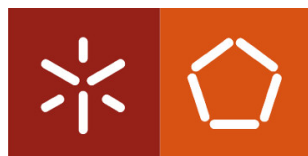


UNIVERSIDADE DO MINHO

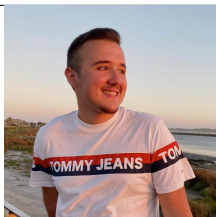
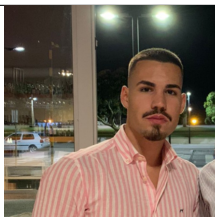

ESCOLA DE ENGENHARIA



Mineração de Dados

EC- Engenharia de Conhecimento
Mestrado em Engenharia Informática

Trabalho Prático - Grupo 11

PG50229	PG50499	PG51246
		
António Fernandes	João Torres	Mário Correia

Junho, 2023

Conteúdo

1	Introdução	2
1.1	Contextualização	2
1.2	Principais objetivos	2
1.3	Trabalhos Relacionados Relevantes	2
2	Fontes de Dados	4
2.1	Soccer players values and their statistics	4
2.2	Dataset - Classificação Equipas	4
3	Análise Exploratória dos dados	6
3.1	Boxplot & outliers	6
3.2	Gráfico de Barras - Média de valor de mercado por temporada	7
3.3	Gráficos tipo Pie - Distribuição do valor de mercado por temporada	7
3.4	Histogramas - Valores de Transferências	8
3.5	Gráfico de Barras - Top 10 países com maior número de atletas	8
3.6	Gráfico de Barras - Valores Médios de Mercado dos Jogadores por Posição	9
3.7	Gráfico de Barras - Valores Médios de Mercado dos Jogadores por Liga	10
4	Processamento de Dados	12
4.1	Tratamento de Missing Values	12
4.2	Feature Engeneering	12
4.3	Feature Selection	12
4.4	One Hot Encoding	12
4.5	Binning - Problema de Classificação	12
5	Modelação de Dados	14
5.1	Problema de Regressão	14
5.1.1	Redes Neurais	14
5.1.2	Random Forest	15
5.1.3	XGBoost	16
5.1.4	LightGBM	17
5.2	Problema de Classificação	18
5.2.1	Regressão Logística	18
5.2.2	Decision Tree	19
5.2.3	Random Forest	20
5.2.4	Support Vector Machine	21
5.2.5	XGBoost	22
5.2.6	LightGBM	23
6	Resultados e Conclusão	25
7	Trabalho Futuro	26

1. Introdução

Este trabalho prático surge no âmbito da unidade curricular de Mineração de Dados, pertencente ao perfil de Engenharia de Conhecimento do Mestrado em Engenharia Informática da Universidade do Minho.

O projeto desenvolvido teve como motivação um estudo que envolvesse recolha, processamento, análise e mineração de dados publicamente acessíveis com o objetivo de extrair conhecimento útil e não óbvio para os utilizadores, de preferência usando e integrando mais do que uma fonte de dados. Assim sendo, o grupo decidiu prever o valor de mercado de jogadores de futebol, tendo as suas especificações.

1.1 Contextualização

O futebol é, indiscutivelmente, o desporto mais popular e amplamente jogado em todo o mundo. Com amantes do desporto em todos os países, a sua influência transcende fronteiras e culturas, unindo pessoas de diferentes origens sob uma paixão comum. Toda esta popularidade e o interesse contínuo no desporto têm implicações significativas na valorização dos jogadores de futebol.

O valor de mercado de um jogador de futebol é determinado por uma combinação de fatores que envolvem, por exemplo, as suas características e habilidades, desempenho, idade, potencial de crescimento e a situação do mercado atual.

Assim sendo, é extremamente importante saber prever o valor de mercado de um jogador para que as equipas consigam fazer uma melhor gestão financeira do clube e consigam melhorar a negociação de possíveis transações.

1.2 Principais objetivos

O principal objetivo deste projeto é realizar o tratamento dos dados e, posteriormente, desenvolver e comparar vários modelos capazes de realizar a previsão do valor de mercado de jogadores de futebol, com base nos dados públicos disponíveis nos datasets. Através da aplicação de técnicas de mineração de dados e análise estatística, pretende-se extrair conhecimento útil e não óbvio a partir desses dados, de forma a proporcionar uma estimativa precisa e confiável do valor de mercado de um jogador.

1.3 Trabalhos Relacionados Relevantes

No âmbito da previsão do valor de mercado de jogadores de futebol, existem diversos trabalhos relacionados que têm contribuído para a evolução e compreensão deste campo.

Um dos trabalhos relevantes é o estudo realizado por Liu et al. (2017), que propõe um modelo baseado em algoritmos de *machine learning* para prever o valor de mercado de jogadores de futebol. O estudo utiliza características como idade, estatísticas de desempenho e informações sobre as equipas e ligas em que os jogadores atuam. Os resultados mostraram que o modelo conseguiu prever o valor de mercado com uma boa precisão, conseguindo demonstrar o potencial da abordagem.

Outro trabalho importante é o de Kostadinova e Stoev (2019), que exploraram a aplicação de técnicas de *deep learning* para a previsão do valor de mercado de jogadores de futebol. O estudo utiliza uma arquitetura de rede neuronal convolucional para extrair características

relevantes dos jogadores a partir de imagens dos seus rostos. Os resultados mostraram que a inclusão de informações visuais pode melhorar significativamente a precisão das previsões de valor de mercado.

Além disso, um estudo realizado por Ribeiro et al. (2020) investigou a importância das redes sociais na determinação do valor de mercado de jogadores de futebol. O estudo analisou a relação entre o desempenho dos jogadores nas redes sociais, medidas através do número de seguidores e do engajamento online, e o seu valor de mercado. Os resultados revelaram uma correlação positiva entre a popularidade nas redes sociais e o valor de mercado dos atletas, destacando a influência cada vez maior das redes sociais no mundo do futebol.

Estes trabalhos relacionados demonstram a diversidade de abordagens utilizadas na previsão do valor de mercado de jogadores de futebol e a importância de considerar diferentes variáveis e fontes de dados para obter previsões precisas.

2. Fontes de Dados

Nesta subsecção, serão apresentados os dois conjuntos de dados utilizados para a realização do nosso trabalho prático (*Data Acquisition*). A primeira fonte corresponde a informação de diversos jogadores e suas estatísticas dentro do campo e o segundo corresponde a um dataset criado pelo grupo com a informação da classificação das equipas nas diferentes ligas. O formato de todos os datasets utilizados é CSV.

2.1 Soccer players values and their statistics

Como primeiro conjunto de datasets (principal fonte de dados) selecionamos o *Soccer players values and their statistics* que corresponde a informação dos diversos jogadores nas seguintes ligas:

- Liga Espanhola - La Liga
- Liga Inglesa - Premier League
- Liga Alemã - Bundesliga
- Liga Italiana - Liga A
- Liga Francesa - Ligue 1

Esta fonte contém três datasets que correspondem as épocas de 2017-2018, 2018-2019 e 2019-2020. A fonte destes datasets é o Kaggle e as estatísticas retiradas são do TransferMarket. O número de entradas da época 17-18 é 2232, época 18-19 é 2232 e da época de 19-20 é 2644. Os três datasets contém 400 colunas e informações desde:

- Estatísticas físicas do jogador: Nome, idade, altura, pé dominante, entre outras.
- Estatísticas dentro do campo: todo o tipo de passes (de um certo número de metros, passes rasteiros, cruzamentos), assistências, golos, etc.
- Estatísticas sobre a equipa: liga, jogos ganhos pela equipa, etc.

Aqui apenas se apresenta um breve resumo das 400 colunas existentes no entanto é de realçar que estes datasets possuem todas as estatísticas relacionadas com um jogador dentro de campo desde as mais elementares até as mais pormenorizadas.

2.2 Dataset - Classificação Equipas

Esta segunda fonte de dados trata-se de um dataset criado pelo grupo que contém, para as diferentes ligas acima referidas e as diferentes equipas constituintes, a classificação da equipa na respetiva época. Esta informação encontra-se no site da FlashScore no arquivos das diferentes ligas. De forma a extrairmos esta informação, inicialmente tentámos técnicas de web scrapping, no entanto, não estava a ser possível uma vez que o ficheiro html desta página era dinâmico. Por este motivo, foi feita a criação manual do dataset. Este contém então as seguintes colunas: época, liga, equipa e classificação. As entradas correspondem as diversas equipas presentes nas ligas e épocas acima referidas que dá um total de 214.

Decidimos criar este dataset de forma a criarmos uma nova coluna nos datasets anteriores que substituam a equipa onde o jogador jogou, pela classificação da mesma nessa época. Desta forma, facilitaríamos o processo de aprendizagem dos diferentes modelos de machine learning. O processo de criação encontra-se no ficheiro python `equipas.py`.

3. Análise Exploratória dos dados

Nesta seção iremos explicar todo processo que envolve a análise e exploração dos dados, que tem como objetivo dar a conhecer características e padrões mais gerais dos dados em estudo.

3.1 Boxplot & outliers

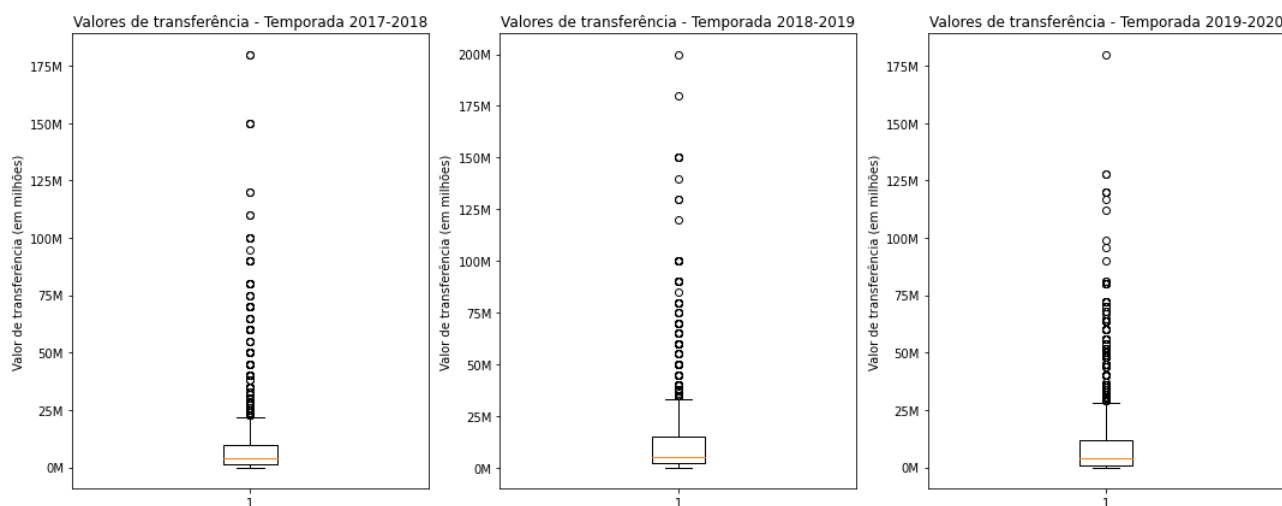


Figura 3.1: Outliers

Através do boxplot obtido conseguimos perceber que, regra geral, a maior parte dos jogadores têm valores de mercado entre os 0 e os 15 milhões. Acima dos 25 milhões são observados outliers.

3.2 Gráfico de Barras - Média de valor de mercado por temporada

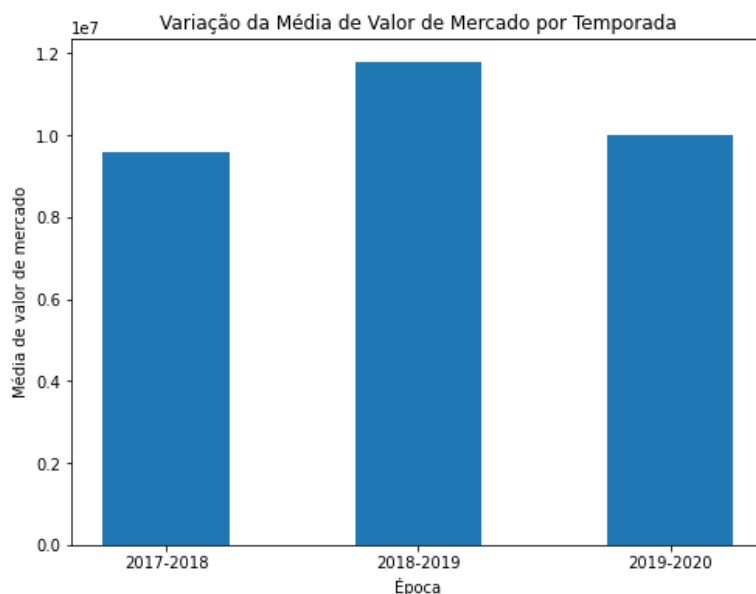


Figura 3.2: Média de valor de mercado por temporada

Através deste gráfico podemos observar a média do valor de mercado em cada época. Ora, a época em que os jogadores apresentavam o maior valor médio de mercado foi em 2018/2019, seguindo-se de 2019/2020 e, por fim, 2017/2018. De notar que estas duas últimas se encontram praticamente iguais.

3.3 Gráficos tipo Pie - Distribuição do valor de mercado por temporada

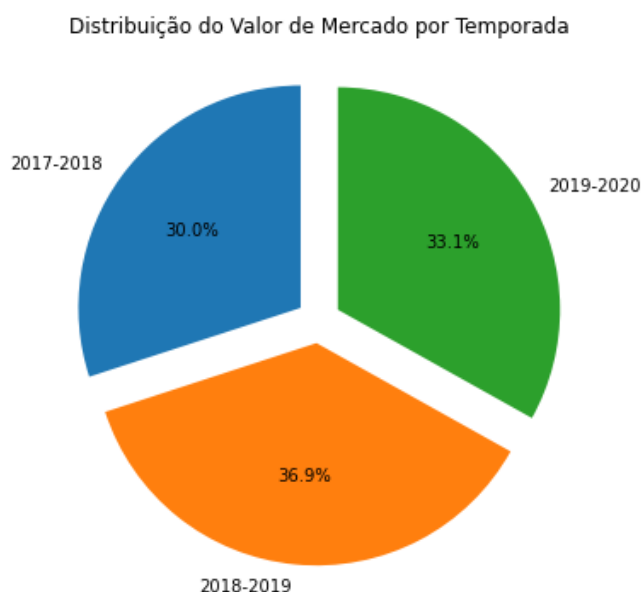


Figura 3.3: Distribuição do valor de mercado por temporada

Este gráfico tipo pie complementa o gráfico anterior, visto que mostra a distribuição do valor de mercado por temporada. Assim sendo, mantém-se a mesma ordem 2018/2019 (36,9%), 2019/2020 (33,1%), 2017/2018 (30,0%),

3.4 Histogramas - Valores de Transferências

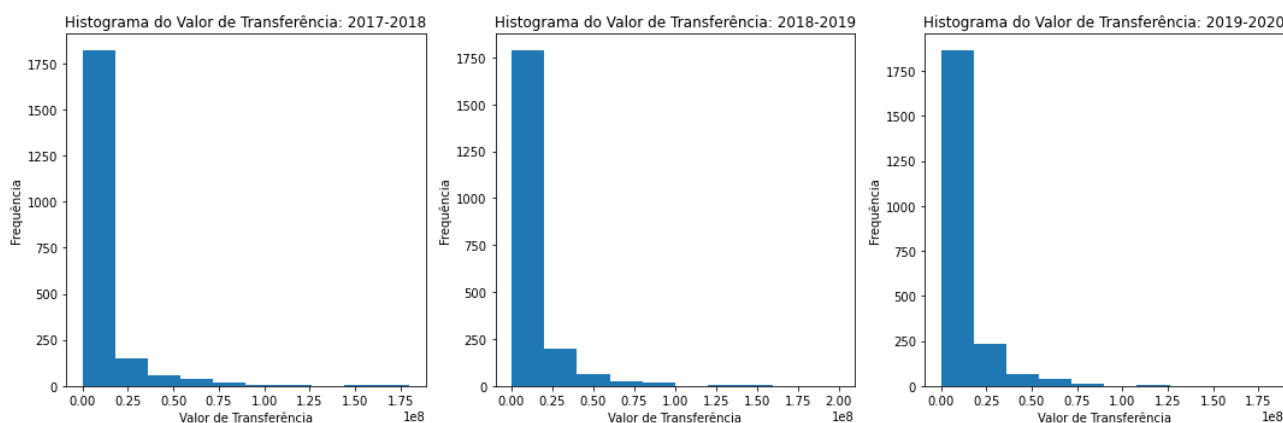


Figura 3.4: Valores de Transferências

Passando agora para os histogramas, podemos observar a distribuição do valor de transferências. Com efeito, todas as épocas apresentam uma distribuição idêntica. Inicialmente, todas apresentam um elevado número de valores de transferências compreendidos entre 0 e 20 milhões (valor de frequência superior a 1750 em todas). De seguida observa-se uma descida considerável e gradual ao longo dos intervalos em todos os gráficos.

3.5 Gráfico de Barras - Top 10 países com maior número de atletas

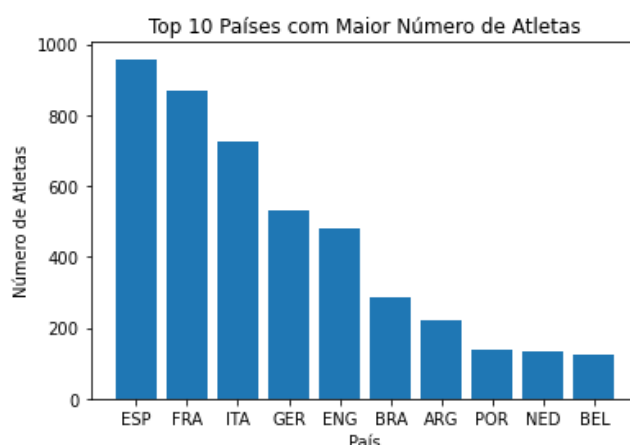


Figura 3.5: Top 10 países com maior número de atletas

Neste gráfico de barra conseguimos perceber qual é o top10 de nacionalidades com mais jogadores. Ora, a ordem observada é a seguinte: Espanha (950), França (perto dos 900), Itália (700), Alemanha (500), Inglaterra (450), Brasil (250), Argentina (220), Portugal (150), Holanda (perto dos 150) e Bélgica.

3.6 Gráfico de Barras - Valores Médios de Mercado dos Jogadores por Posição

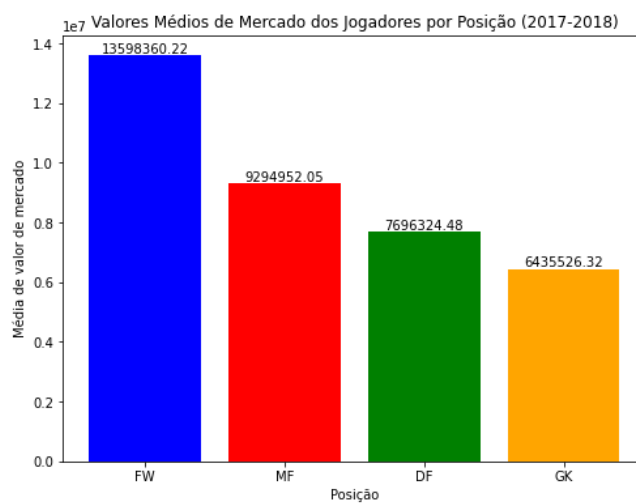


Figura 3.6: Valores Médios de Mercado dos Jogadores por Posição (2017-2018)

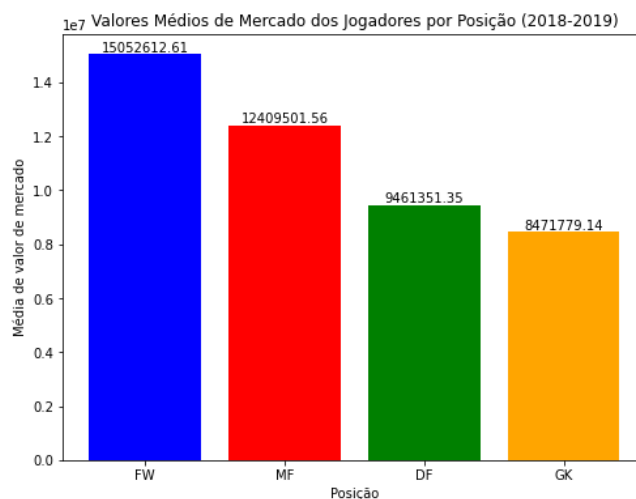


Figura 3.7: Valores Médios de Mercado dos Jogadores por Posição (2018-2019)

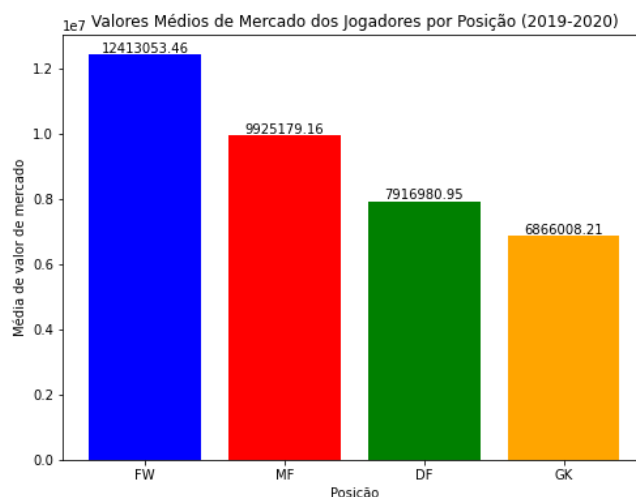


Figura 3.8: Valores Médios de Mercado dos Jogadores por Posição (2019-2020)

Nestes gráficos podemos observar as diferenças dos valores de mercado dos jogadores por posição e, em todas as temporadas, verifica-se que os avançados são sempre os mais caros seguindo-se dos médios, defesas e, por fim, os guarda-redes.

3.7 Gráfico de Barras - Valores Médios de Mercado dos Jogadores por Liga

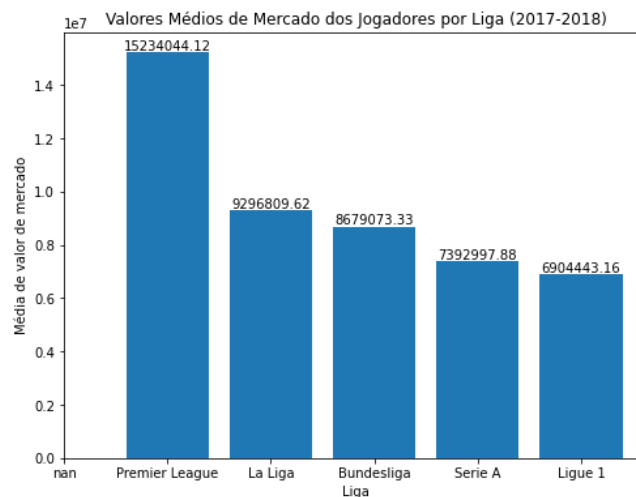


Figura 3.9: Valores Médios de Mercado dos Jogadores por Liga (2017-2018)

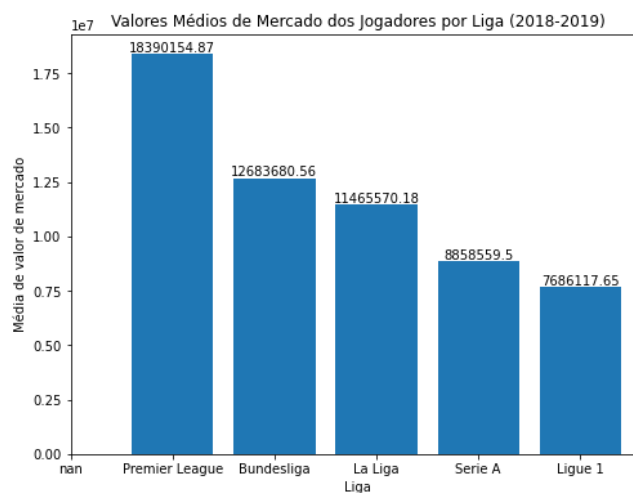


Figura 3.10: Valores Médios de Mercado dos Jogadores por Liga (2018-2019)

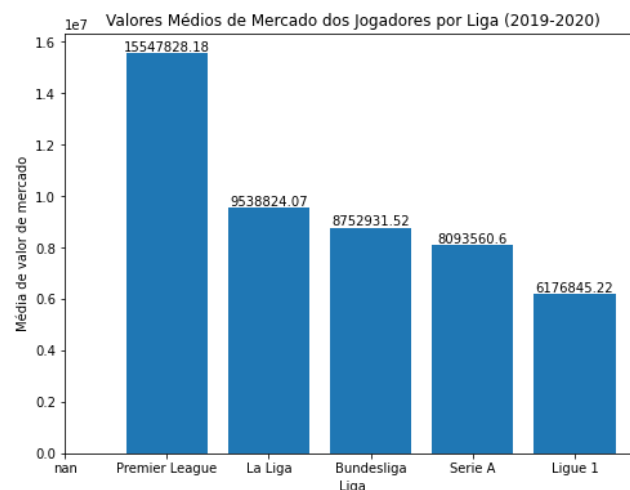


Figura 3.11: Valores Médios de Mercado dos Jogadores por Liga (2019-2020)

Por fim, nestes últimos gráficos conseguimos observar as diferenças de valor médio de mercado por liga. Assim sendo, em 2017/2018 a ordem foi Premier League, La Liga, Bundesliga, Serie A e Ligue 1; em 2018/2019 a liga alemã ultrapassou a espanhola e, portanto a ordem foi Premier League, Bundesliga, La Liga, Serie A e Ligue 1; por fim em 2019/2020 a liga espanhola retomou a 2ª posição e a bundesliga regressou ao 3º lugar: Premier League, La Liga, Bundesliga, Serie A e Ligue 1.

4. Processamento de Dados

Nesta secção iremos abordar qual o pré-processamento utilizado nos diferentes datasets utilizados. Este tratamento assim como a visualização dos dados, encontram-se no notebook `tratamento.ipynb`. Neste notebook tratamos do pré processamento individualmente de cada dataset, assim como o processo de merge dos 4.

4.1 Tratamento de Missing Values

Primeiramente foi verificada a existência de missing values nos nossos datasets. Como não existem, não foi preciso lidar com os mesmos.

4.2 Feature Engeneering

Com o auxilio do dataset criado manualmente pelo grupo com a informação da classificação das equipas, criámos uma coluna adicional aos nossos datasets. Esta coluna apresenta a classificação da equipa na devida época. Feito isto retiramos a coluna `team` do dataset original. Este processo foi realizado através da chamada da função `merge` utilizando como `key`, a época.

4.3 Feature Selection

Uma das partes fundamentais do tratamento de dados, foi a seleção de features. Para tal, retiramos inicialmente algumas colunas categóricas que não ajudam na previsão do valor de mercado do jogador como, época e `attendance`. Também calculamos a correlação das nossas colunas numéricas com a nossa target (coluna `value` - valor de mercado do jogador e coluna `bin` no caso do problema de classificação). Após este cálculo, retiramos todas as colunas que continham correlação abaixo de 0,2. Sabemos que a correlação está entre -1 a 1, logo ao retirar estas colunas apenas deixaríamos apenas as features que ajudem a prever a nossa target.

4.4 One Hot Encoding

Para as colunas como `'nationality'`, `'position'`, `'position2'`, `'foot'`, `'league'` realizamos one hot encoding de forma a tornar estas variáveis categóricas em variáveis numéricas e assim treinar os diferentes modelos.

Este processo é feito antes da modelação nos respetivos notebooks.

4.5 Binning - Problema de Classificação

De forma a converter o nosso problema de regressão num problema de classificação, foi necessário criar uma nova coluna chamada `bins` em que coluna o valor de mercado dos vários jogadores em intervalos de valores. Ou seja, em vez de prever qual o valor exato do jogador, prevemos a classe `bin` que determina a que intervalo de valores o jogador se encontra. Para isto utilizamos 2 técnicas:

- Criação Manual dos Bins (Primeira abordagem) - Inicialmente começamos por determinar os intervalos de valor de mercado para as diferentes classes. Os intervalos foram 0-5

milhões, 5-10 milhões, 10-15 milhões, 15-20 milhões, 20-25 milhões, 25-50 milhões, 50-75 milhões, 75-100 milhões e mais de 100 milhões. Desta forma foram criadas 9 classes. Os valores escolhidos para os intervalos foram conforme a análise das boxplots.

- Equal Height Binning - A forma que acabamos por utilizar depois na modelação foi a equal-height binning que divide os valores em intervalos com igual altura. Para esta técnica podemos definir quantos intervalos desejamos. O grupo testou com vários valores e decidiu que o mais indicado é 4. Para além disso, também produz os melhores resultados na modelação.

5. Modelação de Dados

Para a fase de modelação de dados seguimos duas abordagens. A primeira consiste no problema de regressão de previsão do valor de mercado dos diferentes jogadores. Na segunda, iremos tentar prever a que intervalo de valores se encontram os diferentes jogadores. Isto é, tratar-se-á de um problema de classificação.

Para cada um deles, iremos abordar os algoritmos utilizados e expor seus resultados. Os dados derivam do dataset que contém o merge das diferentes épocas. Para o treino dos modelos foram utilizadas duas abordagens. A primeira consiste no uso da primitiva `train_test_split` (com test size de 20%) a segunda foi através da técnica `k folds cross validation` (com $k = 10$). Utilizámos a segunda, seguindo a sugestão dos professores, de forma a melhor os valores retirados pelos vários modelos.

Como métricas para os algoritmos de regressão calculamos o MSE, MAE e R-Square. Nos algoritmos de classificação utilizámos a Accuracy Score.

Para além disso, nos modelos de regressão apresentámos um gráfico de barras de forma a visualizar os valores previstos e os reais. Desta forma conseguimos analisar melhor suas performances.

5.1 Problema de Regressão

O código responsável pela modelação do problema de regressão encontra-se no `regressao.ipynb`. Após o tratamento de dados referido anteriormente, foram então implementados os seguintes modelos de machine learning:

5.1.1 Redes Neurais

Como é sabido, redes neuronais são composta por camadas de neurónios artificiais conectados, que realizam cálculos e transformações nos dados de entrada para produzir uma saída desejada. A rede neuronal foi criada utilizando a classe `sequential` do TensorFlow.

Resultados - Train Test Split

Calculamos o erro médio quadrático (MSE), o erro médio absoluto (MAE) e o coeficiente de determinação (R-squared) de forma a medir o desempenho do modelo.

Mean Squared Error: 127778847561652.33

Mean Absolute Error: 6461840.805095992

R-squared: 0.6411506538703149

Análise

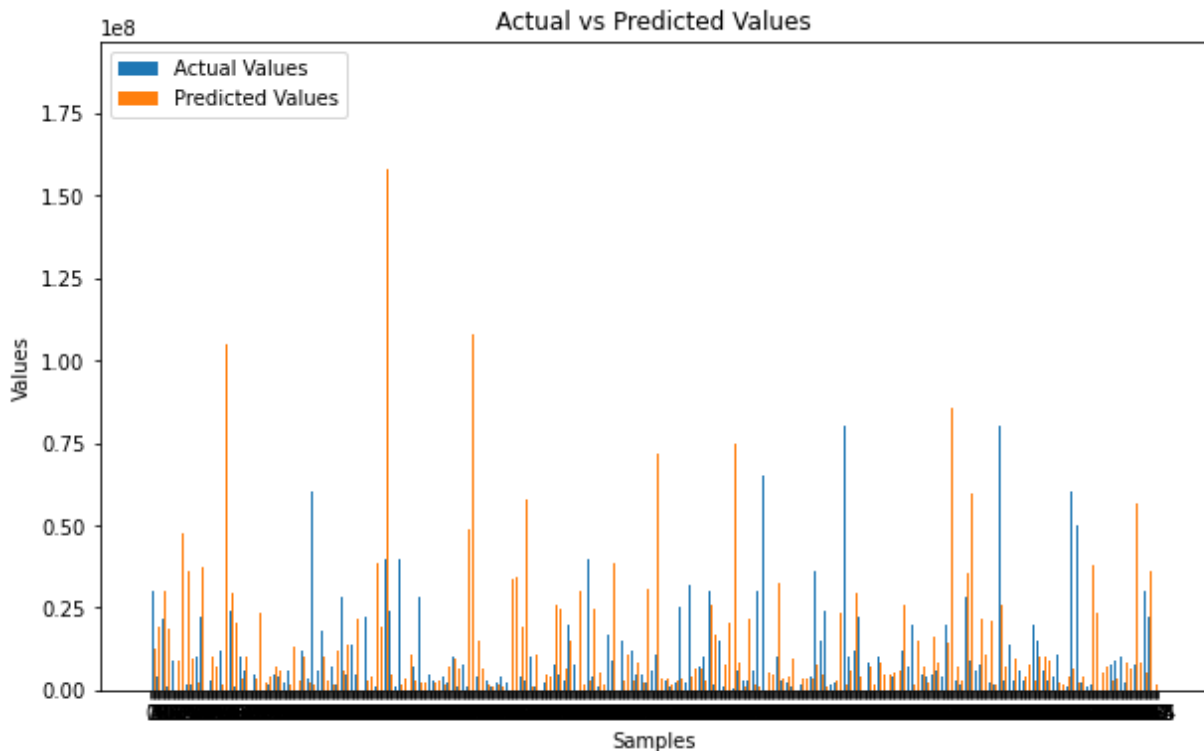


Figura 5.1: Gráfico de barras

Resultados - K Folds Cross Validation

Os valores calculados para esta abordagem foram a média do R-squared dos vários folds e o desvio padrão do mesmo.

Mean R-squared: 0.18596777428930736

Std R-squared: 0.039648630389750576

5.1.2 Random Forest

Este modelo combina várias árvores de decisão individuais para formar um modelo robusto de previsão. Neste algoritmo, utilizamos a classe RandomForestRegressor do scikit-learn para criar um modelo de regressão com 200 estimadores, limitando a profundidade máxima das árvores em 10.

Resultados - Train Test Split

Calculamos o erro médio quadrático (MSE), o erro médio absoluto (MAE) e o coeficiente de determinação (R-squared) de forma a medir o desempenho do modelo.

R-squared: 0.6487169461168014

Random Forest Mean Squared Error: 125084647017613.1

Random Forest Mean Absolute Error: 6047377.712994144

Análise

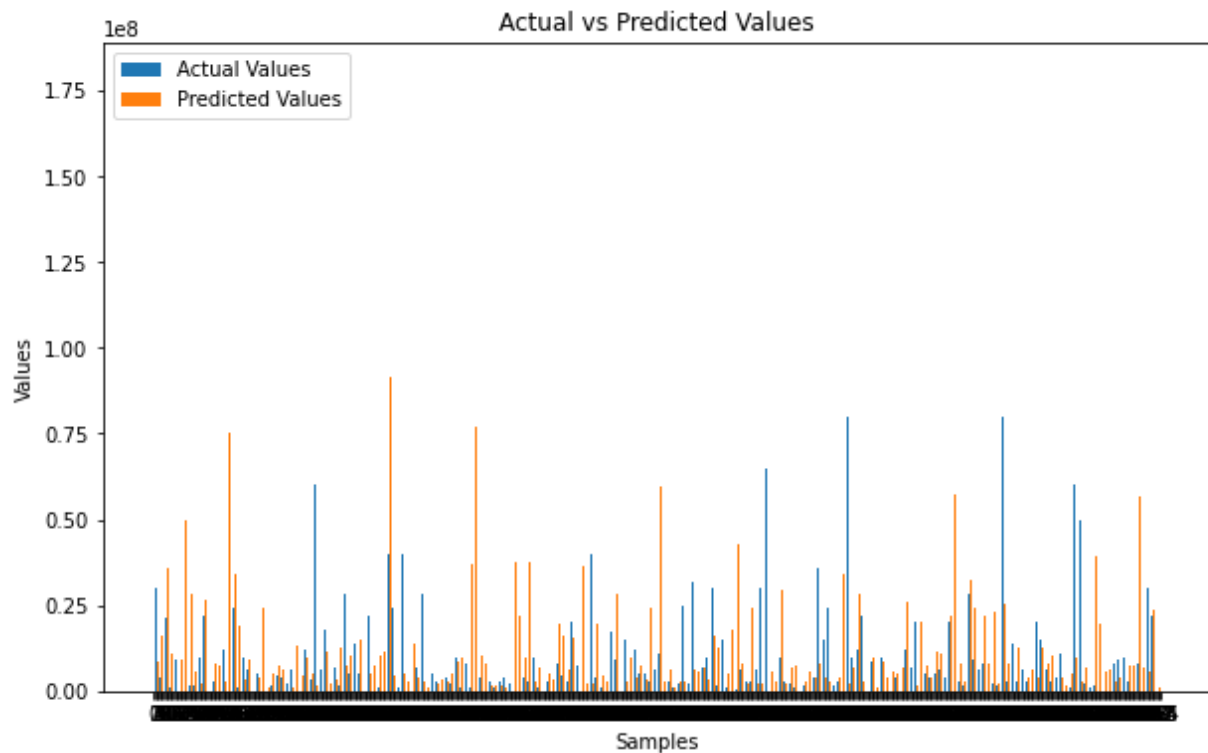


Figura 5.2: Gráfico de barras

Resultados - K Folds Cross Validation

Mean R-squared: 0.6868053641512701

Std R-squared: 0.04449661760595824

5.1.3 XGBoost

Neste algoritmo, utilizamos a biblioteca XGBoost para treinar o modelo de regressão. Para além disso, foi feita a seleção das melhores features para o treino do modelo utilizando a primitiva `feature_importances_`.

Resultados

Calculamos o erro médio quadrático (MSE) e o coeficiente de determinação (R-squared) de forma a medir o desempenho do modelo.

R-squared with Feature Selection: 0.6679944903098001

XGBoost with Feature Selection Mean Squared Error: 118220311308582.75

Análise

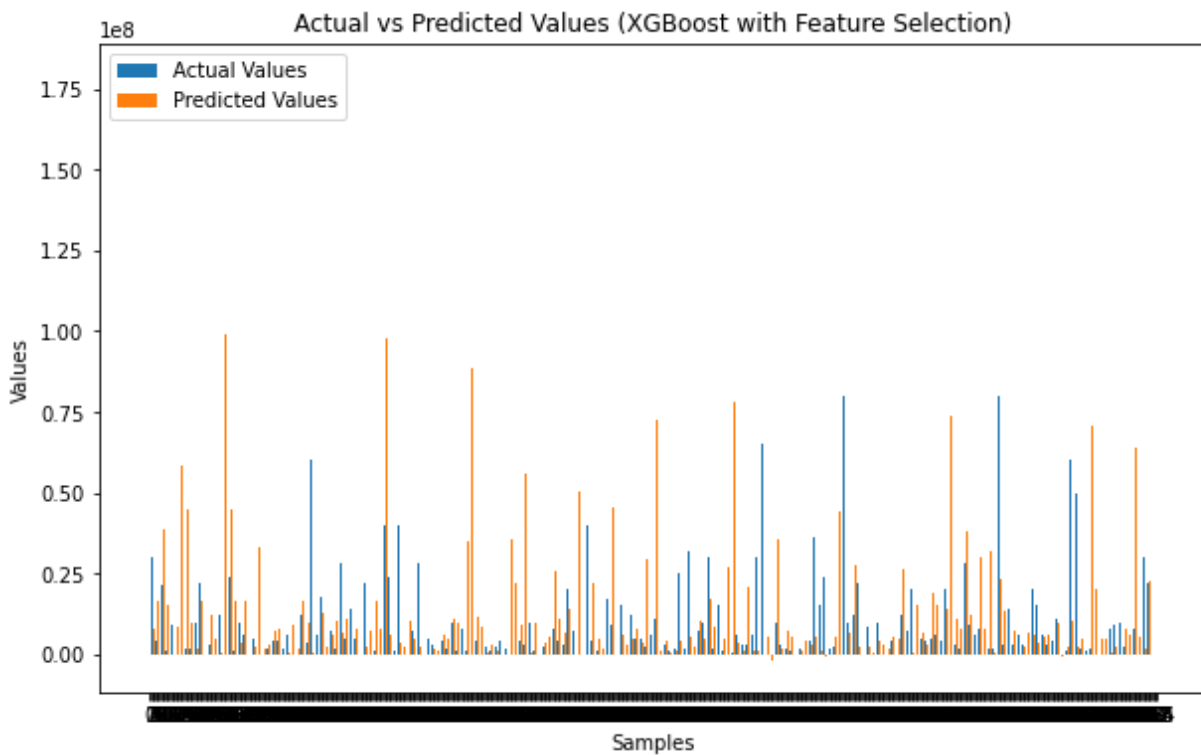


Figura 5.3: Gráfico de barras

5.1.4 LightGBM

Para este algoritmo, estamos a utilizar a biblioteca LightGBM para treinar um modelo de regressão LightGBM. Para além disso, foi feita a seleção das melhores features para o treino do modelo utilizando a primitiva `feature_importances_`.

Resultados

R-squared with Feature Selection: 0.7336260070922958

Análise

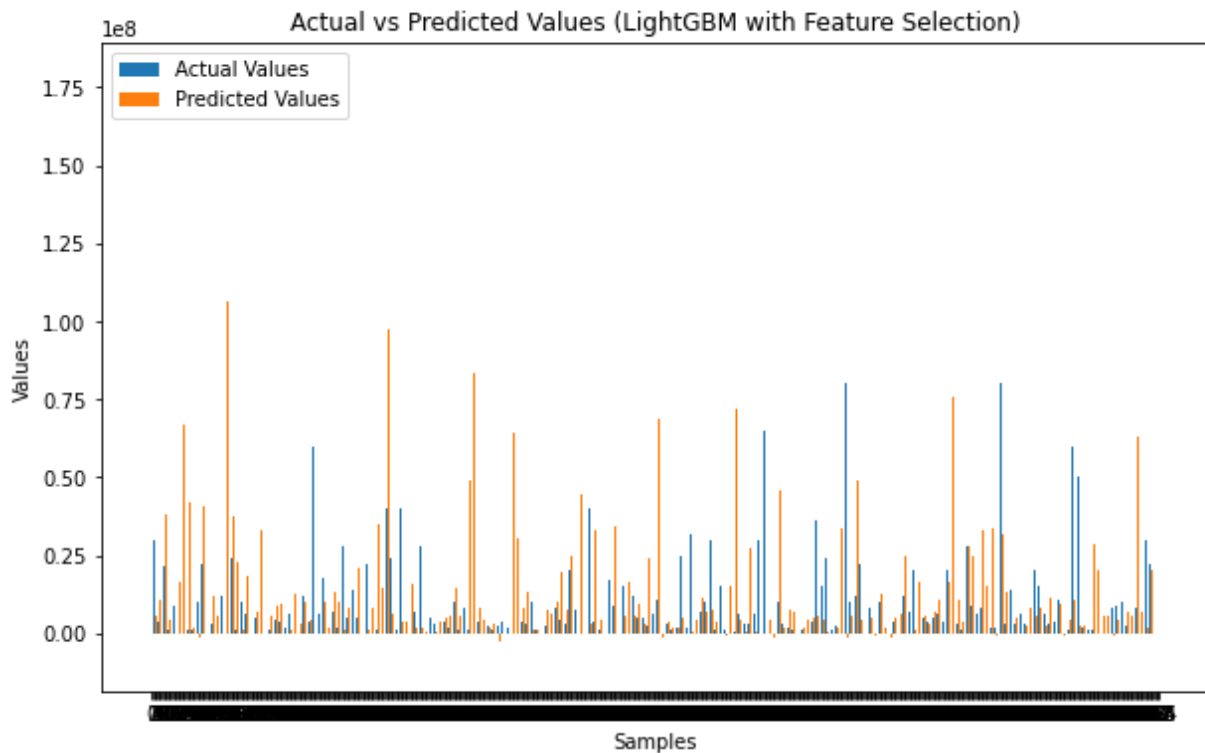


Figura 5.4: Gráfico de barras

5.2 Problema de Classificação

De seguida, iremos abordar o problema de classificação que consiste na previsão dos intervalos de valores de mercado dos jogadores. Foram então criados 4 bins que contêm os seguintes valores:

- Bin 1: 100.0 - 1500000.0
- Bin 2: 1500000.0 - 4500000.0
- Bin 3: 4500000.0 - 12000000.0
- Bin 4: 12000000.0 - 180000000.0

Com 4 bins produzimos os melhores resultados para os modelos de classificação seguintes:

5.2.1 Regressão Logística

Resultados - Train Test Split

Accuracy: 57.82%

Análise

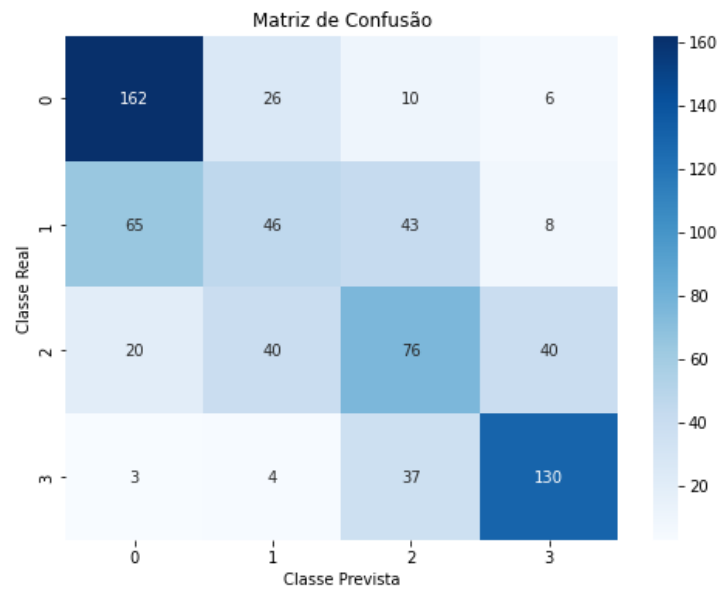


Figura 5.5: Matriz de Confusão

Resultados - K Folds Cross Validation

Os resultados apresentados são a média dos valores obtidos nos diversos folds. Accuracy e o desvio padrão.

Accuracy: 48.19898909284384

Standard Deviation: 0.019351261304396253

5.2.2 Decision Tree

Resultados - Train Test Split

accuracy: 48.88%

Análise

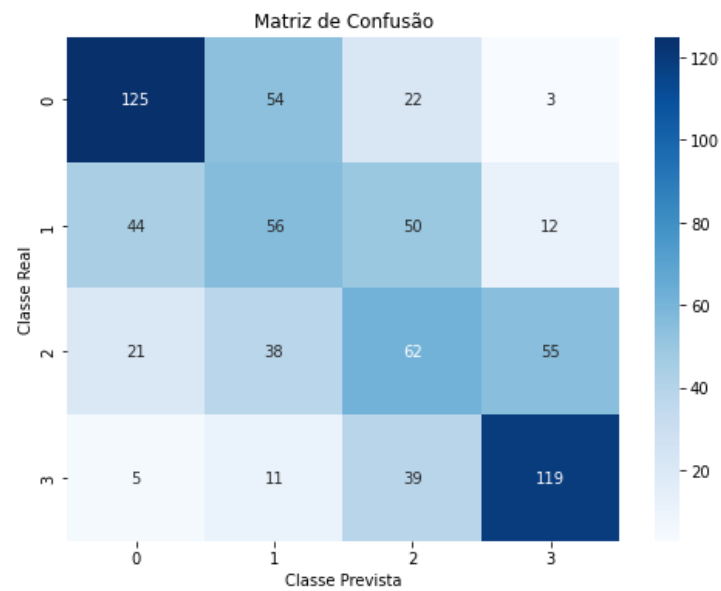


Figura 5.6: Matriz de Confusão

Resultados - K Folds Cross Validation

Decision Tree - Accuracy: 51.300095457177285

Decision Tree - Standard Deviation: 0.018653882533969147

5.2.3 Random Forest

Resultados - Train Test Split

accuracy: 57.12%

Análise

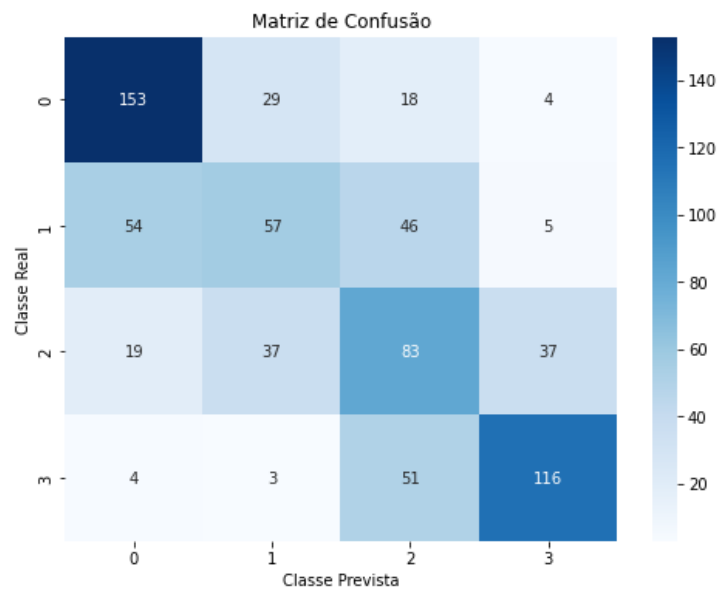


Figura 5.7: Matriz de Confusão

Resultados - K Folds Cross Validation

Random Forest - Accuracy: 57.72694552681408

Random Forest - Standard Deviation: 0.026192117727983376

5.2.4 Support Vector Machine

Resultados - Train Test Split

accuracy: 46.09%

Análise

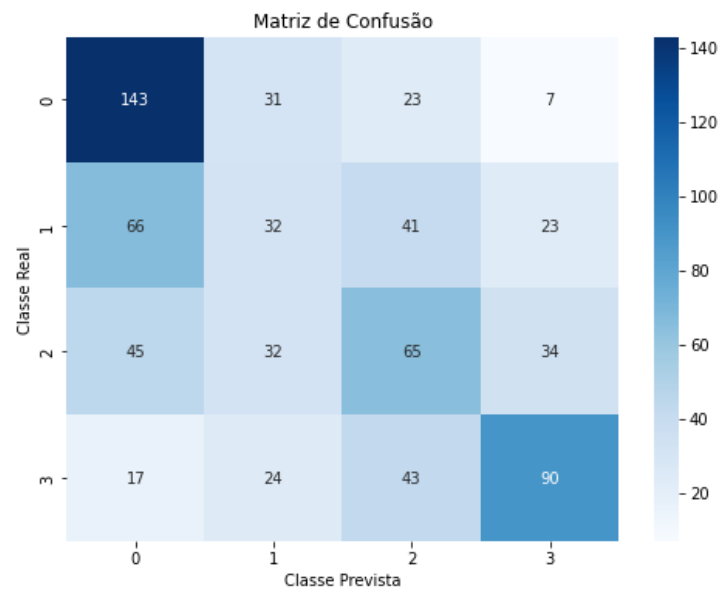


Figura 5.8: Matriz de Confusão

Resultados - K Folds Cross Validation

Support Vector Machine - Accuracy: 44.84507769588282

Support Vector Machine - Standard Deviation: 0.012679431268979096

5.2.5 XGBoost

Resultados - Train Test Split

accuracy: 64.11%

Análise

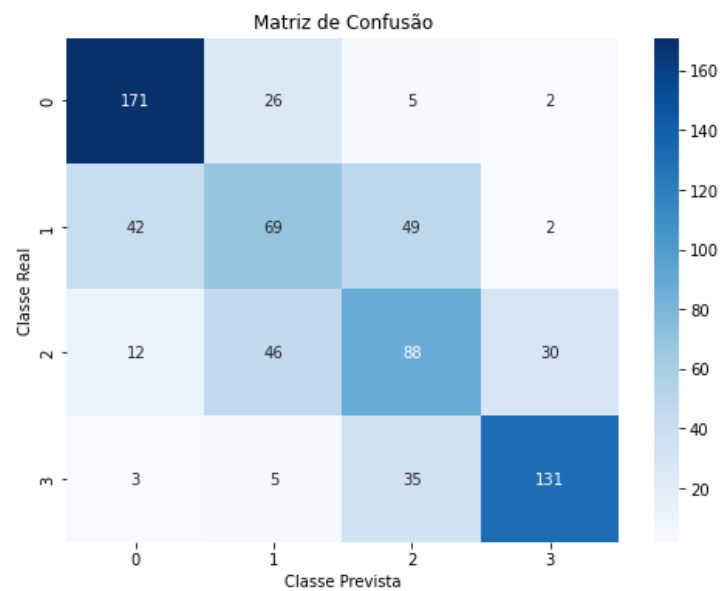


Figura 5.9: Matriz de Confusão

Resultados - K Folds Cross Validation

XGBoost - Accuracy: 62.58876735051563

XGBoost - Standard Deviation: 0.03380715674740031

5.2.6 LightGBM

Resultados - Train Test Split

accuracy: 64.11%

Análise

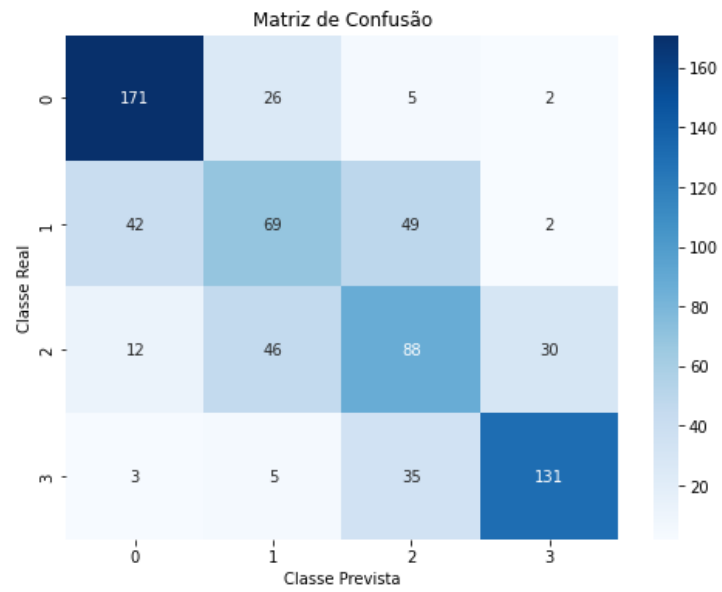


Figura 5.10: Matriz de Confusão

Resultados - K Folds Cross Validation

LightGBM - Accuracy: 62.00186219739293

LightGBM - Standard Deviation: 0.031013973734344762

6. Resultados e Conclusão

Analisando os vários modelos e começando pelo problema de regressão, verificámos que o modelo que obteve melhor performance foi o LightGBM (R-Squared maior). É de notar que os valores da métricas como MSE e MAE encontram-se bastante elevados. No entanto, o valor de mercado dos jogadores é na casa dos milhões e por esta razão os valores apresentados são razoáveis.

Por outro lado, para o problema de classificação o modelo que obteve melhor accuracy foi o XGBoost, com 64%. Apesar deste valor baixo conseguimos observar através da sua matriz de confusão que os valores encontram-se perto da diagonal, sendo por isso um desvio pequeno na previsão dos valores.

Verifica-se também que para valores de performance relativamente alto dos jogadores, o valor de mercado dos mesmos oscila bastante. Daí a dificuldade dos modelos preverem valores mais baixos de valor de mercado (bins 1 e 2 no problema de classificação e valores entre 5 e 25 milhões no problema de regressão). Isto pode-se observar através das várias matrizes de confusão apresentadas e pelos gráficos de barras.

É importante de referir que o grupo tentou outra abordagem na divisão de dados de treino e de teste, utilizando épocas diferentes para treino e uma outra para teste. No entanto, não o conseguimos implementar corretamente e dava valores negativos no cálculo das diferentes métricas. (Encontra-se documentado no código)

Em relação ao desenvolvimento do projeto em si, consideramos que realizamos um bom trabalho. Com a realização do mesmo conseguimos entender que o caso de estudo do nosso trabalho prático é bastante difícil de prever uma vez que existem diversas variáveis que influenciam o valor de mercado do jogador.

Para além disso, as fontes de dados foram alteradas a meio do projeto o que fez com que uma grande parte do desenvolvimento do projeto fosse em vão. Apesar disto e através da análise feita anteriormente, conseguimos um valor razoável de precisão nos dois problemas apresentados.

Posto isto podemos afirmar e concluir, que existem outras métricas que influenciam o valor de mercado do jogador do que propriamente as suas estatísticas dentro do campo. Um exemplo seria, o impacto do jogador nas diferentes redes sociais, eventos exclusivos, projetos de fora, aspetos da vida pessoal, etc.

7. Trabalho Futuro

Como trabalho futuro pretendemos aumentar ainda mais a quantidade e diversidade dos dados. Embora o projeto atual utilize dados públicos disponíveis, é possível expandir a fonte de dados para incluir informações mais abrangentes sobre os jogadores, como por exemplo: dados sobre lesões, histórico de transferências, entre outros. Podemos também usar web scraping para obter novos dados de forma mais eficiente. Quanto mais dados relevantes forem considerados, maior será a precisão das previsões.

Também pretendemos incorporar análise de sentimento das redes sociais. Como mencionado na secção de trabalhos relacionados, as redes sociais têm desempenhado um papel importante na determinação do valor de mercado dos jogadores de futebol. Portanto, seria interessante incorporar esta análise. Teremos de realizar extração de informações relevantes de redes sociais (Twitter e Facebook, por exemplo) e analisar o sentimento expresso pelos fãs e especialistas em relação aos jogadores. Todos estes dados podem fornecer informações interessantes sobre a popularidade e a percepção dos jogadores, que podem ser incorporados ao modelo de previsão.