



Ciclo 1 Fundamentos de Programación

Reto 2: Calculadora de probabilidades equipos de fútbol



Un portal de apuestas deportivas, ha identificado una serie de factores clave para calcular de manera relativamente precisa, las probabilidades de éxito que tendrían equipos de fútbol profesionales reconocidos, al participar en ediciones especiales de torneos internacionales europeos, en los cuales, se invitan equipos relevantes sudamericanos y africanos. El portal, a través de analítica de datos sobre los históricos y curvas de rendimiento de los equipos, ha determinado que los siguientes factores son esenciales para realizar pronósticos acertados en la mayoría de los casos. Estos factores son:

- **Continente** donde se encuentra el proyecto deportivo del equipo.
- **Partidos Ganados.**
- **Partidos Empatados.**
- **Partidos Perdidos.**
- **Goles a Favor.**
- **Goles en Contra.**
- **Diferencia de Goles**, es decir, la diferencia entre los goles a favor y los goles en contra.
- **Posesión del Balón**, porcentaje de posesión, valor continuo entre 0 y 1.
- **Ganador**, cuántas veces el equipo ha ganado el torneo.
- **Valor de la Nómina.**

En el estudio realizado, estos factores o variables han sido relacionados, y el siguiente árbol de decisiones muestra la forma cómo se obtiene el porcentaje de probabilidad de éxito del equipo:

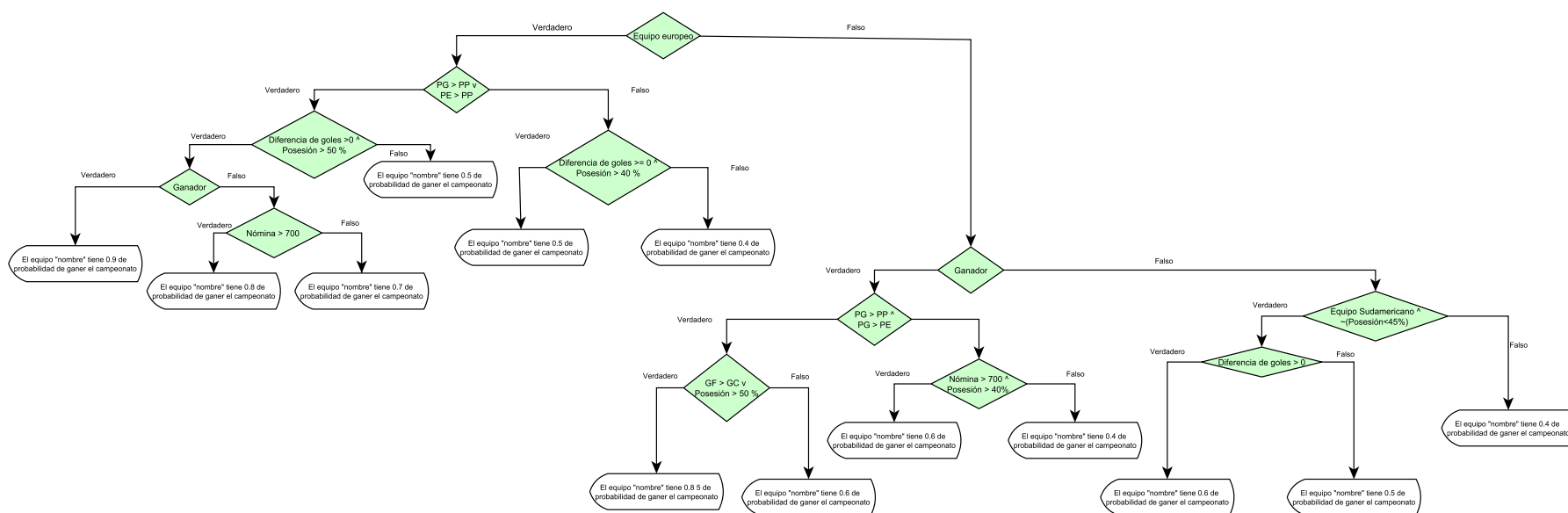


Figura 1. Árbol de decisiones, calculadora de probabilidades



Se requiere entonces, implementar una función que recibiendo las estadísticas recolectadas de un equipo, y encapsuladas en un diccionario, retorne las probabilidades de éxito dentro del torneo, generando un mensaje de la forma **“El equipo {nombre} tiene 0.6 de probabilidad de ganar el campeonato”**, donde el porcentaje corresponde a la clasificación que ha determinado el árbol de decisiones que se muestra en la Figura 1.

El esqueleto o prototipo de la función es el siguiente:

```
def prediccion(equipo:dict) -> str:  
    "Desarrollar aquí la calculadora de probabilidades"  
    pass
```

En la siguiente tabla, se especifican los parámetros (campos, atributos o ítems) que llegan encapsulados en el diccionario **equipo**. Los nombres de dichos parámetros no tienen tildes, porque deben seguir las normas de nombramiento de las variables en Python, adicionalmente, se han omitido las tildes en el contenido de los campos para evitar inconvenientes de codificación.

Entradas

Ítem	Tipo de Dato	Descripción
Nombre	str	Cadena con el nombre del equipo
Continente	str	Cadenas “Europa”, “Sudamerica” o “Africa”
PG	int	Valor entero mayor o igual a 0
PE	int	Valor entero mayor o igual a 0
PP	int	Valor entero mayor o igual a 0
GF	int	Valor entero mayor o igual a 0
GC	int	Valor entero mayor o igual a 0
Posesion	float	Valor flotante (decimal) entre 0 y 1
Ganador	int/str	Valores entre: 0, 1, 2, “3+”
Nomina	int	Valor entero indicando un estimado de la nómina en millones de euros variando entre 0 y 3000

Salida

Tipo de Retorno	Descripción
str	“El equipo {nombre} tiene X.X de probabilidad de ganar el campeonato”

Nótese que la probabilidad de respuesta se ha generalizado como *X.X*, porque no proviene de una variable, si no que corresponde a la rama del árbol de decisiones (Figura 1) donde ha sido clasificado el equipo.

A continuación se presentan 4 casos de prueba (ejemplos), mostrando cómo debe responder la función solicitada a unas entradas específicas:



Caso de Prueba 1:

Diccionario <i>equipo</i>	
Llave	Valor
nombre	"Manchester"
PG	5
PE	3
PP	2
Ganador	2
GF	38
GC	20
Nomina	800
Continente	"Europa"
Posesion	0.52

Llamado:

```
print(prediccion({"nombre":"Manchester", "PG" : 5, "PE": 3, "PP": 2, "Ganador":  
2, "GF": 38, "GC": 20, "Nomina": 800, "Continente": "Europa", "Posesion": 0.52}))
```

Resultado esperado:

```
El equipo Manchester tiene el 0.9 de probabilidad de ganar el torneo
```

Caso de Prueba 2:

Diccionario <i>equipo</i>	
Llave	Valor
nombre	"Boca"
PG	5
PE	2
PP	3
Ganador	"3+"
GF	20
GC	29
Nomina	600
Continente	"Sudamerica"
Posesion	0.45

Llamado:

```
print(prediccion({"nombre":"Boca", "PG" : 5, "PE": 2, "PP": 3, "Ganador": "3+",  
"GF": 20, "GC": 29, "Nomina": 600, "Continente": "Sudamerica", "Posesion":  
0.45}))
```



Resultado esperado:

```
El equipo Boca tiene el 0.4 de probabilidad de ganar el torneo
```

Caso de Prueba 3:

Diccionario equipo	
Llave	Valor
nombre	"Zamalek"
PG	5
PE	2
PP	3
Ganador	0
GF	20
GC	29
Nomina	600
Continente	"Africa"
Posesion	0.45

Llamado:

```
print(prediccion({"nombre":"Zamalek", "PG" : 5, "PE": 2, "PP": 3, "Ganador": 0,  
"GF": 20, "GC": 29, "Nomina": 600, "Continente": "Africa", "Posesion": 0.45}))
```

Resultado esperado:

```
El equipo Zamalek tiene el 0.4 de probabilidad de ganar el torneo
```

Caso de Prueba 4:

Diccionario equipo	
Llave	Valor
nombre	"Manchester City"
PG	5
PE	3
PP	2
Ganador	0
GF	38
GC	20
Nomina	800
Continente	"Europa"
Posesion	0.52



Llamado:

```
print(prediccion({"nombre": "Manchester City", "PG" : 5, "PE": 3, "PP": 2,  
"Ganador": 0, "GF": 38, "GC": 20, "Nomina": 800, "Continente": "Europa",  
"Posesion": 0.52}))
```

Resultado esperado:

El equipo Manchester City tiene el 0.8 de probabilidad de ganar el torneo

Recordar: En la plataforma debe subirse una función con **el mismo nombre, la misma cantidad de argumentos, y el retorno debe tener exactamente la estructura de la salida que se presenta en este documento.**