UNIVERSITAT POLITÈCNICA DE VALÈNCIA

etsinf

Escola Tècnica Superior d'Enginyeria Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

???? ????????? ????????????? ? ?????

DEGREE FINAL WORK

Degree in Computer Engineering

Author: ???????? ???? ???????????

Tutor: ?????? ???? ???????????

Course 2021-2022

Abstract

????

Key words: ?????, ????? ?????, ?????????????

Resum

????

Paraules clau: ????, ?????????, ????, ??????????????????

Resumen

????

Palabras clave: ?????, ???, ???????????????

Contents

List of Figures

List of Tables

CHAPTER 1
Introduction

The aim of the following report is to describe the study carried out on the application of Machine Learning, namely Neural Networks, for the prediction of football match results based on the past performance of the involved teams.

Large amounts of data have been collected during matches by TV channels, football clubs and betting agencies in the last few years. This enables the detailed analysis of the players and teams performance, and therefore the creation of predictive models of their future performance, and consequently the results of the matches. Nowadays there are still few research studies on the matter, but increasingly more researchers are deciding to apply their knowledge in the domain of sport analytics.

This document describes the complete process that has been followed, as well as the different techniques that have been used to create a new database, process the data, perform feature engineering, build a basic baseline model trained with the bookmaker odds, and finally build the final Bagging Neural Networks based model. All this after a revision and critique of the previous relevant related literature.

In order to develop the model, first of all, Multi-layer Perceptron basic models have been trained by seeking the optimal hyper-parameters, as well as searching for the features that better train our model. From the obtained results, more complex models have been trained, to finally obtain a model based on Bagging Neural Networks, where we have X independent neural networks trained with different hyper-parameters and with different input data. At the end we average their predictions and get a final prediction of whether the home team wins, the away team wins or the result is a draw.

1.1 Motivations

The reasons that led me to conduct this research have been several. First, my passion for football motivated me to consider doing this work. Since I was a child I have been passionate about football and even more about the statistics and data of teams and players.

Secondly, from a professional point of view, it was a great opportunity for me to introduce myself into research and Data Science, the field that I want to dedicate to from now on. Also the fact that the scope of Sport Analytics is still little studied motivated me. Nevertheless, the world of football is gradually incorporating Big Data and Artificial Intelligence. For this reason, I also think that in the future it could be a professional

career path in which I could work because of my two passions: football and Data Science.

As I said, it was a great opportunity to introduce me to Data Science in an area that has not been studied much, so I was also encouraged to see it as a challenge. I was conscious of the complexity of the problem, since finding a correlation between the events of a football match and its outcome is no trivial problem given its high component of randomness and the data limitations that we had.

1.2 Objectives

It was difficult to set an initial objective due to the extremely wide-open nature of the study. From the beginning, however, the target was to design and train a model that could predict whether the home team would win, the away team would win or draw. Moreover, the objective was that the model could outperform the trained model with the odds made by the bookmakers. The odds are taken as a reference to be able to evaluate the performance of the model.

Another goal has been to approach or beat the results of past similar work. This objective is hard to achieve as each work uses a different data source and therefore different data, so it is not straightforward to compare. Nonetheless, we have trained some of the models used in previous works with our data in order to compare and improve our model.

As for personal goals, it also includes gaining experience and skills in data processing and Machine Learning technologies employed, as well as consolidating theoretical knowledge learned on statistics and Machine Learning over the last six months.

1.3 Methodology

The methodology used to carry out the work has been the following:

1. Review of previous related work.
   (a) Notation of the designed models and results obtained and/or conclusions reached in each work.
   (b) Write notes on the highlights of each paper, to be possibly used later in this document.
   (c) Brainstorm and design a possible predictive model architecture based on previous literature.
2. Collection and generation of datasets.
   (a) Search and gathering free and open soccer databases available on the Internet.
   (b) Inspection and analysis of the data content of each database so as to identify which ones are compatible among each other, provide a reasonable amount of data and can be used to train our model.
   (c) Creation of the bookmaker odds dataset from a database that contains the betting odds of the matches.
   (d) Creation of datasets from multiple compatible databases, merging datasets and generating new variables from existing ones. This step consists of a long-term work of programming and creativity.

3. Training of the baseline model trained with the bookmaker odds.
4. Data analysis
   (a) In-depth statistical analysis of data.
   (b) Dimensional reduction and feature selection based on the data analysis.
5. Design and training of simple Neural Networks based predictive models.
   (a) Training of simple Multi-layer Perceptron models with different features selected.
   (b) Analysis of the obtained results and of learning of the neural networks.
   (c) Model optimization based on model parameters and hyper-parameters.
   (d) Study and training of models with a deeper architecture.
   (e) Analysis of the obtained results and selection of the best models.
6. Implementation and training of the Bagging-based model.
   (a) Training of the Bagging based model with the corresponding input for each optimized model.
   (b) Analysis and comparison of the results with the results of the simpler models.
   (c) Comparison of results with baseline results and with previous results of related works.
   (d) Study further improvements of the model.
7. Enhance the Bagging-based model.
   (a) Train new models with other data selected with a new approach.
   (b) Analysis of results and comparison with the previous multi-layer perceptron models.
   (c) Optimization of this models, regarding the input and hyper-parameters.
   (d) Adding the new models to the Bagging-based model.
   (e) Analysis and comparison of results with previous Bagging-based model.
   (f) Comparison of results with the baseline and related works.
8. Simulation of a whole season in order to compare the simulated league table with the actual one.

1.4 Thesis structure

????? ????????????? ????????????? ????????????? ????????????? ?????????????

CHAPTER 2
State-of-the-Art

El analisis y el estudio del futbol para encontrar patrones y estrategias que hagan a un equipo ganador es una actividad que se lleva realizando durante más de seis décadas. Ya a comienzos de los 50 del siglo pasado Charles Reep recogía estadísticas a mano que sugerían que la clave para marcar goles era pasar el balón lo más rápido posible de atrás hacia delante, lo que indirectamente conllevó el inicio del *long-ball movement* en el futbol inglés. [Reep and Benjamin, 1968].

Sin embargo es los últimos años la tecnología ha permitido que se recojan grandes volumenes de datos que permiten un análisis más exhaustivo, y sobretodo permiten diseñar algoritmos que saquen información y aprendan de estos datos. Hay tres tecnologías que permiten actualmente la recogida de los eventos que suceden durante un partido de fútbol: *soccer-logs*, *video-tracking* y *GPS data* [Pappalardo et al., 2019]. *Soccer-logs* describen los eventos que ocurren durante un partido y son recogidos mediante un *tagging* software propietario. Los datos recogidos mediante *video-tracking data* describen la trayectoria de los jugadores durante el partido y son recogidos a partir de grabaciones del partido. Por último, *GPS data* describe la trayectoria de los jugadores durante sesiones de entrenamiento y son recogidos mediante dispositivos GPS incorporados en la equipación de los jugadores. Este vertiginoso aumento de la cantidad de datos ha provocado el problema opuesto al que había antes. Hoy, la enorme cantidad de datos se ha vuelto un obstáculo en sí para el análisis de los datos, debido a una falta de lineas metodológicas y modelos teóricos de la toma de decisiones tácticas en el futbol [Rein and Memmert, 2016].

Pese a esta riqueza de datos generada, lo cierto es que estos datasets suelen estar dificilmente disponibles para investigación científica [Pappalardo et al., 2019]. Sin embargo, esto no ha impedido que se hayan realizado estudios acerca del analisis táctico o de la predicción de resultados. Muchos investigadores se han preguntado que relación existe entre el rendimiento y el éxito deportivo, tanto en el futbol como en otros deportes [Yucesoy and Barabási, 2016]. La conclusión de estos estudios es que existe una clara relación en el deporte entre el rendimiento y el éxito o fracaso. Sin embargo, por ejemplo Pappalardo habla de que mientras la victoria o la derrota puede ser explicada por el rendimiento, es difícil detectar empates usando técnicas de Machine Learning [Pappalardo and Cintia, 2018, Ulmer et al., 2013, Tax and Joustra, 2015].

2.1 Tactical analysis

Uno de los ámbitos de estudio principales en el futbol es el analisi de tácticas que permitan mejorar el rendimiento de un equipo. Táctica y estrategia pese a estar inter condi-

cionadas son conceptos distintos, Rein and Memmert define them as "while the team strategy describes the decisions made before the game with respect to how the team wants to play whereas the tactic is the result of the ongoing interactions between the two opposing teams" [Rein and Memmert, 2016]. Las tácticas especifican como un equipo maneja el espacio, el tiempo y las acciones individuales con el objetivo de ganar el partido, y dependen en el status del equipo y del rival, tanto previo al partido como durante, y de factores externos como quien juega de local o visitante, la meteorología o el descanso.

La detección automática de las tácticas y comportamiento de un equipo ha sido objeto de investigación durante los últimos años. Factores como la "possession directness" correlacionados con la posesión de balón y los pases del tercio defensivo del campo al ofensivo son importantes para identificar estilos de juego [Fernandez-Navarro et al., 2016]. Para el estudion de las tácticas, muchos estudios se han centrado en estudiar e identificar las formaciones de los equipos y la distribución de los jugadores en el campo. Bush et al. por ejemplo, investigaron sobre la relación de la formación con el rendimiento fisiológico y con las habilidades técnicas de los jugadores, encontrando que los jugadores corrían más distancia cuando se enfrentaban a formaciones 4-2-3-1 en comparación a 4-4-2 [Bush et al., 2015]. Otro enfoque, dado por Silva et al., se basa en analizar la superioridad numerica en una particular parte del campo. Este enfoque dió como resultado que el control del espacio es un aspecto central en la táctica [Silva et al., 2014].

Relacionado con este enfoque está el Team Centroid method, el cual se basa en el centro geométrico de las posiciones de los jugadores de un equipo, y se usa para analizar el comportamiento de este durante eventos clave en el partido como goles [Rein and Memmert, 2016]. Más recientemente este método se ha extendido calculando la Approximate Entropy (ApEn), una técnica de medida de non-linear time-series. Goncalves et al. usó este metodo para investigar sobre la coordinacion del equipo entre y dentro de los defensas, centrocampistas y atacantes. La investigación mostró que los movimientos eran más regulares con respecto al centroid de sus respectivos subgrupos comparado con los otros grupos [Gonçalves et al., 2014]. Todos estos estudios, sin embargo, se centran en un aspecto muy concreto, por lo que de momento no está claro como las formaciones interactuan con la técnica individual y las tácticas [Rein and Memmert, 2016].

Otros dos enfoque emergentes en el analisis de las tácticas en el futbol se basan en network approaches and Machine Learning methods. En cuanto al primero, la idea básica es modelar los jugadores como nodos y los pases como vertices con pesos, que representan el numero de pases entre ellos. Esta representacion permite encontrar facilmente los jugadores clave en un equipo considerando sus conexiones [Gama et al., 2014]. Wang et al. usaron un Bayesian latent model que fue capaz de identificar automáticamente patrones tácticos, que combinados con información sobre jugadas con éxito, detectaba los más efectivos [Wang et al., 2015]. Machine Learning methods como EM han sido también usados para la detección automática de formaciones, demostrando por ejemplo que los equipos suelen usar formaciones defensivas durante los partidos como visitante [Bialkowski et al., 2014]. Es, sin embargo, en la predicción de los resultados de partidos, basada en las conclusiones y assumptions del analisis previo, donde se usan más los metodos basados en Machine Learning.

2.2 Forecasting football matches

Los humanos siguen siendo mejores prediciendo el resultao de un partido, ya que toman en consideración la calidad técnica de los jugadores así como también los factores emo-

cionales que afectan el resultado final. Sin embargo, los humanos se dejan llevar por sus emociones en la predicción lo que lleva muchas veces a fallos en sus predicciones [Jain et al., 2021]. El rápido desarrollo del Machine Learning, y más en concreto del Deep Learning, durante los últimos diez años ha contribuido al incremento de investigaciones sobre el desarrollo de modelos predictivos de partidos de futbol.

Ulmer y Fernández realizaron un estudio que consistió en un problema de clasificación de 3 clases (home-win, away-win and draw) usando Naive Bayes, Hidden Markov models, SVM, Random Forest y OneVsAll SGD [Ulmer et al., 2013] con partidos de la Premier League inglesa. Ulmer y Fernandez dicen que los retos más grandes a los que se enfrentaron fueron el gran componente aleatorio de los datos y la gran cantidad de outliers que hay en el futbol, como por ejemplo el titulo del Leicester en 2016. En cuanto a su metodología, primero siguieron un proceso de selección de características basado en la literatura anterior y en su intuición. Seleccionaron características como si el equipo jugaba de local o visitante, el Ranking ELO or the "streakness" of the last n games. Con las rachas de los equipos se encontraron con el dilema de qué hacer con los primero n partidos. Los dos enfoques que consideraron fue el de escalar los datos y el de ignorar los n primeros partidos, el cual ofreció mejores resultados. El numero de partidos óptimo, n^*, fue entrenado como hiper-parámetro del modelo en un rango entre 2 y 7. Los modelos que mejor resultado tuvieron fue el OneVsAll SGD con un error de 0.48. En cuanto al SVM, primero se usó un RBF-SVM (0.52 error) pero overfitted, así que se intentó con un modelo más simple, un Linear SVM (0.49) que solventó el overfitting y permitió añadir más características al modelo. Naive Bayes (0.56) y Hidden Markov (0.56) fueron los modelos que peor resultados ofrecieron, el primero por la falsa assumption de la independencia de las muestras y el segundo por la model assumption that past states (result of the matches) are hidden, while they are known.

Más recientemente se han usado modelos de Deep Learning para predecir automáticamente los resultados de los partidos. Rudrapal et al. propusieron un MLP based prediction model, y lo compararon con classical Machine Learning models [Rudrapal et al., 2020]. Con un MLP model with 10 hidden states consiguieron un accuracy del 73.57%, por un 72.92% del Random Forest, un 58.77% del SVM y un 65.84% de Gaussian Naive Bayes. Los modelos los entrenaron con 11,400 partidos de la Premier League, entre la temporada 2000-01 y 2015-16, pero previamente a su entrenamiento realizaron una selección de características. Tras ella, el input de los modelos consistía en características relacionadas con cada equipo, como la habilidad en ataque, en el centro del campo y en defensa, un team rating y un indicador de la racha del equipo; relacionadas con cada jugador, como su valoración, su potencial, su valor de mercado y indicadores del portero, defensa, centro del campo y ataque; y relacionadas con el contexto del partido, como la media de puntos conseguidos en casa y fuera de casa, la forma de los últimos cinco partidos y la diferencia de goles.

Ogunseye et al. intentaron predecir los goles marcados por el Manchester United entre 2009 y 2018 usando un MLP de 6 input units y 5 hidden layers que consiguió un acierto del 73.72% [Ogunseye et al., 2019]. Sin embargo, no sólo la Premier League inglesa ha sido objeto de estudio, otras ligas como la holandesa también han sido objeto de estudio. Tax and Journal, usaron trece temporadas de la Eredivise holandesa para entrenar varios algoritmos de clasificación (Naive Bayes, LogitBoost, MLP, RandomForest, CHIRP, FURIA, DTNB, Decission Tree (J48) and HyperPipes) combinados con técnicas de reducción de la dimensionalidad. Los mejores resultados fueron usando la combinación de PCA (con 15% varianza) con Naive Bayes o un MLP (56% of accuracy). Los modelos fueron entrenados con un dataset híbrido de betting odds and public data features. Sin embargo,

no utilizan Cross-Validation para entrenar sus modelos, debido a la naturaleza temporal de los datos, y de las trece temporadas, utilizan las siete primeras para entrenar y seis para el test. El alto acierto con Naive Bayes, dicen, se puede deber a que la dependencia asumida entre las características desaparece con la aplicación de PCA con poca varianza. [Tax and Joustra, 2015].

Otro estudio interesante es el que realizaron Guan and Wang aplicando un enfoque muy novedoso, una combinación de grey prediction algorithm and extreme learning machine algorithm [Guan and Wang, 2022]. Grey prediction algorithms are suitable for cases where there are small samples of data, they perform worse in large sample data, while extreme learning machines are suitable for cases of large samples. A grey box model combina una estructura parcial teórica con datos y se opone a los modelos black box donde no se asume ningún modelo teórico, y a los white box models que son puramente teóricos. Por otra parte, extreme learning machine is a single hidden layer feedforward neural network con un método de aprendizaje completamente diferente al utilizado por los tradicionales metodos iterativos. En este modelo los input weight son generados aleatoriamente, mientras que los output weights son obtenidos con analisi y cálculo, de tal forma que evita las dificultades de la optimización no lineal de los valores óptimos de los input weights[1]. Los dos modelos se combinan usando una función que usa como input el output de cada modelo. La combinación puede ser lineal (equal weighted average, weighted based on the prediction error or based on the covariance) or non-lineal (weighted geometric average or weighted harmonic average) [Guan and Wang, 2022].

Sin embargo, la mayor parte de las publicaciones recientes se han centrado en la aplicación de RNN y LSTM models para pronosticar el resultado de los partidos de fútbol. Un acierto del 80% se ha conseguido con modelos LSTM [Jain et al., 2021]. Los modelos LSTM son modelos de RNN que estan diseñados para recordar a la vez los valores más antiguos y los más recientes. Si, el modelo decide que una característica es importante para decidir el output se guarda dicha característica por un período de tiempo más largo [Jain et al., 2021]. En este trabajo, el enfoque fue deducir las mejores características a partir de los resultados de los partidos anteriores, por ello hicieron uso de un modelo LSTM entrenado con datos de la liga inglesa entre 2010 y 2018. Los datos incluían por cada partido diferencia de goles de cada equipo y datos generados a partir de otros atributos, como por ejemplo la generación de los goles marcados por cada equipo, la racha de cada equipo en los últimos 3 y 5 partidos o la diferencia de puntos de cada equipo.

Un enfoque muy diferente pero también basados en redes LSTM fue el que hicieron Pettersson y Nyquist. Durante el trabajo se describe el estudio que realizaron sobre diferentes formatos posibles para el input de la [Nyquist and Pettersson, 2017]. Entrenaron el modelo LSTM con datos procedentes de 63 diferentes países, y los datos consistían en eventos que ocurrían durante los partidos, como goles, faltas o penaltis. Para utilizar estos datos como input de su modelo tuvieron que transformar los eventos para que todos tuviesen el mismo tamaño. Las opciones que plantean son usar Deep Embeddings, inspirados por *word2vec*, y que funcionan muy bien para NLP, pero que son aprendidos durante el entrenamiento del modelo; One-Hot Vector con todos los atributos de todos los eventos, esto significa que para cada evento hay muchas columnas vacías ya que todos los atributos menos el id del jugador y del equipo estan one-hot encoded; la última opción es Concatenated Embedding Vectors para todos los atributos, una ligera variación del one-hot vector, ya que usamos un embedding por cada tipo de evento y un lookup para todos los valores en el vector de atributos y concatenamos los embeddings resul-

[1] read about Extreme learning machine here: https://link.springer.com/article/10.1007/s11042-021-11007-7

tantes. El formato finalmente escogido fue este último, y cada sample introducido en el modelo es un evento de un partido. El input consiste de un vector de diez variables, estas variables son a continuación transformadas o bien a un one-hot encoding por cada atributo y luego concatenados, o bien a un embedding lookup por cada característica y luego concatenado para formar un vector más grande [Nyquist and Pettersson, 2017, Rahman et al., 2020].

En este trabajo además se aplican dos enfoques distintos para calcular el error durante el entrenamiento. El primero llamado "Many-To-One", el error sólo es calculado una vez, en el último evento del partido para predecir el resultado, y luego usado para el BP. El segundo llamado "Many-To-Many" o *sequence loss* y es un promedio del error durante la secuencia entera, calculando el error por cada evento del partido y haciendo la media antes de usarlo en el BP [Nyquist and Pettersson, 2017, Rahman et al., 2020]. Durante el entrenamiento y evaluación del modelo proponen siete casos de estudio, los seis primeros son modelos LSTM con diferente arquitectura (diferente numero de capas ocultas y LSTM units) pero todos usan el error "Many-To-Many", mientras que el último caso de estudio utiliza "Many-To-One". Además para evaluar y validar los modelos, para cada partido predicen el resultado final cada quince minutos desde el minuto zero. Hasta el minuto 45 los mejores resultados los tiene un modelo con una capa oculta y 256 LSTM units que usa "Many-To-Many", con resultados de 0.44 (min 0),0.45 (min 15),0.47 (min 30) y 0.52 (min 45). A partir del minuto 45 hasta el final el mejor modelo es el que usa "Many-To-One" con 2 capas ocultas y 256 LSTM units, con resultados de 0.63 (min 60), 0.74 (min 75), 0.88 (min 90) y 0.98 (al final del partido). Posteriormente a este trabajo, Rahman realizó un estudio para predecir el resultado de los partidos de la FIFA World Cup 2018 utilizando la misma arquitectura, formato de input y funciones de error con un acierto del 63% [Rahman et al., 2020].

Otros enfoques se han realizado para intentar predecir eventos de gran impacto durante los partidos o los resultados de partidos de futbol, como por ejemplo haciendo un analisi del sentimiento de los tweets posteados durante el partido [Godin et al., 2014, Yu and Wang, 2015]. También se han realizado tanto modelos estadísticos para predecir los resultados [Koopman and Lit, 2015], como modelos híbridos basados en más de un algoritmo que aumentan el acierto del sistema en su conjunto [Guan and Wang, 2022, Jain et al., 2021].

Otra cuestión interesante es tener en mente cual es la finalidad de estos modelos. Muchos de estos modelos se evalúan comparándose con las cuotas de las casas de apuestas, con la finalidad también de comprobar si el modelo puede competir con las casas de apuestas y hacer ganar dinero. Sin embargo, para ello hay que tener en cuenta en qué partidos apostar. Por ejemplo, habrán cuotas con un umbral donde, aunque el modelo prediga una victoria por un equipo, la cuota es tan baja que no valdrá la pena para nada apostar al partido [Bunker and Thabtah, 2019]. Se han presentado modelos que consiguen batir a las casas de apuestas y sacar una cierta rentabilidad [Koopman and Lit, 2015, Tax and Joustra, 2015]. Godin et al. aseguran que con su modelo han logrado sacar un beneficio del 30% en partidos de la segunda parte de la temporada 2013-2014 de la English Premier League [Godin et al., 2014].

2.3 Critique

????? ????????????? ????????????? ????????????? ????????????? ?????????????

reference	data	model	classes	acc.
[Tax and Joustra, 2015]	13 seasons, odds + features	PCA + Naive Bayes, PCA + MLP	3	54.7
[Ogunseye et al., 2019]	Man.Utd matches 2009-2018	MLP 6 units input, 5 hidden, 2 output w/sigmoid	3	73.72
[Rahman et al., 2020]	World Cup 2018 10 embedded feats.	LSTM 2 layers, 10 & 5 units	3	63.3
[Jain et al., 2021]	2010-2018 EPL past matches features	LSTM 1 layer, 64 units	2	80
[Rudrapal et al., 2020]	11,400 EPL matches 20H-20A features: team, player and head-to-head features	MLP 1 layer, 10 units	2	73.57
[Guan and Wang, 2022]	Not specified	Grey prediction method + Extrene Learning Machine	-	over 80
[Ulmer et al., 2013]	Train: 10 EPL sns. Test: 2 EPL sns. match features and time-dependence	NB, Markov, SVM (lin. & RBF), RF and SGD (best). Underpredic. draws	3	error: 0.48
[Nyquist and Pettersson, 2017]	Embedded events of 63 leagues Note: got underpredicted draws	LSTM M-to-1 1 layers 256 units	3	min 0: 44
		LSTM M-to-M 2 layers 256 units		end: 98

Table 2.1: Results of related works

2.4 Possible solutions

????? ????????????? ????????????? ????????????? ????????????? ?????????????

2.5 Chosen technologies

????? ????????????? ????????????? ????????????? ????????????? ?????????????

CHAPTER 3
Data engineering

Una parte fundamental para generar cualquier tipo de modelo predictivo son los datos con los que se entrena. En esta parte se detalla el proceso por el cual los datos se obtienen, se procesan, se analizan y se seleccionan con el objetivo de pasarle al modelo la selección de los mejores y más necesarios datos para que obtenga buenos resultados tanto en el entrenamiento como en la evaluación.

3.1 Source of data

Tras una búsqueda en profundidad en Internet, finalmente, han sido utilizados dos datasets para el desarrollo del modelo predictivo.

3.1.1. Football-Data

El primer dataset fue obtenido de la pagina web Football-Data[1], la cual de dispone de todos los partidos de las grandes princiales ligas europeas y americanas, al igual que otras ligas europeas más exóticas, como la griega, la finesa o la suiza, entre otras. los resultados, las estadísticas, las cuotas de las casas de apuestas y otros atributos como la fecha del partido. Para este estudio se han usado los datos de las cinco grandes ligas, la inglesa, española, italiana, alemana y francesa, desde la temporada 2005-06 hasta la 2021-22.. En la Table 3.3 se muestran los principales atributos disponibles por cada partido en este dataset. Los atributos con el síbolo $*$ no estan disponibles en todas las ligas, y por tanto serán omitidos. Algunos atributos omitidos han sido cuotas de otras casas de apuestas y otros tipos de cuotas de apuestas, como el Handicap Asiático.

3.1.2. Wyscout

Los datos del segundo dataset, obtenido del trabajo de Pappalardo et al., fueron provistos a este por parte de Wyscout, una compañía líder mundialmente en la industria del futbol [Pappalardo et al., 2019]. Los datos fueron recogidos por Wyscout, usando video expert analysis, llamados *operators*, uno por cada equipo y uno extra que supervisa, y entrenados con una colección de datos de futbol con un software propietario (*tagger*. El tagging consiste en tres fases [Pappalardo et al., 2019]:

1. **Setting formations**: un *operator* recoge las formaciones iniciales y las posiciones y dorsales de los jugadores en el campo.

[1] https://www.football-data.co.uk/

2. **Event tagging**: por cada toque de balón, el *operator* selecciona un jugador y crea un nuevo evento en el timeline. Luego le añade un tipo de evento, y un subtipo. Finalmente le añade las coordenadas en el campo del evento y atributos adicionales.

3. **Quality control**: se realizan dos controles. El primero es automático, un algoritmo matches los eventos taggeados por los dos *operators* para validar que tienen sentido los eventos. El segundo control es manual.

Esta base de datos cuenta con todos los partidos y sus eventos de las cinco grandes ligas europeas (inglesa, española, alemana, italiana y francesa) durante la temporada 2017-2018, además que de la World Cup 2018 y European Cup 2016. En la Table 3.1 podemos observar el total de partidos, eventos y jugadores de los que disponemos.

competition	**#matches**	**#events**	**#players**
LaLiga (Spain)	380	628,659	619
Premier League (UK)	380	643,150	603
Serie A (Italy)	380	647,372	686
Bundesliga (Germany)	306	519,407	537
Ligue 1 (France)	380	632,807	629
World Cup 2018	64	101,759	736
European Cup 2016	51	78,140	552
	1,941	3,251,294	4,299

Table 3.1: List of competitions with their total number of matches, events and players.

Este dataset está almacenado por varios archivos JSON, de cada competición hay un archivo JSON de los partidos y otro de los eventos de partidos. Hay otros archivos JSON que almacenan información sobre los jugadores, equipos, entrenadores y competiciones. En la Table 3.4 se muestran los atributos de los partidos y eventos, que son los datasets más destacados de esta base de datos.

type	**subtype**	**tags**
pass	cross, simple pass	accurate, not accurate, key pass, opportunity, assist, goal
shot		accurate, not accurate, block, opportunity, assist, goal
duel	air duel, dribbles, tackles, ground loose ball	accurate, not accurate
free kick	corner, shot, goal kick, throw in, penalty, simple kick	accurate, not accurate, key pass, opportunity, assist, goal
offside touch	acceleration, clearance, simple touch	counter attack, dangerous ball lost, missed ball, interception, opportunity, assist, goal
foul		no card, yellow, red, 2nd yellow

Table 3.2: Event types, subevent types and tags en el Wyscout dataset.

Football-Data		
attribute	**type**	**description**
Div	string	League division
Date	string	Match date (dd/mm/yy)
HomeTeam	string	Home team
AwayTeam	string	Away team
FTHG	integer	Full Time Home Team Goals
FTAG	integer	Full Time Away Team Goals
FTR	string	Full Time Result (H=Home Win, D=Draw, A=Away Win)
HTHG	integer	Half Time Home Team Goals
HTAG	integer	Half Time Away Team Goals
HTR	string	Half Time Result (H=Home Win, D=Draw, A=Away Win)
Referee*	string	Match referee
HS	integer	Home Team Shots
AS	integer	Away Team Shots
HST	integer	Home Team Shots on Target
AST	integer	Away Team Shots on Target
HF	integer	Home Team Fouls Commited
AF	integer	Away Team Fouls Commited
HC	integer	Home Team Corners
AC	integer	Away Team Corners
HY	integer	Home Team Yellow Cards
AY	integer	Away Team Yellow Cards
HR	integer	Home Team Red Cards
AR	integer	Away Team Red Cards
B365H	float64	Bet365 home win odds
B365D	float64	Bet365 draw odds
B365A	float64	Bet365 away win odds
BSH	float64	Blue Square home win odds
BSD	float64	Blue Square draw odds
BSA	float64	Blue Square away win odds
BWH	float64	Bet&Win home win odds
BWD	float64	Bet&Win draw odds
BWA	float64	Bet&Win away win odds
WHH	float64	William Hill home win odds
WHD	float64	William Hill draw odds
WHA	float64	William Hill away win odds

Table 3.3: Attributes of a match in Football-Data dataset.

Matches		
attribute	**type**	**description**
wyId	integer	identifier of the match
competitionId	integer	identifies the competition on the *competition* document.
dateutc	string	date and time with format YYYY-MM-DD hh:mm:ss
roundID	integer	match-day of the competition
winner	integer	id of the team that won the game, or 0 if ended with a draw
label	string	name of the two clubs and the result of the match
venue	string	stadium where the match was held
dateutc	string	date and time with format YYYY-MM-DD hh:mm:ss
teamsData	dictionary	several subfields describing information about each team
teamsData subfields (for each team):		
hasFormation	integer	0 if no formation, and 1 otherwise
score	integer	number of goals scored by the team during the match
side	string	"home" or "away"
teamId	integer	identifier of the team
coachId	integer	identifier of the coach
bench	list	list of players that started on the bench and some basic statistics about the performance (goals, own goals, cards)
lineup	list	list of players that started on the lineup and some basic statistics about the performance (goals, own goals, cards)
substitutions	list	list of team's substitutions during the match, describing the players involved and the minute

Events		
attribute	**type**	**description**
eventId	integer	identifier of the event's type (Table 3.2)
subEventId	integer	identifier of the event's subtype (Table 3.2)
eventName	string	name of the event's type
subEventName	string	name of the subevent's type
tags	list	list of event tags, additional information about the event (Table 3.2)
eventSec	integer	time when the event occurs (in seconds)
id	integer	unique identifier of the event
matchId	integer	identifier of the match the event refers to
matchPeriod	string	"1H" (first half of the match), "2H" (second half of the match)
positions	list	origin and destination positions associated with the event. Each position is a pair of coordinates (x, y).
playerId	integer	identifier of the player who generated the event
matchId	integer	identifier of the match the event refers to
teamId	integer	identifier of the player's team

Table 3.4: Attributes of matches and events on original Wyscout datasets.

3.2 Dataset creation

Para crear un dataset adecuado para entrenar el modelo se han debido de aplicar diferentes transformaciones, se ha debido limpiar y procesar los datos de los bases de datos y

se han unido para obtener unas base de datos más completa. En total hemos creado tres bases de datos:

1. **Betting odds dataset:** creado para entrenar nuestro modelo baseline. Estará formado por las cuotas de Bet365 de los partidos de la temporada 2017-18 de las cinco grandes ligas europeas.
2. **Wyscout based dataset:** este dataset contendrá los partidos de la temporada 2017-18 de las cinco principales ligas europeas. Cada sample corresponderá con un partido, y los 20 atributos que lo componen representaran las estadísticas correspondientes a los últimos n partidos por parte de los jugadores del line-up de cada equipo. Las características se obtendran o bien directamente del dataset de Wyscout o bien se generarán a partir de los eventos o estadísiticas de Wyscout.
3. **Historical dataset:** este dataset contendrá datos de las cinco grandes ligas europeas desde la temporada 2005-2006 a las 2021-22. Cada partido contendrá estadísticas generales sacadas del dataset de Football-Data.

El principal problema de la creación de los datasets ha sido la unión de ambas bases de datos ya que las llaves primarias de ambas bases de datos (de los partidos) no coincide, como tampoco coincide el formato del nombre de los equipo, en resumen, no había una manera directa de unir ambas bases de datos. La solución que se planteó es una solución iterativa. En la primera iteración se detectan los nombres de los equipos que coinciden en ambos datasets, en la segunda iteración con la ayuda de la librería para Python *difflib* se detectan los nombres de clubes que son muy parecidos en ambos datasets, y por último los equipos restantes (una docena) se cambiaron manualmente. Con esto ya podíamos relacionar ambos datasets, ya que ahora podemos identificar en el dataset de Football-Data (Table 3.3) el nombre de un club con su id en Wyscout (Table 3.4).

Sólo hemos tenido en cuenta el nombre de los equipos para relacionar los partidos de ambas bases de datos ya que sólo estamos interesados en los partidos de la temporada 2017-2018 que son los que aparecen en Wyscout (Table 3.1). No nos hace falta ningun otro identificador del partido ya que, por ejemplo, el partido Málaga vs Celta de Vigo, sólo ocurre una vez en liga por temporada.

dataset	#samples	#attributes
Betting odds dataset	1414	3
Wyscout based dataset	1414	20
Historical dataset		

Table 3.5: Number of samples and attributes of the datasets.

3.2.1. Wyscout based dataset

En este dataset creado, todos los samples son independientes debido a las transformaciones realizas sobre el dataset y a los atributos generados que eliminan la dependencia temporal entre los samples. Los atributos generados tienen en comun que son todos generados a partir de características de los últimos n partidos. Por tanto, cada sample (partido) ya contiene toda la información pasada necesaria para pronosticar su resultado, de hecho no contiene más características que las de los partidos pasados. Así se elimina la dependencia temporal entre los samples, haciéndolos independientes.

Se han utilizado los 4 partidos anteriores para el calculo de los atributos de cada sample. Se decidió usar ese número de partidos para no reducir aún más el limitado número de samples de los que disponemos. Usando 4 partidos se reduce de 1826 a 1414 samples. Además, en anteriores trabajos con un enfoque parecido, los mejores resultados se han obtenido con valores entre 4 y 7 partidos [Ulmer et al., 2013, Jain et al., 2021].

En la literatura se ha puesto en duda que exista una relación entre las rachas de los equipos y el resultado de los partidos, especialmente cuando la racha es de victorias [Heuer and Rubner, 2009]. Heuer y Rubner hablan de que una racha negativa tiene un impacto negativo en la probabilidad de victoria, mientras que una racha de victorias no tiene ningún impacto en la probabilidad de ganar de un equipo. Sin embargo, otros estudios contradicen estas afirmaciones y a la propia intuición, mostrando como una racha de victorias resulta en un decremento de la probabilidad de victoria, y una racha de derrotas con un incremento [Goddard, 2006]. Sin duda sigue sin estar clara la relación entre las rachas y los resultados.

En el dataset creado cada sample contiene 20 atributos, los cuales seran descritos y discutidos a continuación:

- **mins***n*_*S*[2]**:** Representan la media de minutos disputados por el line-up de cada equipo en los últimos *n* partidos. Los valores estaran comprendidos por tanto entre 0 y 90. Este atributo debe indicar al modelo si un equipo sale a jugar con titulares, suplentes o si faltan jugadores habituales de un equipo.
- **shots_11***S***:** Indican el numero absoluto de disparos realizados por el inicial inicial de cada equipo en los últimos *n* partidos. Igual, en caso de que el jugador que más disparos suele hacer en un equipo no juegue el partido, se notará su baja en este atributo.
- **shots_acc_11***S***:** Indican la media de shots on target del line-up en los últimos *n* partidos.
- **goals**_*S***:** La suma de goles marcados por el once inicial de cada equipo en los últimos *n* partidos.
- **goals_ratio**_*S***:** El ratio de goles marcados por el once inicial sobre el total de goles marcados por el equipo en los últimos *n* partidos. Este atributo también será un indicador de si faltan jugadores importantes en el line-up o juegan los titulares.
- **passes_11***S***:** Total de pases dado por el once inicial de cada equipo en los últimos *n* partidos.
- **passes_acc_11***S***:** Ratio de acierto en el pase del once inicial de cada equipo en los últimos *n* partidos.
- **keyPasses**_*S***:** Total de pases clave dados por el once inicial de cada equipo en los últimos *n* partidos. Los pases clave son pases que crean situaciones de peligro que acaban en una ocasión clara para el equipo.
- **ataque**_*S***:** Este es un indicador del poder en ataque de cada equipo en los últimos *n* partidos. Calculado a partir de una weighted average de los goles marcados en los últimos *n* partidos, en el cual el partido más reciente tiene más peso que el más antiguo.

[2]Attribute with "*S*" have two versions, one for each side team: home ("H") and away ("A")

- **defensa_*S*:** Este es un indicador de la consistencia en defensa de cada equipo en los últimos n partidos. Calculado a partir de una weighted average de los goles encajados en los últimos n partidos, en el cual el partido más reciente tiene más peso que el más antiguo.

Es importante destacar que las estadísticas de un equipo en los últimos n partidos sólo afectan a los partidos como local o visitante. Por ejemplo si en el partido Málaga vs Celta de Vigo, el sample contendrá los datos de los últimos n partidos del Málaga como local.

3.2.2. Historical dataset

a completar

3.3 Data analytics

Esta sección consiste en el analisis de los datasets a nivel estadístico para encontrar distribuciones, patrones y outliers y por tanto entender nuestros datos con el objetivo de crear los datasets, seleccionar características y entrenar los modelos.

3.3.1. Prior dataset creation analysis

El primer analisi se realizó antes de crear los datasets con el objetivo de encontrar patrones y anomalías en los datos que pudiesen darnos pistas sobre que características pueden influir más en el resultado de un partido y cuales seguramente no influyan en absoluto.

Draw	Home win	Away win
350	646	418

Table 3.6: Distribution of the match results.

Lo primero que interesa saber es cuantos samples de cada clase hay, para saber como de balanceado esta el dataset. Como se ve en la Table 3.6, el dataset está bastante desbalanceado ya que casi la mitad, un 45.7% de los partidos acabaron en victoria local, mientras que un 29.5% en victoria visitante, y tan solo un 24.8% en empate.

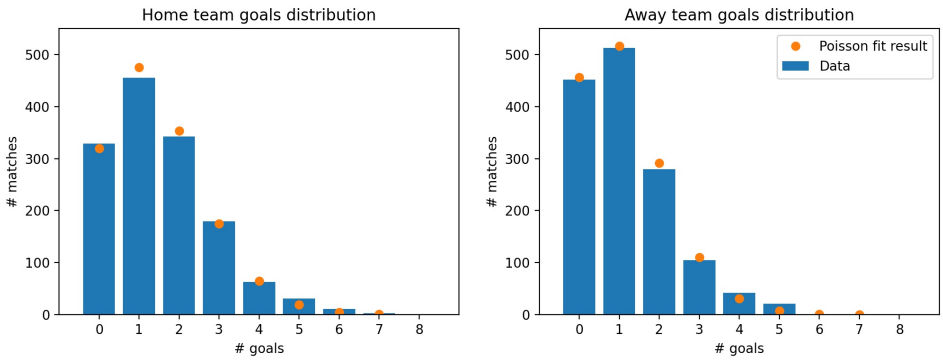

Figure 3.1: Goal distribution with fitted Poisson distribution

En la Figure 3.1 se muestra la distribución que siguen los goles anotados durante la temporada 2017-2018. En el caso que se considere a los goles como eventos que pueden suceder con una probabilidad determinada durante los 90 minutos, y se asumiera que los partidos entre sí son independientes y que los goles en un partido son indepedientes entre sí, se podria afirmar que sigue una distribución de Poisson. Como se aprecia en la Figure 3.1 ambas distribuciones se ajustan mucho a Poisson. Los goles locales siguen una distribución con $\lambda = 1.485$ y los visitantes con $\lambda = 1.13$. Otra conclusión que se puede sacar con las dos distribuciones es que el equipo local tiene ligeramente más probabilidad de marcar que el visitante. A partir de la PCF de Poisson, se calcula que el equipo visitante tiene casi un 70% de probabilidad de marcar ninguno o un gol, mientras que el local sólo un 56%.

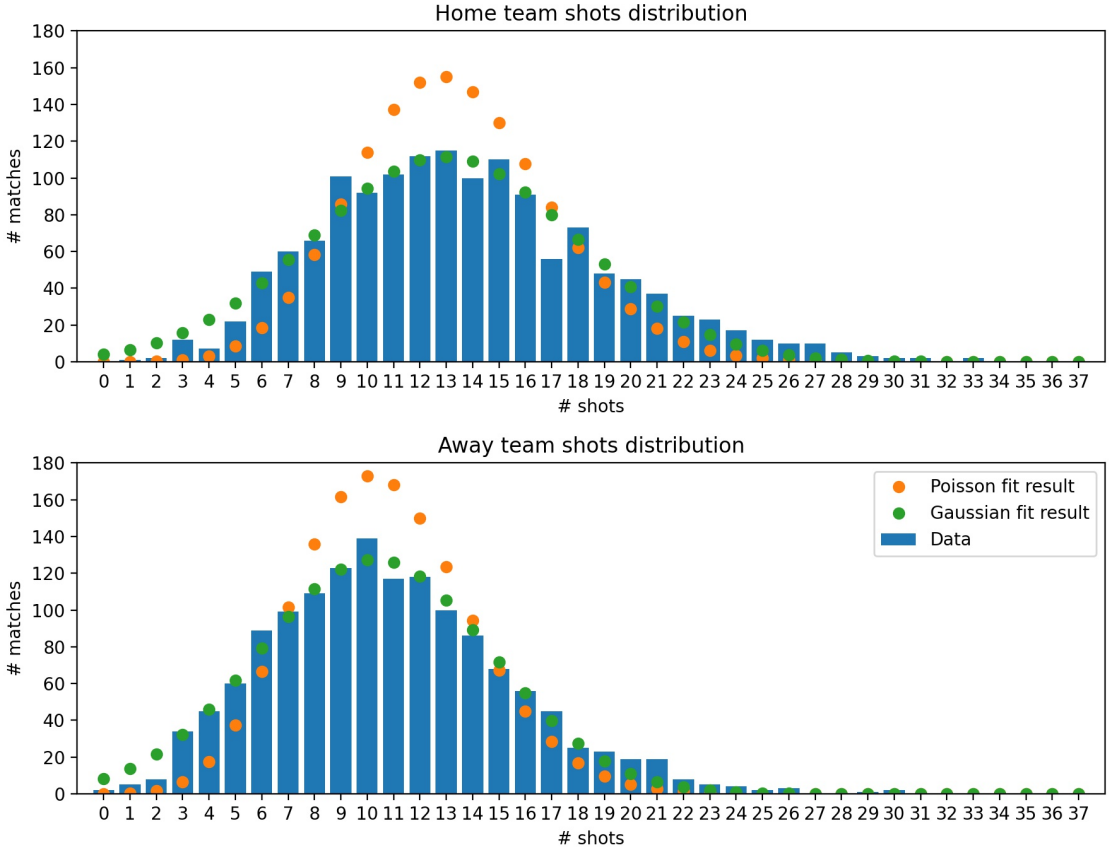

Figure 3.2: Shots distribution with fitted Poisson and Normal distribution

También es interesante conocer la distribución de la cantidad de disparos que se realizan en un partido. En la Figura 3.2 se aprecia que en este caso encaja mejor con una distribución Normal. Corresponden con $\mathcal{N}(\mu$=13.27,σ=5.27) y $\mathcal{N}(\mu$=10.86,σ=4.55) respectivamente. La conclusión que que se puede sacar de aquí es que, los equipos locales tienen más probabilidad de hacer más disparos que los visitantes.

La distribución de los disparos que hace el equipo ganador y el perdedor puede dar una idea (bastante intuitiva) de si existe una relación clara entre el numero de disparos y el resultado de un partido. Descartando los partidos acabados en empate estas son las distribuciones. Como vemos (Figure 3.3) la PDF de la distribución Normal sobreestima el numero de partido con muy pocos disparos (entre 0 y 4), mientras que infraestima los valores con más número de partidos (6, 7 o 9 disparos).

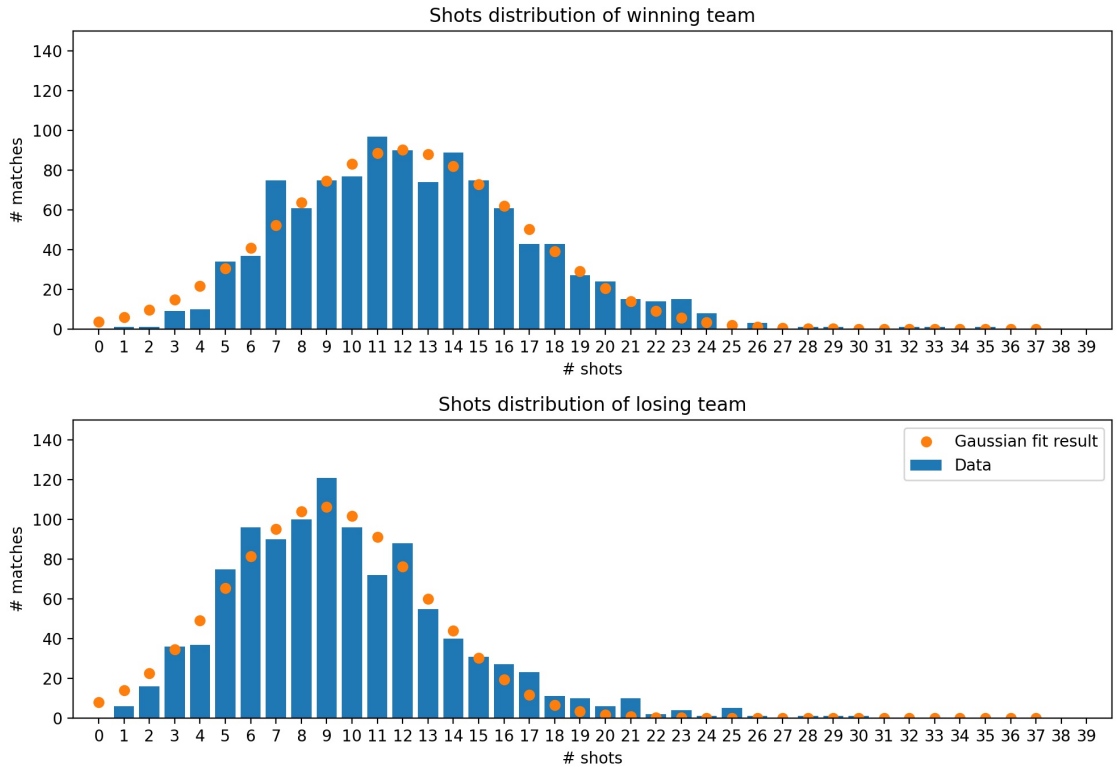

Figure 3.3: Shots distribution of winning and losing teams with fitted Normal distribution

Por último en cuanto a los disparos conviene comparar la distribución de la diferencia de disparos entre el equipo ganador y el perdedor. Hay dos escenarios aquí, cuando gana el equipo que más dispara y cuando ocurre lo contrario, que gana el equipo que menos disparos realiza (Figure 3.4).

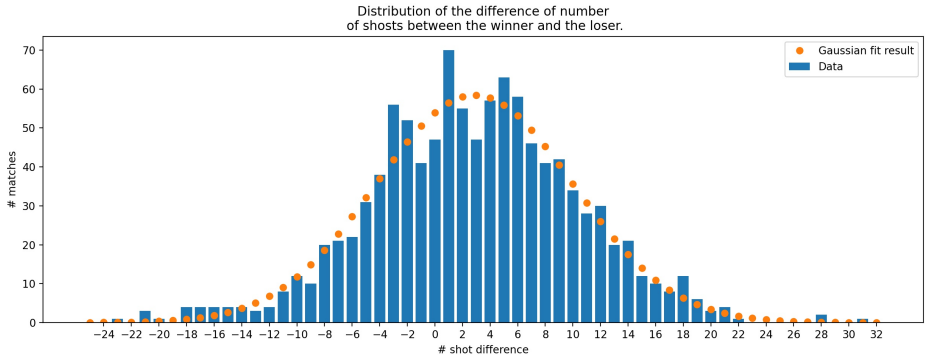

Figure 3.4: Shot distributions between winner and loser

Se puede observar que hay ligeramente más partidos en los que el equipo ganador hace más disparos, pero la diferencia no es muy grande, lo que signifa que hay muchos partidos que estan muy igualados sobre el cesped y que el resultado acaba venciendo hacia un lado u otro. No es raro que un equipo realice más de 10 disparos y acabe ganando, sin embargo hay partidos en que ocurre al contrario, estos casos seran especialmente difíciles de predecir para un modelo predictivo que tome como input una serie temporal de eventos [Nyquist and Pettersson, 2017]. En el 32% de los partidos acaba ganando el equipo

que realiza menos disparos.

Otro aspecto aspecto táctico fundamental en el futbol son los pases. En la siguiente figura (Figure 3.5) se aprecia como el equipo ganador suele hacer más pases durante el partido. En casi el 25% de los partidos el equipo ganador realiza entre aproximadamente 550 y 850 pases, mientras que el perdedor entre 450 y 650. Es cierto que la mediana y el primer cuartil (Q1) de cada distribución son practicamente los mismos, pero el tercer cuartil (Q3) y el rango superior de valores típicos varia mucho entre ambas distribuciones. La conclusión que se puede sacar es que si un equipo realiza más de 600-700 pases es muy poco probable que pierda el partido.

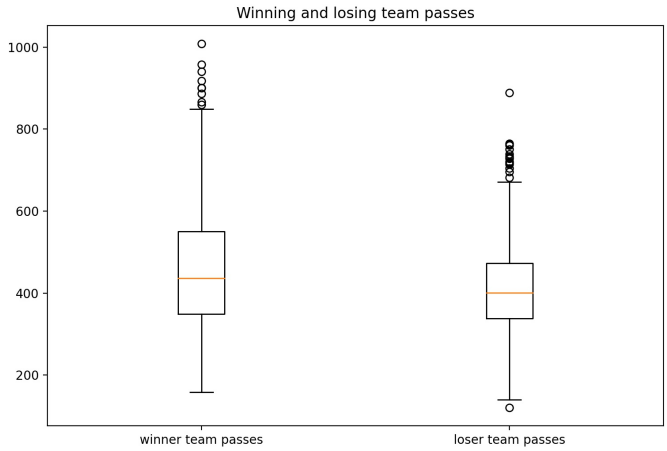

Figure 3.5: Winning and losing team passes box plot.

rank.	team	pos.	acc.
1	Manchester City	1º	0.9
2	PSG	1º	0.89
3	Nice	8º	0.89
4	FC Barcelona	1º	0.88
5	Bayern Munchen	1º	0.88
...			
94	Leganés	17º	0.78
95	Stoke City	14º	0.77
96	Augsburg	12º	0.77
97	Crotone	18º	0.76
98	Getafe	8º	0.73

Table 3.7: Top 5 teams with most pass accuracy and its position in the 2017-2018 league table and the top 5 teams with the less accuracy at the pass.

Pero no sólo es importante la cantidad de pases, sino la calidad de estos. En la Table 3.7 se puede ver como en el top cinco de equipos con más acierto de pase hay 4 campeones de liga. En los cinco equipos con menos acierto es más difícil sacar conclusiones ya que aunque son equipos de la parte media-baja de la tabla, sólo hay un equipo de ellos que descendió, el Crotone. Es curioso ver como el Nice y el Getafe finalizaron ambos en el octavo puesto de sus respectivas ligas. El hecho de que uno sea el tercero con más acierto y el otro sea el que menos (con bastante diferencia del resto) se debe al estilo de

juego de ambos equipos, mientras el Nice apuesta más por la posesión y el control del juego (al igual que los estilos de juego de Manchester City o Barcelona), el Getafe tiene un juego más defensivo, agresivo y de juego a la contra. Se ve por tanto cómo se puede inferir el estilo de juego de un equipo a partir de algunas de sus estadísticas.

rank.	team	goals	acc.
1	FC Barcelona	99	0.48
2	Hertha Berlin	43	0.44
3	PSG	108	0.42
4	Real Betis	60	0.42
5	Manchester City	106	0.41
...			
94	Sassuolo	29	0.3
95	Amiens	37	0.29
96	Nantes	36	0.28
97	Hellas Verona	30	0.28
98	Caen	27	0.27

Table 3.8: Top 5 teams with more and less shot accuracy and the total amount of goals they scored in 2017-2018.

En la Table 3.8 se muestran los equipos con más y menos acierto de cada a portería y los goles que han anotado durante la temporada. Se puede observar como los equipos más anotadores tienen mucho acierto en el disparos, lo que significa que tienen muy buenos delanteros. Destaca que aparezca el Hertha de Berlin en segunda posición. Los equipos con menos acierto son también los equipos que menos goles marcan.

A continuación, se muestra la relación entre las cuotas de apuestas de *BET365* y el resultado de un partido. Con esto se pretende ver si las cuotas esconden una relación directa con la probabilidad de victoria de un equipo, y el grado de esta relación. Realmente las cuotas representan la inversa de la probabilidad que le asigna la casa de apuestas a un resultado. Por eso en la siguiente Figure 3.6 se veran representada las inversas de las cuotas.

Como se observa en la Figure 3.6 las casas de apuestas aciertan la mayoria de partidos en los que hay una alta probabilidad de que gane uno u otro equipo, a partir de una probabilidad de 0.7 aciertan casi todos los partidos. La recta con función $f(x) = 1 - x$ representa la probabilidad 1, si se compara con los puntos de la curva de probabilidades de *BET365* se encuentra que la distancia entre la curva y los puntos es la probabilidad que le asigna implicitamente la casa de apuestas al empate. Es a tener en cuenta que las casas de apuestas siempre asignan una probabilidad mayor a la victoria de uno de los equipos antes que al empate cometiendo muchos errores en la zona media de la gráfica donde estan los partidos más igualados.

3.3.2. New dataset analysis

En esta subsección se analiza estadísticamente algunos de los atributos generados a partir de los datasets originales y se compara con los atributos orignales. En primer lugar, si se compara las distribuciones que siguen los atributos generados *shots_11H* y *shots_11A* con las distribucies de los disparos en cada partido (Figure 3.2) se aprecia que ambas distribuciones tienen formas iguales. Esto es debido a que dichos atributos generados son

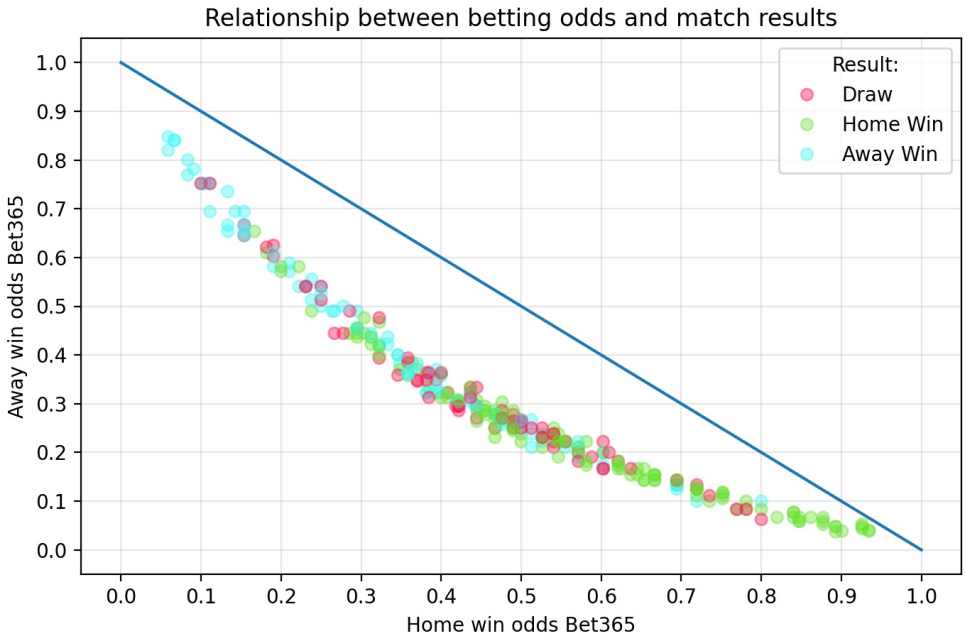

Figure 3.6: Bet365 probabilities based on betting odds.

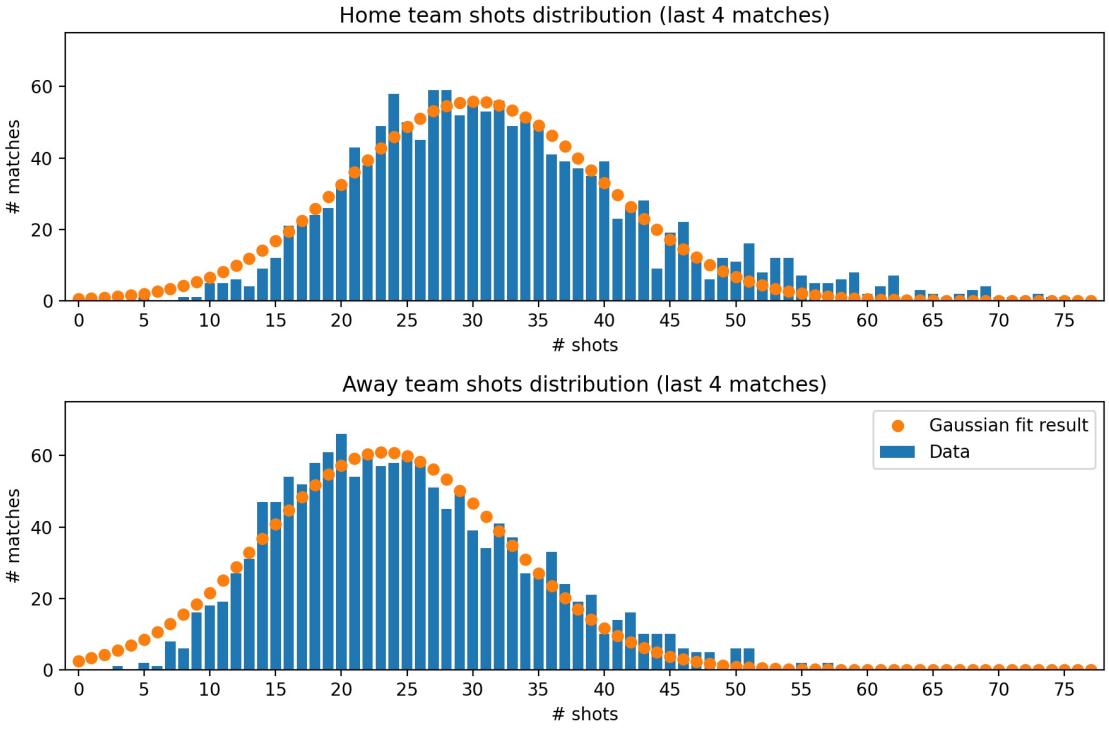

Figure 3.7: *shots_11H* y *shots_11A* distribution and fitted with Gaussian distribution.

la suma de los disparos de los jugadores del line-up de un equipo en los últimos 4 partidos como local o visitante (Figure 3.7), por lo que es lógico que siguen distribuciones semblantes.

Dos atributos totalmente nuevos generados en el nuevo dataset han sido *mins4_H* y *mins4_A*. En la Figure 3.8 se puede observar que siguen una distribución que se puede

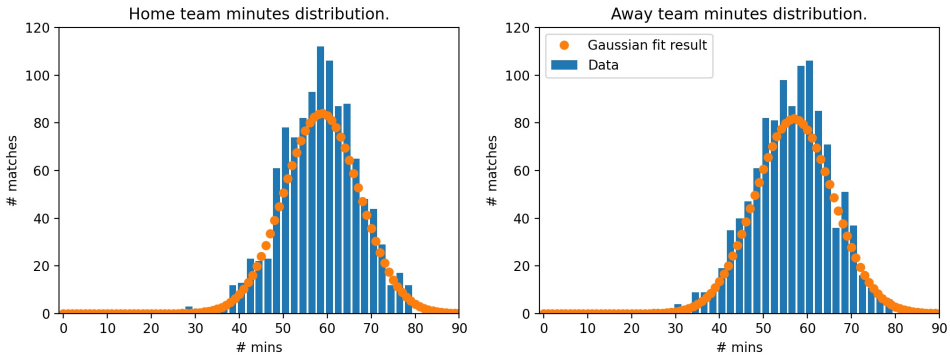

Figure 3.8: *mins4_H* y *mins4_A* distribution and fitted with Gaussian distribution.

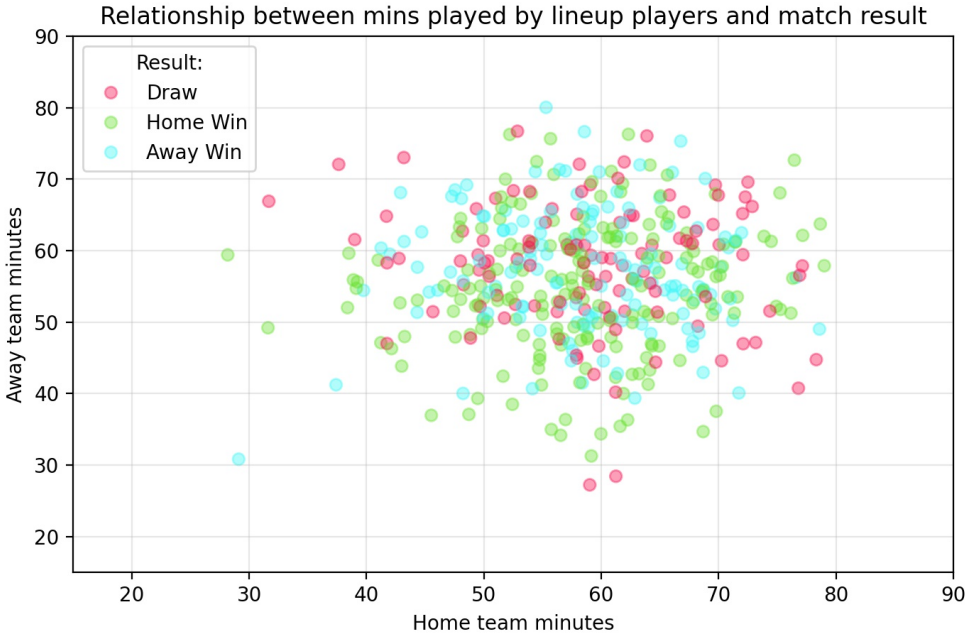

Figure 3.9: Relationship between *mins4_H* and *mins4_A* with match results.

ajustar bastante a una Normal sobretodo en las "laderas" de la PDF de la distribución, ya que en la zona de la media (el "pico") se ajusta peor y la distribución infraestima esos valores. En la Figure 3.9 se muestra la relación entre ambos atributos y el resultado del partido dando como conclusión que no parece que exista una relación evidente entre ambas variables por lo que parece que son independientes. Recordemos, que estos atributos su propósito es indicar al modelo si un equipo juega con su once habitual o con nuevos jugadores, factor que es posible que influya en el resultado de un partido.

En la Table 3.9 se muestra el indicador creado para medir la fortaleza en ataque y la debilidad en defensa de cada equipo. Para el modelo usaremos dos atributos separados para cada equipo, *ataque_H o A* y *defensa_H o A* que utilizaran los goles marcados y encajados respectivamente para indicarlo. Sin embargo, en esta table se muestra el ranking del indicador (unificando ataque y defensa) considerando todos los partidos de la temporada, con el objetivo de mostrar cómo este indicador muestra muy bien el nivel y los resultados de los equipos.

rank.	team	pos.	att.-def.
1	PSG	1º	2.02
2	Juventus	1º	1.99
3	Bayern Munchen	1º	1.98
4	Manchester City	1º	1.94
5	FC Barcelona	1º	1.83
...			
94	Huddersfield Town	16º	0.69
95	Málaga	20º	0.68
96	Hellas Verona	19º	0.66
97	Caen	16º	0.65
98	UD Las Palmas	19º	0.57

Table 3.9: Top 5 teams with more and less atack-defense indicator at season 2017-2018.

3.4 Dimensionality reduction

Como parte de la ingeniería de características y el análisi de estas, a continuación se explica como se ha realizado el proceso de reducción de dimensionalidad de los datos. La reducción de dimensionalidad se encarga de identificar, seleccionar y eliminar los atributos que no aportan o incluso disminuyen el rendimiento del modelo. Se muestra a continuación dos tipos principales de técnicas para llevarlo a cabo. Estas técnicas seran aplicadas al dataset de 20 atributos creado a partir de los datos del Wyscout based dataset.

3.4.1. Feature selection

Existen numerosas técnicas de selección de características pero nosotros vamos a hacer uso de tres técnicas, una técnica de analisi univariado, *Variance Threshold*, y otras tres de analisi multivariado *ANOVA* y *Pearson correlation*. Antes de aplicar las técnicas de selección de características se normalizan los datos para que esten todos en la misma escala y podamos sacar conclusiones precisas y acertadas. La normalización se ha realizado con la función *normalize* del paquete *preprocessing* de la librería de Machine Learning *Scikit-Learn*.

Variance Threshold El primer método de selección de características usado y el más simple es este. Variance Threshold se suele utilizar para comparar con métodos más complejos, y usa un threshold para descartar todas esas variables cuya varianza no lo supere. Con esto se reduce el espacio de los datos de una dimensión D a un espacio de dimensionalidad d, con $D > d$. La varianza mide el spread de los valores de una variable, mide como de lejos está un valor de la media de la variable. El uso de esta técnica se usa con el enfoque de entrenar el modelo con las variables que tengan más varianza, ya que posiblemente al estar los datos más separados en el espacio de dimensión d aumente la separabilidad de los datos respecto a un espacio de las mismas dimensiones pero con variables con menos varianza.

Durante el entrenamiento de nuestro modelo se utilizaran diferentes umbrales para determinar que threshold es el que selecciona, según la varianza, el numero de variables óptimo. En la Figure 3.10 se muestran las varianzas de las variables normalizadas del dataset generado basado en los datos de Wyscout.

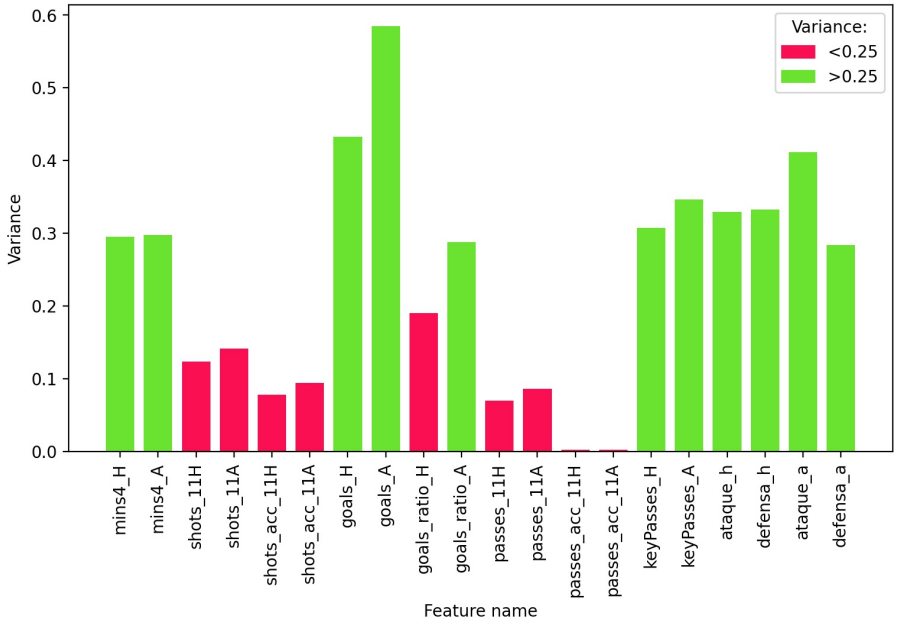

Figure 3.10: Normalized feature variances

De la Figure 3.10 se pueden sacar varias conclusiones. Primero, que posiblemente el ratio de acierto del pase no sea una característica que pueda aportar demasiado a nuestro modelo debido a que no existe cuasi varianza en sus datos. En la Table 3.7 ya se podía intuir que esto era así debido al pequeño rango de los datos, entre 0.73 y 0.9. Ocurre algo similar con los disparos, los pases y el acierto de los disparos, aunque en este caso sí que hay algo de varianza y habrá que ver el resultado de otros métodos para sacar conclusiones. Por otra parte, los goles son las variables con mayor varianza, y por tanto es posible que sea la variable que más ayude a nuestro modelo a saber que equipos estan en mejor forma.

En la Figure 3.10 se muestran en verde las características que esta técnica selecciona con un umbral de 0.25, en total selecciona 11 atributos. Selecciona por lo general en parejas, por ejemplo de los goles selecciona tanto los del equipo local como visitante, ya que las distribuciones que siguen son bastante similares como hemos visto anteriormente (Figure 3.7).

ANOVA **An**alysis **o**f **Va**riance is a parametric statistical hypothesis test for determining whether the means from two or more samples of data come from the same distribution or not. ANOVA muestra la relación entre una variable categórica y una continua, es decir si una depende de la otra. Más concretamente, muestra como cambian las variables contínuas con respecto a la variable categórica. En este caso, asumiendo la independencia de las muestras, se usará One-Way ANOVA para encontrar las relaciones entre las características (variables contínuas) y la clase o resultado (una sóla variable categórica).

En la Figure 3.11 se muestran las p-values resultado de la aplicación de ANOVA a nuestros datos. Valores cercanos al 1 significan una dependencia entre la variable y el resultado del partido (categorical feature), mientras que valores cercanos a zero muestran independencia. Estas variables independientes decrementaran el rendimiento de nuestro modelo al no aportar información para clasificar el partido.

El resultado de ANOVA muestra que sorprendentemente las características de los equipos

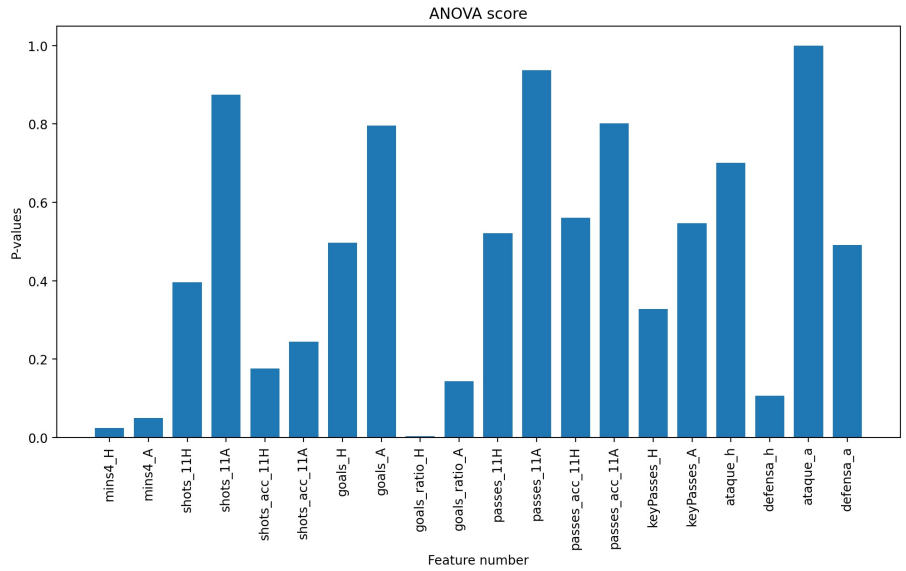

Figure 3.11: ANOVA F-value score showing the dependance between continues variables and categorical target variable.

visitantes varían más dependiendo el resultado del partido. No se ha conseguido encontrar una explicación a esto por el momento. Otra observación sorprendente es la dependencia entre el resultado y el acierto en el pase. Se vió en el Variance Threshold (Figure 3.10) como estas variables tenían una varianza cercana a zero. Fuerte dependencia también de los resultados con los disparos, los goles, los pases totales, los pases clave y el indicador de ataque en los últimos cuatro partidos. Posiblemente se debe a que hay también una fuerte dependencia entre los pases claves, los disparos, los goles y el indicador de ataque. Destaca también la muy poca dependencia con los minutos jugados por los jugadores en los últimos cuatro partidos, como ya se vió en el análisi del dataset (Figure 3.9), y con el ratio de goles marcados en los últimos cuatro partidos por los jugadores del once inicial.

Para seleccionar las características más dependientes con la target feature hemos usado la clase *SelectKBest* del paquete *feature_selection* de la libreria *Scikit-Learn*, y hemos entrenado ANOVA con nuestros datos. La cantidad de variables que se quieran seleccionar se indican con el parámetro k que se pasa como parámetro a la inicialización del objeto *SelectKBest*. Este parámetro k se entrenará con nuestro modelo para identificar el número óptimo de variables.

Pearson Correlation The Pearson correlation coefficient measures the linear relationship between two continuous variables. This one varies between -1 and +1, with 0 implying no correlation. Correlations of -1 or +1 imply an exact relationship. Positive values imply that as x increases, so does y, they are positive correlations. Negative values imply that as x increases, y decreases, and they are negative correlations. The Pearson (ρ) is essentially a normalized measurement of the covariance, it is the ratio between covariance of two variables ($cov(X, Y)$) and the product of their standard deviations (σ).

$$\rho_{X,Y} = \frac{cov(X, Y)}{\sigma_X \sigma_Y} \tag{3.1}$$

Se han comparado todos los atributos entre sí para encontrar dependencias entre dos

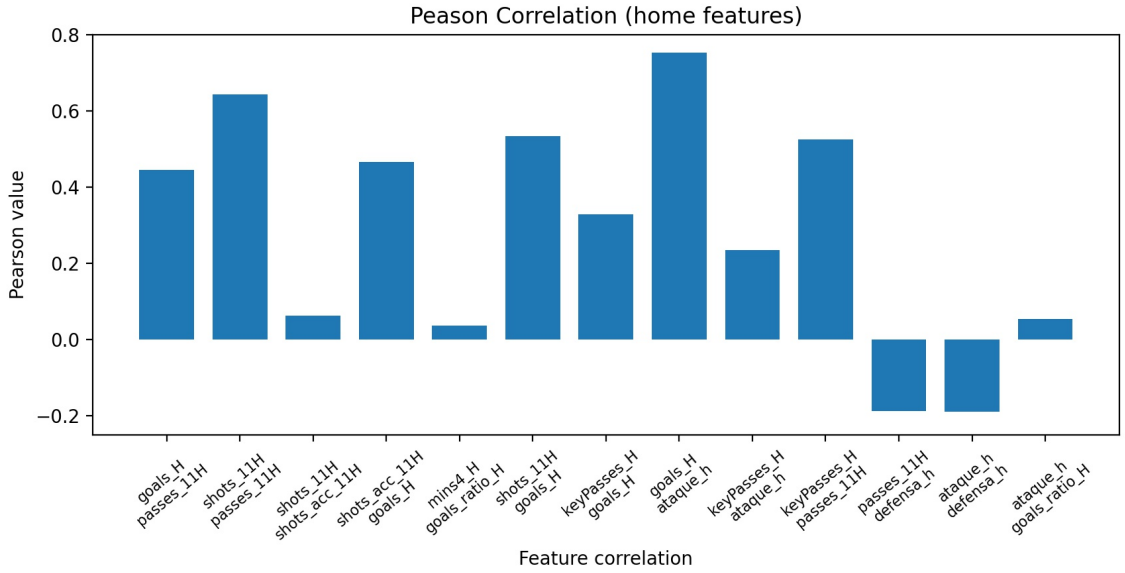

Figure 3.12: Pearson correlation between some home team features of the last four matches as home team.

características para ver si es posible descartar variables debido a una alta dependencia con otra. Estas son las dependencias más y menos fuertes (Table 3.10):

Highest dependency			
rank	**var 1**	**var 2**	**p-value**
1	mins4_H	mins4_A	0.9
2	goals_A	ataque_A	0.77
3	goals_H	ataque_H	0.75
4	passes_11A	passes_acc_11A	0.68
5	passes_11H	passes_acc_11H	0.65

Lowest dependency			
rank	**var 1**	**var 2**	**p-value**
1	goals_H	passes_11A	1.0e−5
2	goals_A	ataque_H	1.97e−4
3	shots_acc_11A	goals_ratio_H	1.1e−3
4	shots_acc_11H	goals_A	1.2e−3
5	shots_acc_11A	passes_11H	3.3e−3

Table 3.10: Pearson correlations.

Se puede observar en la Table 3.10 la fuerte dependencia entre las variables de los goles y el ataque, tal y como se había supuesto en el analisi de ANOVA (Figure 3.11). En la Figure 3.12 se comparan los p-values de varias variables del equipo local. Se puede apreciar también la fuerte dependencia entre pases, pases clave, disparos y goles. También se observa la suave dependencia negativa entre los pases y el indicador de defensa, lo que significa que cuando un equipo hace muchos pases es probable que reciba pocos goles. Pearson destaca también la independencia de algunas variables como podemos ver tanto en la figura como en la tabla, por ejemplo indepdencia entre el numero de disparos y el acierto de cara a puerta, los minutos que han jugado los titulares y el ratio de goles que ha marcado el equipo en los últimos cuatro partidos. En la tabla de menores dependencias

(Table 3.10) se puede ver, como se podía intuir, las varuables más independientes son las de un equipo con las variables del otro equipo.

De este analisi se pueden descartar algunas variables interdependientes, tomando en cuenta también los análisis de Variance Threshold y ANOVA, y suponiendo que eso mejorará el rendimiento del modelo. Por ello si tomásemos una selección de variables a partir de lo analizado tomaríamos estas doce variables:

shots_11H	shots_11A	shots_acc_11H	shots_acc_11A	goals_H	goals_A
passes_11H	passes_11A	ataque_h	defensa_h	ataque_a	defensa_a

Table 3.11: Features selected manually based on intuition and feature selection analysis

El rendimiento del modelo con estas variables seleccionadas será comparado con el rendimiento de las variables seleccionadas con los otros métodos de reducción de dimensionalidad.

3.4.2. Principal Component Analysis (PCA)

Esta técnica de reducción de la dimensionalidad de los datos consiste en un enfoque diferente a las técnicas de selección de características. Es una técnica de *unsupervised learning* en la que se reduce la dimensionalidad de unas muestras calculando los n primeros componentes principales, con los que se transforman los datos originales a un espacio dimensional menor pero conservando la mayor información posible contenida en los datos originales.

The principal components de un conjunto de muestras o puntos en un espacio de dimensión d son un conjunto de d vectores unitarios, en los que el vector i-th indica la dirección de la recta sobre la cual si los puntos son proyectados mantienen la varianza máxima posible de los puntos proyectados y a su vez el vector es ortogonal a los anteriores $i-1$ vectores unitarios. Por lo tanto el problema al que se enfrenta PCA es el mapeo de los datos en un espacio de menos dimensiones, maximizando la varianza de los datos proyectados en dicho low-dimensional space. Es decir se pretende reducir el número de características con la mínima perdida posible de información. También se puede reducir ruido existente en los datos originales.

PCA se suele explicar e implementar con el cálculo de los eigenvectors (matrix $\mathbf{V}$) y eigenvalues (diagonal matrix, $\mathbf{L}$) de la matriz de covarianzas de los datos (matrix $\mathbf{C}$) que se pretenda transformar. Los primeros d eigenvectors corresponden a los n-th principales componentes, a los cuales se proyectan los datos (matrix $\mathbf{X}$) obteniendo así los datos representados en d dimensiones (Eq. 3.4). Se puede calcular el porcentaje de varianza conservado en la transformación, este corresponde con la suma de los primeros d eigenvalues dividido entre la suma de todos los eigenvalues.

$$\mathbf{C} = \mathbf{X}^\top \mathbf{X}/(n-1), \tag{3.2}$$

$$\mathbf{C} = \mathbf{V}\mathbf{L}\mathbf{V}^\top, \tag{3.3}$$

$$\mathbf{Y} = \mathbf{X}\mathbf{V}, \tag{3.4}$$

Sin embargo en este caso se ha hecho uso de una implementación de PCA basada en SVD. Si aplicamos SVD de $\mathbf{X}$, obtenemos:

$$\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^\top, \tag{3.5}$$

donde $\mathbf{U}$ es una matriz unitaria y $\mathbf{S}$ es la matriz diagonal de valores singulares. Sustituyendo en la matriz de covarianza obtenemos:

$$\mathbf{C} = \mathbf{V}\mathbf{S}\mathbf{U}^\top\mathbf{U}\mathbf{S}\mathbf{V}^\top/(n-1) = \mathbf{V}\frac{\mathbf{S}^2}{n-1}\mathbf{V}^\top, \tag{3.6}$$

lo que significa que la matriz derecha de vectores singulares ($\mathbf{V}$) son los eigenvectors y principal directions, y los eigenvalues estan relacionados con los valores singulares ($\mathbf{S}$) [Shlens, 2014]. Los principal components, es decir la proyección de los datos en su nuevo espacio viene dado por:

$$\mathbf{X}\mathbf{V} = \mathbf{U}\mathbf{S}\mathbf{V}^\top\mathbf{V} = \mathbf{U}\mathbf{S} \tag{3.7}$$

En los experimentos se entrenará el hiperparámetro d de las dimensiones a las que reducir el espacio, para así obtener el espacio dimensional óptimo en el que mapear los datos.

CHAPTER 4
Forecasting model

En este capitulo se describen los modelos implementados y los experimentos realizados para entrenarlos y evaluarlos. Primero se explicaran los experimentos realizados del modelo baseline. A continuación se mostraran los resultados de los modelos Multi-Layer Perceptron y se analizara el aprendizaje de estos modelos, se explicará la optimización de sus hiperparámetros y se describiran y compararán sus resultados con el modelo baseline. Finalmente, se diseñará un modelo Bagging a partir de los mejores modelos Multi-Layer Perceptron y nuevos modelos, se realizarán experimentos y se comparará su resultado con los anteriores modelos. Todos los experimentos se realizan con los mismos datos de entrenamiento, validación y test.

4.1 Baseline models

Un modelo baseline es un modelo prototipo, muy simple, que sirve como referencia para evaluar el rendimiento de modelos más complejos. En este caso el modelo baseline recibe como input las cuotas de empate, home-win y away-win de la casa de apuestas *Bet365*. A partir de este input, el modelo deberá predecir el resultado de los partidos. Más tarde se compararan los resultados de modelos más complejos con este modelo, lo que servirá para ver si los nuevos modelos rinden como se espera, ya que si un nuevo modelo iguala o supera el acierto del baseline significará que tiene una mejor capacidad de predicción que las probabilidades que asignan las casas de apuestas.

Se han implementado diversos modelos baseline basados en diferentes algoritmos muy conocidos de Machine Learning, como SVM, Random Forest o MLP. Es importante destacar que las cuotas han sido transformadas a probabilidades (entre 0 y 1), simplemente haciendo la inversa de la cuota.

Baseline I: max odds Este primer baseline es el más sencillo. Simplemente consiste en la predicción que hace *Bet465*, ya que el modelo escoge la máxima probabilidad entre empate, home-win y away-win. Como ya mencionamos en el analisi de las cuotas (Figure 3.6) las casas de apuestas nunca predicen que el partido finalizará en empate, ya que siempre asignan una probabilidad mayor a la victoria de uno de los equipos. El acierto obtenido ha sido del 54.45% y esta es la confusion matrix de los resultado:

Baseline II: SVM El segundo baseline consiste en un modelo SVM entrenado con 3-fold Cross-Validation y los hiperparámetros se han optimizado con una búsqueda randomizada de hiperparámetros, usando la clase de *Scikit-Learn*, *model_selection.RandomizedSearchCV*.

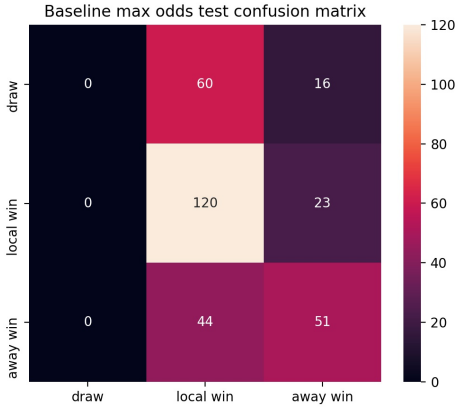

Figure 4.1: Test evaluation of max-odds baseline confusion matrix.

Esta clase ya incorpora la opción del entrenamiento con CV. Los hiperparámetros optimizados han sido el hiperparámetro de regularización de SVM (C), el *kernel* usado en el SVM (linear, polynomial or rbf), el hiperparámetro *gamma*, que es el kernel coefficient para los RBF-SVM y polynomial SVM, y, por último, el hiperparámetro *degree* que indica el grado del polinomio en el kernel polinomial.

El modelo con mejores resultados ha obtenido un acierto en el test del 54.78% con $C = 2.99$, *kernel* =rbf y $gamma = 0.75$.

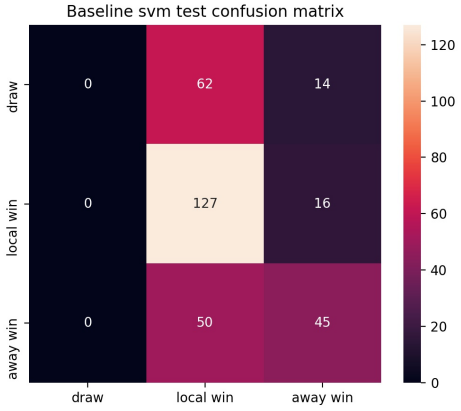

Figure 4.2: Test evaluation of Support Vector Machine based baseline confusion matrix.

Baseline III: Random Forest El baseline basado en un Random Forest también ha sido entrenado con 3-fold CV, y los hiperparámetros optimizados con random search han sido el número de estimadores (árboles de decisión), profundidad máxima de los árboles, el mínimo de samples necesarios para hacer un *split* en el árbol, el mínimo de samples para poder crear una hoja y el máximo de hojas que pueden tener los árboles. Todos estos parámetros se optimizan para evitar el overfitting y obtener la máximo acierto posible en el test.

El modelo Random Forest con mejores resultado ha obtenido un acierto en el test de

54.73%, con 112 árboles de decisión, 96 partidos mínimo para crear un *split*, 4 samples mínimo por hoja y un máximo de 5 hojas por árbol.

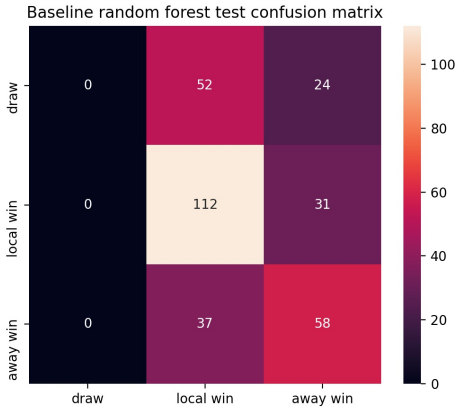

Figure 4.3: Test evaluation of Random Forest based baseline confusion matrix.

Baseline IV: MLP Por último se ha creado un baseline basado en Multi-layer Perceptron con una hidden layer de tres units. Como los anteriores modelos, también ha sido entrenado con 3-fold CV, y en este caso no se ha realizado una optimización de los hiperparámetros. Los hiperparámetros usados han sido Cross Entropy como función de error, SGD como optimizador con un learning rate de 0.1, 100 epochs y un batch-size de 32. Se ha obtenido un acierto del 54.78%, ligeramente superior a los demás modelos, excepto a SVM, que lo iguala.

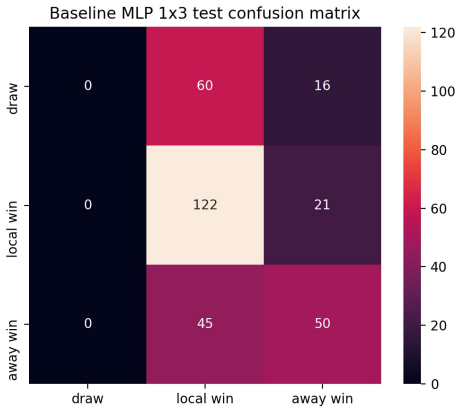

Figure 4.4: Test evaluation of Multi-Perceptron based baseline confusion matrix.

Estos resultados de porcentaje de acierto en el test trataran de ser superados por modelos más complejos basados de redes neuronales. Según las matrices de confusión, todos los modelos se comportan de una manera muy parecida, predeciendo mucho más victorias locales que el resto de resultados posibles, lo cual tiene sentido según la distribución que seguían los resultados. Sin embargo, el notable recalcar que ningún modelo predice ningún empate, posiblemente por lo desbalanceado que está el dataset (Table 3.6).

	Max-Odds	SVM	Random Forest	MLP
%	54.46	54.78	54.73	54.78

Table 4.1: Baseline models test accuracy.

4.2 Multi-Layer Perceptron models

En esta sección se proponen diferentes Multi-layer Perceptron modelos con arquitectures sencillas pero que se espera que superen el rendimiento de los modelos de baseline. Cada modelo es entrenado múltiples veces con distintas combinaciones de hiperparámetros, usando la conocida técnica de Grid Search, con el objetivo de encontrar los hiperparámetros óptimos para cada modelo.

A continuación se mostraran los resultados, y se hará un analisi del aprendizaje de los mejores modelos para ver como se pueden mejorar. Se seguirá un proceso en espiral, ya que se empezará con modelos muy simples y tras analizar sus resultados y dependiendo su aprendizaje, se continuará con modelos más complejos, sin embargo sólo se pasará a un nivel de complejidad mayor si es la solución dado el analisi de los modelos más simples.

El entrenamiento basado en Grid Search ha buscado la optimización de los siguientes hiperparámetros:

Architecture La arquictectura de un modelo basado en redes neuronales es fundamental, y consiste en definir el numero de capas y de neuronas por capa que tiene la red. La arquitectura influye en la rapidez del aprendizaje de la red y también en como aprende, cuanto más profunda sea la red aprenderá más detalles, sin embargo también es más probable que overfit the data. Es por ello que la arquitectura de una red depende del tamaño del dataset, del número de características de los datos y de la forma de estos.

En estos experimentos se ha empezado con una arquitectura de una hidden layer con cinco neuronas (Figure 4.5), se han analizado los resultados y su aprendizaje, y a continuación se ha experimentado con otras arquitecturas, algunas más profundas, para mejorar el rendimiento y el aprendizaje de la red.

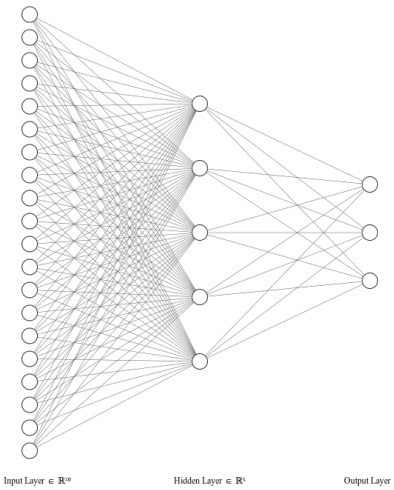

Figure 4.5: Simple Multi-layer Perceptron of 20 input units, 5 hidden units and 3 output units.

En la table de resultados (Table 4.2), se muestran los mejores resultados de cada modelo. Respecto a la arquitectura se puede ver como los modelos de la primera arquitectura tienen un rendimiento considerablemente mejor, especialmente cuando utilizamos una de las tres técnicas de reducción de dimensionalidad (ANOVA, Var. Threshold or PCA). En estos casos se obtiene entre un 5% y 10% más de acierto respecto a los modelos de 3 y 10 hidden neurons. Sin embargo en el caso de utilizar todas las características o las 12 seleccionadas manualmente (Table 3.11) el modelo tiene un acierto muy parecido o incluso mayor con las otras arquitecturas de una capa oculta.

En el caso de la arquitectura de 1 hidden layer con dos neuronas, es posible que su bajo rendimiento sea debido a los pocos parámetros que tiene esta red para optimizar, y por tanto su poca flexibilidad. Esto se ve sobretodo cuando se pasa como input tan sólo dos o cinco caracterísiticias por sample, donde la red neuronal tiene 27 y 36 parámetros respectivamente. Usando más cracterísticas se obtiene entre un 5% más de acierto, y hasta un 10% sin reducción de dimensionalidad. Posiblemente por la mayor flexibilidad de la red neuronal.

A partir de los resultados obtenidos con la red de una capa oculta con cinco neuronas se decidió entrenar capas más complejas, con más parámetros y más profundas, dando así más flexibilidad a la red y que pudiese acoplarse mejor a los datos. Con una red con 20 input nuerons y una capa oculta con diez neuronas pasamos a tener 263 parámetros, lo cual es prácticamente el doble que con las cinco hidden neurons (133 parameters). Sin embargo, los resultados obtenidos en el test no indican una mejora del rendimiento de la red. Esto posiblemente es debido a que existe overfitting de los datos, debido a que tenemos muy pocas muestras para el entrenamiento. Es por ello que con este modelo, si se usan más características se obtiene un mejor resultado (Table 4.2).

Input data

Optimizer En la siguiente figura se muestran los resultados obtenidos según el optimizador usado.

Loss function

Learning rate

Epochs

Batch size

Activation function El primer modelo (Figure 4.5) que hemos entrenado se basa en la arquitectura del modelo propuesto por Ogunseye et al. [Ogunseye et al., 2019] de una sóla capa oculta con cinco hidden units. En nuestro caso el problema de clasificación es de 3 clases y nuestro input inicial es de 20 features, pese a que este modelo se entrenará también con los datasets con dimensionalidad reducida explicados en la sección de Dimensionality Reduction.

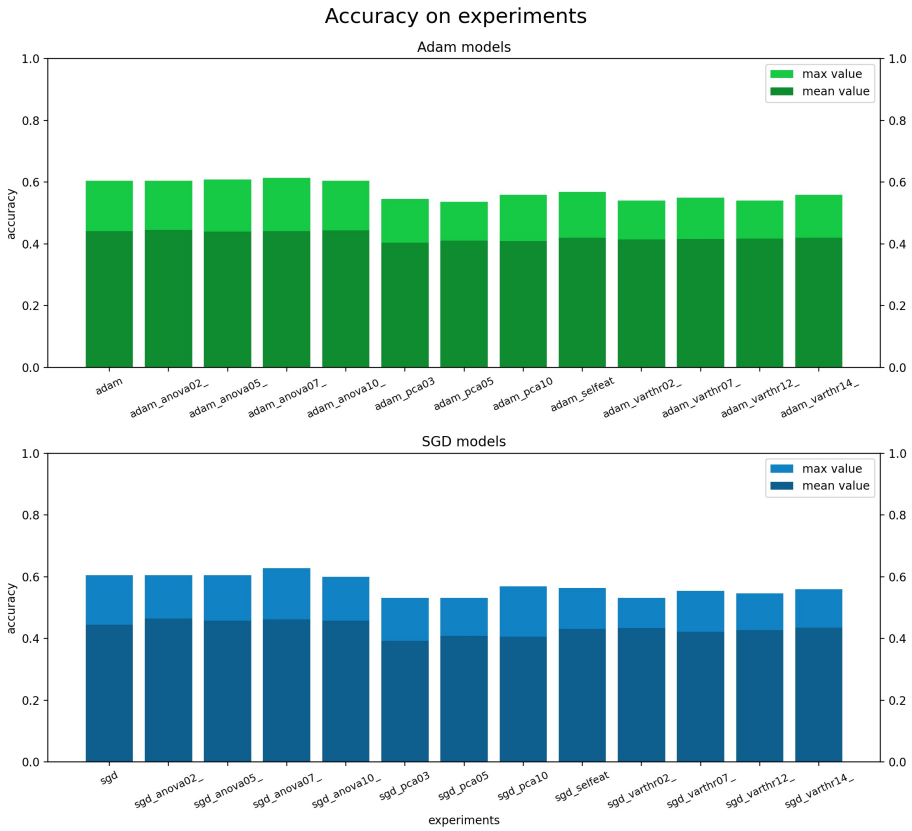

Figure 4.6: POR CAMBIAR

4.3 Bagging model

????? ????????????? ????????????? ????????????? ????????????? ?????????????

Epochs:			Adam			SGD		
			5	**20**	**100**	**5**	**20**	**100**
MLP 1 layer 5 units	20 features		50.82	49.45	56.36	50	50.64	55.73
	sel. features		51.45	52.18	50.45	50.36	51.36	51.45
	ANOVA	2	57	56.55	56.45	56.36	56.55	56.18
		5	56.82	56.55	56.45	55.18	56.91	57
		7	56.64	**57.36**	56.64	55.18	57.18	57.18
		10	57	56.27	56.45	56.36	56.45	56.2
	Thres. Varia.	2	-	-	48.09	-	-	48.36
		7	-	-	48.36	-	-	48.64
		12	-	-	49.91	-	-	49.73
		14	-	-	49.73	-	-	50
	PCA	3	-	-	47.18	-	-	45.73
		5	-	-	47.27	-	-	47.09
		10	-	-	48.18	-	-	48.27
MLP 1 layer 3 units	20 features		50.55	52.27	55.36	48.64	51.09	55.27
	sel. features		51	51.73	51.27	49	51	51.55
	ANOVA	2	45.82	46.18	45.82	47.91	49.27	49.36
		5	45.82	46.36	46.09	47.82	49.73	50.64
		7	51.82	51.73	52	48.64	51.45	51.82
		10	51.45	51.82	51.55	50.45	51.82	52.18
MLP 1 layer 10 units	20 features		51.55	51.82	54.73	50.55	51.82	54.55
	sel. features		51.73	50.91	50.18	51	51.36	51.45
	ANOVA	2	45.91	45.82	46.09	48.45	49.27	48.82
		5	46.18	46	45.91	49.36	50	49.18
		7	51.92	51.64	51.27	50.82	51.36	51.64
		10	51.27	52.18	51.82	51.09	51.27	51.55
MLP 2 layers 5 units	20 features							
	sel. features							
	ANOVA	2						
		5						
		7						
		10						

Table 4.2: Multi-layer based model testing results.

CHAPTER 5
Conclusions

????? ????????????? ????????????? ????????????? ????????????? ?????????????

5.1 Difficulties and limitations during my thesis

5.2 Future work

Bibliography

[Bialkowski et al., 2014] Bialkowski, A., Lucey, P., Carr, P., Yue, Y., and Matthews, I. (2014). Win at home and draw away: Automatic formation analysis highlighting the differences in home and away team behaviors. In *Proceedings of 8th annual MIT sloan sports analytics conference*, pages 1–7. Citeseer.

[Bunker and Thabtah, 2019] Bunker, R. P. and Thabtah, F. (2019). A machine learning framework for sport result prediction. *Applied computing and informatics*, 15(1):27–33.

[Bush et al., 2015] Bush, M., Barnes, C., Archer, D. T., Hogg, B., and Bradley, P. S. (2015). Evolution of match performance parameters for various playing positions in the english premier league. *Human movement science*, 39:1–11.

[Fernandez-Navarro et al., 2016] Fernandez-Navarro, J., Fradua, L., Zubillaga, A., Ford, P. R., and McRobert, A. P. (2016). Attacking and defensive styles of play in soccer: analysis of spanish and english elite teams. *Journal of sports sciences*, 34(24):2195–2204.

[Gama et al., 2014] Gama, J., Passos, P., Davids, K., Relvas, H., Ribeiro, J., Vaz, V., and Dias, G. (2014). Network analysis and intra-team activity in attacking phases of professional football. *International Journal of Performance Analysis in Sport*, 14(3):692–708.

[Goddard, 2006] Goddard, J. (2006). Who wins the football? *Significance*, 3(1):16–19.

[Godin et al., 2014] Godin, F., Zuallaert, J., Vandersmissen, B., De Neve, W., and Van de Walle, R. (2014). Beating the bookmakers: leveraging statistics and twitter microposts for predicting soccer results. In *KDD Workshop on large-scale sports analytics*, pages 2–14. ACM New York, NY, USA.

[Gonçalves et al., 2014] Gonçalves, B. V., Figueira, B. E., Maçãs, V., and Sampaio, J. (2014). Effect of player position on movement behaviour, physical and physiological performances during an 11-a-side football game. *Journal of sports sciences*, 32(2):191–199.

[Guan and Wang, 2022] Guan, S. and Wang, X. (2022). Optimization analysis of football match prediction model based on neural network. *Neural Computing and Applications*, 34(4):2525–2541.

[Heuer and Rubner, 2009] Heuer, A. and Rubner, O. (2009). Fitness, chance, and myths: an objective view on soccer results. *The European Physical Journal B*, 67(3):445–458.

[Jain et al., 2021] Jain, S., Tiwari, E., and Sardar, P. (2021). Soccer result prediction using deep learning and neural networks. In *Intelligent Data Communication Technologies and Internet of Things*, pages 697–707. Springer.

[Koopman and Lit, 2015] Koopman, S. J. and Lit, R. (2015). A dynamic bivariate poisson model for analysing and forecasting match results in the english premier league. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 178(1):167–186.

[Nyquist and Pettersson, 2017] Nyquist, R. and Pettersson, D. (2017). Football match prediction using deep learning. Master's thesis.

[Ogunseye et al., 2019] Ogunseye, A., Balogun, O., Ogunseye, A., and Global, F. S. P. (2019). Artificial neural network approach to football score prediction. *Journal of Artificial Intelligence*, 1:1.

[Pappalardo and Cintia, 2018] Pappalardo, L. and Cintia, P. (2018). Quantifying the relation between performance and success in soccer. *Advances in Complex Systems*, 21(03n04):1750014.

[Pappalardo et al., 2019] Pappalardo, L., Cintia, P., Rossi, A., Massucco, E., Ferragina, P., Pedreschi, D., and Giannotti, F. (2019). A public data set of spatio-temporal match events in soccer competitions. *Scientific data*, 6(1):1–15.

[Rahman et al., 2020] Rahman, M. et al. (2020). A deep learning framework for football match prediction. *SN Applied Sciences*, 2(2):1–12.

[Reep and Benjamin, 1968] Reep, C. and Benjamin, B. (1968). Skill and chance in association football. *Journal of the Royal Statistical Society. Series A (General)*, 131(4):581–585.

[Rein and Memmert, 2016] Rein, R. and Memmert, D. (2016). Big data and tactical analysis in elite soccer: future challenges and opportunities for sports science. *SpringerPlus*, 5(1):1–13.

[Rudrapal et al., 2020] Rudrapal, D., Boro, S., Srivastava, J., and Singh, S. (2020). A deep learning approach to predict football match result. In *Computational Intelligence in Data Mining*, pages 93–99. Springer.

[Shlens, 2014] Shlens, J. (2014). A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*.

[Silva et al., 2014] Silva, P., Travassos, B., Vilar, L., Aguiar, P., Davids, K., Araújo, D., and Garganta, J. (2014). Numerical relations and skill level constrain co-adaptive behaviors of agents in sports teams. *PloS one*, 9(9):e107112.

[Tax and Joustra, 2015] Tax, N. and Joustra, Y. (2015). Predicting the dutch football competition using public data: A machine learning approach. *Transactions on knowledge and data engineering*, 10(10):1–13.

[Ulmer et al., 2013] Ulmer, B., Fernandez, M., and Peterson, M. (2013). Predicting soccer match results in the english premier league. *Doctoral dissertation, Doctoral dissertation, Ph. D. dissertation, Stanford*.

[Wang et al., 2015] Wang, Q., Zhu, H., Hu, W., Shen, Z., and Yao, Y. (2015). Discerning tactical patterns for professional soccer teams: an enhanced topic model with applications. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2197–2206.

[Yu and Wang, 2015] Yu, Y. and Wang, X. (2015). World cup 2014 in the twitter world: A big data analysis of sentiments in us sports fans' tweets. *Computers in Human Behavior*, 48:392–400.

[Yucesoy and Barabási, 2016] Yucesoy, B. and Barabási, A.-L. (2016). Untangling performance from success. *EPJ Data Science*, 5(1):1–10.

APPENDIX A
Neural Networks

????? ????????????? ????????????? ????????????? ????????????? ?????????????

APPENDIX B
Terminology

ANN Artificial Neural Network

BP Back-propagation

CV Cross-Validation

EM Expectation Maximization

LSTM Long-Short Term Memory

MLP Multi-layer Perceptron

NLP Natural Language Processing

PCA Principal Component Analysis

PCF Probability Cumulative Function

PDF Probability Density Function

Ranking ELO

RBF-SVM

RNN Recurrent Neural Network

SGD Stochastic Gradient Descend

SNA Social Network Analysis

SVD Singular Value Decomposition

SVM Support Vector Machine