UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

THIAGO BARP

# Preditor de posicionamento de times em campeonatos baseado em temporadas anteriores

Monografia apresentada como requisito parcial para a obtenção do grau de Bacharel em Engenharia da Computação

Orientador: Prof. Dr. Claudio Fernando Resin Geyer
Co-orientador: Prof. Dr. Julio Cesar Santos dos Anjos

Porto Alegre
2022

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitor: Prof. Carlos André Bulhões
Vice-Reitora: Profª. Patricia Pranke
Pró-Reitora de Graduação: Profª. Cíntia Inês Boll
Diretora do Instituto de Informática: Profª. Carla Maria Dal Sasso Freitas
Coordenador do Curso de Engenharia de Computação: Prof. André Inácio Reis
Bibliotecário-chefe do Instituto de Informática: Alexsander Borges Ribeiro

## AGRADECIMENTOS

Aos meus pais e minha irmã pelo apoio e incentivo que me deram durante toda a graduação e especialmente durante esse trabalho.

Aos amigos, que sempre estiveram ao meu lado, pelo apoio demonstrado ao longo de todo o período que me dediquei a este trabalho.

Ao professor Claudio Geyer e Julio Anjos, por terem sido meu orientador e coorientador e terem desempenhado tal função com dedicação e disponibilidade.

Aos demais professores, pelas correções e ensinamentos que me permitiram apresentar um melhor desempenho no meu processo de formação profissional e acadêmica ao longo do curso.

## RESUMO

No mundo do futebol é possível ver que poucos clubes acabam dominando a maior parte das ligas do mundo. Nos últimos dez anos os mesmos cinco times que ganharam campeonatos domésticos se repetem. No lado oposto, os outros times acabam lutando para não serem rebaixados ou são times que não são ruins a ponto de serem rebaixados, mas nem bons o suficiente para ganharem títulos. Esse trabalho busca confirmar a seguinte afirmação: é possível predizer com alta acurácia quais times irão ficar entre os quatro primeiros em cada temporada. Para isso será feito uma análise completa dos indicadores que podem afetar esses resultados. Partindo do retrospecto do time na temporada anterior será feita uma base com informações consideradas relevantes, exemplo delas seriam os pontos, gols e a divisão do campeonato que o time se encontra. Posteriormente será feita uma análise sobre os seguintes modelos de Inteligência Artificial: Rede Neural, Floresta Aleatória e Floresta com XGBoost. O primeiro será baseado no modelo de aprendizagem profunda sequencial disponibilizado pelo Keras. O segundo utilizara o modelo RandomForestClassifier desenvolvido no SKLearn. O último utilizará o código disponibilizado pelo XGBoost, o XGBoostClassifier. Com base nos resultados obtidos serão apresentadas conclusões sobre cada um dos modelos e sobre o trabalho em geral. Por último serão apresentadas algumas possibilidades de melhoria para futuros trabalhos, disponibilizando ideias que podem melhorar a acurácia do modelo.



**Palavras-chave:** Futebol, Aprendizagem Profunda, Floresta Aleatória.

**Predicting the positioning of teams in leagues based on previous seasons**

## ABSTRACT

In the world of football it is possible to see that few clubs end up dominating most of the leagues in the world. Over the last ten years the same five teams have repeatedly won domestic championships. On the opposite side, the other teams end up fighting so they don't get relegated or they are teams that are not bad enough to be relegated, but not good enough to win titles. This paper seeks to confirm the following statement: it is possible to predict with high accuracy which teams will be among the top four in each season. For this, a complete analysis of the indicators that can affect these results will be carried out. Based on the team's history in the previous season, a base will be created with information considered relevant, such as points, goals and the league division in which the team is. Subsequently, an analysis will be made on the following models of Artificial Intelligence: Neural Network, Random Forest and Forest with XGBoost. The first will be based on the sequential deep learning model provided by Keras. The second will use the RandomForestClassifier model developed in SKLearn. The last one will use the code provided by XGBoost, the XGBoostClassifier. Based on the results obtained, conclusions will be presented on each of the models and on the paper in general. Finally, some improvement possibilities for future papers will be presented, providing ideas that can improve the accuracy of the model.

**Keywords:** Football, Deep Learning, Random Forest.

## LISTA DE ABREVIATURAS E SIGLAS

| | |
|---|---|
| IA | Inteligencia Artificial |
| GE | Globo Esporte |

## LISTA DE FIGURAS

**SUMÁRIO**

# 1 INTRODUÇÃO

O futebol é um dos esportes mais populares do mundo (LABOTZ et al., 2019). A Figura 1.1 exemplifica isso mostrando o ranking das cinco contas com o maior número de seguidores no Instagram, sendo que duas dessas contas pertencem a jogadores de futebol.

Figura 1.1: Contas populares do Instagram


Os jogadores que apareceram na Figura 1.1 são dois dos mais famosos e mais talentosos de todos os tempos. Qualquer pessoa que acompanhou futebol nos últimos anos já ouviu falar deles. Eles foram os vencedores de 12 dos últimos 13 prêmios de melhor jogador do mundo, 7 desses prêmios foram para Lionel Messi.

Ao redor do mundo, milhões de pessoas torcem para seu time do coração conseguir ganhar um titulo. Os campeonatos onde esses títulos são possíveis normalmente se resumem a um número de times competindo entre si para definir quem será o grande campeão da temporada.

O campeonato se estrutura com dois jogos contra cada time da liga: um no seu próprio estádio e um no estádio do oponente. Ao longo do campeonato, os jogos geram pontos: cada vitória representa 3 pontos para o time, cada empate representa 1 ponto e os times não recebem pontos quando perdem jogos. As ligas nacionais comumente tem 20 times, duram em torno de 10 meses e são realizadas uma vez por ano.

O time que joga em seu próprio estádio costuma ter um resultado melhor, conforme exemplificado no artigo (CARMICHAEL; THOMAS, 2005) a porcentagem de vitória dos times que jogam em casa chegam a quase 70%.

Atualmente, na Europa, existem 5 ligas onde jogam os maiores times do mundo. São elas: a Inglesa (Premier League), Espanhola (Laliga), Italiana (Série A), Francesa (League One) e Alemã (Bundesliga). Os principais times dessas ligas dominam o futebol Europeu, representando 100% dos campeões do principal torneio de futebol Europeu desde 2004.

Aumentando o período, tirando o Porto, time que foi campeão da Liga dos Campeões em 2004, todas os times presentes nas finais da competição desde a temporada 1996/97 são representantes das ligas previamente citadas.

## 1.1 Hegemonias

Dentro das ligas ao redor do mundo também existem hegemonias. Por exemplo, das cinco citadas anteriormente somente a Premier League teve mais que 3 times diferentes sendo campões nos últimos 10 anos.

Na Série A, a Juventus havia vencido as últimas 9 competições seguidas, porém na temporada que começou em 2020 essa hegemonia acabou com o título da Internazionale. Na Bundesliga o Bayern possui atualmente uma hegemonia que dura 10 anos.

O sucesso do time nos últimos anos o levou a ser objeto de vários estudos, desde marketing, como visto no artigo (BAENA, 2019), até responsabilidade social, como presente no artigo (D'AMICO; CINCIMINO, 2017).

A maior sequência de títulos da liga nacional atualmente pertence ao Ludogorets, time que ganhou as últimas 11 temporadas do campeonato Búlgaro. Se continuar assim por mais cinco anos ele conseguirá igualar a maior hegemonia da história do futebol mundial, que atualmente pertence pertence ao Tafea, time do país Vanuatu - Oceania, que ganhou o campeonato por 15 temporadas consecutivas, começando em 1994 e terminando em 2009.

## 1.2 Predição dos resultados

Nos campeonatos citados acima, seria possível prever os campeões na maior parte dos casos se escolhêssemos sempre o mesmo time, e majoritariamente o time que ganhou o campeonato em um ano ficou entre os quatro melhores no ano seguinte. Essas constatações indicam que com base nos resultados das temporadas recentes é possível fazer uma predição para a colocação final do time.

Por exemplo: a Juventus, time da primeira divisão do futebol italiano, que vinha de uma sequência de nove títulos, ficou campeã pela última vez na temporada 2019-20, onde teve uma grande queda de rendimento que resultou no pior desempenho do time nos últimos anos e, consequentemente, na temporada seguinte, realizada em 2020-21, a

equipe conquistou a quarta colocação ao invés do tradicional primeiro lugar.

A longa sequência da Juventus se deve também ao ótimo trabalho financeiro do clube, como visto no artigo (DELICE; GERÇEK, 2018) o investimento nos jogadores certos é muito importante para a sequência de títulos que o time conseguiu.

### 1.2.1 Quatro primeiros colocados

Predizer os quatro primeiros times é uma tarefa mais simples do que predizer o campeão da temporada seguinte. Usualmente, os times que fazem uma boa campanha durante uma temporada tendem a ter uma boa campanha na temporada seguinte.

O Manchester City ficou em primeiro lugar em quatro das últimas cinco temporadas da Premier League e na temporada em que não foi campeão o time finalizou o campeonato em segundo lugar. Na temporada atual, o time se encontra com 17 de possíveis 21 pontos, por isso acaba sendo um ótimo exemplo de time consistente.

As evoluções dos times também podem ser notadas olhando o desempenho ao longo das últimas temporadas. Por exemplo: o Arsenal, time da primeira divisão do futebol inglês, ficou em 8º colocado com 61 pontos na temporada 2020-21, melhorou para 5º colocado com 69 pontos na temporada 2021-22 e atualmente está em primeiro colocado com 18 pontos de 21 possíveis na temporada 2022-23.

A tendência é de que ambos os times fiquem entre os quatro primeiros colocados na temporada atual e, observando o desempenho deles nas temporadas recentes, isso poderia ser coerentemente predito.

### 1.2.2 Promoções e Rebaixamentos

No campeonato brasileiro times que ficam entre os quatro últimos colocados da primeira divisão são rebaixados para a segunda divisão. Esses times rebaixados normalmente tem um elenco bom e acabam se destacando perante os times da segunda divisão, por isso tem uma tendência maior de ficar entre os quatro melhores naquela temporada.

Isso acontece de maneira similar nos outros países: um número x de times é rebaixado de uma divisão enquanto o mesmo número de times é promovido para essa divisão, para manter o número de equipes constante.

Existem casos de times promovidos ficarem entre os quatro primeiros já na pri-

meira temporada em uma nova divisão, porém esses casos normalmente são as exceções. O mais comum são times recém promovidos lutarem contra o novo rebaixamento ou, no máximo, ficarem em uma posição intermediária.

### 1.2.3 Quatro últimos colocados

A predição dos últimos colocados acaba sendo um pouco mais difícil, não existe uma constância tão grande devido ao fato de que os times que ficam entre os últimos de um campeonato são rebaixados.

Um dos melhores indicadores para predizer isso acaba sendo se o time foi promovido de divisão na temporada, a competição acaba sendo mais forte conforme o time sobe e muitas vezes o time ter uma boa campanha em uma divisão mais fraca não significa que ele vai conseguir manter o aproveitamento na divisão superior.

Da mesma forma que um time pode ter uma melhora no desempenho através das temporadas ele pode ter uma piora também, um exemplo disso é o Schalke, time da primeira divisão da Alemanha, durante a temporada 2017-18 ele ficou em segundo colocado, na temporada seguinte em 2018-19 ficou em 14º, em 2019-20 ficou em 10º e em 2020-21 ficou em 18º, assim sendo rebaixado para a segunda divisão alemã.

## 1.3 Informações Gerais

Esse trabalho fará uma análise sobre outros produtos já existentes hoje e que tem função similar a ele. Explicará como será feito e citará boas praticas na hora da definição das variáveis. Apresentará um pouco sobre alguns tipos de modelos para predizer as probabilidades de um time ficar entre os quatro melhores em uma temporada. Fará uma análise sobre outros artigos que já foram publicados e acabam tendo objetivos similares ou podem de alguma forma auxiliar o desenvolvimento. Por fim, apontará conclusões sobre o desenvolvimento do modelo e apontara possíveis melhorias para aumentar a acurácia.

No segundo capítulo será feito uma breve análise do estado da arte, referenciando e analisando artigos que tratam de assuntos similares ao apresentado nesse trabalho para ter uma visão geral de como está a literatura.

Ainda nesse capítulo serão analisados e apresentados produtos similares disponíveis na internet para contextualizar a situação do mercado para o modelo idealizado.

No fim do capítulo será apresentada uma visão geral do modelo e uma introdução às tecnologias utilizadas para obter os resultados esperados. Junto com isso serão apresentadas algumas diretrizes que foram levadas em consideração como boas práticas durante a realização do trabalho.

A seguir no terceiro capítulo será apresentada uma descrição detalhada sobre a seleção de cada variável e sua importância. Será explicado como foi feito o tratamento para conseguir deixar a informação coerente para todos os campeonatos e qual a importância disso.

No quarto capítulo será explicado todo o processo de criação e validação do modelo. Partindo do tratamento da base, será exemplificado com figuras o desenvolvimento e os testes do modelo. Serão ainda apresentadas as deduções feitas a partir dos resultados apresentados.

No quinto capítulo será feita uma conclusão sobre o trabalho como um todo, passando sobre pontos positivos, pontos negativos e a visão geral. Na sequência serão apresentadas propostas de melhorias possíveis para futuros trabalhos.

# 2 ESTADO DA ARTE E ANÁLISE DE MERCADO

Na atualidade, a Inteligência Artificial é objeto de muita discussão devido à sua capacidade de trazer resultados imparciais. Para exemplificar um dos tantos casos onde AI poderia ser aplicada, utliza-se o setor bancário que precisa aprovar o crédito dos seus clientes. Nesse caso, a IA poderia ser uma aliada quando estuda, analisa e entende os padrões do perfil de cliente que deve ter o crédito aprovado ou não. Um modelo mais avançado poderia inclusive sugerir o valor a ser aprovado.

## 2.1 Estado da Arte

Abaixo será feito a analise de alguns artigos com assunto relacionado ao e

### 2.1.1 Inteligência Artificial no Futebol

Decisões humanas geralmente são decisões parciais, que levam em consideração sentimentos, tendências e opiniões próprias. A IA analisa os dados disponíveis, reduzindo muito essa parcialidade, trazendo, por consequência, um resultado mais coerente com a verdade, o que se torna uma vantagem quando uma decisão deve ser tomada.

### 2.1.2 Auxilio de Pequenos Times

No artigo (KS, 2020) é feita uma pesquisa sobre as diversas aplicações que a Inteligência Artificial pode ter para o Futebol. Dentre elas está a utilização na caça de talentos, pois a utilização de IA poderia facilitar o processo de busca desses jogadores para times pequenos.

Os jogadores de destaque acabam ganhando mídia e são vendidos aos clubes de maior expressão. Utilizando Inteligência Artificial para entender as necessidades dos times menores é possível buscar no mercado profissionais de menor destaque que complementariam o time para conseguir ter um melhor desempenho na temporada sem precisar exceder os gastos planejados.

### 2.1.3 Apostas Esportivas

Um assunto popular entre os artigos encontrados é a utilização de Inteligência Artificial para apostas esportivas. O artigo (KNOLL; STüBINGER, 2019) foca nas características dos jogadores para prever o resultado do jogo.

Utilizando as principais ligas na Europa, ou seja, Premier League (Inglaterra), Laliga (Espanha), League One (França), Bundesliga (Alemanha) e Série A (Itália) e as ligas de segunda divisão dos mesmos países, para criar a base da plataforma, foram utilizadas doze temporadas e mais de 45 mil jogos. Além dos jogos foram utilizadas as características dos jogadores, por exemplo idade, velocidade, força do chute, dentre outros.

E, no final, é feita uma análise detalhada das probabilidades de retorno financeiro das apostas esportivas utilizando o método defendido pelo artigo, mostrando que um lucro de 30% poderia ter sido obtido na temporada 2013/14 da Premier League.

### 2.1.4 Bases Disponíveis

O artigo (DUBITZKY et al., 2019) cita a ”Open International Soccer Database”, que é uma base disponível com jogos desde o ano de 2000. Possui um total de 1470 times de 52 ligas em um total de 35 países.

Essa base tem como objetivo disponibilizar dados para tornar possível as análises citadas previamente. A lista dos países presentes varia desde os paıses mais importantes como Brasil, Inglaterra, Itália e Alemanha, ate países com menor tradição no futebol como Argélia, Tunísia e Venezuela.

Na Figura 2.1 é possível ver no modelo como são guardadas as informações nessa base de Dados. Na parte inferior da Figura 2.1 estão apresentadas as siglas para ficar mais claro.

## 2.2 Produtos Similares

Para os fãs do esporte, existem plataformas que oferecem diversas informações sobre o time e seus jogadores, como o desempenho e performance nas temporadas anteriores, os resultados dos jogos e as estatísticas do time. Muitas dessas informações são baseadas em resultados já obtidos e fica a critério do leitor fazer a análise futura do seu

Figura 2.1: Open International Soccer Database

| Sea | Lge | Date | HT | AT | HS | AS | GD | WDL |
|---|---|---|---|---|---|---|---|---|
| ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 16-17 | ENG1 | 18/03/2017 | West Bromwich Albion | Arsenal | 3 | 1 | 2 | W |
| 16-17 | ENG1 | 18/03/2017 | Crystal Palace | Watford | 1 | 0 | 1 | W |
| 16-17 | ENG1 | 18/03/2017 | Everton | Hull City | 4 | 0 | 4 | W |
| 16-17 | ENG1 | 18/03/2017 | Stoke City | Chelsea | 1 | 2 | −1 | L |
| 16-17 | ENG1 | 18/03/2017 | Sunderland | Burnley | 0 | 0 | 0 | D |
| 16-17 | ENG1 | 18/03/2017 | West Ham United | Leicester City | 2 | 3 | −1 | L |
| 16-17 | ENG1 | 18/03/2017 | Bournemouth | Swansea City | 2 | 0 | 2 | W |
| 16-17 | ENG1 | 19/03/2017 | Middlesbrough | Manchester United | 1 | 3 | −2 | L |
| 16-17 | ENG1 | 19/03/2017 | Tottenham Hotspur | Southampton | 2 | 1 | 1 | W |
| 16-17 | ENG1 | 19/03/2017 | Manchester City | Liverpool | 1 | 1 | 0 | D |
| ... | ... | ... | ... | ... | ... | ... | ... | ... |

Sea: Season. Lge: 3-letter code for country and league/division from top
Date: Date in DD/MM/YYYY format. HT/AT: Name of home/away team
HS/AS: Goals scored by home/away team, GD: Goal difference as HS − AS, WDL: Match result: W = home win, D = draw, L = away win (loss)

time do coração ou sobre qualquer time que lhe interesse.

Uma plataforma que possua uma análise futura teria uma vantagem competitiva se comparada às plataformas atuais, já que essa ofereceria, além do tradicional, uma predição imparcial do futuro desempenho da equipe escolhida.

### 2.2.1 Aposta 10

O site Aposta10 apresenta dicas esportivas mais voltadas para apostas esportivas. Possuindo especialistas no assunto o site traz algumas dicas do que eles acreditam que pode acontecer. Na Figura 2.2 é se apresentam alguns exemplos disso.

Ao abrir as dicas da aposta é feita uma apresentação breve dos times, dando um contexto, mostrando o que está acontecendo no momento, a provável escalação e algumas outras informações, como exemplificado na 2.3.

E, é possível também entender o que levou esses analistas a terem a opinião apresentada pelo site. Na Figura 2.4 é mostrado o pensamento do analista ao escolher um time ou outro.

### 2.2.2 FiveThirtyEight

Um exemplo que está disponível atualmente de uma análise similar ao que será feito nesse trabalho pode ser encontrado no site (FIVETHIRTYEIGHT, 2022), como mostrado na Figura 2.5. No site são apresentadas as probabilidades do time ser rebaixado,

Figura 2.2: Dicas do Aposta10

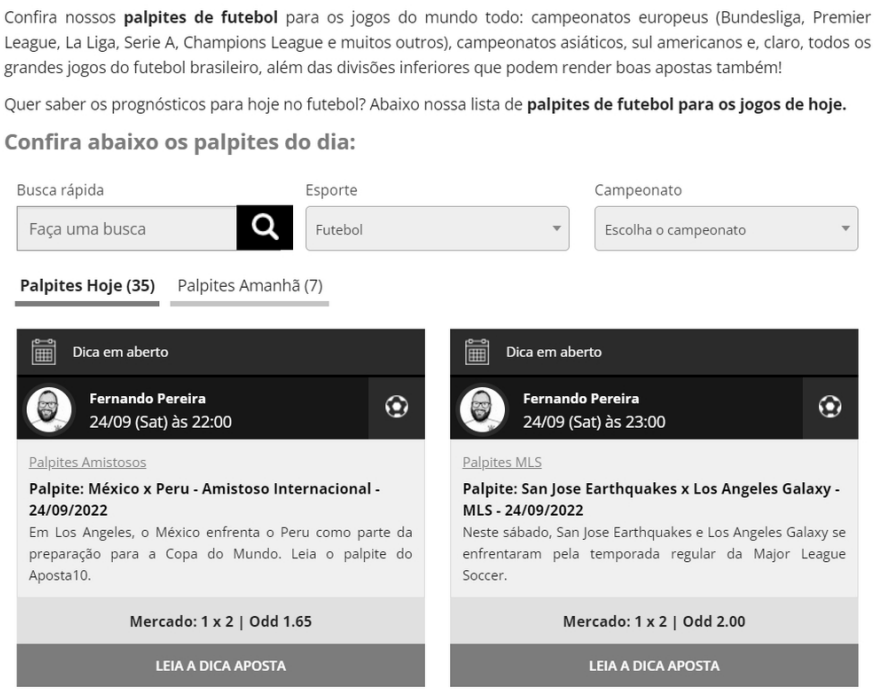
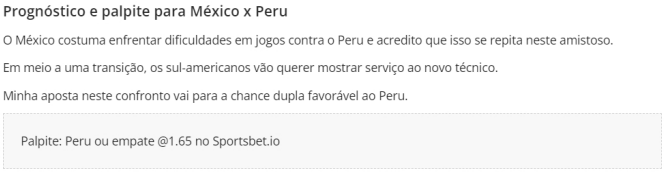


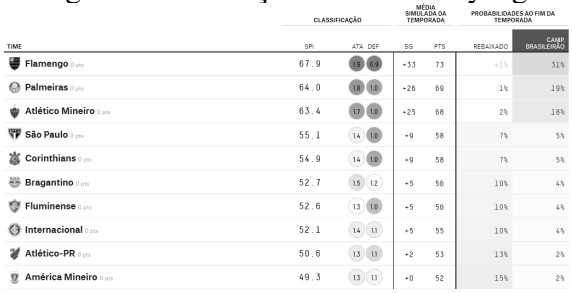

Figura 2.3: Apresentação dos times pelo Aposta10


Figura 2.4: Prognóstico e palpite do Aposta10



as probabilidades do time ganhar o campeonato e as probabilidades do time ficar entre os quatro primeiros colocados.

Figura 2.5: Predição do FiveThirtyEight



| | CLASSIFICAÇÃO | | | MÉDIA SIMULADA DA TEMPORADA | | PROBABILIDADES AO FIM DA TEMPORADA | |
|---|---|---|---|---|---|---|---|
| TIME | SPI | ATA | DEF | SG | PTS | REBAIXADO | CAMP. BRASILEIRÃO |
| Flamengo 0 pts | 67.9 | 1.9 | 0.9 | +33 | 73 | <1% | 51% |
| Palmeiras 0 pts | 64.0 | 1.8 | 1.0 | +26 | 69 | 1% | 19% |
| Atlético Mineiro 0 pts | 63.4 | 1.7 | 1.0 | +25 | 68 | 2% | 18% |
| São Paulo 0 pts | 55.1 | 1.4 | 1.0 | +9 | 58 | 7% | 5% |
| Corinthians 0 pts | 54.9 | 1.4 | 1.0 | +9 | 58 | 7% | 5% |
| Bragantino 0 pts | 52.7 | 1.5 | 1.2 | +5 | 56 | 10% | 4% |
| Fluminense 0 pts | 52.6 | 1.3 | 1.0 | +5 | 56 | 10% | 4% |
| Internacional 0 pts | 52.1 | 1.4 | 1.1 | +5 | 55 | 10% | 4% |
| Atlético-PR 0 pts | 50.6 | 1.3 | 1.1 | +2 | 53 | 13% | 2% |
| América Mineiro 0 pts | 49.3 | 1.3 | 1.1 | +0 | 52 | 15% | 2% |

### 2.2.3 Globo Esporte

Outro exemplo voltado somente para o futebol brasileiro é apresentado no Globo Esporte (GLOBOESPORTE, 2020). Como mostrado na Figura 2.6, referente à temporada de 2020, é possível observar a expectativa dos times e por quais posições eles devem estar disputando.

Figura 2.6: Predição do Globo Esporte


Além disso, para os interessados, o GE mostra o detalhamento da Figura 2.7 explicando os motivos que utiliza para fazer suas predições e dando uma breve explicação da sua classificação.

### 2.2.4 Bet Mines

Enquanto os outros exemplos citados partem de uma análise mais manual, o site Bet Mines se propõe a disponibilizar uma inteligência artificial para utilização do usuário.

Figura 2.7: Critérios do Globo Esporte


Como exemplificado na Figura 2.8 é possível selecionar uma variedade de eventos, datas e ligas.

Figura 2.8: BetMines Machine filtros


Uma vez selecionadas essas informações, o site apresenta jogos que atendam aos critérios selecionados para o usuário fazer uma seleção final e fazer as apostas que sejam de seu interesse.

## 2.3 Aprendizado de Máquina

Para obter o melhor resultado possível, é necessário saber fazer a escolha correta na definição das variáveis e também saber qual o melhor modelo para cada caso. A seguir será dada uma visão geral do modelo e das diretrizes utilizadas para a condução do trabalho.

### 2.3.1 Visão Geral

São vários os fatores que podem influenciar a derrota de um time: ausência de jogadores chave, uma atuação impecável de um goleiro geralmente instável, uma falha do time favorito a vencer o jogo, dentre outras.

Os fatores citados acima são imprevisíveis, mas outros podem ser utilizados para predizer o resultado com uma Inteligência Artificial. Por exemplo: o time A está bem no jogo, tem 75% de posse de bola, 15 chutes a gol e 90% dos passes foram acertados enquanto o time B não tem nenhum chute a gol e apenas 60% de passes acertados. Nesse caso, as chances de derrota do time A são muito baixas.

O artigo feito por (BERRAR; LOPES; DUBITZKY, 2019) traz dois novos modelos para a predição de resultados dos jogos e os compara com o que já existe disponível. Esses modelos propostos utilizam informações históricas para treinar e ensinar essa aplicação.

A ideia trazida no artigo é que, com base no que foi aprendido, essas aplicações façam predições de resultados de jogos. Essas predições acabam sendo similares à solução proposta nesse trabalho. A principal diferença entre elas é que no caso de (BERRAR; LOPES; DUBITZKY, 2019) o resultado predito é para um único jogo, enquanto a predição desse trabalho será a posição do time ao final do campeonato.

Outra diferença é na base de dados utilizada. O artigo utiliza uma base com as informações de cada jogo, enquanto o modelo idealizado utilizará uma base consolidada por temporada. Esses dados serão focados no desempenho do time na temporada anterior, enquanto no artigo é mais voltado para a performance no jogo anterior.

Para obter um bom resultado na predição é necessário escolher variáveis relevantes e cuidar com a redundância. Se uma redundância de informações for causada, o modelo pode ficar influenciado a agir dando muita importância para essas variáveis.

Por exemplo, as variáveis de gols feitos, gols sofridos e saldo de gols são bem

interessantes para um modelo. A medida de gols feitos indica a efetividade do ataque de uma equipe, enquanto os gols sofridos indicam sua solidez defensiva. O saldo de gols é uma representação da diferença entre ambas as variáveis, por isso acaba sendo redundante.

### 2.3.2 Modelo

Para decidir como prosseguir é necessário entender alguns conceitos sobre modelos de Inteligência Artificial. Um dos principais conceitos é a especificidade, que trata da importância da escolha correta das variáveis, abaixo será explicado com mais detalhes o que deve ser evitado.

#### *2.3.2.1 Especificidade*

Enquanto é feita a definição das variáveis é importante cuidar para não especificar demais o modelo. Caso isso seja feito, o modelo terá uma performance acima do esperado quando utilizando a base de treino, porém acabará tendo uma performance abaixo do esperado quando for utilizada uma base de testes diferente.

Isso ocorre porque o modelo entendeu que deveria prever "outliers"querendo sempre acertar todos os resultados e criando uma curva igual a da Figura 2.9.

Essa curva representa o modelo acertando todas as predições em uma base de treino. Porém em qualquer outra base regular, onde não ocorrem tantos valores fora do padrão, o modelo exemplificado teria uma acurácia baixa, pois consideraria que esses valores inesperados são um acontecimento natural e entenderia deveria predizer eles com o valor incorreto.

Ao mesmo tempo, é importante cuidar para que o modelo criado não seja pouco específico, pois, com isso, ele acabaria entendendo incorretamente como criar a curva de predição e criaria algo baseado em uma quantidade menor de informações do que o necessário.

Ao ser pouco específico o modelo criaria uma curva de predição como pode ser observada na Figura 2.10, onde ele considera apenas uma parte da informação e por isso não consegue ter uma acurácia muito alta.

É muito importante especificar corretamente as variáveis para o modelo conseguir criar a curva de predição correta e ter um resultado consistente. Como é possível observar

Figura 2.9: Modelo Muito Especifico

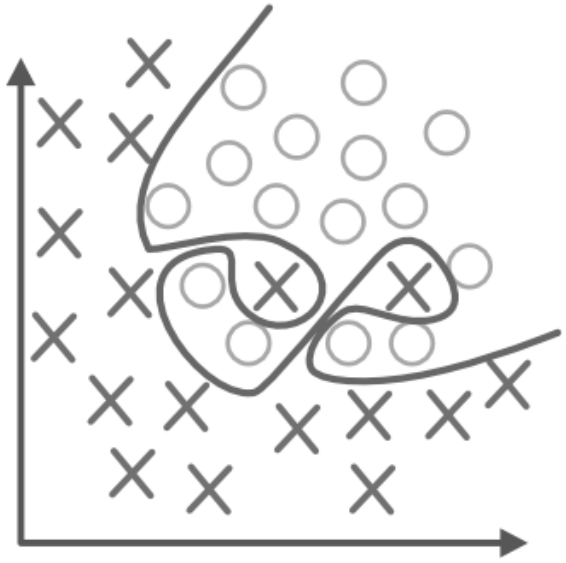
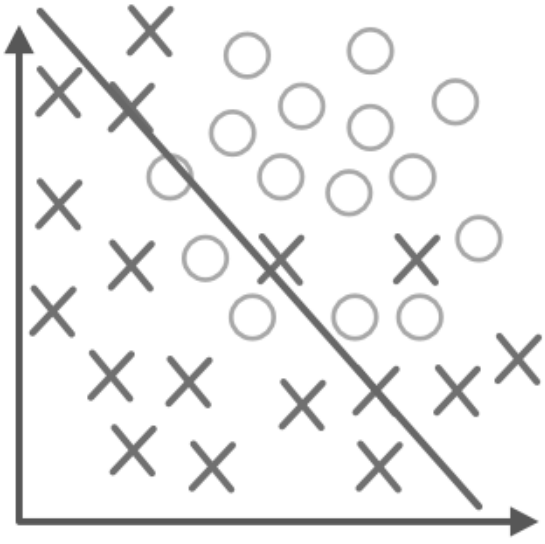
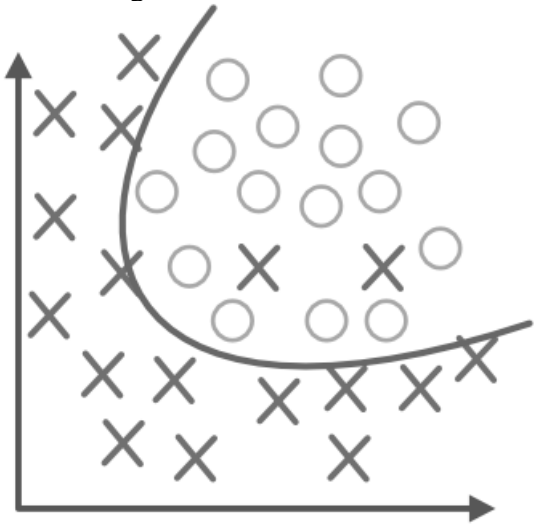


Figura 2.10: Modelo Generico



na Figura 2.11, mesmo sabendo da existência de predições incorretas na base de treino o modelo aceita isso para ter uma maior acurácia nas outras bases de testes.

Figura 2.11: Modelo Ideal



### 2.3.3 Escolha dos Algoritmos

Existem várias opções no mercado para criação de modelos de Aprendizado de Máquina. Abaixo será apresentada uma breve versão de cada uma das opções analisados para a criação do modelo proposto nesse trabalho.

#### *2.3.3.1 Aprendizagem Profunda*

Aprendizagem Profunda, ou "Deep Learning", é um ramo de Aprendizado de Máquina. Aplicações desse tipo buscam utilizar o computador para simular o cérebro humano.

Duas leituras muito interessantes para se aprofundar no assunto são (JANIESCH; ZSCHECH; HEINRICH, 2021) e (HAO; ZHANG; MA, 2016), pois ambas leituras buscam trazer uma visão mais abrangente sobre como funciona um modelo de "Deep Learning"e ajudam a entender mais sobre o assunto.

Assim como um cérebro humano, essas aplicações possuem nodos chamados de neurônios. Esses neurônios são responsáveis por transformar as entradas que eles recebem em um valor que irá ser interpretado posteriormente para gerar os resultados esperados.

Um exemplo da utilização de redes neurais é a leitura de números em imagens. Pessoas diferentes escrevem números de maneiras diferentes, mas se uma aplicação de

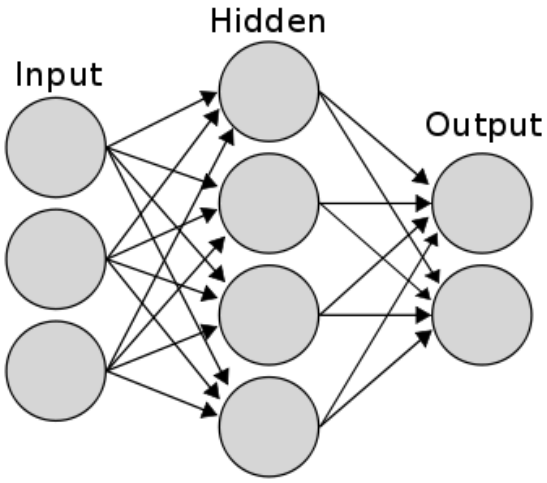
aprendizagem profunda for treinada o suficiente, ela vai conseguir reconhecer os números porque eles tem um padrão. Talvez ela confunda o número 1 e 7 por vezes, porém na maior parte dos casos ela consegue identificar o número.

A estrutura de uma rede neural normalmente é composta por pelo menos 3 camadas, como pode ser observado na Figura 2.12. A primeira camada é onde são colocados os valores de entrada.

As camadas intermediárias, representadas pela segunda camada da Figura 2.12, são as camadas onde existem os neurônios. Essas são as camadas responsáveis pela inteligência do modelo, e são esses neurônios que vão aprendendo os padrões e melhorando a acurácia com a sequência de treinos.

Por fim, a última camada, ela é responsável pela saída do modelo, depois dos valores terem passado por todos os tratamentos necessários, eles chegam nessa camada como o resultado.

Figura 2.12: Exemplo de Rede Neural




*2.3.3.2 Floresta Aleatória*

Para entender o modelo de Floresta Aleatória antes é necessário entender como funciona uma árvore de decisão. O artigo de (MYLES et al., 2004) detalha o funcionamento de árvores com foco na alteração delas para a utilização na química e na bioquímica.

Uma árvore de decisão é composta de um certo número de nodos, sempre com um nodo inicial e ramificações para baixo. Cada nodo pode significar uma pergunta ou uma ação e cada ramificação representa uma resposta ou trilha de ação.

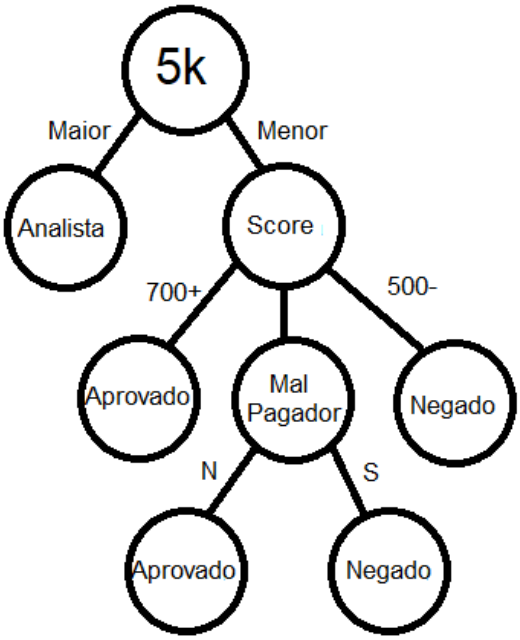
Um exemplo de utilização seria em uma decisão de crédito, onde é possível utilizar

uma árvore de decisão para substituir um analista humano em alguns casos. O nodo inicial da árvore seria utilizado o valor solicitado, e essa árvore só analisaria casos até 5 mil reais. Se o valor fosse maior que isso, a análise seria encaminhada para um analista de crédito.

A segunda variável seria o score de crédito disponibilizado pela Serasa: se o score fosse acima de 700 o crédito seria aprovado automaticamente, e se fosse abaixo de 500, seria negado automaticamente.

Se o score ficasse entre 500 e 700, o critério que determinaria a aprovação ou não do crédito seria o pagamento de contas: se o cliente já deixou de pagar alguma conta previamente, a solicitação seria negada e, se não houvesse inadimplência, a solicitação seria aprovada. A ilustração dessa árvore descrita está representada na Figura 2.13.

Figura 2.13: Exemplo de Árvore




Uma floresta aleatória é uma sequência de árvores como a descrita acima, porém cada árvore é definida de acordo com uma parcela aleatória da base de dados. Supondo que nessa base houvessem mais variáveis e fossem criadas 41 árvores diferentes baseadas nelas.

Entre todas, uma delas poderia ter a idade no lugar do score, outra o estado do cliente no lugar do histórico de pagamentos. Se a solicitação for abaixo de 5 mil, cada árvore retornaria um valor entre aprovado e negado e o valor com a maior representatividade dentro da floresta seria o escolhido.

Em resumo, uma floresta aleatória é apenas uma democracia entre um grupo de árvores, onde o valor mais votado é o escolhido. Para aprofundar mais no assunto de

florestas aleatórias o artigo (CUTLER; CUTLER; STEVENS, 2012) entra no detalhe do algoritmo e tudo o que acontece por trás desse modelo.

O artigo (BREIMAN, 2001) foca mais no detalhe da acurácia desse modelo, testando várias maneiras de fazer, desde entradas randômicas, utilizando uma porcentagem do dataset de treinos para criar as árvores, até variáveis randômicas, utilizando diferentes variáveis do dataset de treino para a criação das árvores.

#### *2.3.3.3 XGBoost*

O modelo implementado pelo XGBoost é bem similar ao implementado pela Floresta Aleatória. A principal diferença entre eles é que em uma floresta randômica os valores para criação das árvores são escolhidos aleatoriamente e a decisão é tomada a partir de uma escolha coletiva.

No XGBoost é criada uma sequência de árvores, uma tentando melhorar o resultado da outra com a intenção de ter a maior acurácia possível, e essa sequência de árvores resulta em apenas uma resposta no fim.

## 2.4 Considerações Finais

A escolha das variáveis corretas é muito importante para qualquer modelo, com a seleção correta é possível ter um modelo com uma acurácia muito alta independente dos testes aplicados, porém se as variáveis escolhidas não passarem as informações necessárias para o modelo, o resultado pode ser um modelo inconsistente.

A escolha do modelo é um pouco mais flexível para esse trabalho. Não é necessário um aprendizado tão detalhado quanto seria para um reconhecimento de números em uma imagem ou do sentimento apresentado em uma frase, por isso as opções de floresta se tornam mais viáveis.

Sabendo disso, será feita um grande estudo da base disponível para conseguir encontrar as variáveis mais importantes e tentar chegar ao melhor modelo possível.

# 3 CRIAÇÃO DO MODELO

O modelo idealizado tem a intenção de predizer se um time vai ficar entre os quatro primeiros colocados em um campeonato nacional na temporada seguinte. O grande ganho que esse modelo apresentará será uma visão geral sobre as expectativas dos times, então mesmo que o usuário não tenha conhecimento sobre o campeonato ele conseguirá ter uma noção do desempenho esperado por cada time.

Isso ajudaria na apresentação de uma plataforma, disponibilizando informações para um melhor entendimento de uma liga, permitindo que o usuário explore mais campeonatos e consiga aprimorar seus conhecimentos sem depender de muita pesquisa, tudo sendo oferecido por um modelo de Aprendizado de Máquina.

## 3.1 Base de dados

A seguir serão especificados os dados utilizados para a criação do modelo e todas as alterações que foram feitas para que a base ficasse da forma mais compreensível para o modelo. A base utilizada é composta por um total de 34 ligas que estão especificadas na Figura 3.1.

Figura 3.1: Lista de Ligas utilizadas

| LeagueName | CountryName | Division | | | LeagueName | CountryName | Division |
|---|---|---|---|---|---|---|---|
| Superliga | Albania | 1 | | | 2. Bundesliga | Germany | 2 |
| Superliga | Argentina | 1 | | | 3. Liga | Germany | 3 |
| Primera B Nacional | Argentina | 2 | | | Super League | Greece | 1 |
| Pro League | Belgium | 1 | | | Serie A | Italy | 1 |
| First Division B | Belgium | 2 | | | Serie B | Italy | 2 |
| Serie A | Brazil | 1 | | | Eredivisie | Netherlands | 1 |
| Serie B | Brazil | 2 | | | Eerste Divisie | Netherlands | 2 |
| Canadian Soccer League | Canada | 1 | | | Tweede Divisie | Netherlands | 3 |
| 1. HNL | Croatia | 1 | | | Primeira Liga | Portugal | 1 |
| 2. HNL | Croatia | 2 | | | Segunda Liga | Portugal | 2 |
| Premier League | England | 1 | | | La Liga | Spain | 1 |
| Championship | England | 2 | | | La Liga 2 | Spain | 2 |
| League One | England | 3 | | | Super League | Switzerland | 1 |
| League Two | England | 4 | | | Challenge League | Switzerland | 2 |
| Ligue 1 | France | 1 | | | Super Lig | Turkey | 1 |
| Ligue 2 | France | 2 | | | 1. Lig | Turkey | 2 |
| Bundesliga | Germany | 1 | | | Major League Soccer | USA | 1 |

Foram utilizadas as tabelas dos campeonatos desde 2012. As informações trazidas por campeonato foram: pontos, vitórias, empates, derrotas, gols marcados, gols sofridos e divisão. Para cada liga e cada temporada essas foram as informações consideradas essenciais.

### 3.1.1 Fonte de Dados

O modelo definido nesse trabalho utilizará uma base já criada e disponível em rede. Essa base possui jogos desde a temporada de 2011, como demonstrado na Figura 3.2, com muitas informações de cada jogo. Essas informações são gols, cartões, escalação, escanteios, números de falta de cada jogador, entre outras.

Figura 3.2: Resultados da Premier League

| | Nome Da Liga | Pais | Time Da Casa | Placar do Jogo | Time Visitante | Temporada |
|---|---|---|---|---|---|---|
| 1 | Premier League | England | Liverpool | 0x1 | Fulham | 2011/2012 |
| 2 | Premier League | England | Stoke City | 1x1 | Everton | 2011/2012 |
| 3 | Premier League | England | Chelsea | 0x2 | Newcastle United | 2011/2012 |
| 4 | Premier League | England | Bolton Wanderers | 1x4 | Tottenham Hotspur | 2011/2012 |
| 5 | Premier League | England | Everton | 4x0 | Fulham | 2011/2012 |
| 6 | Premier League | England | Stoke City | 1x1 | Arsenal | 2011/2012 |
| 7 | Premier League | England | Sunderland | 2x2 | Bolton Wanderers | 2011/2012 |
| 8 | Premier League | England | Swansea City | 4x4 | Wolverhampton Wanderers | 2011/2012 |
| 9 | Premier League | England | West Bromwich Albion | 0x0 | Aston Villa | 2011/2012 |
| 10 | Premier League | England | Wigan Athletic | 4x0 | Newcastle United | 2011/2012 |
| 11 | Premier League | England | Norwich City | 0x3 | Liverpool | 2011/2012 |
| 12 | Premier League | England | Chelsea | 6x1 | Queens Park Rangers | 2011/2012 |
| 13 | Premier League | England | Tottenham Hotspur | 2x0 | Blackburn Rovers | 2011/2012 |
| 14 | Premier League | England | Manchester City | 1x0 | Manchester United | 2011/2012 |
| 15 | Premier League | England | Arsenal | 3x3 | Norwich City | 2011/2012 |
| 16 | Premier League | England | Newcastle United | 0x2 | Manchester City | 2011/2012 |
| 17 | Premier League | England | Aston Villa | 1x1 | Tottenham Hotspur | 2011/2012 |

Além de ser uma base muito completa com relação aos jogos, ela possui uma vasta quantidade de ligas, tendo mais de 50 ligas ao redor do mundo com informações sendo atualizadas diariamente com base nos jogos que estão acontecendo.

O banco de dados selecionado para a extração dos dados é um banco individual e privado que foi disponibilizado ao autor desse trabalho apenas com a intenção de facilitar a criação do modelo.

### 3.1.2 Normalização dos dados

Muitas vezes ligas diferentes possuem regras diferentes. Uma das principais diferenças que existe são o número de times, por exemplo: a Bundesliga possui 18 times, a Série A do Brasil possui 20. Isso faz com que o número de jogos seja diferente entre essas ligas. No campeonato alemão são 34 rodadas totalizando um máximo de 102 pontos enquanto no campeonato brasileiro são 38 rodadas sendo o máximo de pontos 114.

Com um número diferente de pontos em disputa um time terminar a temporada com 75 pontos na Bundesliga significa aproximadamente 74% de aproveitamento, enquanto a mesma quantidade de pontos no brasileirão seria um total de 66% de aproveitamento.

Considerando isso, é necessário normalizar as informações por jogo, dividindo o total de pontos pelo número de jogos realizados. No caso citado previamente os times ficariam com 2,205 pontos por jogo no campeonato alemão e 1,973 pontos por jogo no

campeonato brasileiro.

Da mesma forma que os pontos tem que ser normalizados por jogo, os outros dados também precisam, as informações de diferentes ligas tem que estar coerentes para serem utilizadas pelo modelo. Isso significa que as variáveis finais foram: pontos por jogo, gols marcados por jogo e gols sofrido por jogo.

### 3.1.3 Jogos em casa e fora

Algumas vezes um time tem, na média, uma quantidade baixa de pontos por jogo, mas ele nunca acaba ficando entre os quarto últimos. Da mesma forma, um time pode ter, na média, uma quantidade alta de pontos por jogo, porém não fica entre os quatro primeiros.

Comumente isso é causado pelo desempenho do time fora de casa, já que um time pode passar uma temporada toda invicto em casa, porém não ganhar nenhum jogo disputado fora de casa.

Se o time não perder nenhum jogo em casa ele dificilmente vai ficar em uma das últimas posições do campeonato, mesmo que tenha um desempenho médio similar a outros times da parte inferior da tabela. Muitas vezes a torcida do time incentiva o suficiente e por isso o time consegue se manter nas divisões superiores.

Pelos motivos citados acima, a base foi incrementada com os valores normalizados por jogo e separados por jogos em casa e fora de casa. O modelo usa as mesmas variáveis citadas na seção anterior, pontos por jogo em casa, gols marcados por jogo em casa, gols sofridos por jogo em casa. Da mesma, forma foram criadas todas essas variáveis em jogos fora de casa.

### 3.1.4 Temporadas anteriores

As vezes os times têm temporadas irregulares, por exemplo: o Liverpool, time da primeira divisão do campeonato Inglês, na temporada 2018-19 fez um total de 97 pontos e ficou em segundo colocado. Na temporada seguinte em 2019-20 ele finalizou a temporada com 99 pontos e foi o campeão. Na temporada 2020-21 terminou o campeonato em terceiro lugar com 69 pontos porém na temporada 2021-22 ele conquistou 92 pontos e foi o vice campeão.

A irregularidade não ocorre somente tendo uma temporada abaixo do padrão. No caso do Fortaleza, time da primeira divisão do campeonato brasileiro, na temporada de 2020 ele finalizou o campeonato com 50 pontos ficando em 13º colocado na tabela. Na temporada seguinte ele conseguiu terminar a temporada com 58 pontos e ficar na 4º colocação no campeonato. Na temporada atual, o time se encontra com 31 pontos de 81 possíveis e está em 14 na classificação.

Essas irregularidades acontecem em todos os times do mundo, seja por atrito entre jogadores, mudança de técnico, saída de jogadores, entre outros motivos. Prever essas irregularidades é algo muito difícil, porém é possível mostrar ao modelo que essas irregularidades existem.

Para fazer isso, foram adicionadas as variáveis citadas previamente para a última temporada. As novas variáveis são pontos por jogo na última temporada, gols feitos por jogo na última temporada, gols sofridos na última temporada.

Adicionar somente uma temporada acaba limitando a acurácia, e pode tendenciar o modelo a achar que o desempenho fora do padrão do time na verdade é o normal. Para evitar esse acontecimento foram adicionadas as mesmas variáveis citadas para duas temporadas anteriores.

Com isso o modelo terá o comportamento do time em três temporadas e evitará que uma temporada irregular crie uma tendência e tente predizer que uma temporada com médias baixas pode realmente acabar entre os quatro piores.

### 3.1.5 Rebaixamentos e promoções

Como foi comentado na descrição, promoções e rebaixamentos tendem a colocar o time contra um nível de competição diferente, por isso é muito importante saber quando o time é promovido a uma nova divisão ou quando ele é rebaixado.

Por exemplo: a Chapecoense, time atualmente na segunda divisão do campeonato brasileiro, teve uma campanha onde conquistou 73 pontos na Série B na temporada de 2020 e foi a campeã, assim conseguindo acesso à primeira divisão. Ao chegar na Série A ela ficou na última colocação com apenas 15 pontos e foi rebaixada novamente para a série B.

Um outro exemplo já apresentado previamente pode ser utilizado para mostrar que o contrário também é verdade. O Schalke, citado na introdução, teve uma sequência de temporadas ruins e acabou sendo rebaixado para a segunda divisão do futebol alemão,

porém logo em sua primeira temporada na segunda divisão o time foi campeão da liga e consequentemente promovido novamente para a Bundesliga.

Sabendo disso, foram criados esses dois indicadores para balancear a importância do time estar em uma divisão mais competitiva ou com um nível de futebol mais baixo. A intenção desses indicadores é evitar que uma campanha ótima na segunda divisão tenha o mesmo peso que uma campanha ótima na primeira divisão.

## 3.2 Desenvolvimento do Modelo

Com as variáveis definidas é a hora de testar a aplicação ideal para utilizar nesse modelo. Primeiramente foi realizada a instalação do anaconda com a intenção de utilizar o Jupyter notebook devido a familiaridade do autor com a ferramenta.

Após a instalação do anaconda foram feitas as instalações necessárias para a utilização das aplicações, o Keras para as aplicações de aprendizagem profunda, o Sklearn para as aplicações de floresta randômica e XGBoost para floresta paralelas.

Assim que todas as ferramentas foram instaladas, foi iniciado o desenvolvimento de todas as lógicas em Python. Inicialmente, foi feito o tratamento das variáveis, foram adicionados os indicadores de rebaixamento e promoção, além de adicionar os dados das últimas temporadas para o treinamento e teste do modelo.

### 3.2.1 Tratamento da base

Junto com a criação das variáveis foi criado também um valor indicando se o clube ficou entre os quatro primeiros e um segundo valor indicando se o clube ficou entre os quatro últimos.

Para os quatro primeiros não houve problema, pois os quatro primeiros em qualquer liga sempre serão as mesmas posições. Porém para os quatro últimos essas posições variam, podendo ser a partir do 17 caso esteja em um campeonato com 20 times ou a partir do 15 caso esteja em um campeonato com 18 times.

Por isso, foi criado uma lógica para pegar a última posição da liga e, a partir disso, conseguir pegar quais são as últimas posições. A lógica utilizada para os tratamentos descritos até agora está representado na Figura 3.3.

O posicionamento do time ao final da temporada atual não tem muita relevância

Figura 3.3: Lógica inicial de tratamento da base

```
In [4]: leagueInfo = dataset[['CountryName', 'LeagueName', 'SeasonName', 'Position']]
        leagueLastPlace = pd.DataFrame(leagueInfo.groupby(['CountryName', 'LeagueName', 'SeasonName'], sort=False)['Position'].max())
        leagueLastPlace = leagueLastPlace.reset_index()
        #leagueLastPlace

In [5]: dataset['TopFour'] = 0
        dataset['BotFour'] = 0

        for index in dataset.index:
            for indexLeague in leagueLastPlace.index:
                if dataset.loc[index, ('Position')] <= 4.0:
                    dataset.loc[index, ('TopFour')] = 1
                elif (     (dataset.loc[index, ('Position')] >= (leagueLastPlace.loc[(indexLeague, ('Position'))]-3))
                           & (dataset.loc[index, ('CountryName')] == leagueLastPlace.loc[(indexLeague, ('CountryName'))])
                           & (dataset.loc[index, ('LeagueName')] == leagueLastPlace.loc[(indexLeague, ('LeagueName'))])
                           & (dataset.loc[index, ('SeasonName')] == leagueLastPlace.loc[(indexLeague, ('SeasonName'))])):
                        dataset.loc[index, ('BotFour')] = 1
        #dataset
```

para o modelo. O objetivo desse trabalho é predizer se o time vai ficar entre os quatro primeiros na temporada seguinte, por isso o passo seguinte é transformar essa informação em uma variável na base.

Uma vez que essa informação foi buscada é possível separar a base de treino, testes e o resultado esperado. Para isso foram criadas as variáveis x_treino, que são os valores de entrada para o modelo, y_treino, que é o resultado esperado para cada entrada, x_teste e y_teste que possuem o mesmo significado que os outros, porém relativos à base de teste e não à base de treino.

A base de testes será a base relativa à temporada 2020, pois a temporada 2021 já foi concluída e assim é possível confirmar o resultado apresentado. As outras temporadas farão parte da base de treinos. A lógica para realizar isso pode ser observada na Figura 3.4

Figura 3.4: Separação entre base de treinos e testes

```
In [9]: testData = dataset[ (       (dataset['SeasonName'] == 2020)
                                   )]
        trainData = dataset[~(     (dataset['SeasonName'] == 2021)
                                   | (dataset['SeasonName'] == 2020)
                                   )]

In [10]: features=['Division', 'PointsPerGame', 'GoalsScoredPerGame', 'GoalsConcededPerGame', 'WinPercentage', 'DrawPercentage', 'LossPerc

In [11]: x_train = trainData[features]
         y_train = trainData[['NextSeasonTopFour']]
         x_test = testData[features]
         y_test = testData[['NextSeasonTopFour']]
```

Uma vez que as bases já estão prontas e separadas, é necessário somente a criação do modelo e utilização dessas bases para testar. Foram utilizados três modelos e posteriormente foi avaliado o desempenho deles.

### 3.2.2 Modelo com Keras

O primeiro modelo testado foi o modelo de aprendizado profundo do Keras. Para essa avaliação foi utilizado um modelo padrão sequencial.

Nesse modelo foram adicionadas três camadas, uma que vai receber as variáveis de entrada com 8 nodos, uma intermediária com 4 nodos e uma camada para a saída com um único nodo.

A intenção dessa camada de saída é retornar um valor entre 0 e 1 representando a probabilidade do time ficar entre os quatro primeiros colocados. O código utilizado está apresentado na Figura 3.5

Figura 3.5: Modelo com Keras

```
In [12]: model = Sequential()

         model.add(Dense(8, input_dim=31, activation='relu'))

         model.add(Dense(4, activation='relu'))

         model.add(Dense(1, kernel_initializer='normal', activation='sigmoid'))

In [13]: model.compile(loss='binary_crossentropy', optimizer='adam',metrics=['acc'])
         model
Out[13]: <keras.engine.sequential.Sequential at 0x2049583a520>

In [14]: history = model.fit(x_train, y_train,validation_data = (x_test,y_test), epochs=150, batch_size=10)
```

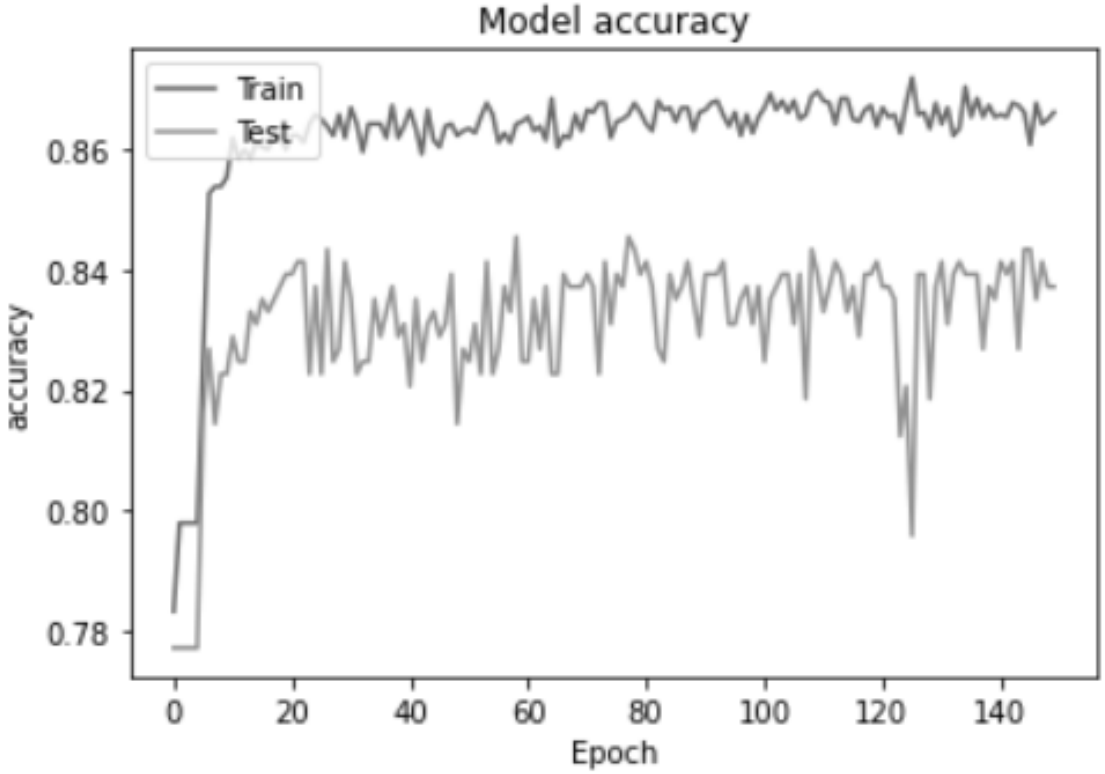
Na Figura 3.6 é possível acompanhar a evolução do aprendizado. Os resultados da base de treinos, representada pela primeira linha do gráfico, começam em torno de 78% e chegam a 87% durante algumas rodadas. Os resultados da base de testes, representada pela segunda linha do gráfico, são levemente piores, começando abaixo de 78% e chegando a aproximadamente 84%.

Como a base é composta por menos de 5 mil linhas, é possível ver que existe um aprendizado inicial porém depois os valores possuem apenas algumas variações. Com uma base mais extensa o aprendizado teria sido maior e consequentemente a acurácia teria aumentado.

Durante a definição de variáveis foi justificada a utilização de cada variável e na Figura 3.7 é possível observar as três variáveis com o maior impacto para as decisões tomadas.

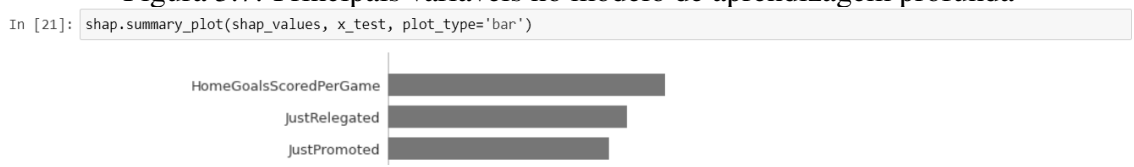
O modelo considerou que ter um ataque eficaz em casa é o principal indicador de sucesso do time em temporadas subsequentes. O modelo também colocou muito peso no fato do time ter sido promovido ou rebaixado na temporada, isso vai ao encontro com o que foi comentado para a seleção dessas variáveis: a diferença na qualidade dos

Figura 3.6: Teste modelo aprendizagem profunda




adversários impacta bastante o resultado esperado.

Figura 3.7: Principais variáveis no modelo de aprendizagem profunda




### 3.2.3 Modelo com Floresta Aleatória

O segundo modelo testado foi o modelo de floresta aleatória do Sklearn que foi implementado conforme a Figura 3.8. O software Sklearn oferece a montagem completa da floresta, basta o usuário indicar a base de testes e treino.

O software oferece também opções para a divisão das bases entre treino e teste, porém para a comparação ser justa foi mantida a mesma base utilizada no modelo de aprendizagem profunda.

A acurácia apresentada na figura 3.8 foi de 82%, porém saber somente essa informação não traz uma visão verdadeira do processo de escolha da floresta.

Figura 3.8: Modelo de Floresta Fleatória

```
In [50]: model = RandomForestClassifier(random_state=0)
         model.fit(x_train, y_train)

         A column-vector y was passed when a 1d array was expected. Please change the shape of y to (n_samples,), for example using rave
         l().

Out[50]: RandomForestClassifier(random_state=0)

In [51]: y_pred = model.predict(x_test)
         y_pred_proba = pd.DataFrame(model.predict_proba(x_test))

In [52]: print("Accuracy:", accuracy_score(y_pred, y_test))

         Accuracy: 0.8206185567010309
```

Na Figura 3.9 é possível observar a divisão detalhada. Na coluna "precision"é possível observar a acurácia do modelo quando prevendo o valor apresentado na linha. Por exemplo, o 0.83 significa que 83% das vezes que o modelo predisse que o time não ia ficar entre os 4 primeiros, ele realmente não ficou.

Na coluna "recall"é apresentado a acurácia do modelo sobre os valores esperados. É possível observar que o modelo acertou 96% dos times que não ficaram entre os quatro primeiros, porém acertou apenas 32% dos times que ficaram entre os 4 primeiros.

Na coluna "f1-score"é apresentada a média harmônica entre as duas primeiras colunas. A coluna "support"apresenta o número de acontecimentos de cada resultado na base. Nesse exemplo são 377 times que não ficaram entre os quatro primeiros e 108 que ficaram entre os quatro primeiros.

Figura 3.9: Explicação da Acurácia do modelo de Floresta Aleatória

```
In [53]: print(classification_report(y_test, y_pred))

                       precision    recall  f1-score   support

                  0.0       0.83      0.96      0.89       377
                  1.0       0.71      0.32      0.45       108

             accuracy                           0.82       485
            macro avg       0.77      0.64      0.67       485
         weighted avg       0.81      0.82      0.79       485
```

Isso significa que o modelo ficou muito conservador, ou seja, a menos que o time tenha uma campanha excepcional, a maior parte das árvores irão predizer que o time não ficará entre os quatro primeiros na próxima temporada.

Esse fato pode acontecer pois a maior parte da base de treinos utilizada é composta por times que não ficaram entre os quatro primeiros. Ou seja, as árvores dessa floresta entendem que devem indicar como 1 apenas os melhores times.

Na Figura 3.10 estão representadas todas as vezes que a floresta predisse que o time ia ficar entre os quatro primeiros colocados e ele não ficou.

Todos os casos estão coerentes com a predição esperada. Na maior parte das vezes foi uma temporada atípica que causou o erro da Floresta. O único caso que poderia ser

apontado como problemático é do time que foi promovido na temporada e ainda assim recebeu a maioria dos votos para ficar entre os quatro primeiros, Porém a campanha do time na divisão inferior realmente foi excepcional, então o erro é compreensível.

Figura 3.10: Predições incorretas pelo modelo de Floresta Aleatória

| TeamName | LeagueName | CountryName | Division | Points | GoalsScored | GoalsConceded | Games | Win | Draw | Loss | JustPromoted | JustRelegated | NextSeasonDivision |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Samsunspor | 1. Lig | Turkey | 2 | 70 | 58 | 30 | 34 | 20 | 10 | 4 | 0 | 0 | 2 |
| SC Cambuur | Eerste Divisie | Netherlands | 2 | 92 | 109 | 36 | 38 | 29 | 5 | 4 | 1 | 0 | 1 |
| De Graafschap | Eerste Divisie | Netherlands | 2 | 77 | 67 | 47 | 38 | 23 | 8 | 7 | 0 | 0 | 2 |
| AZ Alkmaar | Eredivisie | Netherlands | 1 | 71 | 75 | 41 | 34 | 21 | 8 | 5 | 0 | 0 | 1 |
| Huesca | La Liga | Spain | 1 | 34 | 34 | 53 | 38 | 7 | 13 | 18 | 0 | 1 | 2 |
| Lille | Ligue 1 | France | 1 | 83 | 64 | 23 | 38 | 24 | 11 | 3 | 0 | 0 | 1 |
| Philadelphia Union | Major League Soccer | USA | 1 | 40 | 40 | 18 | 20 | 12 | 4 | 4 | 0 | 0 | 1 |
| New York City | Major League Soccer | USA | 1 | 36 | 35 | 21 | 20 | 11 | 3 | 6 | 0 | 0 | 1 |
| Manchester United | Premier League | England | 1 | 74 | 73 | 44 | 38 | 21 | 11 | 6 | 0 | 0 | 1 |
| Royal Excel Mouscron | Pro League | Belgium | 1 | 31 | 32 | 54 | 34 | 7 | 10 | 17 | 0 | 1 | 2 |
| Vasco da Gama | Serie A | Brazil | 1 | 41 | 37 | 56 | 38 | 10 | 11 | 17 | 0 | 1 | 2 |
| Atalanta | Serie A | Italy | 1 | 78 | 90 | 47 | 38 | 23 | 9 | 6 | 0 | 0 | 1 |
| Galatasaray | Super Lig | Turkey | 1 | 84 | 80 | 36 | 40 | 26 | 6 | 8 | 0 | 0 | 1 |
| Besiktas | Super Lig | Turkey | 1 | 84 | 89 | 44 | 40 | 26 | 6 | 8 | 0 | 0 | 1 |

### 3.2.4 Modelo com XGBoost

O terceiro modelo testado utilizou o software XGBoost e a implementação dele está representada na Figura 3.11. Assim como na floresta aleatória a montagem já está pronta e tudo que o usuário precisa fazer é indicar a base de testes e de treino.

Para conseguir ter uma comparação válida a base de treino utilizada é a mesma que foi utilizado nos outros dois modelos.

Figura 3.11: Modelo utilizando XGBoost

```
In [23]: model = xgb.XGBClassifier()
         model.fit(x_train, y_train)

Out[23]: XGBClassifier(base_score=0.5, booster='gbtree', callbacks=None,
                       colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
                       early_stopping_rounds=None, enable_categorical=False,
                       eval_metric=None, gamma=0, gpu_id=-1, grow_policy='depthwise',
                       importance_type=None, interaction_constraints='',
                       learning_rate=0.300000012, max_bin=256, max_cat_to_onehot=4,
                       max_delta_step=0, max_depth=6, max_leaves=0, min_child_weight=1,
                       missing=nan, monotone_constraints='()', n_estimators=100,
                       n_jobs=0, num_parallel_tree=1, predictor='auto', random_state=0,
                       reg_alpha=0, reg_lambda=1, ...)

In [24]: y_pred = model.predict(x_test)

In [25]: print("Accuracy:",metrics.accuracy_score(y_pred, y_test))

         Accuracy: 0.8288659793814434
```

Como pode ser observado na Figura 3.11 o modelo utilizando XGBoost obteve uma acurácia maior que o modelo utilizando Floresta Aleatória. Isso é esperado visto que o modelo em si foi idealizado como uma melhoria das Florestas.

Como é mostrado na Figura 3.12, esse modelo perde um pouco em precisão referente aos times que ele disse que ficariam entre os quatro primeiros, porém ele ganha quando olha para o total de times que ficaram entre os quatro primeiros.

Comparando com os mesmos indicadores da árvore aleatória, a nova versão ganha

14% de acerto comparado ao total de times que finalizaram o campeonato entre os quatro primeiros e perde somente 4% de precisão entre os que ele predisse para ficarem entre os 4 melhores do campeonato.

Além desse ganho na predição houve uma pequena alteração na precisão dos times que não ficaram entre os quatro primeiros. Essa alteração é a mesma para as variáveis "precision"e "recall"mantendo um mesmo valor para o "f1-score". Pode ser observada a consistência nas bases de teste pelos valores da variável "support".

Figura 3.12: Predições incorretas pelo modelo de XGBoost

```
In [33]: print(classification_report(y_test, y_pred))
```

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0.0 | 0.86 | 0.93 | 0.89 | 377 |
| 1.0 | 0.67 | 0.46 | 0.55 | 108 |
| accuracy | | | 0.83 | 485 |
| macro avg | 0.76 | 0.70 | 0.72 | 485 |
| weighted avg | 0.82 | 0.83 | 0.82 | 485 |

A Figura 3.12 também aponta um melhor entendimento da base pelo algoritmo já que o acerto para times que ficaram entre os quatro melhores aumentou. Um dos problemas citados no modelo de Floresta Aleatória era o fato do modelo ter sido muito conservador na escolha dos times que as árvores escolheram como favoritos.

Com essa nova aplicação houve uma redução nessa tendência, exemplificado pela variável "recall"ter melhorado para o modelo. Como esperado, o modelo ficando menos conservador, o número de predições positivas incorretas aumenta.

Isso está exemplificado na Figura 3.13, onde pode-se observar um maior número de erros referentes a times em que a expectativa de ficar entre os quatro primeiros era alta porém acabaram decepcionando.

Novamente, é possível observar uma base bem coerente e com predições esperadas. Porém, no futebol, muitas coisas podem acontecer de uma temporada para outra e por isso é esperado que existam alguns times que decepcionem, da mesma forma como algumas vezes os times surpreendem e ficam entre os quatro primeiros quando ninguém esperava.

Figura 3.13: Predições incorretas pelo modelo de XGBoost

| TeamName | LeagueName | CountryName | Division | Points | GoalsScored | GoalsConceded | Games | Win | Draw | Loss | JustPromoted | JustRelegated | Top4Probability |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Samsunspor | 1. Lig | Turkey | 2 | 70 | 58 | 30 | 34 | 20 | 10 | 4 | 0 | 0 | 97.95% |
| Wolfsburg | Bundesliga | Germany | 1 | 61 | 61 | 37 | 34 | 17 | 10 | 7 | 0 | 0 | 60.72% |
| SC Cambuur | Eerste Divisie | Netherlands | 2 | 92 | 109 | 36 | 38 | 29 | 5 | 4 | 1 | 0 | 74.70% |
| De Graafschap | Eerste Divisie | Netherlands | 2 | 77 | 67 | 47 | 38 | 23 | 8 | 7 | 0 | 0 | 83.08% |
| AZ Alkmaar | Eredivisie | Netherlands | 1 | 71 | 75 | 41 | 34 | 21 | 8 | 5 | 0 | 0 | 73.10% |
| Athletic Club | La Liga | Spain | 1 | 46 | 46 | 42 | 38 | 11 | 13 | 14 | 0 | 0 | 90.11% |
| Huesca | La Liga | Spain | 1 | 34 | 34 | 53 | 38 | 7 | 13 | 18 | 0 | 1 | 91.19% |
| Lille | Ligue 1 | France | 1 | 83 | 64 | 23 | 38 | 24 | 11 | 3 | 0 | 0 | 91.80% |
| Lens | Ligue 1 | France | 1 | 57 | 55 | 54 | 38 | 15 | 12 | 11 | 0 | 0 | 56.65% |
| New York City | Major League Soccer | USA | 1 | 36 | 35 | 21 | 20 | 11 | 3 | 6 | 0 | 0 | 77.82% |
| Manchester United | Premier League | England | 1 | 74 | 73 | 44 | 38 | 21 | 11 | 6 | 0 | 0 | 74.81% |
| Leeds United | Premier League | England | 1 | 59 | 62 | 54 | 38 | 18 | 5 | 15 | 0 | 0 | 63.05% |
| Sheffield United | Premier League | England | 1 | 23 | 20 | 63 | 38 | 7 | 2 | 29 | 0 | 1 | 60.42% |
| Royal Excel Mouscron | Pro League | Belgium | 1 | 31 | 32 | 54 | 34 | 7 | 10 | 17 | 0 | 1 | 52.55% |
| Académica | Segunda Liga | Portugal | 2 | 62 | 44 | 28 | 34 | 17 | 11 | 6 | 0 | 0 | 70.00% |
| Vasco da Gama | Serie A | Brazil | 1 | 41 | 37 | 56 | 38 | 10 | 11 | 17 | 0 | 1 | 72.70% |
| Atalanta | Serie A | Italy | 1 | 78 | 90 | 47 | 38 | 23 | 9 | 6 | 0 | 0 | 81.79% |
| Lazio | Serie A | Italy | 1 | 68 | 61 | 55 | 38 | 21 | 5 | 12 | 0 | 0 | 81.80% |
| Crotone | Serie A | Italy | 1 | 23 | 45 | 92 | 38 | 6 | 5 | 27 | 0 | 1 | 55.33% |
| Galatasaray | Super Lig | Turkey | 1 | 84 | 80 | 36 | 40 | 26 | 6 | 8 | 0 | 0 | 97.79% |
| Besiktas | Super Lig | Turkey | 1 | 84 | 89 | 44 | 40 | 26 | 6 | 8 | 0 | 0 | 76.18% |
| Hatayspor | Super Lig | Turkey | 1 | 61 | 62 | 53 | 40 | 17 | 10 | 13 | 0 | 0 | 60.52% |
| Fatih Karagümrük | Super Lig | Turkey | 1 | 60 | 64 | 52 | 40 | 16 | 12 | 12 | 0 | 0 | 76.26% |
| Denizlispor | Super Lig | Turkey | 1 | 28 | 38 | 77 | 40 | 6 | 10 | 24 | 0 | 1 | 94.37% |
| Vllaznia Shkodër | Superliga | Albania | 1 | 66 | 44 | 22 | 36 | 19 | 9 | 8 | 0 | 0 | 52.41% |

## 3.3 Considerações Finais

Comparando diretamente os modelos de floresta obtemos uma acurácia maior para o modelo utilizando XGBoost. Ele tenta predizer melhor os times que vão ficar entre os top 4 e sacrifica muito pouco para fazer isso, assim se tornando o modelo mais recomendado entre os dois.

Referente ao modelo utilizando o Keras, ele obteve a maior acurácia dos três testados, porém também foi o mais instável. Como pode ser observado na Figura 3.6, ele varia bastante dependendo do teste.

A diferença de acurácia entre os modelos utilizando Keras e XGBoost é menor do que 1% então vai da preferencia do usuário qual utilizar. O XGBoost é melhor na questão desempenho, porém o Keras com a mesma base teve uma acurácia maior. Se a base fosse mais completa a diferença de resultado poderia ter sido maior.

# 4 CONCLUSÃO

A base utilizada no trabalho é muito completa quando se precisa saber qualquer informação de um jogo especifico, porém em relação aos dados de uma temporada ela deixa a desejar.

O obstáculo enfrentado nesse trabalho foi encontrar uma segunda base confiável para substituir a base possuída em pouco tempo. Por esse motivo, escolheu-se manter a base original e remover valores incorretos e temporadas não completas.

Considerando os fatos acima citados, a acurácia dos modelos deixaram a desejar. Os modelos são ótimos em prever times que não vão ficar entre os 4 melhores e são bons em identificar times que tem muita probabilidade de ficar.

O problema encontrado foram os times que ficam nessa fase intermediaria, e comumente tem boas campanhas, mas não são campanhas espetaculares. Os modelos muitas vezes classificavam esses times com baixa probabilidade de ficar entre os quatro mesmo que eles ficassem nessas posições por algumas temporadas.

Outro ponto a ser notado é que independente do tamanho da base, a quantidade de times que não ficou entre os quatro melhores sempre vai ser maior do que a quantidade que ficou, por isso os modelos tendem a ser mais conservadores para manterem a acurácia alta. Se o indicador fosse ficar na metade superior da tabela os resultados provavelmente seriam muito melhores.

## 4.1 Recomendação para trabalhos futuros

Como comentado previamente, tendo a base a partir de 2012 ainda é possível criar um modelo tendencioso, então a primeira recomendação seria buscar dados de mais temporadas para conseguir criar uma base de treinos maior e ter maior precisão com a informação histórica.

Outro ponto que poderia agregar muito seria a folha salarial do time, normalmente times com a folha salarial maior tendem a ter um desempenho melhor pois conseguem trazer jogadores melhores e isso, consequentemente, aumenta o nível de jogo do time.

Porém essa adição teria que ser cautelosa devido ao fato da distribuição do dinheiro estar pouco homogênea entre as ligas. A folha salarial do Manchester City está próxima de 1,4 bilhões de reais enquanto a do Atlético Mineiro é de 130 milhões, e ambos são os atuais campeões de suas ligas.

Por isso seria necessário encontrar alguma forma de normalizar esse dado. Uma opção seria dividir o valor pela maior folha salarial da liga e, quanto mais perto de 1 o numero, maior o salario do time, e, consequentemente, maior as expectativas.

Adicionar um desempenho ao longo do campeonato também auxiliaria bastante, um time começar o campeonato brasileiro com 50 pontos nos primeiros 20 jogos e terminar o campeonato com 60, ou seja, nos últimos 18 jogos ele fez apenas 10 pontos, significa que a probabilidade do time performar bem na próxima temporada é muito menor do que um time que teve o desempenho contrário e fez 50 pontos em 54 possíveis para finalizar o campeonato.

# REFERÊNCIAS

BAENA, V. Global marketing strategy in professional sports. lessons from fc bayern munich. **Soccer & Society**, Taylor & Francis, v. 20, n. 4, p. 660–674, 2019.

BERRAR, D.; LOPES, P.; DUBITZKY, W. Incorporating domain knowledge in machine learning for soccer outcome prediction. **Mach Learn**, v. 108, p. 97–126, 2019.

BREIMAN, L. **Machine Learning**, Springer Science and Business Media LLC, v. 45, n. 1, p. 5–32, 2001. Available from Internet: <https://doi.org/10.1023/a:1010933404324>.

CARMICHAEL, F.; THOMAS, D. Home-field effect and team performance: evidence from english premiership football. **Journal of sports economics**, Sage Publications Sage CA: Thousand Oaks, CA, v. 6, n. 3, p. 264–281, 2005.

CUTLER, A.; CUTLER, D. R.; STEVENS, J. R. Random forests. In: ____. **Ensemble Machine Learning: Methods and Applications**. Boston, MA: Springer US, 2012. p. 157–175. ISBN 978-1-4419-9326-7. Available from Internet: <https://doi.org/10.1007/978-1-4419-9326-7_5>.

DELICE, M. E.; GERÇEK, E. Transfer efficiency and triumph in sports: An experimental study of the italian serie a. **Eracle. Journal of Sport and Social Sciences**, v. 1, p. 68–81, 2018.

DUBITZKY, W. et al. The open international soccer database for machine learning. **Machine Learning**, v. 108, 01 2019.

D'AMICO, D. G.; CINCIMINO, C. P. D. S. **Corporate Social Responsibility in the Sport industry: the FC Bayern Munich case study**. [S.l.]: Università Degli Studi di Palermo, 2017.

FIVETHIRTYEIGHT. **Previsões dos times de futebol**. 2022. Available from Internet: https://projects.fivethirtyeight.com/previsoes-de-futebol. Available from Internet: <https://projects.fivethirtyeight.com/previsoes-de-futebol>.

GLOBOESPORTE. **Guia do Brasileirão**. 2020. Available from Internet: https://interativos.globoesporte.globo.com/futebol/brasileirao-serie-a/guia/guia-do-brasileirao-2020. Available from Internet: <https://interativos.globoesporte.globo.com/futebol/brasileirao-serie-a/guia/guia-do-brasileirao-2020>.

HAO, X.; ZHANG, G.; MA, S. Deep learning. **International Journal of Semantic Computing**, World Scientific Pub Co Pte Lt, v. 10, n. 03, p. 417–439, sep. 2016. Available from Internet: <https://doi.org/10.1142/s1793351x16500045>.

JANIESCH, C.; ZSCHECH, P.; HEINRICH, K. Machine learning and deep learning. **Electronic Markets**, Springer Science and Business Media LLC, v. 31, n. 3, p. 685–695, abr. 2021. Available from Internet: <https://doi.org/10.1007/s12525-021-00475-2>.

KNOLL, J.; STüBINGER, J. Machine-learning-based statistical arbitrage football betting. **KI - Künstliche Intelligenz**, v. 34, 07 2019.

KS, M. Applications of artificial intelligence in the game of football: The global perspective. **Researchers World – Journal of Arts Science and Commerce**, v. 11, p. 18–29, 07 2020.

LABOTZ, M. et al. The world's most popular sport. 2019.

MYLES, A. J. et al. An introduction to decision tree modeling. **Journal of Chemometrics**, Wiley, v. 18, n. 6, p. 275–285, jun. 2004. Available from Internet: <https://doi.org/10.1002/cem.873>.