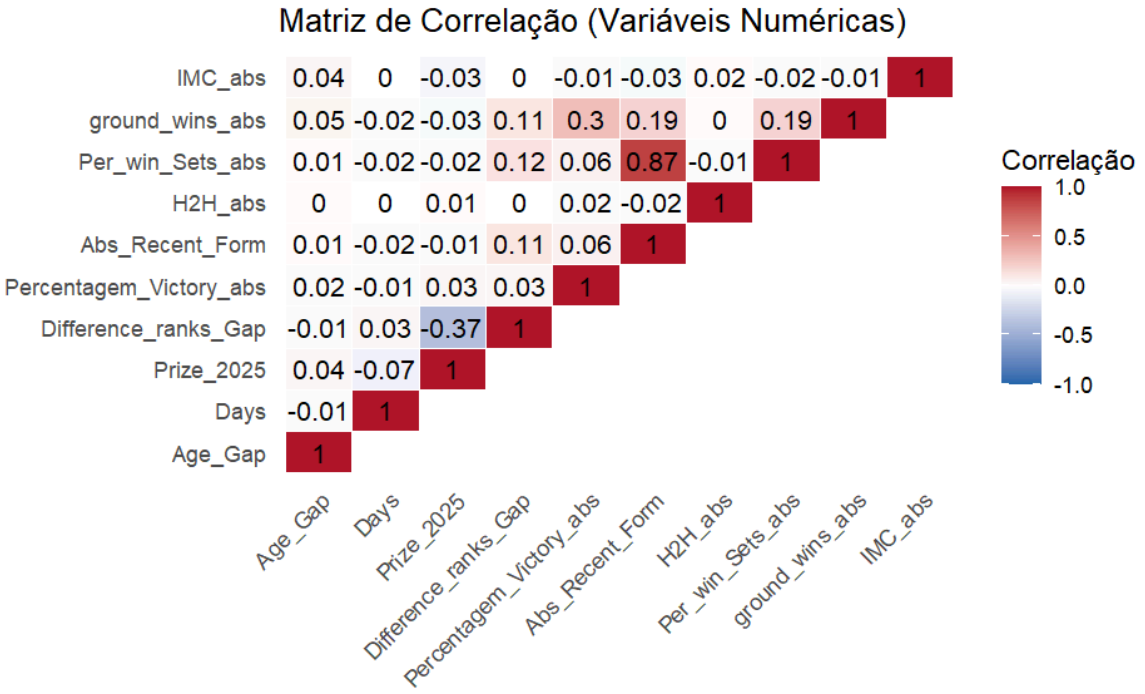
Com o objetivo de entender melhor as relações entre as variáveis do nosso *dataset*, realizámos uma análise de correlação, permitindo-nos identificar possíveis redundâncias, bem como associações relevantes que possam contribuir para a qualidade do modelo.

Foram utilizadas diferentes métricas de correlação consoante o tipo de variáveis analisadas.

3.4.1 Correlação de *Pearson*

Aplicada a variáveis numéricas, mede a força e a direção da relação linear entre duas variáveis. Os valores variam entre -1 (correlação negativa perfeita) e 1 (correlação positiva perfeita), sendo que valores próximos de 0 indicam ausência de relação linear.

Pode-se observar na figura abaixo a matriz de correlação de *Pearson*:



Na matriz apresentada acima, podemos destacar os seguintes pontos principais:

- Correlação alta entre ***Per_win_Sets_abs*** e ***Abs_Recent_Form*** (0.87), o que indica que estas variáveis estão altamente relacionadas. Assim, jogadores com maior percentagem de vitórias tendem também a ganhar uma maior proporção de *sets*. Com isto, foi necessário avaliar a possibilidade de

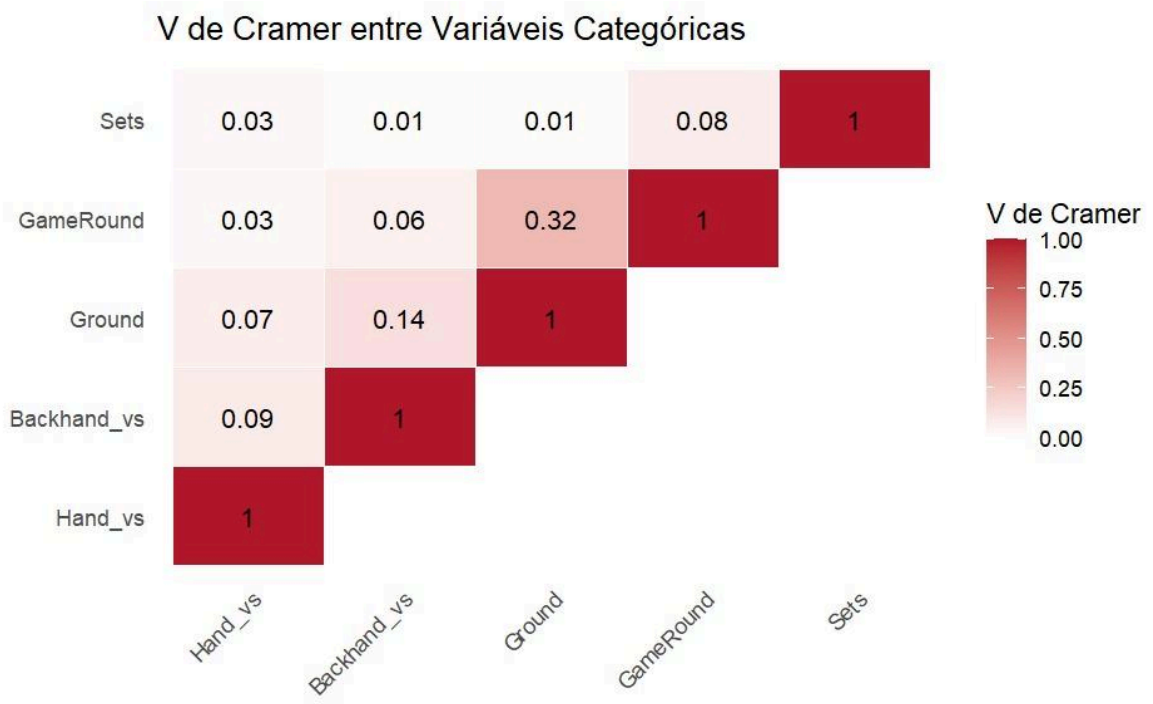
multicolinearidade entre estas duas variáveis na modelação, podendo justificar a exclusão de uma delas.

- Correlação moderada negativa entre ***Difference_ranks_Gap*** e ***Prize_2025*** (-0.37), ou seja, quanto maior a diferença de *rankings* entre os jogadores, menor o valor do prémio com a inflação estabelecida em 2025, o que pode refletir uma lógica competitiva esperada.
- Correlação moderada entre ***Percentagem_Victory_abs*** e ***ground_wins_abs*** (0.30), indicando que os jogadores com maior percentagem absoluta de vitórias tendem a ter melhor desempenho em determinados tipos de piso.
- Correlações fracas ou inexistentes na maioria das restantes variáveis, o que sugere uma fraca ou inexistente relação linear.

3.4.2. V de Cramer

Usado para avaliar a associação entre variáveis categóricas, com valores entre 0 e 1. Quanto mais próximos de 1, mais forte é a associação.

Pode-se observar na figura abaixo a matriz de V de *Cramer*:



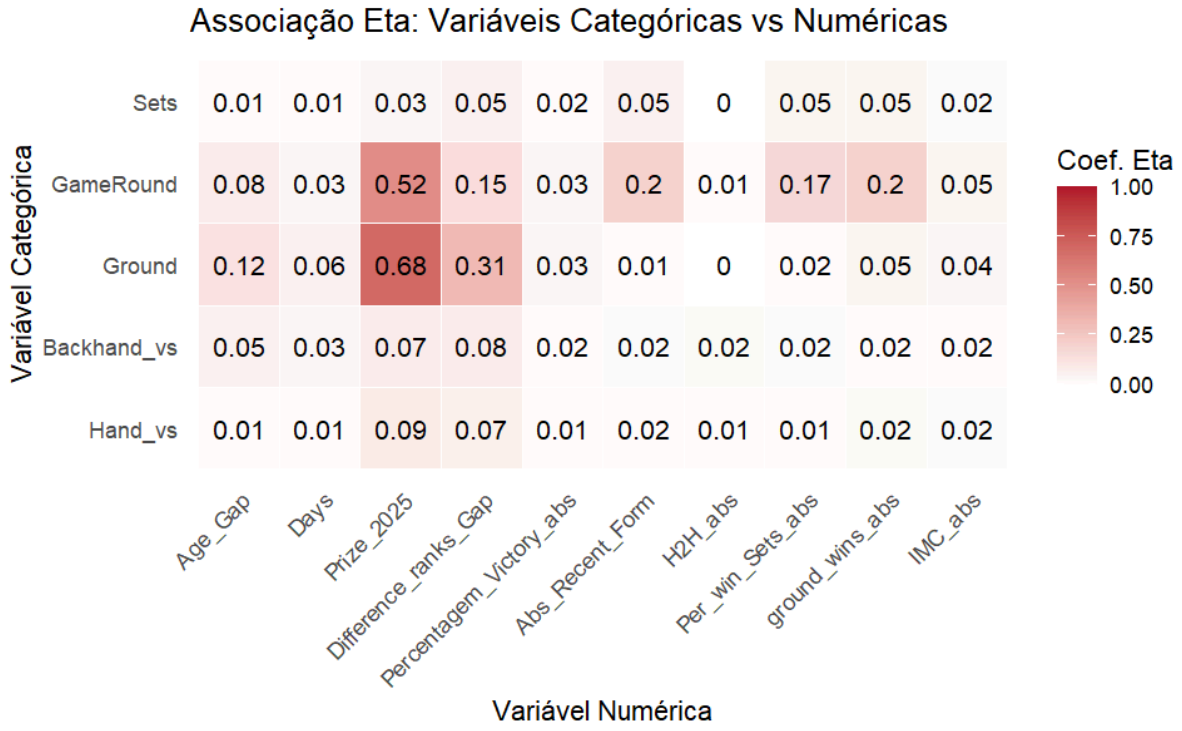
Na matriz apresentada acima, podemos destacar as seguintes observações:

- Associações fracas ou quase inexistentes na maioria das variáveis categóricas, pois a maior parte dos valores encontra-se abaixo de 0.15. Por exemplo, a *target* (***sets***) apresenta um valor muito baixo com todas as outras variáveis, sendo o valor de correlação mais alto entre ***GameRound*** (0.08). Isto diz-nos que o número de *sets* não está associado diretamente a nenhuma das variáveis categóricas analisadas.
- Associação mais forte entre ***GameRound*** e ***Ground*** (0.32), embora ainda moderada. Isto pode refletir que certas rondas de torneios são mais comuns em determinados pisos.

3.4.3. Coeficiente ETA

Utilizado para analisar a relação entre uma variável categórica e uma variável contínua. Valores mais próximos de 1 significam uma maior associação.

Pode-se observar na figura abaixo a matriz de associação ETA:



Na matriz apresentada acima, podemos destacar as seguintes observações:

- A maioria das associações entre variáveis categóricas e numéricas apresenta coeficientes ETA baixos (inferiores a 0.15), indicando que, de forma geral, existe pouca variabilidade explicada pelas variáveis categóricas em relação às numéricas.

- A maior associação da matriz ocorre entre a variável ***Ground*** e ***Prize_2025***, com um coeficiente ETA de 0.68. Este valor indica uma associação forte, sugerindo que a diferença de *rankings* entre os jogadores está relacionada com o valor do prémio com a inflação deste ano. Isto era de esperar, visto que os torneios mais célebres sediados na Bélgica são de solo duro, logo são os que contêm maior valor do prémio.
- De seguida temos a correlação entre ***GameRound*** e ***Prize_2025*** cujo valor é de 0.52, indicando uma correlação moderada. Isto sugere que, à medida que os jogadores avançam nas fases do torneio, o valor do prémio com a inflação de 2025 tende a ser maior, o que é de esperar.
- Verifica-se uma associação moderada entre ***Ground*** e ***Difference_ranks_Gap*** (0.31), o que sugere que o tipo de piso pode estar associado à diferença de *ranking* entre os jogadores. Ou seja, determinados pisos podem favorecer jogadores de *ranking* mais elevado.
- Outras associações que contêm um valor de 0.20 incluem ***GameRound*** com ***Abs_Recent_Form*** (0.20) e com ***ground_wins_abs*** (0.20), sugerindo que a ronda do torneio pode estar associada ao desempenho recente dos jogadores e ao seu histórico de vitórias no piso.

3.4.4. Multicolinearidade entre as variáveis

Foi realizada a análise da multicolinearidade entre as variáveis utilizando o VIF (*Variance Inflation Factor*).

	GVIF Corrigido
IMC_abs	1.00
Hand_vs	1.01
Backhand_vs	1.02
Age_Gap	1.01
Days	1.00
Prize_2025	1.87
Ground	1.32
GameRound	1.05
Difference_Ranks_Gap	1.14
Percentagem_Victory_abs	1.05
Abs_Recent_Form	1.97
H2H_abs	1.00
Per_win_Sets_abs	1.96
Ground_wins_abs	1.09

Os resultados indicaram que nenhum dos valores ultrapassou o limiar de 5, sugerindo que não há problemas significativos de multicolinearidade no conjunto de variáveis selecionadas.

3.5. Imputação da média ou moda

Para lidar com os valores omissos no nosso *dataset*, optámos por substituí-los pela moda nas variáveis categóricas e pela média nas variáveis numéricas.

Reconhecemos que, em alguns casos, especialmente nas variáveis de forma e histórico dos jogadores que desenvolvemos, esta estratégia pode não ser a mais adequada, devido à elevada quantidade de valores em falta. Ainda assim, decidimos avançar com esta abordagem para podermos incluir essas variáveis na modelação, conscientes das suas limitações.

3.6. Equilíbrio das classes

Após a fase de limpeza, o nosso *dataset* ficou com 5647 observações, das quais 3866 correspondiam a jogos terminados em 2 *sets* e 1781 a jogos terminados em 3 *sets*. Esta distribuição revela um desbalanceamento acentuado entre as classes, com os jogos com 2 *sets* a representarem cerca de 68,4% do total e os jogos com 3 *sets* apenas 31,6%.

Este tipo de desequilíbrio pode afetar negativamente o desempenho dos modelos de classificação, levando-os a favorecer a classe maioritária (2 *sets*) e a falhar mais frequentemente na previsão dos casos menos comuns (3 *sets*).

Uma das formas mais comuns de resolver este problema seria aplicar *oversampling*, ou seja, gerar novas observações artificiais da classe minoritária para equilibrar o conjunto de dados. No entanto, como o nosso *dataset* já apresentava muitos valores omissos, que foram imputados com médias e modas, considerámos que aplicar *oversampling* poderia introduzir ainda mais ruído e comprometer a qualidade dos dados. A criação de observações sintéticas sobre dados já parcialmente imputados aumentaria o risco do modelo aprender padrões artificiais e pouco representativos da realidade.

Por esse motivo, optámos por aplicar *undersampling* à classe maioritária (2 *sets*), reduzindo o número das suas observações para igualar o da classe minoritária. O *dataset* final utilizado no treino dos modelos passou assim a contar com 1781 observações por classe, fazendo um total de 3562 linhas.

4. Modeling

4.1. Seleção dos algoritmos

Como previamente mencionado, o objetivo deste projeto foi construir um modelo preditivo capaz de prever o número de *sets* necessários para a conclusão de um jogo de ténis profissional à melhor de 3. Trata-se, portanto, de um problema de classificação binária, uma vez que a variável a prever apresenta duas possibilidades (2 ou 3 *sets*).

Para a modelação preditiva foram utilizados quatro algoritmos de classificação:

- **Regressão Logística**: é um modelo estatístico simples, amplamente utilizado em problemas de classificação binária, como é o caso. Além de apresentar bom desempenho, permite interpretar o impacto de cada variável na probabilidade de um jogo terminar em 2 ou 3 *sets*.
- **Árvore de Decisão**: este algoritmo facilita a interpretação dos resultados, ao apresentar de forma hierárquica os critérios mais relevantes para a classificação, podendo ajudar a identificar padrões que distinguem jogos de 2 *sets* de jogos que exigem 3 *sets* para serem concluídos.
- ***Random Forest**:* Este método combina várias árvores de decisão, cada uma treinada com subconjuntos aleatórios dos dados. Ao reunir os resultados de múltiplos modelos, consegue-se uma previsão mais estável e precisa, reduzindo o risco de *overfitting*. Esta estratégia é conhecida como *bagging* (do inglês bootstrap aggregating). Além disso, o algoritmo fornece uma estimativa da importância de cada variável, permitindo identificar quais os fatores que mais contribuem para prever o número de *set*
- ***Gradient Boosting*:** Este algoritmo pertence à família dos métodos *boosting*, que procuram melhorar o desempenho de modelos simples, como pequenas árvores de decisão, combinando-os de forma sequencial. Ao contrário do *Random Forest*, que cria várias árvores independentes de forma aleatória (*bagging*), o *Gradient Boosting* constroi cada nova árvore com base nos erros cometidos pelas anteriores. A cada iteração, o modelo tenta corrigir os erros anteriores, ajustando os seus parâmetros para melhorar gradualmente a performance. O objetivo principal é reduzir o erro residual a cada passo, resultando num modelo final mais preciso e eficiente.

A avaliação dos modelos foi feita com base na respetiva matriz de confusão e em métricas de desempenho, permitindo comparar a eficácia de cada algoritmo

4.2. Divisão dos Dados - Treino e Teste

Devido à dimensão reduzida do nosso *dataset* final, foi utilizada a técnica de *cross validation* usando a metodologia do *k-fold*, com $k = 5$.

4.3. *Tuning* de hiperparâmetros

Os modelos de *Machine Learning* dependem de hiperparâmetros, que são configurações que definem o comportamento do algoritmo durante o treino. A escolha adequada destes hiperparâmetros é fundamental para obter um bom desempenho, mas normalmente não é possível determinar a melhor combinação sem testar várias possibilidades.

Para otimizar esses valores, utilizámos a função *GridSearchCV* da biblioteca *sklearn*. Esta ferramenta realiza uma busca exaustiva sobre diferentes combinações de hiperparâmetros, permitindo encontrar a configuração que maximiza a performance do modelo. Desta forma, garantimos não apenas a escolha do melhor modelo, mas também a sua melhor configuração para alcançar o máximo potencial.

4.4. Resultados dos diferentes modelos

O critério escolhido para avaliar o desempenho dos modelos foi a *accuracy*, pois o principal objetivo deste projeto era garantir que o modelo fizesse previsões corretas de forma geral, independentemente da classe. Diferentemente de métricas como *recall*, *precision* ou *F1-Score*, que focam mais na identificação correta de uma classe específica ou penalizam certos tipos de erro, a *accuracy* mede a percentagem total de previsões corretas, considerando tanto jogos de 2 *sets* quanto de 3 *sets*.

Como nenhuma das classes ou tipos de erro tinha um impacto claramente mais relevante, não fazia sentido atribuir pesos diferentes a acertos ou falhas em categorias específicas. O objetivo principal era construir um modelo equilibrado e confiável, que acertasse o maior número possível de previsões no geral.

Por isso, a *accuracy* foi a métrica mais adequada, pois oferece uma visão clara da eficácia global do modelo na tarefa de classificação:

$$Accuracy = \frac{Verdadeiros\ Positivos + Verdadeiros\ Negativos}{Verdadeiros\ Positivos + Verdadeiros\ Negativos + Falsos\ Negativos + Falsos\ Positivos}$$

Abaixo apresentamos as matrizes de confusão e os valores de *accuracy* obtidos por cada modelo.

Regressão Logística:

		Previsto	
		2	3
Real	2	1041	740
	3	901	880

$$Accuracy = 0.5394$$

Árvore de Decisão:

		Previsto	
		2	3
Real	2	904	877
	3	813	968

$$Accuracy = 0.5256$$

Random Forest:

		Previsto	
		2	3
Real	2	922	859
	3	766	1015

$$Accuracy = 0.5438$$

Gradient Boosting:

		Previsto	
		2	3
Real	2	920	861
	3	779	1002

$$Accuracy = 0.5396$$

Os resultados obtidos mostram que nenhum dos modelos conseguiu atingir uma *performance* verdadeiramente satisfatória, com todas as *accuracies* a ficarem abaixo dos 55%. Isto demonstra que os modelos não foram eficazes e não cumpriram o objetivo principal do projeto, que era prever corretamente o número de *sets* em cada partida.

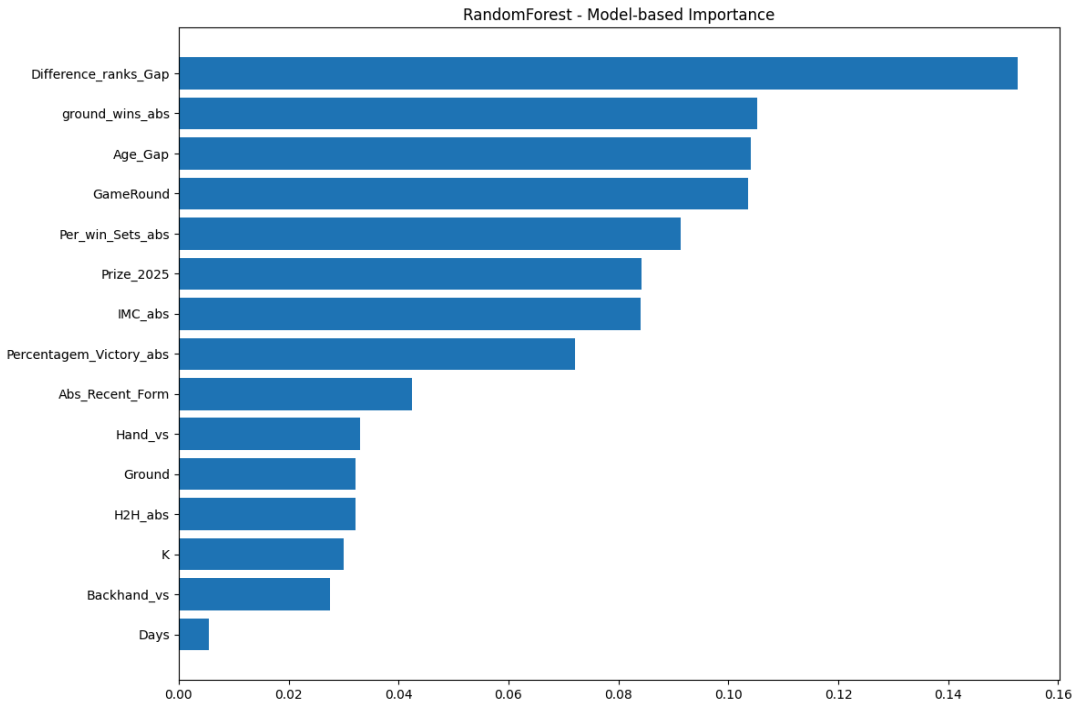
Além disso, os desempenhos registados foram apenas ligeiramente superiores ao acaso, que num problema binário com classes equilibradas, teria uma *accuracy* expectável de 50%.

Ainda assim, dentro das opções testadas, o modelo *Random Forest* destacou-se como o menos mau, obtendo a melhor *performance* (*accuracy* de 0.5438), ainda que com uma margem muito curta face aos restantes modelos.

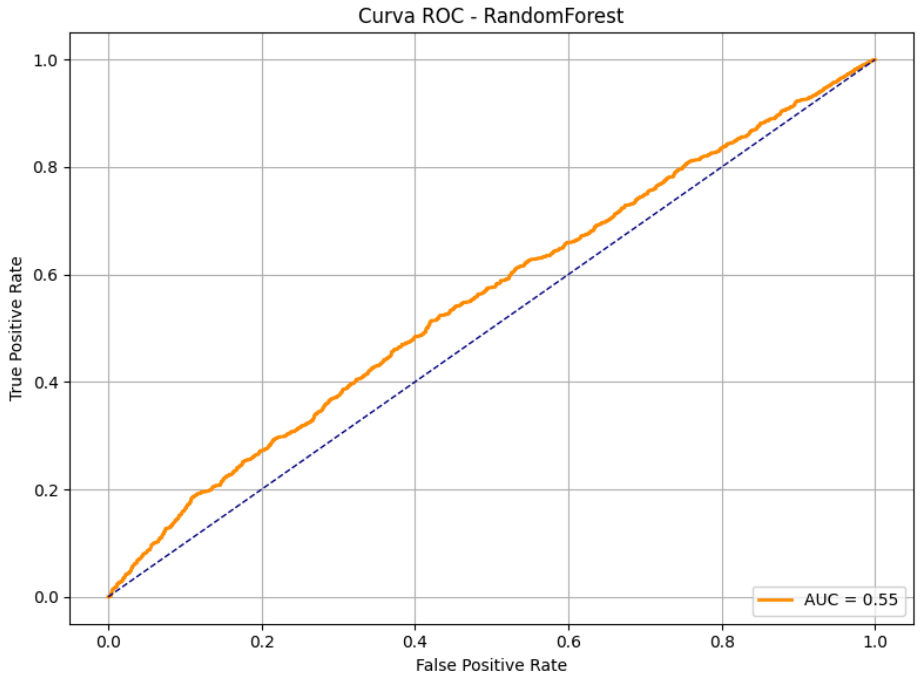
5. Conclusão

Ao longo deste trabalho, foram desenvolvidos diversos modelos de *machine learning* com o objetivo de prever o número de *sets* disputados num jogo de ténis masculino, à melhor de 3, em torneios realizados na Bélgica. Para tal, foi realizada uma extensa fase de preparação dos dados, que incluiu a limpeza, transformação e a criação de novas variáveis.

Apesar da exaustiva preparação dos dados e do esforço aplicado na criação e seleção de variáveis preditoras relevantes, os resultados dos modelos treinados revelaram-se insatisfatórios. Todos os algoritmos testados (Regressão Logística, Árvore de Decisão, *Random Forest* e *Gradient Boosting*) apresentaram valores de *accuracy* baixos, situando-se entre os 52% e os 55%, ou seja, apenas ligeiramente superiores ao que seria obtido por puro acaso. O modelo de *Random Forest* foi aquele que obteve o melhor resultado, embora com uma diferença pequena. Abaixo, apresenta-se o gráfico de importância das variáveis segundo o modelo, evidenciando quais foram os fatores que mais contribuíram para as previsões realizadas.



Adicionalmente, a curva ROC gerada para este modelo revelou uma área sob a curva (AUC) de apenas 0.55, o que indica um desempenho bastante limitado na distinção entre as classes, pouco superior ao valor esperado por puro acaso (0.50).



As variáveis de forma e histórico e o IMC mostraram-se conceptualmente interessantes, mas continham muitos valores omissos que foram imputados, e apesar de algumas delas serem consideradas as mais importantes para o modelo, mesmo assim foram insuficientes para garantir um poder preditivo significativo.

É relevante destacar também que termos usado essas variáveis sabendo deste problema pode não ter sido a melhor opção, mas, devido a limitações de tempo, optámos por incluí-las na mesma para explorar o seu potencial preditivo, reconhecendo, contudo, as limitações associadas a esta decisão.

Além disso, a análise de correlação revelou que nenhuma das variáveis selecionadas apresentava uma associação significativa com a variável *target*. Assim, este desfecho pouco satisfatório já era previsível, uma vez que as variáveis disponíveis não mostraram ser os fatores determinantes para a previsão do número de *sets* num jogo.

Em suma, este trabalho permitiu explorar de forma prática as etapas de pré-processamento, modelação e avaliação de um problema de classificação binária no contexto desportivo, revelando também as dificuldades inerentes à previsão de eventos altamente variáveis como o número de *sets* num jogo de ténis.