



# UNIVERSIDAD TORCUATO DI TELLA

Cómo encontrar el mejor jugador para tu Equipo de Fútbol

Escuela de Negocios - Licenciatura en Tecnología Digital

Tomás Glauberman\*

Ignacio Pardo†

Juan Ignacio Silvestri‡

CABA, Argentina. Diciembre 2024

## Abstract

En la última década, el análisis deportivo ha evolucionado hacia una perspectiva cada vez más matemática y sofisticada. Aplicaciones como el uso de análisis espacial en Basketball (Goldsberry, 2012) y modelos de juegos de suma cero en fútbol (Hirotsu y Wright, 2006) son ejemplos claros de la tendencia creciente en este campo. El béisbol, por mucho tiempo el deporte preferido para la analítica, ha experimentado una profunda transformación con la implementación de Sabermetrics (Baumer y Zimbalist, 2014; Wolf, 2015). La introducción de herramientas analíticas avanzadas ha producido resultados positivos para muchos equipos, lo que subraya el valor de estudiar métricas específicas dentro de cada deporte.

Este desarrollo se centra en el fútbol, un deporte en el cual los análisis previos se han concentrado, en su mayoría, en predecir resultados de partidos y mejorar el rendimiento de los equipos. Sin embargo, este trabajo propone un enfoque diferente al analizar el impacto de los jugadores sobre la posesión de balón y los disparos del equipo desde una perspectiva probabilística.

A partir de la métrica PSL propuesta en el paper “Soccer Networks” (Huang et al., n.d.), planteamos un proceso para comparar el impacto que tienen los jugadores sobre la performance del equipo. Logramos formular una metodología para estudiar la distribución de la performance de un equipo. Luego, proponemos una serie de métodos y métricas para comparar el rendimiento de dos formaciones de jugadores. Además, desarrollamos una forma de representación vectorial (Embeddings) de los jugadores, llamada Player2Vec, un modelo de Machine Learning también basado sobre el modelo de redes de jugadores planteado en el mismo paper “Soccer Networks”. Esto último nos permite desarrollar modelos predictivos sobre el rendimiento de los jugadores en un equipo.

---

\*21F78 | tglaberman@mail.utdt.edu

†21R1160 | ipardo@mail.utdt.edu

‡21Q111 | jsilvestri@mail.utdt.edu

# Índice

<b>Índice</b>	<b>2</b>
<b>Índice</b>	<b>3</b>
<b>1 Agradecimientos</b>	<b>4</b>
<b>2 Introducción</b>	<b>4</b>
<b>3 Motivación Justificación del tema</b>	<b>4</b>
3.1 Relevancia Académica . . . . .	4
3.2 Relevancia Práctica . . . . .	4
<b>4 Objetivos de Proyecto</b>	<b>5</b>
4.1 Objetivo General . . . . .	5
4.2 Objetivos Específicos . . . . .	5
<b>5 Definición del problema</b>	<b>6</b>
5.1 PSL como métrica de Performance . . . . .	6
5.2 Modelo de Red de Jugadores . . . . .	6
5.3 Modelo Predictivo de probabilidades de transición . . . . .	7
5.4 Test de Sensibilidad sobre PSL . . . . .	9
5.5 Modelo Predictivo sobre $r(J, S)$ . . . . .	10
<b>6 Análisis de las distribuciones de los <math>r(J, S)</math></b>	<b>11</b>
6.1 Comparación de las distribuciones de los $r(J, S)$ . . . . .	14
<b>7 Estimación de la Distribución del PSL</b>	<b>17</b>
7.1 Variables Aleatorias para los $r(U, V)$ y PSL por Priors . . . . .	17
7.2 Proceso de Monte Carlo para estimar la distribución del PSL . . . . .	18
7.3 Comparar el impacto sobre el PSL de dos jugadores en una formación . . . . .	19
7.4 Comparación de Distribuciones de PSL . . . . .	20
7.4.1 Comparación de Momentos Estadísticos . . . . .	20
7.4.2 Dominancia Probabilística . . . . .	21
7.4.3 Comparación de CDFs de las distribuciones de PSL . . . . .	21
7.4.4 Dominancia Estocástica . . . . .	22
7.4.5 Conclusiones sobre la Comparación de Distribuciones de PSL . . . . .	22
<b>8 Player2Vec: Embeddings de Jugadores</b>	<b>23</b>
8.1 Definición . . . . .	23
8.2 Modelado de la EPL 2012/13 como Grafo . . . . .	23
8.3 Implementación . . . . .	27
8.4 Visualización y Exploración de los Embeddings . . . . .	27
8.5 Potencial de Player2Vec . . . . .	30
<b>9 Modelo predictivo de Distribuciones de <math>r(U, V)</math></b>	<b>32</b>
9.1 Definición . . . . .	32
9.2 Modelo . . . . .	32
9.3 Datos . . . . .	32
9.4 Implementación . . . . .	32
9.5 Entrenamiento . . . . .	33
9.6 Resultados iniciales . . . . .	33
9.7 Tuning de Hiperparámetros y Arquitectura con Validación Cruzada . . . . .	33
9.8 Espacio de Hiperparámetros . . . . .	34
9.9 Resultados del Tuning de Hiperparámetros . . . . .	34
9.10 Hardware y Tiempos de Entrenamiento . . . . .	35
<b>10 Validación del Modelo de Distribuciones</b>	<b>36</b>

10.1 Caso de Estudio 1: Danny Welbeck al Arsenal . . . . .	36
10.2 Caso de Estudio 2: James Milner al Liverpool . . . . .	38
10.3 Recomendaciones de Transferencias Clave . . . . .	39
<b>11 Resultados</b>	<b>40</b>
<b>12 Discusión</b>	<b>40</b>
<b>13 Conclusiones &amp; Recomendaciones {#conclusiones-&amp;-recomendaciones}</b>	<b>40</b>
<b>14 Referencias bibliográficas</b>	<b>41</b>
<b>15 Apéndices: Tablas, figuras, anexos</b>	<b>41</b>
<b>Índice de Figuras</b>	<b>41</b>
<b>Índice de Tablas</b>	<b>42</b>
<b>Índice de Algoritmos</b>	<b>42</b>

## **Índice**

# 1 Agradecimientos

Este trabajo no hubiera sido posible sin la ayuda de los profesores Gustavo Vulcano (Escuela de Negocios, Universidad Torcuato Di Tella) y Santiago Gallino (The Wharton School, University of Pennsylvania). Además queremos agradecer a Ignacio Vigilante (TIC - Escuela ORT) y Tomás Spognardi (Exactas - UBA) por sus contribuciones al modelo de Player2Vec y al PSL Bayesiano respectivamente.

**Ignacio Pardo quiere agradecer** a su familia, amigos, colegas y jefe (Dario Mischener) por su apoyo y acompañamiento durante el transcurso de su carrera universitaria.

## 2 Introducción

A diferencia de otros deportes como el béisbol o el basketball, el fútbol ha sido tradicionalmente menos propenso a la aplicación de técnicas avanzadas de análisis de datos y aprendizaje automático. Sin embargo, en los últimos años ha habido un crecimiento significativo en el uso de herramientas analíticas para evaluar el rendimiento de los jugadores y los equipos.

En la última década el análisis del futbol ha evolucionado hacia una perspectiva cada vez más matemática y sofisticada. El desarrollo que más impacto tuvo sin dudas es el de la métrica de Expected Goals (**xG**) que permiten evaluar la calidad de las oportunidades de gol de un equipo. El uso de **xG** en el análisis de partidos y jugadores ha permitido una mayor capacidad predictiva y una mejor comprensión del rendimiento de los equipos.

El trabajo en desarrollo “Soccer Networks” (Huang et al., n.d.) propone un modelo de red de jugadores para calcular la probabilidad de disparar al arco antes de perder el balón (**PSL**), lo cual demuestran tiene alta correlación con el rendimiento del equipo y una gran importancia al nivel del **xG**.

Este trabajo profundiza en el análisis de la métrica **PSL** y propone un análisis probabilístico sobre las componentes del modelo de redes de jugadores y su injerencia en el rendimiento de los jugadores y consecuentemente del equipo. Finalmente, proponemos una metodología para comparar el rendimiento de jugadores y formaciones de jugadores en base a la métrica **PSL**.

## 3 Motivación Justificación del tema

El fútbol es uno de los deportes más populares y seguidos en todo el mundo. La capacidad de un equipo para ganar partidos y campeonatos depende en gran medida de la calidad y el rendimiento de sus jugadores. En este contexto, la identificación y selección de los mejores jugadores para un equipo se convierte en una tarea crucial para entrenadores, directores deportivos y analistas de rendimiento.

### 3.1 Relevancia Académica

Desde una perspectiva académica, el análisis del rendimiento de los jugadores de fútbol ha sido un área de interés creciente en los últimos años. La aplicación de técnicas avanzadas de análisis de datos, aprendizaje automático y modelos probabilísticos ha permitido una comprensión más profunda del impacto de los jugadores en el rendimiento del equipo. Algunos ejemplos del estado del arte incluyen el modelo para maximizar la posesión esperada propuesto en el artículo de Rahimian et al. (2023) (Rahimian et al., 2023) y el modelo de redes de jugadores para calcular la probabilidad de disparar al arco antes de perder el balón (PSL) presentado en el trabajo de Huang et al. (Huang et al., n.d.).

Este trabajo se enmarca en esta línea de investigación, contribuyendo al desarrollo de nuevas metodologías y herramientas para evaluar y comparar el rendimiento de los jugadores.

### 3.2 Relevancia Práctica

En el ámbito práctico, la capacidad de identificar a los mejores jugadores tiene implicaciones directas en la toma de decisiones estratégicas y operativas de los equipos de fútbol. La correcta selección de jugadores puede mejorar significativamente el rendimiento del equipo, aumentar las probabilidades de éxito en competiciones y optimizar la inversión en fichajes. Además, el uso de modelos avanzados como Player2Vec y PSL Bayesiano proporciona una ventaja competitiva por su poder predictivo del rendimiento de los jugadores.

## 4 Objetivos de Proyecto

### 4.1 Objetivo General

El objetivo principal de este proyecto es desarrollar y aplicar modelos avanzados de análisis de datos y probabilísticos, para mejorar la evaluación, comparación y selección de jugadores de fútbol. Esto permitirá a los equipos tomar decisiones más informadas y estratégicas, optimizando su rendimiento y aumentando sus probabilidades de éxito en competiciones. Mas concretamente, este trabajo busca responder la pregunta del título “¿Cómo encontrar al jugador ideal para tu equipo de fútbol?”.

### 4.2 Objetivos Específicos

1. **Desarrollar un Modelo de Evaluación del Rendimiento de Jugadores:**
  - Analisar las componentes del modelo de red de jugadores.
  - Explotar el modelo PSL para estimar el impacto de los jugadores en el rendimiento del equipo.
  - Crear una representación vectorial de cada jugador para poder utilizarlas en modelos predictivos.
2. **Comparar el Rendimiento de Jugadores:**
  - Establecer métricas estandarizadas para comparar objetivamente el rendimiento de jugadores en diferentes posiciones y roles.
  - Aplicar técnicas de aprendizaje automático y análisis de datos para identificar patrones y tendencias en el rendimiento de los jugadores.
3. **Optimizar la Selección de Jugadores:**
  - Desarrollar un sistema de recomendación para identificar a los jugadores que mejor se adaptan a las necesidades y estrategias específicas de un equipo.
  - Evaluar la efectividad del sistema de recomendación mediante estudios de caso y análisis de datos históricos.
4. **Validar los Modelos:**
  - Realizar pruebas y validaciones de los modelos desarrollados utilizando datos reales de partidos y jugadores.
5. **Generar Conocimiento y Herramientas para la Comunidad:**
  - Documentar y publicar los resultados y metodologías desarrolladas en el proyecto.
  - Crear herramientas y recursos accesibles para entrenadores, analistas y directores deportivos que deseen aplicar estos modelos en sus equipos.

## 5 Definición del problema

A partir de la pregunta de la investigación, se plantea el problema de encontrar el jugador ideal para un equipo de fútbol. En un comienzo nos encontramos planteando como definir *performance* de un jugador y cómo compararla con otros jugadores. Surgió la necesidad de encontrar una métrica evaluar el impacto de un jugador en el rendimiento de un equipo y como definir estos agentes. Además es necesario poder representar concretamente a un Jugador  $J$ .

### 5.1 PSL como métrica de Performance

En el paper en proceso *How to Find the Right Player for your Soccer Team?* (Huang et al.) se plantea la descomposición del Gol Esperado ( $xG$ ) como:

$$xG(A) = P(A) \cdot PSL(A) \cdot SA(A)$$

Donde  $A$  es el equipo,  $P(A)$  es la posesión del balón,  $PSL(A)$  es la probabilidad patear al arco antes de perder el balón y  $SA(A)$  es la probabilidad de que un disparo al arco se convierta en gol. A diferencia de la posesión del balón y la probabilidad de convertir un disparo en gol,  $PSL(A)$  no es una métrica comúnmente utilizada en el análisis de fútbol ni existen modelos que la calculen. El paper plantea un modelo de red de jugadores que permite calcular  $PSL(A)$  para cada equipo.

### 5.2 Modelo de Red de Jugadores

Utilizando Cadenas de Markov de Tiempo Continuo (CTMC) se puede calcular la probabilidad de que un equipo pierda el balón antes de patear al arco. En este modelo de red de jugadores se plantea un modelo de 14 estados: 11 jugadores ( $J_1 \dots J_{11}$ ), Ganancia, Pérdida y Disparo.

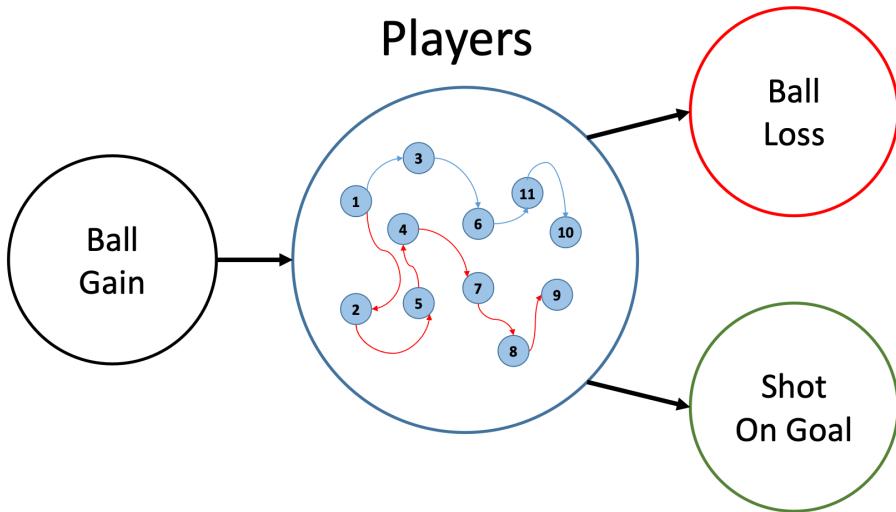


Figure 1: Modelo de Red de Jugadores

El grafo presentado en la figura representa el modelo de red de jugadores. Cada nodo representa un estado y cada arista representa una transición entre estados. El nodo verde representa el estado de disparo al arco, el rojo la pérdida del balón y el azul la ganancia del balón por parte de un jugador. Los ejes entre los nodos se representan con una matriz de adyacencia  $R$  donde cada valor  $r(U, V)$  representa la ratio de transición entre los estados  $U$  y  $V$ .

$$R = \begin{pmatrix} 0 & r(G, J_1) & \dots & r(G, J_{11}) & 0 & 0 \\ 0 & 0 & \dots & r(J_1, J_{11}) & r(J_1, L) & r(J_1, S) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & r(J_{11}, J_1) & \dots & 0 & r(J_{11}, L) & r(J_{11}, S) \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix}$$

A partir de la matriz de ratio de acción sobre tiempo jugado  $R$  (ganancias, pases a otro jugador, disparos o pérdidas) se puede obtener la matriz de transición de estados  $Q$  para el CMTTC de normalizar las filas de  $R$ :

Para cada par de estados  $U$  y  $V$  se define  $q(U, V) = \frac{r(U, V)}{\sum_{i=1}^{14} r(U, i)}$

$$Q = \begin{pmatrix} 0 & q(G, J_1) & \dots & q(G, J_{11}) & 0 & 0 \\ 0 & 0 & \dots & q(J_1, J_{11}) & q(J_1, L) & q(J_1, S) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & q(J_{11}, J_1) & \dots & 0 & q(J_{11}, L) & q(J_{11}, S) \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix}$$

Finalmente a partir de la matriz de probabilidades de transición  $Q$  se puede calcular  $PSL(A)$  como:

$$PSL(A) = [1, 0, \dots, 0] \cdot (I - T)^{-1} \cdot X \cdot [0, 1]^T$$

Siendo  $T$  las probabilidades de transición de los estados transitorios,  $X$  las probabilidades de transición de los estados transitorios a los estados absorbentes e  $I$  la matriz identidad.

A partir de este modelo en el paper se evaluó para una temporada de la Premier League (EPL 2012/13) la diferencia entre los PSL de cada equipo y luego de forma empírica se demuestra como el  $PSL(A)$  tiene alta correlación positiva con el rendimiento del equipo por sobre el contrincante. Finalmente hayamos una métrica significativa de rendimiento de un equipo en la métrica  $PSL$ . Sin embargo, da a lugar a la investigación de como se puede aplicar esta métrica a nivel de jugador y como se puede comparar el rendimiento de jugadores en distintos equipos.

Para evaluar el impacto de un jugador  $J$  se debe, o bien conocer la probabilidad de transición entre  $J$  y los otros 13 estados (10 jugadores, Ganancia, Pérdida y Disparo) o bien lograr estimar la probabilidad de transición entre  $J$  y los otros 13 estados.

En este trabajo se propone un método probabilístico bayesiano para hallar la Distribución del PSL dada la distribución de probabilidades de transición entre cada uno de los 11 jugadores y los otros 13 estados.

### 5.3 Modelo Predictivo de probabilidades de transición

En un comienzo se planteó desarrollar un modelo predictivo para estimar las ratios de transición entre los estados. Optamos por buscar predecir los ratios  $r$  y no las probabilidades de transición  $q$  ya que la normalización no es igual en cada instancia de  $R$ . Mas concretamente buscamos estimar la función  $f$  que mapea los estados  $U$  y  $V$  a la ratio de transición  $r(U, V)$ .

$$\hat{r}(U, V) = f(U, V, \theta)$$

Comenzamos armando un modelo para predecir únicamente los ratios de pases  $r(J_i, J_j)$  entre un jugador  $J_i$  y otro jugador  $J_j$ . Lo que correspondiera a los siguientes valores de la matriz  $R$ :

$$R = \begin{pmatrix} 0 & r(G, J_1) & \dots & r(G, J_{11}) & 0 & 0 \\ 0 & 0 & \dots & r(J_1, J_{11}) & r(J_1, L) & r(J_1, S) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & r(J_{11}, J_1) & \dots & 0 & r(J_{11}, L) & r(J_{11}, S) \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix}$$

Para poder utilizar un modelo de machine learning tradicional necesitamos de poder representar a cada jugador  $J$  de forma vectorial. Armamos un vector de métricas agregadas para un jugador al momento del partido a predecir. Estas métricas incluyen la cantidad de pases, disparos, goles, pérdidas, etc. sobre el total de tiempo jugado, ademas de el equipo en el que juega.

$$J = [\text{Passes}/90, \text{Shots}/90, \text{Goals}/90, \text{Losses}/90, \text{Time Played}, \text{Team ID}]$$

Para el modelo predictivo comenzamos utilizando un modelo de XGBoost para la regresión pero rápidamente observamos que por la naturelza de arbol al predecir con la media de las observaciones por hoja las predicciones resultaban casi discretas, por lo que viramos a explorar un modelo de regresión lineal para predecir los ratios de pases entre jugadores.

Para validar elegimos separar de forma temporal los 380 partidos de la temporada 2012/13 de la EPL: los primeros 269 partidos de entrenamiento; los últimos 111 de test ( $\mu + 2/3\sigma$ ). Ademas para construir el dataset, elegimos agarrar parejas de jugadores de los partidos de Train y removerlos de los mismos para poder en Test predecir ratios de transición entre jugadores que no se vieron en Train.

Luego de entrenar el modelo, para cada instancia de test obtuvimos la matriz de ratios de transición  $R$  y calculamos el PSL real, para luego predecir la matriz de transición  $\hat{R}$  y calcular el PSL predicho. Finalmente calculamos el coeficiente de correlación de Pearson entre el PSL real y el PSL predicho.

En el siguiente gráfico podemos observar como a pesar de predecir muy pobre los ratios de transición al resultar en un coeficiente de correlación de Pearson entre los  $r(J_i, J_j)$  y los  $\hat{r}(J_i, J_j)$  de 0.12, sin embargo al comparar el PSL real del PSL calculado a partir de  $\hat{R}$  se obtiene un coeficiente de correlación de Pearson de 0.85.

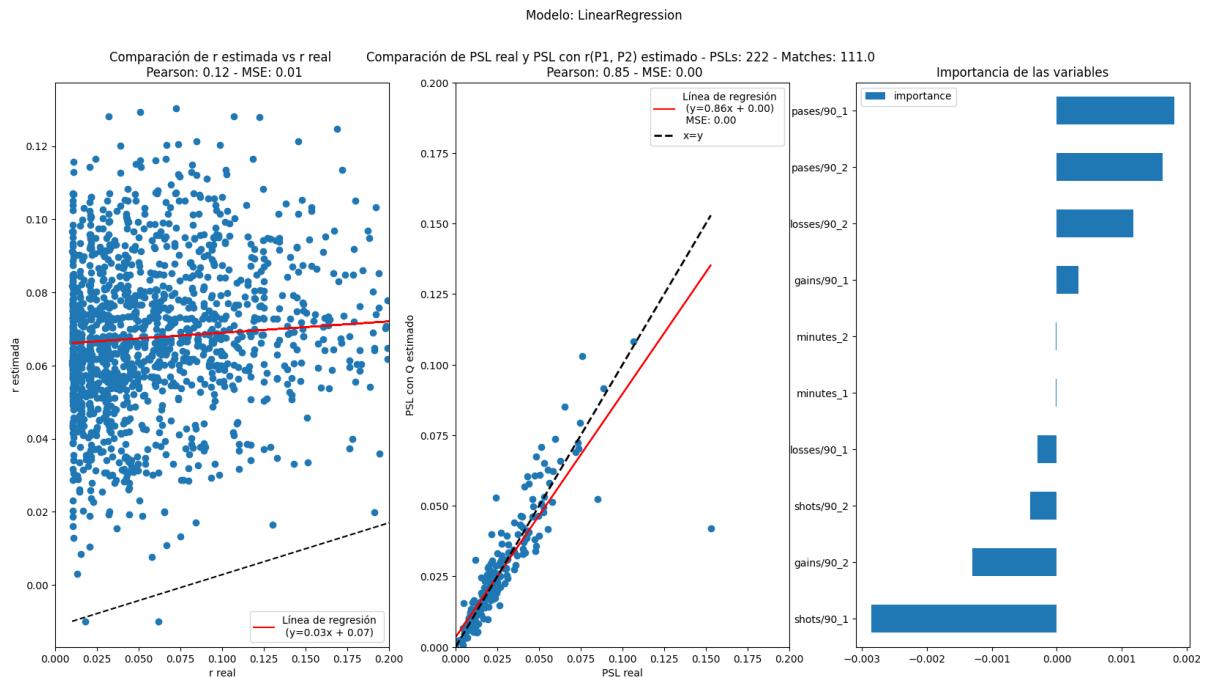


Figure 2: Resultados Modelo de Regresión Lineal

El modelo planteado no es capaz de predecir los ratios de transición, y a pesar de que desarrollamos otros modelos como XGBoost para regresión, Redes Neuronales y Redes Neuronales Probabilísticas (PNNs) no es posible predecir los ratios de transición entre los estados a partir de las métricas de los jugadores. Esto se debe principalmente a la cantidad de datos y la poca relación entre ellos. Al evaluar como resolver la predicción de los  $r(J_i, J_j)$  decidimos observar como cada ratio de transición afecta al PSL.

#### 5.4 Test de Sensibilidad sobre PSL

Para entender mejor la relación entre los ratios de transición y el PSL, se implementó el modelo en una librería de auto-diferenciación (pytorch) y se obtuvo el gradiente de PSL empíricamente. Esto nos permitió entender que estados tienen mayor influencia en la métrica que estamos analizando. Pudimos observar que las transiciones de Jugador a Shot son las que mas inciden sobre el PSL, seguido por las transiciones entre jugadores, tal como se observa en la figura.

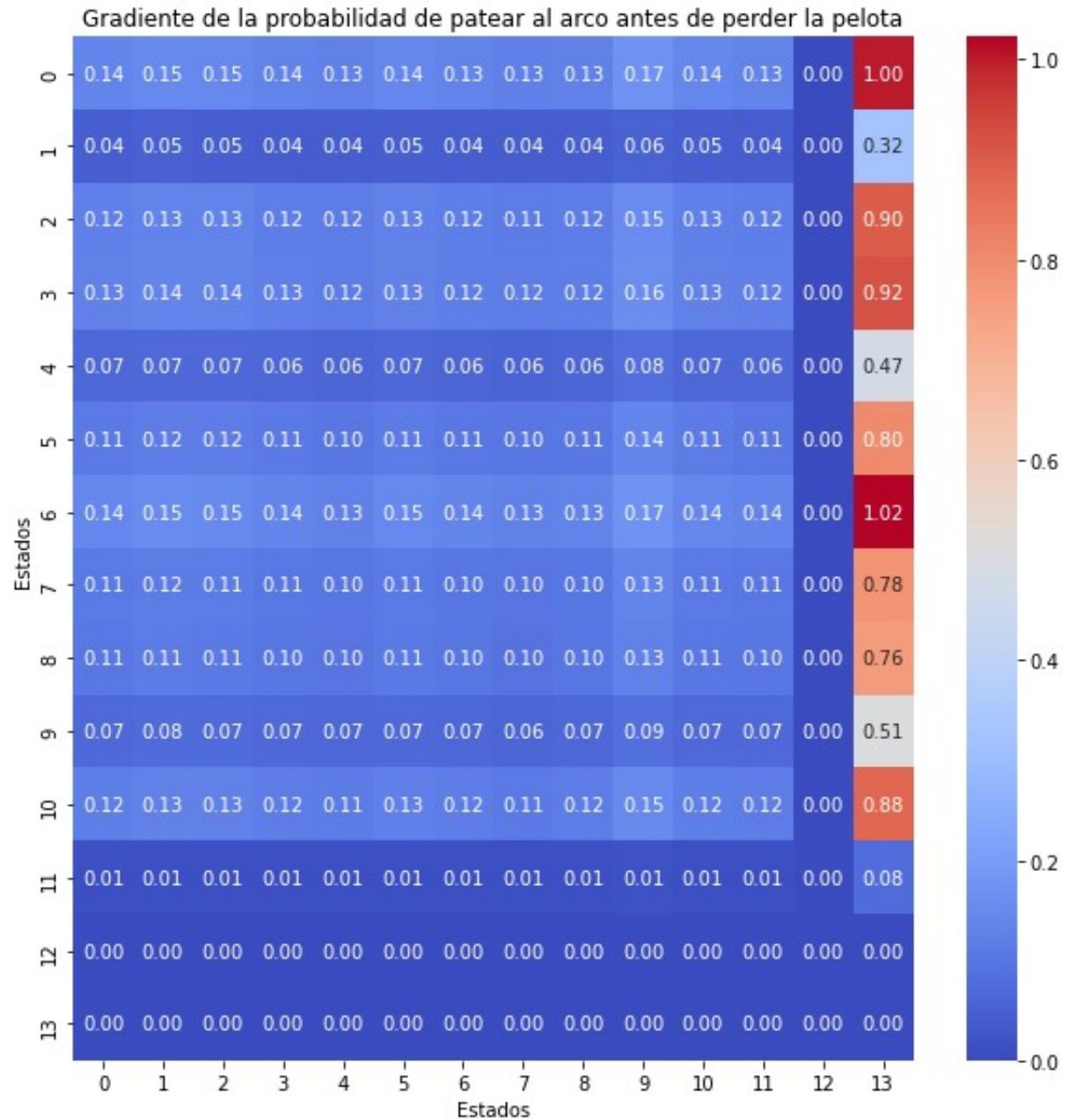


Figure 3: Gradiente del PSL

## 5.5 Modelo Predictivo sobre $r(J, S)$

Luego de lo observado con el Test de Sensibilidad sobre PSL, decidimos cambiar el enfoque de la predicción de los ratios de transición entre jugadores a la predicción de los ratios de transición entre jugadores y el estado de disparo al arco. Esto se debe a que al observar la matriz de ratios de transición  $R$  se observa que los ratios de transición entre jugadores y el estado de disparo son los que más afectan al PSL.

El nuevo modelo se enfoca en la siguiente sección de la matriz  $R$ :

$$R = \begin{pmatrix} 0 & r(G, J_1) & \dots & r(G, J_{11}) & 0 & 0 \\ 0 & 0 & \dots & r(J_1, J_{11}) & r(J_1, L) & r(J_1, S) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & r(J_{11}, J_1) & \dots & 0 & r(J_{11}, L) & r(J_{11}, S) \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix}$$

Para el vector de los jugadores  $J$  se agregó tambien la posición en la que juega (Arquero, Defensor, Mediocampista, Delantero) one-hot-encoded.

Luego se entrenó un modelo de XGBoost para Regresión con el mismo split de Train y Test. Se logró obtener un mejor resultado sobre la predicciones de Train en comparación al modelo anterior, sin embargo al evaluar en Test. Se obtuvo un coeficiente de correlación de Pearson de 0.95 entre los  $r(J_i, S)$  y los  $\hat{r}(J_i, S)$  en Train, pero de 0.08 en Test.

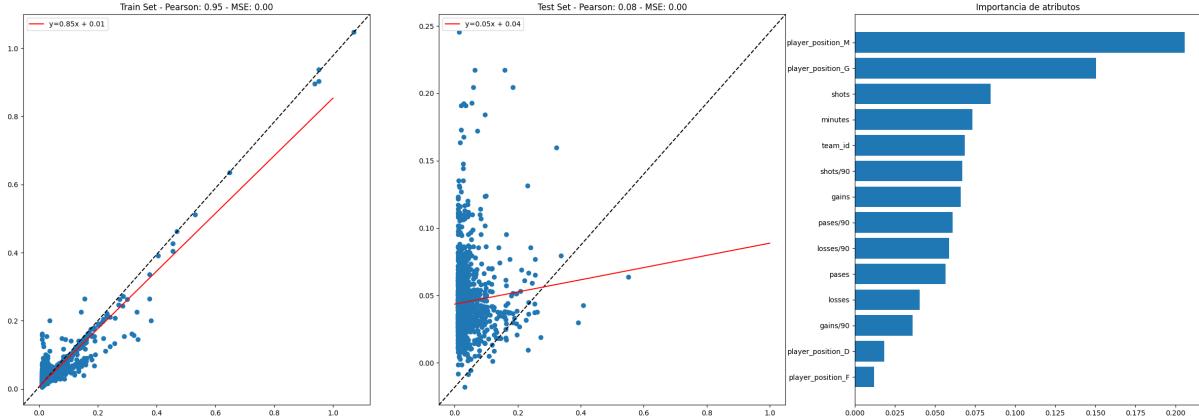


Figure 4: Resultados Modelo de XGBoost

Este resultado junto al del modelo de predicción de ratios de pases nos llevó a buscar una mejor representación vectorial de los jugadores. En la sección de Player2Vec se explica el modelo utilizado para obtener un vector de representación (embedding  $E$ ) de cada jugador. Con este embedding por construimos una red neuronal, el modelo resultante  $f(E_J, \text{partido})$  dado el embedding de los jugadores y el partido predice los ratios de transición entre jugadores y el estado de disparo al arco.

## 6 Análisis de las distribuciones de los $r(J, S)$

En un esfuerzo de comprender mejor el modelo de ratios de transición entre jugadores y el estado de disparo al arco, se decidió analizar las distribuciones de los  $r(J, S)$  para cada jugador en la temporada 2012/13 de la EPL.

Se observó que las distribuciones de los ratios de transición entre jugadores y el estado de disparo al arco tienen moda cercana a 0, lo que indica que la mayoría de los jugadores tienen una baja probabilidad de disparar al arco antes de perder el balón. En la siguiente figura se puede observar la distribución de los  $r(J, S)$  para todos los jugadores de la temporada 2012/13 de la EPL en todos los partidos.

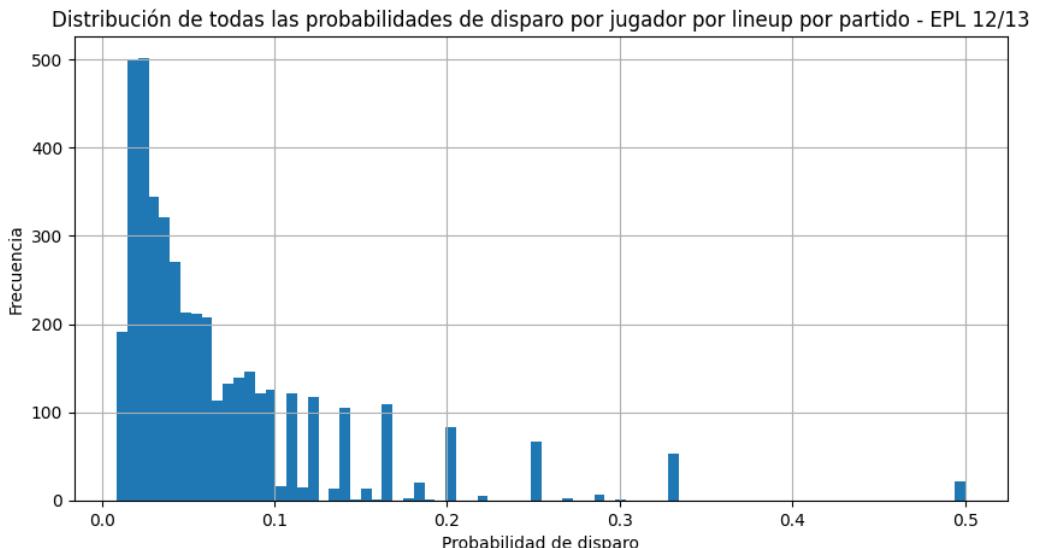


Figure 5: Distribución de todos los  $r(J, S)$

Además, se observó que la distribución de los  $r(J, S)$  de cada jugador no necesariamente sigue una distribución normal ni similar a la de otros jugadores.

Para el siguiente análisis se ajustaron las distribuciones de los  $r(J, S)$  de cada jugador a una distribución de probabilidad beta y se obtuvieron los parámetros  $\alpha$  y  $\beta$  de cada jugador.

Inicialmente presentamos la distribución de dos jugadores a modo de ejemplo: **Sergio Agüero** y **Robin van Persie**

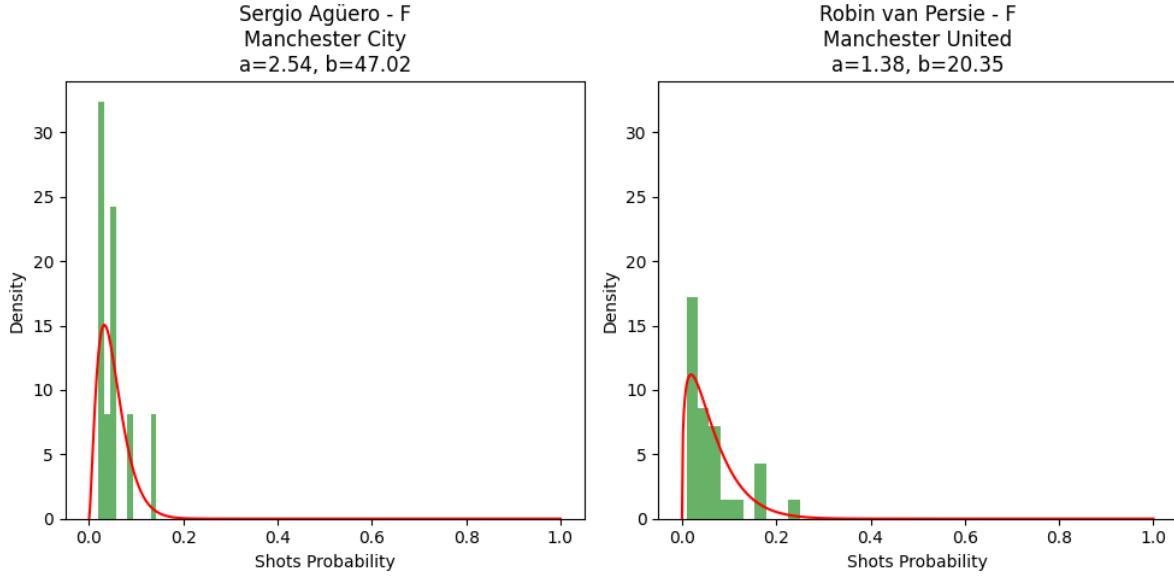


Figure 6: Distribución de los  $r(J, S)$  de Sergio Agüero y Robin van Persie

Luego se analizó la distribución de los  $r(J, S)$  de los 10 jugadores con mayor cantidad de disparos, con mayor sesgo y con mayor suma de disparos a modo de comparación.

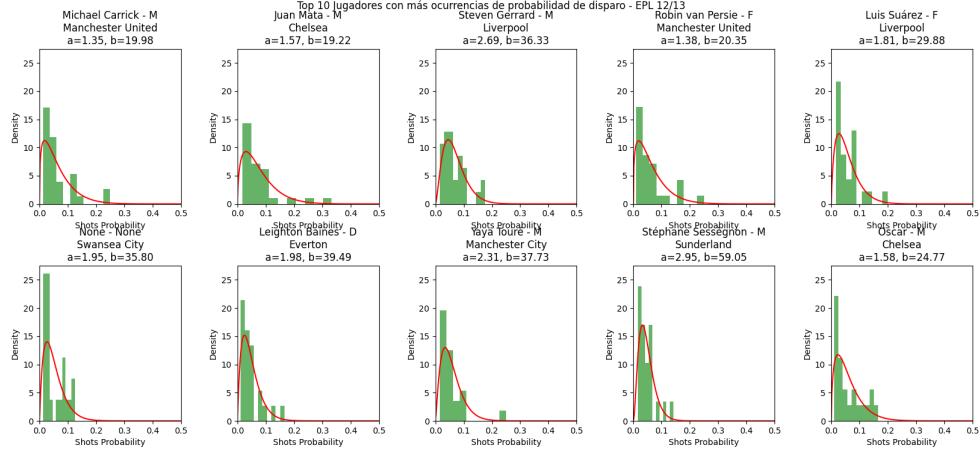


Figure 7: Distribución de los  $r(J, S)$  de los 10 jugadores con mayor cantidad de disparos

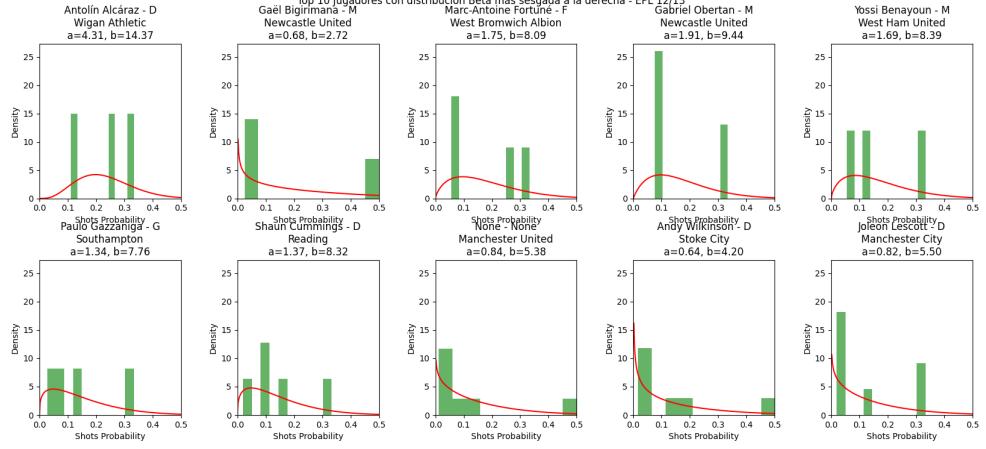


Figure 8: Distribución de los  $r(J, S)$  de los 10 jugadores con mayor sesgo

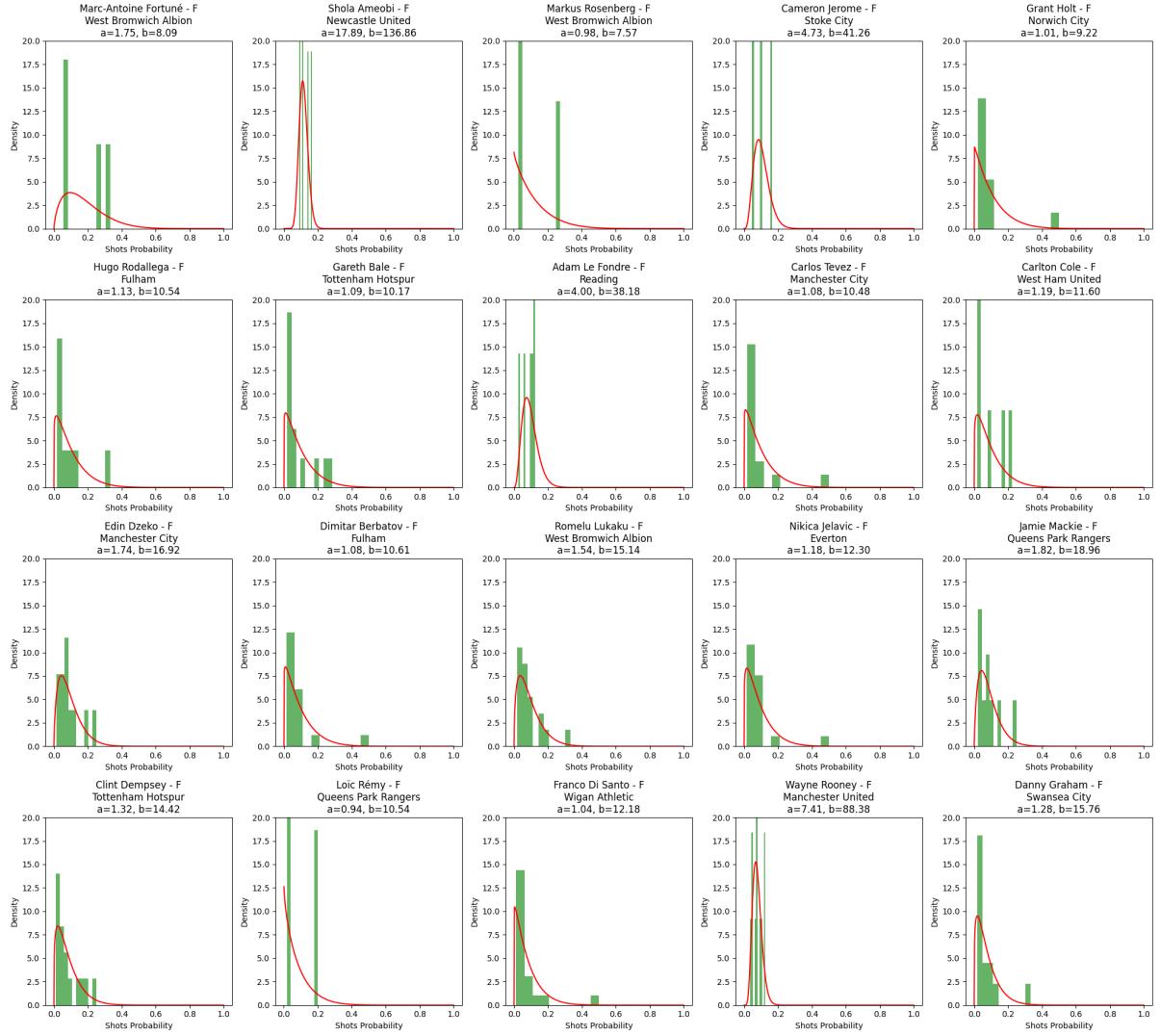


Figure 9: Top 20 Delanteros con distribución Beta más sesgada a la derecha - EPL 12/13

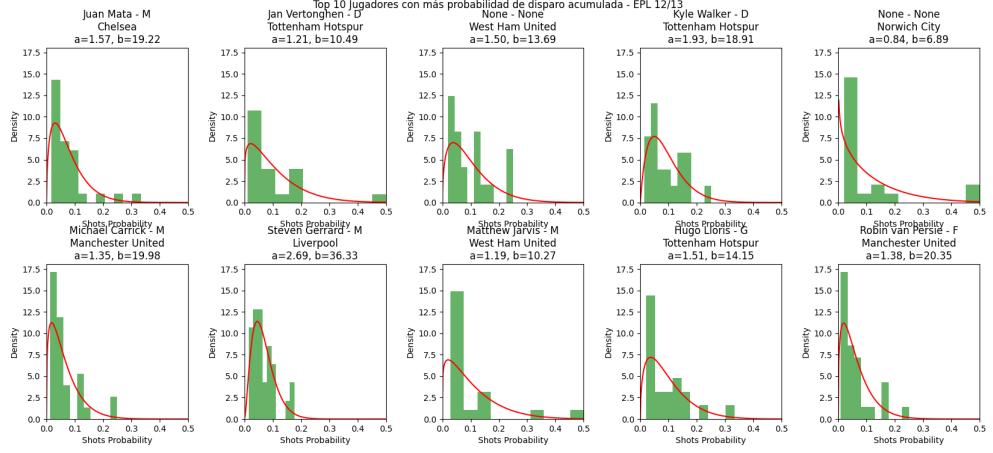


Figure 10: Distribución de los  $r(J, S)$  de los 10 jugadores con mayor suma

## 6.1 Comparación de las distribuciones de los $r(J, S)$

A partir de la distribución ajustada de un jugador, podemos luego hayar por ejemplo jugadores similares en base a la distribución de los  $r(J, S)$  utilizando la divergencia de Kullback-Leibler (KL).

En el siguiente gráfico se observa la distribución de los  $r(J, S)$  de jugadores similares a él en la temporada 2012/13 de la EPL. Además se presentan solapados en otra figura.

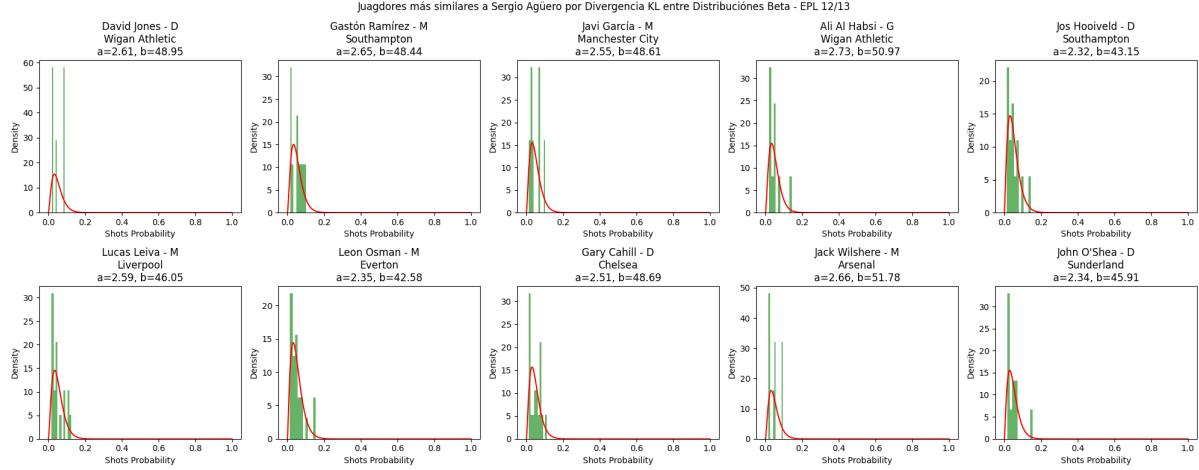


Figure 11: Distribución de los  $r(J, S)$  de jugadores similares a Sergio Agüero

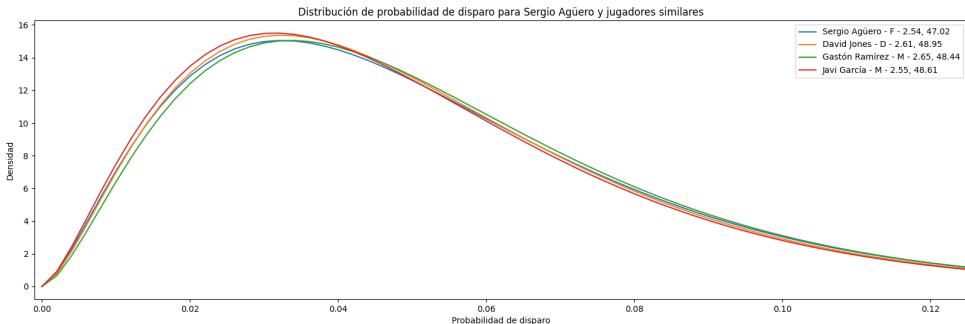


Figure 12: Distribución de los  $r(J, S)$  de jugadores similares a Sergio Agüero Superpuestos

Finalmente podemos agregar la condición de *misma posición* al comparar dos jugadores, en el caso de Agüero de Delantero (F por Forward) y hayan nuevamente jugadores aún más similares a él.

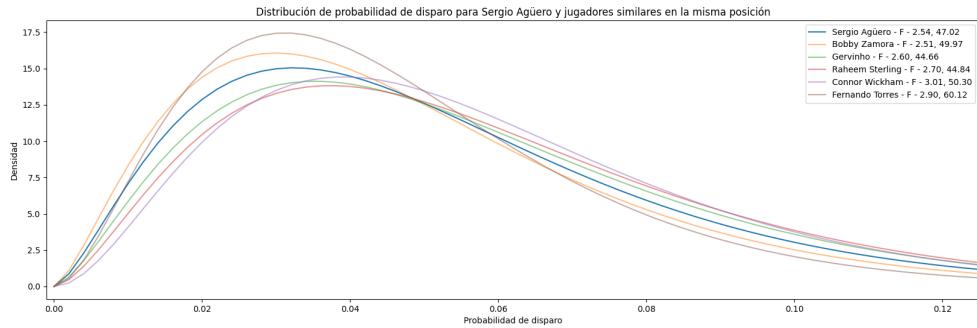


Figure 13: Distribución de los  $r(J, S)$  de jugadores similares a Sergio Agüero de la misma posición

Para conocer mejor la varianza de las distribuciones de los  $r(J, S)$  de los jugadores, se estudió la distribución de los parámetros  $\alpha$  y  $\beta$  de las distribuciones beta ajustadas. Hicimos un análisis de clustering para agrupar a los jugadores en base a sus distribuciones de los  $r(J, S)$ .

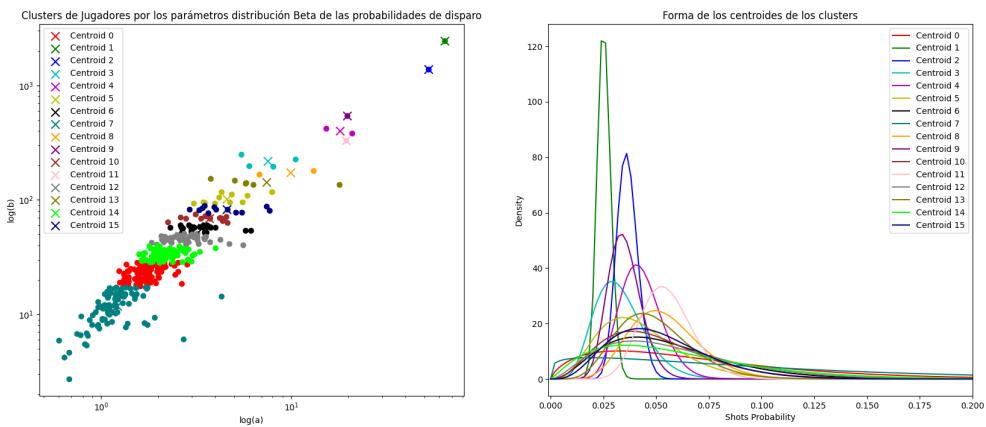


Figure 14: Distribución de los parámetros  $\alpha$  y  $\beta$  de los  $r(J, S)$  de los jugadores

Como un extra, este sistema de clustering nos permite hallar rápido jugadores similares entre sí. A partir de los clusters la siguiente figura presenta las posibles distribuciones en cada cluster.

Distribuciones de las probabilidades de disparo por cluster

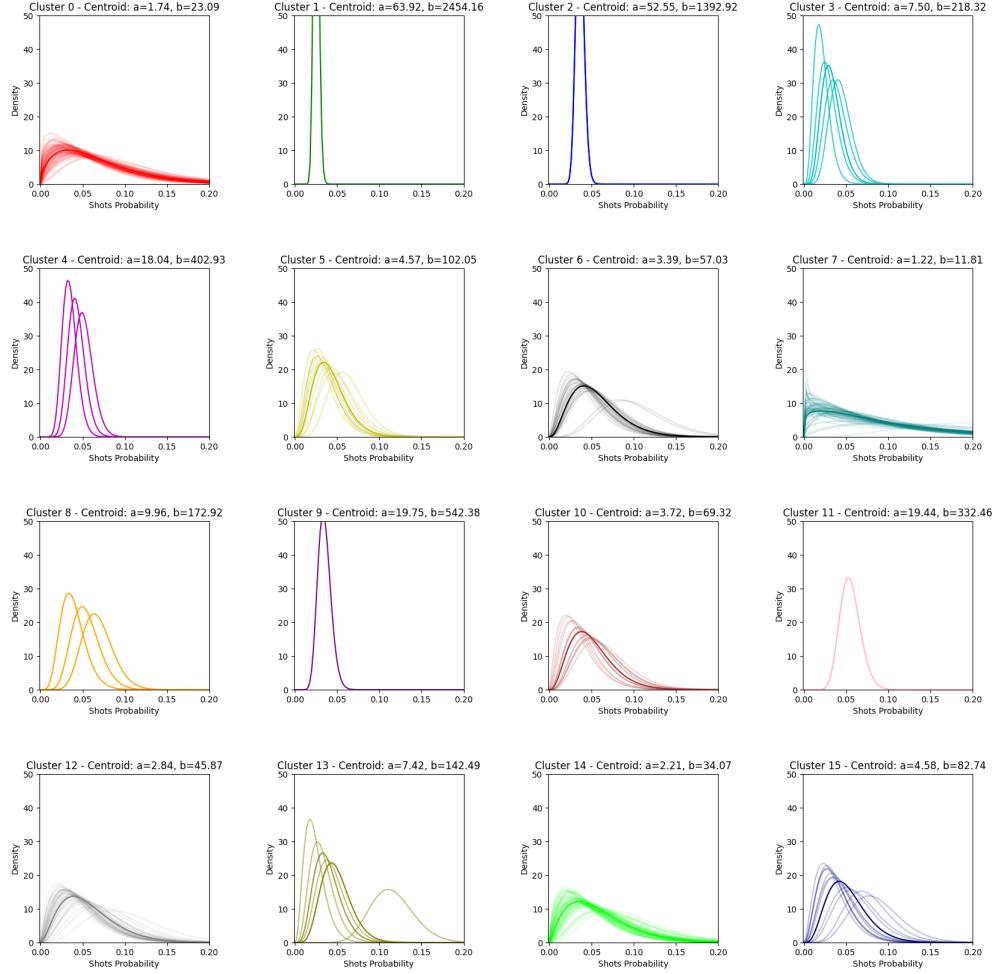


Figure 15: Distribución de los  $r(J, S)$  de jugadores en clusters

## 7 Estimación de la Distribución del PSL

A partir de los resultados obtenidos en el análisis de las distribuciones de los  $r(J, S)$ , se propone un utilizar estas como priors para cada jugador, es decir, se asume que la distribución de los  $r(J, S)$  de un jugador es la distribución a priori de la variable aleatoria  $r(J, S)$  para ese jugador, lo mismo para los  $r(J_i, J_j)$ , los  $r(J, L)$  y los  $r(J, G)$ .

De esta forma, cada jugador  $J$  tiene una distribución a priori para cada uno de los 14 estados, considerando esto, podemos reformular la matriz de ratios de transición como una matriz de variables aleatorias donde cada una se distribuye según la distribución a priori del jugador correspondiente.

### 7.1 Variables Aleatorias para los $r(U, V)$ y PSL por Priors

Para actualizar la notación, sean  $r_{J,V}$  la variable aleatoria que representa la ratio de transición entre el jugador  $J$  y el estado  $V$ , esto incluye  $r_{J,S}$ ,  $r_{J,L}$  y tambien  $r_{G,J}$ , asi como los  $r_{J_i,J_j}$  para  $i, j \in [1, 11]$ .

Luego  $r_{J,V} \sim F_x$  la distribución a priori de la variable aleatoria  $r_{J,V}$ .

Para generalizar el analisis de distribuciones planteadas en la sección anterior, se propone utilizar una distribución KDE (Kernel Density Estimation) a partir de los histogramas de los  $r(J, V)$  para modelar sus distribuciones, ya que no todos los ratios de transición siguen una distribución beta tan bien como los  $r(J, S)$ .

Finalmente obtenemos, para una formación dada de 11 jugadores, una matriz de variables aleatorias  $\mathbf{R}$ .

$$\mathbf{R} = \begin{pmatrix} 0 & r_{G,J_1} & \dots & r_{G,J_{11}} & 0 & 0 \\ 0 & 0 & \dots & r_{J_1,J_{11}} & r_{J_1,L} & r_{J_1,S} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & r_{J_{11},J_1} & \dots & 0 & r_{J_{11},L} & r_{J_{11},S} \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \end{pmatrix}$$

Para mejor claridad, la siguiente vizualización muestra la matriz de variables aleatorias  $\mathbf{R}$  para un equipo de ejemplo. En cada posición se observa la distribución a priori de la variable aleatoria correspondiente.

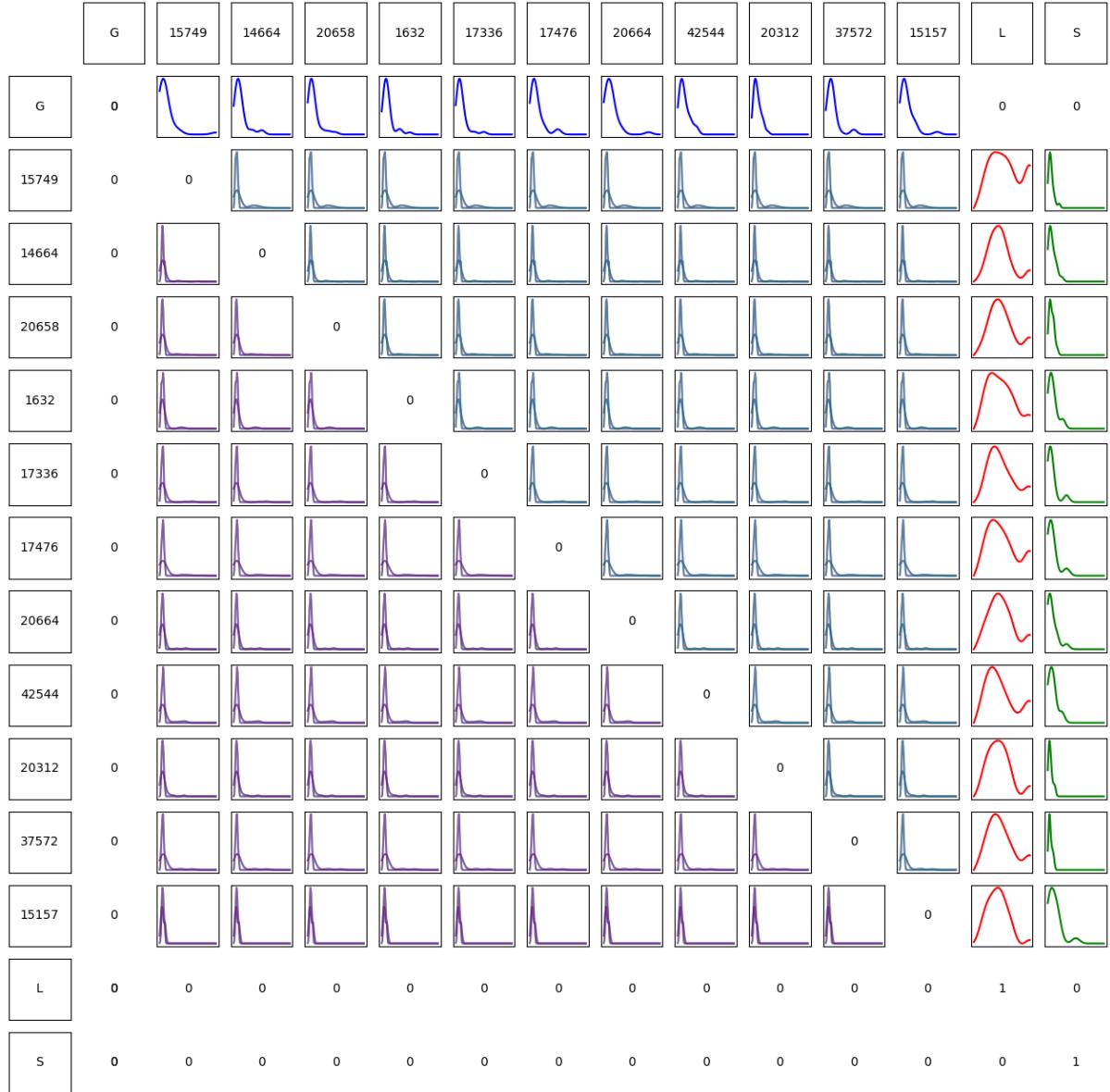


Figure 16: Matriz de Variables Aleatorias  $\mathbf{R}$

## 7.2 Proceso de Monte Carlo para estimar la distribución del PSL

Dado un equipo  $A$  con una formación de 11 jugadores  $L_A$ , se busca estimar la distribución del PSL de ese equipo a partir de las distribuciones a priori de los  $r(U, V)$  de cada jugador. Para ello, se propone un proceso de Monte Carlo para muestrear de las distribuciones a priori de los  $r(U, V)$ .

De la formación  $L_A$  podemos construir la matriz de variables aleatorias  $\mathbf{R}$  a partir de las distribuciones a priori de los  $r(U, V)$  de cada jugador.

Definimos  $\hat{f}_{PSL}^N(L_A)$  como la función distribución de probabilidad empírica de los  $PSL_i$  para la formación  $L_A$  en base a  $N$  simulaciones.

El proceso de Monte Carlo para estimar la distribución del PSL de la formación  $L_A$  es el siguiente:

<b>Input:</b> Número de simulaciones $N$
<b>Input:</b> Formación $L_A = \{J_1, J_2, \dots, J_{11}\}$
<b>Output:</b> Distribución del PSL del equipo $A$
<b>1</b> $\mathbf{R} \leftarrow$ Construir la matriz de variables aleatorias a partir de las distribuciones a priori de los $r(U, V)$ de cada jugador;
<b>2</b> $PSL_i \leftarrow 0$ para $i = 1, 2, \dots, N$ ;
<b>3</b> <b>for</b> $i = 1$ <b>to</b> $N$ <b>do</b>
<b>4</b> $R \leftarrow$ Muestrear de la matriz $\mathbf{R}$ distribuciones a priori de los $r(U, V)$ ;
<b>5</b> $Q \leftarrow$ Normalizar las filas de $R$ ;
<b>6</b> $PSL_i \leftarrow PSL(Q)$ ;
<b>7</b> <b>end</b>
<b>8</b> Estimar la distribución del PSL del equipo $A$ a partir de las $N$ observaciones obtenidas de las simulaciones;

**Algorithm 1:** Simulación del PSL del equipo  $A$

A partir de esta distribución del PSL, se puede realizar comparaciones entre diferentes formaciones de 11 jugadores.

El siguiente gráfico muestra la distribución del PSL de un equipo de ejemplo obtenida a partir de 1000 simulaciones del proceso de Monte Carlo para la formación mas utilizada en la temporada 2012/13 de la EPL del equipo Manchester City.

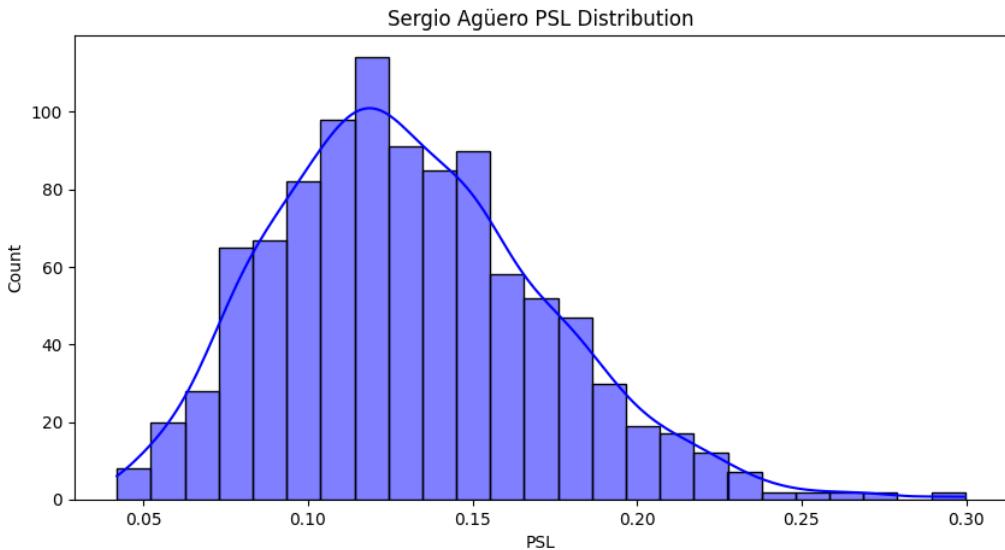


Figure 17: Distribución del PSL del equipo Manchester City

### 7.3 Comparar el impacto sobre el PSL de dos jugadores en una formación

Para comparar el PSL de dos jugadores en una formación, se propone un análisis de sensibilidad que consiste en evaluar el impacto en la distribución del PSL al reemplazar a un jugador por otro en la formación. El proceso para ello es el siguiente:

Se define la Formación  $L_A = \{J_1, J_2, \dots, J_{11}\}$  como la formación original del equipo  $A$ , donde alguno de los jugadores  $J_i$  es el jugador a “original”.

Se define el jugador  $J'$  a comparar con  $J_i$  y la formación  $L'_A = \{J_1, J_2, \dots, J_{11}\}$  como la formación con el jugador  $J'$  en lugar de  $J_i$ .

Luego, se puede computar  $\hat{f}_{PSL}^N(L_A)$  y  $\hat{f}_{PSL}^N(L'_A)$  para comparar las distribuciones del PSL de las formaciones  $L_A$  y  $L'_A$ .

## 7.4 Comparación de Distribuciones de PSL

En la siguiente sección postulamos una serie de métodos y métricas para comparar distribuciones de PSL de dos formaciones. En orden creciente de complejidad y rigurosidad, proponemos:

1. Comparación de Momentos Estadísticos
2. Dominancia Probabilística
3. Dominancia Estocástica

Para explicar la comparación de distribuciones de PSL, se propone un ejemplo de dos formaciones de 11 jugadores distintas, en una formación  $L_{MC}$  se encuentran 10 jugadores del equipo Manchester City + Sergio Agüero y en la otra  $L_{MC}^{\text{Giroud}}$  los mismos 10 jugadores + Olivier Giroud del equipo Arsenal.

Se realizó el proceso de Monte Carlo para estimar la distribución del PSL de cada formación a partir de 1000 simulaciones. Luego en la figura se puede observar las funciones de densidad de probabilidad aproximadas de las distribuciones del PSL de las formaciones  $\hat{f}_{PSL}^{1000}(L_{MC})$  y  $\hat{f}_{PSL}^{1000}(L_{MC}^{\text{Giroud}})$ .

### 7.4.1 Comparación de Momentos Estadísticos

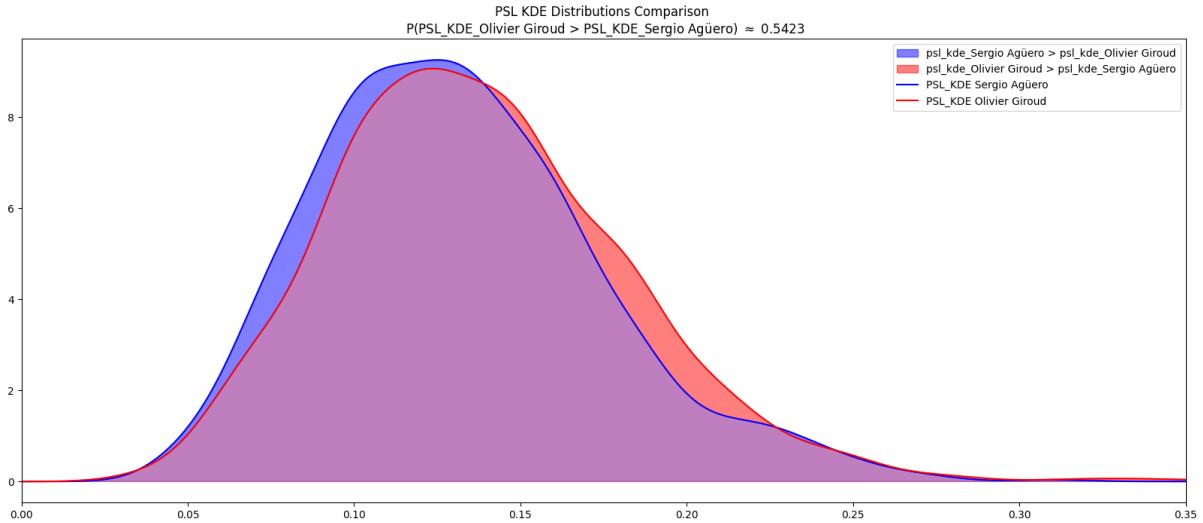


Figure 18: Ejemplo de dos distribuciones de PSL de dos formaciones distintas

Una posible comparación entre las distribuciones de PSL de dos formaciones es “a ojo” observando las funciones de densidad de probabilidad. En este caso puntual se puede observar como el equipo con Agüero tiene una distribución de PSL mas sesgada a la izquierda que el equipo con Giroud.

En un enfoque mas numérico, se puede realizar una comparación por momentos de las distribuciones de PSL de dos formaciones. Se propone comparar la media y la varianza de las distribuciones  $\hat{f}_{PSL}^{1000}(L_{MC})$  y  $\hat{f}_{PSL}^{1000}(L_{MC}^{\text{Giroud}})$  ya que el método de Monte Carlo nos permite obtener una muestra significativa de las distribuciones. Al no ser distribuciones normales, la skewness y la kurtosis nos proveen información adicional sobre la forma de la distribución.

Table 1: Comparación de momentos de  $\hat{f}_{PSL}^{1000}(L_{MC})$  y  $\hat{f}_{PSL}^{1000}(L_{MC}^{\text{Giroud}})$

Jugador	Media	Varianza	Desvio Estándar	Skewness	Kurtosis
Aguero	0.130861	0.041160	0.001694	0.554998	0.362611
Giroud	0.134403	0.043310	0.001876	0.580404	0.405658

Para este caso de ejemplo, se observa que la media y la varianza de las distribuciones de PSL de la formación  $L_{MC}$  y  $L_{MC}^{\text{Giroud}}$  son similares, aunque mayores en la formación con Giroud. Además, el tercer momento (skewness) nos confirma lo observado “a ojo” en las funciones de densidad de probabilidad, la distribución de PSL de la formación con Agüero es mas sesgada a la izquierda que la de la formación

con Giroud. Por último el cuarto momento (kurtosis) nos indica que la  $\hat{f}_{PSL}^{1000}(L_{MC}^{\text{Giroud}})$  tiene colas mas pesadas que la  $\hat{f}_{PSL}^{1000}(L_{MC})$ .

#### 7.4.2 Dominancia Probabilística

Otra forma de comparar las distribuciones de PSL de dos formaciones es a través de la dominancia probabilística.

En este caso, se puede calcular la probabilidad de que una muestra aleatoria de una distribución sea mayor que una muestra aleatoria de la otra distribución. De esta forma podemos tomar samples de las distribuciones  $\hat{f}_{PSL}^{1000}(L_{MC})$  y  $\hat{f}_{PSL}^{1000}(L_{MC}^{\text{Giroud}})$  y calcular la probabilidad de que un sample de la formación con Giroud sea mayor que un sample de la formación con Agüero.

Sean  $X_{L_{MC}} \sim \hat{f}_{PSL}^{1000}(L_{MC})$  y  $X_{L_{MC}^{\text{Giroud}}} \sim \hat{f}_{PSL}^{1000}(L_{MC}^{\text{Giroud}})$  las variables aleatorias que se distribuyen según las distribuciones de PSL de las formaciones  $L_{MC}$  y  $L_{MC}^{\text{Giroud}}$  respectivamente. Luego para evaluar si la formación con Giroud tiene dominancia probabilística sobre la formación con Agüero, se puede calcular la probabilidad  $P(X_{L_{MC}^{\text{Giroud}}} > X_{L_{MC}})$ .

El algoritmo para calcular la dominancia probabilística es el siguiente:

```

Input: Distribuciones de PSL  $\hat{f}_{PSL}^{1000}(L)$  y  $\hat{f}_{PSL}^{1000}(L')$ 
Output: Probabilidad de que un sample de PSL de la formación  $L$  sea mayor que un sample de PSL de la formación con  $L'$ 

1  $N \leftarrow 1000;$ 
2  $M \leftarrow 0;$ 
3 for  $i = 1$  to  $N$  do
4    $PSL \leftarrow$  Muestrear de  $\hat{f}_{PSL}^{1000}(L);$ 
5    $PSL' \leftarrow$  Muestrear de  $\hat{f}_{PSL}^{1000}(L');$ 
6   if  $PSL' > PSL$  then
7     |  $M \leftarrow M + 1;$ 
8   end
9 end
10  $P \leftarrow \frac{M}{N};$ 

```

**Algorithm 2:** Dominancia Probabilística

Para el caso de ejemplo, se obtuvo que la probabilidad de que un sample de PSL de la formación con Giroud sea mayor que un sample de PSL de la formación con Agüero es  $P(X_{L_{MC}^{\text{Giroud}}} > X_{L_{MC}}) \approx 0.5423$ . De esta forma podemos concluir que la formación con Giroud tiene dominancia probabilística sobre la formación con Agüero.

#### 7.4.3 Comparación de CDFs de las distribuciones de PSL

Otra forma de comparar las distribuciones de PSL de dos formaciones es a través de las funciones de distribución acumulada (CDF). Llamemos  $\hat{F}_{PSL}^N(L)$  a la función de distribución acumulada de PSL obtenida a partir de  $N$  simulaciones del proceso de Monte Carlo para la formación  $L$ .

En la siguiente figura se observa la comparación de las CDFs de las distribuciones de PSL de las formaciones  $L_{MC}$  y  $L_{MC}^{\text{Giroud}}$ .

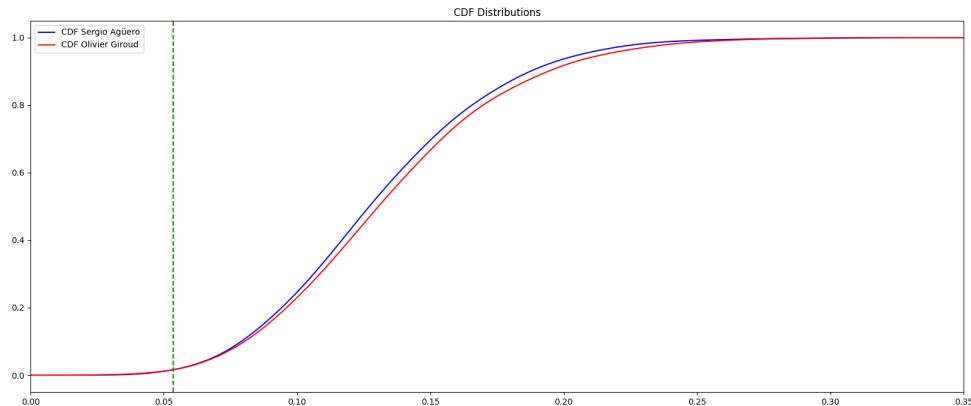


Figure 19: Comparación de CDFs de las distribuciones de PSL de las formaciones  $L_{MC}$  y  $L_{MC}^{Giroud}$

Nuevamente “a ojo” se puede analisar la relación entre las distribuciones  $\hat{F}_{PSL}^{1000}(L_{MC})$  y  $\hat{F}_{PSL}^{1000}(L_{MC}^{Giroud})$ , en este caso podemos ver como la CDF de la formación con Agüero es menor a la de la formación con Giroud en la mayoría de los puntos, lo que indica que la formación con Agüero tiene un PSL menor que la formación con Giroud en la mayoría de los casos.

#### 7.4.4 Dominancia Estocástica

Mas formalmente se puede evaluar la dominancia estocástica entre las CDFs  $\hat{F}_{PSL}^{1000}(L_{MC})$  y  $\hat{F}_{PSL}^{1000}(L_{MC}^{Giroud})$ . La dominancia estocástica es una relación de orden entre dos funciones de distribución acumulada que indica si una distribución es mayor que la otra en todos los puntos.

Especificamente, podemos ver que apartir del umbral resaltado en verde en la figura ( $x = 0.05346757$ ),  $\hat{F}_{PSL}^{1000}(L_{MC}^{Giroud})$  tiene **dominancia estocástica parcial** sobre  $\hat{F}_{PSL}^{1000}(L_{MC})$  (Bawa, 1982; Gustavo, n.d.-a).

#### 7.4.5 Conclusiones sobre la Comparación de Distribuciones de PSL

Dependiendo el grado de rigurosidad provista por una comparación previa, recomendamos contemplar alguno de los consecuentes métodos presentados para comparar distribuciones de PSL. En este caso de ejemplo, se observó que la formación con Giroud tiene dominancia probabilística sobre la formación con Agüero aunque no se puede afirmar que tiene dominancia estocástica.

La comparación por momentos es una forma rápida y sencilla de comparar distribuciones de PSL, sin embargo, no siempre refleja la relación entre las distribuciones. La dominancia probabilística es una métrica intuitiva que nos permite evaluar la probabilidad de que una muestra de una distribución sea mayor que una muestra de la otra distribución. Por último, la dominancia estocástica es una relación de orden más rigurosa que nos permite evaluar si una distribución es mayor que la otra en todos los puntos.

El campo de estudio sobre la Dominancia Estocástica es amplio y complejo, en esta investigación se presentó una humilde introducción al tema y se propuso un método para evaluar la dominancia, por lo que se recomienda profundizar en el tema para una mejor comprensión a la hora de tomar decisiones basado en comparación de CDFs. Recomendamos la publicación “Stochastic Dominance: A Research Bibliography” (Bawa, 1982) que contiene al rededor de 400 referencias sobre el tema.

## 8 Player2Vec: Embeddings de Jugadores

Para poder representar a cada jugador de forma vectorial, se desarrolló el modelo de Player2Vec que permite obtener un embedding de cada jugador en un espacio de  $n$  dimensiones.

Un embedding es una representación numérica de objetos en un espacio de  $n$  dimensiones, donde propiedades o relaciones similares se preservan. En el contexto de jugadores, un embedding transforma las características de cada jugador en un vector de números, de tal manera que jugadores con comportamientos o atributos similares estén más cerca en este espacio vectorial. Esto facilita que modelos como redes neuronales aprendan patrones complejos a partir de estas representaciones compactas.

### 8.1 Definición

Player2Vec es una adaptación de Node2Vec para representar jugadores de fútbol en un espacio vectorial. En este caso, los nodos del grafo representan jugadores, y las aristas entre ellos reflejan la interacción entre los jugadores en partidos de fútbol. A partir de los datos de eventos de partidos (pases, disparos, goles, etc.), se construye un grafo donde los nodos son jugadores y las aristas representan la frecuencia de interacción entre ellos.

### 8.2 Modelado de la EPL 2012/13 como Grafo

A partir de una formación de 11 (Lineup), para un equipo (Team), en un partido (Match), se construye el grafo de la red de jugadores. Llámemos a estos  $G_{L,T,M}$  Grafo de Lineup.

Sean:

- $l \in L = \{0, 3\}$  las formaciones posibles (en la temporada 12/13 se permitían hasta 3 cambios de jugadores)
- $t \in T = \{\text{Local, Visitante}\}$  los equipos que jugaron el partido.
- $m \in M = \{1, 2, \dots, 380\}$  los partidos de la temporada 12/13 de la EPL

$$G_{L,T,M} = (V^{L,T,M}, E^{L,T,M})$$

$L$  = Número de Lineup del equipo en el partido

$T$  = Número de Equipo

$M$  = Número de Partido

$$V^{L,T,M} = \{\text{Gain}^{L,T,M}, J_1^{L,T,M}, J_2^{L,T,M}, \dots, J_{11}^{L,T,M}, \text{Loss}^{L,T,M}, \text{Shot}^{L,T,M}\}$$

$$E^{L,T,M} = \{(J_i^{L,T,M}, J_j^{L,T,M}, r(J_i^{L,T,M}, J_j^{L,T,M})) \mid i, j \in [1, 11]\}$$

$$\cup \{(\text{Gain}^{L,T,M}, J_i^{L,T,M}, r(\text{Gain}^{L,T,M}, J_i^{L,T,M})) \mid i \in [1, 11]\}$$

$$\cup \{(J_i^{L,T,M}, \text{Shot}^{L,T,M}, r(J_i^{L,T,M}, \text{Shot}^{L,T,M})) \mid i \in [1, 11]\}$$

$$\cup \{(J_i^{L,T,M}, \text{Loss}^{L,T,M}, r(J_i^{L,T,M}, \text{Loss}^{L,T,M})) \mid i \in [1, 11]\}$$

Donde cada  $J_i^{L,T,M} \mid i \in [1, 11]$  es un nodo que representa a un jugador en el lineup  $L$  del equipo  $T$  en el partido  $M$ .  $\text{Gain}^{L,T,M}$  es el nodo que representa la ganancia del balón,  $\text{Loss}^{L,T,M}$  la pérdida del balón y  $\text{Shot}^{L,T,M}$  el disparo al arco en el lineup  $L$  del equipo  $T$  en el partido  $M$ .

En la figura se visualiza un ejemplo de un grafo de lineup  $G^{L,T,M}$  genérico con los ejes  $r(J_1^{L,T,M}, U)$  resaltados.

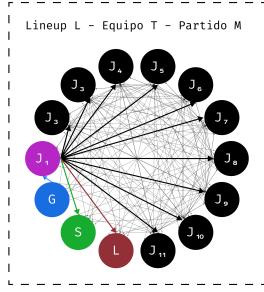


Figure 20: Grafo de Lineup

Luego sean: -  $J_i \mid i \in [0, 522]$  los jugadores reales de la temporada 2012/13 de la EPL

Se construye el grafo de la red de jugadores  $G_{EPL-12/13}$  como la unión de todos los grafos de lineup  $G^{L,T,M}$ .

$$\begin{aligned}
G_{\text{Full}} = (V, E) &= \bigcup_{L,T,M} G^{L,T,M} \\
V &= \{J_1, J_2, \dots, J_{522}, \text{Gain}, \text{Loss}, \text{Shot}\} \\
&\cup \bigcup_{L,T,M} \{J_1^{L,T,M}, J_2^{L,T,M}, \dots, J_{11}^{L,T,M}, G^{L,T,M}, L^{L,T,M}, S^{L,T,M}\} \\
E &= \bigcup_{L,T,M} E^{L,T,M} \\
&\cup \{(J_i, J_j^{L,T,M}, r(J_i, J_j^{L,T,M})) \mid i \in [0, 522], j \in [1, 11], L, T, M\} \\
&\cup \{(Gain, Gain^{L,T,M}, 1) \mid L, T, M\} \\
&\cup \{(Loss^{L,T,M}, Loss, 1) \mid L, T, M\} \\
&\cup \{(Shot^{L,T,M}, Shot, 1) \mid L, T, M\}
\end{aligned}$$

El ratio de transición  $r(J_i, J_i^{L,T,M})$  es el tiempo jugado por el Jugador  $J_i$  en el lineup  $L$  del equipo  $T$  en el partido  $M$  sobre el tiempo total jugado por el Jugador  $J_i$

$$r(J_i, J_i^{L,T,M}) = \frac{\text{Time Played}_{J_i^{L,T,M}}}{\text{Time Played}_{J_i}}$$

La siguiente figura es una visualización de una instancia de un Equipo en un Partido con sus lineups. En este caso el equipo hizo dos cambios en el partido ( $J_4$  por  $J_{12}$  y  $J_2$  por  $J_{13}$ ). Se puede observar como los jugadores reales  $J_4$  y  $J_{12}$  se encuentran representados por el mismo nodo  $J_4^{L,T,M}$  y lo mismo para  $J_2$  y  $J_{13}$  con  $J_2^{L,T,M}$  para sus respectivos lineups. El resto de los nodos de jugadores reales mantienen su identidad en los grafos de lineups.

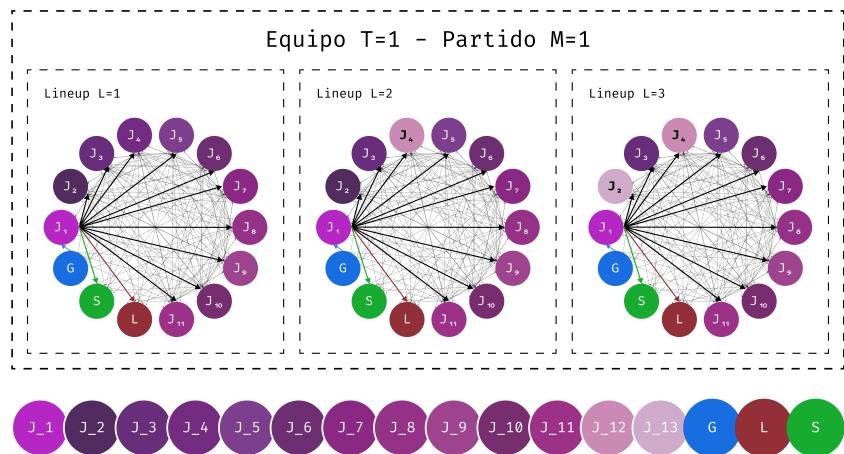


Figure 21: Grafo de Jugadores

El grafo resultante de la composición de todos los grafos de lineup  $G_{Full}$  se puede comprender mejor en la siguiente visualización:

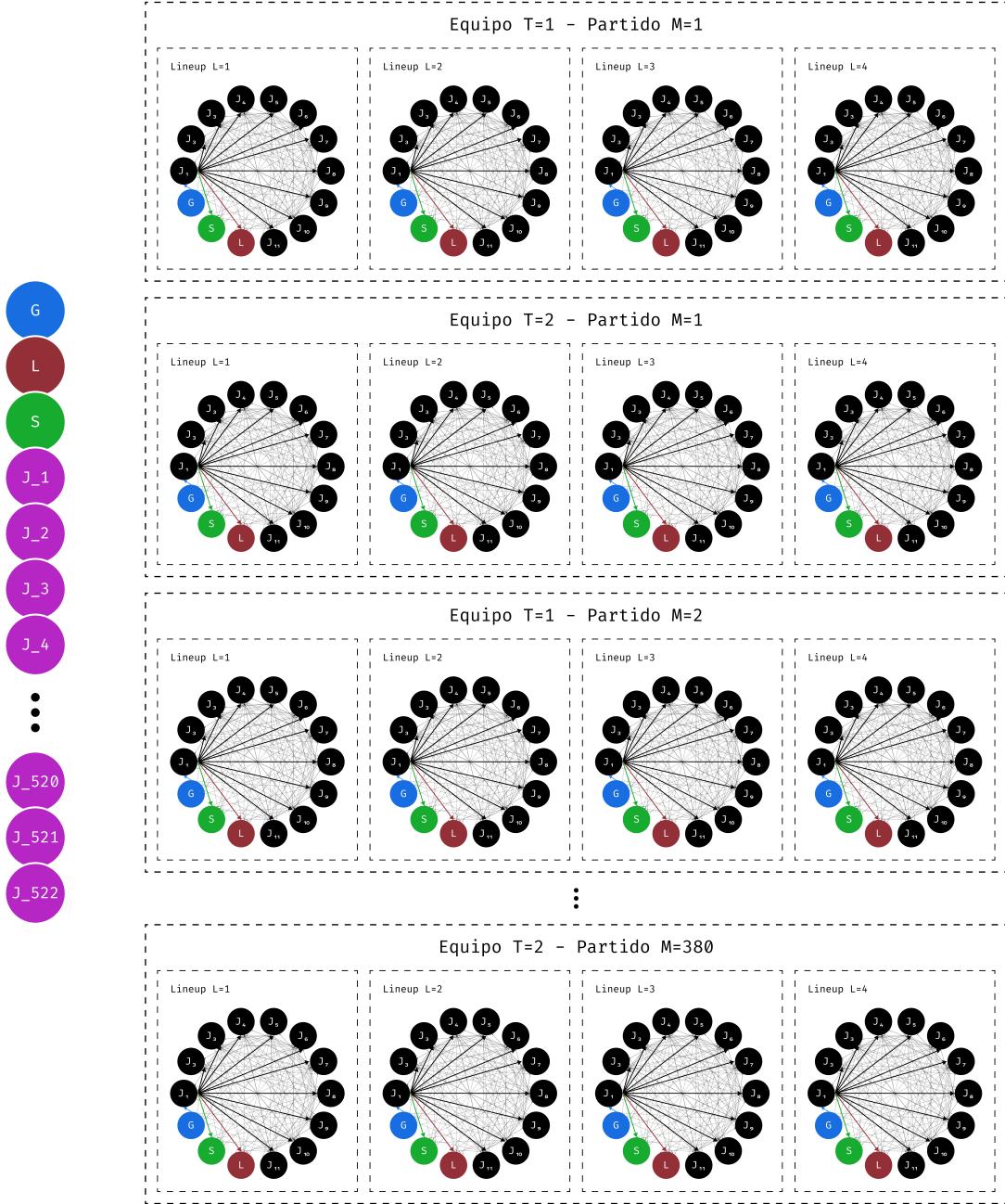


Figure 22: Grafo de Jugadores Completo

Donde al igual que en la figura anterior, los nodos de jugadores reales se encuentran representados por los nodos de los lineups en los que participaron.

El algoritmo en concreto para construir el grafo de la red de jugadores  $G_{Full}$  es el siguiente:

```

Input: Datos de eventos de partidos de la temporada 2012/13 de la EPL
Output: Grafo de la red de jugadores  $G_{\text{Full}}$ 
1  $V \leftarrow \{J_1, J_2, \dots, J_{522}, \text{Gain}, \text{Loss}, \text{Shot}\};$ 
2  $E \leftarrow \emptyset;$ 
3 for partido  $M$  do
4   for lineup  $L$  del partido  $M$  do
5     for jugador  $J_i$  en el lineup  $L$  do
6        $V \leftarrow V \cup \{J_i^{L,T,M}\};$ 
7        $E \leftarrow E \cup \{(J_i, J_i^{L,T,M}, r(J_i, J_i^{L,T,M}))\};$ 
8     end
9      $V \leftarrow V \cup \{\text{Gain}^{L,T,M}, \text{Loss}^{L,T,M}, \text{Shot}^{L,T,M}\};$ 
10     $E \leftarrow E \cup \{(\text{Gain}, \text{Gain}^{L,T,M}, 1), (\text{Loss}^{L,T,M}, \text{Loss}, 1), (\text{Shot}^{L,T,M}, \text{Shot}, 1)\};$ 
11  end
12 end

```

**Algorithm 3:** Construcción del Grafo de Jugadores

### 8.3 Implementación

A partir de calcular las matrices de ratios  $R^{L,T,M}$  para cada lineup  $L$  del equipo  $T$  en el partido  $M$  generamos el grafo dirigido  $G^{L,T,M}$  haciendo uso de la librería `NetworkX` en Python para luego componerlos en  $G_{\text{Full}}$ , el grafo resultante contiene 37521 nodos y 47338 aristas.

Para obtener los embeddings de los jugadores, se utilizó la librería `node2vec` en Python, que implementa el algoritmo homónimo. Se configuró el modelo con una longitud de caminata de 16 nodos, 200 caminatas y un tamaño de ventana de 12 nodos. Se entrenaron 2 modelos de embeddings, uno con 64 dimensiones para utilizar en modelos de Deep Learning y otro con 3 dimensiones.

Para cada uno de los 37521 nodos se obtuvo un embedding, de los cuales nos quedamos solo con los 522 embeddings de los jugadores reales, estos finalmente son la representación vectorial de cada jugador en el espacio de embeddings.

Este modelo hace uso de Node2Vec, que es en sí una adaptación de Word2Vec, una técnica de NLP que permite representar palabras en un espacio vectorial (Grover & Leskovec, 2016; Mikolov et al., 2013).

Node2Vec es un algoritmo que aprende representaciones vectoriales (embeddings) para nodos en un grafo, preservando tanto las relaciones locales como las globales entre ellos. Utiliza técnicas de random walks para capturar el contexto de cada nodo, balanceando entre explorar nodos cercanos y lejanos. Estos embeddings son útiles para tareas de machine learning sobre grafos, ya que capturan de forma eficiente las interacciones entre nodos en el grafo.

En el caso de Player2Vec, los  $k$  random walks resultantes son una secuencia de jugadores y/o estados de juego en un partido de fútbol (Ganancia, Pérdida, Disparo). A modo ilustrativo los siguientes son posibles random walks obtenidos del grafo de la EPL 2012/13:

$$\begin{aligned}
&\text{Random Walk 1: Gain} \rightarrow \text{Gain}^{L,T,M} \rightarrow J_1^{L,T,M} \rightarrow J_7^{L,T,M} \rightarrow \dots \rightarrow \text{Shot}^{L,T,M} \rightarrow \text{Shot} \\
&\text{Random Walk 2: } J_{93} \rightarrow J_{93}^{L,T,M} \rightarrow J_{15}^{L,T,M} \rightarrow J_{21}^{L,T,M} \rightarrow \text{Loss}^{L,T,M} \rightarrow \text{Loss} \\
&\vdots \\
&\text{Random Walk } k : J_{12} \rightarrow J_{12}^{L,T,M} \rightarrow J_{13}^{L,T,M} \rightarrow J_{33}^{L,T,M} \rightarrow \text{Shot}^{L,T,M} \rightarrow \text{Shot}
\end{aligned}$$

La cantidad de random walks  $k$  así como los otros hiperparametros del modelo de Node2Vec fueron seleccionados de forma empírica observando el resultado de los embeddings obtenidos.

### 8.4 Visualización y Exploración de los Embeddings

Para comenzar a explorar el espacio vectorial generado por Player2Vec, se ajustó un modelo inicialmente a partir siguientes hiperparametros:

- Dimensión de embeddings: 3

- Longitud de caminata: 16 nodos
- Número de caminatas: 200
- Tamaño de ventana: 12 nodos

Se entrenó el modelo y se obtuvieron los embeddings de los 522 jugadores de la temporada 2012/13 de la EPL. La siguiente visualización muestra los embeddings de los jugadores en un espacio de 3 dimensiones, el color corresponde al equipo en el que juega el jugador.

Embeddings de 3 dimensiones de los jugadores - Player2Vec

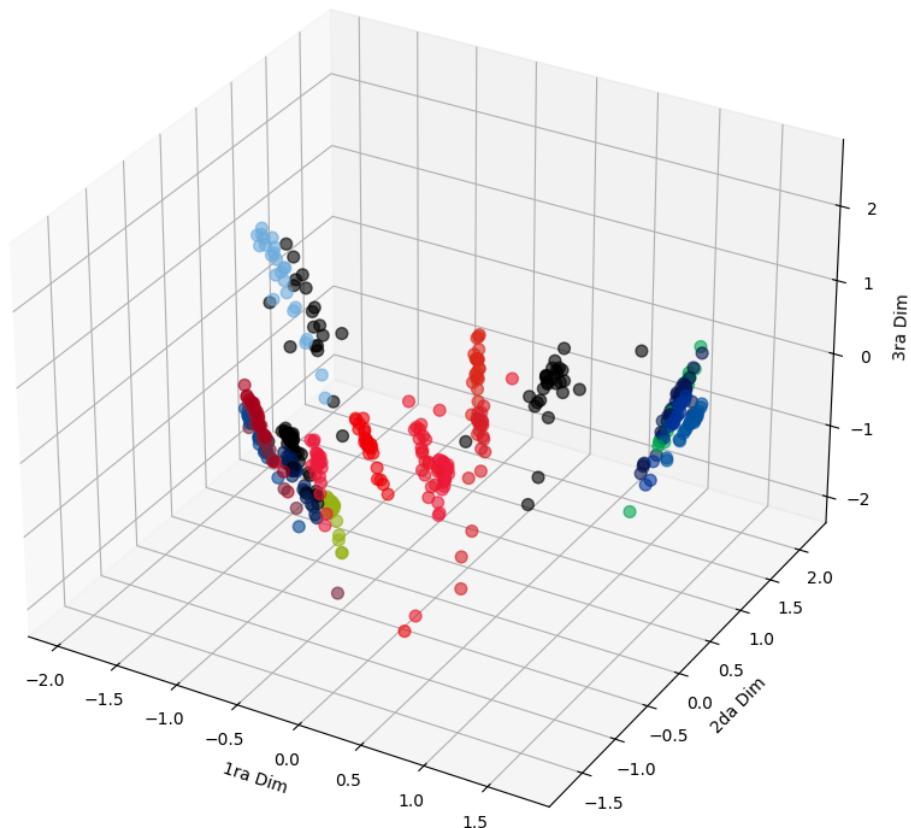


Figure 23: Embeddings de Jugadores en 3D

Para poder visualizar de forma más clara los embeddings de los jugadores, se realizó un PCA para reducir la dimensionalidad de los embeddings a 2 dimensiones. La siguiente visualización muestra los embeddings de los jugadores en un espacio de 2 dimensiones, el color corresponde al equipo en el que juega el jugador.

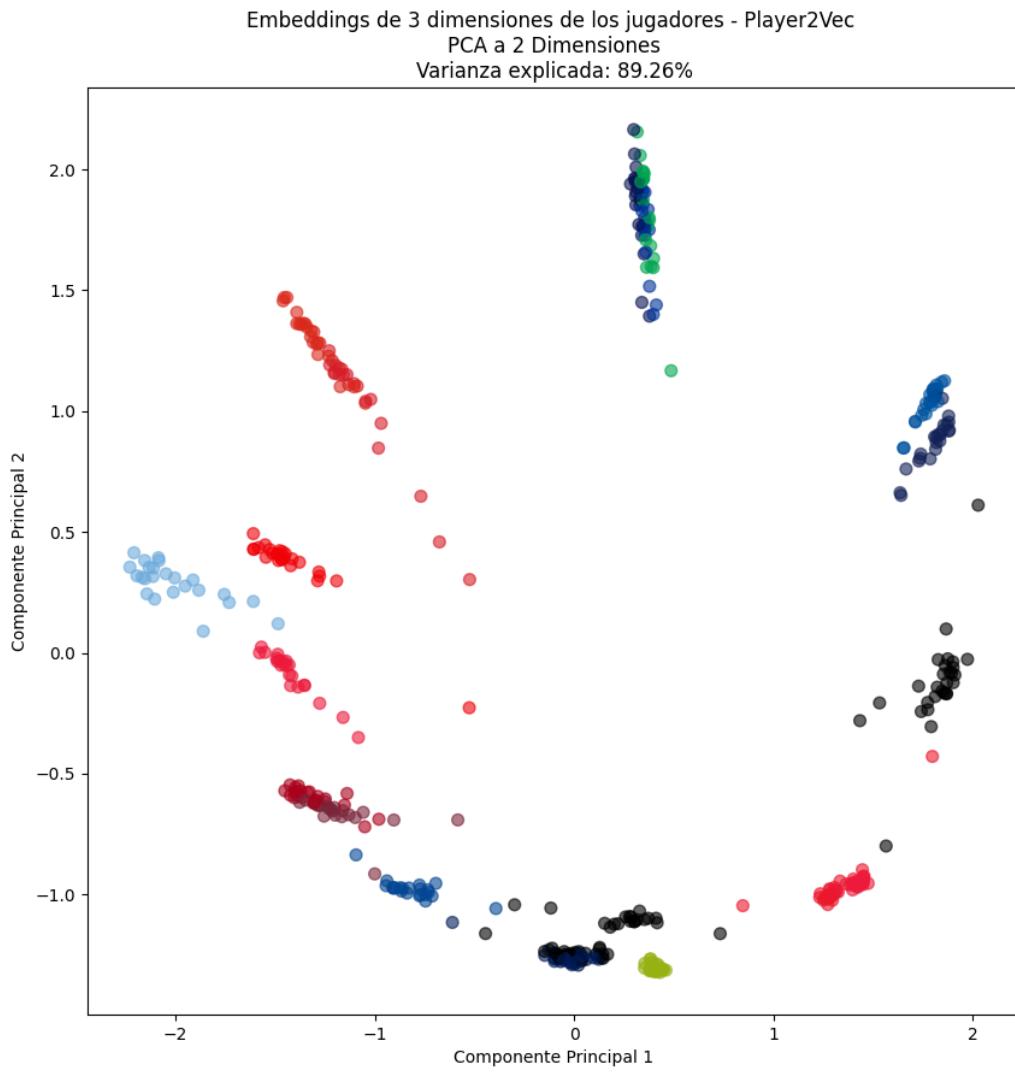


Figure 24: Embeddings de Jugadores en 3D - PCA a 2D

En la figura de los componentes principales se observa como los jugadores de un mismo equipo se encuentran cercanos en el espacio vectorial, lo que indica que los embeddings resultantes de este modelo capturan las relaciones entre los jugadores de un mismo equipo.

Ademas se observa como en este espacio las direcciones en las que se representan a los equipos divergen de forma clara, lo que indica que los embeddings capturan únicamente las diferencias entre los equipos y no las similitudes. Buscan cierta ortogonalidad entre los equipos que no logra existir en este espacio de 3 dimensiones.

Para explotar aún mas las relaciones a aprender por el modelo, se ajustó un segundo modelo con las siguientes características:

- Dimensión de embeddings: 64
- Longitud de caminata: 40 nodos
- Número de caminatas: 500
- Tamaño de ventana: 30 nodos

Luego para explorar los embeddings resultantes se realizó nuevamente un análisis de componentes principales para reducir la dimensionalidad de los embeddings a 2 dimensiones.

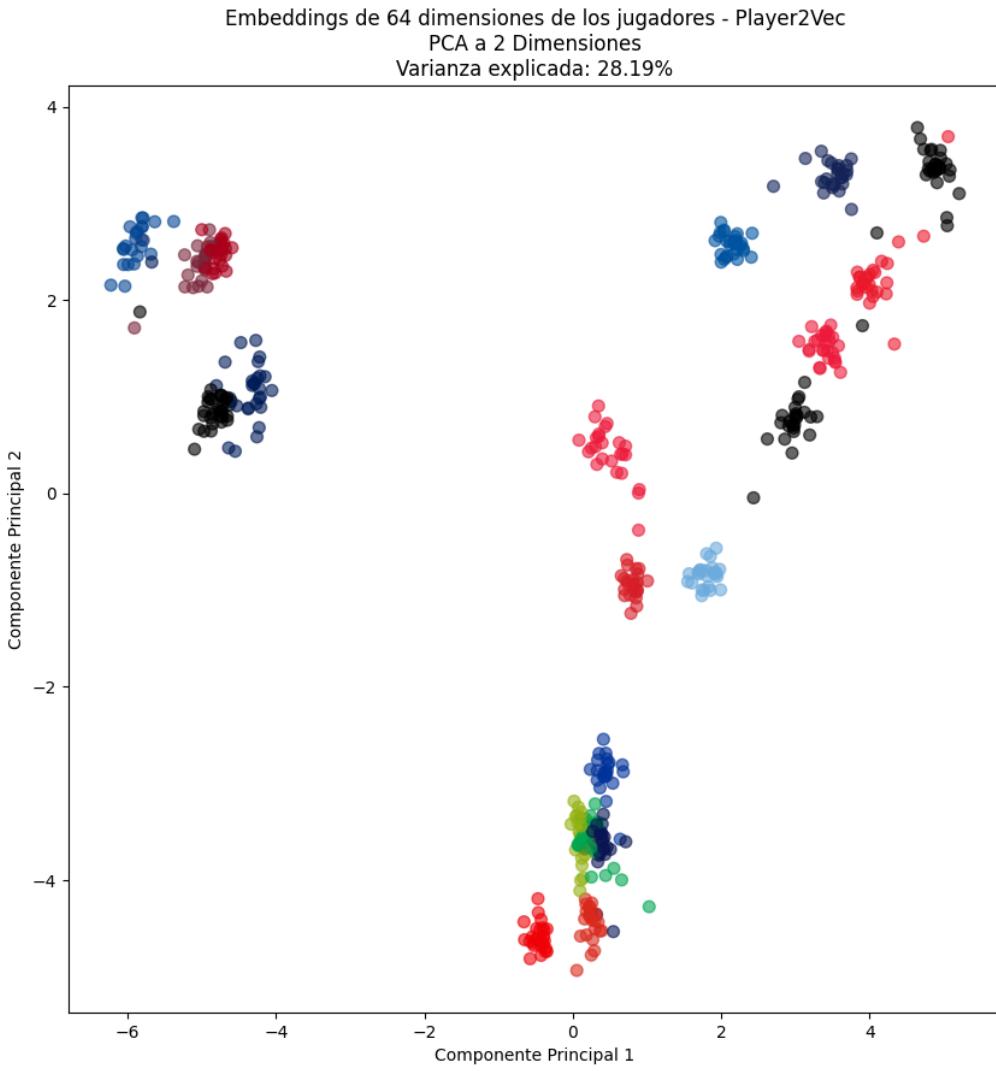


Figure 25: Embeddings de Jugadores en 64D - PCA a 2D

En esta figura resultante se puede observar como la direccionalidad de los equipos desaparece pero se mantienen las relaciones entre los jugadores de un mismo equipo.

## 8.5 Potencial de Player2Vec

Con el modelo planteado de Grafos de Lineups por Equipos y Partidos se puede representar no solo una temporada de una liga, como es nuestro caso, sino que se puede extender a múltiples temporadas y ligas. Esto permitiría poder comparar jugadores de distintas ligas y temporadas, y poder evaluar el rendimiento de un jugador en distintos contextos.

Otra cuestión considerada para expandir es ademas de tener un nodo general por jugador conectado a sus instancias en cada lineup, se podría tener un nodo que represente a un jugador en un equipo, de forma tal que el jugador real esté conectado a su nodo “Jugador en Equipo” y este nodo a su vez conectado a “Jugador en Lineup de Partido de Equipo”. Esto permitiría poder evaluar el rendimiento de un jugador en un equipo en particular y como este se comporta en distintos contextos.

En el paper de *Soccer Networks* donde se plantea el PSL definen una serie de coeficientes  $h$ ,  $a$ ,  $\omega$ , como la performance de un equipo al jugar de local, al jugar de visitante, y la performance ponderada de todos los

otros equipos al jugar de visitante respectivamente. Se podría escalar por ejemplo los ratios de transición entre jugadores y el estado de disparo al arco en función de estos coeficientes para obtener una mejor representación de la performance de un jugador en un partido en particular.

## 9 Modelo predictivo de Distribuciones de $r(U, V)$

El trabajo de Player2Vec nos permite obtener embeddings de jugadores que capturan las relaciones entre ellos en un espacio vectorial. A partir de estos embeddings, se propone un modelo predictivo desde las distribuciones de  $r(U, V)$ .

### 9.1 Definición

Dado un jugador  $J_i$ , se obtiene su embedding  $E(J_i)$  a partir del modelo de Player2Vec. Para este  $J_i$ , se busca predecir los estadísticos Media y Varianza de las distribuciones de ratios de transición de  $J_i$ .

- $r(\text{Gain}, J_i)$
- $r(J_i, \text{Shot})$
- $r(J_i, \text{Loss})$
- $r(J_i, J_j)$
- $r(J_j, J_i)$

Sean  $\mu_{\text{Gain}, J_i}$  y  $\sigma_{\text{Gain}, J_i}$  la media y la varianza de la distribución de  $r(\text{Gain}, J_i)$  respectivamente. Análogamente para las otras distribuciones.

Asumiendo normalidad, podemos decir que  $r(U, V) \sim \mathcal{N}(\mu_{U,V}, \sigma_{U,V})$ .

### 9.2 Modelo

El modelo planteado es una Red Neuronal de la forma:

$$f(E(J_i)) = (\mu_{\text{Gain}, J_i}, \sigma_{\text{Gain}, J_i}, \mu_{J_i, \text{Shot}}, \sigma_{J_i, \text{Shot}}, \mu_{J_i, \text{Loss}}, \sigma_{J_i, \text{Loss}}, \mu_{J_i, J_j}, \sigma_{J_i, J_j}, \mu_{J_j, J_i}, \sigma_{J_j, J_i})$$

La función de pérdida a minimizar es una ponderación de la Divergencia de Jensen-Shannon (JSD) entre la distribución real y la predicha para cada estadístico.

$$JSD(p||q) = \frac{1}{2}D_{KL}(p||m) + \frac{1}{2}D_{KL}(q||m)$$

Donde  $D_{KL}(p||q)$  es la divergencia de Kullback-Leibler entre las distribuciones  $p$  y  $q$  y  $m = \frac{p+q}{2}$ .

### 9.3 Datos

Para entrenar el modelo, se utilizó un dataset de eventos de partidos de la temporada 2012/13 de la EPL. Se separó la temporada en dos mitades (190 partidos cada una). Con la primera mitad, se construyó un grafo de la red de jugadores  $G_{\text{Half}}$  y se obtuvieron los embeddings de los jugadores con Player2Vec (`dimensions=64`, `window=30`, `num_walks=500`, `walk_length=40`). Con la segunda mitad, se obtuvieron las distribuciones de ratios de transición de los jugadores.

Luego, de la segunda mitad se obtuvieron las medias y varianzas de las distribuciones de ratios de transición de los jugadores. Un 80% de los datos se utilizó para entrenar el modelo y un 20% para test.

### 9.4 Implementación

El modelo se implementó en PyTorch. Se utilizó una red neuronal con la siguiente arquitectura:

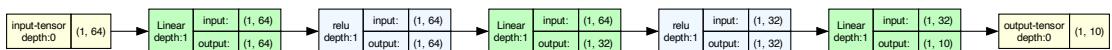


Figure 26: Arquitectura del Modelo Base

La función de pérdida utilizada fue la Divergencia de Jensen-Shannon (JSD) entre las distribuciones reales y predichas ponderando las 5 distribuciones a estimar. Se utilizó la implementación de Divergencia

de Kullback-Leibler de PyTorch `nn.KLDivLoss` en la implementación del modulo JSD hayado en el foro de PyTorch(PyTorch Forums, 2022) y este luego se utilizó para la función de pérdida en el entrenamiento del modelo.

## 9.5 Entrenamiento

Para entrenar el modelo, se utilizó el optimizador SGD (Stochastic Gradient Descent) con una tasa de aprendizaje (lr) de 0.005, un momentum de 0.9 y un weight decay de 0.0005 para prevenir el sobreajuste.

Además, se utilizó un scheduler de tasa de aprendizaje (scheduler) con una estrategia de StepLR, que reduce la tasa de aprendizaje en un factor de 0.1 cada 100 épocas. El entrenamiento se llevó a cabo durante un máximo de 10,000 épocas, con un mecanismo de early stopping para detener el entrenamiento si no se observaban mejoras en el rendimiento del modelo en el conjunto de validación durante un número determinado de épocas consecutivas.

## 9.6 Resultados iniciales

El modelo base se entrenó con los hiperparámetros y arquitectura mencionados anteriormente, su entrenamiento finalizó en el epoch 5244.

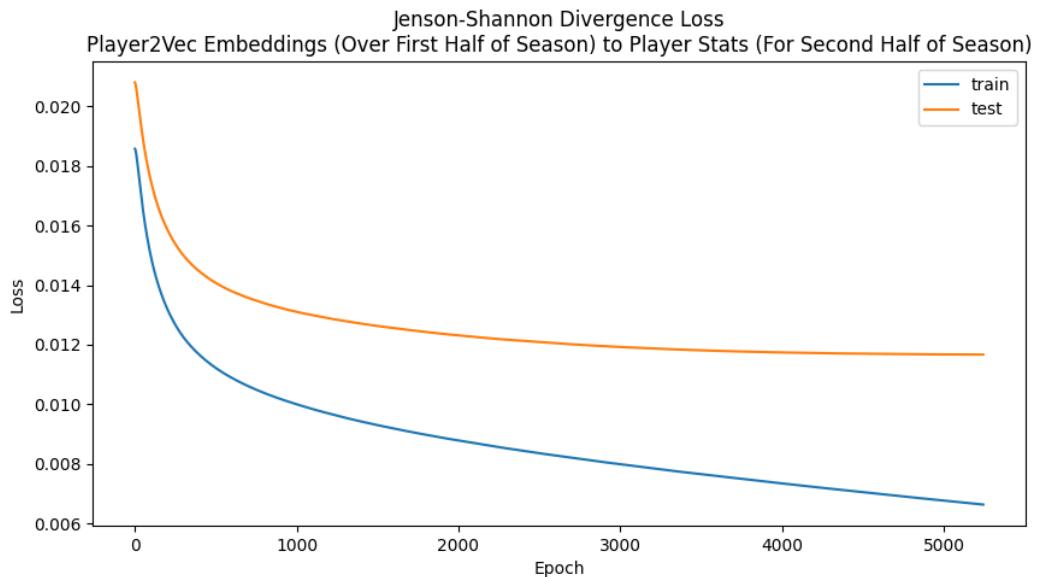


Figure 27: Resultados del Modelo Base

Table 2: Resultados del Modelo Base

Métrica	Entrenamiento	Validación
Loss	0.00664	0.01167

El modelo base obtuvo un rendimiento aceptable en el conjunto de entrenamiento, con una pérdida de 0.00664, pero casi el doble en el conjunto de validación, con una pérdida de 0.01167. Esto puede ser un indicador de sobreajuste, por lo que se procedió a realizar un proceso de tuning de hiperparámetros y arquitectura con validación cruzada.

## 9.7 Tuning de Hiperparámetros y Arquitectura con Validación Cruzada

Para mejorar el rendimiento del modelo, se realizó un proceso de tuning de hiperparámetros y arquitectura con validación cruzada. Se evaluaron distintas combinaciones de hiperparámetros y arquitecturas de red neuronal, y se seleccionó la que mejor rendimiento presentó en el conjunto de validación.

Del 80% de los datos de entrenamiento, se separó un 20% para validación para hacer holdout CV.

Con la librería **Hyperopt** (Bergstra et al., 2015), se realizó una búsqueda de hiperparámetros bayesina. Se evaluaron distintas combinaciones de hiperparámetros, y se seleccionó la que mejor rendimiento presentó en el conjunto de validación.

## 9.8 Espacio de Hiperparámetros

El espacio de hiperparámetros que se exploró fue el siguiente:

- **Tasa de aprendizaje (lr):** {0.1, 0.01, 0.001}
- **Momentum:** {0.8, 0.9, 0.99}
- **Weight decay:** {0.0005, 0.0001}
- **Optimizador:**
  - SGD con momentum
  - Adam
  - AdamW
  - RMSprop con momentum
- **Scheduler:**
  - StepLR con step\_size {50, 100, 200} y gamma {0.1, 0.05, 0.01}
  - MultiStepLR con milestones {50, 100, 200} y gamma {0.1, 0.05, 0.01}
  - ExponentialLR con gamma {0.1, 0.05, 0.01}
  - CosineAnnealingLR con T\_max {50, 100, 200}
- **Early stopping:** {True}
- **Early stopping window:** {10, 20, 50, 100}

Además se implementó una clase de Red Neuronal paramétrica para poder explorar distintas arquitecturas de red neuronal.

## 9.9 Resultados del Tuning de Hiperparámetros

La búsqueda de hiperparámetros se realizó con 1000 iteraciones. La mejor combinación de hiperparámetros y arquitectura de red neuronal encontrada fue la siguiente:

- Tasa de aprendizaje: 0.01
- Momentum: 0.99
- Weight decay: 0.0005
- Optimizador: AdamW
- Scheduler: CosineAnnealingLR
- Early stopping window: 50
- Arquitectura de red neuronal:
  - Activación: LeakyReLU
  - Dropout: 0.3
  - Número de capas: 6
  - Tamaños de capas ocultas: (1024, 1024, 32, 128, 128, 1024)



Figure 28: Arquitectura del Modelo Tuned

Table 3: Resultados del Modelo

	Train loss	Val loss	Test loss
0	0.003932	0.004977	0.008988

El modelo final seleccionado obtuvo un rendimiento aceptable en el conjunto de validación, con una pérdida de 0.004977, aún mejor que la perdida de Train del modelo base. Se procedió a evaluar el modelo en el conjunto de testeo y se obtuvo una pérdida de 0.008988.

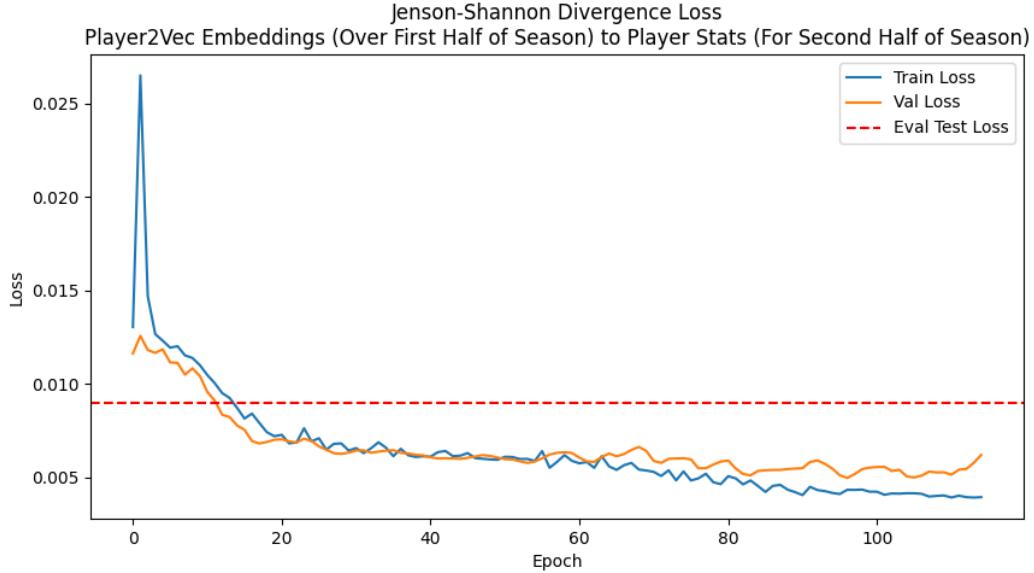


Figure 29: Resultados del Modelo Tuned

El modelo resultante logró generalizar mejor, posiblemente estemos presenciando un caso de overfitting del conjunto de validación, ya que la pérdida en el conjunto de testeo es mayor que en el de validación, sin embargo los resultados obtenidos son satisfactorios.

## 9.10 Hardware y Tiempos de Entrenamiento

El entrenamiento del modelo base se realizó en una máquina con las siguientes características:

Macbook Pro M1 2020 - Procesador: Apple M1 - Memoria RAM: 16 GB - GPU: 8-core GPU - 16-core Neural Engine - OS: macOS Sequoia 15.0.1 (24A348)

- Python 3.10.14
- PyTorch 2.2.2 MPS (Metal Performance Shaders)

El entrenamiento del modelo final tomó tan solo 30 segundos a lo largo de los 115 epochs que duró el entrenamiento por el mecanismo de early stopping.

La búsqueda de hiperparametros de 1000 iteraciones tomó al rededor de 5hs en la misma máquina.

## 10 Validación del Modelo de Distribuciones

Para evaluar la capacidad de nuestro modelo en escenarios reales, desarrollamos un caso de estudio basado en las transferencias más importantes en términos de precio y relevancia mediática de los próximos dos mercados de fichajes dentro de la Premier League. Nos limitamos a transferencias internas de esta liga debido a la falta de datos disponibles sobre otras competiciones. Este ejercicio busca medir qué tan bien nuestras métricas y simulaciones pueden justificar o desaconsejar dichas transferencias en función de su impacto en el rendimiento del equipo.

En este apartado, profundizaremos en el análisis de dos casos específicos: la transferencia de Danny Welbeck al Arsenal y la de James Milner al Liverpool. Finalmente, presentaremos una tabla resumen con seis transferencias clave, acompañadas de las recomendaciones de nuestro modelo.

Para comparar las distribuciones de PSL de los jugadores involucrados en las transferencias, utilizamos las métricas de comparación de distribuciones presentadas anteriormente: comparación de momentos, dominancia probabilística y dominancia estocástica.

### 10.1 Caso de Estudio 1: Danny Welbeck al Arsenal

Para analizar la transferencia de Danny Welbeck (delantero) al Arsenal, evaluamos el impacto en el PSL del equipo simulando su inclusión en lugar de los principales delanteros titulares. Para ello, consideramos dos escenarios: en el primero, Welbeck reemplaza a Theo Walcott creando la formación  $L_{AR1}^{Welbeck}$  cuya distribución es  $X_{L_{AR1}^{Welbeck}} \sim \hat{f}_{PSL}^{1000}(L_{AR1}^{Welbeck})$ . Y en el segundo, a Olivier Giroud creando la formación  $L_{AR2}^{Welbeck}$  cuya distribución es  $X_{L_{AR2}^{Welbeck}} \sim \hat{f}_{PSL}^{1000}(L_{AR2}^{Welbeck})$ .

Los resultados obtenidos son los siguientes:

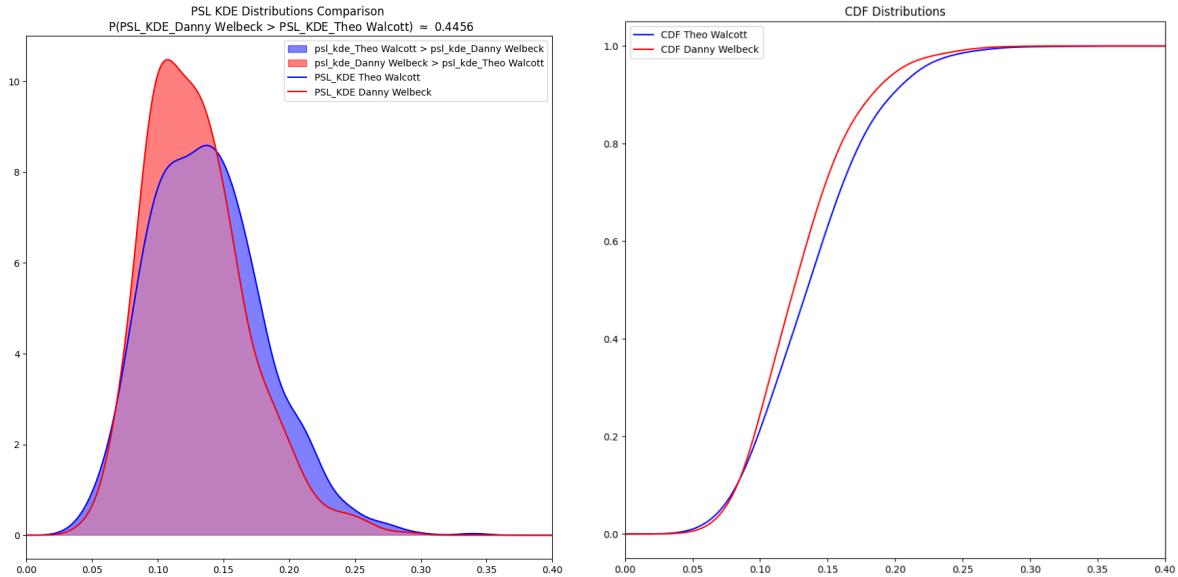


Figure 30: Comparación de CDFs y PDFs de Welbeck y Walcott

Table 4: Comparación de momentos de  $X_{L_{AR1}^{Welbeck}}$  y  $X_{L_{AR1}}$

Jugador	Media	Varianza	Desvio Estándar	Skewness	Kurtosis
Theo Walcott	0.137735	0.043914	0.001928	0.547502	0.421810
Danny Welbeck	0.129325	0.038815	0.001507	0.761120	0.773602

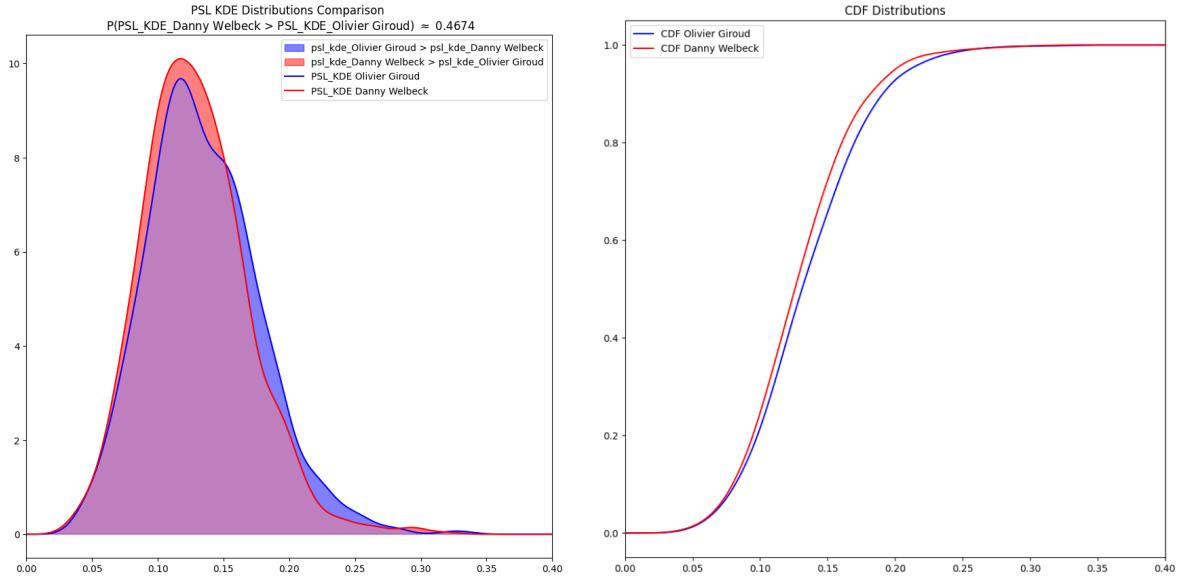


Figure 31: Comparación de CDFs y PDFs de Welbeck y Giroud

Table 5: Comparación de momentos de  $X_{L_{AR2}^{Welbeck}}$  y  $X_{L_{AR2}}$

Jugador	Media	Varianza	Desvio Estándar	Skewness	Kurtosis
Olivier Giroud	0.134801	0.042535	0.001809	0.587343	0.777267
Danny Welbeck	0.128931	0.040130	0.001610	0.700550	1.325063

Como se puede observar, la distribución de la formación con Walcott,  $X_{L_{AR1}}$ , presenta una mayor media en PSL en comparación con  $X_{L_{AR1}^{Welbeck}}$ . Además, al observar “a ojo” la PDF de  $X_{L_{AR1}}$ , notamos un mayor sesgo hacia la derecha, lo que indica que la formación con Walcott posee un PSL superior al de la formación con Welbeck en la mayoría de los casos.

Dado que  $P(X_{L_{AR1}^{Welbeck}} > X_{L_{AR1}}) = 0.44$ , se sigue que  $P(X_{L_{AR1}} > X_{L_{AR1}^{Welbeck}}) = 0.56$ , lo cual indica que la formación con Walcott tiene dominancia probabilística sobre la formación con Welbeck.

Asimismo, podemos concluir que la formación con Walcott domina estocásticamente a la formación con Welbeck, ya que la CDF de  $X_{L_{AR1}}$  es mayor que la de  $X_{L_{AR1}^{Welbeck}}$  en todos los puntos a partir del umbral de 0.07.

Todas estas métricas se repiten a favor de la formación con Giroud,  $X_{L_{AR2}}$ , en comparación con  $X_{L_{AR2}^{Welbeck}}$ .

Por lo tanto, podemos concluir que, según nuestro modelo, la transferencia de Danny Welbeck al Arsenal no es recomendable porque su inclusión en lugar de Theo Walcott o Olivier Giroud disminuiría el PSL del equipo y empeoraría la performance del mismo.

## 10.2 Caso de Estudio 2: James Milner al Liverpool

Para analizar la transferencia de James Milner(Mediocampista) utilizamos la misma metodología que en el caso anterior. Evaluamos el impacto en el PSL del equipo simulando su inclusión en lugar de los principales mediocampistas titulares. Para ello, consideramos dos escenarios: en el primero, Milner reemplaza a Steven Gerrard creando la formación  $L_{LIV1}^{Milner}$  cuya distribución es  $X_{L_{LIV1}^{Milner}} \sim \hat{f}_{PSL}^{1000}(L_{LIV1}^{Milner})$ . Y en el segundo, a Stewart Downing creando la formación  $L_{LIV2}^{Milner}$  cuya distribución es  $X_{L_{LIV2}^{Milner}} \sim \hat{f}_{PSL}^{1000}(L_{LIV2}^{Milner})$ .

Los resultados obtenidos luego de la simulación son los siguientes:

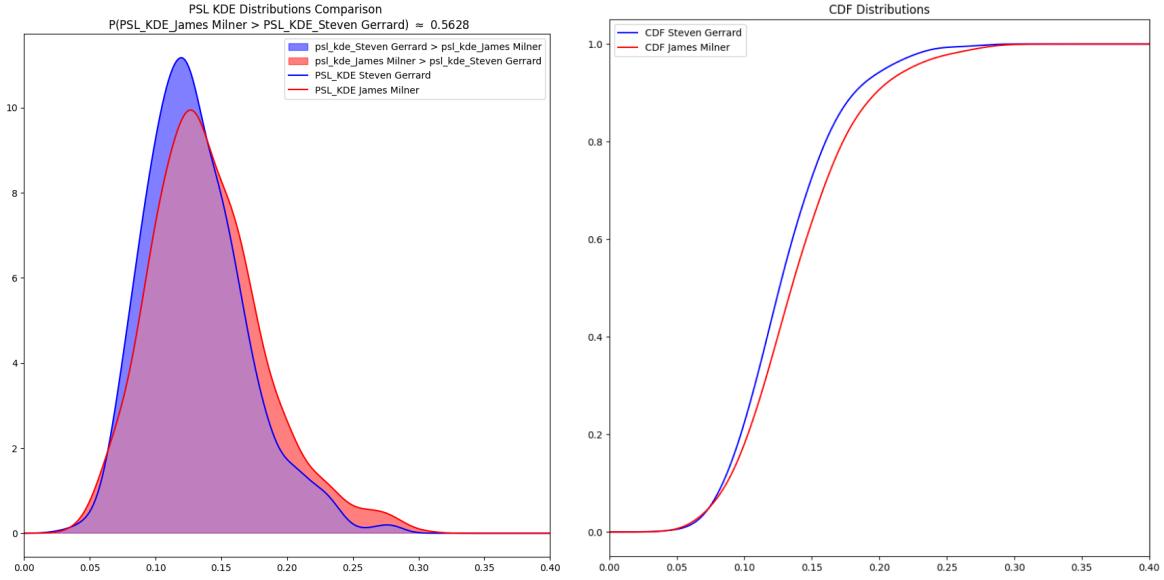


Figure 32: Comparación de CDFs y PDFs de Milner y Gerrard

Table 6: Comparación de momentos de  $X_{L_{LIV1}^{Milner}}$  y  $X_{L_{LIV1}}$

Jugador	Media	Varianza	Desvio Estándar	Skewness	Kurtosis
Steven Gerrard	0.130349	0.038290	0.001466	0.748219	0.862186
James Milner	0.139559	0.043333	0.001878	0.703642	0.712261

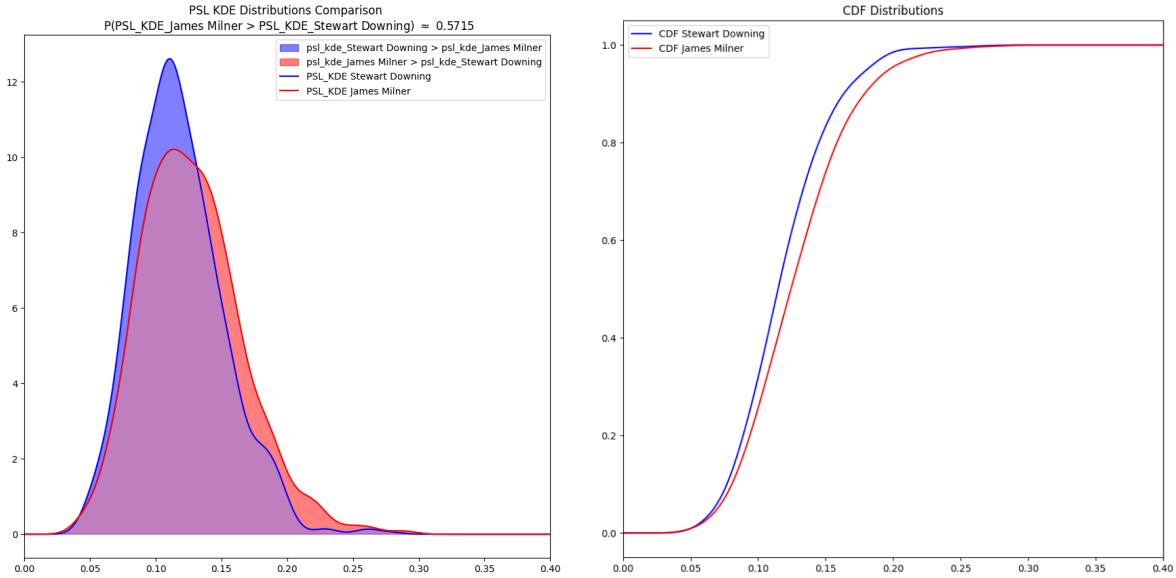


Figure 33: Comparación de CDFs y PDFs de Milner y Downing

Table 7: Comparación de momentos de  $X_{L_{LIV2}^{Milner}}$  y  $X_{L_{LIV2}}$

Jugador	Media	Varianza	Desvio Estándar	Skewness	Kurtosis
Stewart Downing	0.117987	0.033518	0.001123	0.696171	1.087000
James Milner	0.127622	0.038303	0.001467	0.651483	0.768572

Como se puede observar, ambas distribuciones con Milner,  $X_{L_{LIV1}^{Milner}}$  y  $X_{L_{LIV2}^{Milner}}$ , presentan una mayor media en PSL en comparación con  $X_{L_{LIV1}}$  y  $X_{L_{LIV2}}$  respectivamente. Además, al observar “a ojo” las PDFs de  $X_{L_{LIV1}^{Milner}}$  y  $X_{L_{LIV2}^{Milner}}$ , notamos un mayor sesgo hacia la derecha que las originales, lo que indica que las formaciones con Milner poseen un PSL superior en la mayoría de los casos.

Dado que  $P(X_{L_{LIV1}^{Milner}} > X_{L_{LIV1}}) = 0.56$  y  $P(X_{L_{LIV2}^{Milner}} > X_{L_{LIV2}}) = 0.57$ , implica que las formaciones con Milner tienen dominancia probabilística sobre las formaciones originales.

También notamos que las CDFs de  $X_{L_{LIV1}^{Milner}}$  y  $X_{L_{LIV2}^{Milner}}$  son mayores que las originales en todos los puntos a partir del umbral de 0.06, lo que indica que las formaciones con Milner dominan estocásticamente a las formaciones originales.

Por lo tanto, podemos concluir que, según nuestro modelo, la transferencia de James Milner al Liverpool es recomendable porque su inclusión en lugar de Steven Gerrard o Stewart Downing mejoraría el PSL del equipo y aumentaría la performance del mismo.

### 10.3 Recomendaciones de Transferencias Clave

En la tabla a continuación se presentan las seis transferencias más importantes dentro de la Premier League en la temporada 13/14, evaluadas según nuestro modelo. Cada transferencia fue analizada utilizando las distribuciones de PSL del jugador y simulando su impacto en el rendimiento del equipo destino. Las recomendaciones se clasificaron en:

- **Recomendable:** La transferencia mejora el PSL del equipo destino.
- **No Recomendable:** La transferencia empeora el PSL del equipo destino.
- **Indeciso:** La transferencia no tiene un impacto significativo en el PSL del equipo destino.

Table 8: Recomendaciones de Transferencias Clave

Jugador	Posición	Equipo Origen	Equipo Destino	Recomendación
James Milner	Mediocampista	Manchester City	Liverpool	Recomendable
Andros Townsend	Mediocampista	Tottenham	Newcastle	Recomendable
Juan Mata	Mediocampista	Chelsea	Manchester United	Recomendable
Romelu Lukaku	Delantero	Chelsea	Everton	Indeciso
Ryan Bertrand	Defensor	Chelsea	Southampton	No Recomendable
Danny Welbeck	Delantero	Manchester United	Arsenal	No Recomendable

## 11 Resultados

## 12 Discusión

## 13 Conclusiones & Recomendaciones {#conclusiones-&-recomendaciones}

## 14 Referencias bibliográficas

- Bawa, V. S. (1982). Stochastic dominance: A research bibliography. *Management Science*, 28, 698–712. <https://doi.org/10.1287/mnsc.28.6.698>
- Bergstra, J., Komera, B., Eliasmith, C., Yamins, D., & Cox, D. D. (2015). Hyperopt: A python library for model selection and hyperparameter optimization. *Computational Science & Discovery*, 8, 014008. <https://doi.org/10.1088/1749-4699/8/1/014008>
- Grover, A., & Leskovec, J. (2016). *node2vec: Scalable feature learning for networks*. arXiv.org. <https://arxiv.org/abs/1607.00653>
- Gustavo, V. (n.d.-b). *Decision under risk - module IV - NYU stern - master of science in business analytics*.
- Gustavo, V. (n.d.-a). *Decision under risk - module IV - NYU stern - master of science in business analytics*.
- Huang, E., Segarra, S., Gallino, S., & Ribeiro, A. (n.d.). *How to find the right player for your soccer team?*
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). *Efficient estimation of word representations in vector space*. arXiv.org. <https://arxiv.org/abs/1301.3781>
- PyTorch Forums, A. N. K. -. (2022). *Jensen shannon divergence*. PyTorch Forums. <https://discuss.pytorch.org/t/jensen-shannon-divergence/2626/13>
- Rahimian, P., Van Haaren, J., & Toka, L. (2023). Towards maximizing expected possession outcome in soccer. *International Journal of Sports Science & Coaching*, 174795412311544. <https://doi.org/10.1177/1747954123115449>
- Tippett, J. (2019). *The expected goals philosophy: A game-changing way of analysing football*. Independently Published.

## 15 Apéndices: Tablas, figuras, anexos

### Índice de Figuras

1	Modelo de Red de Jugadores . . . . .	6
2	Resultados Modelo de Regresión Lineal . . . . .	8
3	Gradiente del PSL . . . . .	9
4	Resultados Modelo de XGBoost . . . . .	10
5	Distribución de todos los $r(J, S)$ . . . . .	11
6	Distribución de los $r(J, S)$ de Sergio Agüero y Robin van Persie . . . . .	12
7	Distribución de los $r(J, S)$ de los 10 jugadores con mayor cantidad de disparos . . . . .	12
8	Distribución de los $r(J, S)$ de los 10 jugadores con mayor sesgo . . . . .	13
9	Top 20 Delanteros con distribución Beta más sesgada a la derecha - EPL 12/13 . . . . .	13
10	Distribución de los $r(J, S)$ de los 10 jugadores con mayor suma . . . . .	14
11	Distribución de los $r(J, S)$ de jugadores similares a Sergio Agüero . . . . .	14
12	Distribución de los $r(J, S)$ de jugadores similares a Sergio Agüero Superpuestos . . . . .	14
13	Distribución de los $r(J, S)$ de jugadores similares a Sergio Agüero de la misma posición . . . . .	15
14	Distribución de los parámetros $\alpha$ y $\beta$ de los $r(J, S)$ de los jugadores . . . . .	15
15	Distribución de los $r(J, S)$ de jugadores en clusters . . . . .	16
16	Matriz de Variables Aleatorias <b>R</b> . . . . .	18
17	Distribución del PSL del equipo Manchester City . . . . .	19
18	Ejemplo de dos distribuciones de PSL de dos formaciones distintas . . . . .	20
19	Comparación de CDFs de las distribuciones de PSL de las formaciones $L_{MC}$ y $L_{MC}^{Giroud}$ . . . . .	22
20	Grafo de Lineup . . . . .	24
21	Grafo de Jugadores . . . . .	25
22	Grafo de Jugadores Completo . . . . .	26
23	Embeddings de Jugadores en 3D . . . . .	28
24	Embeddings de Jugadores en 3D - PCA a 2D . . . . .	29
25	Embeddings de Jugadores en 64D - PCA a 2D . . . . .	30
26	Arquitectura del Modelo Base . . . . .	32
27	Resultados del Modelo Base . . . . .	33
28	Arquitectura del Modelo Tuned . . . . .	34
29	Resultados del Modelo Tuned . . . . .	35

30	Comparación de CDFs y PDFs de Welbeck y Walcott . . . . .	36
31	Comparación de CDFs y PDFs de Welbeck y Giroud . . . . .	37
32	Comparación de CDFs y PDFs de Milner y Gerrard . . . . .	38
33	Comparación de CDFs y PDFs de Milner y Downing . . . . .	39

## Índice de Tablas

1	Comparación de momentos de $\hat{f}_{PSL}^{1000}(L_{MC})$ y $\hat{f}_{PSL}^{1000}(L_{MC}^{\text{Giroud}})$ . . . . .	20
2	Resultados del Modelo Base . . . . .	33
3	Resultados del Modelo . . . . .	35
4	Comparación de momentos de $X_{L_{AR1}^{\text{Welbeck}}}$ y $X_{L_{AR1}}$ . . . . .	36
5	Comparación de momentos de $X_{L_{AR2}^{\text{Welbeck}}}$ y $X_{L_{AR2}}$ . . . . .	37
6	Comparación de momentos de $X_{L_{LIV1}^{\text{Milner}}}$ y $X_{L_{LIV1}}$ . . . . .	38
7	Comparación de momentos de $X_{L_{LIV2}^{\text{Milner}}}$ y $X_{L_{LIV2}}$ . . . . .	39
8	Recomendaciones de Transferencias Clave . . . . .	40

## Índice de Algoritmos

1	Simulación del PSL del equipo A . . . . .	19
2	Dominancia Probabilística . . . . .	21
3	Construcción del Grafo de Jugadores . . . . .	27