Modelos estadísticos aplicados a la estrategia en los partidos de fútbol

Diego Aranda Madurga

Trabajo de fin de grado de Matemáticas
Universidad de Zaragoza

Director del trabajo: Francisco Javier López Lorente
27 de junio de 2022

Abstract

In this dissertation, we are going to apply statistical methods to football matches' analysis. More specifically, we will analyse FC Barcelona's passes in the 2020-2021 season, a leading team in the Liga Santander whose game is characterised by long possession time with a great control of the ball and by delivering a high number of passes.

Our objective is to build a model that predicts with certitude the probability that a pass with be correct, in respect to a series of characteristics of the pass itself (predictive variables). It is worth noticing that the probability of a right pass follows a Bernoulli distribution (p), which means that a right pass is a binary variable. The models that better predict binary variables are the logistic models.

In addition, we will specify which players are the best and the worse passers in comparison with how difficult the passes that they deliver are. Finally, we will create an auxiliary model to predict which players are dangerous and which ones aren't.

In the first section of this paper we present an introduction which explains the importance of passes in football and the benefits of analysing sports data. Afterwards we explain how the data was processed (and the fact that there exists a very extensive data set, with many variables that are not needed), such as the creation of new variables that can affect the probability of a successful pass.

We will explain the theoretical basis for the logistic models, such as the concept of ODDs and ODDs Ratios. These models are used especially when the variables in question are binary. In these models the probability of a succesful pass is not directly predicted, instead it is estimated through the function:

$$\text{P (succesful pass)} = \frac{e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}}{1+e^{\beta_0+\beta_1 x_1+\cdots+\beta_p x_p}}$$

$x_1,\cdots,x_p$ are the predictive variables.

We create a first logistic model with simple variables to predict the probability of the pass and we check its accuracy via binned graphics. Once we have a final model, which is adjusted correctly, we calculate its ability to predict via ROC curves. The area below the curve (AUC) is an indicator of the power of prediction of the models.

Using the previous logistic model we compare the quality of the pases of the various players to decide who has the best pass ability. We will do this comparing the percentage of successful passes with the number of successful passes that players should deliver according to the model.

In the last part of the dissertation, we will create another logistic model to describe how dangerous a pass is via looking at a simple variable that allows us to discover which players are more dangerous. The objective is to create a holistic observation of the players, to be able to simultaneously observe how dangerous they are in combination with how often they achieve a successful pass.

Índice general

Capítulo 1

Introducción

En España no hay duda de que el fútbol es el deporte rey. Este deporte mueve una cantidad ingente de dinero. Los medios de comunicación: periódicos deportivos, programas de radio y televisión, dedican prácticamente todo el tiempo de sus contenidos a las noticias relacionadas con el fútbol. Por todo ello también tiene una gran influencia en la sociedad en la que vivimos. Prueba de ello es la gran cantidad de público que se congrega en los acontecimientos deportivos y en las transmisiones televisivas cada fin de semana, especialmente cuando juegan equipos grandes con muchos seguidores.

Pero no sólo es afición, también es el deporte más popular del mundo en cuanto a participación, pues lo practican unos 270 millones de personas [1]. El mundial de fútbol, que se organiza cada cuatro años, es el evento deportivo más popular, en el que participan países distribuidos por los cinco continentes.

Y para marcar un gol, se necesitan los pases. Un pase se considera la acción de enviar el balón a un compañero. Los pases son la columna vertebral del fútbol [2], es decir, el pase es el medio más común y seguro de llevar el balón cerca de la portería rival. Se trata del elemento principal de cada combinación, siendo la base del juego en equipo. Sabedores de esto, cabe notar que todos los equipos de fútbol profesionales de la liga española (Liga Santander) destinan grandes cantidades de recursos en contratar a los mejores pasadores. Dada su importancia, es uno de los elementos mas entrenados.

Si hablamos de pases, el equipo de la Liga Santander que más pases realiza es el Fútbol Club Barcelona. Desde la época cuando fue entrenado por Johan Cruyff (1988-1996), el Barcelona ha tenido un estilo propio basado en la posesión de balón, la construcción de jugadas en el centro del campo, con jugadores capaces de mover el balón rápidamente y ofrecer asistencias a los delanteros [3]. La máxima expresión de este estilo se dio en la época de Pep Guardiola como entrenador (2008- 2012), donde fue reconocido como el Barcelona del "tiki-taka", ganando numerosos títulos (un año ganaron todos los títulos posibles, con el famoso "sextete"). Durante todos estos años hasta la actualidad, el Barcelona ha mantenido esta misma idea de juego, usando el pase como elemento principal.

Debido a este estilo de juego, una de las piezas fundamentales del Barcelona son los pasadores. Jugadores capaces de ser muy precisos, sin cometer apenas errores. Son capaces de filtrar los balones entre los jugadores rivales creando ocasiones de gol.

Cabe notar que no todos los pases tienen la misma dificultad. Cuanta más dificultad haya en el pase, menor es la probabilidad de que el pase sea acertado, es decir, que el balón llegue al jugador del mismo equipo deseado. Un pase no tiene la misma dificultad cuando es entre los defensas, es decir, en el campo propio y sin estar presionados por el equipo rival, que un pase en el borde del área rival, donde los jugadores del equipo contrario tienen mucha más posibilidad de interceptarlo.

El análisis de datos en los partidos de fútbol ha sido muy utilizado a lo largo de los últimos tiempos.

Se han realizado una gran cantidad de estudios con el objetivo de analizar a los equipos para ver quienes son mejores jugadores o qué posiciones mejorar. Estos son por ejemplo [4],[5].Para nuestro estudio nos hemos basado en una serie de artículos donde relacionan la probabilidad del acierto del pase respecto a variables que pueden afectar en esta probabilidad [4]. No hemos utilizado exactamente la misma técnica que se utilizan en estos artículos, sino que hemos cogido las ideas de posibles casos de estudio.

El objetivo de nuestro estudio es relacionar la probabilidad de acierto de los pases del Fútbol Club Barcelona respecto a una serie de variables sencillas sobre el propio pase y los jugadores que participan en él. Una vez obtenido este modelo, nuestro objetivo es deducir qué jugadores son los mejores y peores pasadores, es decir, los jugadores que aciertan más pases de los que deberían por su dificultad y los jugadores que erran más pases de los estimados por su dificultad. Este estudio lo haremos mediante regresión logística múltiple.

En la parte final del trabajo haremos un modelo logístico auxiliar para estimar qué jugadores son más o menos peligrosos. Compararemos si está relacionado el concepto peligrosidad con el concepto dificultad. Veremos qué jugadores hacen más pases peligrosos y su relación con su probabilidad de acierto (dificultad).

Capítulo 2

Tratamiento de datos

2.1. Introducción

Los datos con los que hemos realizado el TFG son los correspondientes a los partidos del FC Barcelona en la temporada 2020-2021, en particular, los relativos a los pases entre los jugadores de este equipo. En este capítulo también explicaremos las transformaciones de los datos que hemos utilizado para obtener el conjunto de variables finales (utilizadas en los modelos de predicción). Para facilitarnos la tarea, he creado varios programas de Java que se encuentran en el anexo D.

2.2. Obtención de datos

Los datos de cada partido del Barcelona están sacados de la pagina web: [6]

Me he registrado vía correo electrónico, para poder acceder a la base de datos. Cada partido está en formato .json. Para poder transformarlo a un formato .csv he usado el transformador: [7]

Estas bases de datos contienen información detallada sobre todas las acciones del juego de cada partido, es decir, sobre pases, tiros, faltas, tarjetas, presión, regates... y las características de cada acción de forma cronológica. En el anexo C adjunto alguna fila de la base de datos como ejemplo (Figura 12). Debido a que las bases de datos son demasiado grandes (sobre 250 variables), con demasiada información sobre los partidos, necesitamos un filtrado de variables. Cada partido consta de alrededor de 4000 registros, correspondientes a todas las acciones que ocurren en su trascurso. He hecho un primer filtrado eliminando las columnas no interesantes para nuestro estudio (las relacionadas con otro tipo de acciones como es los tiros, las presiones o los regates). Para realizar este filtrado, he eliminado del partido Real Madrid-Barcelona las variables no necesarias. Una vez obtenido un ejemplo de partido con menos variables, he creado un programa en Java, en el que solo introduciendo el archivo .csv de cada partido, dé como resultado otro archivo .csv que tenga las mismas variables que el archivo filtrado Real Madrid-Barcelona. El programa se encuentra en el anexo como filtrado.java (D.1)

Hay algún partido que no tenía las variables necesarias para nuestro estudio, por lo que ese partido no han sido incluido en el estudio final.

Posteriormente, he introducido una nueva variable al archivo ya filtrado. Es una variable binaria que indica con un 1 cada uno de los tres pases anteriores a un tiro del Barcelona. Las demás filas tienen un valor de 0. Los de los equipos rivales no los he indicado ya que luego procederé a filtrar los datos para quedarme sólo con los pases del Barcelona. Mediante este valor, defino posteriormente el concepto peligrosidad. Este programa devuelve un archivo.csv que contiene los datos del archivo de entrada mas la variable peligrosidad añadida en última posición. El programa de Java se encuentra en el anexo como peligrosidad.java.(D.2)

He introducido nuevas variables al conjunto de datos relacionadas con los duelos. Un duelo es una disputa equilibrada del balón entre dos jugadores de distinto equipo. Se espera que la probabilidad de acierto de un pase sea muy diferente si el pase proviene de un duelo o no. Para introducir estas variables averiguo qué pases van inmediatamente después de un duelo. Dependiendo del tipo de duelo, varío la cantidad de filas entre el duelo y el pase que delimitan si un pase va inmediatamente después de un duelo (de una a tres) . Por ejemplo, si el duelo es aéreo, delimito que el pase se ve afectado por el duelo si va justo en la fila de después. En cambio, si el duelo es raso (terrestre), puede haber más filas entre el duelo y el pase, como un regate o una presión casi en el mismo instante de tiempo, por lo que en este caso he delimitado que un pase va inmediatamente después de un duelo si está a una distancia menor de cuatro filas. Mediante estos datos, y contando con el jugador que da el pase, creo tres variables factor. Una, indicando si el pase se ve afectado por un duelo terrestre o no (llamada dt en el programa de java), otra, si el pase se ve afectado por el duelo aéreo o no (llamada da en el programa de java) y la última, si el jugador que ha realizado el pase es el mismo que ha disputado el duelo (llamada ds en el programa de java). Estas variables también se utilizan en [4]. El programa completo se encuentra en el anexo como duelo.java.(D.3)

Introduzco una nueva variable calculando la posición media de cada jugador en cada partido. Para ello, hago la media de las coordenadas de todas las acciones que realiza cada jugador en cada partido (presión, pases, tiros, duelos, etc) y las añado como dos nuevas variables a cada pase según quien lo realiza. El programa se adjunta en el anexo como posicion.java.(D.4)

Añado dos nuevas variables a cada pase, en las que cuento el número de pases que se llevan en la misma posesión y el tiempo que ha trascurrido desde el último pase. Para ello, considero que dos pases se encuentran en la misma posesión si no hay pases del equipo contrario entre ambos. También considero distinta posesión el pase del balón sacado desde un córner, saque de banda o saque de puerta. He usado el valor 0 por defecto al número y al tiempo de los primeros pases de cada posesión. En este programa también he introducido otra nueva variable factor, siendo 1 si el pase ha sido realizado con los pies y 0 en el caso contrario. En este programa, el archivo.csv de salida solo contiene los pases, no todas las acciones. El programa completo se encuentra en el anexo como pase.java(D.5)

El último programa que he usado, es un filtrado de las primeras columnas (variables) quedándome sólo con las elegidas (copiadas al final en el programa pase) obteniendo un archivo .csv listo para introducir a R. El programa se encuentra en los anexos como eliminado.java(D.6)

He introducido cada partido en el conjunto de estos programas, para obtener los datos con las variables deseadas y filtrados para introducir a R. Notar que en cada partido he de variar algunos parámetros, ya que el orden de las variables en los archivos iniciales no es igual en todos los archivos, ni el número de variables totales. Es decir, algún partido tiene los datos más detallados que otros.

Una vez introducidos los datos de todos los partidos en R, he creado un nuevo dataframe uniéndolos mediante la función cbind() y, eliminando los que no son realizados por el Barcelona obtenemos los datos finales con los que trabajar. Vamos a eliminar los pases realizados por los porteros ya que muchos de ellos se realizan a balón parado y no se consideran iguales al resto de pases. Una vez filtrados los pases realizados por los porteros, se obtiene una muestra de 21776 pases del Barcelona y un total de 15 variables.

Capítulo 3

Variables utilizadas

3.1. Variables dependientes

Como hemos indicado en la introducción, nuestro objetivo es estimar la probabilidad de acierto de cada pase dependiendo de una serie de variables. Notar que cada pase puede ser acertado o fallado, es decir, es una variable binaria. Se denomina pase acertado si el siguiente jugador que toca el balón después del lanzamiento del pase es del mismo equipo del jugador que ha realizado el lanzamiento. Si el primer jugador que toca el balón es del equipo contrario, decimos que el pase se ha fallado. También se determina pase fallado si el balón sale fuera del terreno de juego sin que sea tocado por ningún jugador.

- Pase acertado: Variable binaria en la que se indica si un pase ha sido acertado mediante un 1 y si un pase ha sido fallado mediante un 0.

En la última parte del trabajo haremos un modelo auxiliar para ver qué jugadores realizan más pases peligrosos. Un pase es peligroso si se encuentra entre los tres últimos pases antes de un disparo. No tengo en cuenta si el disparo ha terminado en gol o no. Para considerarse pase peligroso, los pases deben ser del Barcelona y no tiene que haber pases del equipo contrario entre ellos y el tiro. Por lo que la variable es:

- Pase peligroso: Variable binaria en la que se indica si un pase ha sido peligroso mediante un 1 y si un pase no lo ha sido, mediante un 0.

3.2. Variables predictoras

Nuestro objetivo es predecir la variable binaria Y, acierto de pase, respecto a una serie de variables predictoras. Estas variables son elegidas según la influencia que pueden tener para predecir nuestra variable dependiente.

Todos los campos de fútbol de la base de datos tienen unas coordenadas de medida de 120 unidades en el largo del campo, es decir, la distancia entre las dos porterías y 80 unidades en el ancho del campo, es decir, entre las dos líneas de banda. Las coordenadas de los puntos iniciales y finales del pase tienen como origen el punto (0,0), que se define como el córner izquierdo de la portería dónde está el portero del equipo que realiza el pase. Según esta definición, la posición desde donde se ejecuta el pase y la posición dónde se recibe el pase es independiente del lado del campo en el que se está atacando. Notar también que no todos los campos de fútbol tienen exactamente las mismas medidas en metros. En cambio, al usar como medida total del largo (o ancho), una medida fija, si el pase se realiza en una posición inicial (x,y), y suponemos que se recibe m metros más adelante en el eje x (largo), la posición final que resulta es $(x + \frac{\text{m*120}}{\text{longitud del campo}}, y)$. Por lo que las medidas de la posición inicial y final son indicadas como un porcentaje de la longitud del campo. Esta forma de medir es la más beneficiosa para nuestro objetivo, ya que lo que queremos comparar es la distancia respecto al campo, si el lanzamiento o recepción del pase

se encuentra cerca (en porcentaje) o lejos del área rival.

A partir de ahora, cuando hablemos de posiciones o distancias, están definidas de la forma anterior. Vamos a llamar a la posición inicial del pase en el eje X (largo), I0, en el eje Y (ancho), I1, posición final en el eje X, F0 y posición final en el eje Y, F1, con la definición anterior. Notar que las distancias y posiciones según esta definición son positivas. Las variables utilizadas son:

- Distancia media del pase a la portería rival: se han tomado como coordenadas de la portería rival, el centro de esta: (120,40). Primero se calcula las distancia inicial y final del pase a la portería rival mediante la fórmula: DI=$\sqrt{(I0-120)^2+(I1-40)^2}$ y DF= $\sqrt{(F0-120)^2+(F1-40)^2}$ Luego se hace la media de las distancias,

  $$\frac{DI+DF}{2}$$

  Notar que siempre es positiva. Según mi intuición, un crecimiento de la distancia media del pase implicaría una mayor probabilidad de acierto, ya es un pase más lejano a la zona de presión.

- Acercamiento a la portería: está definida como la distancia desde el lanzamiento del pase a la portería rival menos la distancia de la posición final del mismo a la portería rival. El acercamiento es calculado mediante 1a fórmula:

  $$\sqrt{(I0-120)^2+(I1-40)^2}-\sqrt{(F0-120)^2+(F1-40)^2}$$

  Si el acercamiento es positivo, implica que la posición final del pase está mas cerca de la portería rival que la posición inicial.

  Bajo mi intuición, es lógico pensar que un pase cuánto más se acerque a la portería rival, menos posibilidad tiene de ser acertado, ya que los jugadores rivales se suelen encontrar mucho más cerca del balón.

- Longitud del pase: es la distancia entre la posición inicial y la posición final del pase. Se calcula mediante la fórmula:

  $$\sqrt{(F0-I0)^2+(F1-I1)^2}$$

  Notar que la longitud siempre es positiva. Cabe esperar que un pase, cuanto más largo sea, haya menos probabilidad de acierto, ya que es complicado realizar pases largos precisos.

- Largo: Variable factor que indica si un pase tiene longitud mayor que 10 (expresada como largo en ese caso). Si el pase es de longitud menor de 10 está indicado mediante la palabra corto. Esta variable es incluida debido a que el efecto de la longitud en la probabilidad de acierto en un pase varía dependiendo de si es largo o corto. En el capítulo 5 explicaremos con más detalle su influencia.

- Campo: variable binaria que indica si el lanzamiento del pase se ha realizado en el campo propio o en el campo contrario. Se indica como TRUE si se ha realizado en campo del propio o FALSE si se ha realizado en campo rival.

  Se espera que los realizados en campo rival tengan una menor probabilidad de acierto, ya que los jugadores suelen estar mucho más presionados.

- Distancia media a la portería rival del jugador que realiza el pase en el partido: en cada partido, se calcula la posición media de cada jugador. Es decir, se realiza la media de las posiciones de las acciones en las que un jugador participa. En esta media no sólo se cuentan los pases, se cuentan tiros, presiones, regates... Cada acción tiene unas coordenadas, por lo que es posible hacer su

media. Se usa una medida similar en [4]. Luego, con estas coordenadas, se calcula la distancia mediante la fórmula:

$$\sqrt{(\text{posición media x} - 120)^2 + (\text{posición media y} - 40)^2}$$

Los entrenadores varían mucho los esquemas de juego dependiendo de los rivales, por lo que en cada partido, ordenan a sus jugadores situarse en una zona diferente, es decir, la mayoría de las acciones se sitúan sobre esa zona, que puede ser diferente en cada partido para cada jugador. Al introducir esta variable, estamos aportando si un jugador actúa cerca o lejos de la portería rival en ese partido. Cabe esperar que cuanto mayor sea la distancia media de un jugador respecto a la portería rival en ese partido, mayor sea la probabilidad de acertar el pase.

- Parte del cuerpo con la que el jugador golpea balón en el lanzamiento del pase: he considerado una variable factor binaria en la que el valor uno implica que el pase ha sido realizado con el pie, ya sea el pie izquierdo o el derecho. El valor cero indica que el pase ha sido golpeado por alguna de las partes restantes del cuerpo como cabeza, pecho....

  Se espera que cuando un pase se ha realizado con el pie, las probabilidades de acierto sean superiores que si se ha realizado con otra zona del cuerpo.

- Presión de enemigos: variable factor binaria en la que indica si el pase ha sido realizado bajo presión de algún enemigo. Se dice que un pase está en presión si justo antes del momento exacto de realizar el pase, el jugador que lo realiza se encuentra bajo presión, o en el tiempo que tarda el balón hasta que es recibido (aunque sea interceptado por un rival), existe una presión sobre el pasador.

  Se espera que al no haber presión sobre el pasador, la probabilidad de acierto del pase sea mayor que cuando el pasador se encuentra bajo presión.

- Duelo aéreo: variable factor binaria que indica si el pase se ha realizado en un duelo aéreo. Un duelo aéreo es un duelo (disputa equilibrada del balón entre dos jugadores de distinto equipo) en el que el balón no va cerca del suelo. Esta expresado con el factor 0 si el pase no procede de un duelo aéreo y con el factor 1 en caso contrario.

  Se espera que los pases que son realizados justo después de un duelo aéreo tengan menos probabilidad de ser acertados.

- Jugador: es el jugador que da el pase. Se considera que la habilidad individual del jugador que da el pase influya en la probabilidad de acierto.

3.3. Análisis descriptivo

Vamos a realizar un análisis cualitativo de todas las variables que usamos en el estudio. Notar que la muestra de datos del Barcelona en la que estamos trabajando tiene 21776 filas, es decir 21776 pases realizados por jugadores del Barcelona en la temporada 2020-2021. Estos no son la cantidad total de pases ya que de algún partido no ha sido posible obtener las variables explicativas del modelo.

- Pase acertado: Los pases acertados del Barcelona son:

	Acertado	Fallado
Número	19475	2301
Porcentaje	89.43	10.57

Notar que se aciertan muchos mas pases de los que se fallan, algo muy lógico.

- Pase peligroso: los pases peligrosos del Barcelona son:

	No Peligroso	Peligroso
Número	20544	1232
Porcentaje	94.34	5.66

La mayoría de pases no son peligrosos, ya que los tiros son elementos no muy frecuentes en un partido en comparación con los pases.

- Promedio de la distancia del pase a la portería rival:

Mín	1st Qu.	Median	Mean	3st Qu.	Max	Sd
6.115	41.039	54.977	57.812	72.852	123.748	22.457

Notar que la desviación típica es elevada comparada con la media, algo coherente ya que hay pases muy lejanos y muy cercanos a la portería rival. La media aproximadamente coincide con la longitud de mitad del largo del campo (60 con nuestra medida), ya que la mayoría de pases se realizan cerca del centro del campo.

- Acercamiento portería:

Mín	1st Qu.	Median	Mean	3st Qu.	Max	Sd
-46.406	-3.965	2.375	2.618	7.934	79.505	10.662

Notar que el acercamiento a la portería rival es muy desigual. Hay valores extremos, como el máximo, que ocurren cuando los defensas pasan en largo, ya que se acercan mucho a la portería rival. Este hecho se ve reflejado en la desviación típica, siendo 10.66 un valor elevado. La media también se ve influenciada por estos valores extremos, por lo que la medida de centralidad más recomendable es elegir la mediana. Como se observa en la tabla, la mayor parte de los pases son hacia la portería contraria.

- Longitud del pase:

Mín	1st Qu.	Median	Mean	3st Qu.	Max	Sd
0.0	10.20	14.75	17.14	21.41	87.81	10.072

Una longitud tan pequeña como 0.1 puede ser explicada en un córner cuando un jugador saca desplazando mínimamente el balón y acude otro para continuar el ataque. Notar que también hay pases muy largos, que al igual que en el caso anterior suelen ser de los defensas. Hay bastante variabilidad en los datos, como se espera ya que los pases pueden ser muy diferentes en cuanto a longitud.

- Largo:

	Corto	Largo
Número	5189	16587
Porcentaje	23.83	76.17

Notar que la mayoría de los pases tienen una longitud de más de 10 unidades.

- Campo:

	Propio	Rival
Número	8167	13609
Porcentaje	37.5	62.5

  Hay más porcentaje de pases en el campo contrario que en campo propio. Esto es lógico ya que en el campo propio, si el equipo rival no presiona, es relativamente sencillo llegar a campo rival con poca cantidad de pases. En el campo rival es donde tienes que realizar mayor cantidad de pases para crear ocasiones de gol.

- Distancia media a la portería rival del jugador que realiza el pase en el partido

Mín	1st Qu.	Median	Mean	3st Qu.	Max	Sd
56.57	57.19	60.02	62.47	65.40	91.57	6.804

  Los defensas son los que tienen medias mayores, muy cerca de su propia portería. Los delanteros, aunque parezca que solo juegan cerca de la portería rival, también interactúan en acciones defensivas, y saques a balón parado, lo que implica que la posición media sea cerca del medio campo. Como se observa en los cuantiles, la mayoría de de distancias medias se engloba en 10 unidades (57-67).

- Parte del cuerpo:

	Pie	Otra
Número	21297	479
Porcentaje	97.8	2.2

  La mayoría de los pases son realizados con el pie ya que es de la forma en la que se mejor se controla el balón. Solo un 2 por ciento de los pases son lanzados con otras partes del cuerpo.

- Presión de los enemigos: La tabla se encuentra en el anexo A, (Tabla 1). La mayoría de los pases se hacen sin estar estrictamente presionados, ya que los jugadores profesionales suelen hacer pases adelantándose a la posible presión de los jugadores rivales.

- Duelo aéreo: La tabla se encuentra en el anexo A, (Tabla 2). Hay muy pocos pases realizados después de duelos aéreos ya que el juego del Barcelona no se basa en colgar centros, ni pases largos, sino en la combinación con pases rasos.

- Jugadores del Barcelona: Se han eliminado los pases de Carles Aleñá, ya que el jugador sólo ha dado 42 pases en toda la temporada, que es una cantidad muy baja para comparar la diferencia de probabilidad de acierto en el pase con otros jugadores. (Se fue cedido al Getafe en el mercado invernal). La tabla se encuentra en el anexo: (Tabla 3)

  Cabe notar que el Barcelona suele salir con parecidos jugadores titulares en la mayoría de partidos y por tanto estos jugadores tienen más tiempo para realizar pases, por lo que su porcentaje de pases totales es mayor. Los jugadores que más pases realizan son: Leo Messi, Sergio Busquets, Jordi Alba y Frenkie de Jong, titulares indiscutibles para el Barcelona de Ronald Koeman.

  Notar que Junior Firpo es el jugador que más porcentaje de pases acierta y Leo Messi es el jugador que más porcentaje de pases falla. Samuel Umtiti es el jugador menos peligroso y Leo Messi es el jugador más peligroso.

Capítulo 4

Modelo de regresión logística

4.1. Introducción

Estamos interesados en encontrar un procedimiento para estimar la probabilidad de que un pase sea acertado. Vamos a considerar Y como la variable aleatoria acierto en el pase, que toma los valores:

- 1 si el pase ha sido acertado, es decir, el primer jugador que toca el balón después del lanzamiento es del Barcelona.
- 0 si el pase ha sido fallado, es decir, el primer jugador que toca el balón después del lanzamiento es del equipo rival o el balón sale fuera del campo.

Este tipo de problemas donde la variable respuesta es una respuesta binaria son muy comunes en estadística, por ejemplo, saber si una semilla germina o no dependiendo de la cantidad de agua que se le aporta o por ejemplo, si se ha concedido un préstamo o no dependiendo del nivel de ingresos del solicitante.

Los problemas de este tipo se pueden analizar mediante mínimos cuadrados. Podríamos analizar la probabilidad de que un pase sea acertado o fallado mediante mínimos cuadrados tal como se estudia en la asignatura Técnicas de Regresión. Podemos suponer que se trata de una regresión lineal simple, es decir, $Y = \beta_0 + \beta_1 * X_1$. En el caso de regresión lineal múltiple se obtiene las mismas conclusiones. Ocurre que, al tratarse de una recta, para valores extremos del predictor, se obtienen valores de Y menores que 0 o mayores que 1, lo que entra en contradicción con el hecho de que las probabilidades siempre están dentro del rango [0,1] [9].

Otra razón por la que el método ordinario de mínimos cuadrados es inapropiado es debido a que el acierto de cada pase es una variable aleatoria Bernoulli(p), y como consecuencia, su varianza depende de p (en concreto, su varianza es p(1-p)), por lo que no se cumple la hipótesis de varianza constante (homocedasticidad), necesaria para realizar un ajuste de regresión lineal por mínimos cuadrados ordinarios ya que cada pase tiene un coeficiente de p diferente.

Después de estas consideraciones concluimos que usar el método de mínimos cuadrados no es una opción correcta para de analizar este tipo de problemas con variables respuesta binarias.

4.2. Modelos lineales generalizados

El modelo lineal generalizado se define como una generalización del modelo lineal cuando una o varias condiciones no se satisfacen. En particular, esta generalización permite varianzas no constantes y errores con distribuciones no normales, como es nuestro caso, con la distribución Bernoulli. Estos modelos tienen tres componentes principales.

La componente aleatoria es el vector aleatorio Y = (y_1,y_2 , , ,y_n), cuyos elementos son independientes y están distribuidos con función de distribución perteneciente a la familia exponencial (capítulo 5 de [8]). Notar que es nuestro caso, ya que la variable respuesta tiene distribución Bernoulli(p), que se puede escribir como :

$$\mathrm{P[X = x]} = p^x(1-p)^{1-x} = e^{x\log\left(\frac{p}{1-p}\right)+\log(1-p)} \qquad \text{x=0,1}$$

Componente sistemática, formada por las variables explicativas (X_1, , ,X_p), cada variable formada por n observaciones. En nuestro caso está formada por las variables que considero influyentes en la dificultad del pase. Su combinación lineal se denomina predictor [10], que se puede expresar de forma:

$$\eta_i = \beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}$$

donde $\beta = (\beta_0, \beta_1, , ,\beta_p)$ es el vector de estimadores. Es decir $\eta = X\beta$, siendo X la matriz del diseño.

Función de ligadura o enlace: denotamos como $\mu = \mathrm{E}(Y)$ al valor de la esperanza de Y (en nuestro caso p, probabilidad de acierto, ya que Y tiene distribución Bernoulli(p)). La función de enlace g es una función monótona y diferenciable que relaciona μ con el predictor lineal. Esta función cumple $\eta_i = g(\mu_i) \quad \forall i \in (1,n)$. En nuestro caso usaremos $g(\mu) = \log\left(\frac{\mu}{1-\mu}\right)$ [10].

4.3. Modelos logísticos

Mediante regresión lineal, nos encontrábamos el problema que a valores extremos de predictores, se podrían estimar probabilidades fuera de los límites [0,1]. Para solucionar este problema, la regresión logística transforma el valor devuelto por la regresión lineal o regresión lineal generalizada ($Y = \beta_0 + \beta_1 * X$) empleando una función cuyo resultado está siempre comprendido entre 0 y 1. Su objetivo principal es modelar como afectan las variables regresoras en la probabilidad asociada a la variable binaria. [9]

La función que se aplica es conocida como la función sigmoide:

$$\sigma(x) = \frac{e^x}{1+e^x}$$

Notar que para todo valor de $x \in R$, su imagen pertenece al intervalo [0,1]. Para valores muy pequeños de x, la función sigmoide tiene valores muy cercanos a 0. En cambio, para valores muy grandes de x, la función sigmoide tiene valores cercanos a 1.

Cambiando la variable x, por la función de un modelo de regresión lineal generalizado, se obtiene la expresión:

$$P[Y = 1|X_1 = x_1, X_2 = x_2, , ,X_p = x_p] = \frac{e^{\beta_0+\beta_1*x_1+\cdots+\beta_p*x_p}}{1+e^{\beta_0+\beta_1*x_1+\cdots+\beta_p*x_p}} \tag{4.1}$$

La relación entre la probabilidad y una variable predictora tiene forma de curva S, como se muestra en la figura Figura 4.1:

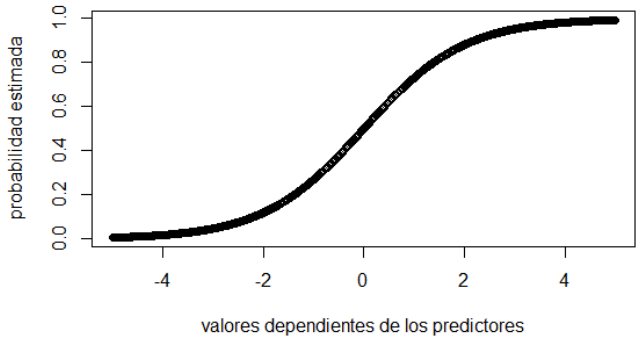


Figura 4.1: Ejemplo de curva S

4.4. ODDs y ODDs Ratio

Para ajustar la función anterior mediante una tendencia lineal, la función anterior puede ser linealizada por medio de lo que se conoce como transformación logística (transformación *logit*). En este caso, en lugar de trabajar directamente con $P[Y=1|X_1=x_1,X_2=x_2,,,X_p=x_p]$, se trabaja con un valor transformado de $P[Y=1|X_1=x_1,X_2=x_2,,,X_p=x_p]$. Despejando $\beta_0+\beta_1*x_1+\cdots+\beta_p*x_p$ de (4.1) se obtiene:

$$\log\left(\frac{P[Y=1|X_1=x_1,X_2=x_2,,,X_p=x_p]}{1-P[Y=1|X_1=x_1,X_2=x_2,,,X_p=x_p]}\right)=\beta_0+\beta_1*x_1+\cdots+\beta_p*x_p$$

Como hemos detallado antes, la función de enlace es $\log\left(\frac{\mu}{1-\mu}\right)$.

Definición. Los ODDs o razón de probabilidad de que el suceso ocurra se definen como el ratio entre la probabilidad de que el suceso ocurra y la probabilidad de que no ocurra el suceso

$$\text{ODDs}=\frac{p}{q}=\frac{p}{1-p}$$

Supóngase que la probabilidad de que un suceso ocurra es de p=0.8, por lo que la probabilidad de que no ocurra es de q=1 - 0,8 = 0,2. En este caso los ODDs de que el suceso ocurra son $\frac{0,8}{0,2}=4$, lo que equivale a decir que se esperan que ocurran 4 sucesos por cada suceso que no ocurra. Los ODDs están comprendidos en el intervalo $[0,\infty]$ dado que el valor de la probabilidad está acotado en el intervalo $[0,1]$.

En nuestro caso, los ODDs son la probabilidad de que un pase sea acertado entre la probabilidad de que un pase sea fallado.

Notar que la transformación de probabilidades a ODDs es monótona, si la probabilidad aumenta también lo hacen los ODDs, y viceversa. Llamando $\alpha=P[Y=1|X_1=x_1,X_2=x_2,,,X_n=x_p]$ se tiene:

$$\frac{\partial\left(\frac{\alpha}{1-\alpha}\right)}{\partial\alpha}=\frac{1}{(\alpha-1)^2}>0\;\forall\alpha\in(0,1)$$

Definición. El ODDs Ratio, OR, es la razón de los ODDs correspondiente a un suceso bajo una cierta condición entre los ODDs del mismo suceso bajo otra condición. En nuestro caso, llamando p a la probabilidad de acierto bajo condiciones A, p|A, q=1-p|A, y llamando p' a la probabilidad de acierto bajo condiciones B, es decir, p'|B, q'=1-p'|B, se tiene que:

$$\text{OR}=\frac{\frac{p}{q}}{\frac{p'}{q'}}$$

Voy a poner un ejemplo para facilitar la interpretación del término ODDs ratio. Suponer que los ODDs de padecer gripe en España a lo largo de la vida son de 6, es decir, por cada 6 personas que tienen gripe en España, hay una que no la tiene. En cambio, en Australia, los ODDs de padecer gripe son de 3. Por cada 3 personas que tienen gripe a lo largo de su vida en Australia, hay una que no la tiene. Entonces el ODDs Ratio entre España y Australia es de $\frac{6}{3} = 2$. Esto quiere decir que por cada dos personas que padecen gripe en España, hay una que padece en Australia.

Por lo que ODDs Ratio es una indicación de como aumentan o disminuyen los ODDs de que ocurra un cierto suceso dependiendo de unas determinadas condiciones.

4.5. Obtención de los estimadores

En los modelos lineales generalizados, los estimadores se calculan mediante el método de máxima verosimilitud. Llamamos al vector de estimadores $\hat{\beta} = \left(\hat{\beta}_0, \hat{\beta}_1, , , \hat{\beta}_p\right)$. Estos estimadores son obtenidos numéricamente por métodos iterativos. En este trabajo no vamos a centrarnos en qué fórmulas utilizan ya que son complejas y los cálculos los realiza un ordenador. Los cálculos y el análisis estadístico lo realizaremos mediante el programa R.

Capítulo 5

Modelo aplicado a los datos

5.1. Modelo logístico inicial probabilidad de acierto del pase

Recordar que nuestro objetivo es predecir el *logit* de la probabilidad de acierto de un pase respecto a una serie de variables predictores, es decir:

$$\log\left(\frac{P[Y=1|X_1=x_1,X_2=x_2,,,X_n=x_p]}{1-P[Y=1|X_1=x_1,X_2=x_2,,,X_n=x_p]}\right)=\beta_0+\beta_1*x_1+\cdots+\beta_p*x_p$$

Dado que tenemos un número elevado de variables predictoras, hemos realizado el modelo de regresión logística paso a paso con criterio AIC. El criterio de información de Akaike (AIC) es una medida de la calidad relativa de un modelo estadístico, para un conjunto dado de datos. Como tal, el AIC proporciona un medio para la selección del modelo [11]. AIC maneja un balance entre la bondad de ajuste del modelo y la complejidad del modelo. He realizado varios modelos introduciendo más variables de las explicadas anteriormente, pero resultaban no ser relevantes. Estas variables son:

- Minuto del partido, esperando que cuanto más tarde sea un pase, el jugador está más cansado y más probabilidad de errar el pase.
- El número de pases en la secuencia de pases, esperando que cuanto más alto sea el número en la misma secuencia de pases, mas posibilidad de presión del rival y menor probabilidad de acertar el pase.
- Variable factor que indica si el pase ha sido realizado justo inmediatamente después de un duelo terrestre. (el balón va cerca del suelo)

De todas estas variables, el modelo indica que no son significativas y al eliminarlas, el AIC disminuye notablemente.

Hacemos un primer modelo:

$$\begin{aligned}\log\left(\frac{y}{1-y}\right)=\beta_0+\beta_1*\text{distancia promedio del pase}+\beta_2*\text{acercamiento}+\beta_3*\text{longitud}+\beta_4*\text{campo}\\ +\beta_5*\text{distancia media del jugador}+\beta_6*\text{duelo aéreo}+\beta_7*\text{parte del cuerpo}+\beta_8*\text{presión}\end{aligned} \tag{5.1}$$

5.1.1. Validación del modelo

Para cada pase, se tiene un valor ajustado de la probabilidad de acertar un pase por el modelo. Esta predicción se obtiene mediante la formula:

$$\frac{e^{\hat{\beta}_0+\hat{\beta}_1*x_1+\hat{\beta}_2*x_2+\hat{\beta}_3*x_3\cdots+\hat{\beta}_8*x_8}}{1+e^{\hat{\beta}_0+\hat{\beta}_1*x_1+\hat{\beta}_2*x_2+\hat{\beta}_3*x_3\cdots+\hat{\beta}_8*x_8}}$$

Siendo $x_1,,,x_8$ las variables regresoras del apartado anterior.

El gráfico binnedplot agrupa los datos en 100 grupos diferentes, mediante los valores ajustados. Es decir, como hay 21776 pases, hay 21776 valores ajustados, por lo que hace 100 grupos de alrededor de 217 o 218 pases según el valor de los valores ajustados, por orden de menor a mayor. (Primer grupo, los 217 valores ajustados más pequeños, segundo grupo, los 217 siguientes valores ajustados). Para cada grupo, calcula la media de los valores ajustados. También calcula la probabilidad de acierto de ese grupo contando los pases acertados y dividiendo por los pases totales. El este gráfico binnedplot se representa la diferencia entre la media de los valores ajustados y el porcentaje real de acierto, frente a la media de los valores ajustados.

Para que se considere que el modelo se ajusta correctamente a los datos, aproximadamente el 95 por ciento de los puntos deben caer dentro de las franjas negras que representan el intervalo de confianza del 95 %.

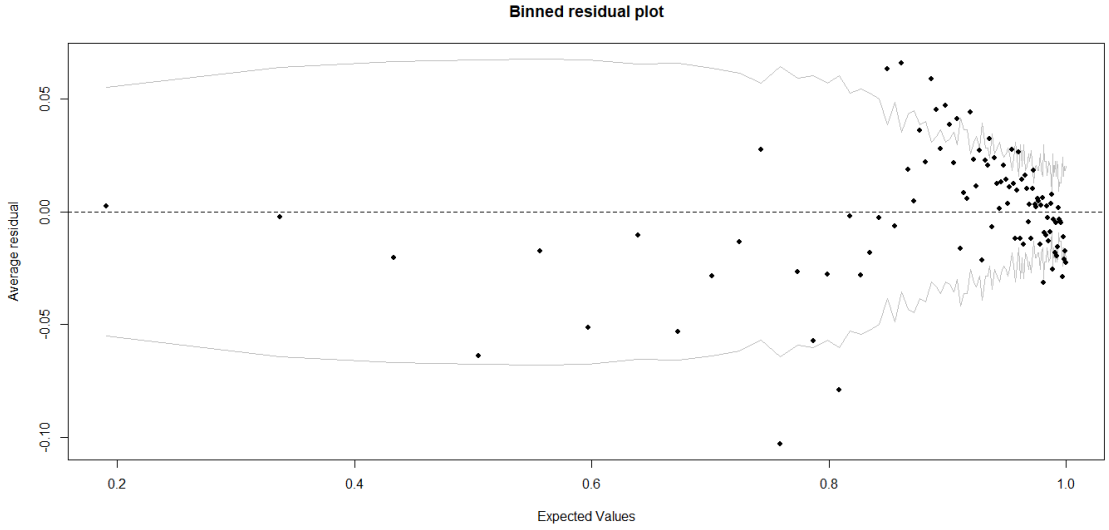


Figura 5.1: Binnedplot del modelo *logit* (5.1)

Además los puntos no deben seguir una tendencia clara, ya que se trata de residuos, y si siguen una tendencia, implica que el modelo no ha sido ajustado correctamente. En nuestro caso, la Figura 5.1 muestra que hay un mayor porcentaje de puntos que se encuentran fuera de los intervalos de confianza.

Representamos los residuos del modelo respecto a las variables numéricas en un binnedplot. Ahora los grupos se hacen según las variables numéricas, es decir, por ejemplo, si representamos los residuos frente al acercamiento, se hacen 100 grupos en orden creciente según el acercamiento. (primer grupo, los 217 que menor acercamiento tienen, segundo grupo, los siguientes 217 que menor acercamiento tienen y así hasta el grupo 100). Se calcula la media de los valores ajustados para cada grupo. También se calcula la probabilidad de acierto de cada grupo contando los pases acertados y dividiendo por los pases totales. Se representa, la diferencia entre la media de los valores ajustados y el porcentaje real de acierto, frente la media del acercamiento de cada grupo. Ver Figura 5.2.

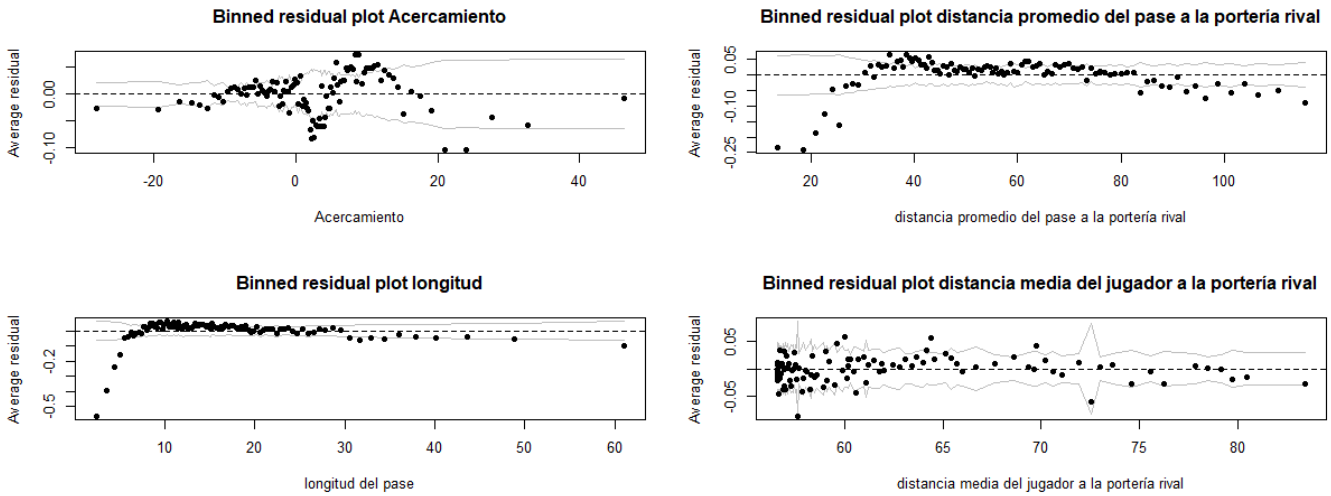


Figura 5.2: Binnedplot de los residuos frente a las variables numéricas del modelo (5.1)

En estos gráficos hay dos variables que no se ajustan correctamente. Representando los residuos frente a la distancia promedio del pase y frente a la longitud se observan que hay muchos puntos que no se encuentran dentro del intervalo de confianza. Tampoco tienen tendencia aleatoria, que es lo que se esperaría si el modelo estuviese bien especificado.

El gráfico de los residuos frente a la distancia promedio del pase tiene una tendencia cuadrática. Es decir, la relación del modelo *logit* de acierto del pase no es lineal respecto a la variable distancia promedio del pase. Por ello, añadimos el término cuadrático al modelo.

Se puede distinguir que los puntos menores de 10 unidades de longitud siguen una tendencia diferente a los puntos que tienen más de 10 unidades de longitud. Para evitar esa tendencia y conseguir que el modelo se ajuste correctamente a los datos, introducimos la variable largo en el modelo (variable factor que indica que un pase tiene más de 10 unidades de longitud), así como su interacción con la longitud.

5.2. Modelo logístico corregido probabilidad acierto del pase

El modelo queda:

$$\log\left(\frac{y}{1-y}\right) = \beta_0 + \beta_1 * \text{distancia promedio} + \beta_2 * \text{distancia promedio}^2 + \beta_3 * \text{acercamiento} +$$

$$\beta_4 * \text{longitud} + \beta_5 * \text{largo} + \beta_6 * \text{campo} + \beta_7 * \text{distancia media del jugador} + \tag{5.2}$$

$$\beta_8 * \text{duelo aéreo} + \beta_9 * \text{parte del cuerpo}$$

$$+\beta_{10} * \text{presión} + \beta_{11} * \text{longitud si el pase es largo}$$

Resumen del modelo:

```
Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)              -8.80055    0.37633   -23.4   <2e-16 ***
promdis                   0.12558    0.00512    24.5   <2e-16 ***
promdis2                 -0.00080    0.00004   -20.3   <2e-16 ***
acercamiento             -0.11246    0.00361   -31.1   <2e-16 ***
longitud                  0.54074    0.02288    23.6   <2e-16 ***
largo[T.largo]            4.76799    0.16387    29.1   <2e-16 ***
campo[T.TRUE]            -0.61032    0.10963    -5.6    3e-08 ***
mediajugadordistan        0.03245    0.00519     6.3    4e-10 ***
da[T.1]                  -1.13556    0.20951    -5.4    6e-08 ***
parte[T.1]                1.85768    0.12075    15.4   <2e-16 ***
under_pressure[T.TRUE]   -0.79126    0.07436   -10.6   <2e-16 ***
longitud:largo[T.largo] -0.54559    0.02301   -23.7   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14692.6  on 21775  degrees of freedom
Residual deviance:  9753.8  on 21764  degrees of freedom
AIC: 9778

Number of Fisher Scoring iterations: 6
```

Figura 5.3: Resumen del modelo *logit* (5.2)

Como se observa en el resumen (Figura 5.3), todas las variables son muy significativas a la hora de predecir si un pase ha sido acertado o fallado.

La exponencial de los coeficientes está muy relacionado con el ODDs ratio. Si aumentáramos una unidad el acercamiento (llamando al acercamiento X_3), y si el resto de variables se mantuvieran constantes, disminuirían los ODDs de acertar el pase en 0.894 veces más que si no se aumentara esa unidad del acercamiento. Esto es debido a que:

$$OR = \frac{e^{\hat{\beta}_3*(X_3+n)+\hat{\beta}_1*(X_1)+...}}{e^{\hat{\beta}_3*(X_3)+\hat{\beta}_1*(X_1)+...}} = e^{n\hat{\beta}_3} = (\text{si n=1}) = e^{-0,11246} = 0,894$$

De hecho, en este caso, el ODDs Ratio entre aumentar una unidad el acercamiento y no, coincide con la exponencial del coeficiente acercamiento.

Aplicando el mismo razonamiento, si golpeáramos el balón con el pie, es decir el coeficiente del predictor parte es uno, y mantenemos constantes el resto de predictores, aumentarían los ODDs de acertar el pase en $e^{1,857}$ =6.409 veces más que si se golpeara con otra parte del cuerpo.

En el caso de la distancia promedio a la portería, si aumentamos en n unidades la distancia, (llamando la distancia X_1) los ODDs Ratio varían de este modo:

$$\frac{e^{\hat{\beta}_1*(X_1+n)+\hat{\beta}_2*(X_1+n)^2+\hat{\beta}_3*(X_3)+...}}{e^{\hat{\beta}_1*(X_1)+\hat{\beta}_2*(X_1)^2+\hat{\beta}_3*(X_3)+...}} = \frac{e^{\hat{\beta}_1*(X_1)+n\hat{\beta}_1+\hat{\beta}_2*(X_1)^2+2n\hat{\beta}_2*(X_1)+n^2\hat{\beta}_2+\hat{\beta}_3*(X_3)+...}}{e^{\hat{\beta}_1*(X_1)+\hat{\beta}_2*(X_1)^2+\hat{\beta}_3*(X_3)+...}} = e^{n\hat{\beta}_1}e^{2n\hat{\beta}_2*X_1}e^{n^2\hat{\beta}_2} =$$

$$= (\text{si n=1}) = 1{,}132 * e^{-0{,}0016 * X_1}$$

Notar que este número es positivo para números menores que 77. Es decir si la distancia es menor que 77, si aumentamos la distancia en una unidad, los ODDs de que el pase fuera acertado aumentarían en $1{,}132 * e^{-0{,}0016 * X_1}$ veces. Notar que el porcentaje de pases con distancia promedia del pase menor de 77 es del 80 %.

En el caso de la longitud ocurre una cosa diferente. Si el pase tiene una longitud menor de 10 unidades y aumentamos n unidades tal que el pase nuevo no pasa de 10 unidades, el ODDs Ratio es $e^{0{,}54074} = 1{,}717$, es decir si aumentamos una unidad, los ODDs de acierto del pase aumentan se multiplican por 1,717. En cambio, si tenemos X_4 unidades de longitud (<10) y aumentamos n unidades (suficientes para que el pase fuera largo), el ODDs Ratio es: (llamamos X_4 a la variable longitud y a su coeficiente del modelo β_4, tomando β_5 al coeficiente de largo si un pase es largo, y llamandoβ_{11} al coeficiente de como aumenta la longitud si un pase es largo)

$$OR = \frac{e^{\hat{\beta}_4 * (X_4+n) + \hat{\beta}_5 + \hat{\beta}_{11} * (X_4+n) + \ldots}}{e^{\hat{\beta}_4 * (X_4) + \ldots}} = e^{n\hat{\beta}_4} e^{\hat{\beta}_5} e^{\hat{\beta}_{11}(X_4+n)} = (\text{si n=1}) = 202{,}06 * e^{-0{,}546(X_4+1)}$$

Si el pase tiene X_4 unidades de longitud (>10 para que sea considerado pase largo) al aumentar n unidades, se obtiene un ODDs Ratio de:

$$\frac{e^{\hat{\beta}_4 * (X_4+n) + \hat{\beta}_5 + \hat{\beta}_{11} * (X_4+n) + \ldots}}{e^{\hat{\beta}_4 * (X_4) + \hat{\beta}_5 + \hat{\beta}_{11} * (X_4) + \ldots}} = e^{n\hat{\beta}_4} e^{n\hat{\beta}_{11}} = e^{n(\hat{\beta}_4 + \hat{\beta}_{11})} = e^{n * -0{,}0048} \approx 1$$

Es decir, a partir de 10 metros en los pases, la longitud no afecta a la probabilidad del pase. Notar que la mayor longitud de un pase es 87.

Los coeficientes de los ODDs Ratio mayores que uno señalan que un aumento de la variable regresora, aumenta los ODDs de acertar el pase (es decir, la variable dependiente). En cambio, un coeficiente del Odd Ratio menor de 1 indican que un aumento de la variable regresora (de los predictores), disminuye los ODDs de acertar el pase.

Notar que excepto la longitud, las variables predictoras varían los ODDs disminuyendo y aumentando como se esperaba. Según el modelo, un aumento de la longitud aumenta la probabilidad de acertar un pase hasta un punto en el que se mantiene constante. La posible explicación es que la mayoría de los pases entre los defensas, o entre los centrocampistas son muy largos, ya que el objetivo es cambiar el balón de una banda a otra y los jugadores se encuentran muy separados. Pero notar que también son muy seguros, ya que son hechos desde unas posiciones defensivas. En cambio, cuando se realizan pases cortos, es debido a que están cerca del área rival, y los jugadores están muy presionados por los rivales, por lo que necesitan combinarse para crear peligro, reduciendo la probabilidad de acierto en el pase.

5.2.1. Validación del modelo

Vamos a comprobar si con el modelo actual se ha solucionado las tendencias que se representaban en el caso anterior. Representamos en un binnedplot los residuos frente a los valores ajustados: (Figura 5 en el anexo B) En nuestro caso, el 95 % de los datos se engloban dentro de los intervalos de confianza, pero los datos siguen una pequeña tendencia, que la solucionaremos con el modelo incluyendo el jugador que ha realizado el pase.

Representando los residuos del modelo respecto a las variables numéricas en un binnedplot, (agrupando en grupos de 217 observaciones en orden de la variable numérica de cada gráfico) , (Figura 6 en el anexo B), se observa que la mayoría de los puntos se encuentran dentro de los intervalos de confianza. Además, no presentan ninguna tendencia.

5.2.2. Curva ROC

Una vez que tenemos un modelo que se ajusta correctamente a los datos, procedemos a validar su capacidad de predicción. Esto lo haremos mediante la curva ROC.

Los modelos de regresión logística nos proporciona el valor estimado de $P(Y = 1|\ X = X_i)$ para el pase i, pero no nos dice si el pase es acertado o fallado. Solo nos indica la probabilidad de ocurrencia que le asigna. Debemos ser nosotros quienes decidamos que un pase es fallado o acertado a partir de esa probabilidad. Por lo que una cuestión clave es decidir a partir de qué valor de corte, c, decidimos que el el pase es acertado si $P(Y = 1|X = X_i) \geq c$ y que es fallado si $P(Y = 1|X = X_i) < c$.

Definición. El concepto sensibilidad, mide la probabilidad de que un pase acertado sea clasificado como que un pase acertado

Definición. El concepto especificidad, mide la probabilidad de que un pase fallado sea clasificado como que un pase fallado.

La curva ROC (Receiver Operating Characteristic) representa la 1-especificidad (pases que se clasifiquen fallados cuando han sido acertados) frente a la sensibilidad para cada posible valor de c. Así, el área bajo la curva ROC (AUC) proporciona una medida global de la capacidad del modelo para separar los pases que son acertados frente a los que son fallados [12]. A partir de 0.7 en el AUC se considera que las predicciones del modelo son aceptables.

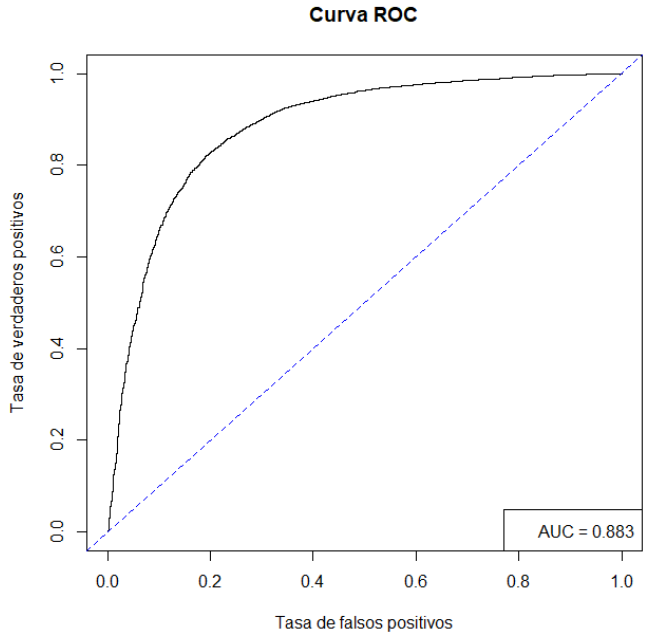


Figura 5.4: Curva Roc modelo *logit* 5.2

El valor del AUC es de 0,883 $>$ 0,7, por lo que el modelo se considera aceptable y tiene buena capacidad predictiva.

5.3. Modelo logístico probabilidad de acierto de un pase incluyendo los jugadores

Aún teniendo el modelo anterior (5.2) buena capacidad predictiva, el gráfico binnedplot de los residuos frente a los valores ajustados presenta una ligera forma no aleatoria (Figura 5 del anexo B). Por ello vamos a introducir al modelo anterior la variable jugadores, ya que cada jugador puede tener una habilidad distinta para hacer un pase.

El modelo es:

$$\log\left(\frac{y}{1-y}\right) = \beta_0 + \beta_1 * \text{distancia promedio} + \beta_2 * \text{distancia promedio}^2 + \beta_3 * \text{acercamiento}$$

$$+\beta_4 * \text{longitud} + \beta_5 * \text{largo} + \beta_6 * \text{campo} + \beta_7 * \text{duelo aéreo} + \beta_8 * \text{parte del cuerpo} + \tag{5.3}$$

$$\beta_9 * \text{presión} + \beta_{10} * \text{longitud si el pase es largo} + \beta_{11} * \text{jugador}$$

Es decir, nuestro modelo es un modelo de regresión logística con efecto fijo debido al jugador. Notar que en este modelo hemos eliminado la distancia media de los jugadores respecto a la portería rival ya que la variable no resultaba ser significativa. Esto es lógico ya que la influencia de los jugadores ya está corregida mediante la variable jugadores.

El resumen de este modelo se muestra en la Figura 7 del anexo B

Hemos comparado los dos modelos, 5.2 y 5.3 mediante un test anova y resulta que la variable jugador sí es relevante en la predicción del acierto del pase, por lo que es correcto incluirla en el modelo. (p-valor=$2{,}08 * 10^{-10}$) Para cada jugador, si hacemos la exponencial del coeficiente que le acompaña, obtenemos el ODDs Ratio del pase entre ese jugador y Ansu Fati (primero en orden alfabético). Ansu Fati tiene el coeficiente 0, ya que respecto a él son comparados los demás jugadores. Notar que el AIC= 9730 disminuye en este modelo en comparación con el anterior. Eso nos da una pista de que este modelo se ajusta más a los datos. Aún así, vamos a comprobarlo dibujando el gráfico binned de los residuos frente a los valores ajustados (Figura 8 del anexo B).

Notar que más del 95 por ciento de los puntos se incluyen dentro del intervalo de confianza, es decir, solo hay 5 o menos puntos fuera de tal intervalo y se ha eliminado la tendencia que se intuía. Este modelo predice mejor la probabilidad de acierto del pase.

Representando la curva ROC de este modelo: (Figura 9 del anexo B) Se observa una buena capacidad predictiva. Notar también que el AUC= 0.885 se ha incrementado respecto al modelo anterior, suponiendo una mejora en la predicción.

Una vez comprobado que el modelo cumple todos los requisitos, fijamos este modelo como modelo final para predecir la probabilidad de acierto de los pases del Barcelona en la temporada 2020-2021.

Capítulo 6

Análisis de resultados

6.1. Diferencias entre jugadores

Para analizar la habilidad de los jugadores respecto de los pases podemos hacerlo a partir de las estimaciones de sus efectos fijos en el modelo (5.3) o, de manera (aproximadamente) equivalente, a partir de las predicciones que obtenemos para ellos en el modelo (5.2). Elegimos esta segunda opción porque nos será más sencillo hacer su interpretación cuando incluyamos el modelo para peligrosidad.

Así, a partir del modelo (5.2) que calcula una predicción de la probabilidad de acierto o fallo del pase, nuestro objetivo es ver qué jugadores del Barcelona son mejores pasadores y qué jugadores son peores pasadores.

Pero claro, el acierto o fallo del pase depende de las variables explicadas en el apartado anterior: es menos probable acertar un pase en el que el acercamiento a portería es elevado, o es más difícil acertar un pase si se efectúa con otra parte del cuerpo que no sean los pies. Nuestro criterio para decidir quién es mejor o peor pasador es ver qué jugador tiene más probabilidad de acierto en los pases que la media de las probabilidades estimadas (valores ajustados del modelo (5.2)) de los pases de ese jugador.

Pongo un ejemplo:

Supongamos que un jugador acierta 3 pases y falla 1. Entonces la probabilidad de acierto del jugador es de $\frac{3}{3+1} = 0{,}75$. Supongamos que mediante el modelo anterior (5.2), hemos predicho que la probabilidad de los pases es: 0.75, 0.5, 0.25, 0.75. Es decir, la probabilidad media estimada de este jugador de acertar los pases es de $\frac{0{,}75+0{,}5+0{,}25+0{,}75}{4} = 0{,}5625$. Como este jugador tiene más probabilidad de acierto que la predicha por el modelo, se puede decir que es buen pasador, ya que acierta más pases de los que debería acertar debido a las características de los pases.

En cambio, supongamos que otro jugador acierta y falla exactamente los mismos pases, es decir, acierta 3 pases y falla 1. La probabilidad de acierto del jugador es $\frac{3}{3+1} = 0{,}75$, la misma que en el caso anterior. Supongamos que las probabilidades de acierto de esos estimadas por el modelo (5.2) son: 0.9, 0.9, 0.75, 0.6. Calculando la media de las probabilidades estimadas es: $\frac{0{,}9+0{,}9+0{,}75+0{,}6}{4} = 0{,}7875$. Como tiene un porcentaje de acierto más bajo que el porcentaje estimado, se deduce que el jugador es peor pasador que lo esperado. Cuanto más positiva sea la diferencia entre el valor real y el valor esperado, mejor pasador es, ya que realiza más pases acertados de los que debería realizar. Cuanto más negativa sea la diferencia entre estos dos valores, el jugador falla más pases de los que debería fallar.

La Figura 6.1 muestra el valor real de cada jugador frente al valor medio estimado por el modelo (5.2):

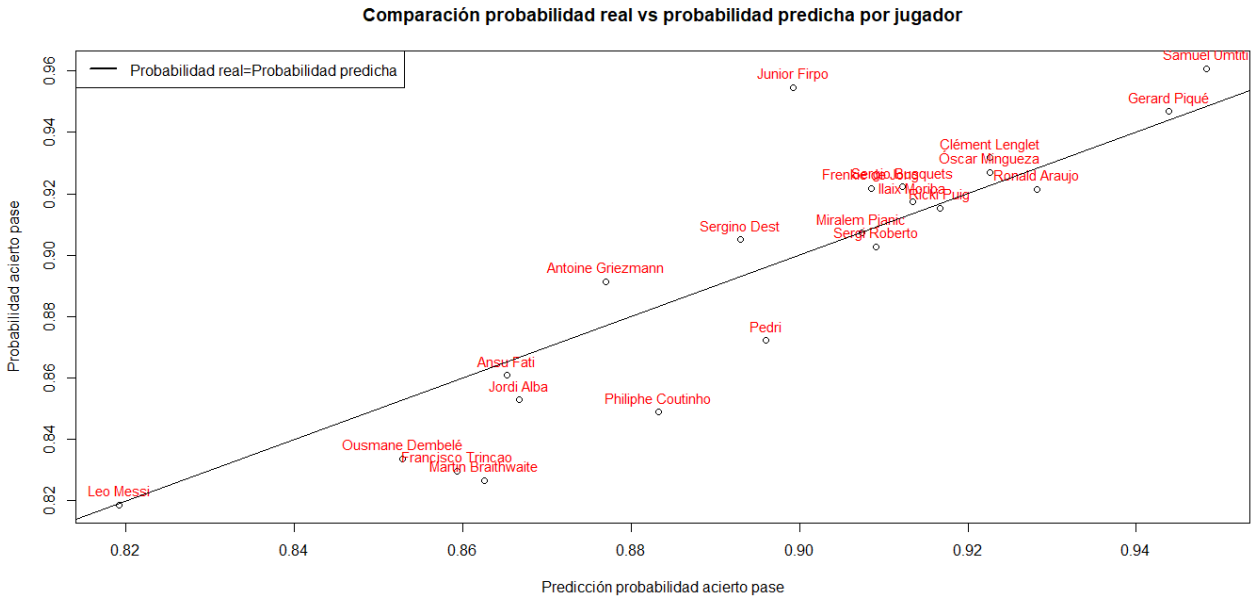


Figura 6.1: Comparación jugadores

Una primera lectura del gráfico se obtiene al mirar el eje de las X, que es el indicativo de la dificultad de un pase. Valores pequeños indican que el pase es difícil y valores grandes indican que el pase es sencillo. Así vemos, que Leo Messi realiza con bastante diferencia, los pases más difíciles, ya que la predicción de la probabilidad de acierto de sus pases es la más baja. Además se encuentra alejado de Ousmane Dembelé, siguiente jugador que realiza pases más difíciles. Notar que es Samuel Umtiti el jugador que realiza pases más sencillos.

Mirando los dos ejes, nos damos cuenta que el jugador que tiene menos probabilidad de acierto en los pases es Leo Messi, pero también tiene menor probabilidad estimada de acertar, ya que sus pases son de gran dificultad. En cambio, jugadores como Martin Braithwaite o Francisco Trincao, tienen probabilidades de acierto parecidas a las de Messi, pero su probabilidad estimada de acierto es muy superior, lo que indica que fallan muchos más pases de los que deberían fallar. Los jugadores que mas probabilidad de acierto tienen son los defensores como Samuel Umtiti, Gerard Piqué o Junio Firmo, ya que juegan en zonas muy defensivas y no arriesgan casi con la pelota. La diferencia entre el valor real y el valor esperado por el modelo se muestra en la Figura 6.2.

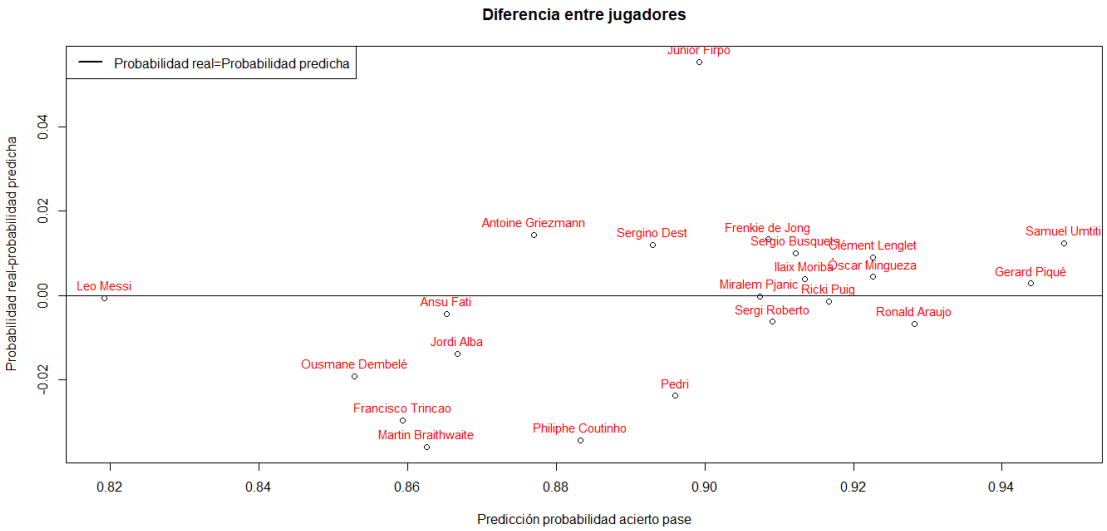


Figura 6.2: Diferencia entre probabilidad real y estimada por jugador

En este gráfico, cuanto mayor sea la diferencia entre la probabilidad real y la probabilidad estimada implica que eres mejor pasador, es decir que aciertas más pases de los que deberías debido a la dificultad de ellos mismos. El mejor pasador según la definición dada antes del Barcelona es Junior Firpo, seguido por Antoine Griezman y Frenkie De Jong. Los peores pasadores según la definición dada antes del Barcelona son Martin Braithwaite, Francisco Trincao y Philipe Coutinho. Notar que Junior Firpo, Martin Braithwaite, Francisco Trincao o Philipe Coutinho tienen un número de pases muy reducido, ya que no son importantes para el equipo y sólo juegan en determinadas ocasiones. Esto ha podido influir a la hora de predecir la probabilidad estimada ya que se ha tenido que realizar con un número pequeño de datos. Dentro de los jugadores que juegan habitualmente, (tienen más de 500 pases), los mejores pasadores son Antoine Griezmann y Frenkie De Jong. Los peores pasadores son Pedri y Ousmane Dembelé.

6.2. Relación entre dificultad y peligrosidad

En esta sección vamos a ver qué jugadores son peligrosos y compararlos con la dificultad de sus pases que realiza. Recordar que hemos definido pase peligroso si es uno de los tres últimos pases antes de un tiro. Nuestro objetivo no es averiguar si un pase es peligroso o no dependiendo de las características del pase, sino deducir qué jugadores realizan más pases peligrosos y qué jugadores realizan pocos pases peligrosos dependiendo de la distancia media del jugador a la portería rival (definida en las variables explicativas). Es decir, queremos ver las diferencias entre los jugadores quitando la influencia de dónde juega un jugador de media (ya que si un jugador juega de media cerca de la portería rival, se espera que sea más probable que realice más pases peligrosos que otro jugador que juega de media a una distancia mayor de la portería rival). Usamos un modelo logístico ya que la peligrosidad de un pase está definida como una variable binaria. Llamando Y a la peligrosidad del pase, el modelo es:

$$\log\left(\frac{P[Y=1|X_1=x_1]}{1-P[Y=1|X_1=x_1]}\right) = \beta_0 + \beta_1 * \text{distancia media del jugador a la portería rival} \qquad (6.1)$$

Notar que no incluimos los jugadores en el modelo ya que al igual que en el apartado anterior queremos ver las diferencia que existen entre ellos.

Validamos el modelo mediante un binnedplot de los residuos respecto la variable numérica distancia media del jugador a la portería rival (para ver si necesitamos introducir al modelo términos cuadráticos) (Figura 10 del anexo B)

Cómo se observa, no presentan ninguna tendencia por lo que no incluimos más términos a nuestro modelo.

El resumen de este modelo se muestra en la Figura 11 del anexo B

Notar que el coeficiente de la distancia media del jugador es negativo, es decir, cuanto mayor sea la distancia media de un jugador a la portería rival, es menos probable que el pase sea peligroso.

Para ver la diferencia entre los jugadores vamos a usar la misma técnica que usamos con el modelo (5.2). Para cada pase se tiene un valor ajustado de la probabilidad de que un pase sea peligroso dependiendo de la distancia media del jugador a la portería rival. Para cada jugador hago la media de los valores ajustados que corresponden a sus pases. Y calculo el porcentaje real de pases peligrosos dividiendo el número de pases peligrosos entre el número de pases totales. Representando la diferencia entre el valor real y el valor ajustado frente a los valores ajustados en la Figura 6.3.

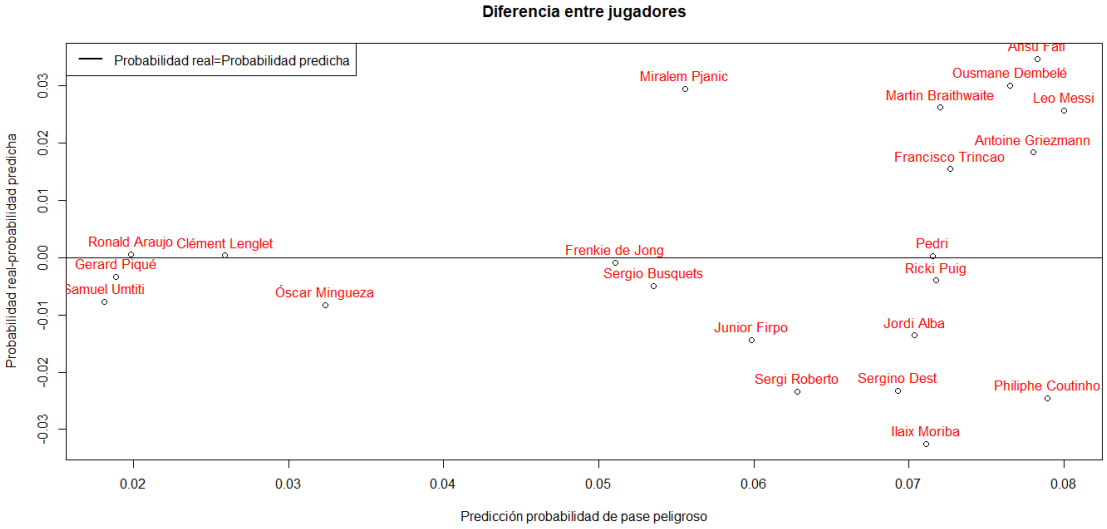


Figura 6.3: Diferencias entre la peligrosidad de los jugadores

En primer lugar, el eje X indica la peligrosidad estimada de un pase según el modelo (6.1). El jugador más peligroso es Leo Messi seguido muy de cerca por Ansu Fati y Antoine Griezmann y el menos peligroso es Samuel Umtiti seguido de Gerard Piqué.

Si la diferencia entre el valor real y el predicho es positiva implica que haces más pases peligrosos en comparación con los que deberías hacer según el modelo (6.1), es decir, debido a la posición en la que un jugador juega de media. (recogida en la variable distancia media a la portería rival). Ansu Fati es el jugador más peligroso en comparación con lo que se esperaba según el modelo. En cambio, Ilaix Moriba es el jugador menos peligroso.

En la Figura 6.4 se representa en el eje X la diferencia entre la probabilidad de acierto de un jugador menos la probabilidad estimada y en el eje Y los valores correspondientes a su peligrosidad.

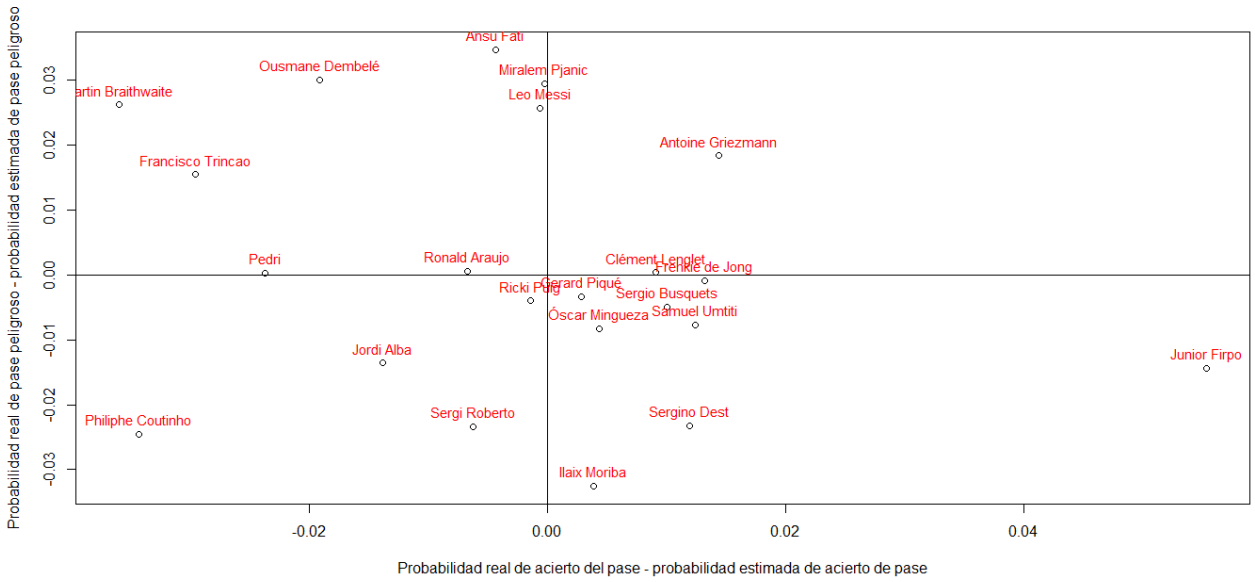


Figura 6.4: Comparación de la dificultad y la peligrosidad de los jugadores

Al observar el gráfico, en primer lugar nos damos cuenta que cuanto más a la derecha se encuentre un jugador implica que es mejor pasador. Por el contrario, si un jugador se encuentra a la izquierda implica que es un mal pasador. Cuánto más arriba se encuentre un jugador, implica que es un jugador peligroso, y por el contrario, si un jugador se encuentra en la parte baja del gráfico implica que es poco peligroso. Por tanto, los mejores jugadores en cuanto a la relación acierto de pase-peligrosidad se encuentran en el cuadrante de arriba a la derecha. Antoine Griezmann es el jugador que es mejor pasador a la vez que es peligroso. Otros, como Leo Messi, es un jugador muy peligroso, pero es un pasador normal, ya que se espera que acierte parecidos pases a los que acierta en realidad. En cambio Philipe Coutinho es un jugador que falla muchos más pases de los que le corresponde (mal pasador) y además es mucho menos peligroso de lo que debería, por lo que concluimos que es un mal jugador en este aspecto.

6.3. Conclusión

Hemos sido capaces de encontrar un modelo de regresión logística para la probabilidad de éxito de los pases del Barcelona con buena capacidad predictiva. Notar que hemos visto que los jugadores tienen mucha importancia en nuestro modelo, por lo que sí que es importante quién realiza el pase.

Después hemos realizado un modelo auxiliar para deducir qué jugadores son peligrosos y cuáles no dependiendo de la distancia a la portería rival.

Gracias a estos modelos, podemos identificar jugadores más o menos precisos y más o menos peligrosos quitando la influencia de las variables regresoras de los modelos.

Anexos

A. Tablas de las variables explicativas

	Sin presión	Con presión
Número	19181	2595
Porcentaje	88.08	11.92

Cuadro 1: Tablas de pases realizados sin presión y con presión

	No duelo aéreo	Duelo aéreo
Número	21612	164
Porcentaje	99.25	0.75

Cuadro 2: Tablas de pases realizados tras un duelo aéreo

	Número de pases	Porcentaje del total	Pases acertados(%)	Pases peligrosos(%)
Junior Firpo	176	0.81	95.5	4.5
Sergino Dest	1042	4.79	90.5	4.6
Francisco Trincao	329	1.51	83	8.8
Ricki Puig	413	1.90	91.5	6.8
Pedri	1448	6.65	87.2	7.2
Ansu Fati	115	0.53	86.1	11.3
Ronald Araujo	636	2.92	92.1	2
Philiphe Coutinho	331	1.52	84.9	5.4
Ilaix Moriba	363	1.67	91.7	3.9
Óscar Mingueza	1328	6.10	92.7	2.4
Martin Braithwaite	173	0.79	82.7	9.8
Sergio Busquets	2261	10.38	92.2	4.9
Jordi Alba	2325	10.68	85.3	5.7
Gerard Piqué	1032	4.74	94.7	1.6
Ousmane Dembelé	854	3.92	83.4	10.7
Antoine Griezmann	1068	4.90	89.1	9.6
Samuel Umtiti	382	1.75	96.1	1
Leo Messi	2034	9.34	81.9	10.6
Sergi Roberto	761	3.49	90.3	3.9
Clément Lenglet	1977	9.08	93.2	2.6
Miralem Pjanić	517	2.37	90.7	8.5
Frenkie de Jong	2211	10.15	92.2	5

Cuadro 3: Tabla del número de pases por jugador, porcentaje del total de pases realizados por el jugador, porcentaje de pases acertados y porcentaje de pases peligrosos

B. Gráficas de los modelos

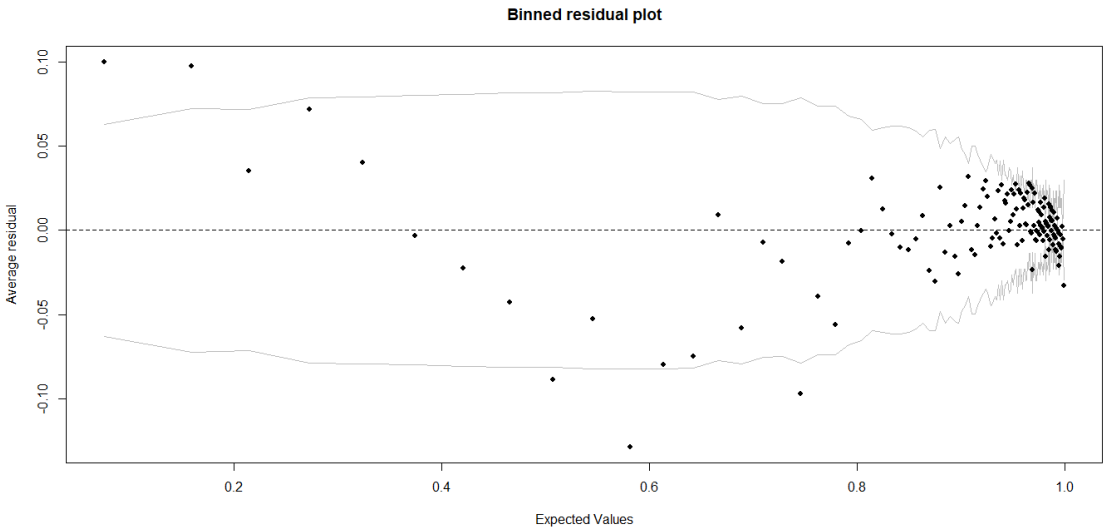


Figura 5: Binnedplot del modelo *logit* (5.2) residuos frente valores ajustados

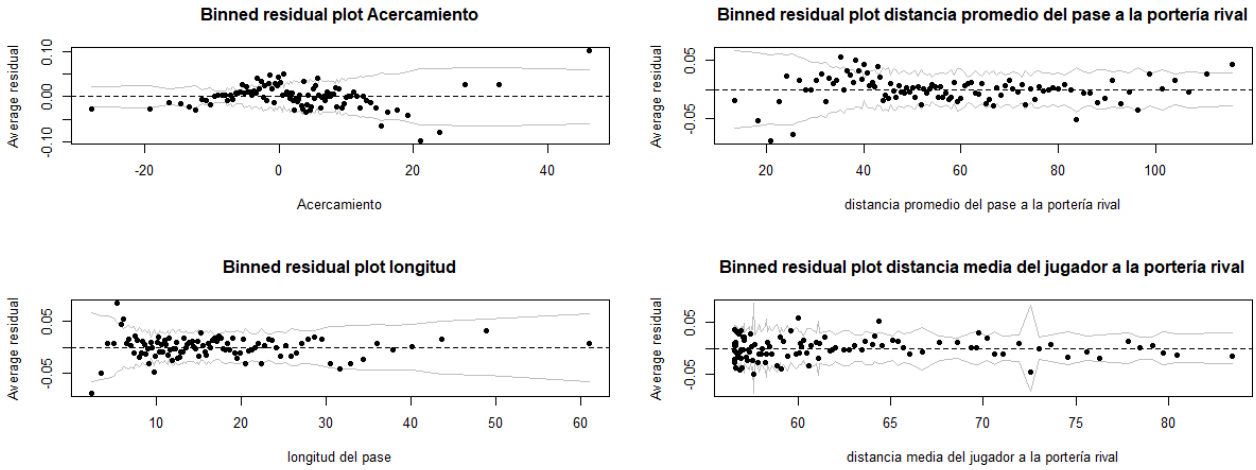


Figura 6: Binnedplot de los residuos del modelo (5.2) frente a las variables dependientes

```
Coefficients:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                    -6.96944    0.39613   -17.6   <2e-16 ***
promdis                         0.12240    0.00521    23.5   <2e-16 ***
promdis2                       -0.00078    0.00004   -19.3   <2e-16 ***
acercamiento                   -0.11272    0.00367   -30.7   <2e-16 ***
longitud                        0.54362    0.02298    23.7   <2e-16 ***
largo[T.largo]                  4.78698    0.16502    29.0   <2e-16 ***
campo[T.TRUE]                  -0.63024    0.11063    -5.7    1e-08 ***
da[T.1]                        -1.18941    0.21081    -5.6    2e-08 ***
parte[T.1]                      1.89512    0.12166    15.6   <2e-16 ***
under_pressure[T.TRUE]         -0.78803    0.07539   -10.5   <2e-16 ***
jugador[T.Antoine Griezmann]    0.26640    0.34304     0.8    0.437
jugador[T.Clément Lenglet]      0.72375    0.34226     2.1    0.034 *
jugador[T.Francisco Trincao]   -0.26169    0.37148    -0.7    0.481
jugador[T.Frenkie de Jong]      0.47373    0.33535     1.4    0.158
jugador[T.Gerard Piqué]         0.70126    0.36187     1.9    0.053 .
jugador[T.Ilaix Moriba]         0.16914    0.39385     0.4    0.668
jugador[T.Jordi Alba]          -0.06922    0.33018    -0.2    0.834
jugador[T.Junior Firpo]         1.41052    0.54137     2.6    0.009 **
jugador[T.Leo Messi]            0.03565    0.32946     0.1    0.914
jugador[T.Martin Braithwaite] -0.35324    0.41594    -0.8    0.396
jugador[T.Miralem Pjanic]       0.19198    0.36897     0.5    0.603
jugador[T.Óscar Mingueza]       0.51973    0.34808     1.5    0.135
jugador[T.Ousmane Dembelé]     -0.16514    0.34146    -0.5    0.629
jugador[T.Pedri]               -0.25225    0.33548    -0.8    0.452
jugador[T.Philiphe Coutinho]   -0.40422    0.37384    -1.1    0.280
jugador[T.Ricki Puig]           0.06684    0.38080     0.2    0.861
jugador[T.Ronald Araujo]        0.49356    0.37692     1.3    0.190
jugador[T.Samuel Umtiti]        1.00101    0.43610     2.3    0.022 *
jugador[T.Sergi Roberto]        0.06716    0.35256     0.2    0.849
jugador[T.Sergino Dest]         0.29836    0.34541     0.9    0.388
jugador[T.Sergio Busquets]      0.40097    0.33573     1.2    0.232
longitud:largo[T.largo]        -0.54822    0.02312   -23.7   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 14692.6  on 21775  degrees of freedom
Residual deviance:  9666.4  on 21744  degrees of freedom
AIC: 9730

Number of Fisher Scoring iterations: 6
```

Figura 7: Resumen del modelo *logit* (5.3)

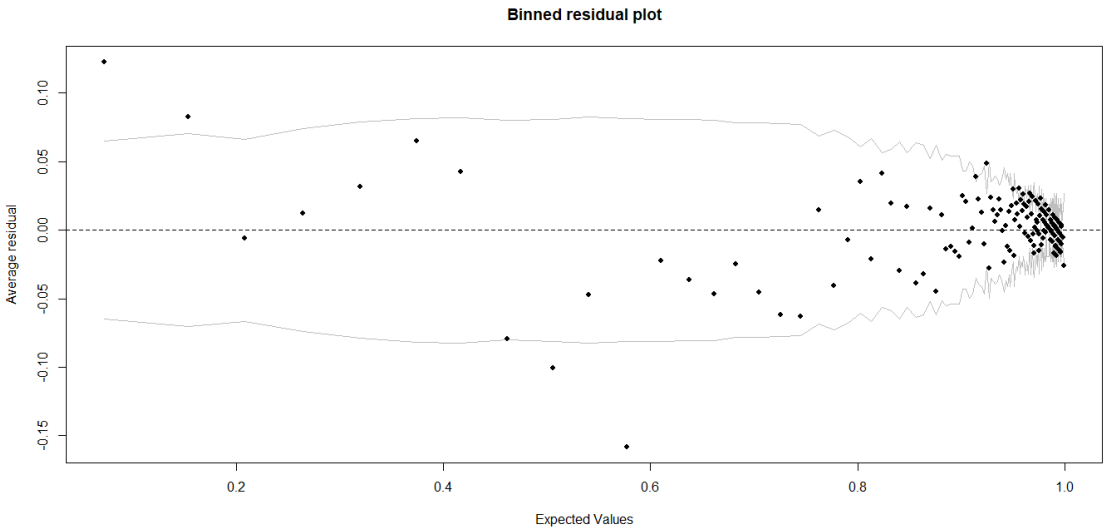


Figura 8: Binnedplot modelo *logit* (5.3) entre residuos y valores ajustados

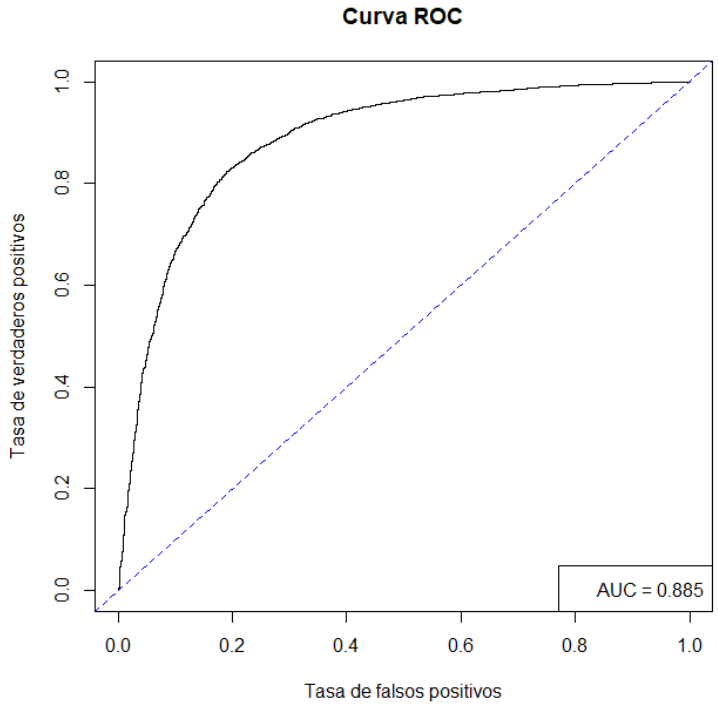


Figura 9: Curva Roc modelo *logit* (5.3)

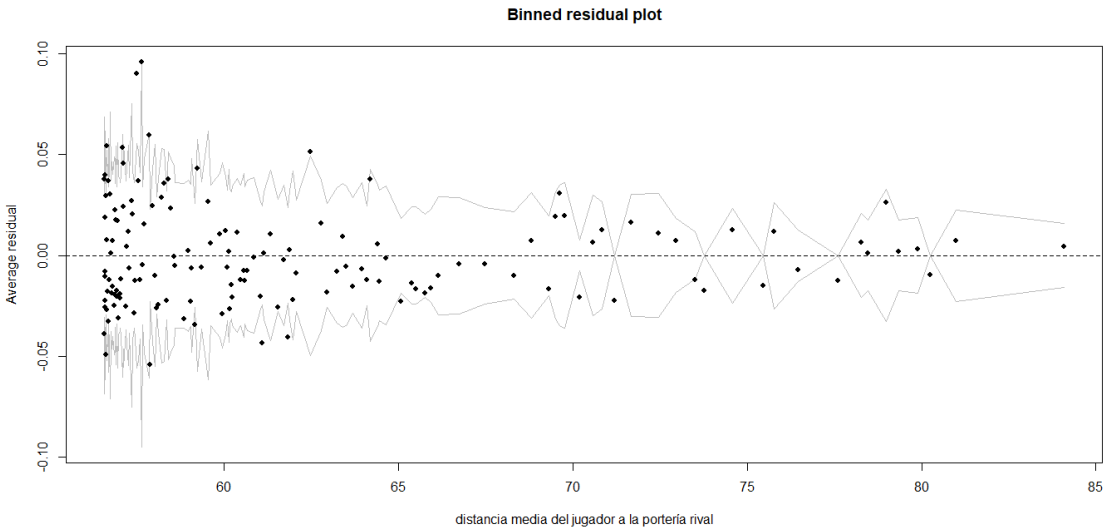
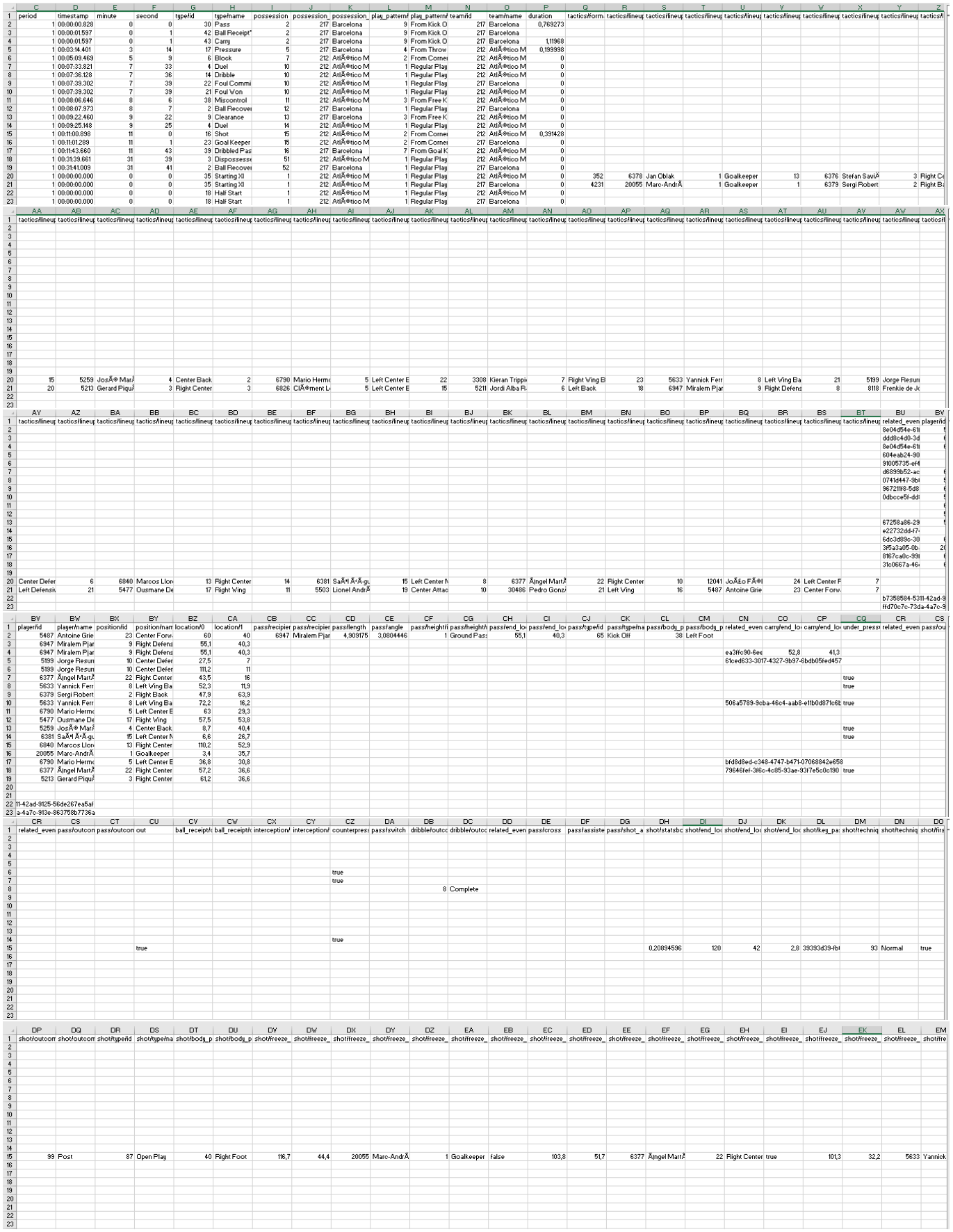


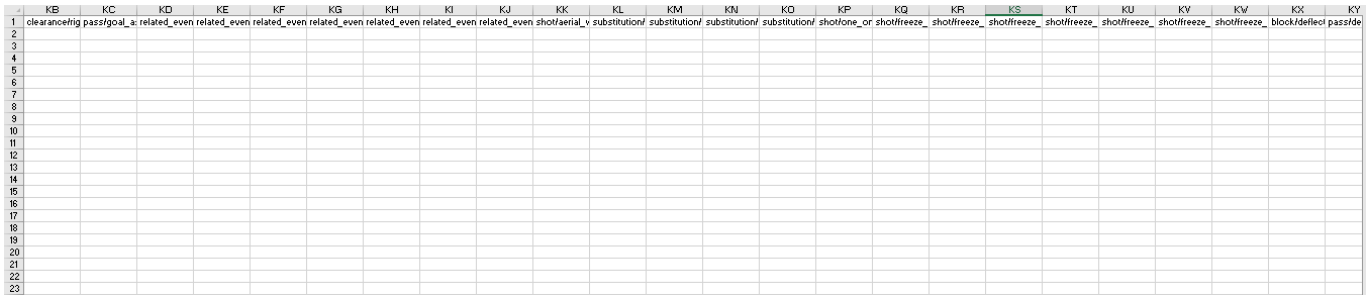
Figura 10: Gráfico binned de los residuos del modelo (6.1) frente a la distancia media del jugador a la portería

```
Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)        2.91749    0.40043   7.286  3.2e-13 ***
mediajugadordistan -0.09404    0.00668 -14.078  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 9470.0  on 21775  degrees of freedom
Residual deviance: 9195.2  on 21774  degrees of freedom
AIC: 9199.2

Number of Fisher Scoring iterations: 6
```

Figura 11: Resumen del modelo (6.1)

C. Ejemplo de fila de datos





	KB	KC	KD	KE	KF	KG	KH	KI	KJ	KK	KL	KM	KN	KO	KP	KQ	KR	KS	KT	KU	KV	KW	KX	KY
1	clearance/rig	pass/goal_a	related_even	related_even	related_even	related_even	related_even	related_even	related_even	shot/aerial_v	substitution/	substitution/	substitution/	substitution/	shot/one_or	shot/freeze_	shot/freeze_	shot/freeze_	shot/freeze_	shot/freeze_	shot/freeze_	shot/freeze_	block/deflect	pass/de
2																								
3																								
4																								
5																								
6																								
7																								
8																								
9																								
10																								
11																								
12																								
13																								
14																								
15																								
16																								
17																								
18																								
19																								
20																								
21																								
22																								
23																								

Figura 12: Ejemplo base de datos de un partido

D. Programas de java

D.1. Filtrado.java

```
package segunda;

import java.io.File;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.Scanner;

public class filtrado {
    public static String primeralinea;
    public static void filtro(String nombreFich) {
        File fentrada = new File(nombreFich);
        File ffiltrado = new File("realmadrid-barcelonafiltrado.csv");
        File fsalida = new File("filtrado.csv");
        Scanner in = null;
        PrintWriter out = null;
        String aux = "";
        int i = 0;
        ArrayList<String> auxlis1 = new ArrayList<>();
        ArrayList<Integer> auxlis2 = new ArrayList<>();
        try {
            try {
                in = new Scanner(ffiltrado);
                out = new PrintWriter(fsalida);
                primeralinea = in.nextLine();
                primeralinea = primeralinea + ";";
                while (aux.equals("pass/shot_assist") == false) {
                    aux = Pase.tipo(i, primeralinea);
                    auxlis1.add(aux);
                    i++;
                }
                in = new Scanner(fentrada);
                String linea = in.nextLine();
                linea = linea + ";";
                i = 0;
                out.print("index;");
                int k = 1;
                while (aux.equals("shot/freeze_frame/15/player/id") == false) {
                    aux = Pase.tipo(i, linea);
                    if (auxlis1.contains(aux)) {
                        auxlis2.add(i);
```

```
                    out.print(aux + ";");
                }
                i++;
            }
            out.println("");
            while (in.hasNext()) {
                out.print(k + ";");
                linea = in.nextLine();
                linea = linea + ";";
                for (int j = 0; j < auxlis2.size(); j++) {
                    out.print(Pase.tipo(auxlis2.get(j), linea));
                    out.print(";");
                }
                out.println("");
                k++;
            }
        } finally {
            if (in != null) {
                in.close();
            }
            if (out != null) {
                out.close();
            }
        }
    } catch (Exception e) {
        System.out.println("Error: " + e.getMessage());
        if (fsalida.exists()) {
            fsalida.delete();
        }
    }
}
```

D.2. Peligrosidad.java

```
package segunda;
import java.io.File;
import java.io.PrintWriter;
import java.util.Scanner;
public class Peligrosidad {
    static String primeralinea;
    public static void peligro(String nombreFich) {
        File fentrada = new File(nombreFich);
        File fsalida = new File("__auxiliarpeligro.csv");
        Scanner in = null, in1 = null;
        PrintWriter out = null;
        String linea = "", linea1 = "", cont1 = "", cont2, tipo1 = "", tipo2 = "";
        boolean aux = false;
        try {
            try {
                in = new Scanner(fentrada);
                out = new PrintWriter(fsalida);
                primeralinea = in.nextLine();
                out.println(primeralinea + "peligrosidad;");
                int j;
                int contador = 0;
```

```
            while (in.hasNext()) {
                contador++;
                linea = in.nextLine();
                out.print(linea);
                if (Pase.pruebatipo(3, linea, primeralinea).equals("30") &&
                    Pase.pruebatipo(7, linea, primeralinea).equals("217")) {
                    in1 = new Scanner(fentrada);
                    in1.nextLine();
                    for (int i = 0; i < contador; i++) {
                        in1.nextLine();
                    }
                    aux = false;
                    j = 0;
                    while (aux == false) {
                        if (in1.hasNextLine()) {
                            linea1 = in1.nextLine();
                            if (Pase.pruebatipo(3, linea1,
                                primeralinea).equals("16")) {
                                if (Pase.pruebatipo(7, linea1,
                                    primeralinea).equals("217")) {
                                    aux = true;
                                    out.print("1" + ";");
                                } else {
                                    out.print("0" + ";");
                                }
                            }
                            if (Pase.pruebatipo(3, linea1,
                                primeralinea).equals("30")) {
                                j++;
                            }
                            if (j == 3) {
                                aux = true;
                                out.print("0" + ";");
                            }
                        } else {
                            aux = true;
                            out.print("0;");
                        }
                    }
                    out.println();
                } else {
                    out.println("0;");
                }
            }
        } finally {
            if (in != null) {
                in.close();
            }
            if (out != null) {
                out.close();
            }
        }
    } catch (Exception e) {
        System.out.println("Error: " + e.getMessage());
        if (fsalida.exists()) {
            fsalida.delete();
        }
    }
```

```
    }
}
```

D.3. Duelo.java

```
package segunda;

import java.io.File;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.Collections;
import java.util.Scanner;

public class duelos {

    static String primeralinea;

    public static void duelo(String nombreFich) {
        File fentrada = new File(nombreFich);
        File fsalida = new File("__auxiliarduelo.csv");
        Scanner in = null;
        PrintWriter out = null;
        String linea = "", linea1 = "", cont1 = "", cont2, tipo1 = "", tipo2 = "";
        boolean aux = false;
        try {
            try {
                in = new Scanner(fentrada);
                out = new PrintWriter(fsalida);
                primeralinea = in.nextLine();
                out.println(primeralinea + "dt;da;ds;");
                int i = 0;
                while (in.hasNext()) {
                    aux = false;
                    linea = in.nextLine();
                    out.print(linea);
                    if (Pase.pruebatipo(3, linea, primeralinea).equals("4")) {
                        tipo1 = Pase.pruebatipo(29, linea, primeralinea);
                        tipo2 = Pase.pruebatipo(31, linea, primeralinea);
                        cont1 = Pase.pruebatipo(0, linea, primeralinea);
                    }
                    if (Pase.pruebatipo(3, linea, primeralinea).equals("30")) {
                        aux = false;
                        cont2 = Pase.pruebatipo(0, linea, primeralinea);
                        if (tipo2.equals("10")) {
                            if (Integer.parseInt(cont2) - Integer.parseInt(cont1) == 1
                                && cont1 != null) {
                                out.println("0;1;1;");
                                tipo2 = "";
                                tipo1 = "";
                                aux = true;
                            }
                        }
                        if ((tipo1.equals("1") || tipo1.equals("4"))) {
                            if (Integer.parseInt(cont2) - Integer.parseInt(cont1) <= 3
                                && cont1 != null) {
                                out.println("1;0;1;");
```

```
                            tipo2 = "";
                            tipo1 = "";
                            aux = true;
                        }
                    }
                    if ((tipo1.equals("13") || tipo1.equals("16"))) {
                        if (Integer.parseInt(cont2) - Integer.parseInt(cont1) <= 3
                            && cont1 != null) {
                            out.println("1;0;0;");
                            tipo2 = "";
                            tipo1 = "";
                            aux = true;
                        }
                    }
                    if (aux == false) {
                        out.println("0;0;0;");
                    }
                } else {
                    out.println("0;0;0;");
                }
            }
        } finally {
            if (in != null) {
                in.close();
            }
            if (out != null) {
                out.close();
            }
        }
    } catch (Exception e) {
        System.out.println("Error: " + e.getMessage());
        if (fsalida.exists()) {
            fsalida.delete();
        }
    }
}
```

D.4. Posicion.java

```
package segunda;

import java.io.File;
import java.io.PrintWriter;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Scanner;

public class Posicion {

    static String primeralinea;

    public static void posi(String nombreFich) {
        File fentrada = new File(nombreFich);
```

```
File fsalida = new File("__auxiliar__posi.csv");
Scanner in = null;
String linea = null, aux = null, aux1 = null;
double aux4;
boolean fin = false;
double numero;
List<Double> lista = new ArrayList<>();
HashMap<String, List<Double>> mapax = new HashMap<>();
HashMap<String, Double> mapafinalx = new HashMap<>();
HashMap<String, List<Double>> mapay = new HashMap<>();
HashMap<String, Double> mapafinaly = new HashMap<>();
try {
    in = new Scanner(fentrada);
    primeralinea = in.nextLine();
    while (in.hasNext()) {
        linea = in.nextLine();
        aux = Pase.pruebatipo(10, linea, primeralinea);
        if (aux.equals("") == false) {
            if (mapax.containsKey(aux)) {
                lista = new ArrayList<>();
                lista = mapax.get(aux);
                String aux2 = Pase.pruebatipo(14, linea, primeralinea);
                if (aux2.equals("") == false) {
                    numero = Double.parseDouble(Posicion.puntocoma(aux2));
                    lista.add(numero);
                }
            } else {
                lista = new ArrayList<>();
                String aux2 = Pase.pruebatipo(14, linea, primeralinea);
                if (aux2.equals("") == false) {
                    numero = Double.parseDouble(Posicion.puntocoma(aux2));
                    lista.add(numero);
                    mapax.put(aux, lista);
                }
            }
        }
        aux = Pase.pruebatipo(10, linea, primeralinea);
        if (aux.equals("") == false) {
            if (mapay.containsKey(aux)) {
                lista = new ArrayList<>();
                lista = mapay.get(aux);
                String aux2 = Pase.pruebatipo(15, linea, primeralinea);
                if (aux2.equals("") == false) {
                    numero = Double.parseDouble(Posicion.puntocoma(aux2));
                    lista.add(numero);
                }
            } else {
                lista = new ArrayList<>();
                String aux2 = Pase.pruebatipo(15, linea, primeralinea);
                if (aux2.equals("") == false) {
                    numero = Double.parseDouble(Posicion.puntocoma(aux2));
                    lista.add(numero);
                    mapay.put(aux, lista);
                }
            }
        }
    }
} catch (Exception e) {
```

```
            System.out.println("Error: " + e.getMessage());
            if (fsalida.exists()) {
                fsalida.delete();
            }
        } finally {
            if (in != null) {
                in.close();
            }
        }
        mapafinalx = Posicion.media(mapax);
        mapafinaly = Posicion.media(mapay);

        try {
            PrintWriter out = new PrintWriter(fsalida);
            in = new Scanner(fentrada);
            out.print(in.nextLine());
            out.println("mediax;mediay;");
            while (in.hasNext()) {
                linea = in.nextLine();
                if (Pase.pruebatipo(4, linea, primeralinea).equals("Pass")) {
                    out.print(linea);
                    aux = Pase.pruebatipo(10, linea, primeralinea);
                    aux = Posicion.comapunto(aux);
                    aux4 = mapafinalx.get(aux);
                    aux1 = String.valueOf(aux4);
                    out.print(aux1 + ";");
                    aux = Pase.pruebatipo(10, linea, primeralinea);
                    aux = Posicion.comapunto(aux);
                    aux4 = mapafinaly.get(aux);
                    aux1 = String.valueOf(aux4);
                    out.print(aux1 + ";");
                    out.println();
                }
            }
            out.flush();
        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
            if (fsalida.exists()) {
                fsalida.delete();
            }
        } finally {
            if (in != null) {
                in.close();
            }
        }
    }

    public static String puntocoma(String j) {
        String aux1 = "";
        String aux2;
        String concat;
        char a;
        for (int i = 0; i < j.length(); i++) {
            a = j.charAt(i);
            aux2 = String.valueOf(a);
            if (aux2.equals(",") == false) {
                concat = aux1.concat(aux2);
                aux1 = concat;
```

```
            } else {
                concat = aux1.concat(".");
                aux1 = concat;
            }
        }
        return aux1;
    }

    public static String comapunto(String j) {
        String aux1 = "";
        String aux2;
        String concat;
        char a;
        for (int i = 0; i < j.length(); i++) {
            a = j.charAt(i);
            aux2 = String.valueOf(a);
            if (aux2.equals(".") == false) {
                concat = aux1.concat(aux2);
                aux1 = concat;
            } else {
                concat = aux1.concat(",");
                aux1 = concat;
            }
        }
        return aux1;
    }

    public static HashMap<String, Double> media(Map<String, List<Double>> m) {
        HashMap<String, Double> mapa = new HashMap<>();
        List<Double> lista = new ArrayList<>();
        double suma = 0;
        for (String x : m.keySet()) {
            lista = new ArrayList<>();
            lista = m.get(x);
            for (int i = 0; i < lista.size(); i++) {
                suma = suma + lista.get(i);
            }
            suma = suma / lista.size();
            mapa.put(x, suma);
        }
        return mapa;
    }
}
```

D.5. Pase.java

```
package segunda;

import java.io.*;
import java.util.*;

public class Pase {

    static String primeralinea;

    public static void ordenar(String nombreFich) {
```

```
File fentrada = new File(nombreFich);
File fsalida = new File("pase.csv");
Scanner in = null;
String linea = "", aux = "", aux0 = "", aux1 = "";
PrintWriter out = null;
boolean fin = false;
ArrayList<String> auxlis1 = new ArrayList<>();
ArrayList<String> auxlis2 = new ArrayList<>();
ArrayList<String> auxlis3 = new ArrayList<>();
int k = 0, min0, min1, seg0, seg1, dec0, dec1;
try {
    try {
        in = new Scanner(fentrada);
        out = new PrintWriter(fsalida);
        primeralinea = in.nextLine();
        out.println(primeralinea +
            "timestamp;minute;possession_team/id;possession_team/name;team/id;team/name;"+
               "player/id;player/name;pass/recipient/id;pass/recipient/name;" +
                   "numero;tiempo;" + "under_pressure;" + "acertado-fallado;" +
                   "location0;" + "location1;" + "endlocation0;" +
                   "endlocation1;" + "parte");
        while (in.hasNext()) {
            linea = in.nextLine();
            if (Pase.pruebatipo(4, linea, primeralinea).equals("Pass")) {
                aux = Pase.pruebatipo(1, linea, primeralinea);
                auxlis3.add(aux);
                out.print(linea);
                aux = Pase.pruebatipo(1, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(2, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(5, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(6, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(7, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(8, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(10, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(11, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(16, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(17, linea, primeralinea);
                out.print(aux + ";");
                aux = Pase.pruebatipo(8, linea, primeralinea);
                auxlis1.add(aux);
                if (auxlis1.size() >= 2) {
                    if (auxlis1.get(auxlis1.size() -
                        1).equals(auxlis1.get(auxlis1.size() - 2))) {
                        fin = true;
                    } else {
                        fin = false;
                    }
                }
                aux = Pase.pruebatipo(20, linea, primeralinea);
```

```
if (aux.equals("") && fin == true) {
} else {
    fin = false;
}
aux = Pase.pruebatipo(25, linea, primeralinea);
auxlis2.add(aux);
if (auxlis2.size() >= 2) {
    if (auxlis2.get(auxlis1.size() - 2).equals("") && fin ==
        true) {
        fin = true;
    } else {
        fin = false;
    }
}
if (fin == true) {
    k++;
    out.print(k);
    out.print(";");
    if (auxlis3.size() >= 2) {
        aux = Pase.reloj(auxlis3);
        out.print(aux);
        out.print(";");
    }
} else {
    out.print("0;");
    out.print("0;");
    k = 0;
}
if (Pase.pruebatipo(24, linea, primeralinea).equals("")) {
    out.print("false;");
} else {
    out.print("true;");
}
if (Pase.pruebatipo(25, linea, primeralinea).equals("")) {
    out.print("1;");
} else {
    out.print("0;");
}
String aux23;
aux23 = Pase.pruebatipo(14, linea, primeralinea);
aux23 = Posicion.puntocoma(aux23);
out.print(aux23 + ";");
aux23 = Pase.pruebatipo(15, linea, primeralinea);
aux23 = Posicion.puntocoma(aux23);
out.print(aux23 + ";");
aux23 = Pase.pruebatipo(18, linea, primeralinea);
aux23 = Posicion.puntocoma(aux23);
out.print(aux23 + ";");
aux23 = Pase.pruebatipo(19, linea, primeralinea);
aux23 = Posicion.puntocoma(aux23);
out.print(aux23 + ";");
if (Pase.pruebatipo(22, linea, primeralinea).equals("40") ||
    Pase.pruebatipo(22, linea, primeralinea).equals("38") ||
    Pase.pruebatipo(22, linea, primeralinea).equals("69") ||
    Pase.pruebatipo(22, linea, primeralinea).equals("")) {
    out.print("1;");
} else {
    out.print("0;");
```

```
                    }
                    out.println();
                }
            }
        } finally {
            if (in != null) {
                in.close();
            }
            if (out != null) {
                out.close();
            }
        }
    } catch (Exception e) {
        System.out.println("Error: " + e.getMessage());
        if (fsalida.exists()) {
            fsalida.delete();
        }
    }
}

public static int numero1(int numero, String linea) {
    int cont = 0;
    int i = 0;
    while (cont != numero) {
        if (linea.charAt(i) == ':') {
            cont++;
        }
        i++;
    }
    return i;
}

public static String tipo1(int numero, String linea) {
    int aux1, aux2;
    String aux;
    aux1 = Pase.numero1(numero, linea);
    aux2 = Pase.numero1(numero + 1, linea);
    aux = linea.substring(aux1, aux2 - 1);
    return aux;
}

public static String puntodospuntos(String j) {
    String aux1 = "";
    String aux2;
    String concat;
    char a;
    for (int i = 0; i < j.length(); i++) {
        a = j.charAt(i);
        aux2 = String.valueOf(a);
        if (aux2.equals(".") == false) {
            concat = aux1.concat(aux2);
            aux1 = concat;
        } else {
            concat = aux1.concat(":");
            aux1 = concat;
        }
    }
    aux1 = aux1 + ":";
```

```
        return aux1;
    }

    public static String reloj(List<String> auxlis3) {
        String aux0 = auxlis3.get(auxlis3.size() - 2);
        String fin;
        String aux1 = auxlis3.get(auxlis3.size() - 1);
        aux0 = Pase.puntodospuntos(aux0);
        aux1 = Pase.puntodospuntos(aux1);
        int min = 0, seg = 0, dec = 0;
        int min0 = Integer.parseInt(Pase.pruebatipo1(1, aux0, primeralinea));
        int min1 = Integer.parseInt(Pase.pruebatipo1(1, aux1, primeralinea));
        int seg0 = Integer.parseInt(Pase.pruebatipo1(2, aux0, primeralinea));
        int seg1 = Integer.parseInt(Pase.pruebatipo1(2, aux1, primeralinea));
        int dec0 = Integer.parseInt(Pase.pruebatipo1(3, aux0, primeralinea));
        int dec1 = Integer.parseInt(Pase.pruebatipo1(3, aux1, primeralinea));
        boolean c1 = false, c2 = false;
        dec = dec1 - dec0;
        if (dec < 0) {
            dec = 1000 + dec;
            c1 = true;
        }
        seg = seg1 - seg0;
        if (seg < 0) {
            seg = 60 + seg;
            c2 = true;
        }
        if (c1 == true) {
            seg = seg - 1;
        }
        fin = String.valueOf(seg) + "." + String.valueOf(dec);
        return fin;
    }

    public static int numero(int numero, String linea) {
        int cont = 0;
        int i = 0;
        while (cont != numero) {
            if (linea.charAt(i) == ';') {
                cont++;
            }
            i++;
        }
        return i;
    }

    public static String tipo(int numero, String linea) {
        int aux1, aux2;
        String aux;
        aux1 = Pase.numero(numero, linea);
        aux2 = Pase.numero(numero + 1, linea);
        aux = linea.substring(aux1, aux2 - 1);
        return aux;
    }

    public static String pruebatipo(int numero, String linea, String primera1) {
        File ffiltrado = new File("realmadrid-barcelonafiltrado.csv");
        Scanner in = null;
```

```
        String aux = "", primera, columna = null;
        int i = 0;
        try {
            try {
                in = new Scanner(ffiltrado);
                primera = in.nextLine();
                columna = Pase.tipo(numero, primera);
                if (numero == 0) {
                    columna = columna.substring(1, columna.length());
                }
                while (aux.equals(columna) == false) {
                    aux = Pase.tipo(i, primera1);
                    i++;
                }
                i = i - 1;
            } finally {
                if (in != null) {
                    in.close();
                }
            }
        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }
        return Pase.tipo(i, linea);
    }

    public static String pruebatipo1(int numero, String linea, String primera1) {
        File ffiltrado = new File("realmadrid-barcelonafiltrado.csv");
        Scanner in = null;
        String aux = "", primera, columna;
        int i = 0;
        try {
            try {
                in = new Scanner(ffiltrado);
                primera = in.nextLine();
                columna = Pase.tipo(numero, primera);
                while (aux.equals(columna) == false) {
                    aux = Pase.tipo(i, primera1);
                    i++;
                }
                i = i - 1;
            } finally {
                if (in != null) {
                    in.close();
                }
            }
        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
        }
        return Pase.tipo1(i, linea);
    }

}
```

D.6. Eliminado.java

```
package segunda;

import java.io.File;
import java.io.PrintWriter;
import java.util.Scanner;
import static segunda.duelos.primeralinea;

public class eliminado {

    public static void eliminar(String nombreFich) {
        File fentrada = new File(nombreFich);
        File fsalida = new File("final.csv");
        Scanner in = null;
        PrintWriter out = null;
        String aux = "";
        try {
            try {
                in = new Scanner(fentrada);
                out = new PrintWriter(fsalida);
                while (in.hasNext()) {
                    aux = in.nextLine();
                    aux=aux+";";
                    for (int i = 37; i < 62; i++) {
                        out.print(Pase.tipo(i, aux)+";");
                    }
                    out.println();
                }
            } finally {
                if (in != null) {
                    in.close();
                }
                if (out != null) {
                    out.close();
                }
            }
        } catch (Exception e) {
            System.out.println("Error: " + e.getMessage());
            if (fsalida.exists()) {
                fsalida.delete();
            }
        }
    }
}
```

E. Código de R

```
library(readr)
final <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/alaves-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                 mediay = col_number(),
                                                                     timestamp =
                                                                     col_time(format =
                                                                     "%H:%M:%S")),
                    trim_ws = TRUE)
final <- within(final, {
  X26 <- NULL
```

```
})

final1 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/atletic-barcelona/final.csv",
                     ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                  mediay = col_number(),
                                                                      timestamp =
                                                                      col_time(format =
                                                                      "%H:%M:%S")),
                     trim_ws = TRUE)
final1 <- within(final1, {
  X26 <- NULL
})
final2 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/atletico-barcelona/final.csv",
                     ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                  mediay = col_number(),
                                                                      timestamp =
                                                                      col_time(format =
                                                                      "%H:%M:%S")),
                     trim_ws = TRUE)
final2 <- within(final2, {
  X26 <- NULL
})
final3 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-alaves/final.csv",
                     ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                  mediay = col_number(),
                                                                      timestamp =
                                                                      col_time(format =
                                                                      "%H:%M:%S")),
                     trim_ws = TRUE)
final3 <- within(final3, {
  X26 <- NULL
})
final4 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-atletic/final.csv",
                     ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                  mediay = col_number(),
                                                                      timestamp =
                                                                      col_time(format =
                                                                      "%H:%M:%S")),
                     trim_ws = TRUE)
final4 <- within(final4, {
  X26 <- NULL
})
final5 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-atletico/final.csv",
                     ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                  mediay = col_number(),
                                                                      timestamp =
                                                                      col_time(format =
                                                                      "%H:%M:%S")),
                     trim_ws = TRUE)
final5 <- within(final5, {
  X26 <- NULL
})
final6 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-betis/final.csv",
                     ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                  mediay = col_number(),
                                                                      timestamp =
                                                                      col_time(format =
                                                                      "%H:%M:%S")),
```

```
                  trim_ws = TRUE)
final6 <- within(final6, {
  X26 <- NULL
})
final7 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-cadiz/final.csv",
                  ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                               mediay = col_number(),
                                                                   timestamp =
                                                                   col_time(format =
                                                                   "%H:%M:%S")),
                  trim_ws = TRUE)
final7 <- within(final7, {
  X26 <- NULL
})
final8 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-celta/final.csv",
                  ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                               mediay = col_number(),
                                                                   timestamp =
                                                                   col_time(format =
                                                                   "%H:%M:%S")),
                  trim_ws = TRUE)
final8 <- within(final8, {
  X26 <- NULL
})
final9 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-elche/final.csv",
                  ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                               mediay = col_number(),
                                                                   timestamp =
                                                                   col_time(format =
                                                                   "%H:%M:%S")),
                  trim_ws = TRUE)
final9 <- within(final3, {
  X26 <- NULL
})
final10 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-getafe/final.csv",
                   ";", escape_double = FALSE, col_types = cols(mediax =
                       col_number(),
                                                                mediay = col_number(),
                                                                    timestamp =
                                                                    col_time(format =
                                                                    "%H:%M:%S")),
                   trim_ws = TRUE)
final10 <- within(final10, {
  X26 <- NULL
})
final11 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-granada/final.csv",
                   ";", escape_double = FALSE, col_types = cols(mediax =
                       col_number(),
                                                                mediay = col_number(),
                                                                    timestamp =
                                                                    col_time(format =
                                                                    "%H:%M:%S")),
                   trim_ws = TRUE)
final11 <- within(final11, {
  X26 <- NULL
})
final12 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-huesca/final.csv",
```

```
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                                 mediay = col_number(),
                                                                     timestamp =
                                                                     col_time(format =
                                                                     "%H:%M:%S")),
                    trim_ws = TRUE)
final12 <- within(final12, {
  X26 <- NULL
})
final13 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-osasuna/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                                 mediay = col_number(),
                                                                     timestamp =
                                                                     col_time(format =
                                                                     "%H:%M:%S")),
                    trim_ws = TRUE)
final13 <- within(final13, {
  X26 <- NULL
})
final14<- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-realmadrid/final.csv",
                   ";", escape_double = FALSE, col_types = cols(mediax = col_number(),
                                                                mediay = col_number(),
                                                                    timestamp =
                                                                    col_time(format =
                                                                    "%H:%M:%S")),
                   trim_ws = TRUE)
final14 <- within(final14, {
  X26 <- NULL
})
final15 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-sevilla/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                                 mediay = col_number(),
                                                                     timestamp =
                                                                     col_time(format =
                                                                     "%H:%M:%S")),
                    trim_ws = TRUE)
final15 <- within(final15, {
  X26 <- NULL
})
final16 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/barcelona-valencia/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                                 mediay = col_number(),
                                                                     timestamp =
                                                                     col_time(format =
                                                                     "%H:%M:%S")),
                    trim_ws = TRUE)
final16 <- within(final16, {
  X26 <- NULL
})
final17 <- read_delim("C:/aa yo/4
    ao/TFG/temporada2021/barcelona-valladolid/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
```

```
                                                        mediay = col_number(),
                                                            timestamp =
                                                            col_time(format =
                                                            "%H:%M:%S")),
                    trim_ws = TRUE)
final17 <- within(final17, {
  X26 <- NULL
})
final18 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/betis-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                            timestamp =
                                                            col_time(format =
                                                            "%H:%M:%S")),
                    trim_ws = TRUE)
final18 <- within(final18, {
  X26 <- NULL
})
final19 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/cadiz-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                            timestamp =
                                                            col_time(format =
                                                            "%H:%M:%S")),
                    trim_ws = TRUE)
final19 <- within(final19, {
  X26 <- NULL
})
final20 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/getafe-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                            timestamp =
                                                            col_time(format =
                                                            "%H:%M:%S")),
                    trim_ws = TRUE)
final20 <- within(final20, {
  X26 <- NULL
})
final21 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/granada-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                            timestamp =
                                                            col_time(format =
                                                            "%H:%M:%S")),
                    trim_ws = TRUE)
final21 <- within(final21, {
  X26 <- NULL
})
final22 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/huesca-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                            timestamp =
                                                            col_time(format =
```

```
                                                        "%H:%M:%S")),
                    trim_ws = TRUE)
final22<- within(final22, {
  X26 <- NULL
})
final23 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/levante-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                        timestamp =
                                                        col_time(format =
                                                        "%H:%M:%S")),
                    trim_ws = TRUE)
final23 <- within(final23, {
  X26 <- NULL
})
final24 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/osasuna-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                        timestamp =
                                                        col_time(format =
                                                        "%H:%M:%S")),
                    trim_ws = TRUE)
final24 <- within(final24, {
  X26 <- NULL
})
final25 <- read_delim("C:/aa yo/4
    ao/TFG/temporada2021/realmadrid-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                        timestamp =
                                                        col_time(format =
                                                        "%H:%M:%S")),
                    trim_ws = TRUE)
final25 <- within(final25, {
  X26 <- NULL
})
final26 <- read_delim("C:/aa yo/4
    ao/TFG/temporada2021/realsociedad-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                        timestamp =
                                                        col_time(format =
                                                        "%H:%M:%S")),
                    trim_ws = TRUE)
final26 <- within(final26, {
  X26 <- NULL
})
final27 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/sevilla-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                        mediay = col_number(),
                                                        timestamp =
                                                        col_time(format =
                                                        "%H:%M:%S")),
```

```
                    trim_ws = TRUE)
final27 <- within(final27, {
  X26 <- NULL
})
final28 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/valencia-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                                 mediay = col_number(),
                                                                     timestamp =
                                                                     col_time(format =
                                                                     "%H:%M:%S")),
                    trim_ws = TRUE)
final28 <- within(final28, {
  X26 <- NULL
})
final29 <- read_delim("C:/aa yo/4 ao/TFG/temporada2021/villareal-barcelona/final.csv",
                    ";", escape_double = FALSE, col_types = cols(mediax =
                        col_number(),
                                                                 mediay = col_number(),
                                                                     timestamp =
                                                                     col_time(format =
                                                                     "%H:%M:%S")),
                    trim_ws = TRUE)
final29 <- within(final29, {
  X26 <- NULL
})
union<-rbind(final,final1,final2,final3,final4,final5,final6,final7,final8,final9,final10,
             final11,final12,final13,final14,final15,final16,final17,final18,final19,final20
             ,final21,final22,final23,final24,final25,final26,final27,final28,final29)
#tomo  slo  los datos del FC Barcelona
datosbarcelona <- subset(union, subset=union$`team/id`==217)
#elimino los que tienen menos de 50 respuestas(solo hay uno)
datosbarcelona$player.id<-as.factor(datosbarcelona$player.id)
summary(datosbarcelona$player.id)
datosbarcelona <- subset(datosbarcelona, subset=datosbarcelona$player.id!=13599)
#conversion de variables
datosbarcelona$numero<-as.numeric(datosbarcelona$numero)
datosbarcelona$tiempo<-as.double(datosbarcelona$tiempo)
datosbarcelona$under_pressure<-as.logical(datosbarcelona$under_pressure)
datosbarcelona$acertado.fallado<-as.logical(as.numeric(datosbarcelona$`acertado-fallado`))
datosbarcelona$location0<-as.numeric(datosbarcelona$location0)
datosbarcelona$parte<-as.factor(datosbarcelona$parte)
datosbarcelona$minute<-as.numeric(datosbarcelona$minute)
datosbarcelona$peligrosidad<-as.logical(datosbarcelona$peligrosidad)
datosbarcelona$dt<-as.factor(datosbarcelona$dt)
datosbarcelona$da<-as.factor(datosbarcelona$da)
datosbarcelona$ds<-as.factor(datosbarcelona$ds)
datosbarcelona$tiempo0<-as.factor(datosbarcelona$tiempo==0)
#poner variables nuevas
datosbarcelona$distancia0 <- with(datosbarcelona,
    sqrt((location0-120)^2+(location1-40)^2))
datosbarcelona$distancia1 <- with(datosbarcelona,
    sqrt((endlocation0-120)^2+(endlocation1-40)^2))
datosbarcelona$acercamiento<- with(datosbarcelona, distancia0-distancia1)
datosbarcelona$longitud<-with(datosbarcelona,
    sqrt((location0-endlocation0)^2+(location1-endlocation1)^2))
datosbarcelona$mediajugadordistan <- with(datosbarcelona,
    sqrt((mediax-120)^2+(mediax-40)^2))
```

```
datosbarcelona$campo <- with(datosbarcelona,location0 < 60,
                             class(location0))
datossinporteros$largo=as.factor(1*(datossinporteros$longitud>10))
levels(datossinporteros$largo)=c("corto","largo")
datosbarcelona$peligrosidad<-as.numeric(datosbarcelona$peligrosidad)
datosbarcelona$promdis<-(datosbarcelona$distancia0+datosbarcelona$distancia1)/2
datosbarcelona$promdis2<-datosbarcelona$promdis*datosbarcelona$promdis
for (i in seq(1,22741)){
  if(datosbarcelona$player.id[i]==17304) {
    datosbarcelona$jugador[i]<-"Junior Firpo"
  }
  if(datosbarcelona$player.id[i]==20055 ) {
    datosbarcelona$jugador[i]<-"Ter Stegen"
  }
  if(datosbarcelona$player.id[i]==21881 ) {
    datosbarcelona$jugador[i]<-"Sergino Dest"
  }
  if(datosbarcelona$player.id[i]==22390 ) {
    datosbarcelona$jugador[i]<-"Francisco Trincao"
  }
  if(datosbarcelona$player.id[i]==24841 ) {
    datosbarcelona$jugador[i]<-"Ricki Puig"
  }
  if(datosbarcelona$player.id[i]==30486 ) {
    datosbarcelona$jugador[i]<-"Pedri"
  }
  if(datosbarcelona$player.id[i]==30756 ) {
    datosbarcelona$jugador[i]<-"Ansu Fati"
  }
  if(datosbarcelona$player.id[i]==32480 ) {
    datosbarcelona$jugador[i]<-"Ronald Araujo"
  }
  if(datosbarcelona$player.id[i]==3501 ) {
    datosbarcelona$jugador[i]<-"Philiphe Coutinho"
  }
  if(datosbarcelona$player.id[i]==39073 ) {
    datosbarcelona$jugador[i]<-"Ilaix Moriba"
  }
  if(datosbarcelona$player.id[i]==43728 ) {
    datosbarcelona$jugador[i]<-"scar Mingueza"
  }
  if(datosbarcelona$player.id[i]==4447 ) {
    datosbarcelona$jugador[i]<-"Martin Braithwaite"
  }
  if(datosbarcelona$player.id[i]==5203 ) {
    datosbarcelona$jugador[i]<-"Sergio Busquets"
  }
  if(datosbarcelona$player.id[i]==5211 ) {
    datosbarcelona$jugador[i]<-"Jordi Alba"
  }
  if(datosbarcelona$player.id[i]==5213 ) {
    datosbarcelona$jugador[i]<-"Gerard Piqu"
  }
  if(datosbarcelona$player.id[i]==5477 ) {
    datosbarcelona$jugador[i]<-"Ousmane Dembel"
  }
  if(datosbarcelona$player.id[i]==5487 ) {
    datosbarcelona$jugador[i]<-"Antoine Griezmann"
```

```
  }
  if(datosbarcelona$player.id[i]==5492 ) {
    datosbarcelona$jugador[i]<-"Samuel Umtiti"
  }
  if(datosbarcelona$player.id[i]==5503 ) {
    datosbarcelona$jugador[i]<-"Leo Messi"
  }
  if(datosbarcelona$player.id[i]==6379 ) {
    datosbarcelona$jugador[i]<-"Sergi Roberto"
  }
  if(datosbarcelona$player.id[i]==6590 ) {
    datosbarcelona$jugador[i]<-"Norberto Neto"
  }
  if(datosbarcelona$player.id[i]==6826 ) {
    datosbarcelona$jugador[i]<-"Clment Lenglet"
  }
  if(datosbarcelona$player.id[i]==6947 ) {
    datosbarcelona$jugador[i]<-"Miralem Pjani"
  }
  if(datosbarcelona$player.id[i]==8118 ) {
    datosbarcelona$jugador[i]<-"Frenkie de Jong"
  }
}
datosbarcelona$jugador<-as.factor(datosbarcelona$jugador)
#elimino los porteros de la base de datos
datossinporteros<-subset(datosbarcelona, subset=datosbarcelona$jugador!="Ter Stegen")
datossinporteros<-subset(datossinporteros, subset=datossinporteros$jugador!="Norberto
    Neto")

#primer modelo
modelo22<-glm(acertado.fallado ~
                promdis+acercamiento+longitud+campo
              +mediajugadordistan+da+parte+
                +under_pressure,family = binomial, data =
                datossinporteros)
binnedplot(fitted.values(modelo22),
           residuals(modelo22, type = "response"),
           nclass = 100,
           xlab = "Expected Values",
           ylab = "Average residual",
           main = "Binned residual plot",
           cex.pts = 0.8,
           col.pts = 1,
           col.int = "gray")
par(mfrow=c(2,2))
binnedplot(datossinporteros$acercamiento,residuals(modelo22, type = "response"),
           nclass = 100,
           xlab = "Acercamiento",
           ylab = "Average residual",
           main = "Binned residual plot Acercamiento",
           cex.pts = 0.8,
           col.pts = 1,
           col.int = "gray")
binnedplot(datossinporteros$promdis,residuals(modelo22, type = "response"),
           nclass = 100,
           xlab = "distancia promedio del pase a la portera rival",
           ylab = "Average residual",
```

```
          main = "Binned residual plot distancia promedio del pase a la portera
              rival",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
binnedplot(datossinporteros$longitud,residuals(modelo22, type = "response"),
          nclass = 100,
          xlab = "longitud del pase",
          ylab = "Average residual",
          main = "Binned residual plot longitud",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
binnedplot(datossinporteros$mediajugadordistan,residuals(modelo22, type = "response"),
          nclass = 100,
          xlab = "distancia media del jugador a la portera rival",
          ylab = "Average residual",
          main = "Binned residual plot distancia media del jugador a la portera
              rival",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
#segundo modelo
modelo33<-glm(acertado.fallado ~
               promdis+promdis2+acercamiento+longitud*largo+campo
             +mediajugadordistan+da+parte+
               +under_pressure,family = binomial, data =
               datossinporteros)
summary(modelo33)
binnedplot(fitted(modelo33),
          residuals(modelo33, type = "response"),
          nclass = 100,
          xlab = "Expected Values",
          ylab = "Average residual",
          main = "Binned residual plot",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
par(mfrow=c(2,2))
binnedplot(datossinporteros$acercamiento,residuals(modelo33, type = "response"),
          nclass = 100,
          xlab = "Acercamiento",
          ylab = "Average residual",
          main = "Binned residual plot Acercamiento",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
binnedplot(datossinporteros$promdis,residuals(modelo33, type = "response"),
          nclass = 100,
          xlab = "distancia promedio del pase a la portera rival",
          ylab = "Average residual",
          main = "Binned residual plot distancia promedio",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
binnedplot(datossinporteros$longitud,residuals(modelo33, type = "response"),
          nclass = 100,
          xlab = "distancia promedio del pase a la portera rival",
```

```
          ylab = "Average residual",
          main = "Binned residual plot distancia promedio",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
binnedplot(datossinporteros$mediajugadordistan,residuals(modelo33, type = "response"),
          nclass = 100,
          xlab = "distancia promedio del pase a la portera rival",
          ylab = "Average residual",
          main = "Binned residual plot distancia promedio",
          cex.pts = 0.8,
          col.pts = 1,
          col.int = "gray")
pred <- prediction(modelo33$fitted.values, datossinporteros$acertado.fallado)
perf <- performance(pred,measure="tpr",x.measure="fpr")
AUC<- performance(pred,measure="auc")
AUCaltura <- AUC@y.values[[1]]
plot(perf,
     main = "Curva ROC",
     xlab="Tasa de falsos positivos",
     ylab="Tasa de verdaderos positivos")
abline(a=0,b=1,col="blue",lty=2)
legend("bottomright",legend=paste(" AUC =",round(AUCaltura,4)))

probmedia <- aggregate(modelo33$fitted.values~datossinporteros$jugador,
    datossinporteros, mean, na.rm=TRUE)
names(probmedia) <- make.names(names(probmedia))
names(probmedia)[c(1)] <- c("jugador")
datossinporteros$acertado.fallado<-as.numeric(datossinporteros$acertado.fallado)
probmedia$datos<-aggregate(datossinporteros$acertado.fallado==1~datossinporteros$jugador,
    datossinporteros, sum, na.rm=TRUE)[,2]
totalpasesporjugador<-aggregate(datossinporteros$acertado.fallado==0~datossinporteros$jugador,
    datossinporteros, sum, na.rm=TRUE)
totalpasesporjugador[,2]<-totalpasesporjugador[,2]+aggregate(datossinporteros$acertado.fallado==1~dat
    datossinporteros, sum, na.rm=TRUE)[,2]
probmedia$datos<-probmedia$datos/totalpasesporjugador[,2]
#dibujo los datos reales vs las medias estimadas
plot(probmedia$modelo33.fitted.values,probmedia$datos,xlab='Prediccin probabilidad
    acierto pase'
     ,ylab='Probabilidad acierto pase',main='Comparacin probabilidad real vs
         probabilidad predicha por jugador')
text(probmedia$modelo33.fitted.values,probmedia$datos,
     labels = (as.character(probmedia[,1])),
     cex = 0.6, pos=3,col = "red")
abline(a = 0, b = 1)
legend("topleft", legend = c('Probabilidad real=Probabilidad predicha'),
       lwd = 2, col = c("black"))

probmedia$diferencia<-probmedia$datos-probmedia$modelo33.fitted.values
plot(probmedia$modelo33.fitted.values,probmedia$diferencia,xlab='Prediccin
    probabilidad acierto pase'
     ,ylab='Probabilidad real-probabilidad predicha',main='Diferencia entre jugadores')
text(probmedia$modelo33.fitted.values,probmedia$diferencia,
     labels = (as.character(probmedia[,1])),
     cex = 0.6, pos=3,col = "red")
abline(a = 0, b = 0)
legend("topleft", legend = c('Probabilidad real=Probabilidad predicha'),
       lwd = 2, col = c("black"))
```

```
#modelo con jugadores
modelo32<-glm(acertado.fallado ~
                promdis+promdis2+acercamiento+longitud*largo+campo
              +da+parte+
                +under_pressure+jugador,family = binomial, data =
                datossinporteros)
summary(modelo32)
library(arm)
binnedplot(fitted(modelo32),
           residuals(modelo32, type = "response"),
           nclass = 100,
           xlab = "Expected Values",
           ylab = "Average residual",
           main = "Binned residual plot",
           cex.pts = 0.8,
           col.pts = 1,
           col.int = "gray")

library(ROCR)
pred <- prediction(modelo32$fitted.values, datossinporteros$acertado.fallado)
perf <- performance(pred,measure="tpr",x.measure="fpr")
AUC<- performance(pred,measure="auc")
AUCaltura <- AUC@y.values[[1]]
plot(perf,
     main = "Curva ROC",
     xlab="Tasa de falsos positivos",
     ylab="Tasa de verdaderos positivos")
abline(a=0,b=1,col="blue",lty=2)
legend("bottomright",legend=paste(" AUC =",round(AUCaltura,4)))

#modelo peligrosidad
modelo1<-glm(peligrosidad~
               mediajugadordistan,family = binomial, data =
               datossinporteros)
summary(modelo1)

library(arm)
binnedplot(datossinporteros$mediajugadordistan,
           residuals(modelo1, type = "response"),
           nclass = 100,
           xlab = "distancia media del jugador a la portera rival",
           ylab = "Average residual",
           main = "Binned residual plot",
           cex.pts = 0.8,
           col.pts = 1,
           col.int = "gray")
probmedia1 <- aggregate(modelo1$fitted.values~datossinporteros$jugador,
    datossinporteros, mean, na.rm=TRUE)
names(probmedia1) <- make.names(names(probmedia1))
names(probmedia1)[c(1)] <- c("jugador")
datossinporteros$peligrosidad<-as.numeric(datossinporteros$peligrosidad)
probmedia1$datos<-aggregate(datossinporteros$peligrosidad==1~datossinporteros$jugador,
    datossinporteros, sum, na.rm=TRUE)[,2]
totalpasesporjugador1<-aggregate(datossinporteros$peligrosidad==0~datossinporteros$jugador,
    datossinporteros, sum, na.rm=TRUE)
totalpasesporjugador1[,2]<-totalpasesporjugador1[,2]+aggregate(datossinporteros$peligrosidad==1~datos
    datossinporteros, sum, na.rm=TRUE)[,2]
probmedia1$datos<-probmedia1$datos/totalpasesporjugador1[,2]
```

```
plot(probmedia1$modelo1.fitted.values,probmedia1$datos,xlab='Prediccin probabilidad
    acierto pase'
     ,ylab='Probabilidad acierto pase',main='Comparacin probabilidad real vs
         probabilidad predicha por jugador')
text(probmedia1$modelo1.fitted.values,probmedia1$datos,
     labels = (as.character(probmedia[,1])),
     cex = 0.9, pos=3,col = "red")
abline(a = 0, b = 1)
legend("topleft", legend = c('Probabilidad real=Probabilidad predicha'),
       lwd = 2, col = c("black"))
probmedia1$diferencia<-probmedia1$datos-probmedia1$modelo1.fitted.values
plot(probmedia1$modelo1.fitted.values,probmedia1$diferencia,xlab='Prediccin
    probabilidad de pase peligroso'
     ,ylab='Probabilidad real-probabilidad predicha',main='Diferencia entre jugadores')
text(probmedia1$modelo1.fitted.values,probmedia1$diferencia,
     labels = (as.character(probmedia[,1])),
     cex = 1.1, pos=3,col = "red")
abline(a = 0, b = 0)
legend("topleft", legend = c('Probabilidad real=Probabilidad predicha'),
       lwd = 2, col = c("black"))


#represento el grafico final.
plot(probmedia$diferencia,probmedia1$diferencia,xlab='Probabilidad real de acierto
    del pase - probabilidad estimada de acierto de pase'
     ,ylab='Probabilidad real de pase peligroso - probabilidad estimada de pase
         peligroso')
text(probmedia$diferencia,probmedia1$diferencia,
     labels = (as.character(probmedia[,1])),
     cex = 0.9, pos=3,col = "red")
abline(a = 0, b = 0)
abline(v=0)
```

Bibliografía

[1] https://es.wikipedia.org/wiki/F

[2] https://www.acadef.es/fundamentos-del-pase-en-el-futbol/

[3] https://es.wikipedia.org/wiki/Johan$_{C}ruyff$

[4] https://www.academia.edu/20341551/Beyond$_{c}ompletion_{r}ate_{e}valuating_{t}he_{p}assing_{a}bility_{o}f_{f}ootballers$

[5] https://eprints.whiterose.ac.uk/134600/1/1-s2.0-S0377221718300365-main.pdf

[6] https://github.com/statsbomb/open-data/

[7] https://www.aconvert.com/es/document/json-to-xls/

[8] https://bookdown.org/roback/bookdown-BeyondMLR/

[9] https://rpubs.com/Joaquin$_{A}R/229736$

[10] http://halweb.uc3m.es/esp/Personal/personas/jmmarin/esp/Categor/Tema3Cate.pdf

[11] https://es.wikipedia.org/wiki/Criterio$_{d}e_{i}nformaci$

[12] https://es.wikipedia.org/wiki/Curva$_{R}OC$