TRABAJO FIN DE GRADO
GRADO EN CIENCIA E INGENIERÍA DE DATOS

Análisis estadístico del data warehouse del R.C. Deportivo de La Coruña y uso de minería de datos aplicada al scouting

Estudiante: Alejandro Rodríguez Pérez

Dirección: Ricardo Cao Abad

Ignacio Lourido Fuertes

A Coruña, June de 2023.

Dedicatoria

A mi familia y a las amistades forjadas durante estos 4 años.

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todas las personas que contribuyeron a la realización de este Trabajo Fin de Grado de manera significativa. En primer lugar, quiero agradecer a Ricardo y a Nacho su inestimable guía y apoyo a lo largo de este proyecto. Su experiencia, orientación y apoyo fueron fundamentales para el éxito de este trabajo. Además, me gustaría mostrar mi gratitud a las amistades hechas a lo largo de estos cuatro años, por su constante apoyo, comprensión y buenos momentos durante este tiempo.

Resumen

En este Trabajo Fin de Grado, se investiga el uso del Big Data en el contexto del fútbol y su aplicación en la toma de decisiones relacionadas con la exploración y fichaje de jugadores. Se analiza el almacén de datos del R.C. Deportivo de La Coruña, que contiene una gran cantidad de información estadística recopilada sobre multitud de jugadores a nivel global y variables relacionadas con aspectos del juego durante los 90 minutos que dura un partido de fútbol.

El estudio se enfoca en la aplicación de técnicas avanzadas de análisis estadístico y minería de datos para explorar los jugadores y evaluar su potencial para ser fichados por el club. Se emplean técnicas como PCA (análisis de componentes principales) y clustering para identificar patrones y agrupaciones de jugadores con características similares.

Además, se utiliza el machine learning para llevar a cabo metodos de regresión y predecir el rendimiento futuro de los jugadores en función de distintas variables de interés, según la posición del jugador en el campo, como expectativa de tiros a puerta, media de pases exitosos por partido o número de interceptaciones exitosas.

Los resultados obtenidos muestran la eficacia de estas técnicas para identificar jugadores con un alto potencial y contribuir a la toma de decisiones informadas en el proceso de fichaje. El uso del big data y la minería de datos en el fútbol demuestra su utilidad para optimizar la selección de jugadores y mejorar el rendimiento global del equipo.

Este estudio destaca la importancia creciente del big data en el ámbito deportivo y ofrece una base sólida para futuras investigaciones en el uso de técnicas analíticas avanzadas para la toma de decisiones estratégicas en el fútbol.

Abstract

This Final Degree Project investigates the use of big data in the context of football and its application in decision making related to the scouting and signing of players. It analyses the data warehouse of R.C. Deportivo de La Coruña, which contains a large amount of statistical information on a multitude of players at a global level and variables related to aspects of the game during the 90 minutes of a football match.

The study focuses on the application of advanced statistical analysis techniques and data mining to explore the players and evaluate their potential to be signed by the club. Techniques such as PCA (principal cmponent analysis) and clustering are used to identify patterns and groupings of players with similar characteristics.

In addition, machine learning is used to carry out the regression analysis and predict the future performance of players based on different variables of interest according to the player's

position on the field, such as expected shots on goal, average number of successful passes per game or number of successful interceptions.

The results obtained show the effectiveness of these techniques to identify players with high potential and contribute to making informed decisions in the signing process. The use of big data and data mining in football demonstrates its usefulness in optimising player selection and improving overall team performance.

This study highlights the growing importance of big data in sports and provides a solid foundation for future research in the use of advanced analytical techniques for strategic decision making in football.

Palabras clave:

- Preprocesado de datos
- Análisis de Componentes Principales
- Clustering Híbrido
- Segmentación de jugadores
- Similitud Coseno
- Machine Learning
- Regresión

Keywords:

- Data preprocessing
- Principal Component Analysis
- Hybrid Clustering
- Player Profile Segmentation
- Cosine Similarity
- Machine Learning
- Regression

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

En la era del Big Data, los datos se han convertido en una de las mayores fuentes de valor en la economía moderna. Las empresas de cualquier ámbito han comenzado a reconocer el enorme potencial de los datos para mejorar su toma de decisiones y generar nuevos modelos de negocio. Para ello, están invirtiendo en tecnologías de análisis de datos, para extraer información valiosa de grandes conjuntos de datos y ser capaces de generar valor a través de ellos.Este nuevo paradigma humano se centra en la recopilación, análisis y uso de datos para mejorar el rendimiento empresarial y crear nuevas oportunidades de mercado.

Una de las áreas donde el crecimiento es mayor es el mundo del deporte. El deporte es un negocio global, donde cada región se ha especializado en algunos deportes específicos según su historia y tradiciones. Así pues, en Europa los deportes predominantes son fútbol, baloncesto, tenis mientras que en Asia dominan deportes como el badminton, criquet o las artes marciales.

El fútbol, es quizás el deporte que más impacto tiene en la sociedad a nivel global ya que posee eventos a nivel mundial como el Mundial de Clubs o el Mundial de Selecciones. La exigencia de este deporte cada vez es mayor, por lo que la búsqueda de nuevas métodos que permitan aumentar el rendimiento de un club a nivel empresarial es constante y a día de hoy ya son muchos los clubs del más alto nivel que se están incorporando a este nuevo paradigma que tiene al dato como punto central.

Estos clubs poseen bases de datos de gran dimensión que contienen multitud de datos de todo tipo relacionados con sus jugadores, como por ejemplo: el numero de sprints que realiza un determinado jugador en un partido, medidas antropométricas o incluso una valoración subjetiva sobre la calidad del descanso durante la noche .

En este trabajo, se analizarán los datos reales almacenados en una base de datos de un club

de futbol semi-profesional que contiene información de multitud de jugadores a nivel global sobre variables relacionadas con el juego generado durante un partido de futbol.

1.1 Motivación y objetivo

La motivación de este trabajo surge de la pasión que tengo tanto por el mundo del deporte como por el mundo de la tecnología y del análisis de datos.

A día de hoy, las oportunidades que se estan generando dentro de este deporte, en cuánto al tratamiento de datos masivos se refiere, es enorme [1], ya que es una tecnología que hasta hace más bien poco era nula en este deporte. Actualmente se está comenzando a integrar poco a poco en España, por lo que la demanda de especialistas que dominen esta tecnología es enorme.

A nivel personal, cuando vi que tenía la ocasión ideal de compaginar mis dos pasiones a través de la oportunidad brindada por el Departamento de Análisis de Datos del R.C.Deportivo de La Coruña [2], decidí aceptar esta línea de investigación que me propusieron, en la que se combina los datos que poseen en su almacén de datos a nivel global juntos con mis conocimientos del grado de Ciencia e Ingenieria de Datos para tratar de generar un valor añadido al club.

El objetivo del trabajo consiste en explotar dicha información y ser capaces de trasmitir información de valor desde un nivel de abstracción mayor de modo que sea posible comprender dicha información de forma directa, sencilla y visual, ya que al fin y al cabo "la información es poder´´.

1.2 ¿Qué es el fútbol?

- **Origen**
  El fútbol [3]tiene sus raíces en la antigua historia de civilizaciones como los chinos, los egipcios y los romanos. Sin embargo, fue en el Reino Unido, en el siglo XIX, donde se establecieron las bases modernas del juego. Las reglas comenzaron a tomar forma en las escuelas y universidades británicas, y en 1863 se fundó la Football Association en Inglaterra, estableciendo un conjunto de reglas unificadas. A partir de ese momento, el fútbol se convirtió en un deporte popular y su popularidad se extendió rápidamente por todo el mundo.

- **Evolución y expansión**
  A lo largo de los años, el fútbol ha experimentado una evolución impresionante que ha

transformado tanto el juego en sí como su influencia en la sociedad. Desde sus humildes comienzos, en los que se jugaba en campos abiertos sin reglas definidas, el fútbol ha crecido exponencialmente y se ha convertido en el deporte más popular del mundo. El número de jugadores y equipos participantes ha aumentado significativamente, abarcando desde ligas locales hasta competiciones internacionales de alto nivel.

La creación de competiciones internacionales emblemáticas, como la Copa Mundial de la FIFA y la Liga de Campeones de la UEFA, ha llevado al fútbol a un nivel global. Estos torneos reúnen a los mejores jugadores y equipos de todo el mundo, generando un fervor y una pasión sin precedentes. La Copa Mundial, en particular, se ha convertido en el evento deportivo más seguido y celebrado a nivel mundial, capturando la atención de miles de millones de personas durante su celebración cada cuatro años.

Además, el fútbol ha abrazado plenamente la era digital. Las transmisiones en vivo de los partidos permiten a los fanáticos de todo el mundo seguir y apoyar a sus equipos favoritos sin importar su ubicación geográfica. La tecnología también ha desempeñado un papel crucial en la evolución del fútbol, ya que, actualmente, muchos de los estadios modernos están equipados con sistemas de iluminación de última generación, césped sintético de alta calidad y tecnología de videoarbitraje (VAR) para tomar decisiones más justas. Estos avances tecnológicos han mejorado la experiencia de juego tanto para los jugadores como para los espectadores.

Los avances en el análisis estadístico también han proporcionado una comprensión más profunda del juego, permitiendo a los entrenadores y analistas estudiar y mejorar el rendimiento de los jugadores y equipos. Por último, las redes sociales han conectado a los fanáticos de una manera sin precedentes, creando comunidades en línea donde los seguidores pueden compartir su amor por el fútbol, discutir jugadas destacadas y participar en debates apasionados.

En resumen, la evolución del fútbol ha sido extraordinaria. Desde sus inicios modestos, ha crecido en términos de participación, interés público y relevancia en la sociedad. La combinación de tácticas de juego cada vez más sofisticadas, avances tecnológicos, competiciones internacionales emocionantes y la integración en el mundo digital ha llevado al fútbol a un nivel global, convirtiéndolo en un fenómeno cultural y social de enorme magnitud.

- **Reglas básicas**
  En el siguiente listado se muestran las reglas básicas que definen este deporte, para

así poder entender un poco mejor el contexto en el que se desarrolla este proyecto y comprender mejor los datos con los que se va a trabajar.

- **Número de jugadores**: Cada equipo está compuesto por once jugadores titulares (comienzan el partido), incluyendo a un portero y hasta un máximo de 15 jugadores suplentes, puediendo realizar un máximo de 6 cambios por partido.

- **Duración del partido**: Un partido de fútbol se divide en dos tiempos de 45 minutos cada uno, con un descanso de 15 minutos entre ellos.

- **Objetivo del juego**: El objetivo principal del juego es marcar más goles que el equipo contrario. Para ello, el balón debe cruzar completamente la línea de gol entre los postes y por debajo del travesaño.

- **Fuera de juego**: Un jugador se encuentra en posición de fuera de juego si se encuentra más cerca de la línea de gol contraria en comparación con el balón y el penúltimo defensor en el momento en que se realiza el pase.

- **Faltas y tarjetas**: El árbitro sanciona las faltas cometidas por los jugadores, ya sea por juego brusco, infracciones tácticas o conductas antideportivas. Las faltas pueden resultar en tiros libres directos o indirectos, y en algunos casos, el árbitro puede mostrar tarjetas amarillas o rojas a los jugadores según la gravedad de la infracción.

- **Saques de banda, saques de esquina y saques de meta**: Cuando el balón sale por los laterales, se realiza un saque de banda. Si el balón sale por la línea de gol, el equipo contrario realiza un saque de esquina si fue tocado por el equipo defensor, o un saque de meta si fue tocado por el equipo atacante.

- **Árbitro y árbitros asistentes**: Un partido de fútbol es dirigido por un árbitro principal, quien toma las decisiones finales. También cuenta con árbitros asistentes, quienes ayudan a tomar decisiones en situaciones difíciles, como fuera de juego o faltas no vistas por el árbitro principal.

- **Tiros penales**: Se concede un tiro penal cuando se comete una falta dentro del área de penalización. El equipo que recibe el tiro penal tiene la oportunidad de

marcar un gol sin oposición desde el punto de penalización, mientras que el portero intenta detener el disparo.

1.3 Recolección de datos en el fútbol

El proceso de recolección de datos en el fútbol ha experimentado un crecimiento significativo en los últimos años[4]. Esta idea surge de la necesidad de obtener información cuantitativa y cualitativa sobre el rendimiento de los jugadores y los equipos, con el fin de obtener valor a partir de la información y facilitar la toma de decisiones.

Existen diversos dispositivos y tecnologías utilizadas en la recolección de datos en el fútbol. Uno de los más comunes es el uso de sensores integrados en prendas de vestir o en dispositivos portátiles que los jugadores llevan durante los entrenamientos y partidos. Estos sensores pueden registrar datos como la distancia recorrida, la velocidad, los cambios de dirección, la frecuencia cardíaca y otros parámetros fisiológicos. Además, se utilizan cámaras de alta velocidad y sistemas de seguimiento de posición para capturar información sobre los movimientos y la interacción de los jugadores en el campo.Por último, también se pueden recoger datos de manera manual a través de empresas especializadas en este sector, y que envían a personal de la empresa a los estadios con la función de registrar manualmente los sucesos que rodean a cada jugador a lo largo de los 90 minutos.

La información recolectada se procesa y analiza utilizando software especializado que permite extraer conocimientos útiles. Los datos recopilados pueden ser utilizados para evaluar el rendimiento individual de los jugadores, identificar patrones de juego, analizar la eficacia de las tácticas y estrategias utilizadas, y realizar comparaciones con otros equipos o jugadores de élite. Estos análisis pueden ayudar a los entrenadores y directores técnicos a tomar decisiones más informadas en términos de alineación, tácticas de juego y entrenamiento.

El potencial de los datos en la mejora de un club de élite es extraordinario. La información recopilada puede ayudar a identificar fortalezas y debilidades de los jugadores y el equipo en general. Esto permite desarrollar planes de entrenamiento más específicos y personalizados para abordar áreas de mejora y potenciar las habilidades individuales. Los datos también pueden ser utilizados a nivel individual para monitorizar la carga física y evitar lesiones, optimizar la recuperación y establecer objetivos de rendimiento realistas.

Además, la recolección de datos en el fútbol ofrece la oportunidad de tomar decisiones

estratégicas en áreas como la identificación y el reclutamiento de talentos. Los datos pueden ayudar a evaluar el potencial de los jugadores jóvenes y comparar su rendimiento con estándares establecidos. Esto permite a los clubes de élite identificar e incorporar a los talentos emergentes, aumentando su capacidad de éxito en el futuro.

En conclusión, el proceso de recolección de datos en el fútbol seguido de un análisis y procesamiento realizado por especialistas ha surgido como una herramienta valiosa que poseen los clubes de élite para la mejora de diversas áreas, como por ejemplo, en el Sevilla F.C. [5]

Capítulo 2

Contenido demostrativo

En los siguientes capitulos se irá mostrando, paso a paso, el desarrollo que se ha seguido para realizar este Trabajo de Fin de Grado, así como las explicaciones necesarias para comprender la motivación detras de cada decisión tomada y la interpretación de los resultados generados durante del trabajo.

2.1 Gestión del Proyecto

En esta sección se trata toda la información relacionada con la gestión del proyecto: la metodología de desarrollo escogida y la gestión del tiempo, de los costes y de los riesgos.

Esto nos permite determinar de una forma clara qué objetivos se deben cumplir y cuál es la ruta que se ha de seguir para que el proyecto sea exitoso, siendo capaz de prevenir posibles problemas, gracias a una buena planificación y estructuración del trabajo.

2.1.1 Metodología

Durante este proyecto, la metodología de trabajo que se ha utilizado, ha sido una metodología de tipo ágil, concretamente la metodología *scrum* [6].

La metodología *scrum* es una metodología ágil de gestión de proyectos que se centra en la entrega iterativa e incremental de productos, en este caso, la herramienta para la exploración de jugadores en base a análisis estadístico y *machine learning*.

En esta metodología, el proceso comienza con la creación del *backlog* del producto, que es una lista priorizada de todos los elementos que deben ser desarrollados para el producto. Estos elementos se conocen como historias de usuario y son descripciones breves de las características o funcionalidades requeridas. Las objetivos planteados que debe cumplir la herramienta son los siguientes:

1. Realizar un análisis descriptivo de los datos para así tener una primera aproximación de cuál es la naturaleza de los datos con los que se va a trabajar.

2. Mediante técnicas estadísticas, filtrar los datos para eliminar redundancia y trabajar con aquellos datos que sean significativos dentro de la base de datos.

3. Utilizando el análisis de componentes principales, realizar una reducción de la dimensión del almacén de datos a 3 dimensiones, para poder realizar visualizaciones 2D y 3D y conseguir una interpretación de cada una de las componentes en relación a los posibles perfiles de jugador.

4. A través de la técnica de clustering, realizar una ordenación de jugadores con carcaterísticas similares en clusters para analizar los diferentes perfiles existentes y comprender el significado de cada cluster.

5. Partiendo de una descripción cualitiva sobre el perfil de un jugador deseado por la dirección deportiva del R.C. Deportivo de La Coruña, utilizar los clusters creados previamente junto a diferentes medidas de distancia, para analizar cuáles serían las 5 opciones dentro del mercado que mejor satisfagan la descripción dada.

6. Una vez obtenidos los diferentes jugadores que mejor se adaptan al perfil buscado, realizar predicciones mediante modelos de regresión, sobre diferentes variables de interés para la dirección deportiva del club.

Para cumplir dichos objetivos, la metodología define los *sprints*, que son períodos cortos de tiempo, generalmente de 1 a 4 semanas. Al comienzo de cada *sprint*, se selecciona uno de los objetivos del backlog del producto, y en dicho *sprint* se focaliza el trabajo en el objetivo seleccionado.

Durante el *sprint* se llevan han llevado a cabo reuiones en intervalos de dos semanas con ambos directos del trabajo, para mostrar los avances obtenidos durante ese sprint y recibir *feedback* para lograr el objetivo planteado. Al finalizar el *sprint*, se realiza una reunión final con ambos directores, con la idea de validar si se ha logrado el objetivo planteado, o si es necesario dedicarle un nuevo sprint para obtener un mejor resultado.

A medida que los sprints se van sucediendo, el producto evoluciona y se va desarrollando de forma incremental. De este modo, se promueve una mentalidad de mejora continua, donde cada sprint aporta un aprendizaje y el proyecto se adapta en función del *feedback* recibido.

2.1.2 Estructura de descomposición del trabajo:

En esta sección, se muestra la ruta establecida para tratar de lograr un proyecto exitoso en el plazo de tiempo establecido. Una vez realizado el proyecto, es posible decir que dicho plan se ha seguido de forma estricta a lo largo del desarrollo del proyecto.

La estructura de descomposición del trabajo [7] es un gráfico visual sencillo de interpretar que agrupa todas las tareas necesarias del proyecto en un *timeline*, estableciendo fecha de inicio y fecha de fin para cada una de ellas. Se ha utilizado la herramienta *Toggl Plan* para la elaboración de este gráfico, ya que es una herramienta web gratuita con la que es posible generar estructuras de descomposición del trabajo de forma sencilla y rápida.

En la figura 2.1 se puede apreciar de forma global la planificación planteada en el proyecto (cada color esta asociado a un *sprint* determinado con su respectivo objetivo de la sección 2.1.1). El enlace para acceder a la planificación y así, poder profundizar en cada una de las tareas, es el siguiente:
`https://plan.toggl.com/#pg/3MRpBy5YyV2oPnaPIlJZe8hsKuFLnbfV`
(*NOTA:* por defecto, el *timeline* se sitúa en la fecha actual, por lo que es necesario desplazar la barra deslizante hasta el día 1 de diciembre de 2022, fecha en la que comenzó este proyecto.)

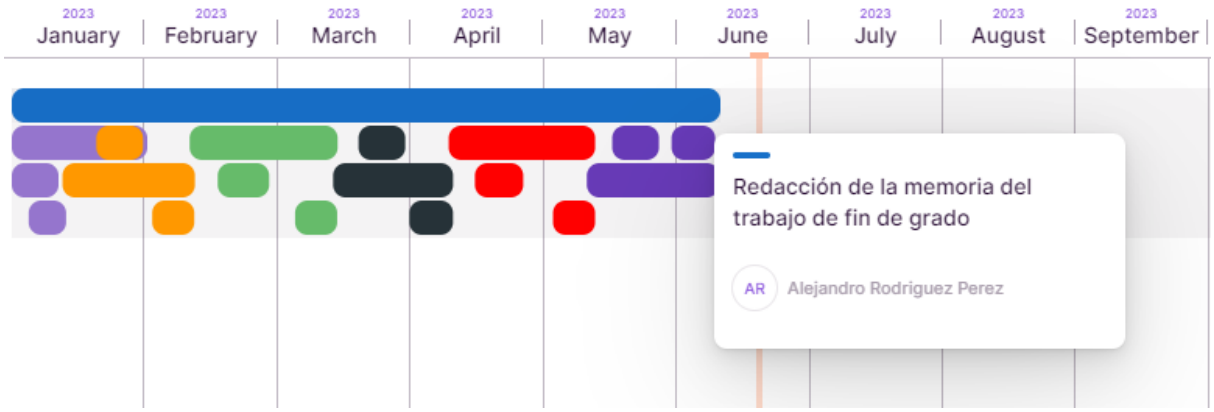


Figura 2.1: Estructura de descomposición del trabajo del proyecto

2.1.3 Gestión de costes

1. **Costes de recursos humanos:**

   En esta sección, se estiman los costes asociados con el equipo de desarrollo, que en este caso está formado por tan solo una persona, pero a la que hay que añadir a los dos tutores de este proyecto. En primer lugar, para calcular el coste total de la persona de

desarrollo y de los dos tutores asociados, se considera su rol como científico e ingeniero de datos [8],personal de investigación [9] y director de departamento[10], respectivamente. Tras un estudio de los sueldos, en el sector de las TIC en España, en la tabla 2.1 se ha establecido una estimación del coste asociado a cada rol.

Puesto laboral	Salario anual
Ingeniero de datos	31.250 €
Catedrático Universitario	47.000 €
Director de departamento de análisis de datos	82.000 €

Tabla 2.1: Salario medio español para los roles involucrados en este proyecto

Por tanto,en España, el trabajo de Ingeniero de datos está remunerado con 16.03€ \hora. El trabajo de fin de grado supone un total de 12 créditos *ECS*, donde para cada crédito le corresponde un total de 25 horas de trabajo, por lo que la estimación total de horas dedicadas a este proyecto es de 300 horas. Haciendo las cuentas, el coste total del desarrollador es de 4.800€.

Respecto a ambos tutores, sus roles tienen un coste de 25€ \hora y 45€ \hora respectivamente. Durante el proyecto, se ha concertado un total de 6 reuniones con una duración media de 1 hora cada reunión, por lo que el coste total asociado a los tutores asciende a 420€

Sumando ambas cantidades, el coste humano se estima en 5220€

2. **Coste del material:**

   Durante este proyecto, el único material utilizado ha sido un ordenador portátil de la marca *TOSHIBA* y modelo *Satellite PRO* con un procesador intel core i7 y 8 GB de memoria RAM, valorado actualmente en 200€.

3. **Coste total del proyecto:**

   En la tabla 2.2 se puede observar el coste total asociado a este trabajo de fin de grado.

2.2 Análisis inicial de la base de datos

En esta sección se realiza una exploración introductoria de los datos recopilados. Este análisis proporciona una visión general de la estructura y características de la base de datos utilizada en el Trabajo de Fin de Grado.

2.2.1 Dimensión de la base de datos

A partir de la plataforma online de recolección de datos de fútbol *"Wyscout"*, el R.C Deportivo de La Coruña me ha proporcionado un almacén de datos que consta de 55648tres registros, donde cada registro representa un jugador de fútbol federado en la FIFA (Federation International Football Association (FIFA)) perteneciente a algún equipo a nivel mundial, tanto de las grandes ligas europeas como de ligas menores en países menos desarrollados. Para cada jugador, estan registradas 190 variables relacionadas con aspectos antropométricos (peso, edad, altura, etc), con aspectos económicos y legales (valor de mercado, fecha de expiración de contrato) y la gran mayoría con aspectos propios del fútbol y lo que sucede durante los 90 minutos de juego (numero de tiros a puerta, número de tarjetas amarillas, número de pases completados, número de entradas realizadas, etc).

2.2.2 Definición de las variables y formato de los datos

Profundizando un poco más en los aspectos que cubren las variables mencionadas en la seccion anterior, la plataforma *"Wyscout"*, posee un glosario de metadatos asociado al almacén de datos [11], que incluye para cada variable su definición y contenido audiovisual para poder visualizar dicha variable en una acción real de juego.

Dentro de las 190 variables del almacén, existe diferentes áreas que intervienen en el entorno de un jugador de fútbol, en la tabla 2.3 se muestra algunos ejemplos de variables para dichas áreas:

Tabla 2.3: Diferentes áreas del almacén de datos

Área deportiva	**Área antropométrica**	**Área legal y económica**
stcurr_offensive_duels_avg	*stcurr_height*	*stcurr_valor*
stcurr_passes_avg	*peso*	*stcurr_contract_expires*
stcurr_head_goals	*stcurr_age*	*stcurr_market_value*

Respecto a cuál es el formato original de los datos que contienen cada una de las variables definidas previamente, existe unanimidad, ya que todos ellos son de tipo *character*, ya que así es como está establecido en el proceso de exportación de datos del almacén de datos al archivo .csv que se me proporcionó. Este formato de dato es el equivalente en RStudio al tipo de dato tradicionalmente denominado *string* y que representa cadenas de texto.

2.2.3 Exploración inicial de la base de datos

En una primera exploración del almacén, se puede observar la gran riqueza que existe en cuanto a datos se refiere y la diversidad y heterogeneidad que hay a nivel geográfico, ya que posee datos de jugadores de todos los rincones del globo terráqueo, desde la *Ýokary Liga* (Turkmenistán) hasta la *Kolmonen* (Finlandia) pasando por la *Thai League* (Tailandia). Además de tener registradas todas estas ligas, también tiene registradas todas las diferentes divisiones que posee cada una de ellas, incluidas las ligas de desarrollo para jugadores jóvenes.

Por lógica común, en aquellas ligas situadas en países subdesarrollados, el proceso de registro y recolección del dato irá acorde al desarrollo del país, por lo que la calidad y consistencia de los datos a lo largo de las 190 variables distará mucho de los datos recogidos en los jugadores de las grandes ligas europeas, dónde el número de datos faltantes es nulo y la precision y calidad del dato es mucho mayor en comparación.

Continuando con el análisis inicial, se desarrolló la función *plot_distribución_jugadores()* que simplemente crea un gráfica interactivo a través de la librería *plotly* en el que se muestra un histograma que contiene el recuento que existe para cada una de las posiciones de campo dentro del dataset. El resultado se muestra en la figura 2.2

(*NOTA:* la variable mostrada en la figura 2.2 es una variable artifical llamada *"posicion_general´´* ya que en la variable original del almacén de datos que define la posición de un jugador dentro del campo, se realiza una segmentación bastante profunda, es decir, las principales posiciones dentro del campo son: defensa, centrocampista y atacante. Sin embargo, en el almacén de datos, dentro de los defensas, se realiza una segmentación entre defensa lateral y defensa central, y, a su vez, dentro de los defensas centrales se realiza una segmentación entre defensa central diestro y defensa central zurdo.

Debido a esto, dentro de la variable original del almacén, existe un gran número de subconjuntos, lo cuál, dificulta su visualización, por lo que se creó una variable artifical con un mayor nivel de abstracción, es decir, que a todos los defensas centrales, los etiqueta como defensas centrales, independientemente de si son centrales diestros o centrales zurdos (esto mismo se aplica para los centrocampistas y para los atacantes) para así poder obtener un gráfico más visual e interpretativo.)

2.2.4 Correlación entre variables

En el análisis de datos donde existe un número tan elevado de variables, uno de los primeros aspectos que se debe tener en cuenta es la correlación presente entre las variables del almacén de datos, ya que, a partir de los resultados obtenidos, se debe determinar la línea de trabajo a seguir.

Recurso	Coste
Recurso humano	5220 €
Recurso material	200 €
Coste Total	**5420€**

Tabla 2.2: Coste total asociado a este trabajo de fin de grado

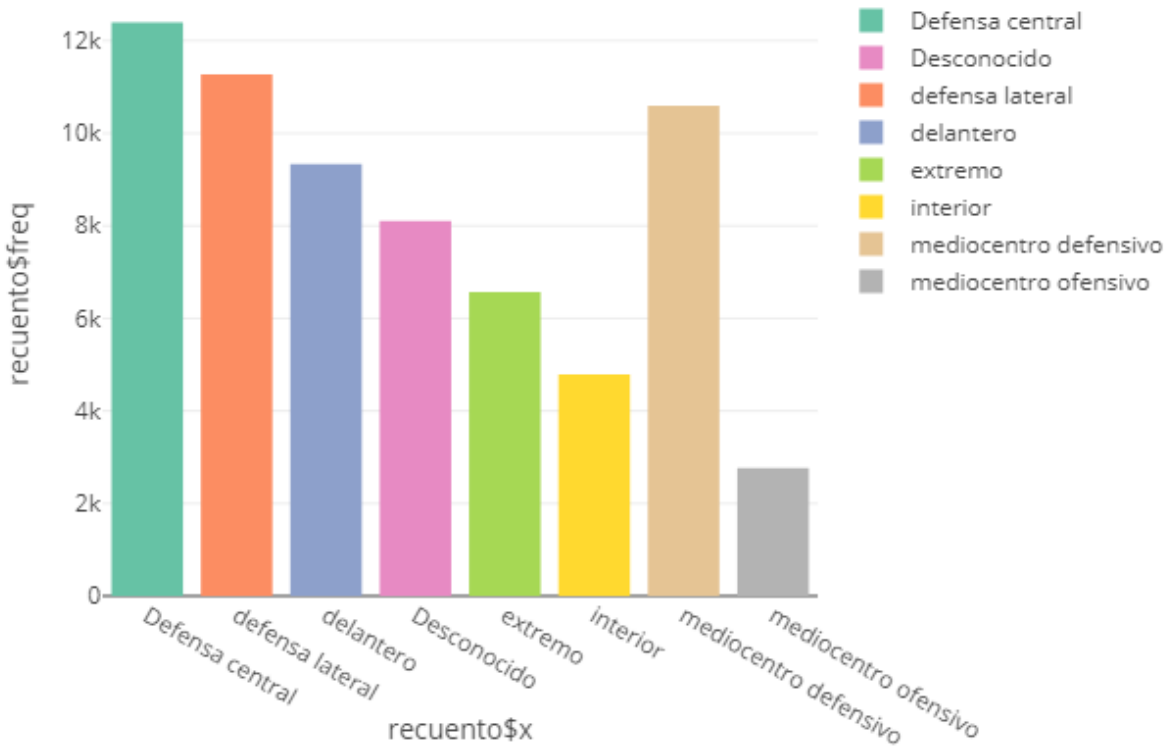


Figura 2.2: Distribución de la variable *"posicion_general´´* dentro del almacén

La correlación entre variables es una medida estadística que evalúa la relación lineal entre dos o más variables. Indica la fuerza y dirección de la asociación lineal entre ellas [12]. Un coeficiente de correlación cercano a una muestra una correlación positiva fuerte, donde las variables tienden a moverse juntas. Si el coeficiente es cercano a -1, indica una correlación negativa fuerte, dónde las variables se mueven en direcciones opuestas. Un coeficiente cercano a 0 implica una correlación débil o inexistente.

Para analizar la correlación entre variables en el almacén de datos proporcionado, se ha desarrollado la función *var_correlacionadas()* que utiliza como *inputs*: un dataset, su respectiva matriz de correlación y un umbral mínimo de correlación que se debe cumplir (sólo son relevantes las correlaciones próximas a -una o a 1), y como *output* genera una matriz donde la primera columna son el par de variables correlacionadas y la segunda columna el coeficiente de correlación entre ambas.

```
df[,c(4:6,8:32,34,35,37:42,
      44:64,66:70,72,74:78,80:91,
      93:95,97:118,120:128)]<- as.numeric(c(df[,4],df[,5],df[,6],df[,8],df[,9],df[,10],df[,11],df[,12],df[,13],df[,14],
                                            df[,15],df[,16],df[,17],df[,18],df[,19],df[,20],df[,21],df[,22],df[,23],df[,24],
                                            df[,25],df[,26],df[,27],df[,28],df[,29],df[,30],df[,31],df[,32],df[,34],df[,35],
                                            df[,37],df[,38],df[,39],df[,40],df[,41],df[,42],df[,44],df[,45],df[,46],df[,47],
                                            df[,48],df[,49],df[,50],df[,51],df[,52],df[,53],df[,54],
                                            df[,55],df[,56],df[,57],df[,58],df[,59],df[,60],df[,61],df[,62],df[,63],df[,64],
                                            df[,66],df[,67],df[,68],df[,69],df[,70],df[,72],df[,74],df[,75],df[,76],df[,77],
                                            df[,78],df[,80],df[,81],df[,82],df[,83],df[,84],df[,85],df[,86],df[,87],df[,88],
                                            df[,89],df[,90],df[,91],df[,93],df[,94],df[,95],df[,97],df[,98],df[,99],df[,100],
                                            df[,101],df[,102],df[,103],df[,104],df[,105],df[,106],df[,107],df[,108],df[,109],
                                            df[,110],df[,111],df[,112],df[,113],df[,114],df[,115],df[,116],df[,117],df[,118],
                                            df[,120],df[,121],df[,122],df[,123],df[,124],df[,125],df[,126],df[,127],df[,128]))
df[,43] <- as.factor(df[,43])
df[,65] <- as.Date(df[,65],df[,119])
```

Figura 2.3: Conversión de las variables a su formato correcto (*"numeric"*, *"character"*, *"factor"* o *"date"*) como se explica en la sección 2.2.2

2.2.5 Preproceso de los datos

Preproceso de las variables

Como se mencionó en la sección 2.2.2, únicamente existía un formato de dato para todas las variables, siendo un problema crítico, ya que, por definición propia de un gran número de variables, muchos datos deberían tener formato *numeric* porque representan aspectos cuantificables del juego. Además muchos de los algoritmos y funciones utilizados en este proyecto requieren de datos numéricos como *input* por lo que su formato ha de ser *numeric*, que es el equivalente en R al formato tradicionalmente conocido como *float*.

Para realizar esta tarea de la fase de preprocesamiento, se analizó de manera secuencial, una a una, todas las variables existentes dentro del almacén y, de manera manual, se modificó

el formato a de la variable a su valor correcto mediante la funcion *as.numeric()*, como se puede observar en la figura 2.3.

Filtrado de datos

Como se explicó en la sección 2.2.2, el almacén posee una gran cantidad de datos, y, en consecuencia, habrá muchos datos cuya calidad sea deficiente. Por ello, se desarrolló la función *filtrar_dataset()*, que es capaz de filtrar el almacén de datos original proporcionado por el R.C Deportivo de La Coruña en base a las condiciones que se especifique. En este caso, se decidió filtrar el almacén de datos con el objetivo de conseguir un almacén que contenga datos de calidad para obtener resultados concluyentes y precisos, facilitando así, la explicación del trabajo realizado a lo largo de este proyecto.

Siguiendo las recomendaciones del director empresarial, he decidido generar un almacén de datos reducido que contiene datos de calidad sobre los jugadores que cumplen las siguientes condiciones, y que generan resultados concluyentes y sencillos de validar:

- **Dicho jugador compite en LaLiga, Premier League o Primera Division RFEF:**

  LaLiga: Representa la competición de más alto nivel en España, por lo que los datos procedentes de sus jugadores serán datos precisos y de extrema calidad. Estos serán los datos que se utilizarán para realizar, el análisis de componentes principales y el clustering híbrido, explicado en las secciones 2.5.1 y 2.5.2.

  Premier League: Representa la competición de más alto nivel en Reino Unido, por lo que los datos procedentes de sus jugadores serán datos precisos y de extrema calidad.Estos serán los datos que se utilizarán para realizar las predicciones de jugadores extranjeros utilizando *machine learning* explicado en la sección 2.5.4.

  Primera Division RFEF: Representa la tercera competición de más alto nivel en España, en la que actualmente compite el R.C Deportivo de La Coruña, por lo que los datos procedentes de sus jugadores son datos ligeramente precisos pero con un margen de mejora bastante amplio en cuanto a calidad se refiere. Estos serán los datos que se utilizarán para aplicar la similitud coseno explicada en la sección 2.5.3.

- **Posición distinta a la de Portero:**

  Para este proyecto, aquellos jugadores cuya posición sea la de portero, no son útiles, ya que los objetivos planteados en la sección 2.1.1 se aplican a jugadores de campo y en ningún caso a los porteros.

- **Umbral de quinientos minutos jugados:**

  En este caso, el conjunto de jugadores que no superen este umbral, son jugadores que interesen dentro de este proyecto,a pesar de que puedan poseer valores extremadamente positivos para ciertas variables, ya que el objetivo final es encontrar posibles jugadores para ser fichados por el R.C Deportivo de La Coruña, y por tanto la consistencia de los datos es un pilar fundamental a tener en cuenta.

- **País de competición España o Reino Unido:**

  Como se explicó previamente, el número de ligas a nivel global es enorme, por lo que muchas de ellas comparten un mismo nombre. Debido a esto, países como Rusia también posee una competición llamada ”Premier League”, por lo que es necesario especificar los países de la competición, para así mantener únicamente los jugadores que nos interesan dentro del almacén.

En la figura 2.4 se muestra el ejemplo de uso de la función *filtrar_dataset()* para lograr obtener el almacén de datos ***df_jugadores***, que cumple los requisitos explicados previamente y que será **el almacén de datos en el que se basa este trabajo**:

```
df_jugadores <- filtrar_dataset(df_original,
                                name = list(condicion = "igual", valor = c("LaLiga",
                                                                           "Primera Division RFEF",
                                                                           "Premier")),
                                stcurr_texto_pos1_es = list(condicion = "desigual", valor = "Portero"),
                                stcurr_minutes_on_field = list(condicion = "mayor", valor = 500),
                                country = list(condicion = "igual", valor = c("Spain","England")))
```

Figura 2.4: Ejemplo de uso en el proyecto de la función *filtrar_dataset()*

A pesar de lo mencionado previamente, cabe destacar que, este proyecto, se ha planteado con el objetivo de desarrollar una herramienta de análisis de datos que sea capaz de generalizar su uso, y que el **único requisito necesario** para, obtener resultados como los que se muestran en este memoria, sea un **almacén de datos** que contenga datos relacionados con jugadores de fútbol y sus estadísticas durante un partido de fútbol.

Por ello, se explicará el proceso llevado a cabo para tratar de subsanar los problemas que se generan en caso de trabajar con un almacén de datos mediocres, tratando de mantener lo más alta posible la riqueza del almacén pero buscando unos mínimos de calidad en el dato.

Eliminación de variables estériles

Una vez realizado el filtrado a nivel de registros, es necesario filtrar las 190 variables iniciales para mantener únicamente aquellas que aporten información valiosa y sean de utilidad para el posterior análisis. Por ello, el primer filtrado que se realizó a nivel de variables fue eliminar aquellas variables en las que la proporción de elementos igual a cero es mayor o igual al 70%. La explicación de esta decisión es bastante sencilla, ya que, como se comentó en la sección anterior, el almacén original contenía registros de porteros, por lo que dentro del almacén también existen variables enfocadas a la recolección de datos en aspectos del juego que involucran exclusivamente a los porteros.
Por ello, el resto de registros de la base de datos que no son porteros, van a poseer valores igual a 0 en estas variables, ya que no realizan ninguna interacción en el aspecto que se está a medir.

En la figura 2.5 función *eliminar_variables_valores_unicos()* cuya funcionalidad es la que se muestra a continuación y que consiste en identificar aquellas variables que cumplen la condicion de que la media de datos igual a 0 en la respectiva variable sea superior a un umbral establecido por el usuario (normalmente el 90 por ciento). Posteriormente se eliminarán dichas variables del almacén, ya que una vez estudiado el almacén de datos original, dichas variables se relacionan con aspectos del juego que involucran a los porteros,y por ello, la mayoría de jugadores registran un valor igual a 0 para dichas variables, y la media de elementos igual a 0 es elevada. Una vez revisadas las variables eliminadas, se pueden recuperar aquellas variables que a pesar de poseer un alto valor de NAs siguen siendo variables de interés para el posterio análisis.
Este paso elimina un total de 24 variables del almacén de datos.

```
eliminar_variables_valores_unicos <- function(df,porcentaje_identicos){
  variables_eliminar <- names(colMeans(df == 0)[colMeans(df == 0) >= porcentaje_identicos])
  print(paste0("Se van a eliminar las siguientes variables del datastet: "))
  print(variables_eliminar)
  df <- df[, !names(df) %in% variables_eliminar]
  return(df)
}
df_jugadores <- eliminar_variables_valores_unicos(df_jugadores,0.9)
```

Figura 2.5: Eliminación de variables estériles mediante la función *eliminar_variables_valores_unicos()*

Eliminación de variables con elevado porcentaje de NAs

El siguiente paso respecto al filtrado de variables fue la eliminación de aquellas variables que posean un elevado numero de *NAs* registrados, que en Rstudio, es el que corresponde a valores perdidos o un elevado numero de valores "", que corresponde al *NA*. En la figura 2.6

se muestra la función *porcentaje_NA_variables()* que se basa en el mismo procedimiento de la sección anterior, pero en este caso detectando valores perdidos. Este paso eliminó un total de 4 variables del almacén de datos.

```
porcentaje_NA_variables <- function(df){
  porcNA_jugadores <- round(colMeans(is.na(df) | df == "") * 100,2)
  return(porcNA_jugadores[porcNA_jugadores > 0])
}
porcentaje_NA_variables(df_jugadores)
```

Figura 2.6: Función para detectar el porcentaje de *NA*

2.2.6 Filtrado por formato de dato

Como se comentó en la sección 2.2.2, el único formato de dato existente originalmente dentro del almacén de datos era *character*, sin embargo en la sección 2.2.5 se modificaron ciertos variables. Como los algoritmos que se usará son capaces de procesar datos númericos, llegados a este punto, se filtró el almacén de datos para que solo permaneciesen las variables con dicho formato de datos, como se muestra en la figura 2.7, y así poseer únicamente datos númericos dentro del almacén. Este paso elimino un total de 24 variables del almacén.

```
# SUBCONJUNTO VARIABLES NUMERICAS
df_jugadores_num <- df_jugadores[, sapply(df_jugadores, is.numeric)]
```

Figura 2.7: Eliminación de aquellas variables cuyo formato no sea *numeric*

Filtrado manual

Por último, con la ayuda de mi director del departamento de Análisis de datos del R.C.Deportivo de La Coruña,en la figura 2.8, se muestra un filtrado manual sobre variables que se consideraban inútiles en cuánto a su aportación para tratar de conseguir los objetivos propuestos. En este paso se eliminaron un total de 31 variables.

Substitución de valores NA

Una vez realizado el filtrado de datos a nivel de registros y de variables, los datos restantes dentro del almacén ya son datos de valor y con los que se puede comenzar a trabajar. Sin embargo, uno de los problemas encontrados más adelante fue la aparición de valores NA

distribuidos de manera aleatoria a lo largo de todas las variables, sin seguir algún patrón concreto mediante el cuál poder subsanar dicho problema. Por tanto, tras analizar el contexto en el que se producen estos NAs y consultándolo con mis directores, se llegó a la conclusión de que lo más adecuado era substituir estos valores NA por valores igual a 0, como se muestra en la imagen 2.9.

2.3 Análisis estadístico del almacén de datos

En este capítulo se van a estudiar la naturaleza, distribución y características de los datos desde un punto de vista estadístico, para detectar si existe algún comportamiento extraño en alguna de las variables y ser conscientes de ello a lo largo del proyecto. Las medidas estadísticas y gráficos utilizados para este análisis son los siguientes:

1. **Desviación típica**

   La desviación típica [13], también conocida como desviación estándar, es una medida estadística que cuantifica la dispersión o variabilidad de un conjunto de datos en relación con su media. Indica qué tan dispersos están los valores individuales con respecto a la media.

   Matemáticamente, la desviación típica se calcula dle siguiente modo:

   $$S = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}}$$

   Una desviación típica baja indica que los datos tienden a estar cerca de la media y que hay poca dispersión entre ellos. Por otro lado, una desviación típica alta indica una mayor dispersión, es decir, que los datos están más alejados de la media y se distribuyen en un rango más amplio.

   La desviación típica es una medida útil para comparar la dispersión de diferentes conjuntos de datos y comprender la variabilidad inherente a un conjunto de observaciones. Además, se utiliza en muchas técnicas estadísticas y en la interpretación de resultados en diferentes campos de estudio.

2. **Curtosis y Asimetría**

   La *curtosis* [14]y la *asimetría* [15] son medidas estadísticas utilizadas para describir la forma de la distribución de un conjunto de datos.

   La curtosis se refiere a la medida de la concentración de los datos alrededor de la media de una distribución. Indica si los valores se encuentran principalmente cerca de la media o si hay una dispersión más amplia en las colas de la distribución. Un valor de curtosis

```r
# SUBCONJUNTO VARIABLES NUMERICAS
df_jugadores_num <- df_jugadores[, sapply(df_jugadores, is.numeric)]

# BORRADO MANUAL DE VARIABLES
borrar <- c("id_wy","stcurr_current_team_id",
            "stcurr_current_team_color","stcurr_on_loan",
            "stcurr_height","stcurr_valor",
            "stcurr_market_value","stcurr_total_matches",
            "stcurr_matches_in_start","stcurr_matches_substitued",
            "stcurr_xg_save_avg","stcurr_shot_block_avg",
            "stcurr_goalkeeper_punch_avg","stcurr_goalkeeper_exits_avg",
            "stcurr_conceded_goals_avg","stcurr_gk_aerial_duels_won",
            "stcurr_goalkeeper_punch_accuracy","stcurr_goalkeeper_claim_avg",
            "stcurr_gk_aerial_duels_avg","stcurr_shots_against_avg",
            "stcurr_xg_save","stcurr_matches_played_for_national_team",
            "stcurr_save_percent","stcurr_age","stcurr_primary_position_percent",
            "stcurr_second_position_percent",
            "stcurr_third_position_percent","X","compe_id","conteo","division")
df_jugadores_num <- df_jugadores_num[ , !(names(df_jugadores_num) %in% borrar)]
```

Figura 2.8: Eliminación manual de aquellas variables acordadas con el tutor

```r
# SUSTITUCION DE NA´S por 0
df_jugadores_num[is.na(df_jugadores_num)] <- 0
```

Figura 2.9: Sustitución de los valores *NA* en el almacén por 0

alto indica una mayor concentración en torno a la media, mientras que un valor bajo indica una dispersión más uniforme. Matemáticamente se calcula del siguiente modo:

$$\text{Curtosis} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^4}{n \cdot S^4} - 3$$

Por otro lado, la *asimetría* mide la falta de simetría en una distribución. Puede ser positiva o negativa, lo que indica la dirección del sesgo de los datos. Una asimetría positiva significa que la cola derecha de la distribución es más larga o más pesada, mientras que una asimetría negativa indica que la cola izquierda es más larga o más pesada. Un valor de asimetría cero indica una distribución perfectamente simétrica. Matemáticamente se calcula del siguiente modo:

$$\text{asimetría} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^3}{n \cdot S^3}$$

Ambas medidas, curtosis y asimetría, proporcionan información importante sobre la forma y el comportamiento de los datos, lo que puede ser útil en el análisis estadístico y la comprensión de las características de un conjunto de datos determinado.

3. **Rango intercuartilico**

   El rango intercuartílico (IQR, por sus siglas en inglés) [16] es una medida estadística utilizada para describir la dispersión de un conjunto de datos. Se calcula como la diferencia entre el tercer cuartil (Q3) y el primer cuartil (Q1) de un conjunto de datos.

   El primer cuartil (Q1) representa el valor que divide a los datos en el 25% inferior de la distribución, mientras que el tercer cuartil (Q3) divide a los datos en el 75% inferior de la distribución. Al restar Quna de Q3, se obtiene el rango intercuartílico, que abarca el rango intermedio de los datos donde se encuentra la mitad central de la distribución.

   El rango intercuartílico es una medida robusta frente a valores atípicos o extremos, ya que se basa en los valores que se encuentran en los percentiles 25% y 75% de los datos, excluyendo los valores más extremos. Proporciona una medida de dispersión que se centra en los datos más representativos y evita la influencia de valores atípicos que pueden sesgar otras medidas de dispersión, como el rango o la desviación estándar.

   El rango intercuartílico es útil para identificar la variabilidad en el rango medio de los datos y puede ser utilizado para detectar valores atípicos.

4. **Histograma**

   Un histograma [17] es una representación gráfica de una distribución de datos numéricos en forma de barras. Es una herramienta visual utilizada en estadística para analizar la frecuencia con la que ocurren diferentes valores o rangos de valores dentro de un conjunto de datos.

5. **Boxplot**

   Un boxplot [18], también conocido como diagrama de caja y bigotes, es una representación gráfica que muestra la distribución de un conjunto de datos numéricos a través de cinco estadísticas principales: el valor mínimo, el primer cuartil (Q1), la mediana (Q2), el tercer cuartil (Q3) y el valor máximo.

   Proporciona información valiosa sobre la distribución y las características de los datos, incluyendo la simetría, la dispersión, la presencia de valores atípicos y la concentración de los datos alrededor de la mediana. También es útil para comparar diferentes conjuntos de datos y detectar diferencias en sus distribuciones.

Una vez se han definido cada una de las medidas y gráficos utilizados y explicado los rasgos que identifican cada una de ellas, se procede a mostrar con imagenes los resultados que genera la función *plots_variables_numericas_jugadores()* que simplemente necesita como *input*, un dataframe que contenga variables numéricas para proceder a analizarlas secuencialmente.

En la figura 2.10, se puede apreciar, para la variable *stcurr_cross_from_left_avg*, que los datos están muy concentrados (como indica el IQR, la curtosis y la desviación tipica) alrededor de la media, que posee un valor de 0.67, la cual es relativamente baja, y con una larga cola hacia la derecha en su distribución (como indica la asimetría). La explicación de esto es que dicha variable cuantifica el número de centros por partido que el jugador realiza desde la banda izquierda, y por naturaleza del juego, la gran mayoría de jugadores no realiza esta acción durante el partido, excepto los especialistas, los cuáles son aquellos registros detectados en el boxplot de la figura 2.11 y mostrados de manera numérica por pantalla.

Por limitaciones de espacio en la memoria, se ha mostrado únicamente el análisis de una de las variables, sin embargo, durante el proyecto se han realizado análisis de cada una de las variables y se han extraído las siguientes deducciones sobre los datos:

1. No se cumple la hipótesis de normalidad para la mayoría de las variables.

2. Las variables que guardan relación entre sí dentro del juego, poseen varianzas similares.

```
[1] "*****************************************************************************"
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.0400  0.2300  0.6798  0.8100  7.5800
Standar Desviation  stcurr_cross_from_left_avg =  1.046762
IQR stcurr_cross_from_left_avg =  0.77
Coef Asimetria stcurr_cross_from_left_avg =  2.357095
Kurtosis stcurr_cross_from_left_avg =  9.116094
Datos Atipicos stcurr_cross_from_left_avg 2.81 2.67 3.41 2.63 2.72 3.04 2.73 4.46 7.58 2.57 3.11
 2.12 4.35 4.52 3.01 2.38 3.32 4.39 2.13 3.51 2.16 2.59 5.28 3.16 3.52 2.31 3.66 2.46 2.96 2.42
 4.42 2.64 2.1 2.74 2.67 2.04 2.96 1.99 5.57 2.84 2.57 3.03 2.79 3.16 5 5.22 3.09 2.33 3.02 3.54
 2.08 3.39 4.59 4.16 2.06 4.36 2.53 3 3.37 4.85 3.38 3.72 2.18 4.24 5.02 2.83 2.49 3.09 3.18 2.68
 3.4 2.26 4.51 2.59 2.28 2.64 2.49 2.85 3.27 4.6 5.94 2.33 2.94 4.18 2.52 2.14 2.07 2.4 4.63 3.53
 4.22 2.29 3.39 2.49 2.81 2.43 4.29 2.48 3.15 3.16 2.53 2.37 2.21 2.92 3.43 2.64 2.97 2.57 3.37
 4.52 3.52
```

Figura 2.10: Medidas estadísticas para la variable *stcurr_cross_from_left_avg*

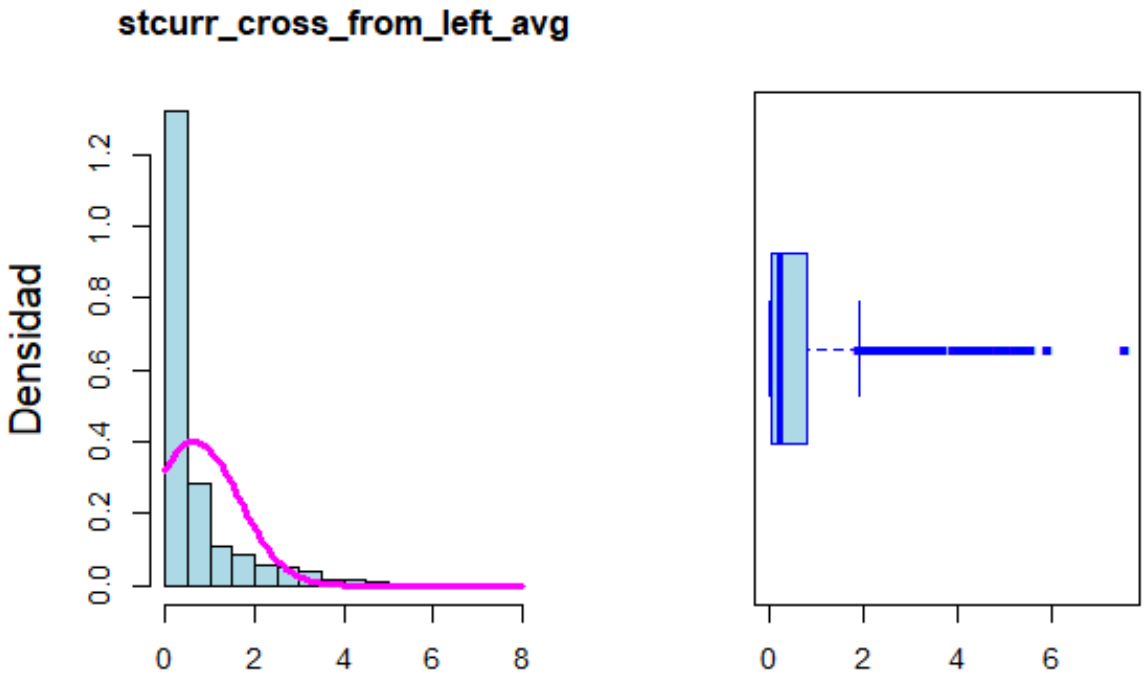


Figura 2.11: Histograma y Boxplot para la variable *"stcurr_cross_from_left_avg"*

3. En algunas de las variables, un elevado porcentaje de los datos tiene un valor igual a 0

4. Existe un gran abanico de escalas dentro de las variables, yendo desde magnitudes decimales (media de tiros por partidos) hasta magnitudes de varios dígitos (valor de mercado de un jugador).

En conclusión, existe una gran variabilidad dentro de los datos debido a que el número de situaciones y contextos que cubren el total las variables es muy grande, por lo que será necesario homogeneizar el almacén de datos para evitar que se generen sesgos en los algoritmos de *machine learning* hacia las variables que poseen un mayor rango de valores (como por ejemplo, el valor de mercado) y conseguir una mejor eficiencia a la hora de procesar los datos.

2.4 Normalización del almacén de datos

Una vez realizado el análisis de los datos, se decidió normalizar el almacén de datos. Este proceso consiste en transformar los valores de las variables para tratar de homogeneizar la estructura entera con el objetivo de que todas las variables tengan una escala común y se muevan en rangos de valores similares, para evitar cualquier tipo de sesgo en el procesamiento de los datos[19]. Dentro de la normalización existe dos métodos de referencia:

1. **Normalización min-max:[20]**

   Escala los valores de la variable para que se encuentren entre un rango específico, generalmente entre 0 y 1. Se utiliza la siguiente fórmula:

$$x_{\text{norm\_Min\_Max}} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

2. **Normalización Z-score:[20]**

   Transforma los valores de la variable para que tengan una media de 0 y una desviación estándar de 1. Se utiliza la siguiente fórmula:

$$x_{\text{norm\_Z\_Score}} = \frac{x - \text{mean}(x)}{\text{std}(x)}$$

La diferencia entre ambos es la escala final resultante. Inicialmente se decidió aplicar ambos métodos al almacén de datos con el objetivo de realizar una comparación de los resultados. Finalmente se decidió normalizar el almacén de datos con la **normalización Z-score**, ya que se obtenían resultados mucho más precisos y satisfactorios a la hora de realizar el clustering.

2.5 Técnicas utilizadas

En este capítulo se va a proceder a explicar las técnicas de procesamiento de datos aplicadas al almacén de datos, el cuál, ha sido filtrado y normalizado.

2.5.1 Análisis de componentes principales (PCA)

En este punto, a pesar de haber realizado un filtrado de variables en las secciones previas, el almacén de datos sigue constando de 89 variables, el cúal es un número demasiado elevado en el que seguramente haya patrones internos entre variables y redundancia entre datos que nos permitan comprimir y reducir dicho numero de variables.

Para ello se aplica un análisis de componentes principales (PCA) a los datos [21], que es una técnica de reducción de dimensión utilizada en *machine learning* y estadística para identificar patrones y reducir la complejidad de datos de alta dimensión.

Tiene como objetivo transformar un conjunto de datos compuesto por variables posiblemente correlacionadas en un nuevo conjunto de variables no correlacionadas llamadas componentes principales, donde cada componente explica un porcentaje determinado de la información contenida dentro del almacén de datos. Estos componentes son combinaciones lineales de las variables originales y se ordenan de mayor a menor siguiendo el criterio de porcentaje de variabilidad explicada.

De este modo, analizando las combinaciones lineales de las componentes resultantes de aplicar PCA, podremos obtener una primera aproximación de los distintos perfiles de jugadores y el significado inherente a cada componente, a través de los coeficientes asignados en cada componente a cada una de las variables.

En la figura 2.12 se muestran el porcentaje de variabilidad asociada a cada componente.

Como se puede apreciar, las tres primeras componentes aglutinan la mitad de la variabilidad explicada en el almacén, con un total de 50.7% de variabilidad. Esto es un hecho a tener en cuenta, ya que éste ha sido el número de componentes que se han seleccionado para realizar este proyecto.

En este caso, la relación pérdida de información / reducción de la dimensión es bastante positiva, al tener la posibilidad de pasar de 89 variables a tres variables, y mantener la mitad de la información del dataset.

Además, en caso de querer reducir la pérdida en el porcentaje de variabilidad explicada, simplemente habría que aumentar el número de componentes que se utilizan para reducir la dimensión del almacén.

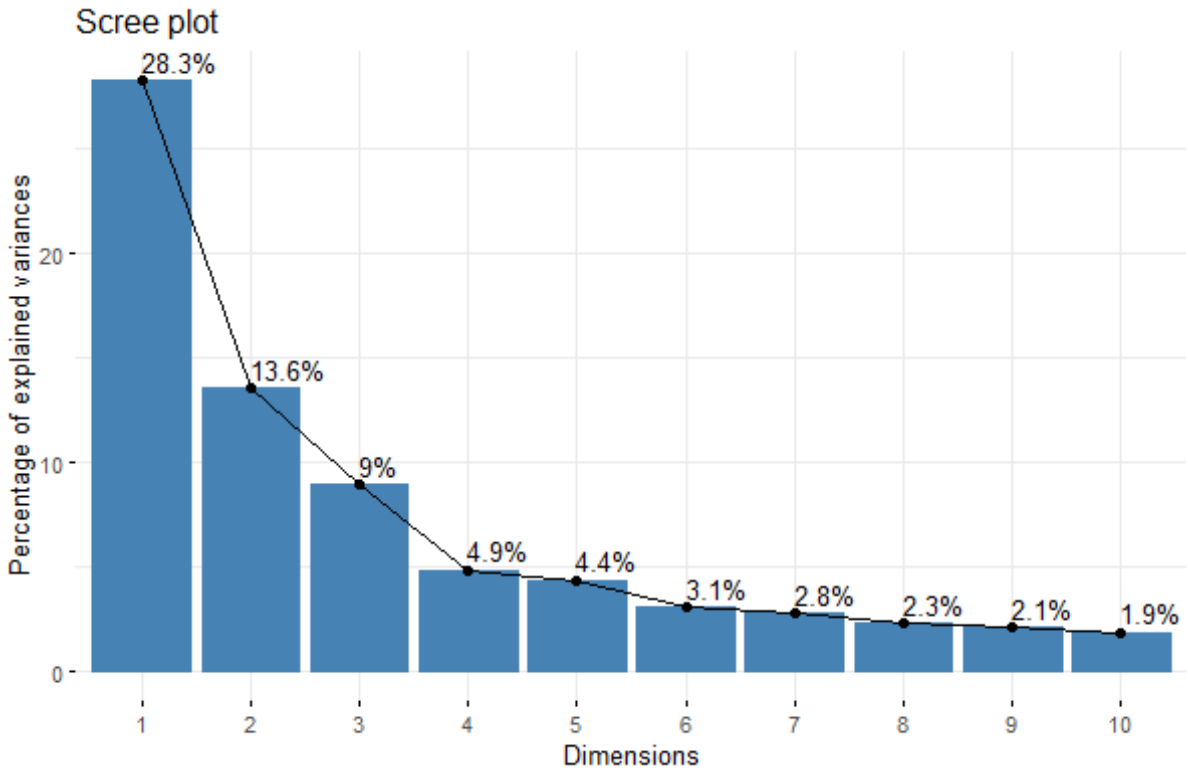


Figura 2.12: Porcentaje de variabilidad explicada por cada componente (sección 2.5.1)

Por otro lado, mediante el uso del paquete *factoextra()* se pueden generar gráficos como el de la figura 2.13 donde se muestra la contribución que aporta cada variable a la primera componente (eje X) y a la segunda componente (eje Y), en base a aplicar los cuadrados de los cosenos de los ángulos entre las variables originales y los ejes de los componentes principales. La leyenda de colores nos ayuda a interpretar mejor el gráfico, a continuación se muestra el significado de cada color:

- **Color Verde:**

  La variable esta altamente asociada a dicha componente y tendrá un papel relevante dentro de ella.

- **Color Naranja:**

  La variable se ha de tener en cuenta a la hora de interpretar la respectiva componente pero con un nivel de relación menor al verde.

- **Color Negro:**

  La variable es irrelevante a la hora de analizar la componente.

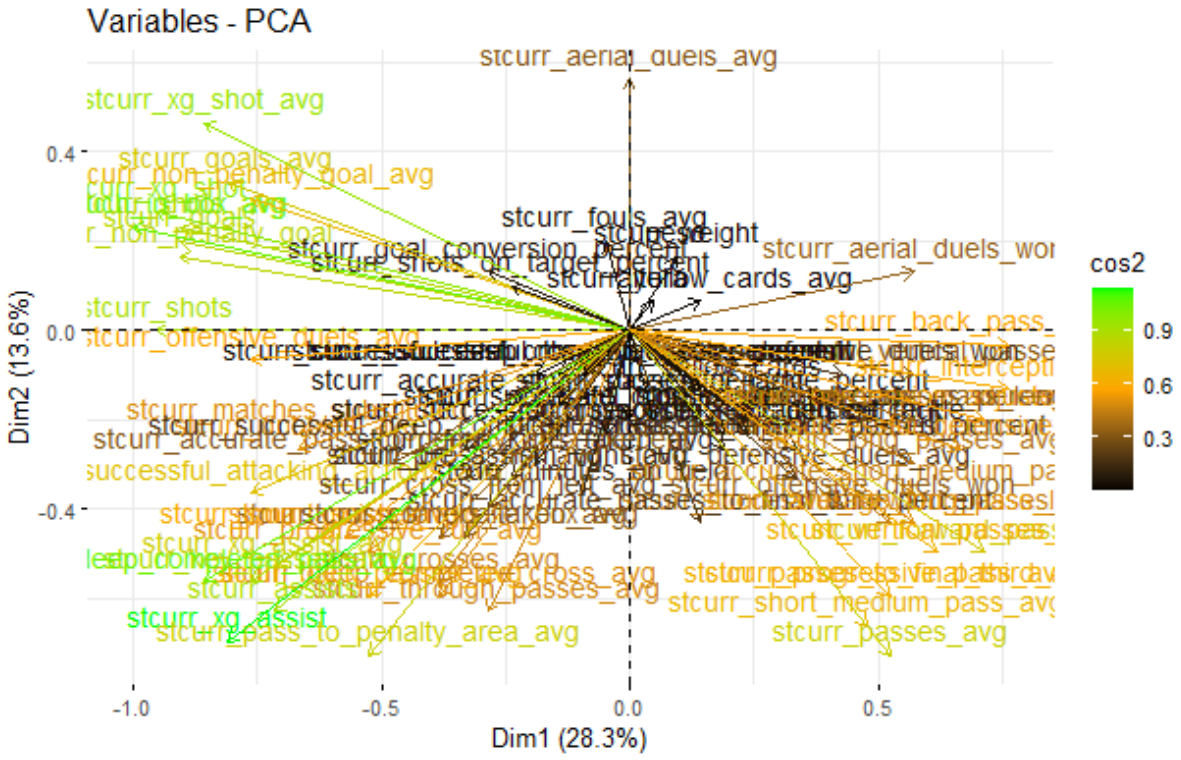


Figura 2.13: Contribución de las variables a las dos primeras componentes principales PCA (sección 2.5.1)

Como se puede apreciar, este gráfico es una buena herramienta para realizar una primera aproximación a la interpretación de las dos componentes que mayor porcentaje de variabilidad explican, aunque a continuación, se procede a explicar el análisis que se ha realizado de manera más profunda.

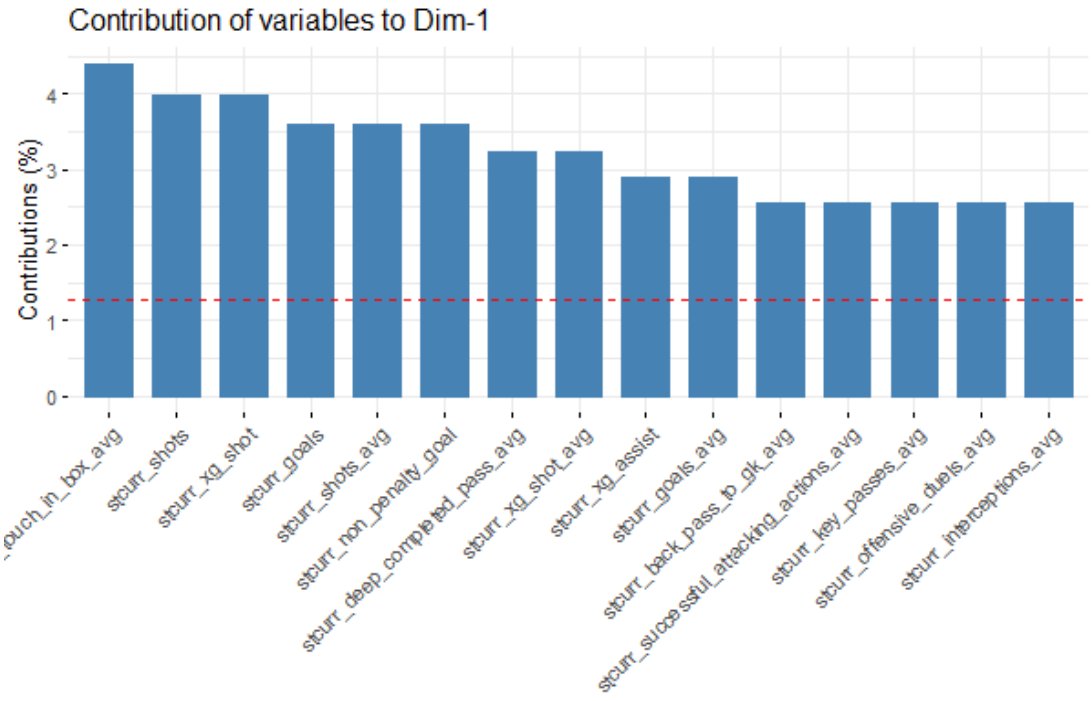
- **Interpretación de las tres primeras componentes principales**

  En esta sección se tratará de dar una explicación al significado e interpretación de cada una de las tres componentes utilizadas en este proyecto en relación al perfil de jugador asociado.
  Para ello, se han analizado los coeficientes asociados a cada variable y los jugadores cuya contribución es mayor para cada un de ellas respectivamente a través de las funciones *variables_influyentes()* y *jugadores_influyentes()* que reciben como parámetros el número de la componente a analizar y el N ranking de variables o jugadores a mostrar.

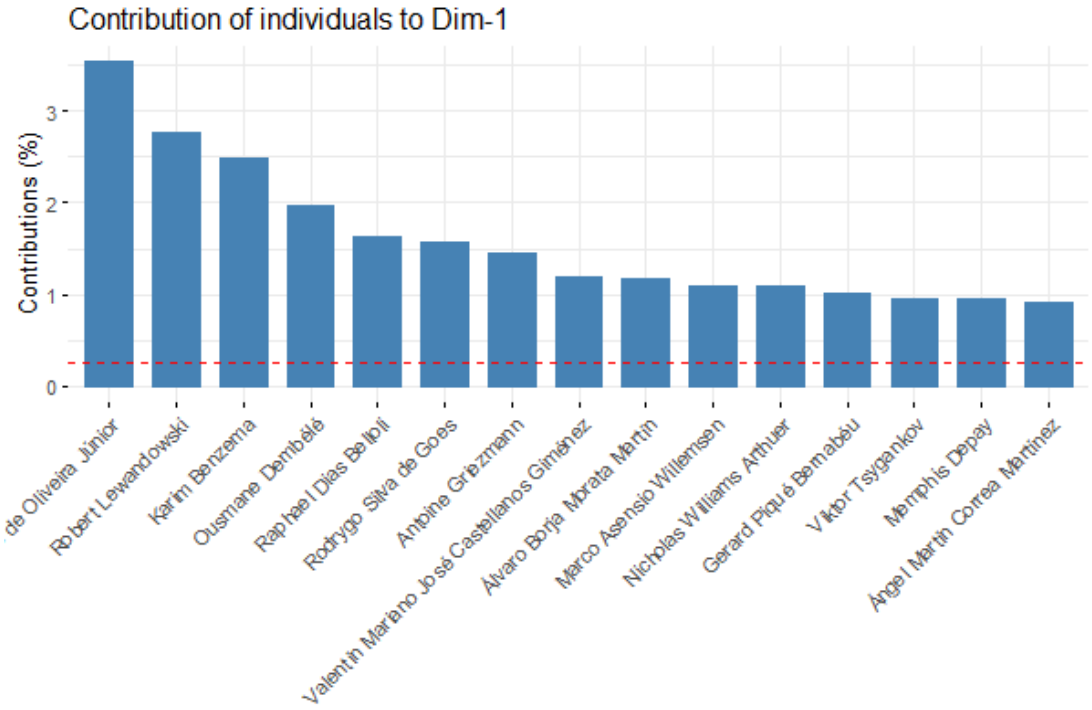
  - **Primera componente principal:**
    En la figura 2.14 se muestran las 15 variables y jugadores más influyentes. Analizando las variables, se puede apreciar como tres de las 15 son de caracter ofensivo y relacionadas con la posición de delantero, por ejemplo, *"stcurr_touch_in_box_avg´´* o *"stcurr_shots´´*. Respecto a los jugadores con mayor contribución dentro de la componente, la línea a seguir es la misma, 14 de los 15 jugadores son de caracter ofensivo y específicamente juegan de delanteros. Por tanto se puede concluir que uno de los perfiles de jugador asociados con esta componente es el de delantero.

    Sin embargo, se decidió realizar un análisis más profundo de la componente para tratar de dar explicación a esas dos variables y al jugador que representa un perfil totalmente radical al de la mayoría. Por eso se desarrolló la función *datos_jugadores_pca()* que devuelve los coeficientes asociados a cada variable, y los respectivos valores de los jugadores más contribuidores para ellas. En la figura 2.15 se muestra los coeficientes de la primera componente y los valores que toma los jugadores más relevantes, destacando a Gerard Piqué Bernabéu:
    Se puede apreciar, como todas las variables relacionadas con los aspectos ofensivos tienen un aspecto en común, concretamente, los coeficientes que reciben son de caracter negativo, por tanto, un valor negativo de un jugador para la primera componente se interpretará como que dicho jugador tiene un elevado potencial como jugador ofensivo y como delantero concretamente.

    Por otro lado, las dos variables que contrastan con las mayoría son variables relacionadas con jugadores defensivos concretamente con la posición de central (*"st-*



(a) Variables más relevantes



(b) Jugadores más relevantes

Figura 2.14: Primera componente principal PCA (sección 2.5.1
)

curr_back_pass_to_gk_avg" y *"stcurr_interceptions_avg"*). Ambas tienen un aspecto en común, concretamente, ambos coeficientes son positivos, por tanto, un valor positivo de un jugador arbitratio para la primera componente se interpretará como que dicho jugador tiene un elevado potencial como jugador defensivo y como central concretamente.

- **Segunda componente principal**

  El procedimiento para analizar esta componente es idéntico, primero se comprueban variables y jugadores que más contirbuyen a la componente para observar si existe unanimidad o si hay varios perfiles de jugador involucrados.

  En caso de haber unanimidad, se puede deducir la interpretración de la componente a partir de estas dos fuentes de información y en caso de haber discrepancia, es necesario analizar una tercera fuente de información que son los coeficientes de las variables, para observar la segmentación que realiza entre las variables que poseen coeficientes positivos y negativos. Los resultados de esta componente se muestran en la figura 2.16

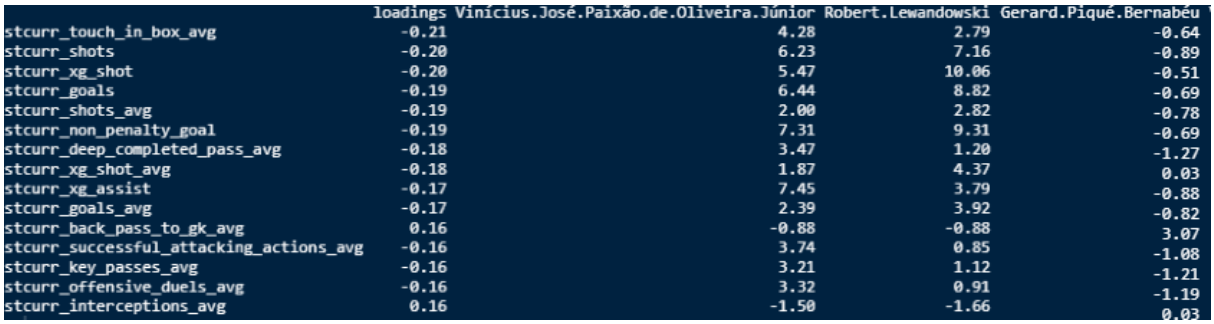
  En este caso, vemos como la mayoría de variables y jugadores influyentes están relacionadas con aspectos de creación de juego y especialistas en obtener asistencias (*"stcurr_passes_avg´´*, *"stcurr_xg_assist´´*, *"stcurr_short_medium_pass_avg´´*, etc) por lo que analizando el coeficiente de estas variables, se puede observar como todos ellos son negativos, y por tanto un valor negativo para esta componente representa el perfil de jugador mencionado previamente.

  Por otro lado, el subconjunto de variables con coeficiente positivo (*"stcurr_aerial_duels_avg´´*, *"stcurr_xg_shot_avg´´*), están relacionadas con un perfil muy concreto de jugador, que se trata de jugadores que juegan de delanteros referencia y son especialistas en el juego aéreo con poca movilidad dentro del campo.

- **Tercera componente principal**

  En esta última componente el procedimiento es el mismo.

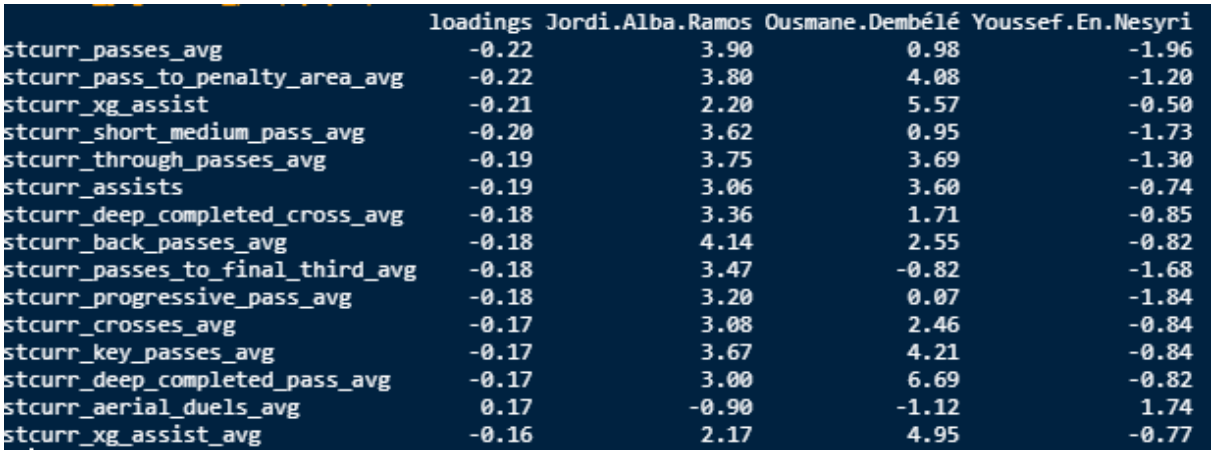
  Para esta componente, en la figura 2.17 se puede detectar como la segmentación realizada por los coeficientes positivos utiliza variables similares a las usadas para representar a los delanteros en la primera componente, sin embargo, introduce ciertas variables que diferencian un perfil concreto dentro de esta posición como *"stcurr_accurate_passes_percent´´*, *"stcurr_successful_vertical_passes_percent´´* o *"stcurr_accurate_passes_to_final_third_percent´´*, que son características de un delantero con una movilidad alta dentro del campo y con una participación activa



	loadings	Vinícius.José.Paixão.de.Oliveira.Júnior	Robert.Lewandowski	Gerard.Piqué.Bernabéu
stcurr_touch_in_box_avg	-0.21	4.28	2.79	-0.64
stcurr_shots	-0.20	6.23	7.16	-0.89
stcurr_xg_shot	-0.20	5.47	10.06	-0.51
stcurr_goals	-0.19	6.44	8.82	-0.69
stcurr_shots_avg	-0.19	2.00	2.82	-0.78
stcurr_non_penalty_goal	-0.19	7.31	9.31	-0.69
stcurr_deep_completed_pass_avg	-0.18	3.47	1.20	-1.27
stcurr_xg_shot_avg	-0.18	1.87	4.37	0.03
stcurr_xg_assist	-0.17	7.45	3.79	-0.88
stcurr_goals_avg	-0.17	2.39	3.92	-0.82
stcurr_back_pass_to_gk_avg	0.16	-0.88	-0.88	3.07
stcurr_successful_attacking_actions_avg	-0.16	3.74	0.85	-1.08
stcurr_key_passes_avg	-0.16	3.21	1.12	-1.21
stcurr_offensive_duels_avg	-0.16	3.32	0.91	-1.19
stcurr_interceptions_avg	0.16	-1.50	-1.66	0.03

Figura 2.15: Valores de Coeficientes y Jugadores para la primera componente principal PCA (sección 2.5.1

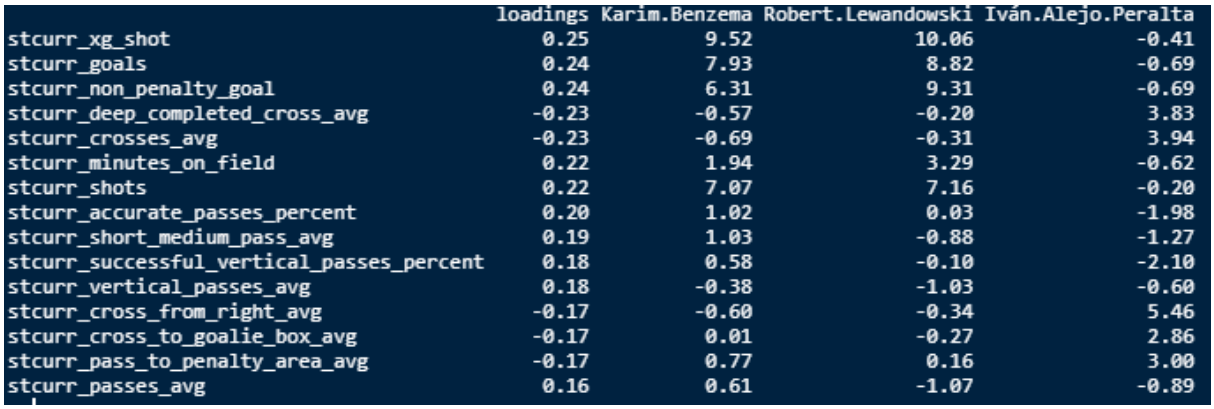
)



	loadings	Jordi.Alba.Ramos	Ousmane.Dembélé	Youssef.En.Nesyri
stcurr_passes_avg	-0.22	3.90	0.98	-1.96
stcurr_pass_to_penalty_area_avg	-0.22	3.80	4.08	-1.20
stcurr_xg_assist	-0.21	2.20	5.57	-0.50
stcurr_short_medium_pass_avg	-0.20	3.62	0.95	-1.73
stcurr_through_passes_avg	-0.19	3.75	3.69	-1.30
stcurr_assists	-0.19	3.06	3.60	-0.74
stcurr_deep_completed_cross_avg	-0.18	3.36	1.71	-0.85
stcurr_back_passes_avg	-0.18	4.14	2.55	-0.82
stcurr_passes_to_final_third_avg	-0.18	3.47	-0.82	-1.68
stcurr_progressive_pass_avg	-0.18	3.20	0.07	-1.84
stcurr_crosses_avg	-0.17	3.08	2.46	-0.84
stcurr_key_passes_avg	-0.17	3.67	4.21	-0.84
stcurr_deep_completed_pass_avg	-0.17	3.00	6.69	-0.82
stcurr_aerial_duels_avg	0.17	-0.90	-1.12	1.74
stcurr_xg_assist_avg	-0.16	2.17	4.95	-0.77

Figura 2.16: Segunda componente principal PCA (sección 2.5.1

)



	loadings	Karim.Benzema	Robert.Lewandowski	Iván.Alejo.Peralta
stcurr_xg_shot	0.25	9.52	10.06	-0.41
stcurr_goals	0.24	7.93	8.82	-0.69
stcurr_non_penalty_goal	0.24	6.31	9.31	-0.69
stcurr_deep_completed_cross_avg	-0.23	-0.57	-0.20	3.83
stcurr_crosses_avg	-0.23	-0.69	-0.31	3.94
stcurr_minutes_on_field	0.22	1.94	3.29	-0.62
stcurr_shots	0.22	7.07	7.16	-0.20
stcurr_accurate_passes_percent	0.20	1.02	0.03	-1.98
stcurr_short_medium_pass_avg	0.19	1.03	-0.88	-1.27
stcurr_successful_vertical_passes_percent	0.18	0.58	-0.10	-2.10
stcurr_vertical_passes_avg	0.18	-0.38	-1.03	-0.60
stcurr_cross_from_right_avg	-0.17	-0.60	-0.34	5.46
stcurr_cross_to_goalie_box_avg	-0.17	0.01	-0.27	2.86
stcurr_pass_to_penalty_area_avg	-0.17	0.77	0.16	3.00
stcurr_passes_avg	0.16	0.61	-1.07	-0.89

Figura 2.17: Tercera componente principal PCA (sección 2.5.1

)

en la creación de juego.

Respecto a las variables con coeficiente negativo, la mayoría están involucradas en acciones generadas desde las bandas del terreno de juego y con la acción de envíar balones al área desde estas posiciones (*"stcurr_cross_from_right_avg´´*, *"stcurr_cross_from_left_avg´´*, *"stcurr_pass_to_penalty_area_avg´´*, etc). Dentro de los distintos perfiles de jugador, existe un perfil que posee valores muy elevados para estas variables, que son la de extremo y la de lateral ofensivo. Dentro de los coeficientes negativos, también existe un subconjunto minoritario de variables relacionadas con mediocentros puros, que acumulan un grán número de pases por partido y están en contacto con el balón durante elevados períodos de tiempo(*"stcurr_deep_completed_cross_avg´´ y "stcurr_crosses_avg´´*), de ahí surge la existencia de jugadores con valores elevados para dichas variables y que se identifican con este perfil concreto dentro de los jugadores más relevantes para esta componente.

Para resumir, antes de comenzar esta sección, la dimensión del almacén era de 89 variables, y una vez aplicado el procedimiento explicado en la sección, se decidió reducir la dimensión del almacén a tres componentes como se muestra en la figura 2.18 (la cuál ha sido generada mediante la herramienta web *plotly*[1]) ya que a partir de la interpretación de esos 3 valores, es posible abarcar la mayoría de posiciones dentro del campo, desde defensas hasta delanteros. En dicha figura, la intensidad de color de cada punto esta asociada con el valor que toma para la respectiva componente de uno de los 3 ejes.

La reducción de dimensión se realiza a través del producto de cada una de las variablas por su respectivo coeficiente y posteriormente la suma de los 89 valores, para generar un único valor asociado a cada correspondiente componente.

A continuación, se enumeran los objetivos conseguidos en esta sección:

1. Reducción de la dimensión del almacén a tres componentes.
2. Compresión de la información para ser capaces de procesarla de forma más rápida.
3. Capacidad de representación visual de los datos en un gráfico de 3D.

[1] Enlace al gráfico: https://chart-studio.plotly.com/~arodriguezperez85/1?share_key=pAVse8tfqC3KmbZ4jldrMP

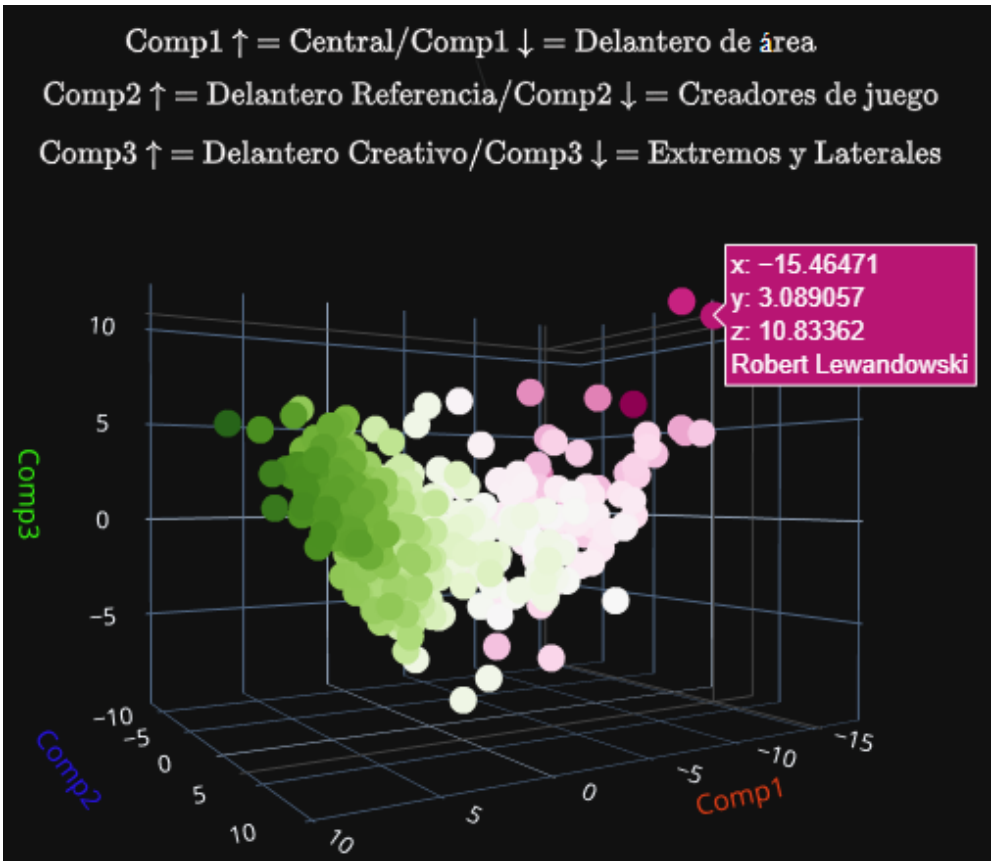


Figura 2.18: Gráfico 3D con las 3 primeras componentes principales de la sección 2.5.1

4. Primera aproximación a la deducción del perfil de un jugador.

2.5.2 Clustering híbrido

En esta sección, se explica el procedimiento seguido para aplicar la técnica de clustering al almacén de datos [22], que se trata de una ténica de aprendizaje no supervisado, cuyo objetivo es detectar patrones subyacentes en los datos que permitan generar distintos *clusters* donde cada uno de ellos represente un perfil de jugador concreto.

- **Selección de parámetros**:

  Dentro de cada *cluster* habrá un grupo de jugadores cuya distancia espacial es relativamente baja por lo que se interpreta que dichos jugadores son similares entre sí. Por otro lado, la distancia espacial entre cada grupo será lo más grande posible, para que la diferenciación entre perfiles sea notable.

  Para tratar de lograrlo, existe un grán número de algoritmos cuya elección se basa en las siguientes tres preguntas que dan respuesta a los parámetros necesarios para implementar uno de ellos:

  1. **Algoritmo**

     ¿El usuario conoce de antemano el número de grupos subyacentes?
     En función de la respuesta, el algoritmo será *particional* o *jerárquico.*

  2. **Distancia**

     ¿Cómo se va a medir la distancia espacial entre jugadores?
     A día de hoy, existe un grán número de fórmulas que determinan la distancia espacial entre dos puntos. Las más destacables son la *Distancia euclídea* y la *Distancia Manhattan*[23].

  3. **Método**

     ¿Que procedimiento se va a seguir a la hora de añadir jugadores a un cluster?
     Como se explicó previamente, el algoritmo itera sobre los registros del dataset, y los añade al grupo más cercano, por tanto los grupos serán flexibles y evolucionarán en cada iteración. Este parámetro modela el comportamiento del cluster a medida que evoluciona. Los valores más habituales son: *"average", "single", "complete" y "ward"*[24].

  Sin embargo, ha surgido una nueva vertiente denonimanda clustering *híbrido* [25], que combina las dos vertientes existentes a la hora de escoger el tipo de algoritmo. Dicha

vertiente comienza dividiendo el conjunto de datos en *clústeres* utilizando el algoritmo k-means (particional) para, posteriormente, aplicar un enfoque jerárquico usando el algoritmo k-means nuevamente en cada clúster obtenido que da como resultado una estructura jerárquica de clústeres. El punto fuerte de esta vertiente se basa en la combinación las fortalezas de ambos y mejora la precisión y robustez del clustering.

Por tanto, a partir de esta explicación se justifica la decisión de utilizar el clustering *híbrido* en este proyecto ya que es el que mejor resultado ha obtenido en comparación con el resto.

Respecto a los parámetros de distancia y método escogidos que mejor resultado producen, su justificación se basa en la función *comparacion_metodos_cluster()* que tiene como *inputs* una lista con los posibles métodos, una lista con las diferentes métricas de similitud y un dataset. Con esos parámetros, realiza todas las combinaciones posibles de metodo-distancia y devuelve el coeficiente de agrupamiento para cada combinación.

Dicho coeficiente se genera en los clusters jerárquicos y se utiliza para evaluar la calidad de los clústeres generados por el algoritmo en función de la cohesión interna de los clústeres [26]. Valores más altos del coeficiente indican una mayor cohesión y homogeneidad dentro de los clústeres, lo que sugiere una mejor separación de los grupos.

En consecuencia, a partir del *output* de la función *comparacion_metodos_cluster()* que se muestra en la figura 2.19, los méjores parámetros metodo-distancia serán aquellos que obtengan el mayor coeficiente de agrupamiento.

En resumen, los parametros utilizados son:

1. **Algoritmo**

   Clustering híbrido, mediante la funcion *HKmeans()* que combina clustering jerárquico y particional.

2. **Distancia**

   Distancia Manhattan: Se calcula sumando las diferencias absolutas entre las coordenadas de dos puntos en un espacio de características. A nivel espacial, la distancia representa la distancia recorrida en términos de bloques o segmentos de calles. Solo permite movimientos en dirección vertical u horizontal dentro del plano N dimensional, no en diagonal.

```
> comparacion_metodos_cluster(metodos,metricas,datos_jug_reducidos_Zscore)
Resultado para cluster_average_euclidean_agnes :
Valor de ac: 0.9346887

Resultado para cluster_average_manhattan_agnes :
Valor de ac: 0.9375492

Resultado para cluster_single_euclidean_agnes :
Valor de ac: 0.8771022

Resultado para cluster_single_manhattan_agnes :
Valor de ac: 0.8929794

Resultado para cluster_complete_euclidean_agnes :
Valor de ac: 0.9556736

Resultado para cluster_complete_manhattan_agnes :
Valor de ac: 0.9594613

Resultado para cluster_ward_euclidean_agnes :
Valor de ac: 0.9891781

Resultado para cluster_ward_manhattan_agnes :
Valor de ac: 0.9887136
```

Figura 2.19: Coeficientes de aglomeración de cada cluster generado a partir de combinación distancia-método distinta

Su fórmula matemática para 2D, donde P = (x_1, y_1) y Q = (x_2, y_2), es la siguiente:

$$D_{\text{Manhattan}}(P, Q) = |x_2 - x_1| + |y_2 - y_1|$$

3. **Método**

   Método de Ward: Trata de minimizar la varianza total dentro de cada *clúster* al fusionar clústeres en cada iteración. En cada una de ellas, se seleccionan los dos *clústeres* candidatos a ser fusionados en base a que la varianza total dentro del nuevo *clúster* sea mínima. Esta medida de similitud entre clusters se conoce como la suma de cuadrados de las diferencias (SSD), cuya fórmula matemática es la siguiente:

$$\text{SSD} = \sum_{i=1}^{n} \sum_{j=1}^{m} (x_{ij} - \bar{x}_j)^2$$

   En la fórmula de la Suma de Cuadrados de las Diferencias (SSD), los parámetros representan lo siguiente:

   - n: El número de objetos o elementos en el clúster.
   - m: El número de variables.
   - x_{ij}: El valor del objeto i en la variable j.
   - $\bar{x}_j$: La media de los valores en la variable j en el clúster.

- **Número óptimo de clusters**

  Como se mencionó previamente, en el clustering híbrido se mezclan tanto parte del clustering particional como del divisivo, por lo que existe una confrontación a la hora de decidir el número óptimo de clusters, ya que, como se explicó previamente, los procedimientos a seguir son diferentes en ambas vertientes. Por un lado, para la parte de clustering particional se conoce a priori los grupos subyacentes, que en este caso, se podría identificar un grupo por cada subconjunto de posiciones existente dentro del almacén, pero por el otro lado, en el clustering divisivo se desconoce el número de grupos óptimo. Por tanto, se ha decidido utilizar las siguientes técnicas relativas a la selección óptima del número de clusters [27] a pesar de tener una ligera idea del número de grupos que podría existir dentro del dataset:

  - **Dendograma**

    Esta técnica es utilizada en el clustering aglomerativo, bajo la hipótesis de que se desconoce el número de grupos subyacente en los datos. Mediante la librería

fviz_dend() se genera el dendograma de la figura 2.20, en el que se puede observar como se segmenta de manera precisa 2 grande grupos cuya diferencia radica en si el jugador es defensa o atacante. A continuación, se muestra el perfil de jugador predominante asociado a cada uno de los colores mostrado en la figura 2.20, en base a los jugadores que conforman su respectivo grupo. Dicho perfil se ha validado con ayuda del director empresarial.

* **Rojo:** Defensa Central
* **Amarillo:** Defensa lateral
* **Verde:** Mediocentro defensivo
* **Azul claro:** Extremos
* **Azul oscuro:** Mediocentros ofensivos
* **Rosa:** Delanteros.

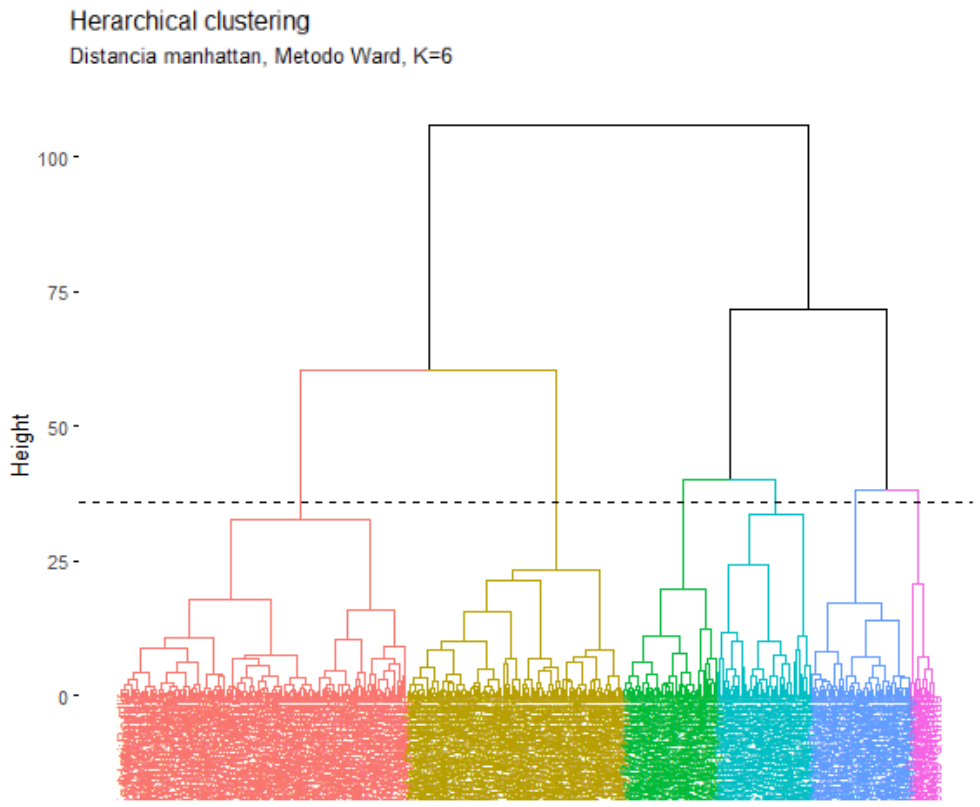


Figura 2.20: Dendograma generado con los 6 clusters procedentes del almacén de datos

– **Método del codo**

Esta es una técnica utilizada para determinar el número óptimo de clústeres tratando de identificar el punto de inflexión en un gráfico que muestra la suma de las

distancias al cuadrado de los puntos dentro de cada clúster respecto al número de clústeres.

La idea detrás de la prueba del codo es que a medida que aumenta el número de clústeres, la inercia disminuye. Sin embargo, llega un punto en el que agregar más clústeres no mejora significativamente la compacidad de los clústeres, y la inercia deja de disminuir rápidamente. Este punto de inflexión en el gráfico se considera un buen candidato para el número óptimo de clústeres.

En este caso, se puede ver en la figura 2.21 que el número óptimo serían cuatro grupos.

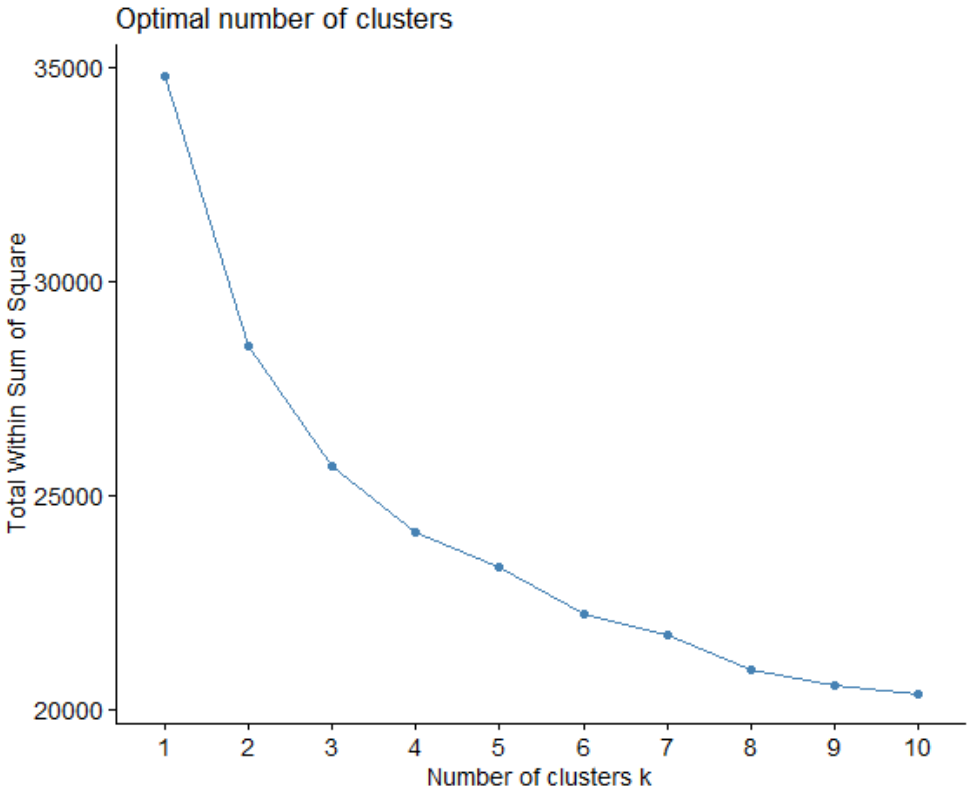


Figura 2.21: Método del codo para la selección del número óptimo de clusters

– **Métodos estadísticos**

Mediante la libreria *fviz_nbclust()* se pueden aplicar diferentes métodos estadísticos según los parámetros que se especifiquen dentro de la función. En este caso se ha utilizado el método *"gap_stat"*, que genera el gráfico de la figura 2.22 donde se muestra los valores de gap para diferentes valores de k (número de clústeres). La interpretación se basa en buscar el punto donde el valor de gap alcanza un máximo. Este punto representa el número óptimo de clústeres para nuestro conjunto

de datos, ya que la distancia observada entre los clústeres es mayor que la esperada al azar, lo cuál, sugiere una estructura de agrupamiento significativa. A medida que se aumenta el número de clústeres, se espera que el valor de gap aumente inicialmente, pero luego se estabilice o disminuya. Dicha prueba también concluye que el número óptimo de clusters es tres.

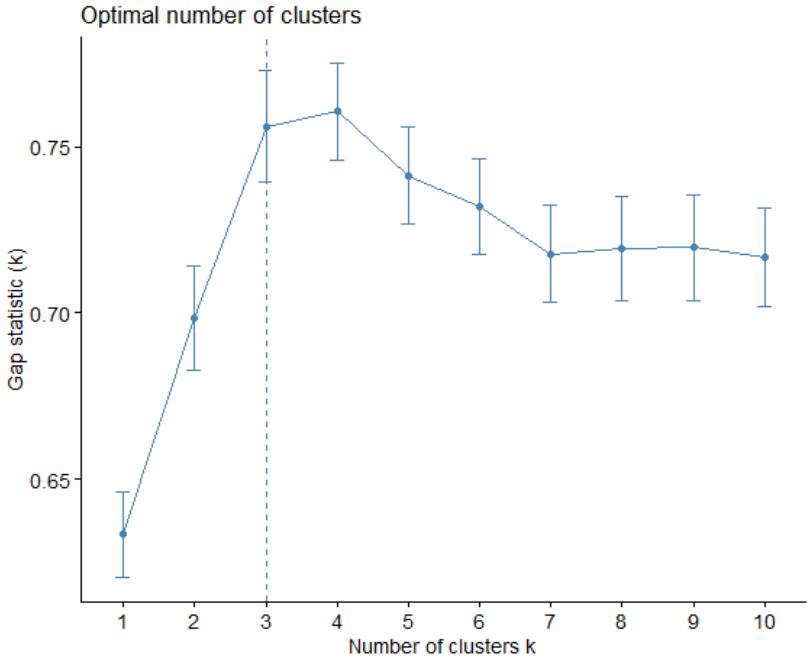


Figura 2.22: Método estadístico del máximo gap para la selección del número de clusters

Una vez mostrados los resultados de estas técnicas, se puede confirmar que el número óptimo de clusters será un número bajo. Con ayuda del director empresarial, se llegó a la conlcusión de que el número más adecuado de clusters era seis, ya que era el número que realizaba la segmentación más precisa de las diferentes posiciones dentro del campo, generando grupos coherentes y con una sencilla interpretación.

– **Resultados obtenido**

Para generar los clusters se ha utilizado la función *hkmeans()* a la que se le introduce como *input* los datos, el tipo de distancia, el tipo de método de agrupamiento y el número de clusters. A partir del *output* que genera, en la figura 2.23 se pueden visualizar los clusters en una representación interactiva 3D generada con la herramienta web *plotly*[2] o en 2D mediante la librería *fviz_cluster()* como se muestra en

[2] Enlace al gráfico: https://chart-studio.plotly.com/~arodriguezperez85/6

la figura 2.24, donde a cada registro se le asigna un identificador, el eje X se define como la primera componente principal y el eje Y como la segunda componente principal (cuya interpretación se encuentra en la sección 2.5.1)

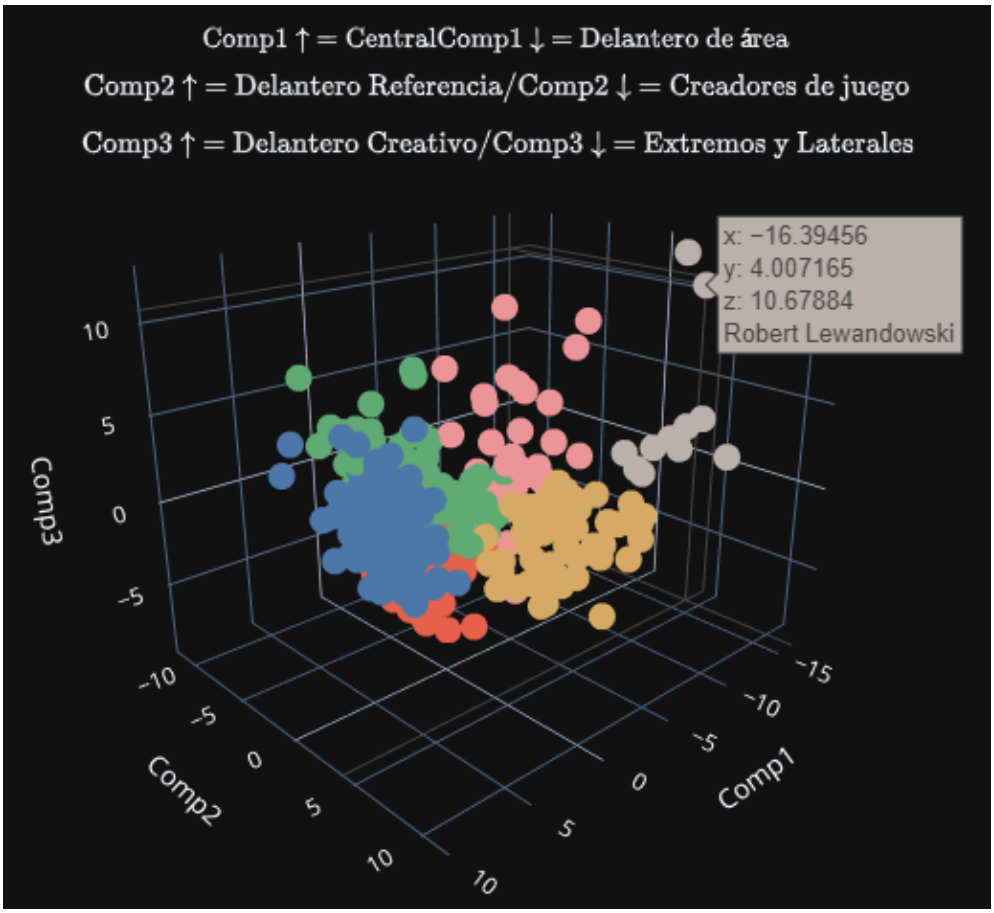


Figura 2.23: Representación gráfica 3D sobre las tres primeras componentes principales de los seis clusters generados con hkmeans() y los parámetros de la sección 2.5.2

Gracias al conocimiento sobre cada una de las componentes se puede obtener una primera aproximación del significado de cada cluster. Sin embargo, puede ser útil, analizar los clusters en gráficos donde los ejes X e Y no sean las dos primeras componenes principales (ya que puede haber clusters formados por jugadores, para los cuáles, dichas componentes son irrelevantes), por ello, se desarrolló la función *generar_graficos_clusters()* a la que se le introduce, los datos a procesar, el límite superior de componentes que el usuario desea utilizar y el número de clusters a generar.

Con todo esto, la función realiza todas las combinaciones posibles dos a dos, desde

la primera componente hasta el límite superior establecido y se genera un gráfico con el número de clusters determinado cuyos ejes es una de las combinaciones dentro del conjunto de todas ellas. En la figura 2.25 se puede ver cómo se comportan los seis clusters, donde el eje X representa la segunda componente y el eje Y la tercera componente.

Para dar un paso más allá en la validación de los clusters y la extracción de la información que subyace dentro de cada cluster de la figura 2.24 y conocer el perfil de jugador asociado a dicho cluster se desarrollaron dos funciones.

La función *tabla_cluster()* muestra una tabla donde cada columna representa un cluster y cada fila representa una valor de la variable artificial *"posicion_general´´* (explicada en la sección 2.2.3), de este modo se puede conocer la posición predominante de los jugadores que componen cada cluster como se muestra en la figura 2.26 y ,por ende, el perfil de jugador asociado al cluster.

La segunda función, *jugadores_dentro_cluster()* devuelve los nombres de los jugadores que contiene cada uno de los clusters. De este modo, si hay algún cluster que nos llama la atención porque tiene un número excesivamente bajo de jugadores o porque en la figura 2.26 se detecta que un cluster contiene un perfil de jugador atípico para dicho clusters, se puede comprobar de que jugador se trata, para poder tratar de buscar una justificación a ese resultado. En la figura 2.27 se muestra el output generado por la función (se ha reducido el número de registros que devuelve por motivos de espacio en la memoria).

En base a los resultados obtenidos por estas dos funciones, se ha asignado los siguientes perfiles de jugador a cada uno de los clusters:

1. ***Cluster 1***: Defensas Centrales
2. ***Cluster 2***: Laterales
3. ***Cluster 3***: Mediocentros defensivos y laterales con elevado perfil defensivo
4. ***Cluster 4***: Extremos
5. ***Cluster 5***: Mediocentros ofensivos de perfil creativo con el balon y con mucho contacto de balón
6. ***Cluster 6***: Delanteros

- **Objetivos conseguidos:**

A continuación, se enumeran los objetivos conseguidos en esta sección:

1. Selección óptima de parámetros para la creación de clusters.
2. Creación y visualización de clusters respecto a cualquiera de las componentes principales de la sección 2.5.1
3. Interpretación y validación de los clusters generados.

2.5.3 Cálculo de similitud entre jugadores

En esta sección se procederá a explicar el procedimiento seguido para predecir a qué cluster pertenece un jugador externo al almacén de datos original y cuál es el ranking de jugadores más similares en base a la similitud coseno.

- **Explicación de similitud coseno**

  En la sección 2.5.2 se mencionó que las dos medidas espaciales más conocidas para cuantificar la distancia entre dos jugadores eran la *euclídea* y la *Manhattan*. Estas medidas, puede que no sean la mejor opción a la hora de comparar un jugador con otro, ya que, si existen dos jugadores con registros muy similares para las variables más relevantes del almacén, estas medidas van a determinar que ambos jugadores son muy similares. Sin embargo, por naturaleza en el mundo del fútbol, puede que haya jugadores con registros similares pero que su desempeño en el campo sea muy diferente.
  En el artículo [28] se explica la medida de distancia utilizada en este trabajo y,con el siguiente ejemplo, se trata de facilitar la comprensión de este concepto.

  Explicación de la similitud coseno mediante un caso hipotético:

  En un contexto imaginario, se quiere comparar la similitud de tres jugadores de talla mundial entre sí: Lionel Messi, Cristiano Ronaldo y Joao Félix. Al tratarse de jugadores de impacto mundial, es fácil conocer que perfil se parece más entre sí.En este caso, Lionel Messi posee un perfil de juego dentro del campo más similar a Joao Felix, sin embargo, los registros de Joao Felix para determinadas variables relevantes como por ejemplo: goles,remates a puerta, acciones de ataque exitosas... son muy pobres en comparación a las de Cristiano Ronaldo, que tiene valores muy similares a los del Lionel Messi pero su perfil dentro del campo difiere bastante con el de Lionel Messi.

  En este caso,como se muestra en la figura 2.28, la distancia *euclídea* va a determinar que la distancia entre Lionel Messi con Cristiano Ronaldo es menor y con Joao Felix es mayor.

  Aquí es cuando la similitud coseno entra en juego, ya que no se basa en la comparación de variables, sino que define la similitud de dos jugadores como la diferencia entre los

ángulos de los vectores que definen a cada uno de ellos. En este caso, como se muestra en la figura 2.29, aunque Joao Felix posee peores registros para las variables del almacén, el ángulo de apertura de su vector con el de Lionel Messi es menor que el de Cristiano Ronaldo, que posee valores muy similares pero con un perfil dentro del campo distinto. Por ello la similitud coseno, determinará, para Lionel Messi, una mayor similitud con Joao Felix que con Cristiano Ronaldo.

- **Aplicación de similitud coseno**

  Al comienzo de este proyecto, uno de los objetivos planteados era desarrollar herramientas que aportasen valor e información en la toma de decisiones para el R.C.Deportivo a la hora de realizar *scouting* de jugadores para futuras temporadas. Dentro del *scouting* se pueden dar las dos siguientes contextos:

  - **Debilidad dentro de la plantilla**

    A lo largo de la temporada, puede darse el caso de que ciertos jugadores no hayan cumplido las expectativas generadas a principio de temporada y por tanto la dirección deportiva tome la decisión de reforzar dichas posiciones para la próxima temporada.

    En este contexto, la dirección deportiva del R.C.Deportivo tendrá en mente un perfil de jugador concreto para substituir al actual jugador de la plantilla. En el mundo del fútbol, se utilizan jugadores de talla mundial como referencia para buscar un determinado perfil de jugador, ya que son jugadores extremadamente famosos, sobre los cuáles, se conoce a la perfección su estilo de juego y su desempeño dentro del campo. A causa de esto, resulta sencillo asociar un perfil de jugador a un jugador de élite mundial.

    A través de la función *similitud_depor_vs_BD()* el perfil de jugador seleccionado de manera subjetiva por la dirección deportiva, materializado en un jugador de talla mundial, se podrá extrapolar de manera objetiva dentro de un espacio de búsqueda acotado en función de variables como nacionalidad, edad, liga, etc donde los posibles candidatos resultantes sean una opción viable y realista para el R.C Deportivo de La Coruña. De este modo, será posible encontrar los jugadores con mayor porcentaje de similitud respecto al jugador de referencia establecido, y que a su vez, sean candidatos viables económicamente para ser fichados.

  - **Intento de fichaje frustrado**

Otra situación en la que resulta útil la aplicación de la función *similitud_depor_vs_BD()* es para aquellos casos en los que el club desee incorporar un jugador concreto pero dicho jugador por cualquier motivo, externo al club, acabe siendo imposible de incorporar. En este caso, a través de esta función será posible devolver un ranking de jugadores similares a dicho candidato inicial, ya que las condiciones del espacio de búsqueda se pueden configurar para que los candidatos sean similares a dicho jugador inical. A partir del ranking generado se podrá obtener un listado de jugadores similares y así poder comenzar a trabajar en la incorporación de jugadores similares.

- **Oportunidad de mercado**

  Dentro del *scouting* deportivo existe otra situación en la que es posible incorporar nuevos jugadores. A lo largo de la temporada, hay muchos jugadores descontentos con el club en el que militan ya sea por falta de minutos, desafinidad con el cuerpo técnico, inqiuetud por probar nuevas aventuras... y que una vez finalizada la temporada, deciden ofrecerse a nuevos equipos para tratar de fichar por uno de ellos.

  Para este caso, se ha desarrollado la función *similitud_BD_vs_depor()*, cuyo funcionamiento es muy similar a la función *similitud_depor_vs_BD()*, pero el contexto es el inverso. En este caso, a partir de un jugador que no milita en las filas del R.C Deportivo de La Coruña , la función *similitud_BD_vs_depor()*, devuelve el ranking de similitud con los jugadores que componen la plantilla del R.C Deportivo de La Coruña, así como sus proyecciones para las tres primeras componentes principales.

  De este modo, en caso de darse esta situación, la dirección deportiva del R.C Deportivo de La Coruña, podrá comparar el perfil del jugador candidato a ser incorporado, con los jugadores que militan en la plantilla del club, y así, contar con una fuente de información extra a la hora de tomar una decisión estratégica.

- **Ejemplo de uso**

  A continuación se muestran un ejemplo de uso para el caso de la función *similitud_depor_vs_BD()* y de la función *similitud_BD_vs_depor():*

  - ***similitud_depor_vs_BD()*:**

    En la figura 2.30 se muestra un ejemplo de uso para mostrar visualmente la capacidad de esta función.

En este caso, se supone que la dirección deportiva no está contenta con el rendimiento de su delantero centro ”Arturo Juan Rodríguez Pérez-Reverte” y desean sustituirlo por otro delantero con un perfil asociativo, goleador y participativo en la creación de juego. Después de discutirlo, llegan a la conclusión de que el famoso jugador del F.C.Barcelona “Robert Lewandowski´´, es el jugador que mejor reúne los requisitos exigidos. Tanto el F.C.Barcelona (Primera División) como el R.C Deportivo de La Coruña (Primera División RFEF) se encuentran en diferentes categorías dentro del fútbol español, por lo que la dirección deportiva decide acotar el espacio de búsqueda para encontrar el ranking de jugadores más similares a “Robert Lewandowski´´ a los jugadores de su misma división.

Cabe destacar que para realizar el ranking de jugadores más similares, previamente se realiza una clusterización dentro del espacio de búsqueda establecido siguiendo la metolodogía utilizada en la sección 2.5.2, para así lograr una primera segmentación de los candidatos por jugadores en base a su posición dentro del campo.
Una vez realizados los *clusters*, la función *similitud_depor_vs_BD()* descarta todos los candidatos, excepto aquellos que compongan el *cluster* más próximo a “Robert Lewandowski´´. En dicho cluster, la posición de jugador determinante dentro del cluster seguramente sea la delantero centro, lo cuál, es un requisito fundamental a la hora de buscar jugadores similares. Gracias a esto se reduce aún más el espacio de busqueda, reduciendo el coste computacional de la función y aumentando la precisión en los resultados.

En la figura 2.30, se muestra el *output* de la función *similitud_depor_vs_BD()* para esta situación, que consta de:

1. Ranking de los N jugadores más similares a “Robert Lewandowski´´ en función de la similitud coseno, con el espacio de búsqueda acotado a los jugadores que compiten en la división ”Primera División RFEF”
2. Proyección en las tres primeras componentes principales de la sección 2.5.1 para los jugadores del ranking y del jugador de referencia.
3. Asignación del jugador al cluster más próximo utilizando la similitud coseno, siguiendo la metodología de la sección 2.5.2.

– ***similitud_BD_vs_depor()*:**
Supongamos que el jugador “Mario Soberón Gutiérrez´´, delantero centro del CD Eldense, equipo que compite en Primera Division RFEF (la misma competición que

el R.C.Deportivo), decide que quiere cambiar de equipo y a la dirección deportiva del R.C.Deportivo, le llega una oferta de dicho jugador con intención de incorporarse a la plantilla del cuadro gallego para la temporada que viene.
A la hora de estudiar si sería una buena incorporación o no, es importante tener en cuenta con qué jugadores de la actual plantilla se va a disputar el puesto en el campo dicho jugador, ya que si son jugadores con un buen rendimiento deportivo, el club puede que no esté interesado en incorporar a "Mario Soberón Gutiérrez´´.

En la figura 2.31, se muestra el output de la función *similitud_BD_vs_depor()* para este caso. Como se puede apreciar, en este caso los dos jugadores más similares son "Alberto Quiles Piosa´´ y "Lucas Pérez Martínez´´, los cuáles, son piezas clave dentro del equipo, por lo que seguramente la dirección deportiva no este interesada en incorporar a este jugador.

- **Objetivos conseguidos**

  A continuación se enumeran los objetivos logrados en esta sección:

  1. Aplicación de la similitud coseno para obtener el N-ranking de jugadores más similares a un valor de referencia dado, para facilitar la captación de jugadores.
  2. Aplicación de las secciones 2.5.1 y 2.5.2 a dichos candidatos, para extraer información de valor sobre ellos.

2.5.4 *Machine learning* aplicado al *scouting*

En esta sección se procederá a explicar la metodología seguida a la hora de aplicar *machine learning* en el *scouting* deportivo, con el objetivo de obtener estimaciones precisas en aquellas variables de interés para la dirección deportiva sobre los posibles candidatos a ser incorporados a la plantilla, es decir, con el objetivo de aplicar regresión.

La regresión es una técnica que busca modelar la relación entre variables de entrada y una variable de salida continua. Utilizando algoritmos de *machine learning*, el modelo de regresión puede aprender patrones y tendencias en los datos para hacer predicciones precisas sobre valores numéricos desconocidos.
El uso de estas técnicas permite un enfoque flexible y potente a la hora de realizar las predicciones, gracias a su capacidad para manejar relaciones no lineales y complejas entre las variables[29].

- **Problemática en el *scouting* deportivo:**

Uno de los principales problemas identificados por la dirección deportiva del club a la hora de realizar una incorporación de un jugador que compite en ligas extranjeras a la española, es el desconocimiento e incertidumbre sobre cuál será su rendimiento en la respectiva liga en la que compita el R.C Deportivo de La Coruña.

El conjunto de ligas de fútbol a nivel global es enorme, y a causa de ello, existe una gran heterogeneidad en cuánto a recursos económicos, exigencia deportiva, calidad futbolística... en cada una de las ligas. Por ello, a pesar de que un jugador de fútbol posea unos valores extraordinarios dentro del almacén de datos, no significa que dicho jugador sea un fichaje de éxito asegurado, ya que puede que el nivel de exigencia en la competición de dicho jugador diste batsante de la exigencia de la competición en la que reside el R.C Deportivo de La Coruña.

Por ello, el enfoque en este proyecto respecto a esta sección, es entrenar un modelo de *machine learning* con los datos procedentes de los jugadores que militan en la misma competición que el R.C Deportivo de La Coruña, para que dicho modelo sirva como herramienta para predecir el desempeño que va a tener un jugador extranjero en la propia competición del R.C Deportivo de La Coruña, ya que ha sido entrenado con datos propios de la competición y, por ende, los hiperparámetros del modelo estarán ajustados a esa competición.

Una vez entrenado el modelo, simplemente se introducirá como *input*, las respectivas variables de entrenamiento del candidato a ser fichado con los datos que ha generado en su liga, y el modelo devolverá la predicción para la variable de interés que dicho jugador obtendría en caso de jugar en la misma competición que el R.C Deportivo de La Coruña.

- **Aspectos a tener en cuenta en *machine learning***

  A la hora de entrenar un modelo de *machine learning* para tratar de resolver un problema ya sea de regresión o de clasificación, es aconsejable tener en cuenta los dos siguientes aspectos:

  - **Selección de características[30]**

    En escenarios donde el conjunto de características es extenso, es fundamental identificar y utilizar en la fase de entrenamiento del modelo solo aquellas características relevantes y significativas que guardan una estrecha relación con la variable a predecir.

    De este modo, se eliminan características irrelevantes y ruidosas, se mejora la comprensión del problema y sus factores influyentes, y ayuda a optimizar el ren-

dimiento computacional. Respecto a la predicciones, se mejora la precisión e interpretación del modelo utilizado, lo que lleva a mejores resultados y una toma de decisiones más sólida en el ámbito de la regresión.

- **Validación cruzada[31]**

  Esta técnica es esencial para evaluar de manera objetiva el rendimiento y la capacidad predictiva de un modelo en datos no vistos previamente.

  Cuando se trata de regresión, el objetivo principal a la hora de entrenar el modelo es predecir valores numéricos continuos tratando de reducir al mínimo el error entre el valor predicho y el valor real. Sin embargo, la verdadera medida del rendimiento de un modelo radica en su capacidad para generalizar y predecir con precisión en un contexto fuera del conjunto de entrenamiento.

  Aquí es donde entra en juego la validación cruzada. En lugar de depender únicamente de una única partición de datos para entrenar y evaluar el modelo, la validación cruzada divide el conjunto de datos en múltiples subconjuntos. Luego, se realiza un proceso iterativo en el que cada subconjunto se utiliza como conjunto de prueba mientras el resto se utiliza como conjunto de entrenamiento. Esto permite evaluar cómo se comporta el modelo en diferentes escenarios y proporciona una medida más robusta de su desempeño general.

  La importancia de la validación cruzada radica en su capacidad para detectar problemas como el sobreajuste o el subajuste. El sobreajuste ocurre cuando un modelo se adapta demasiado a los datos de entrenamiento, lo que puede llevar a un rendimiento deficiente en nuevos datos. Por otro lado, el subajuste ocurre cuando un modelo no es lo suficientemente flexible para capturar las relaciones complejas en los datos. La detección de uno de estos dos fenómenos permite un mejor ajuste de los hiperparámetros del modelo, y así, mejorar su capacidad de generalización y obtener resultados más precisos en situaciones reales.

- **Modelo de *machine learning* utilizado:**

  Dentro del mundo del *machine learning* existe una gran variedad a la hora de decidir cuál es el mejor modelo a utilizar. Para tomar esta decisión se ha de tener en cuenta el tipo de problema a resolver, la naturaleza de los datos utilizados o si es necesario ser capaz de justificar el modelo y sus predicciones (en caso de utilizar redes neuronales, sería muy complejo darle una explicación a los resultados obtenidos).

  En este proyecto se ha decidio utilizar uno de los métodos más populares en el ámbito del *machine learning*, los árboles de decisión, en concreto los *Random Forest*[32].

Son un método de aprendizaje automático fácil de interpretar, que combina múltiples árboles de decisión para realizar predicciones precisas. Cada árbol se construye de forma independiente utilizando diferentes subconjuntos aleatorios de los datos de entrenamiento y variables predictoras y se divide repetidamente en función de las características de los datos para formar ramas y hojas, buscando reducir de la impureza o el aumento de la pureza (Indice de Gini, entropía, error de clasificación...).

Una vez construido el bosque de árboles, la predicción se realiza siguiendo un enfoque *ensemble* como se muestra en la figura 2.32, que consiste en promediar las predicciones realizadas por cada uno de los árboles. Este enfoque aprovecha la fuerza de múltiples árboles de decisión para ofrecer predicciones precisas y versátiles y reducir la varianza y el sobreajuste, ya que los árboles se benefician de la diversidad y la combinación de múltiples modelos.

- **Aplicación del modelo de *machine learning***

  En la tabla 2.3 se muestran algunas de las variables para las distintas áreas que contiene el almacén de datos proporcionado por el R.C Deportivo de La Coruña. Para realizar regresión, las variables relacionadas con el área deportiva suelen ser las variables de interés a predecir. Dichas variables poseen una granuralidad temporal a nivel de temporada completa, es decir, que el valor que toma una variable para un determinado jugador, es el valor promedio obtenido a lo largo de todos los partidos que ha disputado durante la temporada.

  Por tanto, los valores resultantes después de realizar la regresión serán los valores promedios esperados para la temporada, y en ningún caso, valores esperados para un partido concreto.

- **Caso de uso**

  En un hipotético caso de que la dirección deportiva haya validado los resultados obtenidos para el jugador de la *Premier League*, "Cody Gakpo", en la sección 2.5.3 y sea una opción real y viable para ser fichado, el último paso consiste en predecir cómo se va a comportar dicho jugador,en la competición en la que se encuentra el R.C Deportivo de La Coruña, en base a las estadísticas que ha obtenido a lo largo de la temporada en su respectiva competición. En este caso, la dirección deportiva está interesada en conocer cuál será la expectativa de tiros para el jugador.

En la figura 2.33, se muestra el *outuput* de la función *prediccion_variable()*, en la que mediante la librería *rfe()* se entrena un modelo de *random forest* utilizando tanto selección de características como validación cruzada, y devuelve la predicción para la variable de interés (*"stcurr_xg_shot"*) junto a las variables más importantes a la hora de realizar la predicción para dicha variable.

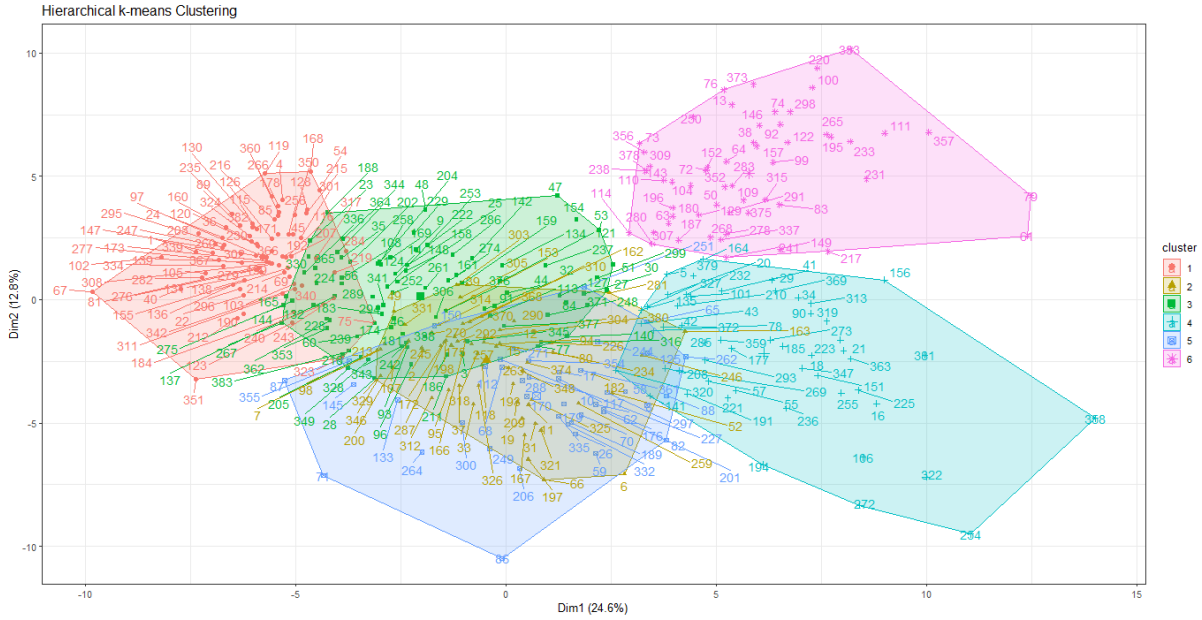


Figura 2.24: Representación gráfica 2D sobre las dos primeras componentes principales de los seis clusters generados con hkmeans() y los parámetros de la sección 2.5.2

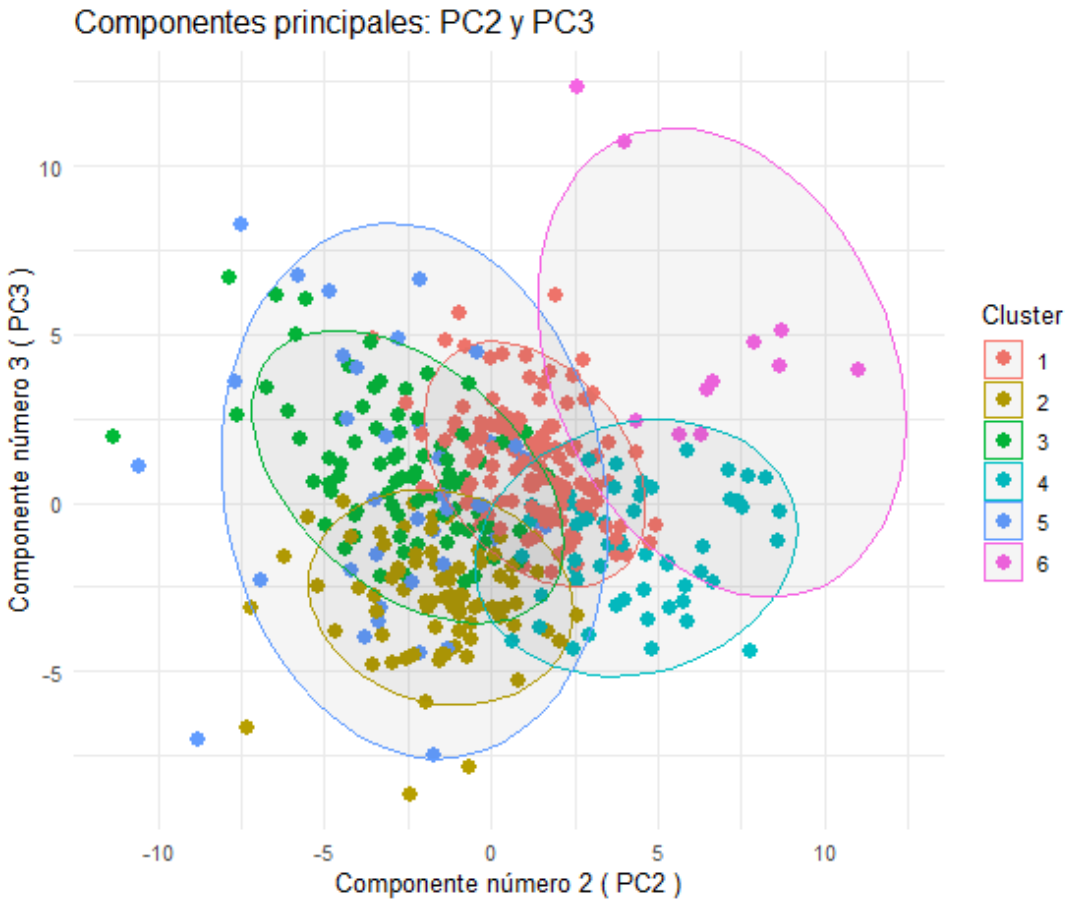


Figura 2.25: *Output* devuelto por la función *generar_graficos_clusters()* para los clusters de la figura 2.24 sobre la componentes dos y tres

	1	2	3	4	5	6
Defensa central	74	1	7	0	0	0
Defensa lateral	3	55	15	1	4	0
Delantero	0	0	3	11	0	50
Extremo	0	6	6	26	0	4
Interior	0	1	10	8	19	1
Mediocentro defensivo	3	1	46	2	11	0
Mediocentro ofensivo	0	0	2	4	8	1

Figura 2.26: Output de la función *tabla_cluster()* para los clusters de la figura 2.24

```
> head(jugadores_dentro_cluster(grupos),5)
                         cluster_1              cluster_2              cluster_3
1            Wassim Keddari Boullif Iván Fresneda Corraliza       Pablo Barrios Rivas
2          John Nwankwo Donald Okeh  Lautaro Emanuel Blanco    Pablo  Ibáñez Lumbreras
3                      Cenk Özkacar          Omar El Hilali      Ángel Algobia Esteves
4 Cristhian Andrey Mosquera Ibarguen       Raúl  Parra Artal       Kaiky Fernandes Melo
5                        Loïc Badé    Arnau Martínez López José Ángel Carmona Navarro
                  cluster_4              cluster_5                  cluster_6         cluster_7
1  Jon Magunazelaia Argoitia       Pablo Torre Carral      Daniel González Flores      Karim Benzema
2 Rodrigo  Sánchez Rodriguez  Pablo Martin Páez Gavira              El Bilal Touré Robert Lewandowski
3  Nicholas Williams Arthuer      Pedro González López      Lázaro Vinicius Marques  Iago Aspas Juncal
4      Abdessamad Ezzalzouli              Luka Modrić         Raúl García Escudero  Antoine Griezmann
5        Gabriel Veiga Novas David Josué Jiménez Silva Edinson Roberto Cavani Gómez   Viktor Tsygankov
```

Figura 2.27: Output de la función *jugadores_dentro_cluster()* para los clusters de la figura 2.24

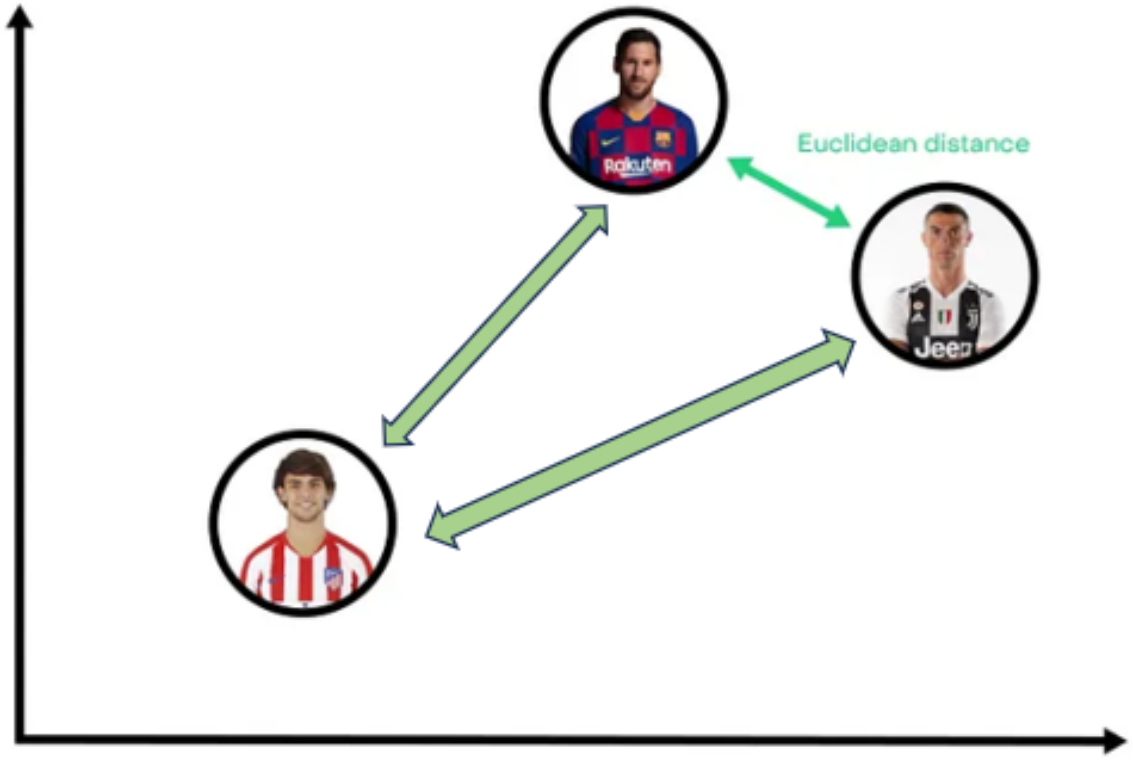


Figura 2.28: Distancia euclídea para el caso hipotético de la explicación de la similitud coseno (sección 2.5.3

)

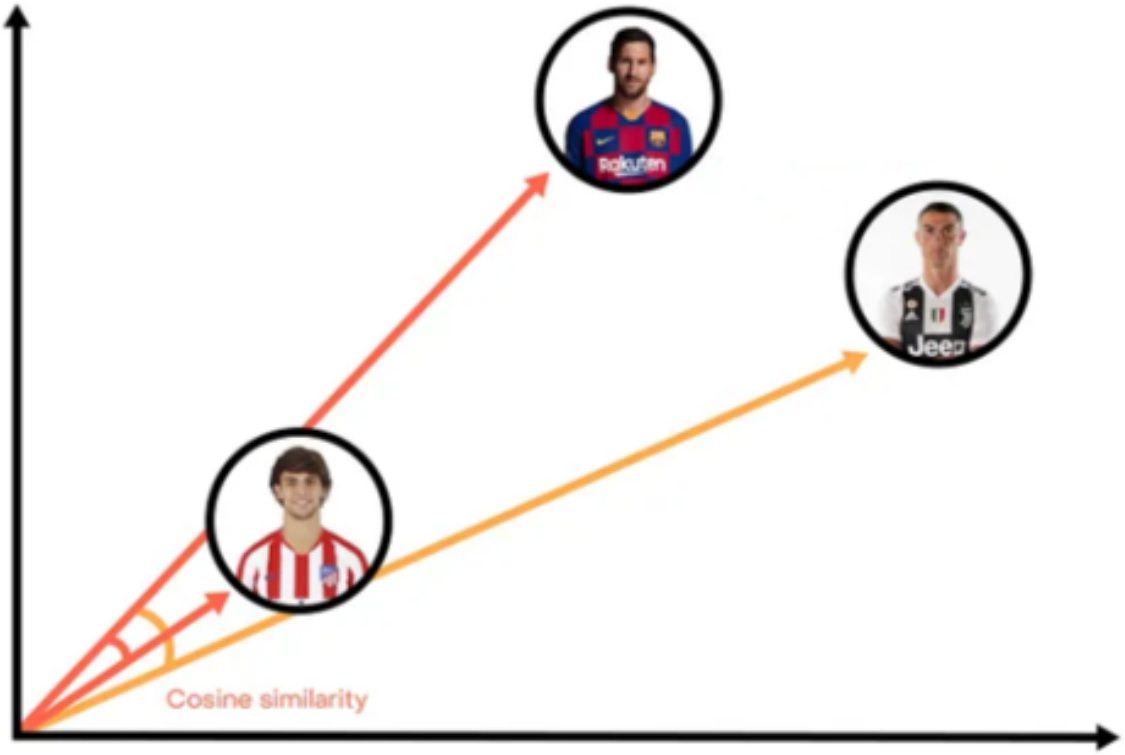
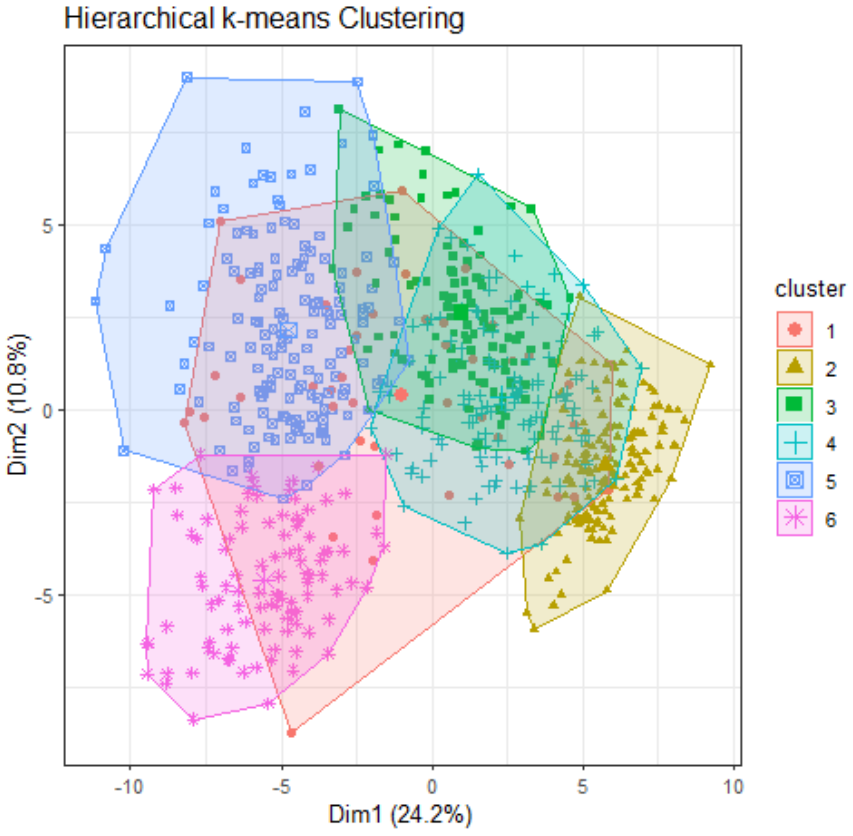


Figura 2.29: Similitud coseno para el caso hipotético de la explicación de la similitud coseno (sección 2.5.3

)

```
**********  % SIMILITUD VS Robert Lewandowski **********

Mario Soberón Gutiérrez al 76.2 %

Dionisio Emmanuel Villalba Rojano al 73.1 %

Christian Borrego Isabel al 72.2 %

Iván Martínez Gonzálvez al 70.1 %

Javier Martón Ansó al 66.1 %

**********      VALORES PCA     **********

                              Comp1 Comp2 Comp3
Mario Soberón Gutiérrez       -7.36  3.01  1.06
Dionisio Emmanuel Villalba Rojano -6.77  3.23  3.53
Christian Borrego Isabel      -7.02  3.42  1.64
Iván Martínez Gonzálvez       -8.18  6.38  3.20
Javier Martón Ansó            -7.62  5.69  2.58
                  Comp.1 Comp.2 Comp.3
Robert Lewandowski -13.75   4.71  10.05
El cluster asignado a Robert Lewandowski es: 6
```

(a) Output generado por la función



(b) Clusters generados por la función

Figura 2.30: Caso de uso de la función *similitud_depor_vs_BD()* explicado la sección 2.5.3

```
********************  % SIMILITUD VS Mario Soberón Gutiérrez ********************

Alberto Quiles Piosa al 63.5 %

Lucas Pérez Martínez al 56.1 %

Arturo Juan Rodríguez Pérez-Reverte al 45.4 %

Mario Soriano Carreño al 41.2 %

Max Svensson Río al 36.8 %

********************      VALORES PCA     ********************

                                  Comp1 Comp2 Comp3
Alberto Quiles Piosa               -7.40  2.80  6.26
Lucas Pérez Martínez               -8.20 -0.77  4.13
Arturo Juan Rodríguez Pérez-Reverte -5.06  3.52 -1.20
Mario Soriano Carreño              -3.95 -1.83  4.27
Max Svensson Río                   -5.10  2.96 -1.64
                       Comp.1 Comp.2 Comp.3
Mario Soberón Gutiérrez  -7.36   3.01   1.06
```

Figura 2.31: Caso de uso de la función *similitud_BD_vs_depor()* para la sección 2.5.3

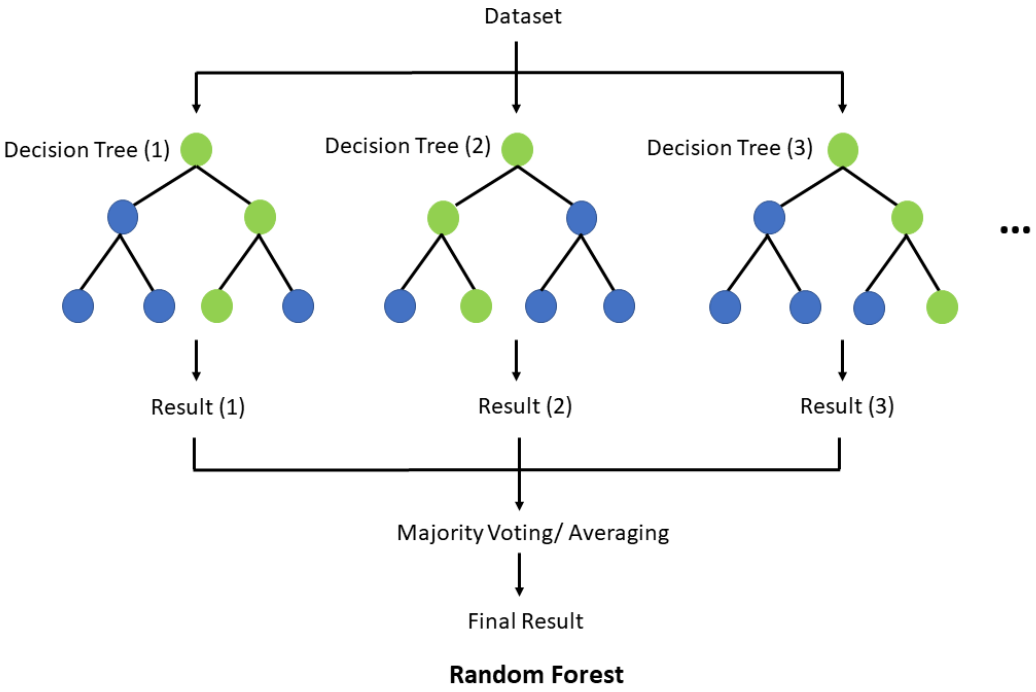


Figura 2.32: Estructura interna de un *random forest*

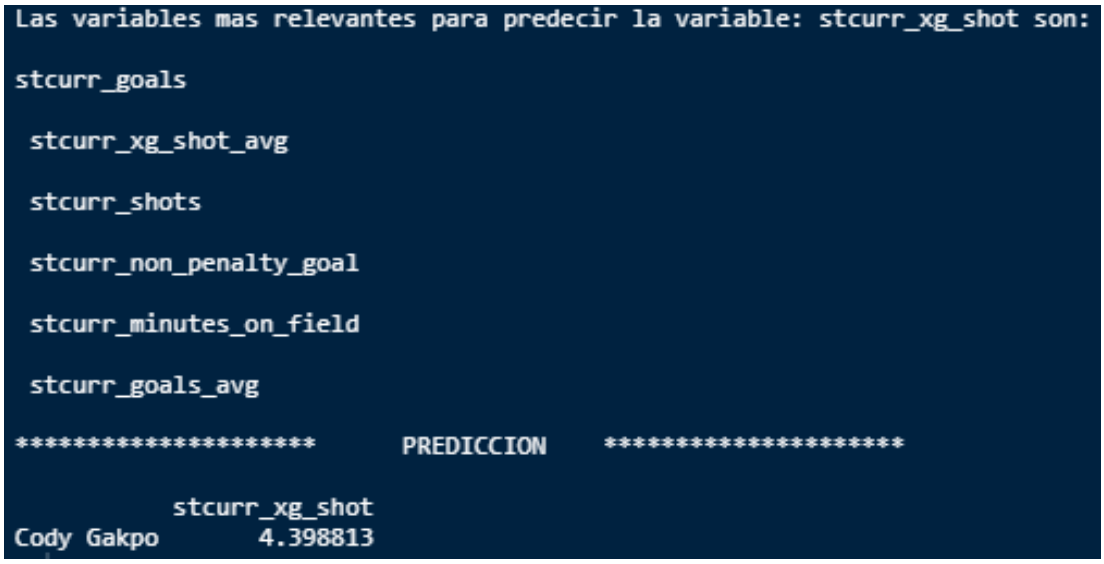


Figura 2.33: *Output* de la función *prediccion_variable()* para el jugador ”Cody Gakpo” y la variable *”stcurr_xg_shot”*

Capítulo 3

Conclusiones

Último capítulo de la memoria, donde se presentarán: la situación final del trabajo, las lecciones aprendidas, la relación con las competencias de la titulación en general y en particular, posibles líneas futuras, ...

3.1 Situación final del trabajo

A lo largo de esta memoria, se ha explicado el procedimiento, paso a paso, para lograr los objetivos planteados en la sección 2.1.1. A partir del almacén de datos proporcionado por el R.C Deportivo de La Coruña como único elemento de partida, se ha conseguido desarrollar una herramienta capaz de aplicar análisis estadístico y *machine learning* en el ámbito de la exploración de jugadores, aportando así, información relevante y de valor, para la dirección deportiva del club.

En pocas palabras, esta herramienta es capaz de conocer, a partir de los datos de un jugador desconocido para la dirección deportiva del club, de que perfil de jugador se trata (sección 2.5.2), cuáles son los jugadores más similares a dicho jugador dentro de un espacio de búsqueda acotado (sección 2.5.3), predecir cuál será el comportamiento de dicho jugador en ciertas variables de interés para una liga diferente a la suya (sección 2.5.4) y por último, conocer cómo de segmentada esta su posición dentro del campo en función de los valores que obtenga en las componentes principales (sección 2.5.1),es decir, si el desempeño del jugador en su posición es más de caracter ofensivo,defensivo o de creación de juego.

Un punto a añadir, sería que todo el trabajo realizado a lo largo de este proyecto se puede adaptar a las condiciones que el usuario desee, es decir, en este caso, el almacén de datos que se utiliza contiene datos de jugadores que abarcan todas las posiciones del terreno de juego, sin embargo, si el usuario desea conocer los resultados que genera la herramineta para una posición concreta del campo, simplemente se necesita introducir un almacén de datos que

contenga jugadores para dicha posición concreta, y se generarán nuevos resultados adaptados a esta situación. Por ejemplo, si se quiere conocer los distintos perfiles de jugador que existen para la posición de ”Defensa central” (central contundente, central participativo en el juego, central goleador...), solamente se necesita introducir un almacén de datos con jugadores cuya posición sea la de ”Defensa central”.

3.2 Lecciones aprendidas

Durante el transcurso de este proyecto, se han aprendido las siguientes lecciones:

1. La importancia de los datos radica en su calidad y precisión, no en su cantidad.
2. Realizar un correcto preprocesado de los datos es fundamental a la hora de trabajar con algoritmos de aprendizaje automático, tanto a nivel de filtrado como de normalización.
3. Es necesario dedicar tiempo a la correcta interpretación de los resultados obtenidos a la hora de aplicar técnicas como análisis de componentes principales o clustering, ya que, de este modo, es mucho más sencillo validar los resultados obtenidos y confirmar la eficacia de las técnicas.
4. A la hora de aplicar *machine learning* para resolver un problema, los pasos de selección de características y validación cruzada en la fase de entrenamiento del modelo (sección 2.5.4) mejoran considerablemente la precisión de los resultados.

3.3 Relación con las competencias del grado

En la página web de la titulación de *Ciencia e Ingeniería de Datos* [33] se puede encontrar definidas cada una de las competencias del grado. A continuación se muestran las competencias relacionadas en cada una de las secciones de esta memoria:

- **Sección 2.1**:

Tabla 3.1: Competencias del título asociadas a la sección 2.1

Básicas	Específicas	Transversales
A16	*B2*	*C1*
A31	*B3*	*C3*

. . . . . *(continúa en la siguiente página)* . . . . . .

Tabla 3.1 – *(viene de la página anterior)*

Básicas	Específicas	Transversales
.	*B4*	.
.	*B9*	.

- **Sección 2.2**:

Tabla 3.2: Competencias del título asociadas a la sección 2.2

Básicas	Específicas	Transversales
A15	.	.
A27	.	.

- **Sección 2.3**:

Tabla 3.3: Competencias del título asociadas a la sección 2.3

Básicas	Específicas	Transversales
A18	.	.
A19	.	.

- **Sección 2.5**:

Tabla 3.4: Competencias del título asociadas a la sección 2.5

Básicas	Específicas	Transversales
A1	.	.
A2	.	.
A3	.	.

. . . . . *(continúa en la siguiente página)* . . . . . .

Tabla 3.4 – *(viene de la página anterior)*

Básicas	Específicas	Transversales
A4	.	.
A5	.	.
A6	.	.
A18	.	.
A26	.	.

- **Sección 3**:

Tabla 3.5: Competencias del título asociadas a la sección 3

Básicas	Específicas	Transversales
.	*B1*	.
.	*B2*	.
.	*B3*	.
.	*B4*	.
.	*B5*	.

3.4 Posibles futuras líneas de trabajo

En esta sección, se muestran a continuación las posibles líneas de trabajo que se pueden emprender a partir de este proyecto:

1. Aplicación de algoritmos propios del *deep learning* a la hora de realizar regresión.
2. Desarrollo de una aplicación web sencilla e interactiva que permita el uso de la herramienta desarrollada en este proyecto por parte personal no especializado.
3. Implementación de técnicas de procesamiento paralelo en los algoritmos utilizados, para poder escalar el procesamiento de datos a una escala mayor.

4. Uso de interpolación en la sección 2.5.3, para utilizar dos jugadores de referencia, en lugar de un único jugador, y así, poder encontrar jugadores que posean características similares a dos jugadores dados.

Lista de acrónimos

FIFA Federation International Football Association. 11

Bibliografía

[1] E. Ángulo, "El auge de la ingeniería y el análisis de datos en el deporte," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://esi.uclm.es/index.php/2022/06/25/el-auge-de-la-ingenieria-y-el-analisis-de-datos-en-el-deporte/

[2] "Nacho lourido, director del nuevo departamento de tecnología analítica y deportiva," 2021, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.rcdeportivo.es/noticias/nacho-lourido-director-del-nuevo-departamento-de-tecnologia-analitica-y-deportiva

[3] "Historia del fútbol," 2021, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.competize.com/blog/historia-futbol-resumen-origen-torneos-reglas/

[4] J. Vivas, "Analítica de datos en el fútbol," 2021, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.joakimvivas.com/tech/analitica-datos-futbol/

[5] E. Contreras, "Así ficha el sevilla f.c.: departamento i+d, data y algoritmo entre 18.000 futbolistas," 2020, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.marca.com/primera-plana/2020/02/08/5e3709e822601d8a5b8b458e.html

[6] J. Hurtado, "Cómo funciona la metodología scrum: Qué es y cómo utilizarla," consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.iebschool.com/blog/metodologia-scrum-agile-scrum/

[7] A. Riveros, "Que es una edt," 2020, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.ealde.es/que-es-edt-proyectos/

[8] "Sueldos para el puesto de ingeniero de datos en españa," 2023, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.glassdoor.es/Sueldos/ingeniero-de-datos-sueldo-SRCH_KO0,18.htm

[9] "Cuánto gana un profesor de universidad en españa," 2023, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.businessinsider.es/cuanto-gana-profesor-universidad-espana-1119435

[10] "Sueldos para el puesto de director de departamento de análisis de datos," 2023, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.glassdoor.com/Salaries/barcelona-head-of-data-salary-SRCH_IL.0,9_IM1015_KO10,22.htm

[11] "Wyscout data model glossary," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://dataglossary.wyscout.com/

[12] H. Suárez, "¿qué es una correlación? ... y herramientas de análisis de datos," 2015, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.incibe.es/incibe-cert/blog/que-es-una-correlacion-y-herramientas-de-analisis-de-datos

[13] J. López, "Desviación estándar o típica," 2020, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://economipedia.com/definiciones/desviacion-tipica.html

[14] F. Sanjuan, "Curtosis," 2020, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://economipedia.com/definiciones/curtosis.html

[15] "Asimetría (estadística)," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.probabilidadyestadistica.net/asimetria-estadistica/

[16] P. Rodó, "Rango intercuartílico," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://economipedia.com/definiciones/rango-intercuartilico.html

[17] "¿qué es un histograma?" 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.tibco.com/es/reference-center/what-is-a-histogram-chart

[18] D. Montes, "Diagrama boxplot," 2018, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.pgconocimiento.com/diagrama-boxplot/

[19] "Cómo estandarizar datos y por qué es importante," 2018, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.ayuware.es/blog/como-estandarizar-datos/

[20] K. Rojas, "Transformación, estandarización e imputación de datos," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://bookdown.org/keilor_rojas/CienciaDatos/transformación-estandarización-de-datos.html

[21] J. Amat Rodrigo, "Análisis de componentes principales (principal component analysis, pca) y t-sne," 2017, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.cienciadedatos.net/documentos/35_principal_component_analysis

[22] ——, "Clustering y heatmaps: aprendizaje no supervisado," 2017, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.cienciadedatos.net/documentos/37_clustering_y_heatmaps

[23] ——, "Medidas de distancia," 2017, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.cienciadedatos.net/documentos/37_clustering_y_heatmaps#Medidas_de_distancia

[24] ——, "Hierarchical clustering," 2017, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.cienciadedatos.net/documentos/37_clustering_y_heatmaps#Hierarchical_clustering

[25] ——, "Hierarchical k-means clustering," 2017, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.cienciadedatos.net/documentos/37_clustering_y_heatmaps#Hierarchical_K-means_clustering

[26] P. Araneda, "Cluster jerárquico," 2023, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://rpubs.com/paraneda/hclust

[27] J. Amat Rodrigo, "Número óptimo de clusters," 2017, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.cienciadedatos.net/documentos/37_clustering_y_heatmaps#Número_Óptimo_de_clusters

[28] B. Pimpaud, "Player similarities and interpolation - towards data science," 2021, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://towardsdatascience.com/player-similarities-interpolation-aecbf6423c72

[29] V. Román, "Machine learning supervisado: Fundamentos de la regresión lineal," 2019, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://medium.com/datos-y-ciencia/machine-learning-supervisado-fundamentos-de-la-regresión-lineal-bbcb07fe7fd

[30] "Qué es la selección de características en machine learning," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://keepcoding.io/blog/seleccion-de-caracteristicas-en-machine-learning/

[31] "Cross-validation : definición e importancia en machine learning," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://datascientest.com/es/cross-validation-definicion-e-importancia

[32] J. Amat Rodrigo, "Random forest con python," 2020, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://www.cienciadedatos.net/documentos/py08_random_forest_python

[33] "Competencias del grado de ciencia e ingeniería de datos," 2022, consultado el 22 de junio de 2023. [En línea]. Disponible en: https://guiadocente.udc.es/guia_docent/index.php?centre=614&ensenyament=614G02&consulta=competencies