

# 1. Introducción

En este proyecto se desarrolla un modelo de programación lineal para determinar la cantidad máxima de puntos que un equipo necesita y aun así descender en el fútbol argentino. El modelo considera las reglas del torneo, incluyendo la tabla anual y los promedios históricos, y se utiliza la librería OR-Tools de Google para resolver el problema.

## Objetivo

Determinar los resultados necesarios en las fechas restantes que todos los equipos necesitan para que logren mantenerse en el torneo.

## 2. Formulación del Problema

### 2.1 Variables de Decisión

- **victoria [equipo, fecha]** (Binaria) = Vale 1 si el equipo **i** gana en la fecha **f**. 0 sino
- **empate [equipo\_local, fecha]** (Binaria) = Vale 1 si el equipo **i** empata en la fecha **f**. 0 sino
- **puntos\_en\_fecha [equipo, fecha]** = Puntos que hace el equipo **i** en la fecha **f**
- **puntos\_acumulados [equipo]** = Puntos acumulados por equipo en las fechas restantes.
- **total\_anual [equipo]** = Total de puntos de cada equipo al final del torneo
- **W\_tabla [equipo\_i, equipo\_j]** = Vale 1 si el **equipo\_i** supera al **equipo\_j** en puntos de la tabla final. 0 sino. para cada par (**equipo\_i**, **equipo\_j**) en el fixture.
- **W\_promedio [equipo\_i, equipo\_j]** = Vale 1 si el **equipo\_i** supera al **equipo\_j** en promedio al final de temporada. 0 sino. para cada par (**equipo\_i**, **equipo\_j**) en el fixture.
- **promedio\_temporada [equipo]** = promedio final de las últimas 3 temporadas.
- **B\_tabla[equipo]** = 1 si la sumatoria de **W\_tabla** para cada equipo  $\leq 1$ . 0 sino. (Esto nos indica si el equipo **i** desciende por puntos)
- **W\_promedio\_ajustado [equipo\_i, equipo\_j]** = Vale 1 solo si **W\_promedio** es 1 Y **B\_tabla** es 0. Esto "filtra" los equipos que ya descienden por tabla para el cálculo de quienes descienden por promedio.
- **B\_promedio[equipo]** = 1 si la sumatoria de **W\_promedio\_ajustado** para cada equipo  $\leq 1$ . 0 sino. (Esto nos indica si el equipo **i** desciende por promedio)

## 2.2 Restricciones

### 1. Cálculo de puntos por fecha para cada equipo:

$$puntos\_en\_fecha[equipo, fecha] = 3 * Victoria[equipo, fecha] + Empate[equipo, fecha]$$

### 2. Cálculo de puntos acumulados para cada equipo:

$$puntos\_acumulados[equipo] = \sum puntos\_en\_fecha[equipo, fecha]$$

### 3. Cálculo de puntos totales para cada equipo:

$$total\_anual[equipo] = puntos\_acumulados[equipo] + PTS_{24}$$

### 4. Cálculo de promedio totales para cada equipo:

$$promedio\_temporadas[equipo] = (pts + puntos\_acumulados[equipo]) / pj$$

### 5. Cálculo de W\_tabla:

$$total\_anual[equipo] - total\_anual[equipo2] \leq M * W\_tabla[equipo, equipo2]$$

$$total\_anual[equipo] - total\_anual[equipo2] \geq 1 - M * (1 - W\_tabla[equipo, equipo2])$$

### 6. Cálculo de W\_promedio:

$$promedio\_temporadas[equipo] - promedio\_temporadas[equipo2] \leq M * W\_promedio[equipo, equipo2]$$

$$promedio\_temporadas[equipo] - promedio\_temporadas[equipo2] \geq 0.000001 - M * (1 - W\_promedio[equipo, equipo2])$$

(El valor 0.000001 es un epsilon que se usa para evitar ambigüedades en el caso que los promedios sean iguales)

### 7. Cálculo de B Tabla:

$$suma\_W\_tabla \leq 1 + M * (1 - B\_tabla[equipo])$$

$$suma\_W\_tabla \geq 1 - M * B\_tabla[equipo]$$

### 8. Cálculo de W\_promedio\_ajustado:

$$W\_promedio\_ajustado[equipo, equipo2] \leq W\_promedio[equipo, equipo2]$$

$$W\_promedio\_ajustado[equipo, equipo2] \leq 1 - B\_tabla[equipo2]$$

$$W\_promedio\_ajustado[equipo, equipo2] \geq W\_promedio[equipo, equipo2] - B\_tabla[equipo2]$$

### 9. Cálculo de B\_promedio:

$$suma\_W\_promedio\_ajustado \leq 1 + M * (1 - B\_promedio[equipo])$$

$$suma\_W\_promedio\_ajustado \geq 1 + 0.000001 - M * B\_promedio[equipo]$$

#### 10. Un único resultado por partido:

$$victoria[equipo, fecha] + empate[equipo, fecha] \leq 1$$

#### 11. Si empata el local empata el visitante

$$empate[equipo\_local, fecha] - empate[equipo\_visitante, fecha] = 0$$

#### 12. Dependencia de Resultado

$$victoria[equipo\_local, fecha] + victoria[equipo\_visitante, fecha] + empate[equipo\_local, fecha] = 1$$

#### 13. No pueden descender los mismos por tabla y promedio

$$B\_tabla[equipo] + B\_promedio[equipo] \leq 1$$

#### 14. Asegurar que desciendan 2 por tabla y 2 por promedio

$$B\_tabla[equipo] \text{ for } equipo \text{ in } equipos = 2$$

$$B\_promedio[equipo] \text{ for } equipo \text{ in } equipos = 2$$

#### 15. Obligo al descenso de mi equipo a salvar por tabla o promedio

$$B\_tabla[equipo] + B\_promedio[equipo] = 1$$

## 2.3 Función Objetivo

Maximizar los puntos totales de equipo\_a\_salvar:

$$\text{Maximizar } total\_anual[equipo\_a\_salvar]$$

### 3. Desafíos en el proceso de desarrollo

- En un principio, se planteó que la variable **W** representara únicamente al equipo que se busca salvar del descenso. Sin embargo, al integrar los promedios, se identificó una limitación en este enfoque: no sería posible determinar cuáles son los otros tres equipos que descienden, además del equipo a salvar. Esto generaría un problema, ya que no se podría diferenciar si un equipo desciende por la tabla anual o por el promedio, incumpliendo con la consigna que establece que deben descender cuatro equipos (dos por tabla y dos por promedio).
- El equipo ajustó la solución para que las variables **W\_tabla** y **W\_promedio** se correspondieran correctamente con cada equipo. Sin embargo, esto generó un nuevo desafío: filtrar a los equipos que ya habían descendido por tabla para evitar que un equipo descendiera tanto por tabla como por promedio.

Para resolverlo, incorporaron una variable auxiliar denominada **W\_promedio\_ajustado**, que toma el valor de 1 solo si **W\_promedio** es igual a 1 y **B\_tabla** es igual a 0. Esto permitió excluir a los equipos que ya habían descendido por tabla del cálculo de **B\_promedio**.

- La implementación de estas soluciones requirió añadir más variables al modelo, lo que incrementó significativamente el tiempo de ejecución, llegando a aproximadamente 30 minutos por corrida completa. Además, se observaron pequeñas inconsistencias en las ejecuciones, posiblemente debido al redondeo de decimales. También se notó que cambiar el valor de **M** introducía ligeras variaciones en los resultados, lo que podría influir en la precisión final del modelo.

### 4. Resultados

El equipo de trabajo utilizó el Solver para determinar la mayor cantidad de puntos que un equipo puede alcanzar y aún así permanecer en zona de descenso. A partir de este resultado, definieron la lógica para calcular la cantidad mínima de puntos necesarios para evitar el descenso.

La metodología final consistió en ordenar los puntos de cada equipo en la tabla y comparar los del equipo objetivo con los del equipo inmediatamente superior. De esta forma, si el equipo en cuestión supera al equipo siguiente en puntos, logra evitar el descenso. Por ejemplo, en una tabla simplificada de los últimos resultados, considerando que descienden únicamente dos equipos por puntos:

Equipo a salvar: Belgrano

- Sarmiento: 56 pts
- Belgrano: 58 pts
- Racing: 60 pts

Aquí Belgrano debería hacer como mínimo 61 puntos para poder salvarse ya que si hace 59 o 60 todavía estaría por debajo o empatado con Racing.

Al correr el programa completo nos dieron los siguientes resultados:

(El valor de la Función Objetivo representa al equipo que se encuentra por debajo)

```
El equipo River Plate no tiene forma de descender.
Valor de la F0: 52.0
El equipo Racing Club necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Boca Juniors necesita al menos 53.00 puntos para salvarse.
El equipo Estudiantes (LP) no tiene forma de descender.
El equipo Godoy Cruz no tiene forma de descender.
Valor de la F0: 52.0
El equipo Argentinos necesita al menos 53.00 puntos para salvarse.
El equipo Talleres (C) no tiene forma de descender.
Valor de la F0: 52.0
El equipo Huracan necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Def y Justicia necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo San Lorenzo necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Newells necesita al menos 53.00 puntos para salvarse.
El equipo Velez no tiene forma de descender.
Valor de la F0: 52.0
El equipo Instituto necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Independiente necesita al menos 55.00 puntos para salvarse.
Valor de la F0: 51.999999999999999
El equipo Belgrano necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Atl Tucuman necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Gimnasia (LP) necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Rosario Central necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 59.0
El equipo Union necesita al menos 61.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Riestra necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Barracas Central necesita al menos 54.00 puntos para salvarse.
Valor de la F0: 58.0
El equipo Platense necesita al menos 59.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Banfield necesita al menos 54.00 puntos para salvarse.
Valor de la F0: 52.0
El equipo Tigre necesita al menos 53.00 puntos para salvarse.
Valor de la F0: 61.0
El equipo Lanus necesita al menos 66.00 puntos para salvarse.
Valor de la F0: 58.0
El equipo Central Cba (SdE) no puede salvarse por si mismo
Valor de la F0: 53.0
El equipo Sarmiento (J) no puede salvarse por si mismo
Valor de la F0: 52.0
El equipo Ind Rivadavia necesita al menos 54.00 puntos para salvarse.
```

## 5. Conclusiones

- El equipo sostiene firmemente que el modelo planteado permite identificar de manera correcta los puntos mínimos necesarios para que cada equipo evite el descenso. No obstante, reconocen que la complejidad del problema puede generar pequeños errores que podrían alterar ligeramente algunos resultados.

## 6. Anexos

- **Código Fuente:** proyecto\_torneo\_argentino.py
- **Datos de Entrada:** promedios.csv, fixture.csv