Michel de Farias Albuquerque

Aprendizado de Máquinas na previsão de resultados de jogos de futebol

Niterói - RJ, Brasil

15 de janeiro de 2025

Universidade Federal Fluminense

Michel de Farias Albuquerque

Aprendizado de Máquinas na previsão de resultados de jogos de futebol

Trabalho de Conclusão de Curso

Monografia apresentada para obtenção do grau de Bacharel em Estatística pela Universidade Federal Fluminense.

Orientador(a): Prof. Dra. Karina Yuriko Yaginuma

Niterói - RJ, Brasil

15 de janeiro de 2025

Universidade Federal Fluminense

Michel de Farias Albuquerque

Aprendizado de Máquinas na previsão de resultados de jogos de futebol

Profa. Dra. Karina Yuriko Yaginuma
Departamento de Estatística – UFF

Prof. Dr. Hugo Henrique Kegler dos Santos
Departamento de Estatística – UFF

Prof. Dr. Jaime Antonio Utria Valdes
Departamento de Estatística – UFF

Niterói, 15 de janeiro de 2025

Ficha catalográfica automática - SDC/BIME
Gerada com informações fornecidas pelo autor

A345a Albuquerque, Michel de Farias
Aprendizado de Máquinas na previsão de resultados de jogos de futebol / Michel de Farias Albuquerque. - 2025.
54 f.

Orientador: Dra. Karina Yuriko Yaginuma.
Trabalho de Conclusão de Curso (graduação)-Universidade Federal Fluminense, Instituto de Matemática e Estatística, Niterói, 2025.

1. Aprendizado de Máquinas. 2. Aprendizagem Supervisionada. 3. Previsões no Futebol. 4. Produção intelectual. I. Yaginuma, Dra. Karina Yuriko, orientadora. II. Universidade Federal Fluminense. Instituto de Matemática e Estatística. III. Título.

CDD - XXX

Bibliotecário responsável: Debora do Nascimento - CRB7/6368

Resumo

Diante da facilidade atual de obtenção de dados de eventos esportivos e ao crescimento de sites de apostas esportivas no Brasil, o objetivo deste trabalho é encontrar um modelo de Aprendizado de Máquinas que seja capaz de prever os resultados de partidas de futebol. Para isso, são utilizados dados da Premier League, das temporadas 2016/2017 e 2017/2018. Foram aplicados os modelos de Regressão Logística, Redes Neurais Artificiais, Máquina de Vetores de Suporte e XGBoost, buscando verificar o modelo com o melhor desempenho nas previsões. Neste trabalho verificou-se que o modelo com a melhor performance foi o de redes neurais, obtendo uma acurácia de 61,59%.

Palavras-chave: Aprendizado de Máquinas. Aprendizagem Supervisionada. Previsões no Futebol.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução p. 10

1.1 Revisão Bibliográfica . . . . . p. 13

1.2 Objetivos . . . . . p. 16

1.3 Organização . . . . . p. 16

2 Materiais e Métodos p. 17

2.1 Base de Dados . . . . . p. 17

2.2 Pré-Processamento . . . . . p. 19

2.3 Validação Cruzada K-fold . . . . . p. 20

2.4 *Recursive Feature Elimination* (RFE) . . . . . p. 21

2.5 Modelos de Aprendizado de Máquina Supervisionados . . . . . p. 22

2.5.1 Regressão Logística . . . . . p. 23

2.5.2 Redes Neurais Artificiais . . . . . p. 24

2.5.3 *Support Vector Machine* (SVM) . . . . . p. 26

2.5.4 XGBoost . . . . . p. 28

2.6 Matriz de Confusão e Medidas de Desempenho . . . . . p. 28

3 Análise dos Resultados p. 31

3.1 Análise Descritiva . . . . . p. 33

3.2 Seleção de Variáveis . . . . . p. 38

3.3 Resultados . . . . . p. 40

3.3.1 Caso 1 - Acurácia comparando dois campeonatos contra um . . p. 41

3.3.2 Caso 2 - Acurácia comparando a variável Resultado com três classes contra duas . . . . . p. 43

3.3.3 Caso 3 - Acurácia comparando os resultados em relação às médias das partidas anteriores . . . . . p. 45

3.4 Melhor Modelo: Redes Neurais . . . . . p. 46

4 Conclusões p. 50

Referências p. 52

Lista de Figuras

1 Exemplo dos dados obtidos a partir do site FBREF(Fonte: fbref.com) . p. 17

2 Exemplo dos dados utilizados no trabalho . . . . . p. 18

3 Melhor curva através da Função de Verossimilhança (Fonte: youtube.com/@statquest) . . . . . p. 24

4 Composição de uma Rede Neural Artificial (Fonte: opencadd.com.br/o-que-sao-redes-neurais/) . . . . . p. 25

5 Rede Neural Artificial para Classificação Multiclasse (Fonte: rpubs.com/jessicakubrusly/redes-neurais-6) . . . . . p. 26

6 Gráfico de um modelo SVM para classificação binária (Fonte: analytics-vidhya. com/) . . . . . p. 27

7 Transformação não-linear onde o SVM linear não se aplica diretamente (Fonte: commons.wikimedia.org/wiki/File:Kernel-Machine.png) . . . . p. 27

8 Gráficos com as frequências absoluta e relativa das 3 classes da variável alvo Resultado com a base contendo os dois campeonatos) . . . . . p. 33

9 Boxplots das variáveis com aparente relação com a variável Resultado. . p. 34

10 Boxplots das variáveis sem relação aparente com a variável Resultado. . p. 35

11 Gráficos com a frequência absoluta e a porcentagem das duas classes da variável alvo Resultado com a base contendo os dois campeonatos) . . . p. 36

12 Gráficos com a frequência absoluta e a porcentagem das três classes da variável alvo Resultado com a base contendo um campeonato) . . . . . p. 37

13 Gráficos com as frequências absoluta e relativa das duas classes da variável alvo Resultado com a base contendo um campeonato) . . . . . p. 37

14 Gráfico que mostra a acurácia dos diferentes modelos gerados pela função RFE assim como número de variáveis presentes nos mesmos. . . . . p. 38

15 Importância das variáveis . . . . . p. 40

16 Gráficos com a comparação da acurácia dos modelos gerados a partir de 1 e 2 campeonatos . . . . . p. 43

17 Gráficos com a comparação da acurácia dos modelos gerados utilizando três e duas classes . . . . . p. 45

18 Gráficos da acurácia média em relação ao tamanho da base . . . . . p. 45

19 Gráficos da acurácia média em relação ao tamanho da base . . . . . p. 46

Lista de Tabelas

1 Variáveis e suas descrições (médias se referem aos últimos 3 jogos) . . . p. 19

2 Validação Cruzada K-Fold com 5 folds . . . . . p. 21

3 Probabilidades de ocorrência das classes . . . . . p. 23

4 Matriz de Confusão Binária . . . . . p. 29

5 Matriz de Confusão Multiclasse . . . . . p. 30

6 Matriz de Confusão Binária para Medidas de Desempenho . . . . . p. 30

7 Exemplo de aplicação da metodologia . . . . . p. 31

8 Variáveis selecionadas e suas descrições . . . . . p. 39

9 Acurácia dos modelos para os campeonatos de 2016/2017 e 2017/2018 . p. 41

10 Sensibilidade e Especificidade dos Modelos . . . . . p. 42

11 Acurácia dos modelos com o campeonato de 2017/2018 . . . . . p. 42

12 Sensibilidade e Especificidade dos modelos a partir de um campeonato p. 43

13 Acurácia dos modelos para os campeonatos de 2016/2017 e 2017/2018 para duas classes . . . . . p. 44

14 Sensibilidade e Especificidade dos modelos . . . . . p. 44

15 Valores dos hiperparâmetros que geraram o melhor modelo de Rede Neural p. 48

16 Comparação entre os resultados originais e previstos finais . . . . . p. 48

1 Introdução

Não é novidade a paixão do povo brasileiro pelo futebol. O esporte é assistido, lido, escutado e conversado em lares, trabalho, transportes coletivos, bares, restaurantes e estádios por milhões de brasileiros que diariamente buscam informações sobre seu time de coração ou rivais.

Essa adoração pelo jogo não é exclusiva do brasileiro. O futebol foi criado oficialmente, com um modelo semelhante ao praticado hoje, pelos ingleses em 1863, a partir da criação da *Football Association*, padronizando regras e uniformizando o modo de jogar, além de criar as primeiras competições. Desde então a popularidade do jogo só cresceu, sendo hoje o esporte mais acompanhado do mundo, como mostra o site Penalty [1], obtendo índices consideráveis de crescimento até em países onde há pouco tempo o futebol era algo isolado, como Estados Unidos e a China.

Com essa popularidade, gerando níveis altos de audiência, o ambiente do futebol passou de um mero jogo entre 22 jogadores para um evento que necessita de milhares de pessoas para acontecer. De acordo com a matéria do site SportsJob [2], o esporte envolve hoje profissionais de outras áreas como por exemplo medicina, nutrição e psicologia para o bem estar dos atletas, engenharia na produção de equipamentos e infraestrutura que permitem o aumento do nível dos jogos, e marketing na exploração do lado comercial.

Uma outra área que está sendo explorada no futebol é a estatística. Com a evolução da capacidade computacional, a área vem sendo utilizada tanto por clubes quanto pela mídia e torcedores. Ao se reunir informações de jogadores, clubes e competições numericamente, a utilização de métodos estatísticos permite o entendimento e dá mais credibilidade a essas informações, que antes eram passadas subjetivamente.

Outra aplicação da área, vinda da modelagem estatística, é a previsão de resultados de eventos esportivos. Através do aprendizado de máquinas, se torna cada vez mais

[1] https://www.penalty.com.br/blog/post/os-esportes-mais-populares-do-mundo
[2] https://sportsjob.com.br/18-areas-para-trabalhar-no-futebol/

comum a utilização dos modelos com o intuito de prever os resultados de partidas e campeonatos. Isso porque um outro ramo que se popularizou na última década foi a de apostas esportivas.

Em evidência e com grande projeção de crescimento e arrecadação, vão surgindo cada vez mais empresas desta área. No Brasil por exemplo, são mais de 450 sites de apostas esportivas com expectativas de um faturamento de 100 bilhões de reais entre 2024 e 2026, segundo o site Poder360[3].

Com uma grande quantidade de concorrentes, as empresas buscam se diferenciar no mercado tentando expor cada vez mais suas marcas. Com isso, aliada a dificuldade financeira dos clubes, entraram diretamente no futebol ao virarem patrocinadoras de times. De acordo com a revista Exame [4] por exemplo, todos os 20 clubes da primeira divisão do campeonato brasileiro de futebol em 2023 tinham ao menos um patrocínio de sites de apostas.

Isso faz com que os times tenham mais dinheiro para investirem em elencos mais fortes e melhorar sua infraestrutura, os campeonatos tenham a capacidade de oferecerem mais dinheiro aos participantes, aumentando assim o interesses dos clubes neles, e com isso gerando maior competitividade e maior audiência.

Se antes o mercado de patrocínios no futebol era mais diversificado, onde se viam bancos, montadoras de carros, operadoras de planos de saúde, telecomunicações e empresas do setor de energia; hoje o setor é dominado pelas casas de apostas. Segundo o site Poder360 [5], o valor em patrocínios para o ano de 2023 chega aos 327 milhões de reais.

Também há um outro lado, onde as apostas atrapalham o esporte por abrirem margens para dúvidas nos resultados. Já foram vistos escândalos no futebol envolvendo times e jogadores que agiam na tentativa de manipular os resultados em prol de apostas feitas por terceiros, como aconteceu em 2023 no campeonato brasileiro segundo o jornal Estadão [6].

Isso se dá por falta de clareza de como as empresas funcionam e também de uma regulamentação para o setor no país, fazendo com que tudo seja feito livremente sem qualquer regra ou fiscalização. Porém com a regulamentação sancionada pelo governo brasileiro em dezembro de 2023, com implementações de regras de atuação, fiscalização e

[3]https://www.poder360.com.br/economia/mercado-de-apostas-on-line-deve-crescer-a-partir-de-2024/

[4]https://exame.com/casual/todos-os-20-times-da-serie-a-tem-sites-de-apostas-esportivas-como-patrocinadores/

[5]https://www.poder360.com.br/opiniao/casas-de-apostas-investem-r-327-milhoes-no-brasileirao/

[6]https://www.estadao.com.br/esportes/futebol/esquema-de-apostas-saiba-tudo-escandalo-envolve-futebol-brasileiro-npres/

combate à ilegalidade, como mostra o site IDWALL [7], o cenário tende a melhorar.

De acordo com o site SitPicks [8], 5% dos apostadores a longo prazo conseguem lucro. E devido a essa alta incerteza e dificuldade de se acertar resultados de jogos de futebol, um meio que vem sendo explorado na tentativa de conseguir uma melhora no desempenho é a utilização da estatística e da computação, através do aprendizado de máquinas e seus modelos de previsão.

Com o constante desenvolvimento computacional, tanto em relação à performance quanto a acessibilidade, torna-se cada vez mais fácil obter material sobre qualquer tipo de assunto, incluindo o futebol. Como os modelos de aprendizado de máquina supervisionados dependem de dados confiáveis para obterem resultados satisfatórios, o acesso a informações proporciona a criação de banco de dados com diversas variáveis sobre campeonatos, times e jogadores de todas as partes do mundo.

Para este trabalho foram utilizados dados da primeira divisão do campeonato inglês de futebol, a Premier League. Fundada em 1992, substituindo o Campeonato Inglês, o novo formato do futebol inglês tinha o objetivo de devolver à Inglaterra a fama de ter um dos melhores campeonatos de futebol europeu, que ocorreu na década de 70 mas foi perdida na de 80 por conta da precariedade dos estádios e clubes aliada ao vandalismo das torcidas.

Desde a sua fundação a liga vem em constante crescimento, e hoje é considerada a maior e melhor liga do mundo, sendo o campeonato de maior audiência entre todos os esportes em 2022 no mundo segundo apurou o site Poder360 [9], a frente da La Liga (primeira divisão do campeonato espanhol de futebol) e da NFL (elite do futebol americano dos Estados Unidos), sendo referência para os diversos torneios ao redor do mundo. Um fato interessante é o crescimento da audiência da liga até em países onde o futebol não é tão popular, como aconteceu nos Estados Unidos em 2023 e 2024, quando houve a maior audiência do campeonato da história segundo o site SBJ[10].

Em relação à liga brasileira, as grandes vantagens da inglesa são a melhor estrutura dos clubes, permitindo que os times encontrem cenários mais uniformes, menos casos de interferência externa como o time levar punições por incidentes causados pela torcida, e um calendário melhor planejado, permitindo que os times tenham tempo para descansar

[7] https://blog.idwall.co/regulamentacao-das-apostas-esportivas-no-brasil/.
[8] https://sitpicks.com/what-percentage-of-sports-bettors-win/.
[9] https://www.poder360.com.br/opiniao/audiencia-global-do-esporte-explode-no-mundo-digital/
[10] https://www.sportsbusinessjournal.com/Articles/2024/05/23/nbc-premier-league-viewership-best-in-united-states

e treinar entre as partidas, além de remarcação e remanejamento de jogos em caso de necessidade.

1.1 Revisão Bibliográfica

O trabalho de (BABOOTA; KAUR, 2018) utiliza dados da Premier League, totalizando 11 campeonatos diferentes como fonte de dados, de 2005 até 2016. Segundo o autor, o que fará com que os modelos criados tenham um bom resultado será a forma com que as variáveis obtidas são manipuladas e expostas na geração dos modelos de previsão.

Além de variáveis de estatísticas de jogos, como número de chutes a gol e escanteios, utilizam algumas variáveis de cunho subjetivo, como a classificação do nível de cada setor (defesa, meio de campo e ataque). Ao todo o trabalho começa com 33 variáveis, porém, para modelos que necessitam, são utilizados processos de seleção de variáveis, excluíndo variáveis menos importantes, e outras que eram correlacionadas foram agrupadas formando novas.

As variáveis são divididas em dois grupos, um contendo valores para estatísticas de mandantes e visitantes de cada jogo, e outro com as diferenças dessas estatísticas para cada partida, onde em ambas guardam dados dos últimos 6 jogos dos times. Os modelos preditivos utilizados foram *Naive Bayes*, Florestas Aleatórias, *Support Vector Machine* e *Gradient Boosting*, sendo os dois últimos os que obtiveram os melhores resultados, com 58% de acurácia. Os dados de treino foram os 9 primeiros anos e os de teste os 2 últimos.

Em (HUCALJUK; RAKIPOVIC, 2011a) o objetivo é prever resultados na fase de grupos da Liga dos Campeões da Europa. Para isso, foram utilizados dados de um ano do torneio, com 30 variáveis, que foram reduzidas para 20 ao longo do processo de treinamento. Além de variáveis de estatísticas dos jogos, os autores também utilizam algumas variáveis subjetivas para classificar os times quanto aos seus diferentes níveis. São exemplos de variáveis utilizadas o número médio de gols feitos e concedidos, posição na liga, número de jogadores machucados e a fase do time ao se analisar os últimos 6 jogos.

Os autores relatam que a qualidade do resultado passa por como as variáveis serão apresentadas aos modelos preditivos, então foram realizados diversos testes afim de encontrar o efeito que cada variável tinha na acurácia do modelo sendo elas apresentadas de diferentes modos. Um exemplo do que foi feito é de que inicialmente se tinha a ideia de que a variável contendo o número de pontos dos últimos 6 jogos do time teria grande impacto no modelo, porém se obteve um melhor resultado ao separá-la em outras três:

número de vitórias, derrotas e empates.

Foram utilizados os seguintes modelos preditivos para efeito de comparação: *Naive Bayes*, Redes Bayesianas, *LogitBoost*, K-vizinhos mais próximos, Redes Neurais e Florestas Aleatórias, sendo o de Redes Neurais o que obteve o melhor desempenho (acurácia de 60%). Vale ressaltar o modo como foram feitos os treinamentos dos modelos, onde foram utilizados diferentes tamanhos para as bases de treino e teste, além de validação cruzada. A fase de grupos contém 6 rodadas, cada uma contendo 16 jogos, e as bases de treino tiveram de 3 a 5 rodadas; as bases de teste contém as rodadas restantes afim de comparar ao final a diferença de acurária entre os diferentes modos de análise.

Em (JOSEPH; FENTON; NEIL, 2006), por outro lado, os autores utilizaram em seu trabalho a análise de apenas um time, o Tottenham, ao longo de duas temporadas (1995/1996 e 1996/1997). Duas temporadas pois, segundo os autores, os times se modificam constantemente com trocas de jogadores e portanto este é um limite adequado.

As variáveis utilizadas envolvem as escalações do time para cada jogo, atribuindo um nível de qualidade para ela e verificando se os principais jogadores estarão presentes na partida, realizando uma comparação subjetiva entre os níveis das escalações do Tottenham e de seu adversário.

Embora o foco seja avaliar a performance das Redes Bayesianas, outros modelos foram utilizados para comparação: Naive Bayes, Árvore de Decisão, variações das Redes Bayesianas e KNN. A conclusão foi de que as Redes Bayesianas obtiveram um bom resultado, com a acurácia na média dos trabalhos utilizados como referência. Uma observação feita pelos autores era que esperava-se um aumento da acurácia ao aumentar o número de observações dos dados de treino, porém ao realizarem alterações nos dados o resultado foi o oposto. Indicando que os dados incluídos provavelmente fugiam do padrão do restante das observações presentes anteriormente nos dados de treino.

Já em (LIMA, 2022) visa comparar o desempenho de diferentes modelos em dados de diferentes tamanhos de 5 ligas de futebol. Foram utilizados dados das últimas 5 e 10 temporadas de cada uma das seguintes ligas: brasileira, inglesa, francesa, italiana e espanhola.

As variáveis utilizadas foram as chances (probabilidades) de vitória, empate e derrota para diferentes casas de apostas em relação a cada uma das partidas de cada um dos campeonatos. Terminada a montagem da base de dados, foi estabelecida uma divisão de 80% para dados de treino e 20% pra testar os modelos, tanto ao se trabalhar com 5

temporadas quanto com 10.

Os modelos de previsão utilizados foram: Naive Bayes, Regressão Logística Binomial e Multinomial, SVM e Árvore de Decisão. O melhor modelo foi o estimado pela Regressão Logística, que na grande maioria dos casos obteve o melhor de desempenho tanto em relação aos diferentes campeonatos quanto ao se comparar os dados com 5 ou 10 temporadas. A acurácia do modelo girou em torno dos 60%.

O autor (ARAUJO et al., 2018) obtém um banco de dados referentes a diferentes campeonatos europeus, com um total de 58 mil observações, e a partir deles tem como objeto prever resultados de partidas de futebol que estão em andamento, recebendo dados do primeiro tempo para prever o final.

A variáveis utilizadas foram gols feitos pelos times visitante e mandante ao final do primeiro tempo de jogo, média da diferença entre gols feitos e gols sofridos pelos dois times (visitante e mandante) e o aproveitamento (porcentagem de pontos ganhos) de ambos. Sobre os modelos preditivos escolhidos, foram 8 no total, com destaque para Regressão Logística, Árvore de Decisão, Florestas Aleatórias e AdaBoost, que obtiveram os melhores resultados. Um detalhe para o treinamento do modelo, onde 98% dos dados foram alocados para os dados de treino e 2% para o teste, e de forma aleatória. Foram realizados 3 testes para cada um dos modelos, a partir de uma randomização dos dados que altera então a ordem das observações, fazendo com que os modelos fossem alimentados de diferentes formas. O modelo que obteve os melhores resultados foi de Regressão Logística, com uma média de 62% de acurácia.

Já (SCHNEIDER, 2018) utiliza dados do campeonato inglês de 2002 até 2017, totalizando mais de 6000 observações (partidas). Utiliza além de variáveis contendo estatísticas da partidas, variáveis subjetivas para classificar o nível dos times após cada um dos jogos, totalizando 21 variáveis.

Sobre os modelos escolhidos, estão entre eles: Regressão Logística, KNN, SVM, Naive Bayes Gaussiano e Multinomial. Para o treinamento dos modelos, os dados foram separados entre campeonatos de 2002 até 2015 (até 2014/2015) para treino e 2015 (2015/2016) e 2017 para teste. Vale ressaltar que, em cada uma das temporadas, foram testadas as exclusões das primerias 3, 10 e 19 partidas, afim de avaliar a aplicação fórmulas para a obtenção de variáveis.

Também são utilizadas técnicas de validação cruzada e otimização de hiperparâmetros afim de buscar uma maior acurácia para os modelos. Os que obtiveram os melhores

resultados foram Regressão Logística e SVM, com 54% de acurácia.

1.2 Objetivos

O objetivo do trabalho é, a partir de dados da Premier League, realizar a previsão de partidas de futebol utilizando modelos de aprendizado de máquina, buscando atingir resultados, no mínimo, semelhantes aos obtidos em trabalhos similares, na casa dos 60% de acurácia.

Além disso, identificar possíveis alterações que gerem aumento de acertos do modelo, como modificações nas variáveis e seleção das que geram valores maiores de acurácia dos modelos.

1.3 Organização

A apresentação do trabalho está organizada em três partes. O Capítulo 2 detalhando os materiais e modelos de aprendizado de máquina utilizados. O Capítulo 3 apresentando os resultados gerados em cada situação abordada, apresentando uma análise descritiva dos dados, as variáveis mais importantes para o trabalho e tabelas e gráficos com os resultados de cada um dos modelos com suas respectivas acurácias. E finalizando com a conclusão final no Capítulo 4.

2 Materiais e Métodos

2.1 Base de Dados

Para a realização do trabalho, os dados da Premier League foram obtidos do site FBREF (*Football Reference*), onde os dados estão organizados conforme mostra a Figura 1, através da linguagem R (R Core Team, 2014) com o pacote worldfootballR (ZIVKOVIC, 2022) . Foram coletados dados da temporadas 2016/2017 e 2017/2018, totalizando 760 observações, onde cada uma se refere à um confronto entre dois dos 20 clubes participantes em cada edição. A ideia de duas temporadas se dá por conta da falta de informação de quantos campeonatos são necessários para obter melhores resultados. Esses testes são realizados durante o trabalho.

Aston Villa Player Stats Glossary

Summary | Passing | Pass Types | Defensive Actions | Possession | Miscellaneous Stats

						Performance											
Player	#	Nation	Pos	Age	Min	Gls	Ast	PK	PKatt	Sh	SoT	CrdY	CrdR	Touches	Tkl	Int	Blocks
Wesley Moraes	9	BRA	FW	23-018	68	0	0	0	0	1	0	0	0	27	0	0	1
Jonathan Kodjia	26	CIV	FW	30-053	22	0	0	0	0	0	0	0	0	5	0	0	0
Jack Grealish	10	ENG	LW	24-095	90	0	0	0	1	3	0	0	0	43	0	0	1
Anwar El Ghazi	21	NED	RW	24-225	79	0	0	0	0	1	0	0	0	31	1	0	0
Trézéguet	17	EGY	RW	25-074	11	0	0	0	0	0	0	0	0	2	0	0	0
Henri Lansbury	8	ENG	LM	29-063	65	0	0	0	0	0	0	0	0	31	0	0	1
Douglas Luiz	6	BRA	LM	21-219	25	0	0	0	0	0	0	0	0	16	1	0	0
Marvelous Nakamba	11	ZIM	CM	25-329	90	0	0	0	0	1	0	0	0	45	2	2	0
John McGinn	7	SCO	RM	25-057	90	0	0	0	0	0	0	0	0	53	2	1	1
Matt Targett	18	ENG	LB	24-087	90	0	0	0	0	0	0	1	0	66	0	3	2
Kortney Hause	30	ENG	CB	24-151	90	0	0	0	0	1	0	1	0	79	1	0	3
Björn Engels	22	BEL	CB	25-090	90	0	0	0	0	0	0	0	0	54	2	4	0
Frederic Guilbert	24	FRA	RB	24-355	90	0	0	0	0	0	0	0	0	75	2	3	1
Tom Heaton	1	ENG	GK	33-243	90	0	0	0	0	0	0	0	0	39	0	0	0
14 Players					**990**	**0**	**0**	**0**	**1**	**7**	**0**	**2**	**0**	**566**	**11**	**13**	**10**

Figura 1: Exemplo dos dados obtidos a partir do site FBREF(Fonte: fbref.com)

Sobre as variáveis, no banco de dados inicial há 90 variáveis, sendo que algumas registram informações gerais das partidas, como nomes dos times e rodada da partida; e

outras informam o que ocorreu durante os jogos, como número de gols das equipes, posse de bola e defesas dos goleiros.

Vale ressaltar que como os jogos envolvem duas equipes, então as variáveis que se referem à estatísticas dos jogos aparecem duas vezes, informando um valor para equipe que joga em casa (mandante) e outro para a equipe que joga fora de casa (visitante). Por exemplo, para registrar o número de finalizações em um jogo, há uma variável para o número de chutes do time mandante e uma variável com essa mesma informação, agora para a equipe visitante. A Figura 2 mostra um exemplo da organização dos dados utilizados no trabalho no software RStudio. Utilizando a linha 1 como referência, está representado o time mandante na variável *Home Team*, Arsenal, e o visitante pela *Away Team*, Leicester City, o número de gols do time mandante com a variável *Home Score* e o número de gols do visitante pela *Away Score* por exemplo.

	Home_Team	Away_Team	Home_Score	Away_Score	Home_Gls	Home_Ast	Home_PK	Home_PKatt	Home_Sh	Home_SoT
1	Arsenal	Leicester City	4	3	4	4	0	0	27	10
2	Watford	Liverpool	3	3	3	1	0	0	9	4
3	Crystal Palace	Huddersfield Town	0	3	0	0	0	0	14	4
4	West Bromwich Albion	Bournemouth	1	0	1	1	0	0	16	6
5	Chelsea	Burnley	2	3	2	2	0	0	19	6
6	Everton	Stoke City	1	0	1	1	0	0	9	4
7	Southampton	Swansea City	0	0	0	0	0	0	29	2
8	Brighton & Hove Albion	Manchester City	0	2	0	0	0	0	6	2
9	Newcastle United	Tottenham Hotspur	0	2	0	0	0	0	6	2
10	Manchester United	West Ham United	4	0	4	4	0	0	21	5
11	Swansea City	Manchester United	0	4	0	0	0	0	5	0
12	Bournemouth	Watford	0	2	0	0	0	0	6	2
13	Southampton	West Ham United	3	2	3	1	2	2	12	3
14	Leicester City	Brighton & Hove Albion	2	0	2	1	0	0	14	4
15	Burnley	West Bromwich Albion	0	1	0	0	0	0	20	0

Figura 2: Exemplo dos dados utilizados no trabalho

Um ponto importante é que os dados apresentados possuem informações sobre partidas já finalizadas, e portanto, para se prever os resultados de um jogo futuro, que ainda não ocorreu e com isso não se tem informações sobre posse de bola, número de finalizações, números de defesas dos goleiros por exemplo, não haveria como alimentar o modelo para realizar a previsão. Para resolver esse problema, foram consideradas a média dos últimos jogos dos times para cada uma das variáveis.

Agora as variáveis passaram a informar não as estatísticas de certa partida, mas sim como os times chegavam para o próximo duelo de acordo com jogos anteriores. Portanto decidir o intervalo de jogos anteriores para obter a média que gerasse melhores resultados também virou uma questão a ser respondida.

Um exemplo de como as variáveis ficaram utilizando a média das últimas partidas

disputas é mostrada na Tabela 1. Além disso, as variáveis que informavam características gerais dos confrontos foram excluídas, enquanto outras foram adicionadas, incluindo a "Resultado", que é a variável alvo e indica se o vencedor foi o mandante do jogo, o visitante ou se houve empate. Portanto, a base de dados final possui 97 variáveis.

Tabela 1: Variáveis e suas descrições (médias se referem aos últimos 3 jogos)

Variável	Descrição
Resultado	Desfecho da partida: Vitória, Empate ou Derrota do time Mandante
Gols Marcados	Média de gols marcados
Posse de Bola	Média da porcentagem de tempo em que o time teve a bola
Total de Finalizações	Média de tentativas de chutes ao gol do adversário
Chances Claras	Média de oportunidades reais de gol criadas
Dribles	Média de dribles efetuados com sucesso
Passes Certos	Média da porcentagem de passes certos
Escanteios	Média de escanteios cobrados
Faltas	Média de faltas cometidas
Impedimentos	Média de impedimentos marcados contra a equipe
Roubadas de Bola	Média da quantidade de bolas recuperadas
Divididas	Média de disputas ganhas no chão
Interceptações	Média de passes do adversário interceptados
Duelos Ganhos	Média do total de disputas de bola ganhas
Defesas do Goleiro	Média de defesas do goleiro
Pontuação Acumulada	Pontuação acumulada do time ao longo do campeonato
Pontuação3	Pontuação conquistada nas últimas 3 partidas disputadas

2.2 Pré-Processamento

Após obter a base, como nem sempre os dados estão em um formato ideal para aplicação, é necessário prepará-los para a utilização nos modelos de aprendizado de máquinas, com o objetivo de conseguir os melhores resultados possíveis. Essa etapa de preparação é chamada de pré-processamento dos dados.

Nesta etapa são realizadas atividades como limpeza dos dados, com a procura por dados faltantes e duplicados por exemplo; e transformação das variáveis, manipulando-as com o intuito de conseguir adequá-las aos modelos e tentando encontrar a formatação das mesmas que gere o melhor resultado, como agrupamento ou separação de variáveis, exclusão das que contenham informações semelhantes ou que não tenham relação relevante com a variável que se quer estudar e prever, e a padronização, discutida a seguir.

Como em alguns casos os dados numéricos podem estar em diferentes escalas, utilizá-los desta forma nos modelos pode fazer com que eles aprendam mais sobre uma variável,

por ter valores maiores ou variar mais, do que outra, que possua valores menores ou variar menos.

Para contornar esse problema, podem ser utilizadas técnicas de redimensionamento dos dados, como a padronização. Seu intuito é de colocar os dados em escalas parecidas, fazendo com que o modelo possa aprender igualmente sobre todas as variáveis.

No caso, o método é aplicado subtraindo-se o valor de uma variável de uma dada observação (x_i) pela sua média amostral ($\overline{x}$), dividido pelo seu desvio padrão (S), como mostrado na fórmula a seguir:

$$Z_i = \frac{x_i - \overline{x}}{S}. \tag{2.1}$$

2.3 Validação Cruzada K-fold

Após a adequação dos dados, pode-se realizar então o treinamento dos modelos. O processo é iniciado ao se dividir os dados em duas bases: uma de treino e outra de teste. A base de treino, que geralmente contém cerca de 70 a 80% dos dados da base original, é utilizada para realizar o treinamento, ou seja, para estimar os parâmetros de um modelo de acordo com os dados expostos a ele.

Porém na divisão dos dados em treino e teste de forma simples, apenas uma vez, caso os dados de treino não tenham uma diversidade, pode ocorrer um problema onde o modelo gerado a partir deles tenha dificuldade de fazer previsões quando exposto à novos dados. Ele pode gerar resultados muito bons caso os dados de testes sejam parecidos com os utilizados para treinar o modelo, porém podem ser ruins caso sejam diferentes. Para corrigir isso, é utilizado o método de Validação Cruzada K-Fold.

Conforme (HASTIE; TIBSHIRANI; FRIEDMAN, 2009), a ideia do método é fazer com que todo o conjunto de dados participe do treinamento. Isso é feito ao se repetir o processo de divisão dos dados em treino e teste K vezes, criando os chamados Folds, e dividindo os dados em K parcelas dentro deles, onde cada parcela será utilizada para treinamento K-1 vezes. Um exemplo é dado na Figura 2, onde há 5 folds e portanto cada parcela dos dados participará do treinamento 4 vezes. Com toda a base de dados participando do treinamento, e assim expondo o modelo a uma quantidade maior de dados, aumenta-se a chance de obter um modelo que tenha uma maior capacidade de

Iteração 1	Treino	Treino	Treino	Treino	Teste
Iteração 2	Treino	Treino	Treino	Teste	Treino
Iteração 3	Treino	Treino	Teste	Treino	Treino
Iteração 4	Treino	Teste	Treino	Treino	Treino
Iteração 5	Teste	Treino	Treino	Treino	Treino

Tabela 2: Validação Cruzada K-Fold com 5 folds

generalização, obtendo assim melhores resultados diante de novos dados. Deste modo, pode-se passar agora ao treinamento utilizando os modelos.

2.4 *Recursive Feature Elimination* (RFE)

Segundo (PEDREGOSA et al., 2011), a RFE é uma técnica de seleção de variáveis, encontrada no R (R Core Team, 2014) no pacote *caret* (Kuhn; Max, 2008), utilizada para fazer a escolha da melhor combinação de variáveis preditoras a partir de modelos que possuem a capacidade de informar ao usuário a importância das variáveis utilizadas. Exemplos de modelos são as Florestas Aleatórias, a Regressão Simples e o Naive Bayes.

Um ponto que faz os resultados serem melhores do que só observar as importâncias geradas pelos modelos acima é a opção que a função RFE possui de utilizar validação cruzada.

Segundo o *RDocumentation*[1], os passos que o algoritmo segue são:

- Divisão da base em treino e teste, realizando o processo de acordo com o que foi definido na validação cruzada;
- Treinamento do modelo com todas as variáveis;
- Previsão em relação a base teste;
- Cálculo das variáveis mais importantes para o modelo;
- Testes criando modelos com diferentes variáveis e diferentes tamanhos (tamanhos

[1]https://www.rdocumentation.org/packages/caret/versions/6.0-92/topics/rfe

podem ser definidos previamente pelo usuário), e sempre reavaliando ao final a importância das variáveis;

- Finalizadas as combinações possíveis, é estimado o modelo com a melhor combinação de variáveis de acordo com a função, além da apresentação dessas melhores variáveis.

2.5 Modelos de Aprendizado de Máquina Supervisionados

O trabalho utilizará modelos de aprendizagem supervisionada, ou seja, modelos que são treinados a partir de dados que contém as variáveis de entrada e a de saída. Especificando, serão utilizados modelos de classificação, onde, ao invés de se prever valores numéricos, serão previstas características. Resumidamente, modelos que calculam a probabilidade de uma observação pertencer a uma dada classe, classificando-a de acordo com o valor obtido.

Como se está realizando as previsões com os mesmos dados utilizados no treinamento do modelo, é mais indicado que se apliquem a novas observações fora da base de treinamento para então se ter uma melhor noção do desempenho da estimação, fazendo com que a tarefa se aproxime da realidade, já que o objetivo é, após estimar o modelo, colocá-lo diante de novos dados e realizar previsões.

Por isso, é reservada uma amostra teste, tendo-se então uma melhor noção do rendimento do modelo quando exposto à novos dados. A partir do resultado, pode-se então verificar por exemplo se modelo sofre de *overfitting*, que ocorre quando o modelo aprende demais com a amostra de treinamento, tendo bons resultados com dados semelhantes à aquele, porém tendo dificuldades e consequentemente desempenho inferior com dados mais gerais.

A partir desse ponto, pode-se repetir o processo realizando novos pré-processamentos e comparando os resultados dos modelos, ou aplicar os dados a diferentes tipos de modelos de aprendizado de máquina. O processo termina quando se chegam a resultados que o autor considerar razoáveis para o problema que se têm em mãos.

Sobre os modelos utilizados no trabalho serão os seguintes: Regressão Logística, SVM, Redes Neurais e XGBoost. Vale ressaltar que, para a obtenção das previsões, todos realizam através de suas respectivas metodologias o cálculo da probabilidade de ocorrência para cada uma das classes. A que obtiver o maior valor, será escolhida como previsão.

Um exemplo está na Tabela 3, com um exemplo fictício de previsão dos resultados de 5 partidas, onde há a representação das probabilidades de ocorrência de cada uma das classes da variável alvo, que são os resultados possíveis de uma partida de futebol: vitória do mandante, do visitante e empate. E então, dada a maior probabilidade de ocorrência em cada observação, a maior indicará o resultado da previsão. Como exemplo a primeira observação, com o confronto entre Arsenal e West Ham, a probabilidade de vitória do mandate segundo o modelo é a maior (em verde), e portanto será atribuída na previsão a previsão da classe Mandante.

Tabela 3: Probabilidades de ocorrência das classes

ID	Time Mandante	Time Visitante	Prob. Mandante	Prob. Visitante	Prob. Empate	Previsão
1	Arsenal	West Ham	0.6	0.1	0.3	Mandante
2	Aston Villa	Chelsea	0.33	0.37	0.3	Visitante
3	Tottenham	Man City	0.3	0.55	0.15	Visitante
4	Man. United	Wolves	0.7	0.12	0.18	Mandante
5	Everton	Liverpool	0.2	0.27	0.53	Empate

Nas subseções a seguir serão apresentados os modelos utilizados, apresentando uma explicação simplificada para cada um deles. Foi adotada essa abordagem pois o foco do trabalho é a obtenção de resultados, não sendo então necessário um maior detalhamento dos modelos.

2.5.1 Regressão Logística

A Regressão Logística pertencente ao conjunto dos Modelos Lineares Generalizados. É voltado exclusivamente para problemas de classificação, ou seja, trabalha com a variável alvo sendo do tipo categórica, prevendo probabilidades de uma observação pertencer a uma dada classe.

De acordo com (J et al., 2013), o cálculo dessas probabilidades é realizado através da função logística:

$$P = \frac{1}{1 + e^{-y}} \tag{2.2}$$

Sendo:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n.$$

Neste caso, P se refere à probabilidade da observação pertencer a uma dada classe da variável alvo e $x_i's$ às variáveis independentes. A ideia é, a partir de valores obtidos através das variáveis independentes, estimar os melhores parâmetros β_i', esperando então

gerar uma equação capaz de realizar previsões para um dado problema. Essa tarefa de estimar os parâmetros é realizada por uma função de verossimilhança, que testa diferentes valores para os β_i' afim de encontrar valores que maximizem os resultados.

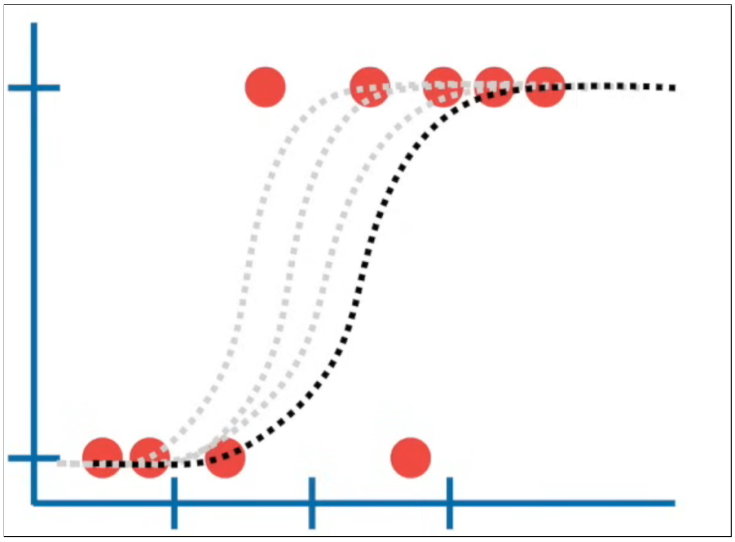

Figura 3: Melhor curva através da Função de Verossimilhança (Fonte: youtube.com/@statquest)

Como mostra a Figura 3, são geradas curvas a partir da estimação dos parâmetros, e o objetivo é encontrar a que melhor se ajuste aos dados do problema. Estimados os melhores parâmetros, pode-se então calcular as probabilidades de ocorrência para cada uma das classes da variável preditora, e a que obtiver a maior será a escolhida.

2.5.2 Redes Neurais Artificiais

Segundo Rodrigo Nogueira[2], as Redes Neurais Artificiais tem como base o sistema nervoso humano. No homem, o sistema nervoso é formado por neurônios, cuja função é transmitir impulsos (informações) ao cérebro através de sua capacidade de receber e propagar estímulos externos ou internos ao organismo. Se um neurônio humano é composto por dendritos, que são responsáveis pela recepção de estímulos, pelo corpo celular, onde é encontrado o núcleo e processada toda informação recebida, e o axônio, encarregado de conduzir os sinais para outros neurônios; nas Redes Neurais Artificiais também há três componentes: uma camada de entrada, uma camada oculta e uma camada de saída, como pode-se observar na Figura 4

De acordo com (HASTIE; TIBSHIRANI; FRIEDMAN, 2009), a camada de entrada tem o papel de receber as informações sobre as variáveis preditivas. Portanto não possui

[2]https://www.pet.ifc-camboriu.edu.br/wp-content/uploads/GDSE-Redes-Neurais.pdf

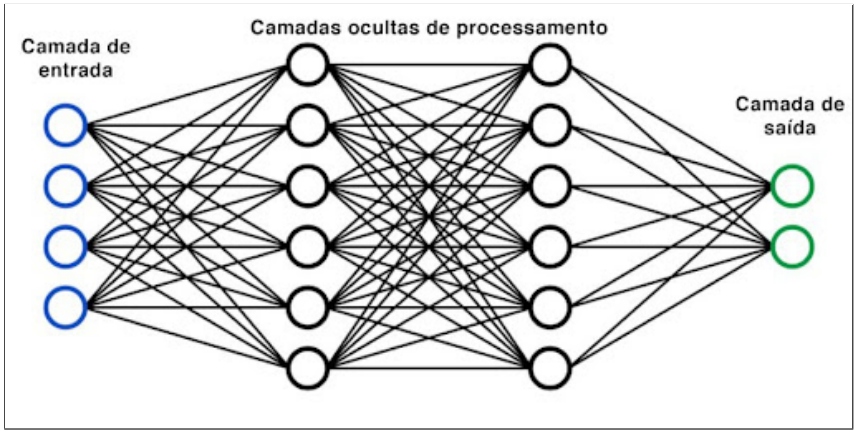

Figura 4: Composição de uma Rede Neural Artificial (Fonte: opencadd.com.br/o-que-sao-redes-neurais/)

nenhum neurônio, e o número de dimensões corresponde ao número de variáveis de entrada. Já a camada oculta pode possuir uma ou mais camadas de neurônios, onde cada uma deve ter pelo menos um neurônio. Os neurônios por sua vez são compostos pelos pesos sinápticos e pelo limiar de ativação, que serão estimados no processo, um combinador linear que é responsável por agregar as informações obtidas dos neurônios anteriores ou da camada de entrada, e uma função de ativação, responsável por realizar os cálculos para por exemplo, em um modelo de classificação, obter as probabilidades e assim conseguir diferenciar uma classe de outra. E a camada de saída, que tem geralmente um neurônio (porém pode ter mais caso se esteja trabalhando com problemas de classificação com mais de duas classes), que tem o papel de agregar os dados gerados pela camada oculta e fornecer uma resposta adequada. Um exemplo pode ser observado na Figura 5, onde há três variáveis de entrada, uma camada oculta com duas camadas sendo uma com quatro neurônios e outra com três, e na camada de saída, por se tratar de um problema de classificação multiclasse com três classes, as probabilidades da observação pertencer a cada uma das classes.

Conforme exemplifica Jéssica Kubrusly[3], para a estimação dos parâmetros é utilizado o método da Retropropagação. O processo de estimação inicia ao se definir, de forma aleatória, valores para os pesos sinápticos e para os limiares de ativação. Após isso, para cada uma das observações, são encontrados valores de entrada e de saída para cada função de ativação, além do valor para estimativa da variável de saída; essa é a etapa da propagação, começando do lado esquerdo com as variáveis de entrada e indo até o direito com a saída. Os valores dos parâmetros então são atualizados e então se faz o processo de volta, realizando os cálculos agora da direita para esquerda, que é a etapa da

[3]https://rpubs.com/jessicakubrusly/redes-neurais-7

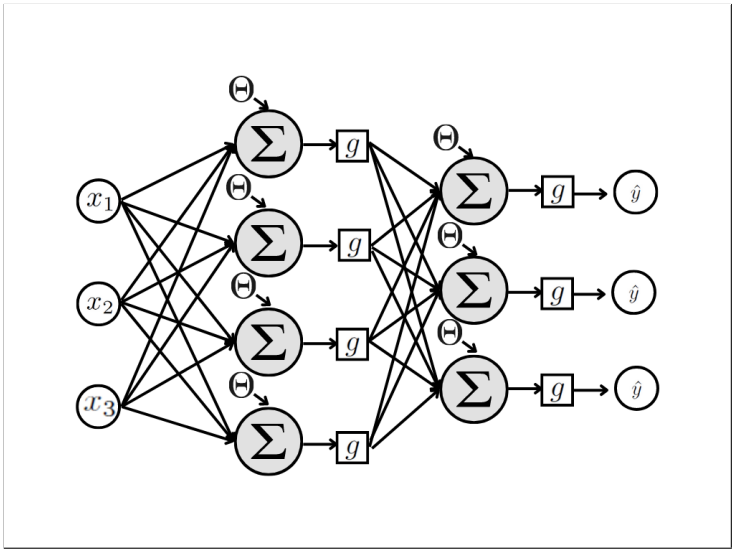

Figura 5: Rede Neural Artificial para Classificação Multiclasse (Fonte: rpubs.com/jessicakubrusly/redes-neurais-6)

retropropagação. E então esses dois processos são realizados até que se atinja um número de passos de iterações preestabelecidos ou que se obtenha uma estabilidade nos resultados, ou seja, resultados com pouca variação entre sí.

2.5.3 *Support Vector Machine* (SVM)

O SVM é um algoritmo de aprendizado de máquina que pode ser utilizado tanto para regressão quanto para classificação. Segundo (HASTIE; TIBSHIRANI; FRIEDMAN, 2009), ele funciona ao dividir as observações em grupos distintos através de um plano, chamado de hiperplano.

Supondo que se esteja trabalhando com o modelo de classificação binário (duas classes) como mostra a Figura 6,a ideia é de, projetados os pontos que representam as observações da base de dados em questão, encontrar a reta que distancie os dados em duas classes da melhor maneira possível. Isso é feito ao pegar os pontos mais afastados de cada grupo e que estejam mais próximos do hiperplano, que são chamados de vetores de suporte, tentando assim encontrar o melhor ajuste de forma que a distância entre cada um dos dois pontos seja a maior possível para a reta (hiperplano), e consequentemente configurando uma maior separação entre os dados. Essa distância entre os pontos e o hiperplano é chamada de margem.

Porém há casos em que, por conta dos dados, como mostra a Figura 7, não é possível projetar uma reta que separe os dados em grupos distintos. Neste situação, é utilizado

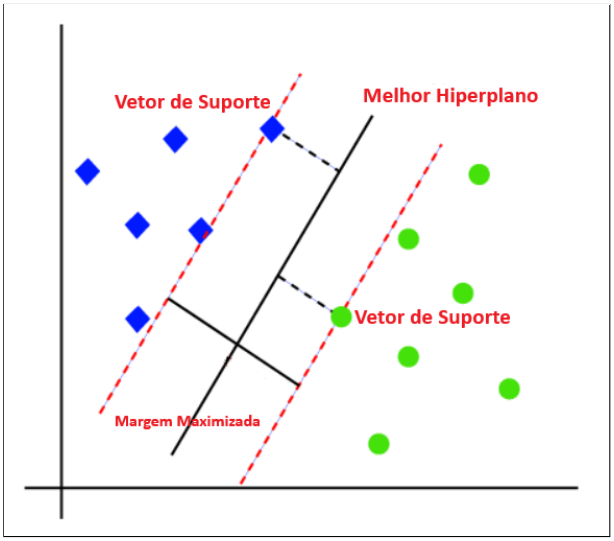

Figura 6: Gráfico de um modelo SVM para classificação binária (Fonte: analyticsvidhya.com/)

o SVM não linear, que faz uma transformação não-linear do espaço, e após essa etapa pode-se fazer a separação das classes com o SVM linear através de uma reta. Portanto é feita uma separação não linear no espaço inicial, que após a transformação passa a ser linear.

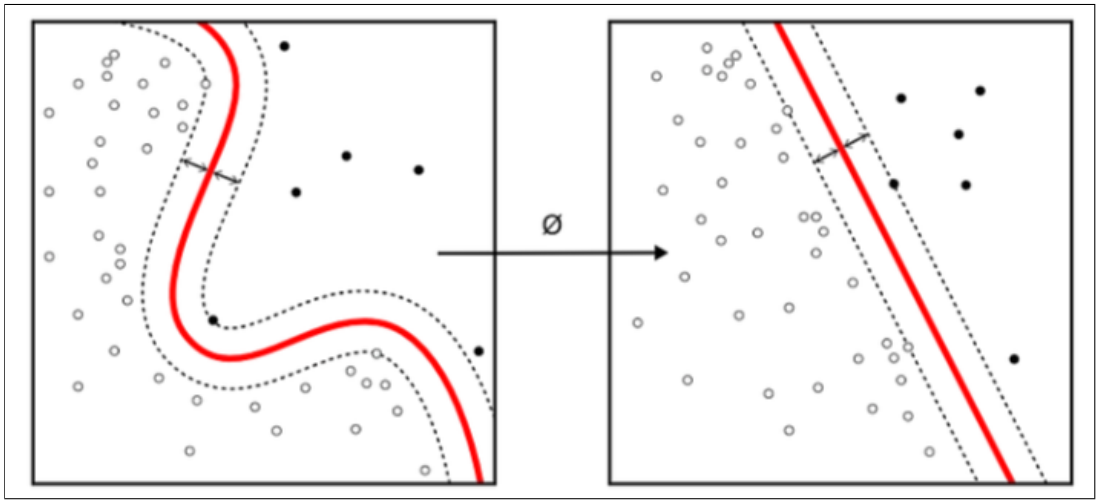

Figura 7: Transformação não-linear onde o SVM linear não se aplica diretamente (Fonte: commons.wikimedia.org/wiki/File:Kernel-Machine.png)

Para os casos de classificação multiclasse, um dos métodos utilizados pelo algoritmo é fazer comparação entre as classes duas a duas, ou seja, várias classificações binárias. Ao final do processo, o modelo gera a probabilidade da observação pertencer a cada uma das classes, e a que obtiver o maior valor é a escolhida.

2.5.4 XGBoost

Uma maneira encontrada para aumentar o desempenho de modelos de previsão é, de acordo com (HASTIE; TIBSHIRANI; FRIEDMAN, 2009), combinar vários algoritmos, que são chamados de fracos, em um único, formando então um modelo forte. A ideia é fazer com que os algoritmos sejam colocados para trabalhar em sequência, onde o posterior tentará melhorar os erros dos anteriores.

Esse método de combinar classificadores recebe o nome de *Ensemble*, e há duas categorias dentro dele: *Bagging* e *Boosting*. Dado que os modelos podem ser diferenciados através da variância e do viés, *Bagging* e *Boosting* tratando respectivamente desses dois aspectos.

Os modelos que utilizam o *Bagging* buscam reduzir a variância, através da correção de modelos anteriores em que ocorreu *overfitting*, ou seja, o modelo aprendeu demais. Já os que utilizam o *Boosting* visam diminuir o viés, ou seja, tentam corrigir os erros de modelos anteriores em situações em que o modelo aprendeu pouco e os resultados estão distantes dos observados, corrigindo então o *underfitting*.

Um exemplo de modelo *Boosting* é o XGBoost. O modelo combina classificadores mais fracos, e ao final do processo obter um classificador mais forte. Essa tarefa, de acordo com (MARINHO, 2021), é realizada ao se atribuir pesos maiores para as previsões erradas a cada modelo criado, fazendo que os modelos posteriores foquem na correção das falhas.

No caso do XGBoost, assim como outros modelos do tipo *Boosting*, utiliza como base o modelo de árvore de decisão, e se destaca em relação aos demais por possuir características que favorecem o aumento do desempenho, tanto em relação ao tempo de treinamento, com a capacidade de realizar processamento paralelo, como em relação aos resultados, por conseguir lidar com valores ausentes tanto na base de treino quanto na de teste e por conseguir realizar um autoajuste dos parâmetros do modelo.

2.6 Matriz de Confusão e Medidas de Desempenho

Uma maneira de verificar o desempenho de um modelo de classificação é utilizar uma matriz de confusão. Nela, de acordo com (J et al., 2013), são relacionados os dados observados com as previsões, de maneira a mostrar as taxas de acerto e erro que o modelo obteve tanto de maneira geral como para cada uma das classes.

Tabela 4: Matriz de Confusão Binária

		Previsão	
		Sim	Não
Real	Sim	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Não	Falso Positivo (FP)	Verdadeiro Negativo (VN)

A Tabela 4 mostra um exemplo de classificação binária, com a classe "Sim" sendo a considerada positiva e a classe "Não" a negativa, tem-se que:

- Verdadeiro Positivo (VP): valores para classe positiva que foram previstos corretamente.
- Verdadeiro Negativo (VN): valores para a classe negativa previstos corretamente.
- Falso Negativo (FN): valores que são da classe positiva mas que foram previstos incorretamente como sendo da classe negativa.
- Falso Positivo (FP): valores que são da classe negativa mas que foram previstos incorretamente como sendo da classe positiva.

E uma maneira de ajudar a avaliar os resultados através da matriz de confusão é utilizar as medidas de desempenho. Em geral, as mais usadas são:

- **Acurácia**: quantidade de acertos do modelo em relação ao total

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN}.$$

- **Sensibilidade**: quantidade de acertos em relação à classe positiva

$$Sensibilidade = \frac{VP}{VP + FN}.$$

- **Especificidade**: quantidade de acertos em relação à classe negativa

$$Especificidade = \frac{VN}{VN + FP}.$$

No caso do problema envolver classificação multiclasse, a abordagem é semelhante. A Tabela 5, por exemplo, apresenta uma matriz de confusão obtida após a aplicação de um modelo de aprendizagem de máquina para um problema de classificação com a variável resposta sendo o resultado de jogos de futebol para um determinado campeonato, contendo três classes: vitória do Mandante, vitória do Visitante ou Empate.

Tabela 5: Matriz de Confusão Multiclasse

	Real		
Previsão	Mandante	Visitante	Empate
Mandante	7	8	9
Visitante	1	2	3
Empate	3	2	1

As colunas indicam os valores reais, observados, e as linhas representam as previsões. As diagonais apresentam os valores previstos corretamente, enquanto as demais os erros. Por exemplo, para a classe Visitante, na coluna 2, o modelo fez a previsão correta em dois casos, porém em 8 ele errou ao indicar que se tratava de vitória do Mandante e em 2 de Empate.

No caso das medidas de desempenho, o objetivo é de realizar comparações binárias, ou seja, analisar uma variável em relação às demais.

Tabela 6: Matriz de Confusão Binária para Medidas de Desempenho

		Previsão	
		Positiva	Negativa
Real	Positiva	7	4
	Negativa	17	8

Utilizando como exemplo a classe Mandante, a Tabela 6 faz a comparação dessa classe com as outras, sendo Mandante a classe positiva e as outras a negativa. Houveram 7 previsões corretas nos casos em que a observação se tratava da classe Mandante (Verdadeiros Positivos) e 8 (Verdadeiros Negativos) nos casos em que a classe da observação não era a Mandante, independentemente de acerto das demais classes; em contrapartida, em 4 oportunidades houve a previsão para outra classe quando a observação era Mandante (Falsos Negativos) e 17 vezes foi previsto Mandante quando na realidade a observação se tratava de Empate ou Visitante (Falsos Positivos). E a partir disso, obtendo-se os valores para Verdadeiros Positivos e Falsos Negativos por exemplo, pode-se calcular a Sensibilidade para a classe Mandante, que é a classe 1,

$$Sensibilidade_{Mandante} = \frac{VP_{Mandante}}{VP_{Mandante} + FN_{Mandante}}.$$

3 Análise dos Resultados

Antes de entrar nos resultados, será feita uma breve explicação das decisões tomadas e ações realizadas até se chegar nas bases que foram utilizadas para treinar e testar os modelos. Como já mencionado, visto que os dados correspondem a informações de partidas já finalizadas, existia o problema de prever partidas futuras pois não teriam as informações das mesmas. Como solução, cada variável agora representa o valor daquela característica para as últimas partidas. Um exemplo está na Tabela 7, onde está representada uma sequência de jogos do Chelsea, sendo 5 partidas já ocorridas, e a sexta (em vermelho), contra o United, em que as equipes ainda vão se enfrentar. Pode-se observar portanto que não existe informação sobre essa sexta partida. Então, para contornar esse problema, foram calculadas as médias das 3 partidas anteriores (em azul) para que assim seja possível realizar a previsão para a variável alvo Resultado. Ou seja, agora ao invés de uma observação conter exatamente os valores das estatísticas da partida que representa, terá a média dos últimos 3 jogos.

Tabela 7: Exemplo de aplicação da metodologia

ID	Time_Mand	Time_Visit	Chutes_Mand	Chutes_Visit	Posse_Mand	Resultado
1	Chelsea	Aston Villa	19	7	65	Mandante
2	Arsenal	Chelsea	16	13	54	Empate
3	Chelsea	Leicester	20	11	60	Mandante
4	Tottenham	Chelsea	12	12	49	Mandante
5	City	Chelsea	18	11	62	Visitante
6	United	Chelsea				

Porém a média das 3 partidas anteriores é apenas uma suposição, sendo necessário então identificar qual o melhor conjunto de partidas anteriores que resultará em melhores previsões. Como apenas utilizar a partida anterior ou médias de 7 partidas ou mais não estavam gerando resultados satisfatórios, optou-se então por construir bases com médias variando de 2 até 6 partidas anteriores.

Outro ponto a ser considerado é a escolha dos jogos que vão compor o conjunto das partidas anteriores para o cálculo da média. Exemplicando com a média de 3 jogos ante-

riores, as opções são duas: realizar uma separação, escolhendo apenas os últimos 3 jogos do time como mandante caso ele seja mandante no próximo, ou, caso ele seja visitante, os últimos 3 como visitante; ou apenas escolher as últimas 3 partidas, independemente do mando de jogo.

Considere o exemplo da Tabela 7, caso a escolha seja realizar a separação entre mandante e visitante, para a observação 6 será feita a média das 3 partidas anteriores do United como mandante e das 3 últimas partidas do Chelsea como visitante, sendo para o Chelsea então a média das 3 últimas partidas presentes nas observações 2, 4 e 5, onde o clube foi visitante, das variáveis referentes ao time visitante. Se não for realizada a separação, para o Chelsea por exemplo, vão ser utilizadas as informações das variáveis presentes nas observações 3, 4 e 5, sem especificar o mando de campo.

No presente trabalho, após a realização de testes e a verificação de que não houve diferença significativa entre escolher os últimos jogos do time independentemente de mando de campo ou não, optou-se pelo método de escolher os últimos jogos de acordo com o mando de campo.

Ainda foi identificado no trabalho que os modelos tiveram dificuldades para realizar a previsão de empates, onde os acertos para classe em questão ficaram muito próximos de zero. Portanto, uma nova base com a variável de interesse com apenas duas classes foi criada, transformando Visitante e Empate em apenas uma: a Não Mandante. O objetivo é verificar se haveria uma melhora nas previsões utilizando duas classes.

Já para a divisão dos dados em treino e teste, foi utilizado como teste o segundo turno do último campeonato, ficando então a base teste com 190 observações. A base de treino por outro lado variou em relação ao tamanho de acordo com o número de campeonatos escolhidos e ainda de acordo com a média das últimas partidas disputadas. Isso porque as primeiras rodadas dos campeonatos tinham que ser excluídas por conta da ausência de partidas anteriores para calcular a média para essas observações.

Em relação à quantidade de campeonatos utilizados, é necessário fazer a observação de que o trabalho ficou limitado a campeonatos depois de 2016, pois os dados de campeonatos anteriores apresentaram muitos problemas, indo até 2019, por conta da pandemia (Covid-19, 2020) e das adversidades provocadas por ela nos anos seguintes. Optou-se então por utilizar os campeonatos realizados entre 2016 e 2018, e então surgiu outra questão a ser respondida: se com apenas o primeiro turno de um campeonato (2017/2018) seria possível prever o segundo turno do mesmo, ou se seria necessário aumentar a base de dados de treino com outra temporada anterior (2016/2017) para obter melhores resultados.

Então no total serão 3 casos diferentes de comparação, listadas a seguir:

- **Caso 1**: Acurácia com dois campeonatos contra um,
- **Caso 2**: Acurácia com três classes contra duas,
- **Caso 3**: Acurácia entre as médias das partidas anteriores.

3.1 Análise Descritiva

A base de dados, com o caso principal, utilizada no trabalho, contêm os dois campeonatos que ocorreram entre 2016 e 2018 e 3 classes, possuindo 760 observações e 97 variáveis. Entre essas variáveis, a variável de interesse Resultado é mostrada na Figura 8 com a quantidade de observações para cada uma de suas classes.

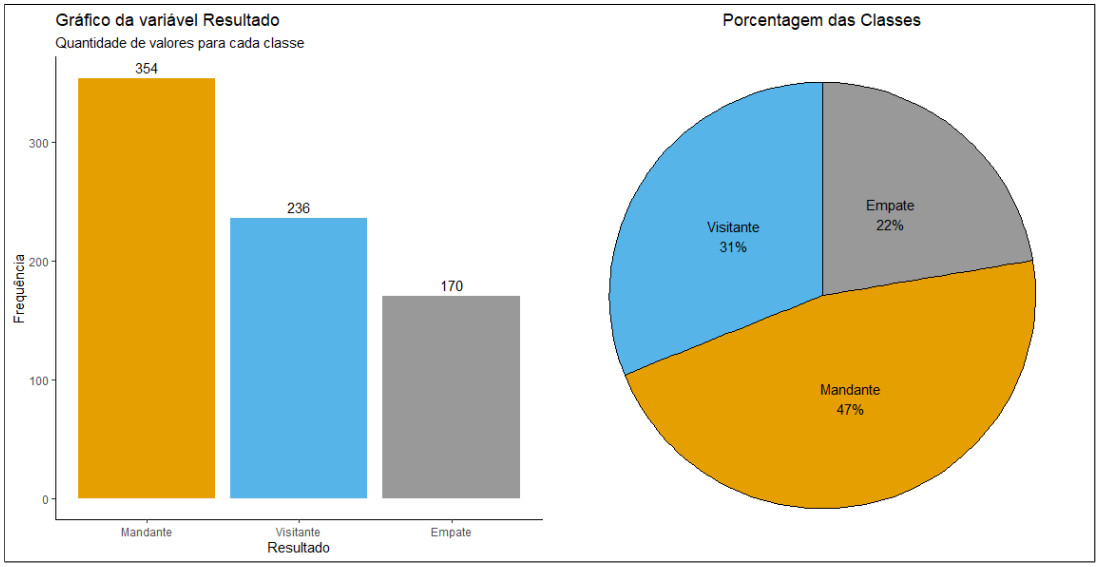

Figura 8: Gráficos com as frequências absoluta e relativa das 3 classes da variável alvo Resultado com a base contendo os dois campeonatos)

Pode-se verificar que há uma quantidade maior de vitórias de mandantes, representando quase metade das observações, seguida pela classe Visitante e depois, em menor quantidade, a classe Empate.

Na análise das variáveis explicativas, pretende-se verificar indícios de relação com a Resultado. Ou seja, deve-se observar se determinados valores em uma dada variável explicativa se adequam mais a uma das classes da variável alvo, facilitando então sua identificação.

Em geral, utilizando ferramentas gráficas, não foi observada a existência de muitas variáveis que possuam forte relação com a variável Resultado. Destacam-se, a seguir, as que mais apresentaram indícios de relação e, em contrapartida, as que menos parecem influenciar a variável alvo. Os gráficos da Figura 9 apresentam exemplos de variáveis com aparente relação com a variável Resultado.

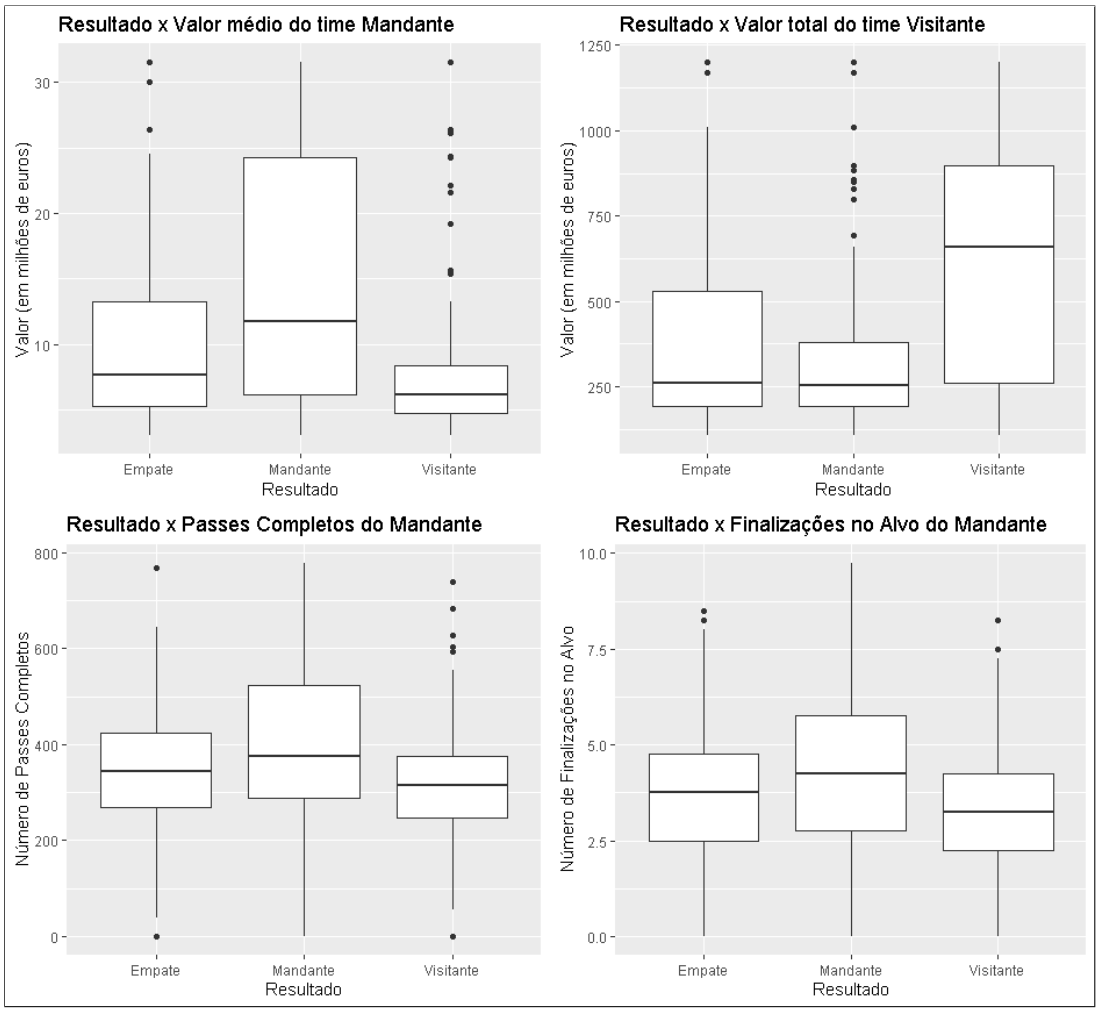

Figura 9: Boxplots das variáveis com aparente relação com a variável Resultado.

Pode-se verificar como destaques variáveis que representam os valores em dinheiro dos times em relação aos jogadores que compõem seus respectivos elencos, como a que indica o valor médio dos jogadores do time Mandante e a que mostra a soma total dos valores dos jogadores do time Visitante. Existem indícios de uma relação positiva entre Resultado e clubes que possuem jogadores mais valiosos, seja em relação a vitória de mandantes ou visitantes.

Outras variáveis em destaque são as que se referem a passes completos e finalizações dos mandantes, porém mostrando uma relação menor com a variável alvo em comparação com as mencionadas anteriormente. Exemplicando, é de se esperar que clubes que joguem

em casa ou times mais valiosos finalizem mais e tenham mais dominio do jogo, tendo então mais passes completos. Porém acontecimentos dentro de uma partida podem influenciar nessas estatísticas. Um exemplo é um clube mais valioso que domina o seu adversário, tendo mais finalizações e passes, mas que ao conseguir fazer um gol, muda sua postura em campo e passa a se preocupar mais com a defesa na tentativa de assegurar o resultado. A partir desse ponto, o adversário, que está perdendo, passa a se lançar mais ao ataque, conseguindo mais finalizações e mais passes do que se esperava, terminando a partida com estatísticas superiores, mas sem conseguir igualar o placar. Ao final do jogo o time mais valioso vence a partida, porém tendo estatísticas inferiores do que o adversário mais limitado. Esta é uma situação das várias que podem ocorrer indicando que não necessariamente o melhor time e que conseguiu a vitória terá estatísticas superiores.

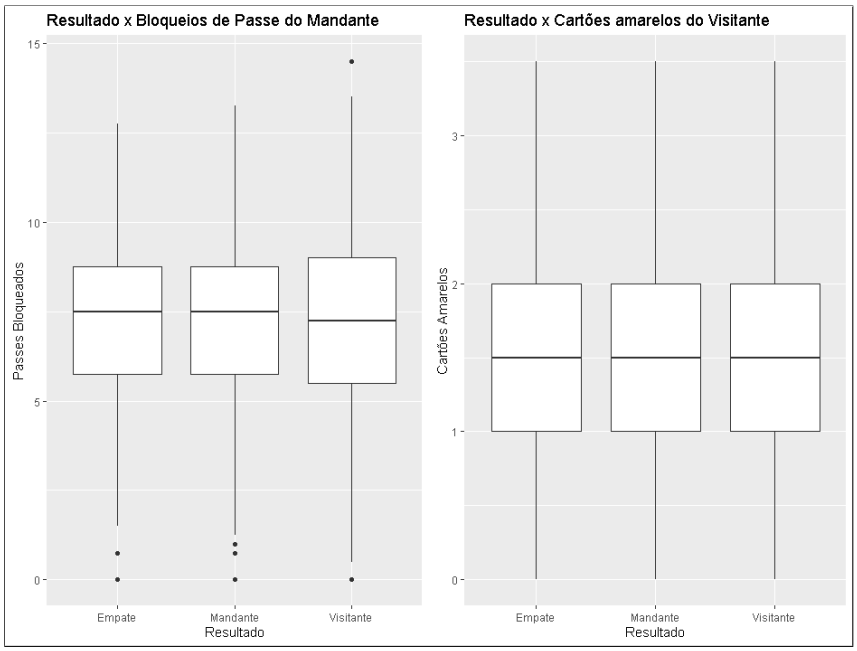

Figura 10: Boxplots das variáveis sem relação aparente com a variável Resultado.

Sobre as variáveis que aparentemente não possuem relação com a variável alvo, a Figura 10 apresenta dois exemplos. Não há indícios de que as variáveis que indicam os bloqueios de passes dos mandantes e os cartões amarelos dos visitantes tenham impacto suficiente à ponto de diferenciar as classes da variável alvo. Porém, elas não devem ser descartadas pois, em conjunto com outras variáveis, podem ser importantes para os modelos. Por isso são necessários outros métodos para completar a análise de importância das variáveis.

Portanto, variáveis que não apresentaram indícios de relação com a variável alvo devem ser colocadas em observação para verificar nos próximos testes se eles devem ou

não continuar na base que participará do treinamento dos modelos.

Sobre os demais testes feitos, como a transformação das classes Visitante e Empate da variável alvo Resultado na classe Não Mandante, diminuindo então de três para duas classes, a Figura 11 apresenta a nova proporção entre as classes, indicando maior equilíbrio na quantidade.

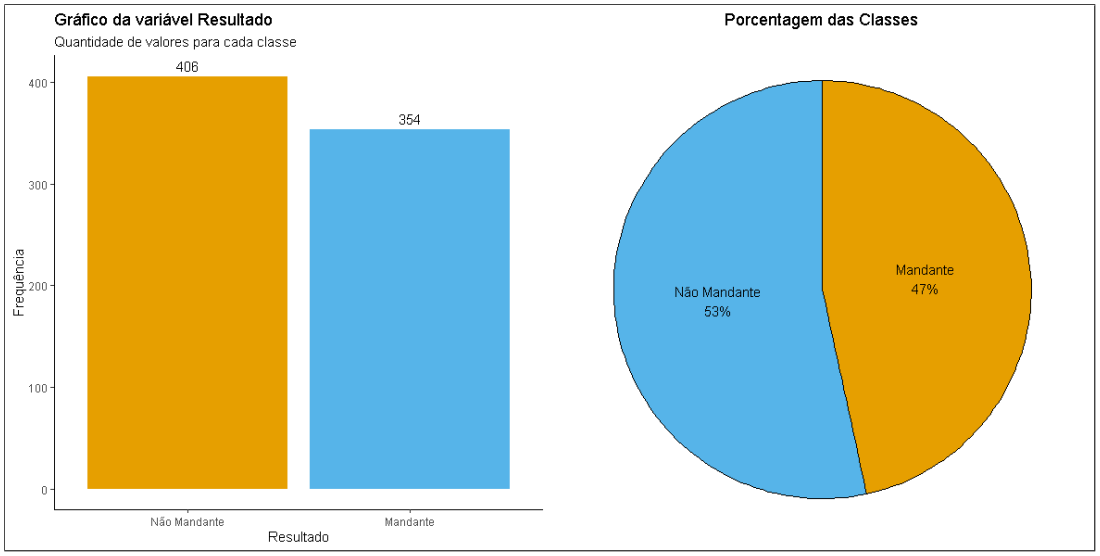

Figura 11: Gráficos com a frequência absoluta e a porcentagem das duas classes da variável alvo Resultado com a base contendo os dois campeonatos)

Além disso, há também o caso em que se busca verificar se só com os dados de um campeonato é possível prever os resultados do segundo turno, ou se é necessário adicionar um campeonato para obter melhores previsões. A base para um campeonato (2017/2018) contém 380 observações, e a Figura 12 mostra a configuração dos dados, onde pode-se verificar que não há grande diferença na distribuição da variável Resultado em relação à base com dois campeonatos (760 observações), com a classe Mandante com mais observações.

Para o caso com um campeonato e duas classes, apresentada na Figura 13, também não há diferença em relação ao mesmo caso com dois campeonatos, apresentando um equilíbrio na quantidade de observações das duas classes.

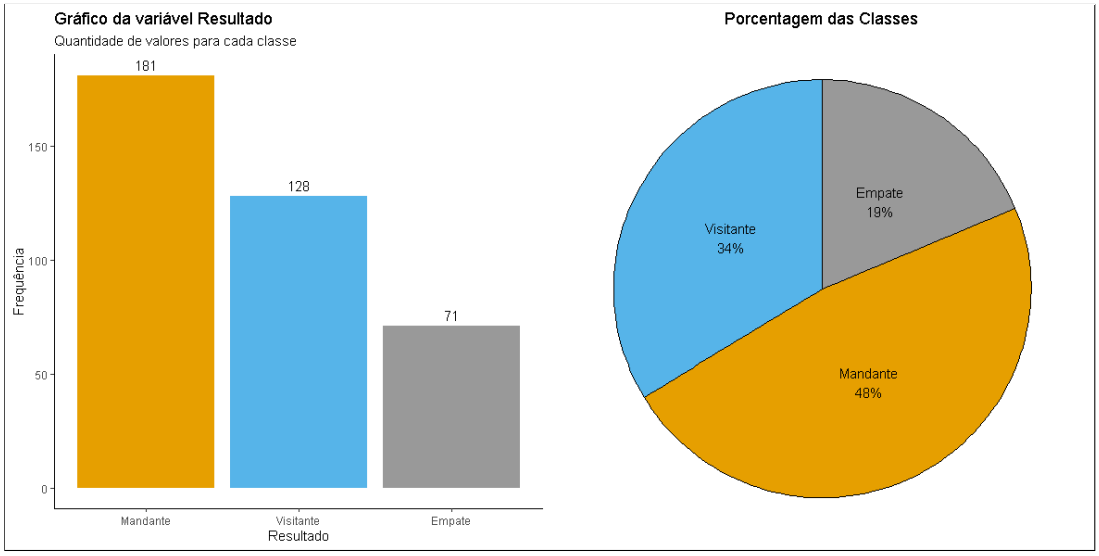

Figura 12: Gráficos com a frequência absoluta e a porcentagem das três classes da variável alvo Resultado com a base contendo um campeonato)

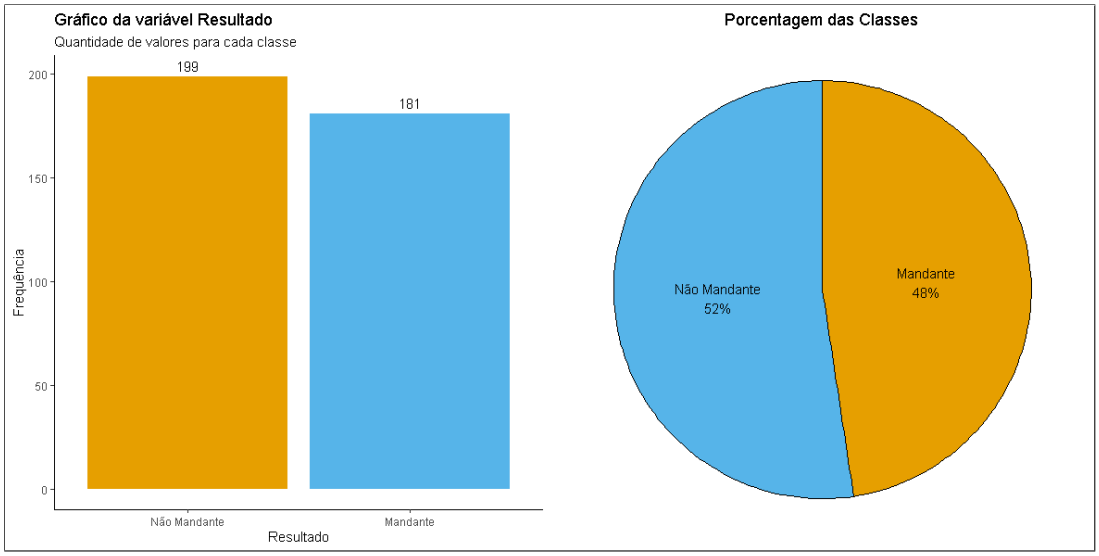

Figura 13: Gráficos com as frequências absoluta e relativa das duas classes da variável alvo Resultado com a base contendo um campeonato)

3.2 Seleção de Variáveis

Para selecionar as melhores variáveis para o treinamento dos modelos, a estratégia que melhor obteve resultados foi a aplicação do método RFE, que utiliza um modelo de Árvores de Decisão e sua capacidade de gerar a importância das variáveis. O resultado de maior acurácia, como mostra a Figura 14, considera 33 variáveis.

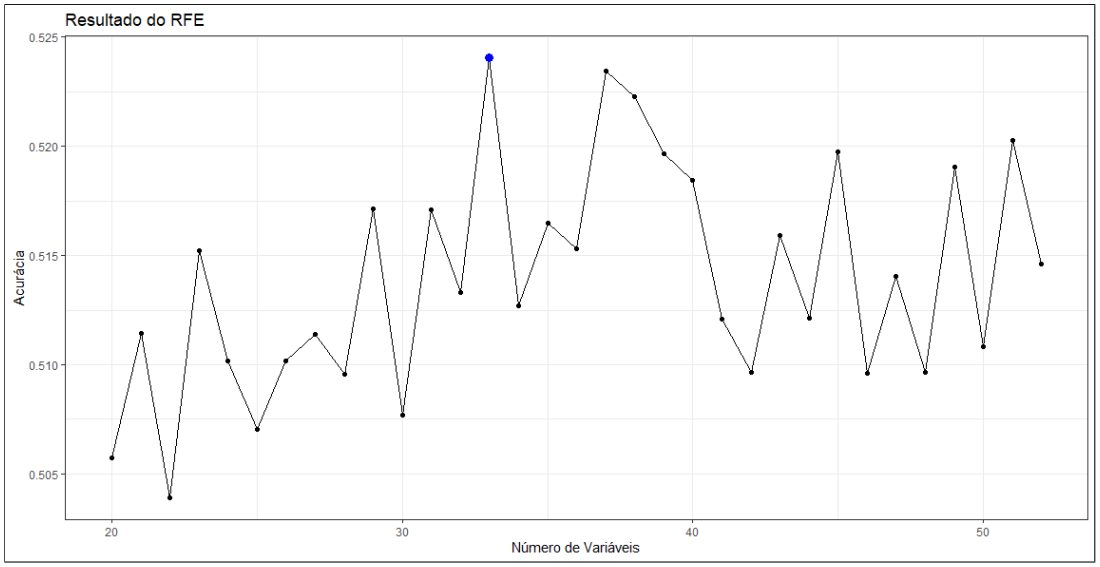

Figura 14: Gráfico que mostra a acurácia dos diferentes modelos gerados pela função RFE assim como número de variáveis presentes nos mesmos.

Também foram realizados outros testes, porém as alterações realizadas não surtiram efeito positivo. Um exemplo foi o teste de multicolinearidade, que apontou que haviam variáveis explicativas altamente correlacionadas. Porém, ao se testar conjuntos de dados com diferentes combinações sem variáveis com grande correlação entre sí não geraram modelos com bom rendimento. Isso quer dizer que, apesar de altamente correlacionadas, as variáveis individualmente traziam informações para o modelo que outras não conseguiam captar, e portanto a sua simples exclusão prejudicava a taxa de acerto dos modelos.

Outros métodos testados foram a Análise de Componentes Principais (PCA), que faz uma redução de dimensionalidade de conjuntos de dados, e o rebalanceamento de classes, que equilibra o número de observações para cada uma das classes da variável alvo na base. Porém, como já mencionado, não houveram melhorias nos resultados.

Portanto, para o treinamento dos modelos, as 33 variáveis escolhidas segundo o método RFE são as presentes na Tabela 8, contendo não só os nomes originais da variáveis bem como suas respectivas descrições.

Dado que cada característica é apresentada como uma variável para o time mandante

Tabela 8: Variáveis selecionadas e suas descrições

Variável	Descrição
Home valor merc	Valor de mercado total do Mandante
Home media merc	Valor médio de mercado do Mandante
Home Carries	Bolas conduzidas pelo time Mandante
Home SCA	Jogadas criadas para finalizações pelo Mandante
Home Att 3rd Touches	Toques na bola no terço final de campo pelo Mandante
Home Mid 3rd Touches	Toques na bola no meio do campo pelo Mandante
Home Cmp percent Passes	Porcentagem de passes certos do Mandante
Home SoT	Finalizações no alvo pelo Mandante
Home Cmp Passes	Passes completos pelo Mandante
Home PrgC Carries	Bolas conduzidas para frente pelo Mandante
Home Touches	Toques na bola do Mandante
Home PrgP Passes	Passes para frente pelo Mandante
Home Att Take Ons	Jogadores superados com dribles no ataque pelo Mandante
Home Score	Gols do Mandante
Home Clr	Bolas tiradas do campo de defesa pelo Mandante
Home Att Passes	Passes no ataque pelo Mandante
Home Sh Blocks	Chutes bloqueados pelo Mandante
Away media merc	Valor médio de mercado do Visitante
Away valor merc	Valor de mercado total do Visitante
Away Clr	Bolas tiradas do campo de defesa pelo Visitante
Away Mid 3rd Touches	Toques na bola no meio do campo pelo Visitante
Away Tkl percent Challenges	Porcentagem de divididas ganhas pelo Visitante
Away Cmp Passes	Passes completos pelo Visitante
Away Touches	Toques na bola do Visitante
Away SCA	Jogadas criadas para finalizações pelo Visitante
Away Att Passes,	Passes no campo de ataque pelo Visitante
Away Carries	Bolas conduzidas pelo Visitante
Away Final Third	Chegadas ao último terço de campo pelo Visitante
Away PrgP Passes	Passes para frente pelo Visitante
Away Sh Blocks	Finalizações bloqueadas pelo Visitante
Away Lost Challenges	Disputas Perdidas pelo Visitante
Away Def Pen Touches	Toques dentro da sua pequena área pelo Visitante
Away GCA SCA	Jogadas de gol criadas pelo Visitante

(prefixo *Home*) e uma para o visitante (prefixo *Away*), verifica-se que tanto características de mandantes como visitantes são importantes para os modelos, dada as quantidades de variáveis para cada categoria. Também se percebe que a maioria das variáveis se referem a características de ataque, sendo apenas 5 das 33 sobre defesa. E as que apresentaram melhor relação com a variável alvo Resultado foram as que indicavam os valores dos clubes a partir da soma total do valor de seus jogadores, indicando então que o poderio econômico dos clubes parece ser relevante em resultados de partidas de futebol. A Figura 15 apresenta as variáveis mais importantes segundo o método utilizado, sendo que *Mean*

Decrease Accuracy indica o quanto o modelo perde em média em acurácia quando a variável em questão for retirada.

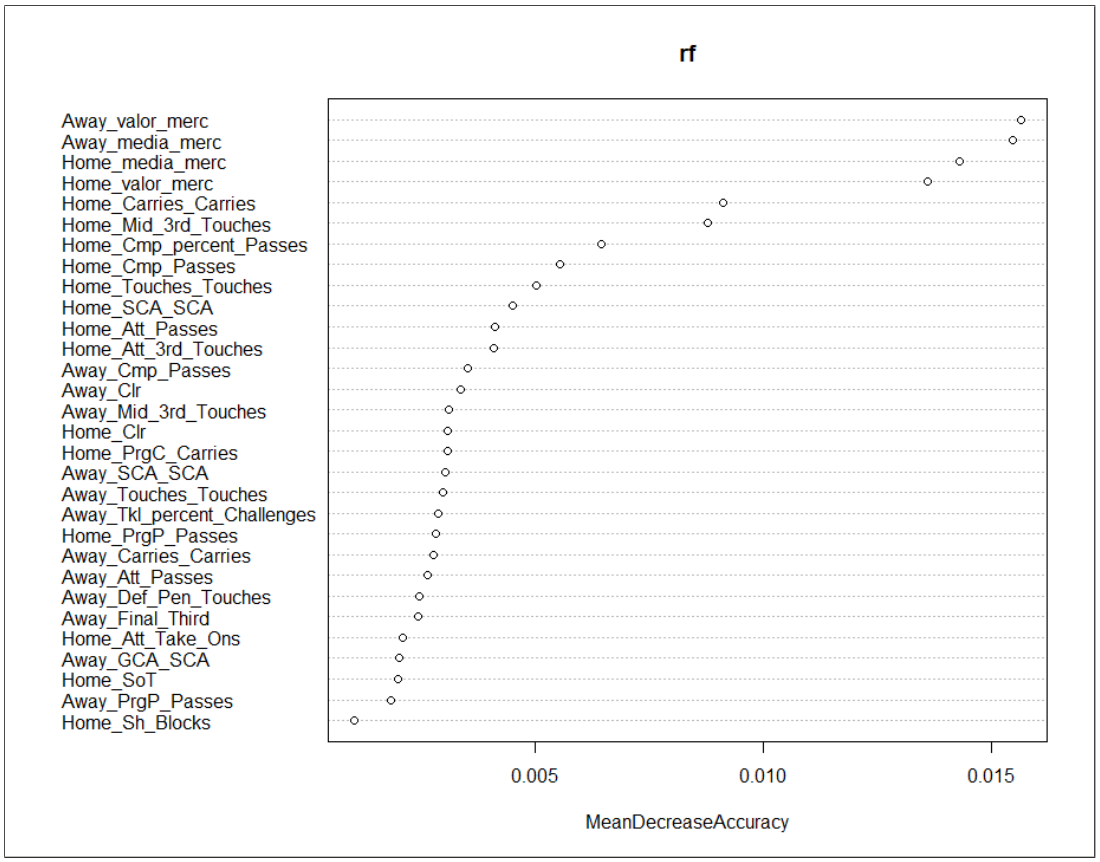

Figura 15: Importância das variáveis

3.3 Resultados

Para a construção dos modelos foi utilizada um caso base, com dados contendo informações de dois campeonatos da *Premier League* (2016/2017 e 2017/2018), além de utilizar os 3 casos possíveis de resultados na variável alvo: vitória do Mandante, do Visitante ou Empate.

Ainda foram feitas alterações nas variáveis por conta da substituição do valor observado por uma média de n partidas anteriores, sendo o valor de n variando entre 2 e 6. Além disso, foram realizados treinamentos dos modelos com as bases realizando 3 comparações: se apenas um campeonato bastaria para um bom desempenho nas previsões para o segundo turno ou se é necessário adicionar mais um campeonato nos dados (Caso 1), se a diminuição de 3 para 2 classes impactaria nos resultados (Caso 2), e qual seria a média de partidas anteriores que gera as melhores previsões (Caso 3).

Portanto a seguir serão apresentados os resultados para os 3 casos citados, relembrando que os resultados tem como referência uma mesma base de testes, que contém os 190 jogos do segundo turno do campeonato inglês de 2017/2018. Apenas há alterações nas bases de treino.

3.3.1 Caso 1 - Acurácia comparando dois campeonatos contra um

A seguir, na Tabela 9, os resultados após os treinamentos dos modelos com as bases com dados das temporadas de 2016/2017 e 2017/2018, variando os intervalos para o cálculo da média de jogos passados e levando em consideração o mando de campo ao selecionar as partidas anteriores:

Tabela 9: Acurácia dos modelos para os campeonatos de 2016/2017 e 2017/2018

-	Regressão L.	SVM	Redes Neurais	XGBoost
Média2	0.5579	0.5789	0.6105	0.5842
Média3	0.5684	0.5789	0.5684	0.5842
Média4	0.5789	0.58	0.6159	0.60
Média5	0.57	0.575	0.575	0.585
Média6	0.5789	0.5789	0.5789	0.5895

Pode-se verificar que em geral, os modelos obtiveram resultados parecidos, sendo o de Regressão Logística o que obteve os piores resultados. O SVM foi o terceiro colocado, com o XGBoost com resultados com menor variação na segunda colocação, e o modelo de Redes Neurais em primeiro. Sobre a média das partidas anteriores, o destaque foi para a que utilizou 4 partidas, obtendo a melhor acurácia em todos os modelos (em azul na tabela), incluindo o melhor que foi o de Redes Neurais, obtendo uma acurácia de 61,59%, destacado em verde.

Verificando agora a sensibilidade e a especificidade apresentados na Tabela 10, que contém os valores apenas para modelos utilizando média 4, já que foram os destaques em todos os modelos, pode-se verificar que todos os modelos tiveram resultados ruins em relação aos empates, sem conseguir acertar nenhuma previsão. Para as vitórias de mandantes, todos tiveram um rendimento acima de 80% na sensibilidade, porém baixa especificidade, indicando então uma dificuldade para se prever valores para a classe negativa, que nesse caso é composta por empates e vitória do visitante. Para vitórias de visitantes, como já dito anteriormente, todos os modelos tiveram baixa taxa de acertos, obtendo então baixa sensibilidade e uma alta especificidade, por conta dos acertos com mandantes.

Por ser mais equilibrado, obtendo maior taxa de acertos para a classe visitantes, o modelo de Redes Neurais foi o destaque. Já o XGBoost, que obteve resultados mais consistentes em relação a acurácia, apresentou alta taxa de acerto para a classe mandante e baixa para a visitante.

Tabela 10: Sensibilidade e Especificidade dos Modelos

Modelos	Medidas	Classes		
		Empate	Mandante	Visitante
Regressão Logística	Sensibilidade	0.0000	0.8316	0.5000
	Especificidade	1.0000	0.3789	0.8359
SVM	Sensibilidade	0.0000	0.8947	0.4032
	Especificidade	1.0000	0.3263	0.8750
Redes Neurais	Sensibilidade	0.0000	0.8421	0.5968
	Especificidade	1.0000	0.4737	0.8203
XGBoost	Sensibilidade	0.0000	0.9263	0.4194
	Especificidade	1.0000	0.3158	0.9141

Na Tabela 11 estão os resultados utilizando uma base apenas com o campeonato de 2017/2018 e 3 classes.

Tabela 11: Acurácia dos modelos com o campeonato de 2017/2018

-	Regressão L.	SVM	Redes Neurais	XGBoost
Média2	0.5368	0.5526	0.5526	0.4947
Média3	0.5053	0.5737	0.5158	0.5421
Média4	0.5632	0.5842	0.5105	0.5526
Média5	0.5368	0.5684	0.5526	0.5474
Média6	0.5368	0.5579	0.5263	0.5474

Pode-se verificar que em geral, os modelos obtiveram resultados parecidos novamente, sendo o destaque para o SVM, com 58,42% de acurácia. Sobre a média das partidas anteriores, o destaque foi para a que utilizou 4 partidas, obtendo a melhor acurácia em 3 dos 4 modelos. Apenas o de Redes Neurais apresentou seus melhores resultados utilizando as medias 2 e 5, com 55,26% de acurácia.

Comparando então o caso com dois ou um campeonato, a Figura 16 apresenta os gráficos das acurácias utilizando os dois cenários. Percebe-se que utilizando dois campeonatos, ou seja, uma base maior para o treinamento gerou modelos com maior acurácia. Os modelos de Redes Neurais, Regressão Logística e XGBoost tiveram ganhos com uma base maior; apenas o SVM se manteve com resultados praticamentes iguais.

Na Tabela 12 são apresentadas a sensibilidade e a especificidade dos melhores modelos para cada um dos 4 utilizados. Em geral, todos os modelos perderam acurácia em relação

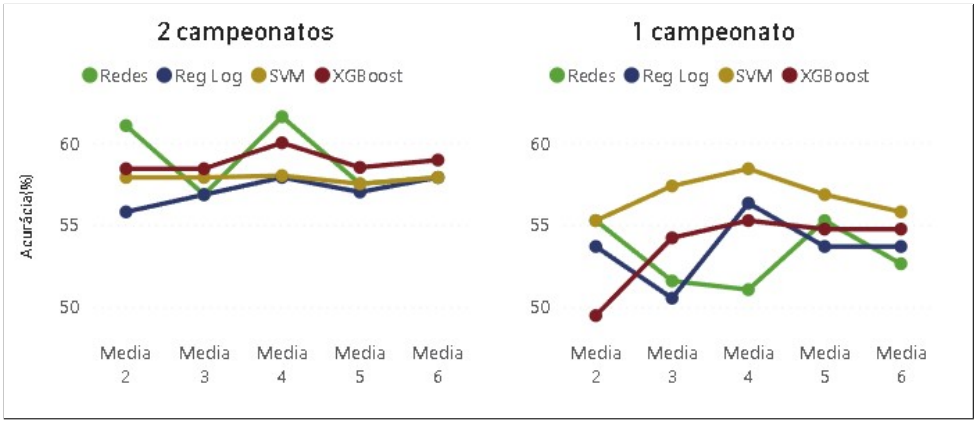

Figura 16: Gráficos com a comparação da acurácia dos modelos gerados a partir de 1 e 2 campeonatos

à classe Mandante e não apresentaram mudanças relevantes em relação às outras duas classes. Por isso os resultados com dois campeonatos foram melhores. A exceção ficou para o SVM, que manteve praticamente os mesmos valores para sensibilidade e especificidade de todas as classes, sendo então o destaque perante a perda de acurácia dos outros.

Tabela 12: Sensibilidade e Especificidade dos modelos a partir de um campeonato

Modelos	Medidas	Classes		
		Empate	Mandante	Visitante
Regressão Logística	Sensibilidade	0.0000	0.7684	0.5484
	Especificidade	1.0000	0.4842	0.7344
SVM	Sensibilidade	0.0000	0.8842	0.4355
	Especificidade	1.0000	0.3368	0.8750
Redes Neurais	Sensibilidade	0.0000	0.7579	0.5323
	Especificidade	1.0000	0.4632	0.7344
XGBoost	Sensibilidade	0.0000	0.8105	0.4516
	Especificidade	1.0000	0.3474	0.8203

3.3.2 Caso 2 - Acurácia comparando a variável Resultado com três classes contra duas

Como os modelos não conseguiram prever empates e tiveram dificuldades com a classe Visitante, decidiu-se por unir as observações da classe Empate com a Visitante, formando então a Não Mandante. Como os modelos conseguem prever com acurácias acima dos 80% para Mandante, então a decisão foi de restringir a análise a verificar quando é mais provável o Mandante vencer e quando não, na tentativa de aumentar a acurácia dos modelos em geral. Os resultados estão na Tabela 13, utilizando dois campeonatos.

Pode-se verificar uma melhora significativa dos resultados em relação à acurácia utili-

Tabela 13: Acurácia dos modelos para os campeonatos de 2016/2017 e 2017/2018 para duas classes

-	Regressão L.	SVM	Redes Neurais	XGBoost
Média2	0.6579	0.6684	0.6474	0.6105
Média3	0.6737	0.6263	0.6263	0.6105
Média4	0.6474	0.6526	0.6211	0.6211
Média5	0.6421	0.6526	0.6158	0.6316
Média6	0.6526	0.6211	0.6211	0.6368

zando bases com duas classes. Quanto à média das partidas anteriores a ser utilizada e o modelo, médias menores apresentaram melhores resultados em 3 dos 4 modelos, como a média de duas partidas anteriores que gerou o melhor resultado com a Regressão Logística, com uma acurácia de 67,37%. Em seguida vieram o SVM e Redes Neurais, e com os piores resultados o modelo XGBoost.

Os valores da Tabela 14 apresentam a sensibilidade e a especificidade para o presente caso, onde, como só há duas categorias para a variável alvo, a classe positiva é a Mandante e a negativa a Não Mandante. Portanto a sensibilidade indica a taxa de acertos para a classe positiva e a especificidade para a negativa. Então, ao comparar os valores obtidos com os da Tabela 10, verifica-se uma diminuição considerável dos acertos de Mandante e um também considerável valor para previsões da classe Não Mandante, que representa Visitante e Empate.

Dado que, com duas classes, o número de observações para cada uma delas ficou equilibrado, enquanto que havia um número maior de Mandante em relação à Visitante e Empate na base com 3 classes, há indícios de que o modelo tenha aprendido de mais sobre previsões para a classe Mandante por conta de ter mais observações. Lembrando que foi realizada uma tentativa de rebalancear a base, diminuindo a diferença de observações para as 3 classes, porém os resultados não foram satisfatórios.

Tabela 14: Sensibilidade e Especificidade dos modelos

	Regressão L.	SVM	Redes Neurais	XGBoost
Sensibilidade	0.5684	0.4632	0.5579	0.4842
Especificidade	0.7789	0.8421	0.7368	0.7895

Na Figura 17 é possível perceber o aumento considerável de acurácia dos modelos gerados a partir de bases com a variável alvo com duas classes em relação às com 3. Com duas classes, todos os modelos atingiram acurácia acima dos 60%, com SVM e Regressão Logística ficando próximos dos 70%. Enquanto que com 3 classes, no geral os resultados foram abaixo dos 60%.

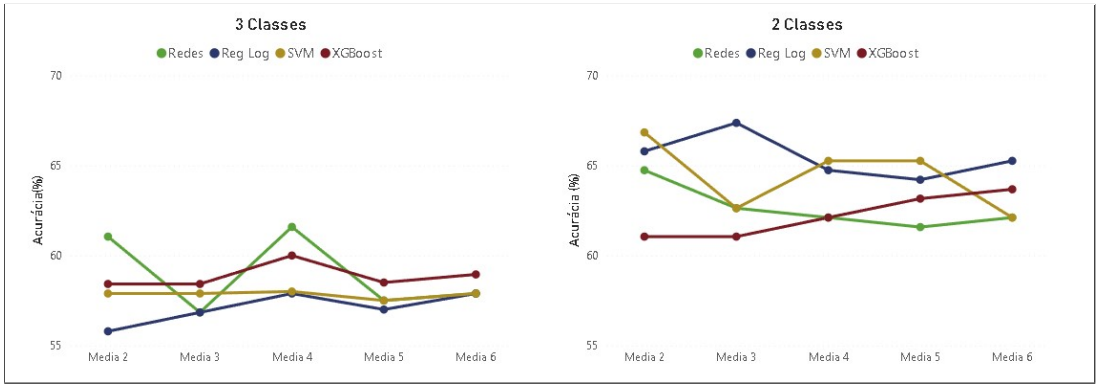

Figura 17: Gráficos com a comparação da acurácia dos modelos gerados utilizando três e duas classes

3.3.3 Caso 3 - Acurácia comparando os resultados em relação às médias das partidas anteriores

Nesta subseção verifica-se qual a melhor média de partidas anteriores obteve os melhores resultados em média para os modelos para os casos com um ou dois campeonatos e para a base tendo três ou duas classes na variável alvo.

A Figura 18 mostra os resultados para a comparação entre as bases com um e dois campeonatos, e a média que obteve os melhores resultados em média para os 4 modelos utilizados foi a de 4 partidas anteriores.

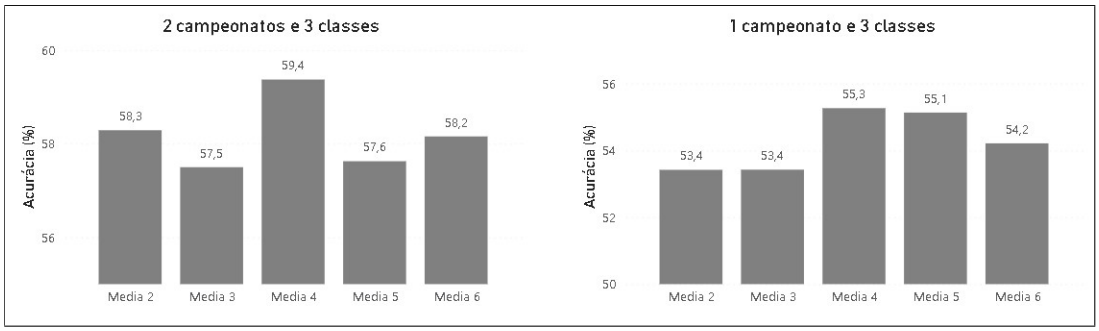

Figura 18: Gráficos da acurácia média em relação ao tamanho da base

Já a Figura 19 mostra os resultados entre três e duas classes. Neste caso, há diferença entre as melhores médias de partidas anteriores obtidas, onde quando se utilizam três classes, o ideal parece ser utilizar a média de 4 partidas anteriores, e quando se têm duas, o melhor parece ser utilizar uma média menor, onde a de duas partidas anteriores em média obteve os melhores resultados para a acurácia.

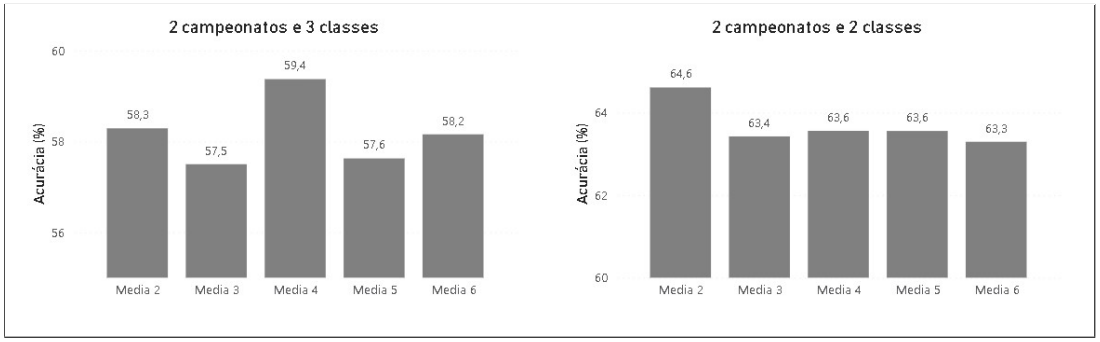

Figura 19: Gráficos da acurácia média em relação ao tamanho da base

3.4 Melhor Modelo: Redes Neurais

O modelo, utilizando o caso com dois campeonatos e três classes que será utilizado para realizar a comparação com os trabalhos utilizados como referência, com os melhores resultados foi o de Redes Neurais. Para conseguir esse feito, foi necessário trabalhar com diferentes combinações de valores para os hiperparâmetros do modelo, até que chegasse ao resultado mais satisfatório.

Sobre os hiperparâmetros, segundo a documentação do *Scikit Learn* [1], eles são parâmetros definidos pelo usuário antes do processo de treinamento de um modelo e que influenciam o comportamento e a performance do modelo. São diferentes dos parâmetros do modelo, que são ajustados automaticamente durante o treinamento. São importantes pois precisam ser definidos previamente e podem ter um impacto significativo na capacidade do modelo de fazer previsões precisas. Existem as definições padrões para se estimar um modelo, porém para buscar melhores resultados, é necessário alterá-los. Sobre a função, a utilizada é a nnet (VENABLES; RIPLEY, 2002a) . Essa função pertence ao pacote *caret* (Kuhn; Max, 2008) , que por sua vez utiliza uma outra função de redes neurais como base, do pacote *MASS* (VENABLES; RIPLEY, 2002b), que possui uma quantidade maior de hiperparâmetros para serem testados. O que a função do *caret* faz é uma otimização iterativa, tentando estimar os melhores valores para os hiperparâmetros originais, com o intuito de obter os melhores resultados.

Os hiperparâmetros da função *nnet* dentro do pacote caret são:

- Size: indica a quantidade de neurônios presentes na camada oculta
- decay: é um parâmetro utilizado para tentar evitar sobreajuste do modelo.

[1] https://scikit-learn.org/stable/modules/grid$_search.html$

Esses valores são utilizados dentro da função *train*, que, segundo a página RDocumentation [2], é utilizada para realizar o treinamento dos modelos, possuíndo uma série de parâmetros customizáveis, sendo os utilizados no trabalho os seguintes:

- data: indica os dados utilizados para o treinamento do modelo.
- method: indica qual o modelo utilizado
- trControl: (opcional). Recebe valores, a partir da função trainControl, de métodos que são aplicados nos dados, como no caso do trabalho a Validação Cruzada K-Fold, onde foram utilizados os seguintes hiperparâmetros :
  - method: o método de reamostragem que será aplicado
  - number: número de Folds
  - repeats: número de repetições
- tuneGrid: (opcional). Indica os valores dos hiperparâmetros escolhidos para o modelo

E ainda são utilizados os seguintes parâmetros na função *train*, originais da função do pacote *MASS* e mais específicos para o modelo de Redes Neurais:

- maxit: indica a quantidade máxima de iterações

Vale ressaltar que os primeiros hiperparâmetros dos modelos, não só o de Redes Neurais, são obtidos através do hiperparâmetro *tuneLenght* da função *train*, que gera uma quantidade de modelos predeterminada, mostrando o melhor modelo gerado e os valores dos hiperparâmetros utilizados. Após essa etapa, a partir dos valores obtidos, é utilizado o *tuneGrid*.

Outro ponto importante é que o modelo construído a partir do *nnet* do pacote *caret* resulta em uma rede neural com apenas uma camada, sendo possível apenas alterar o número de neurônios presentes nessa camada.

A Tabela 15 mostra os valores dos hiperparâmetros utilizados para gerar o melhor modelo:

E para finalizar, a Tabela 16 apresenta a comparação do resultado final real da temporada 2017/2018 da Premier League a partir das previsões realizadas pelo modelo que

[2] https://www.rdocumentation.org/packages/caret/versions/4.47/topics/train

Tabela 15: Valores dos hiperparâmetros que geraram o melhor modelo de Rede Neural

Função/Hiperp.	**Hiperparâmetro**	**Valor**
tuneGrid (train)	Size	1
	decay	0.6
train	data	base_media_4
	method	nnet
	maxit	1000
trControl (trainControl)	method	repeatedcv
	number	10
	repeats	3

gerou os melhores resultados a partir da utilização de dois campeonatos e 3 classes na variável alvo, que foi o de Redes Neurais utilizando uma média de 4 partidas anteriores, além de levar em consideração o mando de campo, para a realização dos cálculos.

Tabela 16: Comparação entre os resultados originais e previstos finais

Posição	Real	Previsão	Diferença
1	Man City	Man City	0
2	Man United	Liverpool	+2
3	Tottenham	Arsenal	+3
4	Liverpool	Man United	-2
5	Chelsea	Chelsea	0
6	Arsenal	Leicester	+3
7	Burnley	Burnley	0
8	Everton	Tottenham	-5
9	Leicester	Huddersfield	+7
10	Newcastle	Southampton	+7
11	Crystal Palace	Everton	-3
12	Bournemouth	Watford	+2
13	West Ham	Crystal Palace	-2
14	Watford	Brighton	-1
15	Brighton	Newcaslte	-5
16	Huddersfield	Swansea	+2
17	Southampton	West Bromwich	+3
18	Swansea	Bournemouth	-6
19	Stoke City	West Ham	-6
20	West Bromwich	Stoke City	-1

Pode-se verificar que o modelo conseguiu acertar 3 dos 5 melhores times do campeonato, incluindo o campeão, que garantem vaga para disputar as competições europeias do ano seguinte. Já entre os rebaixados, apenas um foi previsto corretamente. Além disso, a coluna Diferença apresenta a diferença entre a posição prevista e a observada para cada um dos times, verificando-se que a maioria dos resultados obteve uma diferença igual ou

inferior a 3 posições da real. Porém 6 previsões, com valores em vermelho, apresentaram uma discrepância mais elevada, entre elas 2 times previstos como rebaixados.

4 Conclusões

Dado o problema de prever resultados de jogos de futebol, objetivando saber o vencedor da partida ou se ela terminará empatada, ao se fazer uma análise dos resultados obtidos através dos modelos de Aprendizado de Máquinas e compará-los com os conseguidos nos trabalhos utilizados como referência, o objetivo de atingir uma acurácia acima dos 60% foi conquistado através do modelo de redes neurais, com 61,59% de acurácia. Lembrando que o melhor entre as referências obteve 62% de acurácia, de (ARAUJO et al., 2018), enquanto que o pior, de (HUCALJUK; RAKIPOVIC, 2011b), obteve 58%.

Em relação à base de dados, foram utilizadas 33 variáveis, havendo uma boa participação tanto de variáveis representando características dos mandantes (17), quanto de visitantes (16), mostrando que possuir variáveis relacionadas aos dois times que jogam a partida é importante. Sobre as características das variáveis, a maioria delas representavam informações de ataque, sendo somente 5 das 33 sobre defesa.

Já para a média das partidas anteriores, que foi o método encontrado para contornar o problema de não se possuir as estatísticas dos jogos ainda não realizados, impossibilitando a previsão dos resultados das mesmas com a metodologia utilizada, foi encontrado que a média das últimas 4 partidas para substituir o valor em cada observação de cada variável obteve melhores resultados em todos os modelos, ao se utilizar a base padrão com dois campeonatos e 3 classes.

Além disso, após verificar que os modelos tinham grande dificuldade para prever resultados da classe Empate, com todos obtendo uma taxa de acerto de 0% para a classe, também dificuldades para prever resultados para a classe Visitante, porém conseguindo bons resultados para a Mandante, foram realizados testes com bases contendo duas categorias: Mandante e Não Mandante, esta última formada pelas classes Visitante e Empate. E o desfecho foi positivo em relação à acurácia, com uma melhora nos resultados em todos os modelos, com o de Regressão Logística por exemplo obtendo uma acurácia de 67,37%. Porém, para as medidas de desempenho a sensibilidade (previsão para a classe

Mandante) apresentou uma grande redução, enquanto a especificidade, que representa os acertos para a classe Não Mandante, apresentou bons resultados. Então há a indicação de que a redução de três para duas classes melhorou a acurácia dos modelos e os bons resultados da classe Mandante quando se tinham 3 classes ocorreram por conta dessa classe ter um número maior de observações em relação às demais.

Em relação à quantidade de campeonatos a serem utilizados, uma base com dois campeonatos, gerando então uma base de treino maior, gerou melhores resultados para a acurácia dos modelos quando se quer realizar a previsão de um turno inteiro de um campeonato.

Para trabalhos futuros, seria interessante o teste com variáveis que tivessem mais foco nos jogadores e nas escalações das equipes, já que o projeto atual utilizou como variáveis apenas estatísticas dos times durante os jogos, desconsiderando por exemplo se o time era titular ou reserva, se o principal ou principais jogadores estavam em campo, se os melhores jogadores estão tendo bons desempenhos nas últimas partidas ou na atual temporada, a quantidade de jogos que o time fez na semana, entre outras variáveis.

Referências

ARAUJO, R. et al. Proposta de ferramenta de predição utilizando machine learning. *Universidade Federal do Rio Grande*, 2018.

BABOOTA, R.; KAUR, H. Predictive analysis and modelling football results using machine learning approach for english premier league. *International Journal of Forecasting*, 2018.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. *The Elements of Statistical Learning*: Data mining, inference, and prediction. [S.l.]: Springer, 2009.

HUCALJUK, J.; RAKIPOVIC, A. Predicting football scores using machine learning techniques. *Universidade de Zagreb, Faculdade de Engenharia Eletrica e Computaçao*, 2011.

HUCALJUK, J.; RAKIPOVIC, A. Predicting football scores using machine learning techniques. *Universidade de Zagreb, Faculdade de Engenharia Eletrica e Computaçao*, 2011.

J, G. et al. *An Introduction to Statistical Learning*: with applications in r. [S.l.]: Springer, 2013.

JOSEPH, A.; FENTON, N.; NEIL, M. Predicting football results using bayesian nets and other machine learning techniques. *Knowledge Based Systems*, 2006.

Kuhn; Max. Building predictive models in r using the caret package. *Journal of Statistical Software*, v. 28, n. 5, p. 1–26, 2008. Disponível em: ⟨https://www.jstatsoft.org/index.php/jss/article/view/v028i05⟩.

LIMA, J. H. M. Aplicaçao de machine learning para apostas esportivas. *Universidade Federal de Pernambuco, Centro de Ciencias Sociais Aplicadas*, 2022.

MARINHO, T. L. Otimização de hiperparâmetros do xgboost utilizando metaaprendizagem. *Universidade Federal de Alagoas. Instituto de Computação. Maceio*, 2021.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, v. 12, p. 2825–2830, 2011. Disponível em: ⟨https://scikit-learn.org/stable/modules/feature_selection.html#rfe⟩.

R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria, 2014. Disponível em: ⟨http://www.R-project.org/⟩.

SCHNEIDER, C. F. Machine learning aplicado na previsão de resultados de partidas de futebol: um estudo de caso para comparação de diferentes classificadores. *Universidade Federal do Rio Grande do Sul, Escola de Engenharia, Departamento de Engenharia Elétrica*, 2018.

VENABLES, W. N.; RIPLEY, B. D. *Modern Applied Statistics with S*. Fourth. New York: Springer, 2002. ISBN 0-387-95457-0. Disponível em: ⟨https://www.stats.ox.ac.uk/pub/MASS4/⟩.

VENABLES, W. N.; RIPLEY, B. D. *Modern Applied Statistics with S*. Fourth. New York: Springer, 2002. ISBN 0-387-95457-0. Disponível em: ⟨https://www.stats.ox.ac.uk/pub/MASS4/⟩.

ZIVKOVIC, J. *worldfootballR: Extract and Clean World Football (Soccer) Data*. [S.l.], 2022. R package version 0.6.2. Disponível em: ⟨https://CRAN.R-project.org/package=worldfootballR⟩.