

# ***NBA PLAYERS STATS***

Jose Daniel Figueroa Arenas  
Kevin Daniel Castro Mendoza  
Juan Camilo Velandia Amortegui

Presentación General – Proyecto Final







# ***PRIMERA ENTREGA***





# ***OBJETIVOS PRINCIPALES***

Desarrollar un clasificador con el cual los equipos pueden identificar con precisión la posición ideal en la que juega un jugador basándose en sus estadísticas individuales de rendimiento, como puntos, asistencias, rebotes y minutos jugados.





# DATA SET



**Temporadas  
2021 - 2024**



**30 Columnas y  
+ 2K Filas**



***Nombre  
Posicion  
Edad  
Equipo***

**Datos de los  
Jugadores**



**Estadísticas  
por partido**

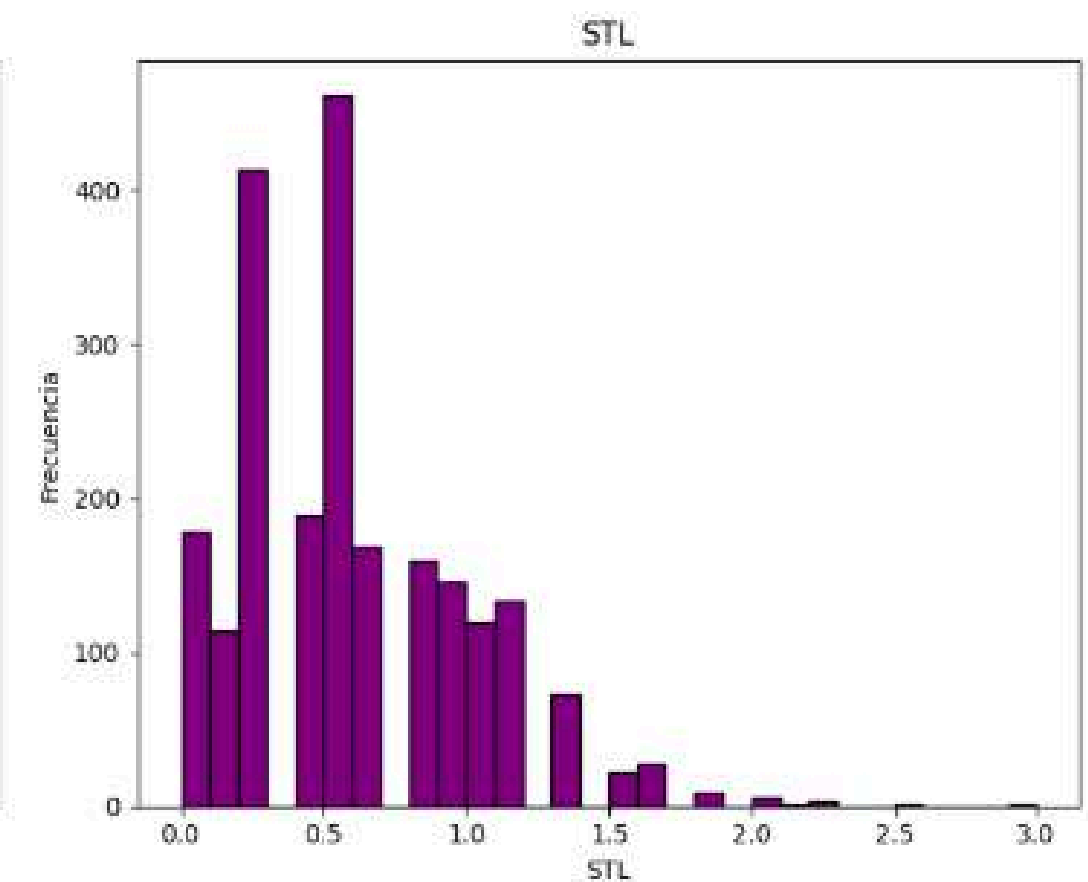
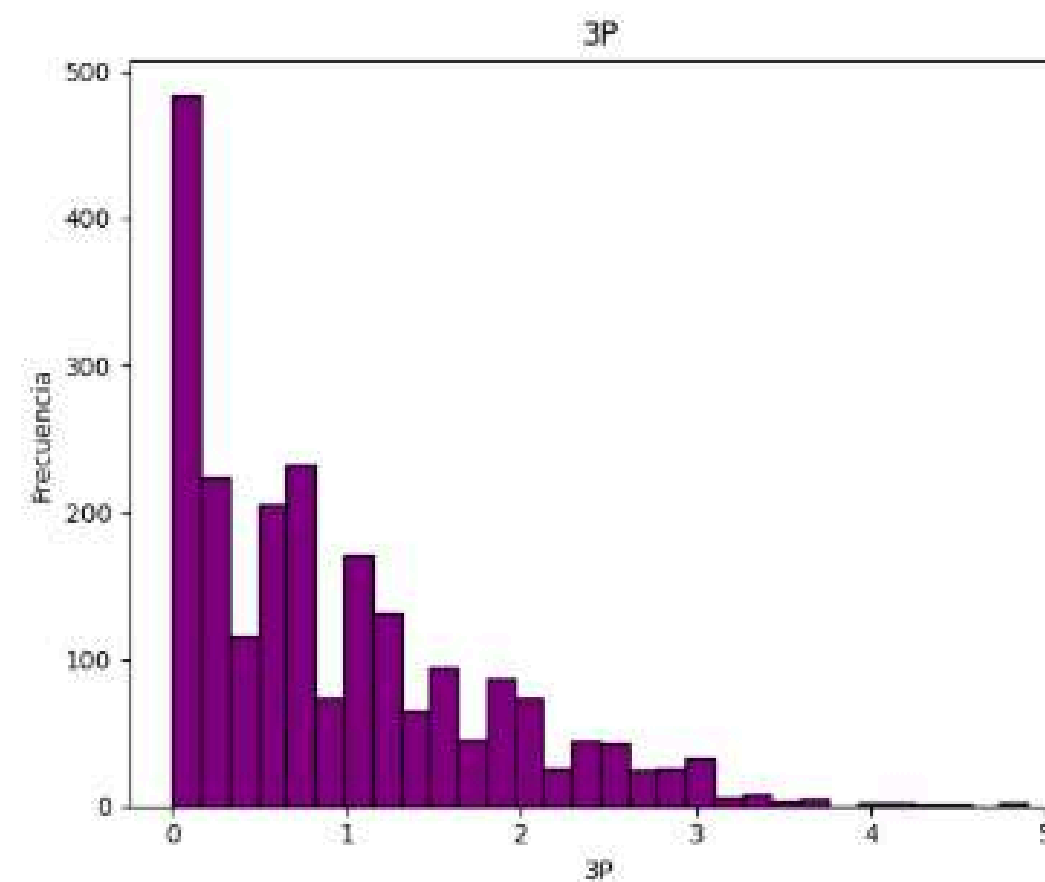
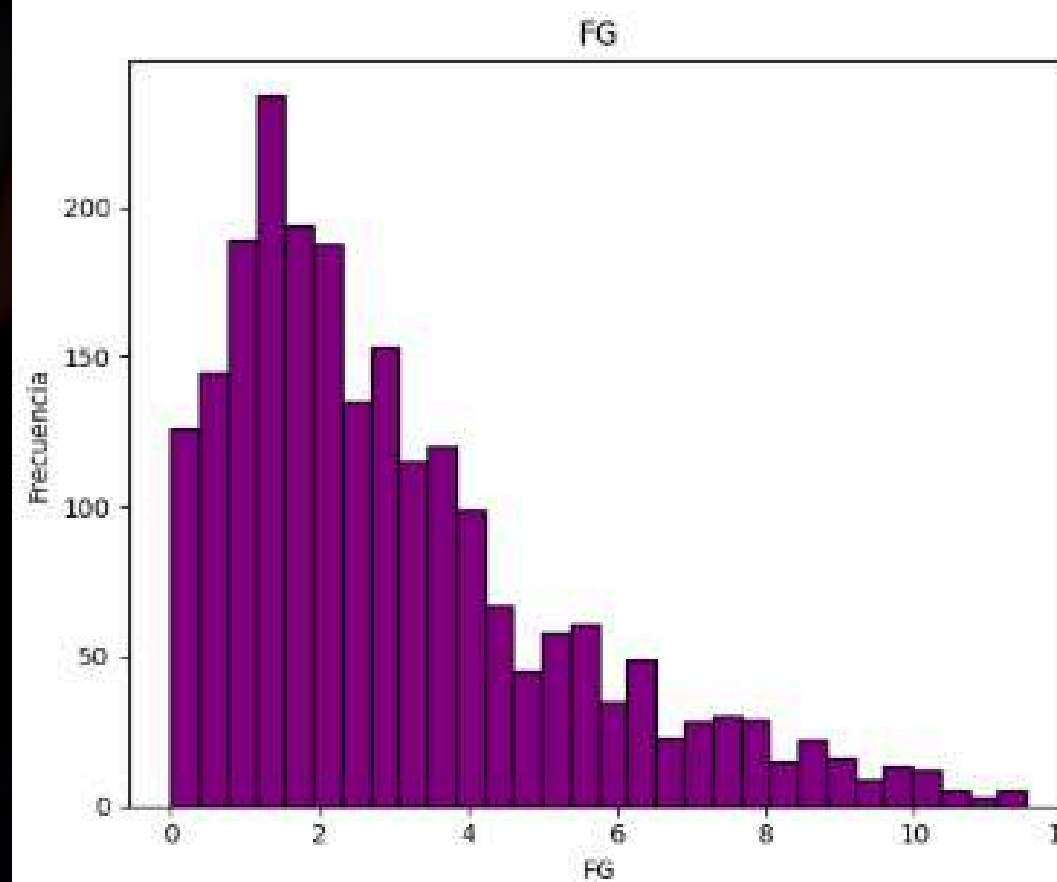
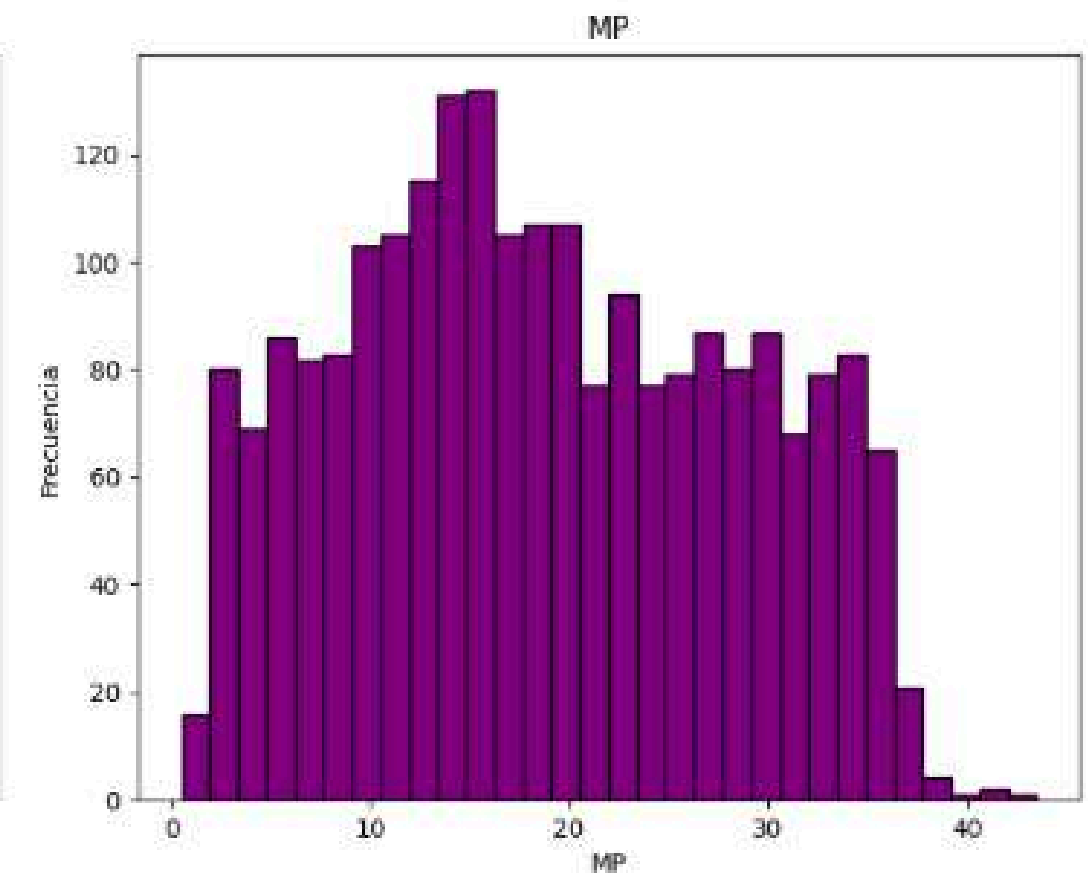
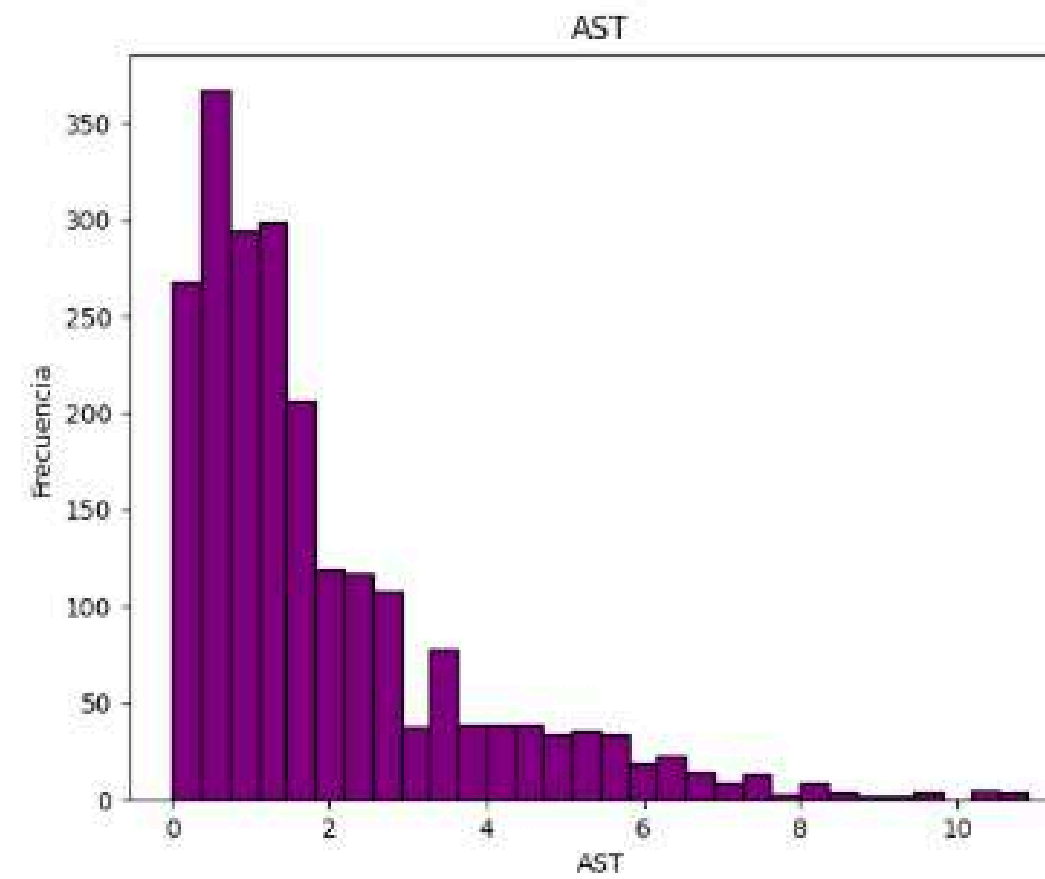
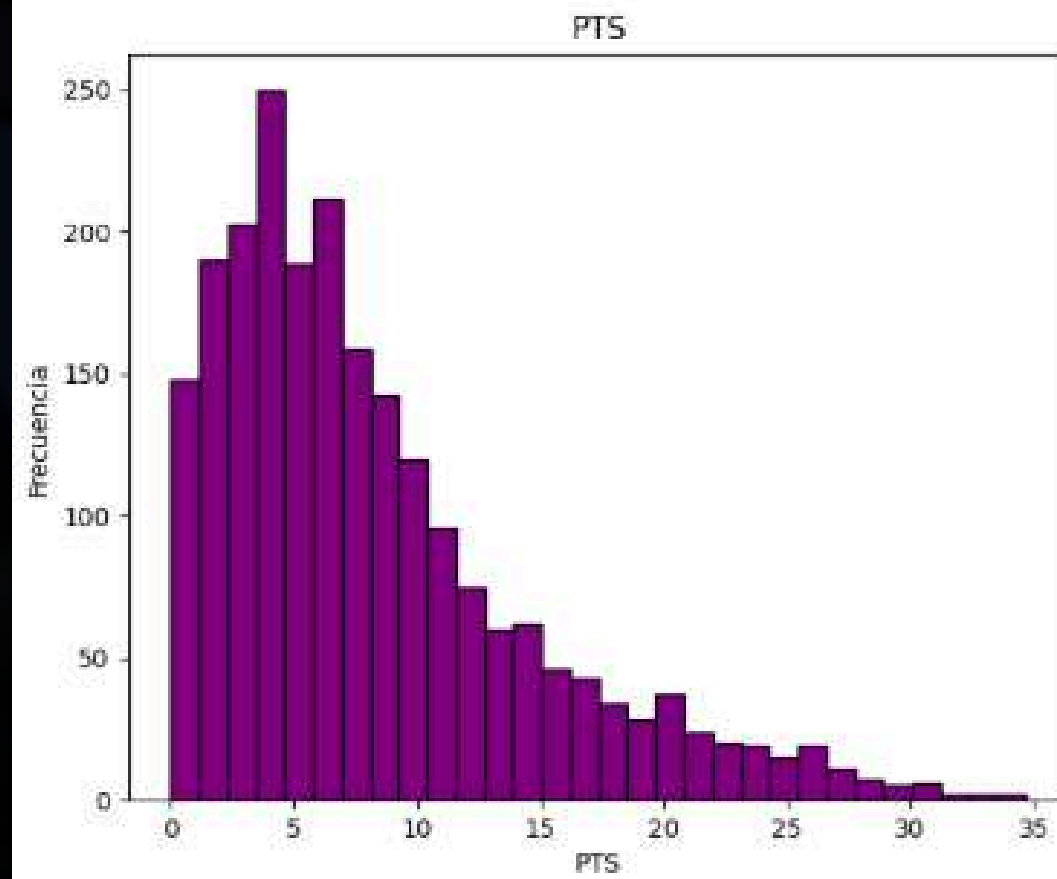
***Anotaciones  
Tiros  
Rebotes  
Bloqueos  
Asistencias  
Minutos jugados***



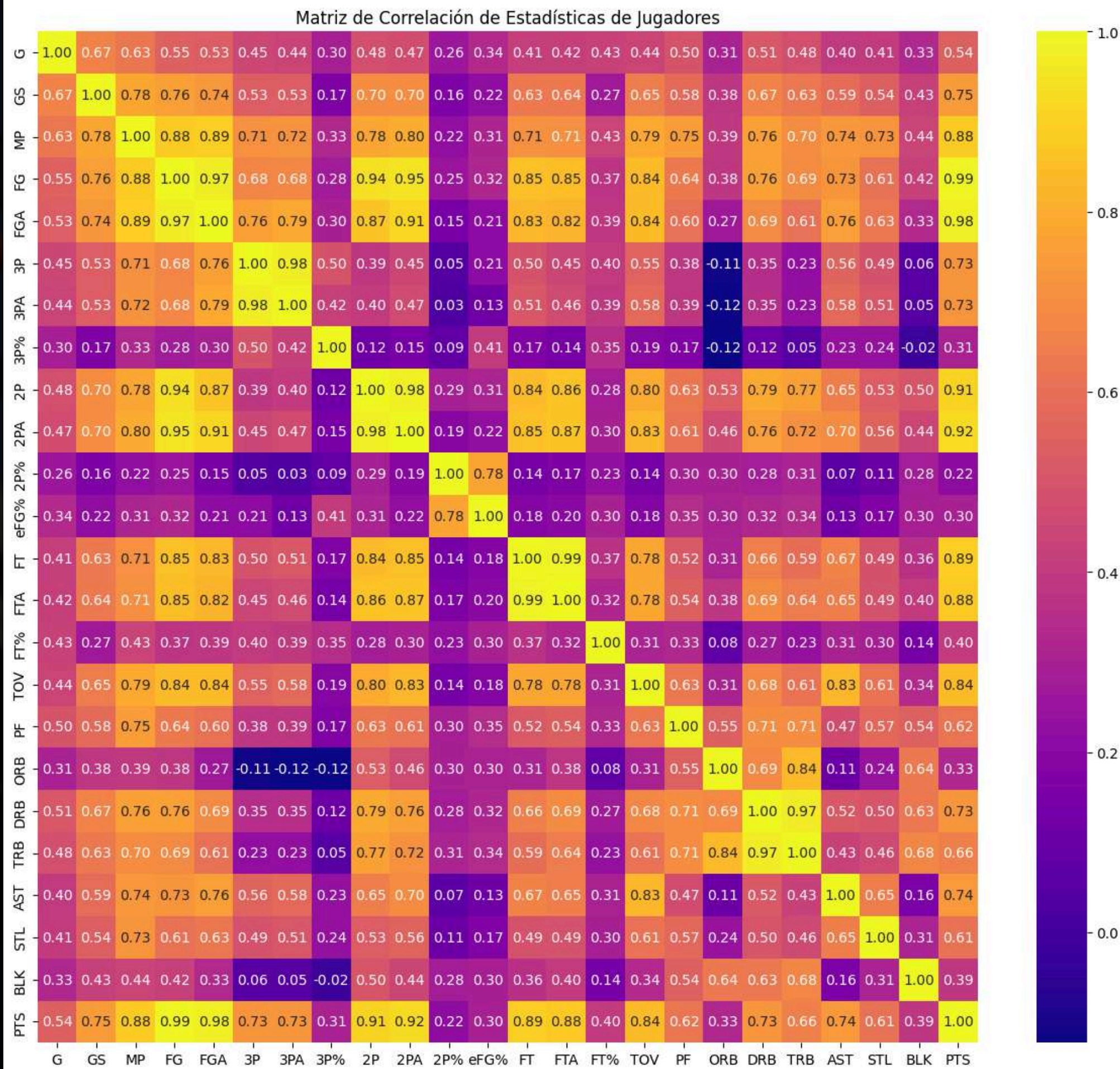
# ***ANALISIS***

- **Nuestro análisis comienza con la limpieza y manipulación de datos, eliminando valores nulos y normalizando las estadísticas.**
- **Realizar una exploración inicial entre variables clave, como anotaciones, minutos jugados, bloqueos, asistencias, etc.**
- **Como meta final seria usar técnicas de machine learning para entrenar un modelo que prediga la posición óptima de un jugador en función de sus estadísticas.**

# HISTOGRAMAS







PTS = Puntos

BLK = Bloqueos

STL = Robos

AST = Asistencias

TRB = Rebotes Totales

DRB = Rebotes Defensivos

ORB = Rebotes Ofensivos

PF = Faltas Personales

TOV = Pérdidas

FT% = Porcentaje de Tiros Libres Hechos

FTA = Intentos de Tiros Libres

FT = Tiros Libres Hechos

eFG% = Porcentaje Efectivo de Canastas

2P% = Porcentaje de Dos Puntos Hechos

2PA = Intentos de Dos Puntos

2P = Dos Puntos Hechos

3P% = Porcentaje de Triples Hechos

3PA = Intentos de Triples

3P = Triples Hechos

FGA = Intentos de Canasta de Campo

FG = Canastas de Campo Hechas

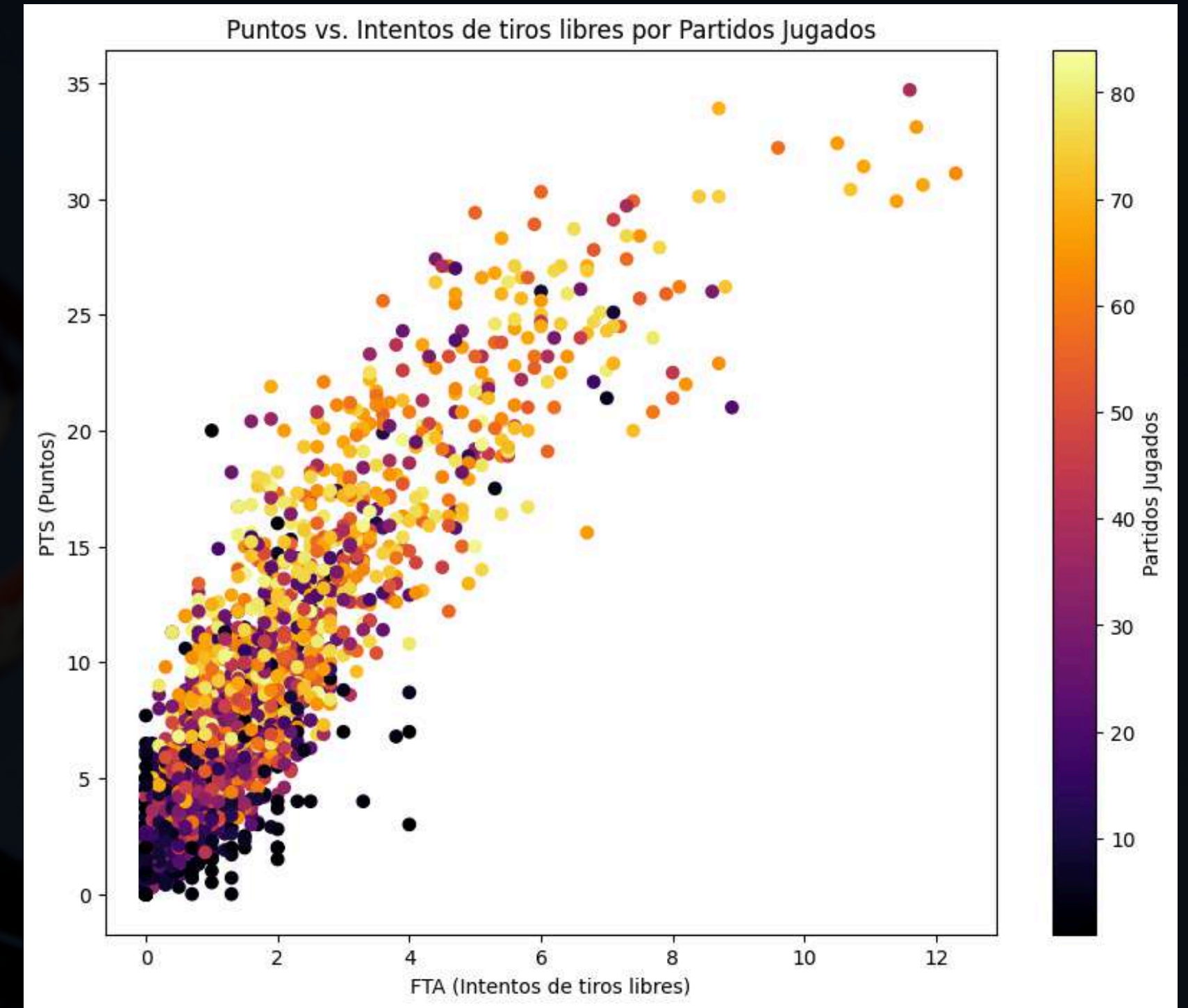
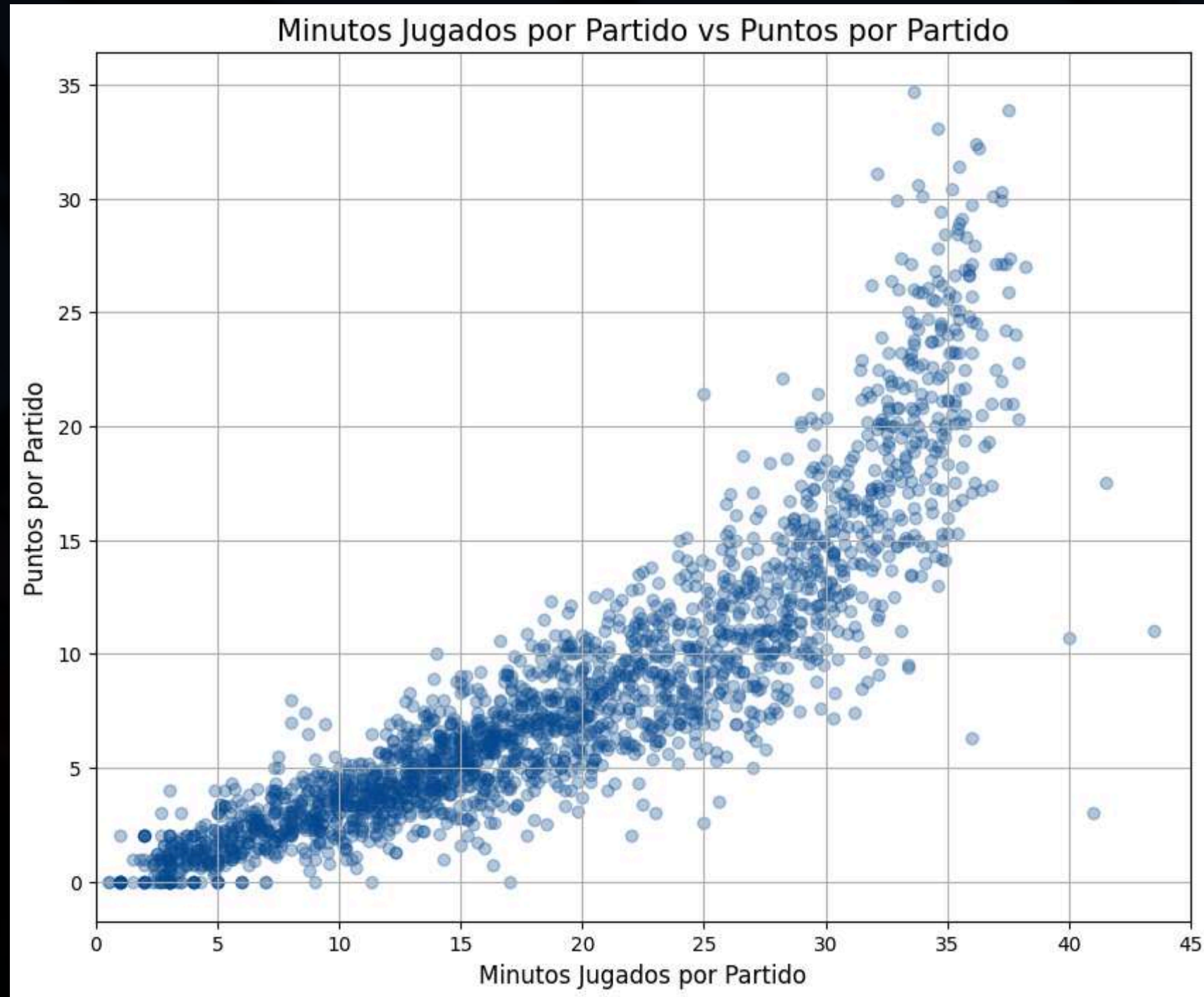
MP = Minutos Jugados por Partido

GS = Partidos Iniciados

G = Partidos Jugados

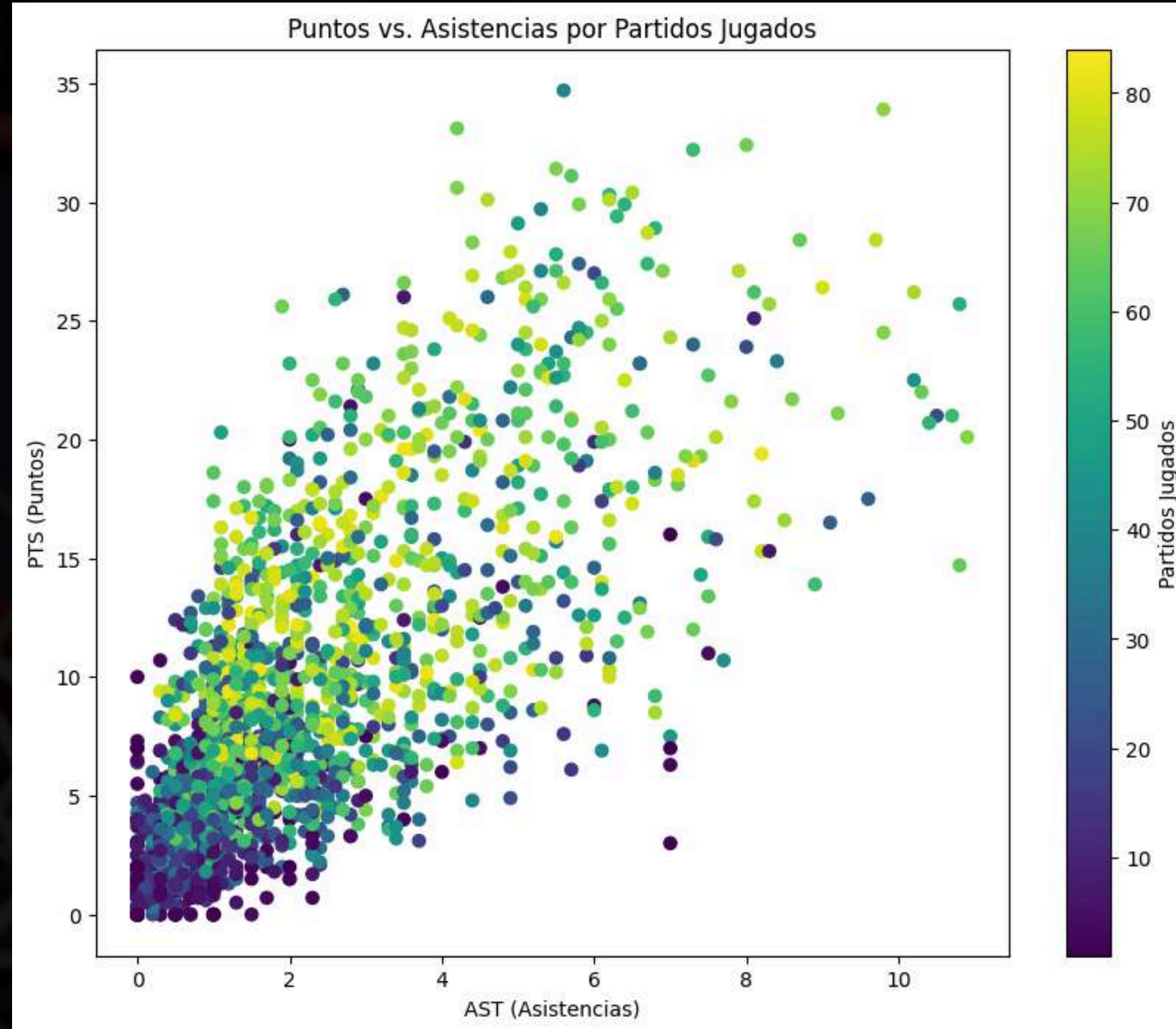


# GRAFICOS DE DISPERSIÓN





# GRAFICOS DE DISPERSIÓN







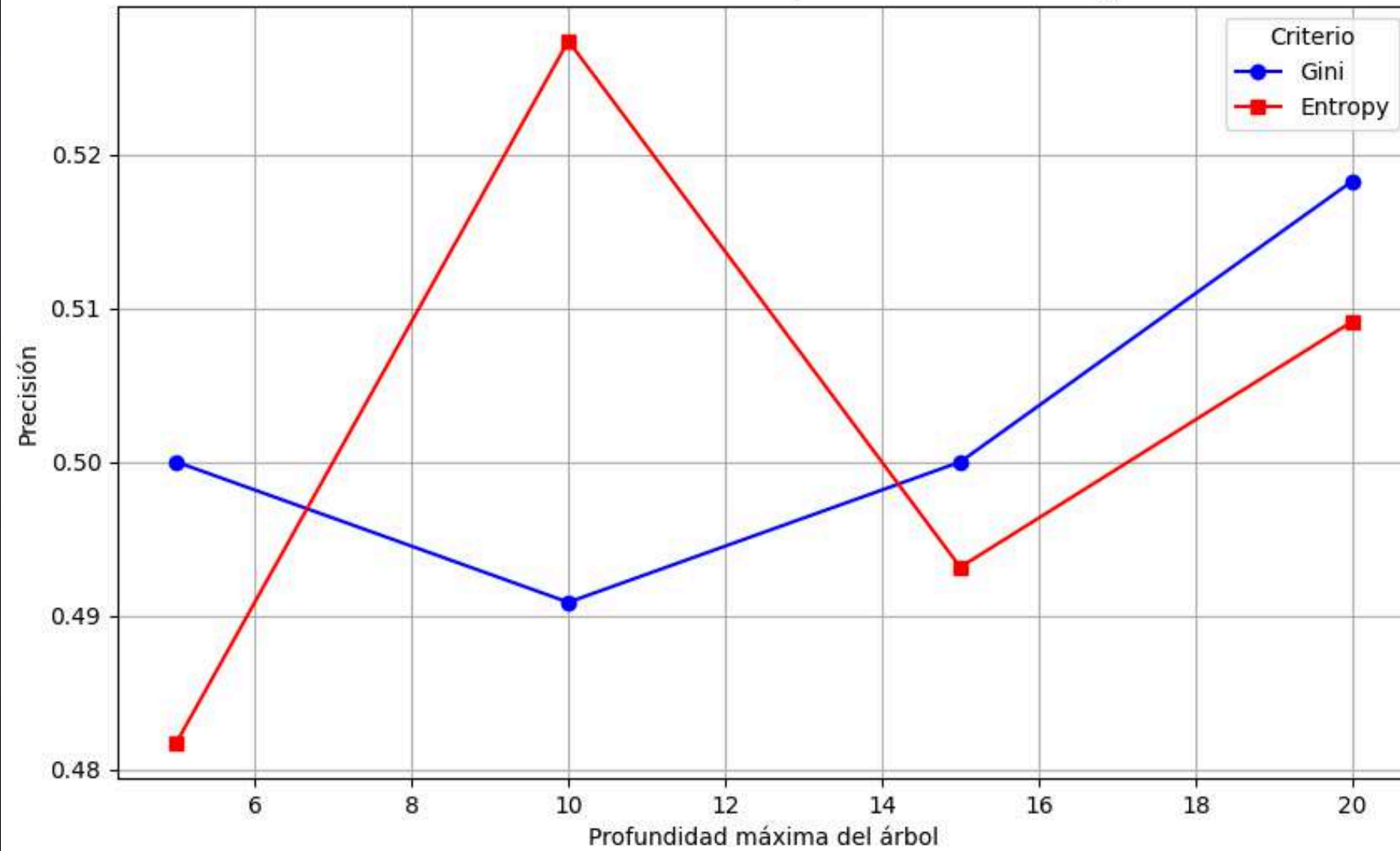
# ***SEGUNDA ENTREGA***



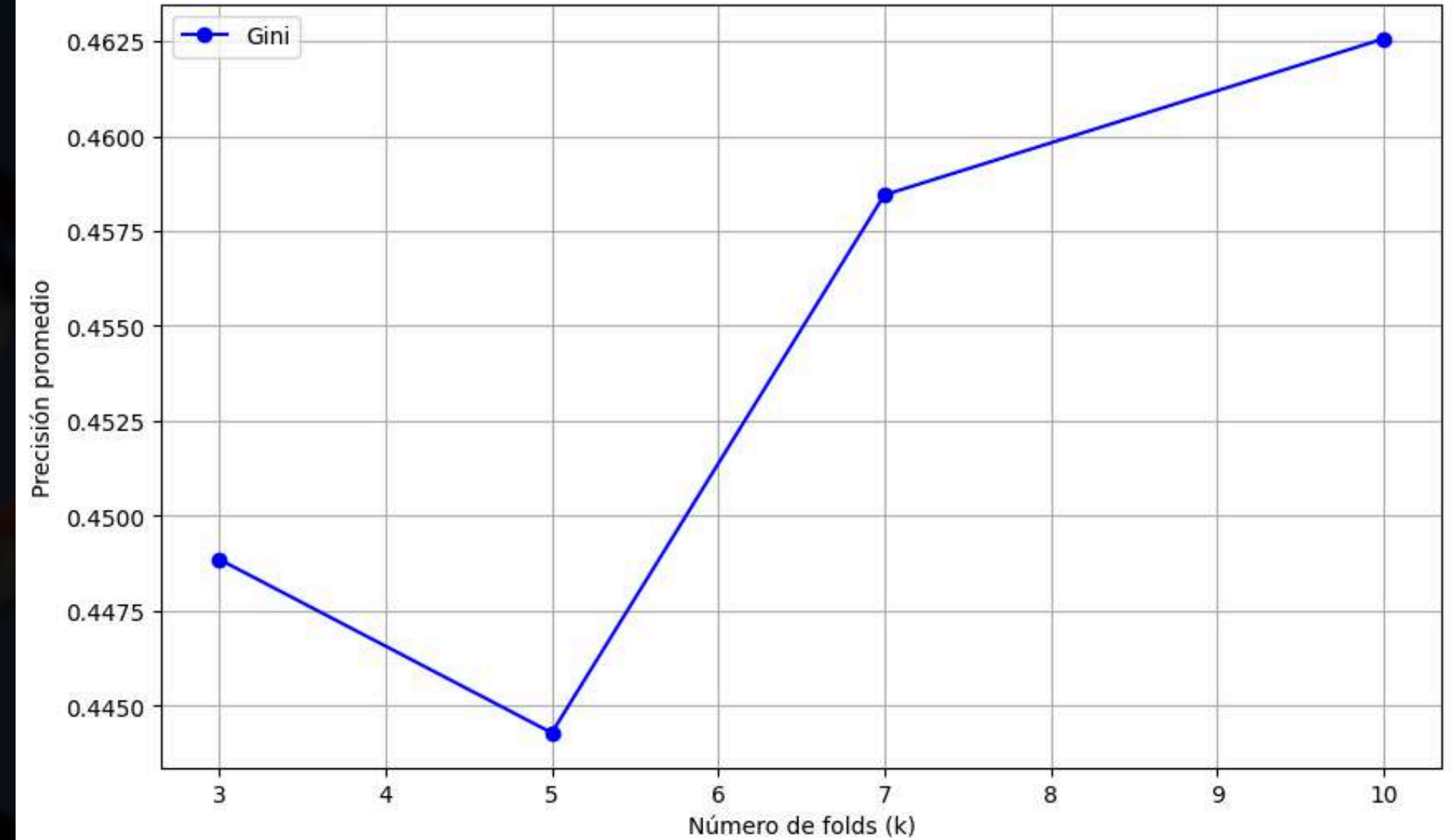
# DECISION TREE: CLASSIFIER



Precisión del modelo en función de la profundidad del árbol y el criterio



Precisión del modelo Decision Tree en función del número de folds

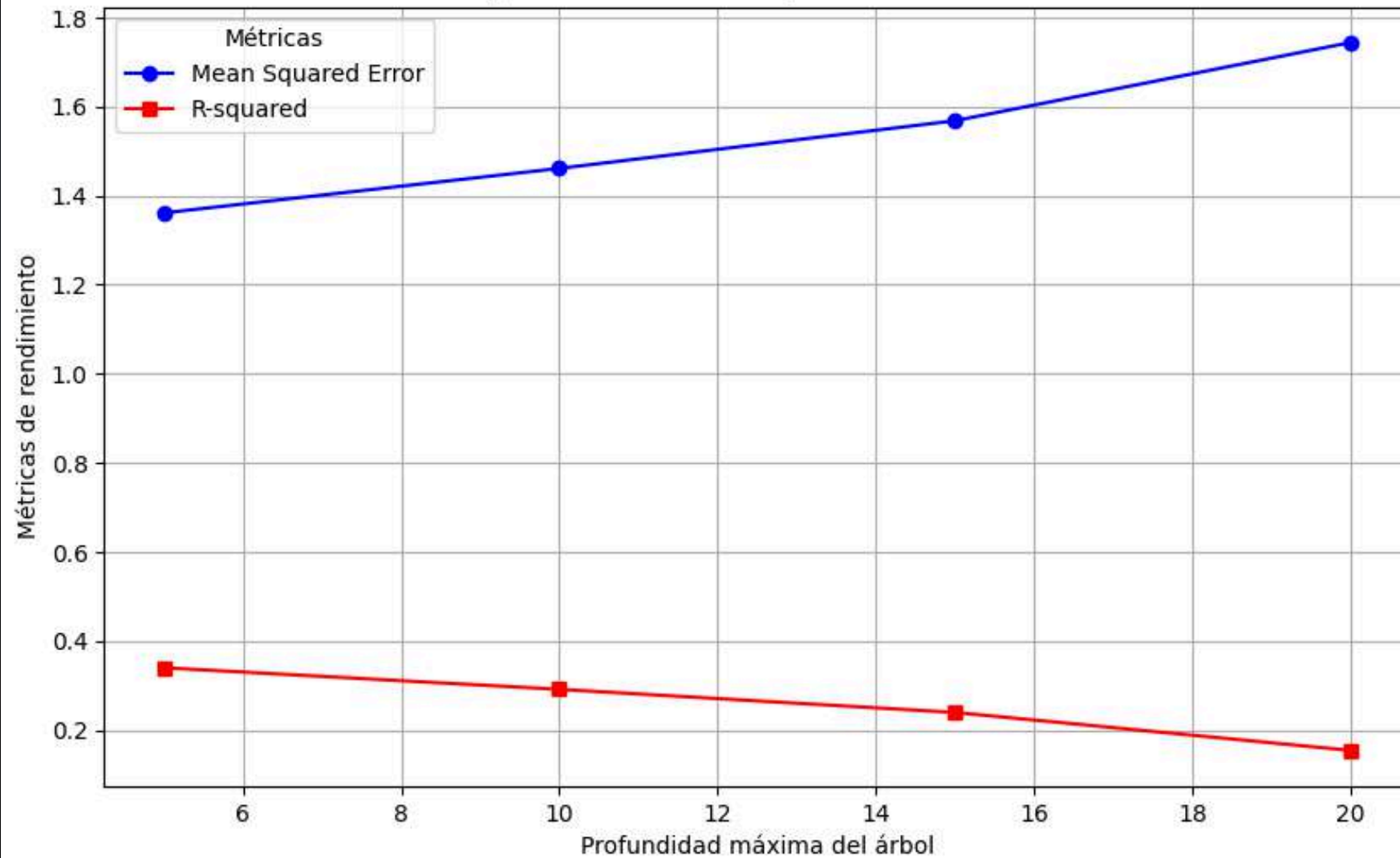




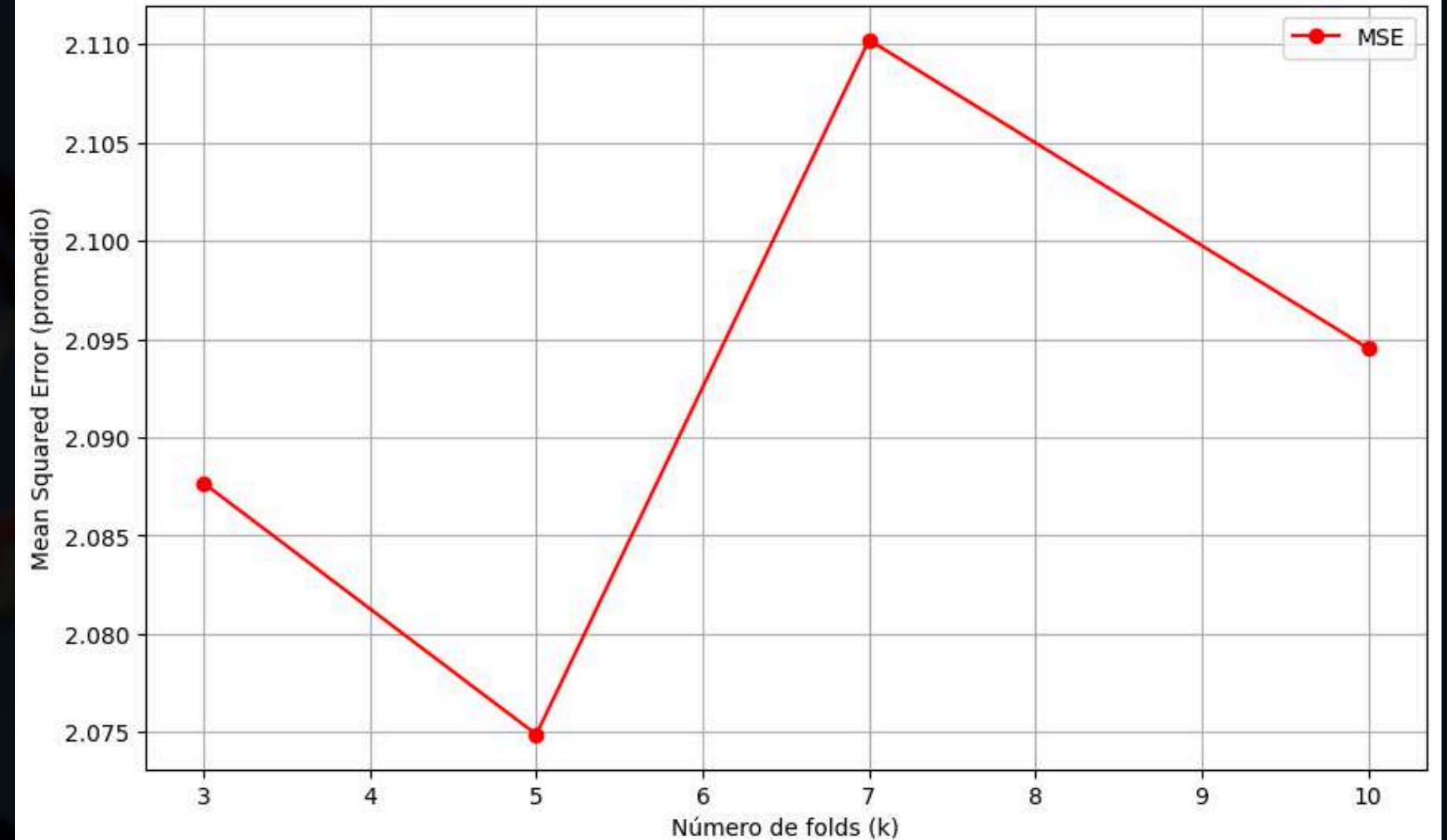
# DECISION TREE: REGRESSOR



MSE y R<sup>2</sup> en función de la profundidad del árbol

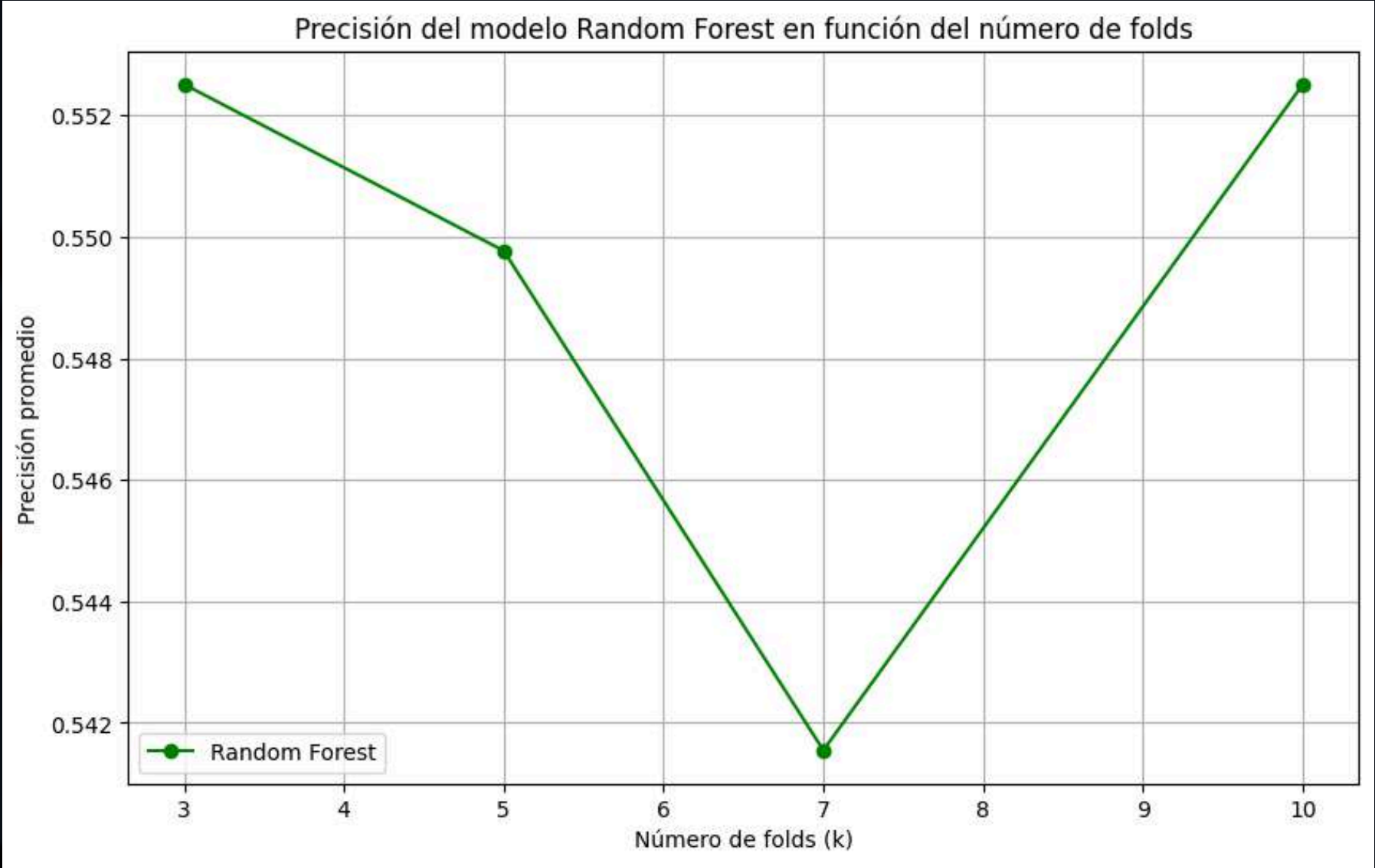
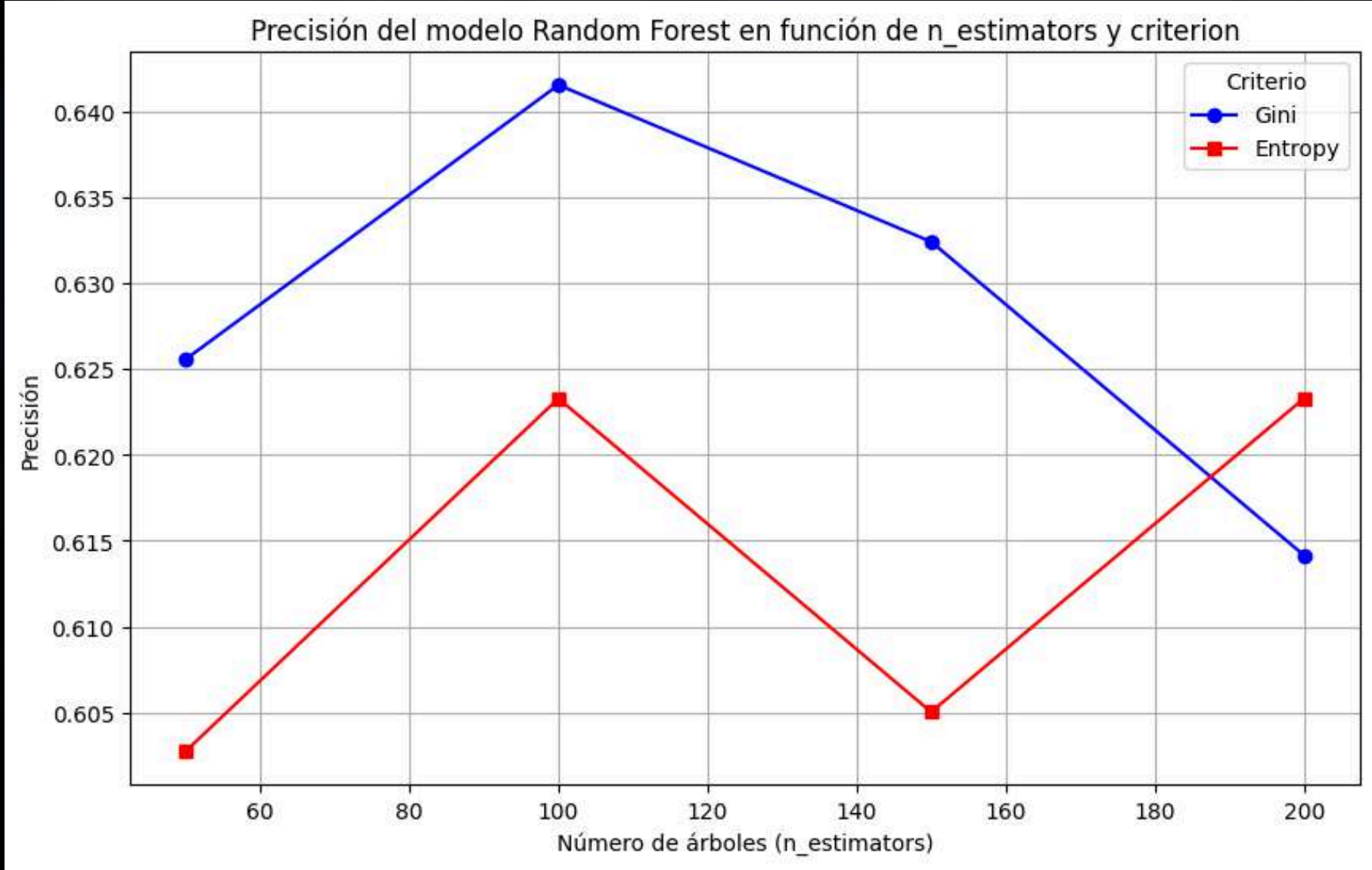


MSE del modelo Decision Tree en función del número de folds



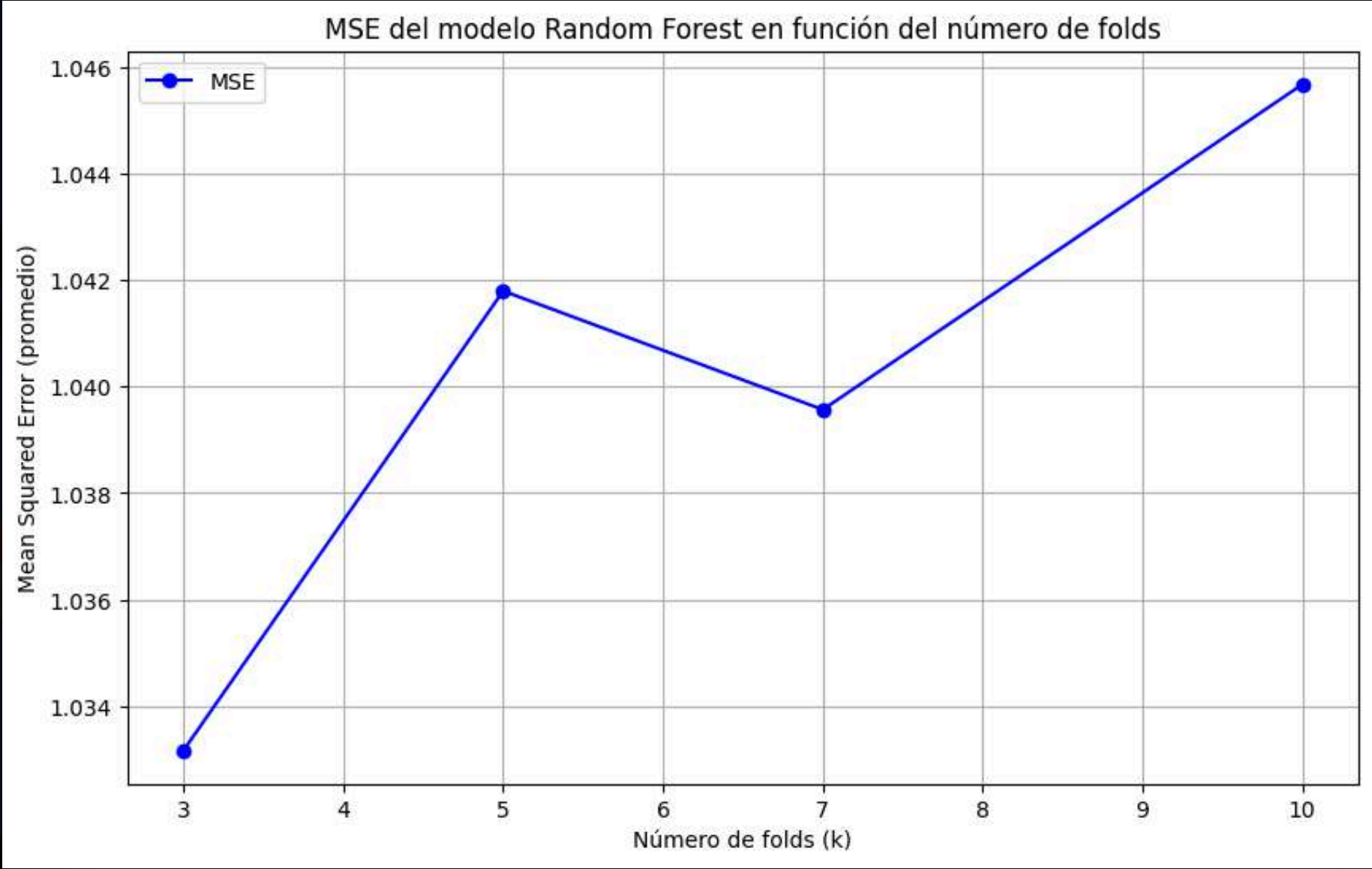
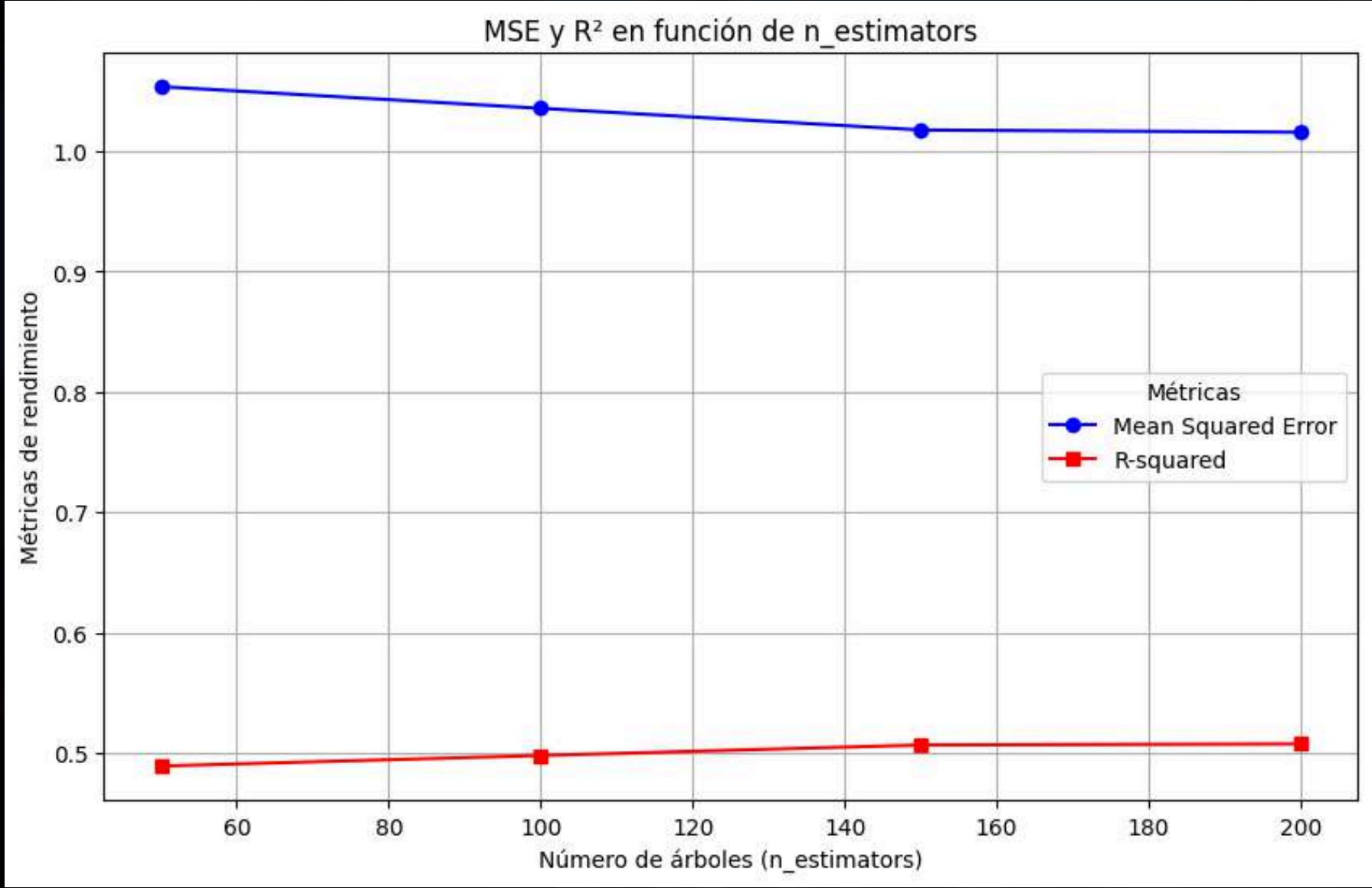


# RANDOM FOREST: CLASSIFIER





# RANDOM FOREST: REGRESSOR







# ***TERCERA ENTREGA***



# ***OBJETIVOS PRINCIPALES***

- Diseñar y experimentar con redes neuronales utilizando configuraciones de 3, 6 y 10 capas ocultas para identificar y aprender patrones de datos específicos.
- Implementar una prueba de validación para verificar la capacidad del modelo de predecir correctamente la posición de un jugador.
- Desarrollar un ejemplo práctico que permita clasificar y seleccionar a los mejores jugadores para cada posición a partir de un conjunto de datos de jugadores.

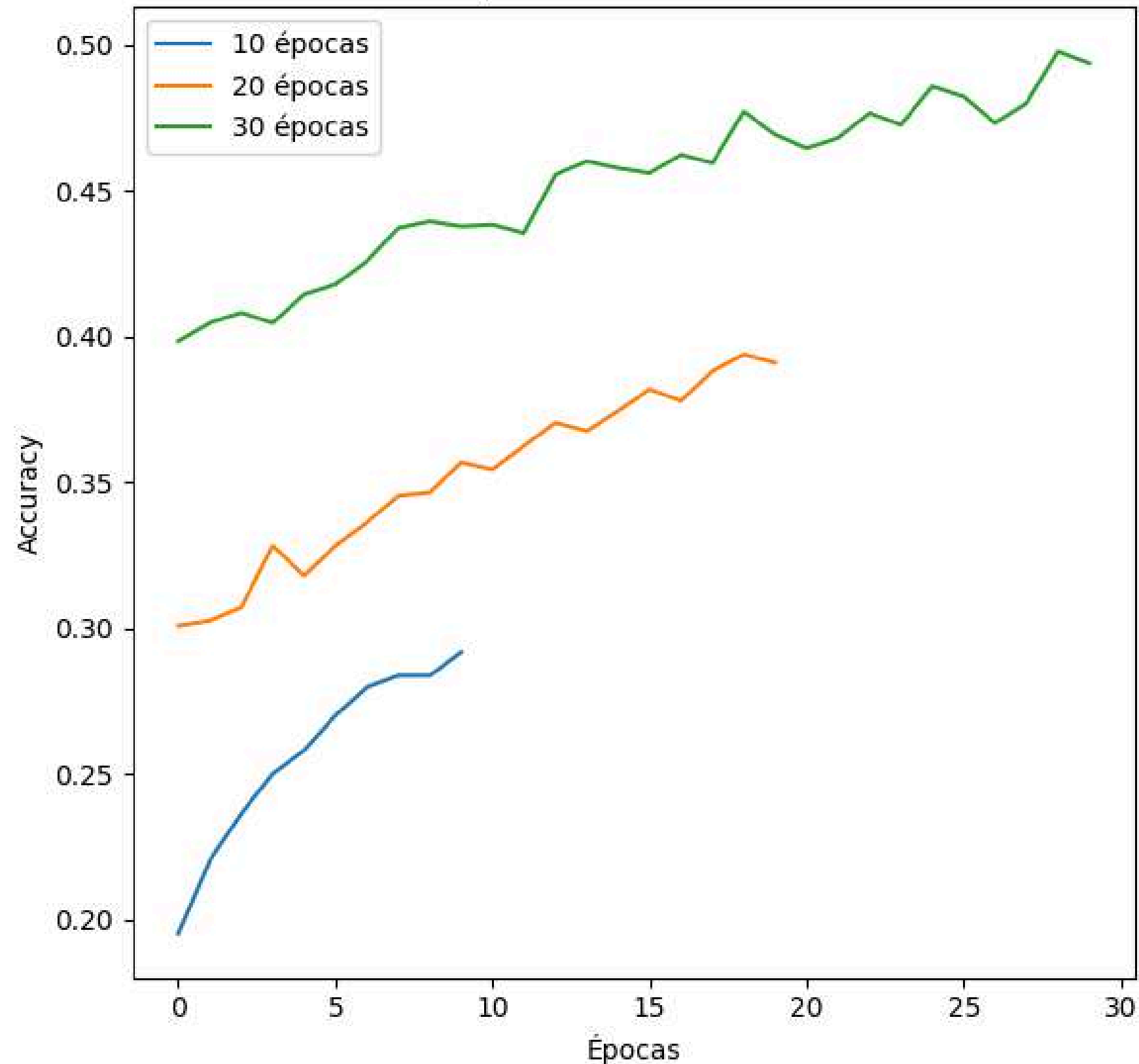




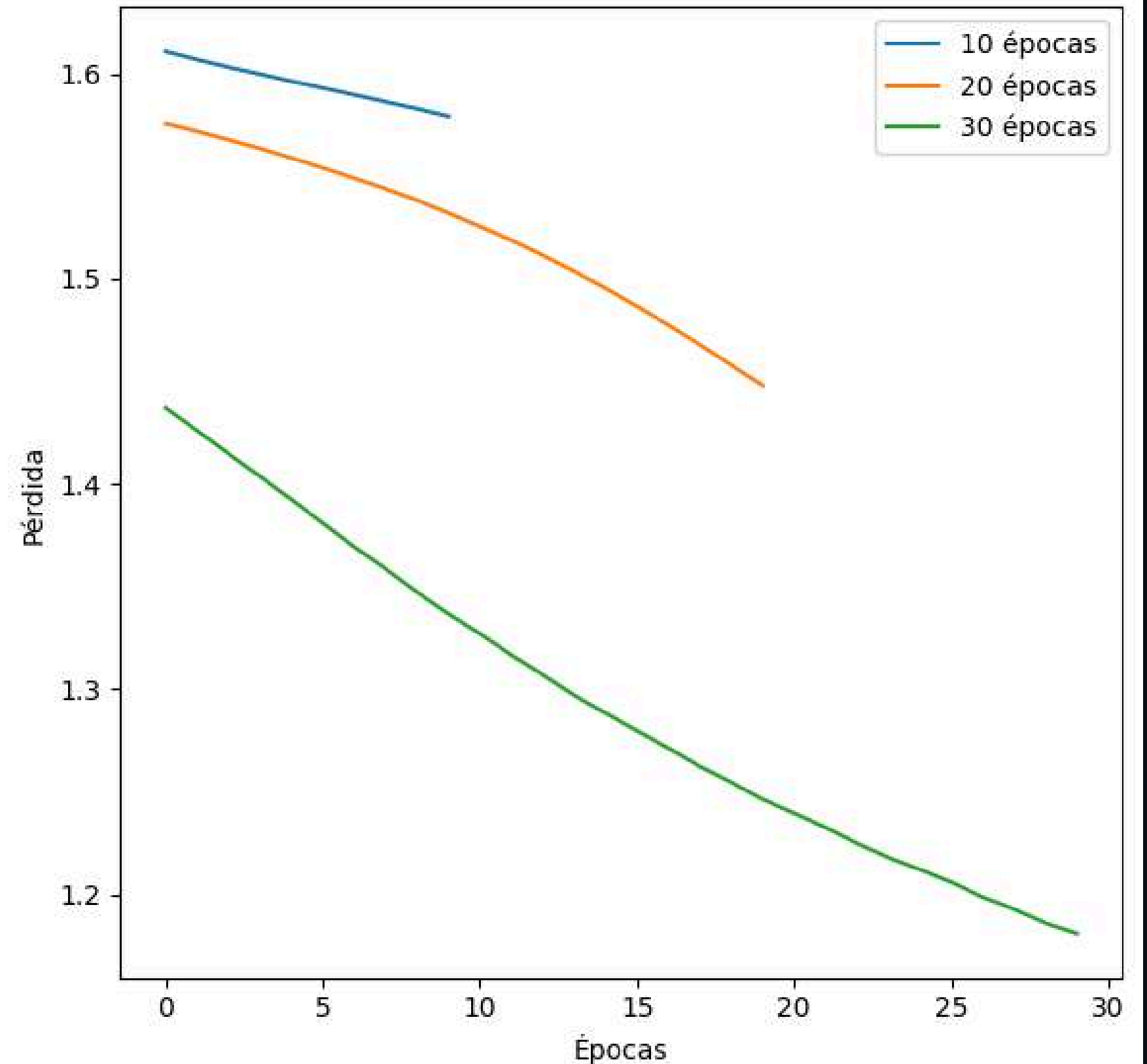
# DEEP LEARNING CON 3 CAPAS OCULTAS



Accuracy durante el entrenamiento



Pérdida durante el entrenamiento

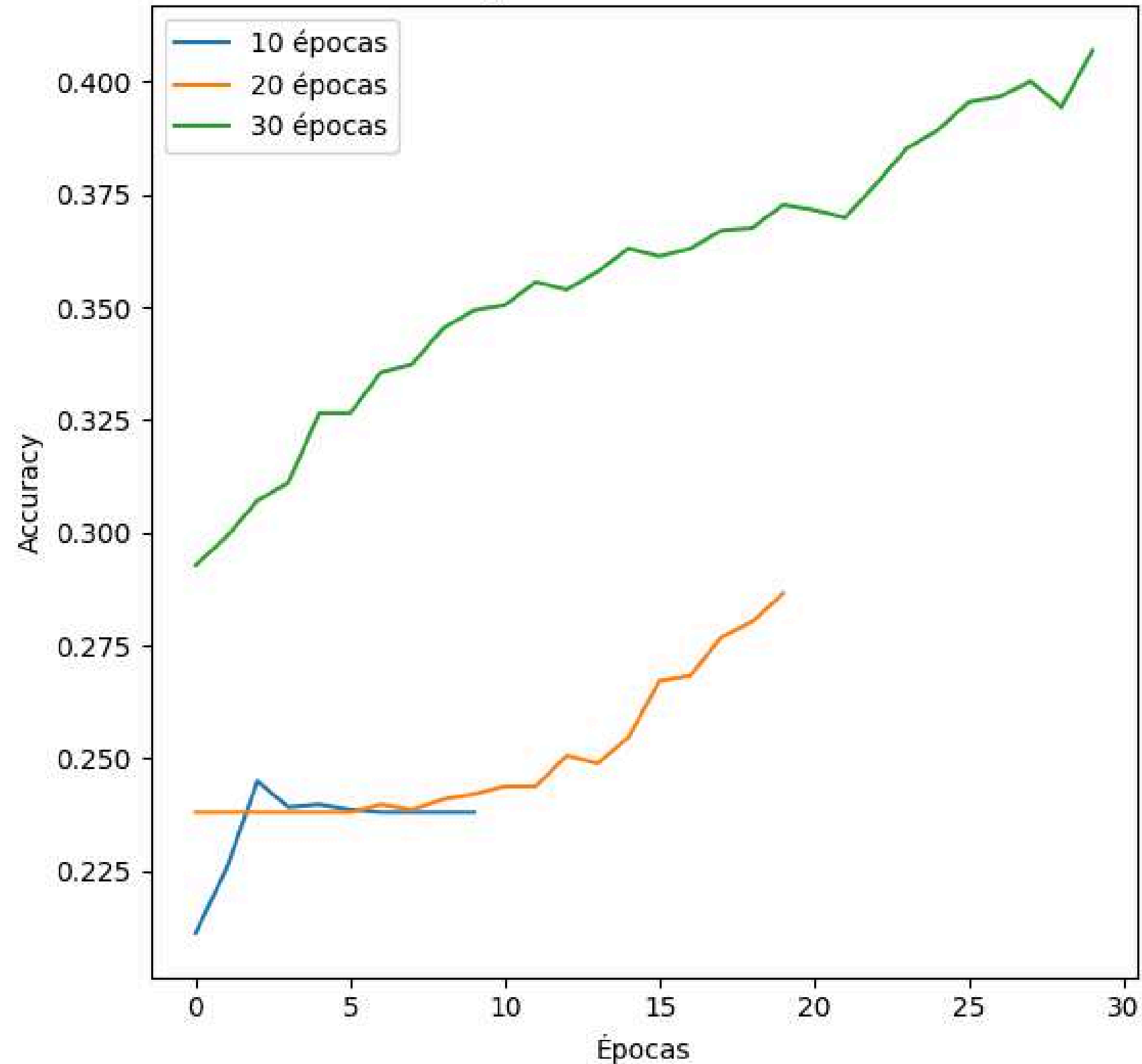




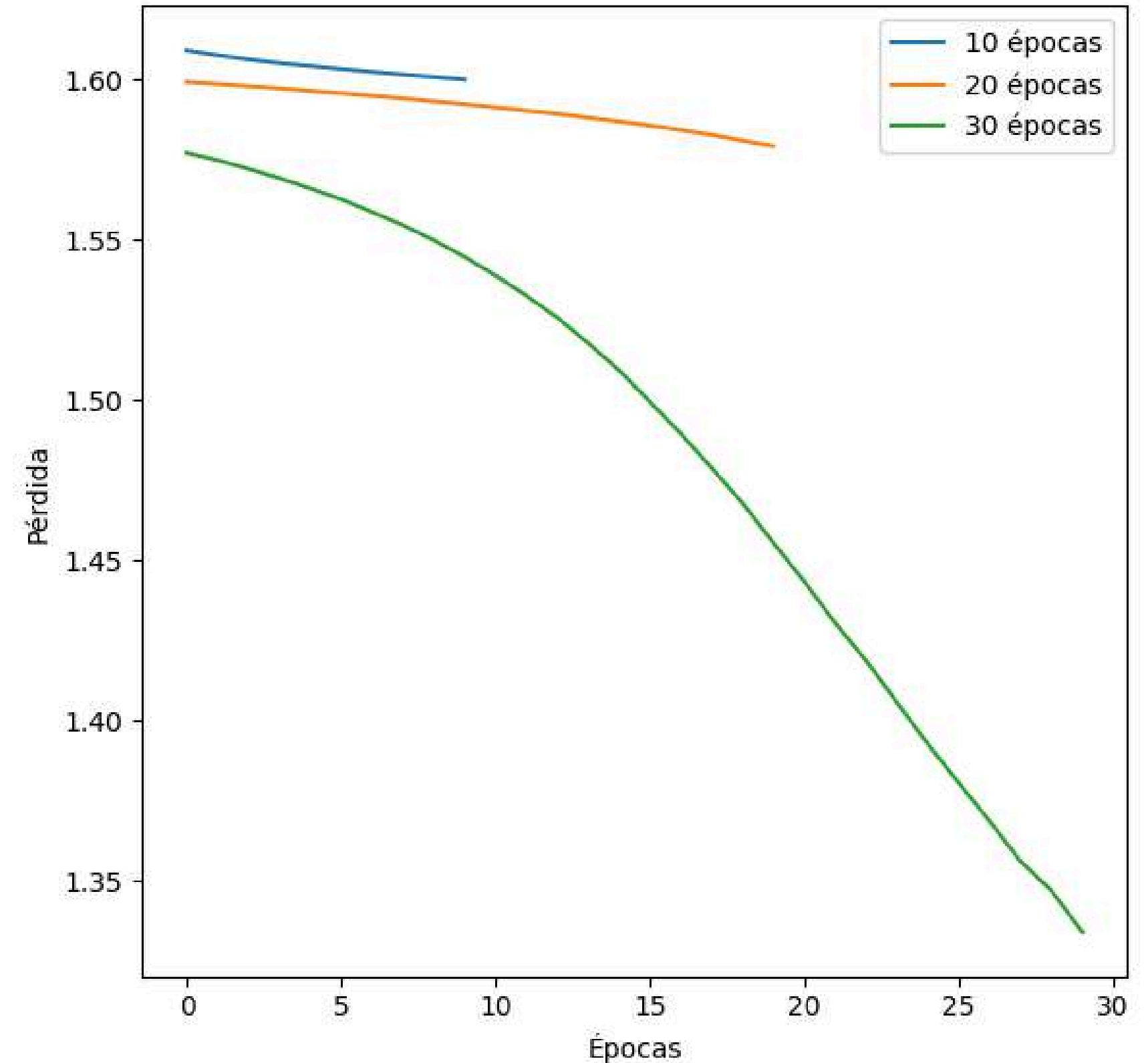
# DEEP LEARNING CON 6 CAPAS OCULTAS



Accuracy durante el entrenamiento



Pérdida durante el entrenamiento

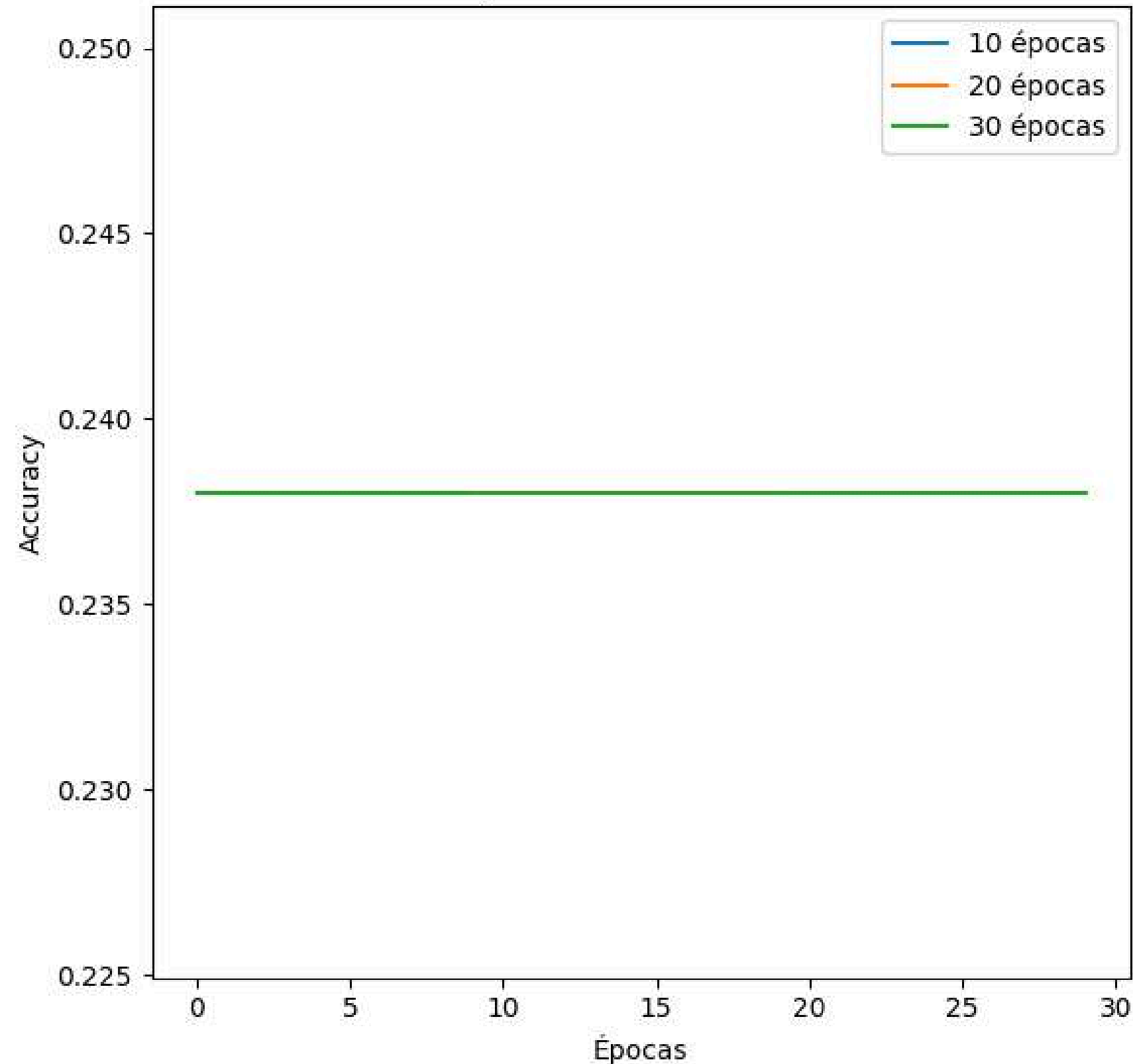




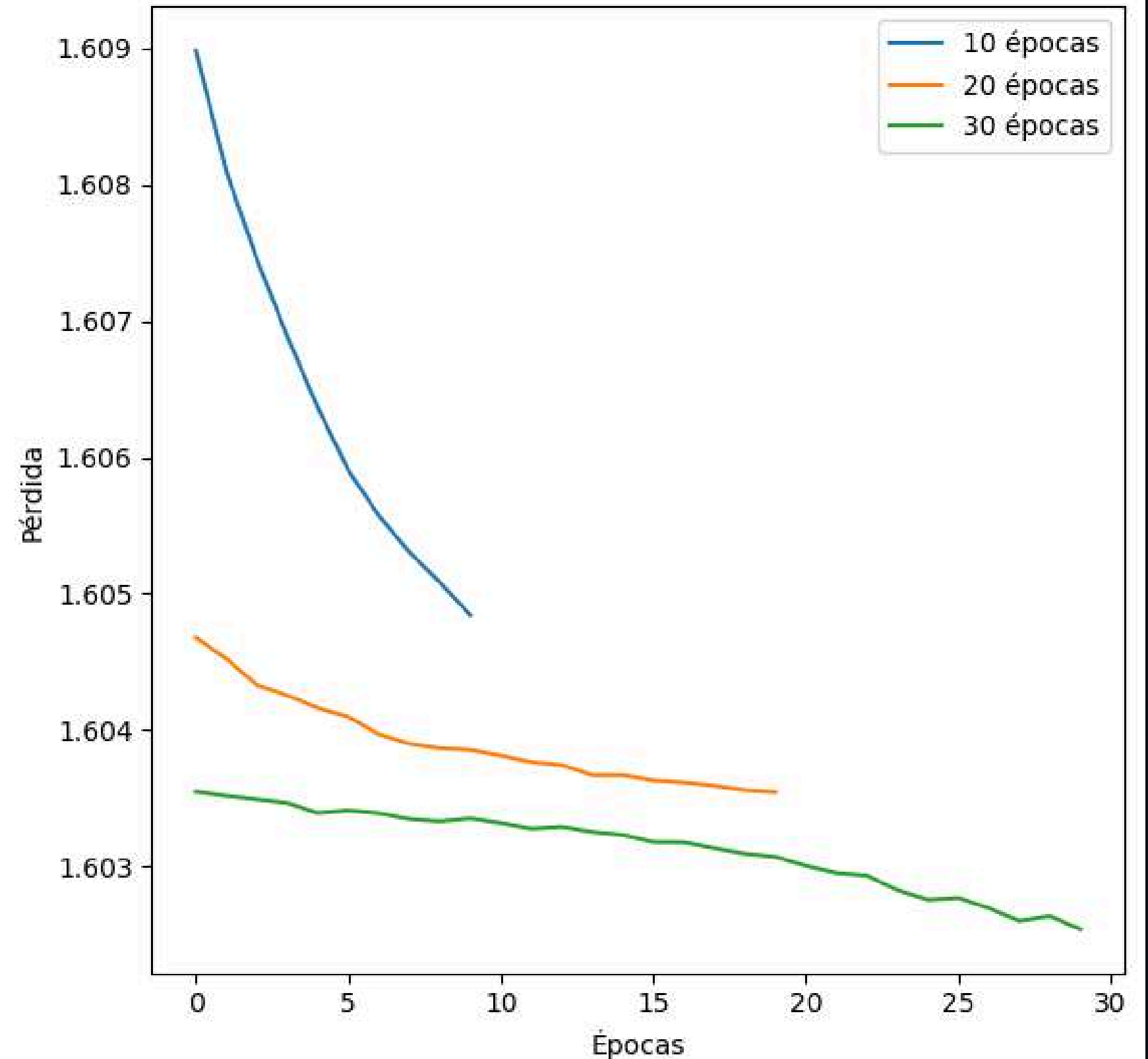
# DEEP LEARNING CON 10 CAPAS OCULTAS



Accuracy durante el entrenamiento



Pérdida durante el entrenamiento







# ***EJEMPLO 1: PLAYER POSITION***







## *Estadísticas del jugador:*

```
new_data = pd.DataFrame({  
    'Age': [39],  
    'Height': [6.7],  
    'PTS': [23],  
    'AST': [9.1],  
    'TRB': [8],  
})
```



## *Según el aprendizaje:*

1/1 — 0s 37ms/step  
La posición predicha para el jugador es: 2  
Posiciones posibles: [0 1 2 3 4]

- PG (Point Guard) → 0
- SG (Shooting Guard) → 1
- SF (Small Forward) → 2
- PF (Power Forward) → 3
- C (Center) → 4







# ***EJEMPLO 2: DREAM TEAM***





# ¿QUE SE HIZO?

La idea es seleccionar una alineación ideal de 5 jugadores (PG, SG, SF, PF, C) basada en las siguientes estadísticas clave de nuestro dataset

- **Estadísticas ofensivas:** (PTS), (AST).
- **Estadísticas defensivas:** (TRB), (STL), (BLK).
- **Eficiencia general:** EFG% (eficiencia en tiros de campo).
- **Restricción de posición:** Solo puede haber un jugador de cada posición (Pos = PG, SG, SF, PF, C).
- **Otras restricciones opcionales:** Límite de minutos jugados promedio (MP), balance ofensivo/defensivo.





**Funcion fitness:** Evalúa qué tan buena es una alineación, sumando las estadísticas normalizadas de los jugadores y penalizando alineaciones inválidas.

**Funcion crossover:** Combina dos alineaciones para generar nuevas, mezclando jugadores de ambas.

**Funcion Mutacion:** Realiza cambios aleatorios en una alineación para explorar nuevas soluciones.



# RESULTADOS:



## Mejor Alineacion:

	Player	Pos	PTS	AST	TRB	STL	BLK	eFG%
233	Tim Frazier	PG	3.7	3.3	1.9	0.3	0.1	0.372
657	Dennis Schröder	SG	13.5	4.6	3.3	0.8	0.1	0.490
1961	Trey Murphy III	SF	14.8	2.2	4.9	0.9	0.5	0.579
2107	Jeremy Sochan	PF	11.6	3.4	6.4	0.8	0.5	0.483
906	Thomas Bryant	C	12.1	0.7	6.8	0.3	0.6	0.691

Fitness de la mejor alineación:  
2.9675290200413356





***THANK YOU***