DOI: 10.5748/16CONTECSI/DSC-6151

Aplicação de Algoritmos de Machine Learning para Classificação Automática de Problemas Futebolístico.*

Adão Baptista Pereira Lopes

Departamento de Informática, Escola de Ciências e Tecnologia,
Universidade de Évora, Rua Romão Ramalho, 59, 7000-671 Évora, Portugal
abpl@uevora.pt

Resumo Atualmente, as empresas ou organizações, inclusive as da área do desporto, evidenciam uma grande demanda pela análise de dados. Existe um vasto interesse na obtenção da previsão de resultados das várias modalidades do desporto, principalmente do futebol, onde podemos encontrar várias tipologias de previsão pretendidas. Desde a previsão exata de resultados, à previsão de número de golos, passando pela previsão se ambas as equipas marcam golos ou não, o objetivo é o de alcançar a previsão correta. É exatamente sobre a vertente "ambas as equipas marcam golo ou não" que incide este projeto, procurando através da aplicação de várias técnicas de Machine Learning (ML), promover o auxílio da previsão, permitindo verificar se ambas as equipas marcam golo ou não. Sendo certo, que a área da tecnologia tem evoluído a um ritmo galopante e constante, verifica-se que há um crescimento volumoso dos dados no desporto, pelo que o projeto pretende analisar os dados do campeonato Português (LIGA NOS) desde a época 1994/1995 até à atualidade. De igual modo, visa apresentar diversos algoritmos de classificação como K-Nearest Neighbor (KNN), Suport Vector Machine (SVM), Decision Tree (DT), Linear Regression (LR)), e demonstrar o que melhor se adapta neste case study, através dos métodos de avaliação de cada um dos modelos. A aferição do desempenho da previsão do problema do estudo estará baseada na Confusion matrix, em micro-average, macro-average e F1 Score.

Keywords: Machine Learning· Classificação Automática · Futebol · k-Nearest Neighbor.

1 Introdução

No Desporto, o futebol é uma das modalidades mais apaixonantes, que granjeia milhares de adeptos em todo o mundo. Um dos motivos associados a este manifesto interesse coletivo, prende-se com o facto de ser um jogo imprevisível, com inúmeras variáveis que pode decidir um jogo, independentemente do poder

* Este trabalho foi desenvolvido no âmbito da Disciplina (Classificação Automática e Métodos de Núcleo

económico das equipas. Um dos factos recentes que comprova isso mesmo, é o ocorrido na Época 2015/2016 onde contra todas as possibilidades ou previsões existentes, o Leicester City, equipa que havia sido recém promovida na Premier League, foi a vencedora do campeonato inglês, onde, é sobejamente conhecida a existência de vários gigantes do futebol europeu e mundial.

Ultimamente, há vários projetos[12][26][22] que recorrem ao auxílio de técnicas de ML, para encontrar padrões que possam ajudar a descobrir informações peculiares, através da análise de dados para previsão de resultados do Jogo. Neste artigo, procura-se utilizar várias técnicas de Machine Learning, para implementação de um modelo de classificação num jogo de futebol do campeonato Português (Liga NOS). A questão central deste estudo, incide na aplicação desse modelo de classificação para prever a possibilidade de ambas as equipas marcarem golos.

2 Estado da arte

O processo de aprendizagem há muito tempo que fascina psicólogos, filósofos, e cientistas de todas as áreas do conhecimento, o que dificulta concretizar uma definição precisa desse conceito. Na área computacional, a grande motivação é colocar o computador num nível tal que consiga pensar como um cérebro humano [20]. A Inteligência Artificial (IA) surge assim, como uma área de estudo computacional, que tem como objetivo construir dispositivos e programas que simulem a capacidade humana de raciocinar, tomar decisões prévias e resolver certos problemas[29].

O campo de estudo do ML consiste em terem-se programas computacionais que conseguem imitar o comportamento de aprendizagem humana[20]. ML pode assim ser entendido como uma sub-área da IA, que tem como finalidade estudar sistemas, e reter o máximo de aprendizagem possível, ou aprender com os dados[7].

Trabalhos Relacionados Na atualidade existem vários projetos e pesquisas no âmbito de utilização da IA, propriamente ML, para prever dados de jogos de futebol. Há uma vasta literatura a abranger esse problema, sendo que na maior parte das literaturas encontradas, as estratégias para previsão baseiam-se em modelos de RL, KNN e SMV[10].

Foi realizada uma pesquisa, através do modelo de regressão logística, prevendo os resultados dos jogos da Premier League do Barclays de 2015/2016, tendo-se concluído que existem várias variáveis significativas na previsão do resultado, com destaque para o poder defensivo tanto da equipa visitante como da equipa visitada[22]. Escolhendo somente as variáveis mais significativas, pode-se aumentar a precisão da previsão em 18%.

O FIFA 2015 e o Campeonato Europeu, serviram de base de conhecimento, para a pesquisa sobre a previsão de resultados de futebol[26]. Nesse projeto, recolheram-se dados reais, bem como dados virtuais dos jogadores a nível físico (Aceleração, Força, Velocidade, etc.), além de dados relacionados com a técnica

futebolística (Dribles, Precisão de Passe, etc.), dos jogadores no jogo da Play Station (FIFA 2015). A recolha de dados virtuais foi justificada como forma de economizar tempo, e para permitir a comparação com os dados reais. Desta forma, robusteceram o conjunto de dados para aumentar o nível de precisão do algoritmo[26]. os algoritmos aplicado neste projeto são: Regressão Linear, SVM, RBF, e regressão Logística.

A Regressão Logística aplicada num conjunto de dados da Premier League da temporada 2014-2015, avaliou 9 características (Casa e Visitante, Golos Short, Odds, Força de ataque, Índice de desempenho de jogadores, Índice do desempenho dos diretores, e vitórias das equipas), gerando uma precisão de cerca de 95%[12], portanto bastante elevada. É importante realçar que mesmo sistema de alta precisão para previsão de dados futebolísticos, não é confiável, derivado ao facto, que um jogo de futebol depende de muitos fatores imperdíveis.

No entanto, nenhum dos projetos debruça o problema da previsão se ambas as equipas marcam golo ou não. Em contrapartida, a maioria dos projetos desta natureza, preocupam-se mais com a previsão do resultado exato do jogo[25], mas outros também abordam o estudo das diferenças dos golos num jogo[18][28][14]. Os projetos desta natureza são os que mais se assemelham com a questão cerne deste estudo.

Machine Learning (ML) pode ser entendido como o estudo de algoritmos, com capacidade de melhorar a sua performance numa situação baseada em experiência predecessora. Esta área está fortemente relacionada com o reconhecimento de padrões e estatística. A ML tem como finalidade desenvolver técnicas computacionais que investiga simulação do processo de aprendizagem humano, e construir sistemas capacitados para aquisição de conhecimento automático[6].

Comummente, os algoritmos de ML lidam com experiências precedentes. A universidade de Califórnia criou um centro de repositório[1], com conjuntos de dados de diversas naturezas disponíveis. Há muitos outros repositórios de dados disponíveis, referindo-se a título de exemplo o KDnuggets[2]. Antes de prever ou classificar algo, é necessário treinar previamente o algoritmo. E para isso é necessário escolher as características mais relevantes para o caso em estudo. Selecionar as características com maior relevância, e eliminar as irrelevantes é central para melhorar o ML[16].

Algoritmos Supervisionadas e Não Supervisionadas Existem dois métodos de aprendizagem comummente utilizados em técnicas de ML: (I) através dos algoritmos supervisionados, em que as entradas e saídas no conjunto da aprendizagem de dados é conhecida. No decorrer da aprendizagem, o modelo afina as suas variáveis, para assim poder mapear as entradas e saídas correspondentes[19]. O ponto crucial deste algoritmo é a capacidade de fazer previsões com um alto nível de desempenho; é construir um classificador que, de modo correto, possa

[1] *http* : *//archive.ics.uci.edu/ml/datasets.html*
[2] *https* : *//www.kdnuggets.com/datasets/index.html*

constituir uma classe para novos exemplos. Nos (II) algoritmos não supervisionados, o programa analisa os dados, verifica os que podem ser agrupados. Não há resultado alvo[19].

3 Algoritmos de Classificação

Para o desenvolvimento deste estudo, o framework utilizado é o Scikit-Learn (Python)[3]. É um módulo de Python que integra uma imensa gama de algoritmos de ML tanto para problemas supervisionados como não supervisionados. Este pacote concentra-se em trazer o ML para especialistas, utilizando uma linguagem de alto nível (Python). Proporciona facilidade na usabilidade, um alto desempenho, tem muita documentação oficial e é consistente[21]. Elencam-se de seguida os algoritimos/técnicas de classificação:

Suport Vector Machine (SVM) Éconhecida por ser uma técnica de classificação muito poderosa, principalmente para dados com dimensões volumosas[3]. A ideia base pode ser salientada geometricamente. Se os dados estão em um espaço, o algoritmo encontra o hiperplano que separa os dados com a maior margem exequível. Com esse hiperplano, é presumível classificar os dados[8].

Por outras palavras, o algoritmo separa os pontos de dados utilizando uma linha se for duas dimensões, no caso for três dimensões utiliza um hiperplano. Esta linha é escolhida de tal forma que será mais importante dos pontos de dados mais próximo em duas categorias. SVM é uma técnica de classificação, que procura encontrar um modelo onde a separação entre as classes tenha a maior margem possível.

As SVM foram originalmente projetadas para classificação binária[2]. Esta técnica é aplicada para classificação de problemas de diversas naturezas, como por exemplo Economia[11], Desporto (Futebol)[1], Medicina[8][9]„ onde é de grande utilidade para detetar os padrões de várias doenças, principalmente o cancro.

Decision Tree (DT) A estrutura de dados é composta por um conjunto de elementos, que armazena informações em nós, que são designadas de Árvores em informática. Ela possui um nó principal que geralmente é denominado por root(raiz).

$$Entropia = \sum_{i=1}^{k} P_i log_2 P_i \tag{1}$$

A nível hierárquico é o maior nó, e as ligações a partir do nó raiz, são designadas de filhos. Estes nós filhos, podem ter os seus próprios filhos, e assim sucessivamente. Se o nó não tem nenhum filho é denominado como nó folha ou terminal.

Sabendo essas definições, torna-se mais fácil compreender o que é uma DT. É uma árvore que tem regra nos seus nós, e o processo decisório é representado

[3] $http://scikit-learn.sourceforge.net.$

pela folha da árvore[30]. Sumariamente, numa DT a tomada de uma decisão é um caminho percorrido a partir do nó raiz até um nó folha.

Geralmente, é um dos algoritmos mais usados para solucionar problemas de classificação. A categorização é feita usando algumas técnicas. A Formula 2[27] apresenta a técnica de Gini, e a Formula 1[27] apresenta a Entropia. Para o problema deste estudo utilizaram-se essas duas categorias. Obteve-se o mesmo resultado, tanto na precisão do modelo, como no confusion matrix, e nas outras formas de avaliação do modelo.

$$Gini = 1 - \sum_{i=1}^{k} P_i^2 \tag{2}$$

Linear Regression (LN) A regressão linear é um outro algoritmo para classificação, que tenta encontrar uma linha reta que atravessa um gráfico disperso de pontos. Esta reta vai estar o mais próximo possível de todos os pontos, ou seja, encontrar a melhor linha. Essa linha é chamada de linha de regressão [23].

k-Nearest Neighbors (KNN) Esta técnica é um método para classificação que se baseia nos exemplos de aprendizagem mais próximos na dimensão dos atributos. Os exemplos de treino são colocados no espaço de atributos multidimensional[5].

O parâmetro mais importante desta técnica é o K. Define a quantas unidades de distância os elementos podem estar para serem considerados vizinhos[17]. O KNN é um algoritmo de aprendizagem supervisionado. A ideia ecuménica desse algoritmo traduz-se em encontrar os k exemplos rotulados mais próximos do exemplo não classificado[6].

Resumidamente, o KNN é um algoritmo simples que prevê pontos de dados desconhecidos, baseando-se na proximidade dos seus vizinhos. O valor de k é um fator crítico, porque é dele que depende a precisão da previsão. A título de exemplo vemos que a classificação é totalmente diferente, consoante o K for 3 ou for 20. Para determinar o cálculo da distância mais próxima de um ponto, este algoritmo utiliza as funções básicas de distância de Euclides, representada na Fórmula 3, que é uma das várias possibilidades para o cálculo das distâncias.

É aconselhável, ao decidir qual o melhor valor de K, fazer uma comparação de diferentes valores K's efetuando-se vários testes para descobrir o K ideal. O valor de K a escolher, é o que tiver a maior taxa de acertos. Na fase de treinar, o algoritmo KNN exige pouco esforço. Por outro lado, o custo da operação para marcar um novo exemplo não classificado é um pouco alto. Na pior das hipóteses, esse novo exemplo tem que ser comparado com todos os dados contido no conjunto de treino[6].

O Algoritmo KNN usa a aprendizagem baseada em instâncias. Isto quer dizer, que utiliza o conjunto de dados de treino para classificar os pontos de dados desconhecidos. Apesar de KNN ser um algoritmo de classificação, é amplamente usado para prever e fazer estimativas. Tendo em conta valores históricos, os

valores ideais de K na maioria dos casos está situado no intervalo entre 3 a 10[23].

$$Euclides = \sqrt{\sum_{i=1}^{K}(x_i - y_i)^2} \tag{3}$$

Deste modo, podemos observar que não existe somente um único valor de K apropriado [6]. A escolha desse valor vai depender muito da natureza do caso de estudo. É também muito importante realçar,que para um mesmo caso de estudo, o melhor valor de K, depende muito dos números das características escolhidas, e as suas relevância para o caso em estudo.

"Utilizar valores ímpar de K é mais apropriado, utilizando-se no caso a classe maioritária, para evitar situações de empate[6]. Em alternativa, há outra abordagem para a resolução desse problema. Consiste em atribuir pesos a cada um dos k vizinhos mais próximos. Eles são ordenados em ordem crescente, para a determinação da classificação, sendo que a classe dos exemplos de maior similaridade tem um peso maior, do que as classes com pouca similaridade"[6].

Este método de classificação é aplicado em problemas de diversa natureza, como por exemplo na Medicina[15][4] para identificação dos fatores de risco no cancro da próstata, com base em variáveis clínicas e demográficas; na Agricultura[24] para previsão do clima, estimando parâmetros da água do solo, para simular precipitações e outras variáveis meteorológicas; na área Financeira[31][13] para prever o mercado de ações, que inclui a previsão da descoberta de tendências de mercado, planear estratégias de investimento, identificando os melhores ações. Ainda pode se acrescentar, taxa de câmbios, falência dos bancos, classificação de crédito, gestão de empréstimos etc.

4 Dataset

A Figura 1 infra exposta, pretende representar o conjunto de dados do campeonato português desde a temporada 1994/1995 até à temporada atualmente em curso de 2018/2019. No total contabiliza-se uma média de 300 jogos por ano, sendo que nas 21 épocas atinge-se um acumulado de cerca de 7000 jogos. No conjunto de dados desde o ano 1994 até 2000 encontramos somente 4 características(Equipa de casa (HomeTeam); Equipa visitante (AwayTeam),; Golos da equipa de casa (FTHG), e Golos da equipa visitante (FTAH))

A partir da temporada 2000/2001 até 2016/2017, há mais duas características que são os golos nos intervalos da equipa de casa e da visitante (HTHG, HTAG). Nas duas últimas temporadas, constata-se que o conjunto de dados é mais completo, englobando todas as estatísticas do jogo, como cantos, amarelos, vermelhos, faltas cometidas, remates à baliza, entre outras características. O conjunto de dados foi desenvolvido a partir do repositório de dados dísponivel em Football Betting, Scores e Results Service (FBSRS)[4].

[4] $http://www.football-data.co.uk/portugalm.php$

Div	Date	HomeTeam	AwayTeam	FTAG	FTR	HTHG	HTAG	HTR	HS	AS	Ambas Marcam
P1	06/08/17	Aves	Sp Lisbon	0	2 A	0	1	A	12	12	
P1	06/08/17	Setubal	Moreirense	1	1 D	1	0	H	6	12	
P1	07/08/17	Feirense	Tondela	1	1 D	0	1	A	12	13	
P1	07/08/17	Portimonense	Boavista	2	1 H	0	1	A	12	5	
P1	07/08/17	Rio Ave	Belenenses	1	0 H	1	0	H	11	10	
P1	08/08/17	Maritimo	Pacos Ferreira	1	0 H	0	0	D	7	4	
P1	09/08/17	Benfica	Sp Braga	3	1 H	2	1	H	15	6	

Figura 1. O Conjunto de Dados (Campeonato Português).

A partir do conjunto de dados DCP[5] obtido da FBSRS, foram criadas algumas variáveis, com o objetivo de enriquecer a base de conhecimento estatístico, de forma a permitir que a cada jornada os dados alterem dinamicamente. Inicialmente, foram descartados vários atributos que não apresentam nível de correlação com outras características relevantes para o estudo em causa. A título de exemplo podemos citar, as odds das diferentes casas de apostas que foram ignoradas, e outros atributos que não têm dados para todos os jogos, por exemplo cartão vermelho e amarelo.

Na totalidade, esta abordagem vai utilizar seis variáveis (Equipa da casa (EC), Equipa visitante (EV), Poder de ataque da equipa da casa (PAEC), Poder de ataque da equipa visitante (PAEV), Média de golos da equipa da casa em jogos em casa (MGECEC) e Média de golos da equipa visitante em jogos fora (MGEVEF)).

As duas primeiras variáveis (EC e EV), são obtidas a partir do conjunto de dados DCP. É de enorme importância, termos como fonte os dados do passado para a classificação[18]. O PAEV e PAEC, são obtidos através do Sofifa[6]. São intervalo de valores fixos de 0 a 100, que é atribuído a cada equipa, de acordo com a atual desta época 2018/2019. As equipas que não se encontra na Liga Nós, é atribuído o valor 50.

A MGECEC e MGEVEF têm um pormenor de extrema importância. Ela beneficia a equipa da casa, uma vez que normalmente, no futebol a equipa da casa tem uma estatística melhor de média de golos. Pode- se justificar este dado, pelo facto da equipa da casa ter um maior apoio dos adeptos, o que pode levar à concentração dos jogadores, à mentalidade de encarar um jogo em casa, factores que conjugados implicam que as equipas tenham um melhor resultado em casa, com exceção raríssima de alguns casos. A média de golos entre Benfica vs Porto, é diferente da média do golo entre Porto vs Benfica. Neste caso a MGECEC (Benfica), é calculada a partir de todos os jogos realizados por Benfica em casa contra o Porto, e golos marcados, onde é dividido o número total de golos pelo número total dos jogos entre eles.

[5] Significa: Dataset do Campeonato Português
[6] $https://sofifa.com/teams?lg=308\&col=sa\&sort=asc\&showCol=ta$

5 Avaliação dos Classificadores

Para saber quão bom é o modelo de classificação, é imprescindível calcular a precisão da previsão dos dados de teste. Além da precisão há outros métodos para avaliar o quão bem um modelo se comporta, utilizando o confusion matrix (MC) e classification report (CR) (Micro_Average, Macro_Average, F1-Score).

Para calcular os métodos de avaliação (Recall (4a), Acuracy (4b) e Precision (4c) dos classificadores utilizam-se as seguintes fórmulas:

$$Recall = \frac{TP}{TP + FN}, Acuracy = \frac{TP + TN}{Total}, Precision = \frac{TP}{TP + FP}, \quad (4)$$

O primeiro quadrante da MC pertence a True Positive (TP), o segundo quadrante é False Positive (FP), o terceiro quadrante False Negative (FN), e o último quadrante é True Negative (TN).

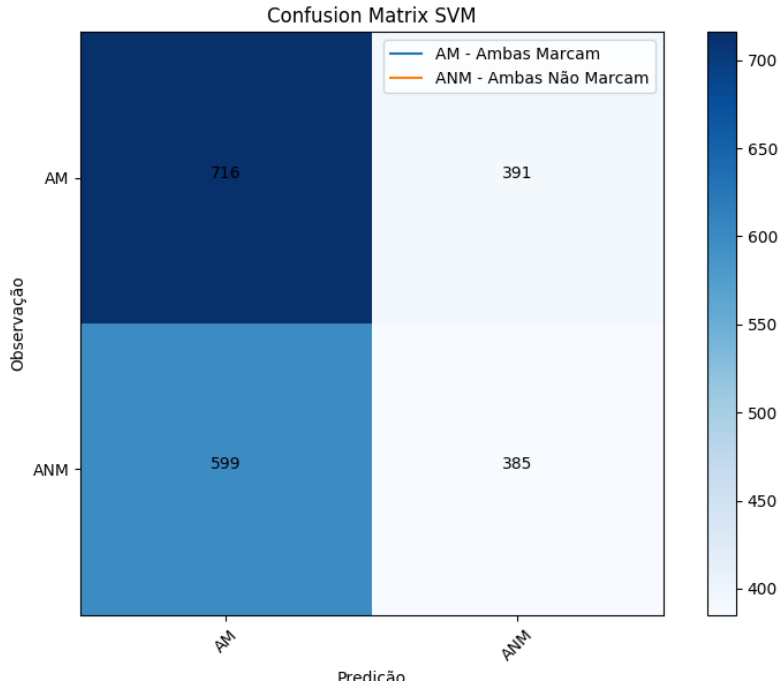

Figura 2. Matriz Confusion - SVM

Parâmetros e Precisão - SVM A precisão deste algoritmo depende, dos valores dos parâmetros e das suas combinações. A grande questão, é encontrar o

melhor hiperplano na separação dos conjuntos de dados. A precisão depende dos parâmetros C, Gamma, e o Kernel. Feita a análise desses valores neste caso de estudo, foi criado um intervalo de valores para cada parâmetro, tendo-se chegado à seguinte conclusão:

- Usar o intervalo [0.001, 0.01, 0.1, 1, 10, 100] para o parâmetro C, o melhor valor é 0,01;
- Usar o intervalo [0.0001, 0.001, 0.01, 0.1] para o parâmetro gamma, o melhor valor é 0,0001;
- Para o parâmetro Kernel, linear é melhor que rbf.

Parâmetros e Precisão (DT) O primeiro parâmetro para ajustar é a profundidade máxima(max_depth). Isso indica o quão profunda a árvore pode ser. Quanto mais profunda a árvore, mais divisões ela tem e capta mais informações sobre os dados. A árvore foi ajustada com uma profundidades variando de 1 a 32, e executa-se o treino e testa-se, a qual profundidade tem melhor pontuação Area Under The Curve (AUC). De acordo com Figura 3 à esquerda, observa-se que a profundidade que teve melhor pontuação AUC é 3.

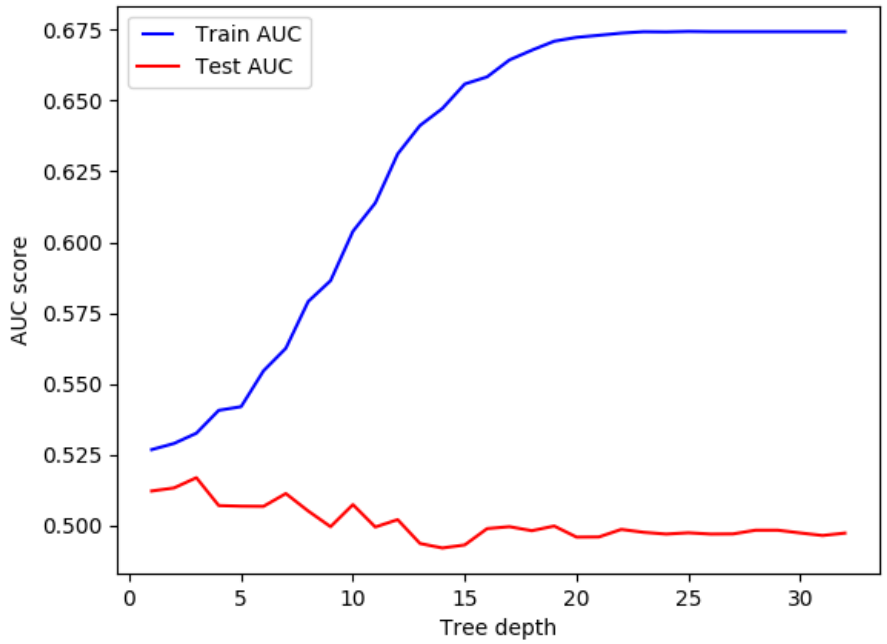

Figura 3. Profundidade Máxima e Folha de Amostra Mínimo

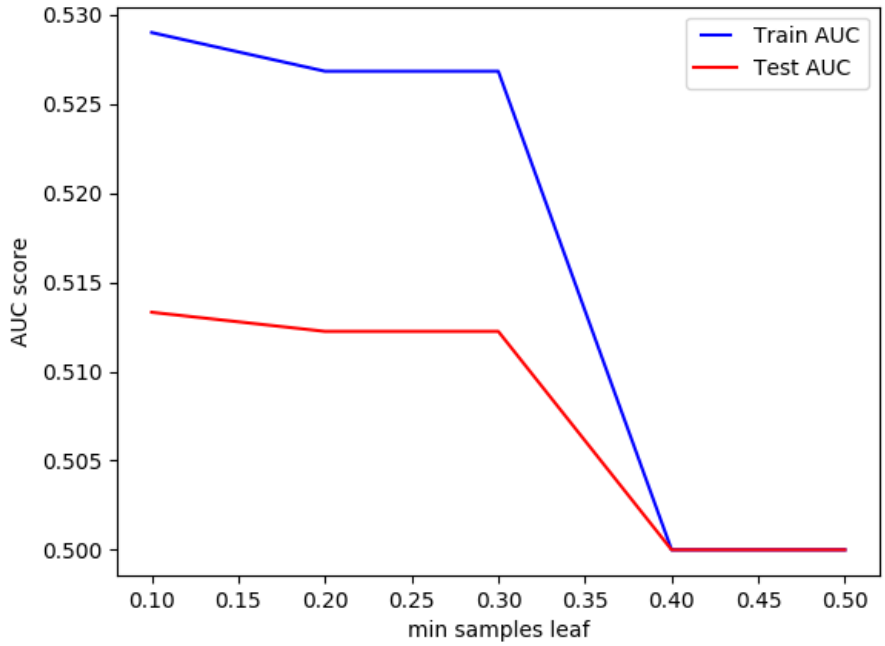

Figura 4. Profundidade Máxima e Folha de Amostra Mínimo

O segundo parâmetro é o min_samples_leaf. É o número mínimo de amostras necessárias para estar em um nó folha. Este parâmetro descreve o número mínimo de amostras nas folhas. Segundo a Figura 3 à direita, observa-se que 5 é o melhor resultado.

De todos os algoritmos de classificação aplicado a este problema, o LR é o algoritmo que teve a pior precisão de previsão, apesar de segundo a Tabela 2 ter obtido melhor precisão do que SVM.

Parâmetros e Precisão - KNN O algoritmo KNN classifica exemplos, tendo em vista a classe dos k vizinhos mais adjacentes. No caso de K ser igual a 1, então o novo dado é classificado com a mesma classe do exemplo mais adjacente. E, no caso de K ser maior do 1, então são consideradas as classes dos K exemplos mais adjacentes para realizar a classificação.

Uma vez que para ter uma boa classificação com este algoritmo, é imprescindível escolher o parâmetro K mais ideal, foram aplicadas duas formas para descobrir o melhor K, como podemos observar na Figura 7, em que o valor mais ideal de K é 20.

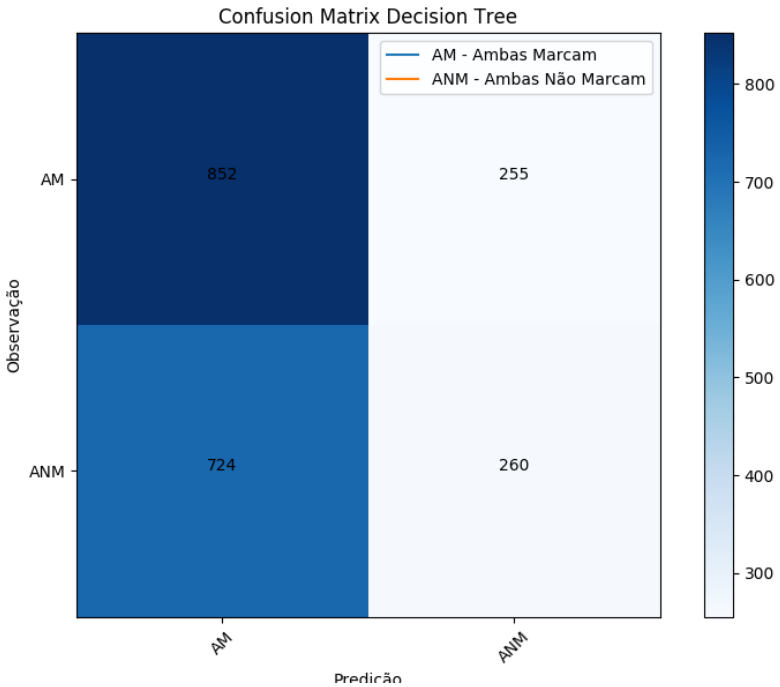

Figura 5. Matriz Confusion - DT

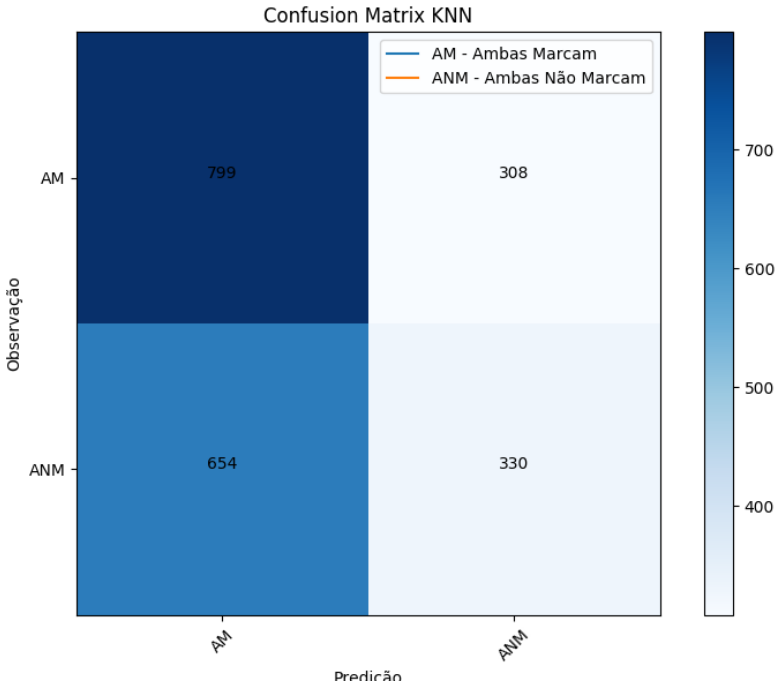

Figura 6. Matriz Confusion - KNN

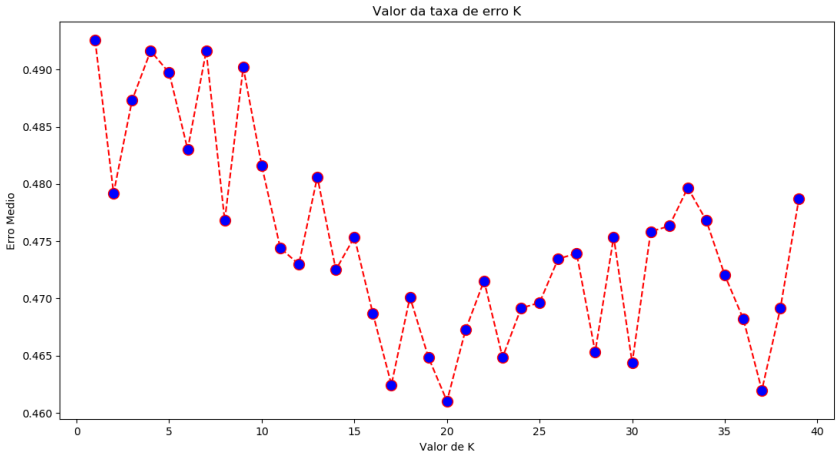

Figura 7. Valor K mais indicado para o algoritmo KNN.

6 Caso de Estudo

Tabela 1. Classification Report dos algoritmos aplicados ao caso de estudo

Algoritmos	Target	Precision	Recal	F1_Score	Suport
SVM	*Ambas Não Marcam*	0,54	0,65	0,59	1107
	Ambas Marcam	0,50	**0,39**	**0,44**	984
	Micro Average	0,53	0,53	0,53	2091
	Macro Average	0,52	0,52	0,51	2091
	Weighted avg	0,52	0,53	0,52	2091
DT	*Ambas Não Marcam*	0,54	**0,77**	**0,64**	1107
	Ambas Marcam	0,50	0,26	0,35	984
	Micro Average	0,53	0,53	0,53	2091
	Macro Average	0,52	0,52	0,49	2091
	Weighted avg	0,52	0,53	0,50	2091
KNN	*Ambas Não Marcam*	**0,55**	0,72	0,62	1107
	Ambas Marcam	**0,52**	0,34	0,41	984
	Micro Average	**0,54**	**0,54**	**0,54**	2091
	Macro Average	**0,53**	**0,53**	**0,52**	2091
	Weighted avg	**0,53**	**0,54**	**0,52**	2091

O problema a ser estudado com as várias técnicas de algoritmos de classificação, é se ambas as equipas marcam golo ou não num jogo. Baseado em registos passados, o modelo vai prever a cada jornada o resultado se ambas as equipas

marcam golo na Liga NOS. Conforme evidenciado na Tabela 2, foi efetuada a previsão da 18a jornada da liga portuguesa, e o modelo de classificação com o melhor resultado é o KNN.

Resultado Obtidos e Avaliação dos Algoritmos Neste caso de estudo, foram aplicadas várias percentagens na divisão entre os dados de treino e de teste. A melhor percentagem foi 30 % para os dados de teste e 70 % para os dados de treino. Assim sendo, existem 4877 amostras no conjunto de treino, e 2091 amostras no conjunto de teste.

Inicialmente, utilizando o SVM com os valores default, sem escolher o melhor valor do parâmetro, os resultados obtidos segundo a tabela 1, demonstram que com o algoritmo SVM aplicado neste projeto, existe uma precisão de previsão de cerca 52,65 %, como pode se pode observar na Figura 2. Mas de acordo com a Tabela 2, aplicando este método à 18a Jornada da Liga NOS, obtemos uma taxa de acertos de 44,44 %, em 9 jogos, tendo-se acertado em 4 jogos.

A posteriori, quando foram designados os melhores valores de C, Gamma e Kernel, obtemos uma melhoria de 0,5 %. A Figura 2 à esquerda, demonstra a MC antes de encontrar o melhor valor de cada um dos parâmetros. A figura da direita demonstra a MC depois de ser aplicado o melhor valor dos parâmetros. Pode-se observar uma melhoria na precisão do algoritmo. Aplicando novamente à 18a Jornada da Liga NOS obtemos uma precisão de 55,56 %, onde em 9 jogos se acerta em 5 jogos.

Tabela 2. Precisão da 18ª Jornada da Liga NÓS

Equipa Casa	**Equipa Visitante**		**Ambas Marcam**	**KNN**	**SVM**	**Decision Tree**	**Linear Regression**
Rio Ave	Feirense	0-0	0	0	1	1	0
Boavista	Portimonense	0-2	0	0	0	0	0
Aves	Guimaraes	2-1	1	1	1	1	0
Santa Clara	Maritimo	0-1	0	0	0	1	0
Sp Lisbon	Moreirense	2-1	1	0	0	0	0
Belenenses	Tondela	2-2	1	0	0	0	0
Setubal	Benfica	0-1	0	0	0	0	0
Chaves	Porto	1-4	1	1	0	1	1
Nacional	Sp Braga	0-3	0	0	1	0	1
			100%	**77,78%**	44,44%	55,56%	55,56%

O algoritmo de classificação DT, obteve a segunda melhor precisão da previsão se ambas as equipas marcam. A Tabela 1 mostra que tem uma precisão de 53,18 %. Conforme a Tabela 2, aplicando-se este método à 18a Jornada da Liga NOS, obtemos uma precisão de 55,56 %, o que quer dizer que em 9 jogos, acertou 5 jogos.

Segundo a Tabela 1, com o K igual a 20 obteve-se uma precisão de previsão de 53,99%. Aplicando este algoritmo à 18a Jornada tivemos a melhor classificação com cerca de 77,18 %, o que quer dizer que em 9 jogos, acertou-se 7 jogos. Desta forma pode-se considerar, que para este caso de estudo onde existem muitas variáveis imprevisíveis, obter este resultado é muito bom. Não foi encontrado nenhum projeto com similaridade que pudesse ser comparado com este modelo implementado.

7 Conclusão e Crítica

Os custos de erro ou medir adequadamente o desempenho de classificadores através da taxa de erro ou precisão, assume um papel preponderante em ML, uma vez que o real objetivo consiste em construir classificadores com baixa taxa de erro em novos exemplos[19].

Neste caso de estudo, não é fácil encontrar as características ideais para minimizar a taxa de erro, uma vez que, conforme já referido, no futebol existem muitas variáveis imprevisíveis, elencando-se a título de exemplo os fatores meteorológicos, psicológicos dos jogadores e treinadores, entre outras características que não se podem definir. Se fosse possível, seria de extrema importância para os classificadores melhorar as suas performances.

O desenvolvimento do presente estudo possibilitou a aquisição de conhecimento, análise, aplicação das técnicas de ML, implementando classificadores para dar resposta, face aos jogos futuros da liga portuguesa ou qualquer competição nacional. Além disso, também permitiu descobrir entre os algoritmos de classificação qual deles se adapta melhor à natureza deste projeto, tendo-se concluído que K-Nearest Neighbor conseguiu ter a melhor precisão de previsão dos resultados, com cerca de 54%.

Dada a importância do assunto do projeto, propõe-se como trabalho futuro o estudo, descoberta, desenvolvimento e implementação de novas características que podem ser relevantes, de forma a aumentar a precisão do modelo classificador construido. Também, para trabalho futuro, a necessidade de efetuar um estudo profundo, modificando o conjunto de dados, adicionando duas características (Data do jogo e poder defensivo das equipas), e estudar a importância dessas features face ao problema de estudo em destaque. Pode também, alargar-se o horizonte deste projeto, em vez somente fazer a previsão se ambas as equipas marcam, podendo-se avançar e aplicar o estudo em prever qual das equipas vence o jogo, e também ver se as duas equipas juntas marcam mais de 2.5 golos, que é uma das novas facetas nas casas das apostas.

Resumidamente, conclui-se que ao objeto de estudo podem ser acrescentadas mais características e continuar com a mesma precisão ou pior. Tem que se ter em consideração a relevância da nova característica face ao estudo em questão. Foi adicionado o poder do meio campo das equipas e a precisão manteve-se a mesma em todos os classificadores., o que permite registar que esta característica não tem relevância se as equipas marcam golos ou não. Finalmente, propõe-se o estudo mais profundo do algoritmo LR, comparando com o modelo implemen-

tado, para saber se obterá melhores resultados. Efetuando mais um teste com a 19o Jornada da Liga NOS, novamente o algoritmo KNN em 9 jogos acertou 7 jogos, enquanto que SVM acertou 5 jogos e o DT acertou 4 jogos, o que atesta que neste caso de estudo, o KNN é melhor algoritmo para a previsão.

Referências

[1] N. Ancona, G. Cicirelli, A. Branca, and A. Distante. Goal detection in football by using support vector machines for classification. In *Neural Networks, 2001. Proceedings. IJCNN'01. International Joint Conference on*, volume 1, pages 611–616. IEEE, 2001.

[2] H. Chih-Wei and L. Chih-Jen. A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks*, 13(2):415–425, March 2002.

[3] V. Cortes, C.and Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[4] B. Deekshatulu, P. Chandra, et al. Classification of heart disease using k-nearest neighbor and genetic algorithm. *Procedia Technology*, 10:85–94, 2013.

[5] B. Faria, L. Reis, N. Lau, and G. Castillo. Machine learning algorithms applied to the classification of robotic soccer formations and opponent teams. In *2010 IEEE Conference on Cybernetics and Intelligent Systems*, pages 344–349, June 2010.

[6] C. Ferrero. *Algoritmo kNN para previsão de dados temporais: funções de previsão e critérios de seleção de vizinhos próximos aplicados a variáveis ambientais em limnologia*. PhD thesis, Universidade de São Paulo, 2009.

[7] P. Flach. *Machine learning: the art and science of algorithms that make sense of data*. Cambridge University Press, 2012.

[8] T. Furey, N. Cristianini, N. Duffy, D. Bednarski, M. Schummer, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.

[9] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422, 2002.

[10] A. Heuer and O. Rubner. Towards the perfect prediction of soccer matches. *arXiv preprint arXiv:1207.4561*, 2012.

[11] Z. Huang, H. Chen, C. Hsu, W. Chen, and S. Wu. Credit rating analysis with support vector machines and neural networks: a market comparative study. *Decision support systems*, 37(4):543–558, 2004.

[12] C. Igiri and E. Nwachukwu. An improved prediction system for football a match result. *IOSR Journal of Engineering (IOSRJEN)*, 4:12–20, 2014.

[13] S. Imandoust and M. Bolandraftar. Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background. *International Journal of Engineering Research and Applications*, 3(5):605–610, 2013.

[14] D. Karlis and I. Ntzoufras. Bayesian modelling of football outcomes: using the skellam's distribution for the goal difference. *IMA Journal of Management Mathematics*, 20(2):133–145, 2008.

[15] I. Kuncheva, Ludmila. Editing for the k-nearest neighbors rule by a genetic algorithm. *Pattern Recognition Letters*, 16(8):809–814, 1995.

[16] P. Langley et al. Selection of relevant features in machine learning. In *Proceedings of the AAAI Fall symposium on relevance*, volume 184, pages 245–271, 1994.

[17] J. Lope, D. Maravall, and J. Martin. Robust high performance reinforcement learning through weighted k-nearest neighbors. *Neurocomputing*, 74(8):1251–1259, 2011.

[18] M. Maher. Modelling association football scores. *Statistica Neerlandica*, 36(3):109–118, 1982.

[19] J. Monard, M.and Baranauskas. Conceitos sobre aprendizado de máquina. *Sistemas Inteligentes-Fundamentos e Aplicações*, 1(1):32, 2003.

[20] B. Natarajan. *Machine learning: a theoretical approach*. Elsevier, 2014.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[22] D. Prasetio and D. Harlili. Predicting football match results with logistic regression. In *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, pages 1–5, Aug 2016.

[23] P. Rudin. Football result prediction using simple classification algorithms, a comparison between k-nearest neighbor and linear regression, 2016.

[24] L. Samaniego and K. Schulz. Supervised classification of agricultural land cover using a modified k-nn technique (mnn) and landsat remote sensing imagery. *Remote Sensing*, 1(4):875–895, 2009.

[25] H. Schmidt. Uso de técnicas de aprendizado de máquina no auxílio em previsão de resultados de partidas de futebol., 2017.

[26] J. Shin and R. Gasparyan. A novel way to soccer match prediction. *Stanford University: Department of Computer Science*, 2014.

[27] M. Shouman, T. Turner, and R. Stocker. Using decision tree for diagnosing heart disease patients. In *Proceedings of the Ninth Australasian Data Mining Conference-Volume 121*, pages 23–30. Australian Computer Society, Inc., 2011.

[28] R Stefani. Predicting score difference versus score total in rugby and soccer. *IMA Journal of Management Mathematics*, 20(2):147–158, 2009.

[29] J. Texeira. *Inteligência artificial*. Pia Sociedade de São Paulo-Editora Paulus, 2014.

[30] D. Wu. Supplier selection: A hybrid model using dea, decision tree and neural network. *Expert Systems with Applications*, 36(5):9105–9112, 2009.

[31] Q. Yu, A. Sorjamaa, Y. Miche, A. Lendasse, E. Séverin, A. Guillén, and F. Mateo. Optimal pruned k-nearest neighbors: Op-knn application to financial modeling. In *Hybrid Intelligent Systems, 2008. HIS'08. Eighth International Conference on*, pages 764–769. IEEE, 2008.