

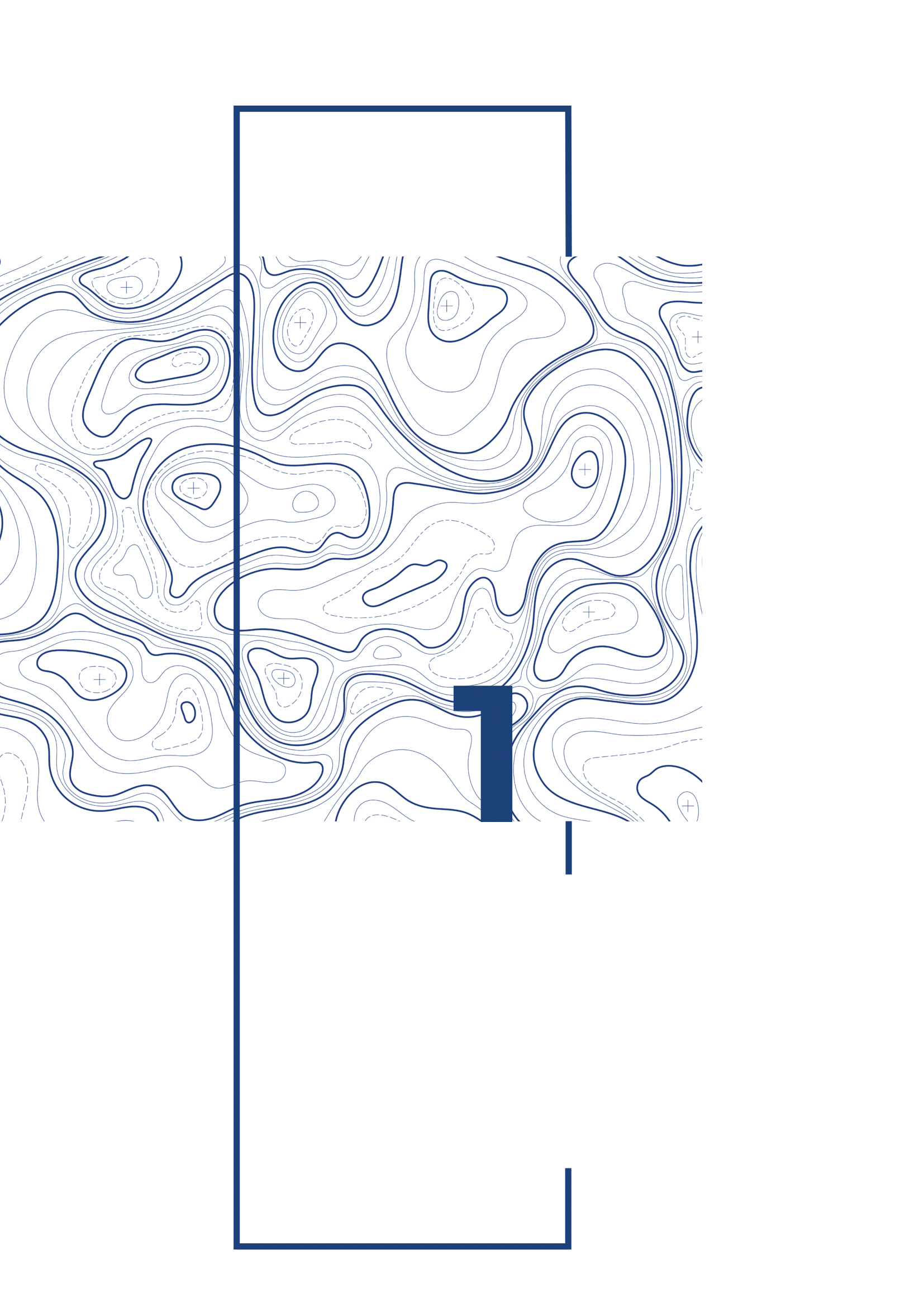
# **Proyecto de Datos I**

## **Preparación de datos**

**Óscar Marín Esteban**

**Carlos Mantilla Mateos**

**Álvaro Enol Alonso Ortega**



# **EXTRACCIÓN DE LOS DATOS**

**Vamos a usar diferentes fuentes de datos, entre ellas las siguientes:**

**Fuente 1: Página de Whoscored (**[**Link**](https://www.whoscored.com/Regions/206/Tournaments/4/Seasons/9682/Stages/22176/PlayerStatistics/Spain-LaLiga-2023-2024)**), de esta página hemos sacado todas las características generales de los jugadores.**

**Librerías o herramientas:**

**Las librerías utilizadas son selenium para hacer todo el tema relacionado con la extracción de datos, pandas para el manejo del dataframe, HTML para el manejo en si de diferentes elementos de la web y time para esperar el tiempo suficiente antes de acceder a un elemento.**

**Pseudocódigo, Aspectos y Problemas:**

**En cuanto a la captura, lo que hacemos es, primero que nada, inicializar el navegador, una vez inicializado, tuvimos un problema con las cookies, porque al iniciarlo nos salía un aviso de cookies, por lo que lo que hicimos fue una función que pasa las cookies nada más iniciar la web, que pique en el botón correspondiente. Una vez pasadas las cookies pasamos a leer la tabla leemos la primera página de el primer año seleccionado, tenemos dos bucles, uno por los años y otro por el número de páginas que hay, en este segundo, para a partir de la primera página, teníamos un problema al cambiar la página, ya que salía un icono privacy que no se podía quitar, por lo que optamos por la solución de bajar la página un poco para que desapareciera con (driver.execute\_script("window.scrollBy(0, 750)")), a partir de aquí ya habríamos hecho todo lo necesario para tener todos los datos en nuestras manos.**

**Fuente 2: Página de Transfermarket, de aquí hemos sacado principalmente, aunque se pondrá detalladamente más adelante, la posición exacta, la edad y el precio.**

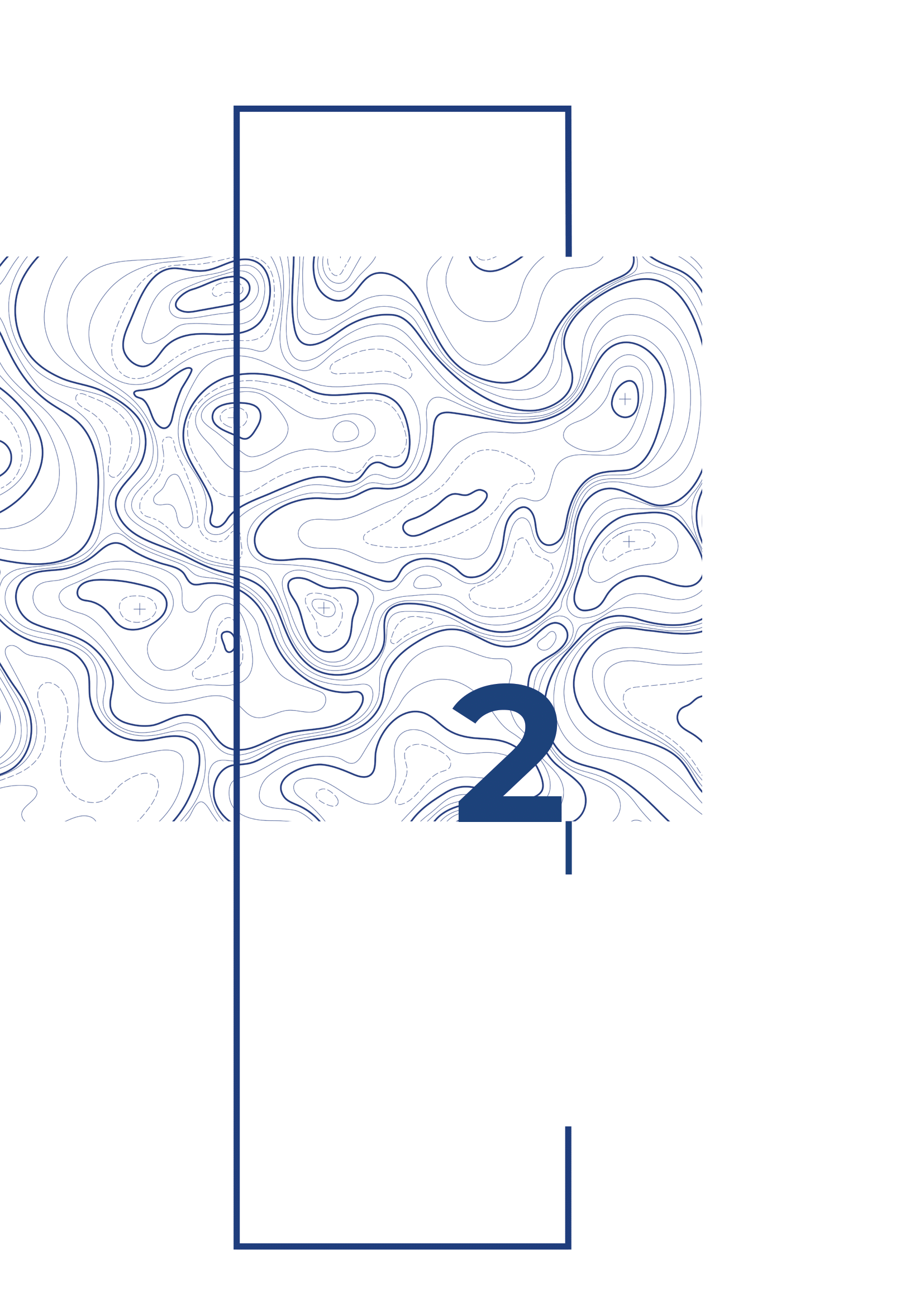
**Librerías o herramientas:**

**En este caso, hemos utilizado una API, ya que al sacarlo normal no nos dejó, por lo que tras investigar un poco descubrimos esta API.**

**Pseudocódigo, Aspectos y Problemas:**

**A partir de aquí, todo fue bastante sencillo, simplemente usar la API, e ir añadiendo lo leído a un dataframe, que luego descargaremos, lo realmente difícil fue encontrarla y aplicarla al principio.**

**.**

* + - 1. ****

# **TRANSFORMACIÓN DE LOS DATOS**

* + - 1. **Variables:**
    1. Leyenda de utilidad(la utilidad la hemos tratado para todas las variables, considerando TODOS los modelos):
    2. SI
    3. NO
    4. QUIZÁS
    5. WhoScored:
    6. Todas las variables aquí que ponen número son numéricas, todas las variables que ponen tasa son tasas y todas las que ponen % son de porcentaje.

**Apps: Son las apariciones de un jugador en la temporada completa, entre paréntesis las veces que han empezado de suplentes.**

**Mins: Número de minutos jugados**

**Goals: Número de Goles en la temporada**

**Assists: Número de asistencias en la temporada**

**Yel: Número de tarjetas amarillas**

**Red: Número de tarjetas rojas**

**SpG: Número de tiros por partido**

**PS%: Porcentaje de acierto de pase**

**AerialsWon: Duelos aéreos ganados por partido**

**MotM: Número de veces que ha sido jugador del partido**

**Tackles: Número de segadas por partido**

**Inter: Número de intercepciones por partido**

**Fouls: Número de faltas por partido**

**Offsides\_won: Tasa de fueras de juego provocados por partido**

**Clear: Número de despejes por partido**

**Drb\_deffensive: Número de veces que le han regateado su defensa por partido**

**Blocks: Tasa de bloqueos de tiro por partido**

**OwnG: Número de goles en propia**

**KeyP: Número de pases clave por partido**

**Drb\_offensive: Número de veces que han regateado por partido**

**Fouled: Tasa de faltas recibidas por partido**

**Offsides: Tasa de fueras de juego por partido**

**Disp: Tasa de pérdidas de balón**

**UnsTch: Tasa de malos controles por partido**

**AvgP: Tasa media de pases por partido**

**Crosses: Tasa de centros por partido**

**LongB: Tasa de pases largos por partido**

**Transfermarket:**

**Age, Id son enteras, Height es numérica decimal, y el resto son strings**

**Id: Id del jugador**

**Position: Posición en el campo**

**DateofBirth: Año de nacimiento**

**Age: Edad del Jugador**

**Nationality: nacionalidad**

**CurrentClub: Club Actual**

**Height: Altura del jugador**

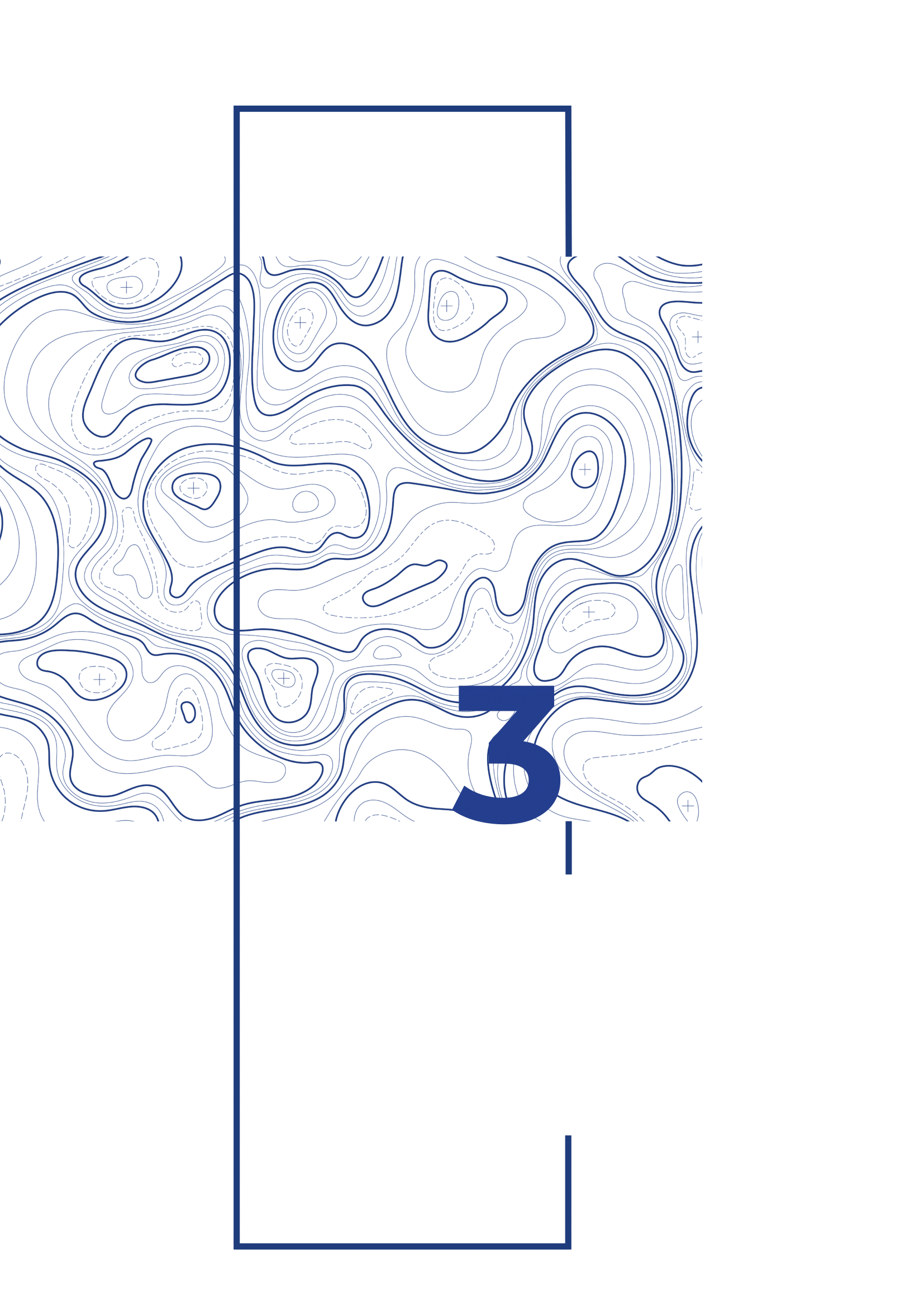
**Foot: Pie dominante**

**JoinedOn: Fecha de incorporación al equipo actual**

**SignedFrom: El equipo en el que estaba antes de que lo ficharan**

**MarketValue: Valor de mercado del jugador**

**Status: Jugador lesionado o no**

****

# **CARGA DE LOS DATOS**

## **Evaluación de éxito del sistema**

La medida en este caso que se puede utilizar para medir el éxito del sistema podría ser un error relacionado con la oferta aceptada por el jugador si es fichado por otro equipo, las ofertas de otros equipos y su relación con la nuestra si el jugador ha aceptado nuestra oferta y el valor de mercado del jugador al final de la temporada.

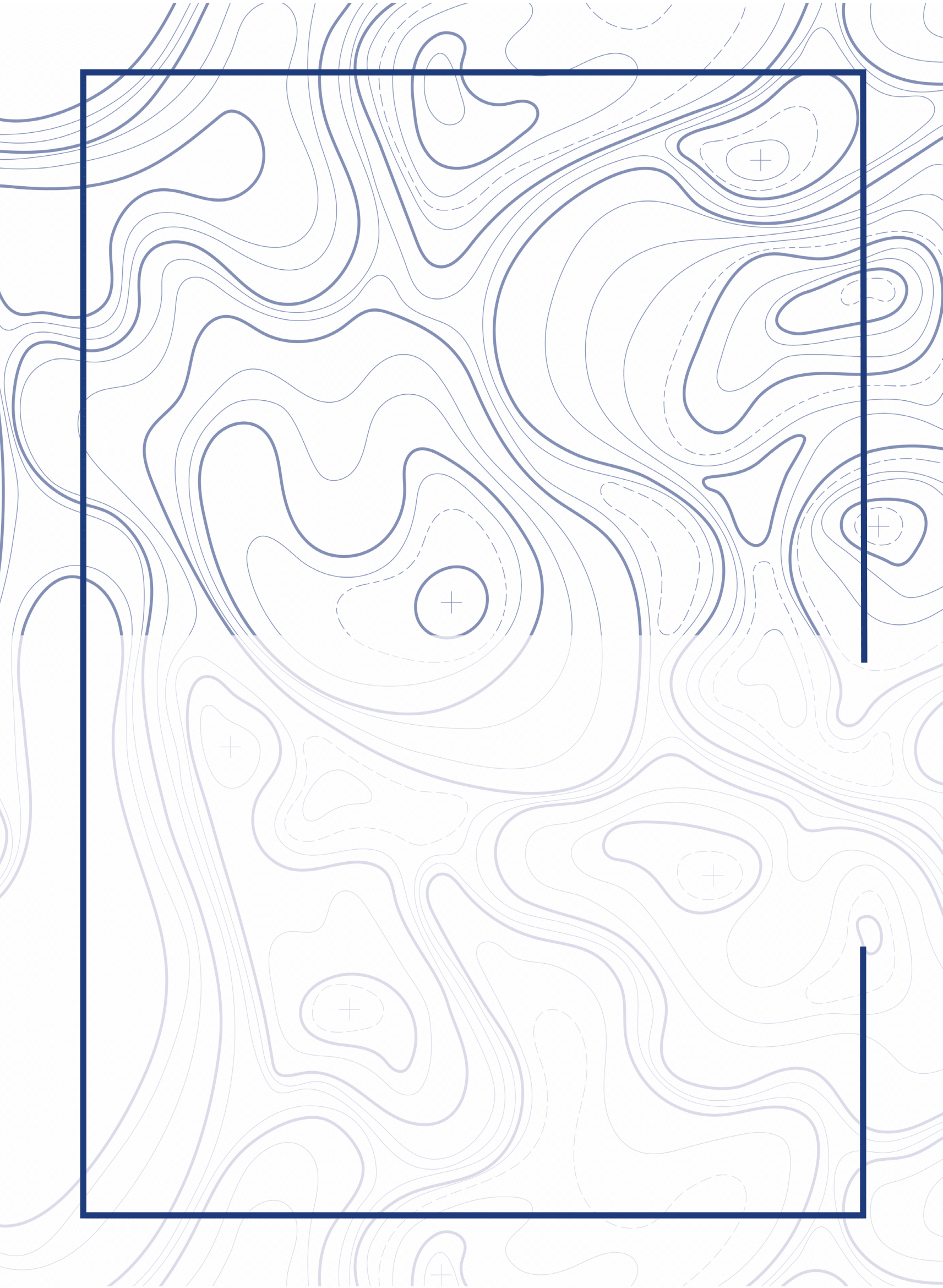
Además, el valor de mercado del jugador al final de la temporada se puede comparar con el que nuestro modelo predecía a lo largo de esta, así como su evolución según los partidos jugados o los campeonatos ganados. Estas medidas nos pueden ayudar a mejorar las predicciones de nuestro modelo y determinar sí dichas predicciones se ajustan a la realidad.

Este modelo no es el único que se puede crear, por tanto, la existencia de otros modelos puede ayudar a mejorar el nuestro.

1. **Evaluación de éxito de la tarea de aprendizaje automático**

Las medidas que nos pueden ayudar a evaluar la tarea del aprendizaje automático pueden ser la coherencia con valores del pasado, los errores cuadráticos medios, el cumplimiento de determinadas hipótesis como independencia, homocedasticidad y normalidad de los residuos.

Estas medidas nos ayudan a saber si nuestro sistema es válido para realizar las predicciones, así como la posibilidad de error de nuestro modelo, la cual debemos minimizar. En cambio, la evaluación de éxito del sistema nos permite saber si funciona correctamente con valores reales, no con los del test, así como su precisión en un ámbito real. La evaluación de la tarea de aprendizaje automático es imprescindible: si nuestro modelo falla, la empresa no tendrá ningún uso para ello y no contratará nuestros servicios. En cambio, la evaluación de éxito del sistema sirve para ver su precisión en ámbitos reales, por lo que siempre podemos ajustar nuestro sistema para hacerlo más efectivo según nuevos valores. La empresa querrá un sistema que evolucione en el tiempo, ya que es posible que el modelo cambie a lo largo de los años y sea necesario retocar el sistema.

****