Artículo Original Pág. 13 - 34 ISSN 2415-2323

ILUMATE VOLUMEN 12

VISUALIZACIÓN Y ANÁLISIS DE LOS JUGADORES DEL FIFA 2021 APLICANDO ALGORITMOS DE MACHINE LEARNING

Visualization and Analysis of FIFA 2021 Teams Applying Machine Learning Algorithms

Albert Omar Araya Cory
albertaraya944@gmail.com
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle- Bolivia

José Antonio Aguilar Fernández
jaaguilarfernandez@outlook.es
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle- Bolivia

Resumen

El objetivo de la presente investigación es el de realizar un análisis comparativo sobre las características de los jugadores de futbol del FIFA entre los años 2020 – 2021 y la vida real. Para el desarrollo de esta investigación se utilizó dos Dataset del videojuego. Se aplicó métodos de visualización de datos con Python y se realizó un análisis para la detección de anomalías mediante algoritmos de aprendizaje no supervisado.

Palabras clave

Pandas, Numpy, Seaborn, Scikit-Learn, Matplotlib, Dataset, Video juego, Clustering, Futbol, Python

Abstract

The objective of this research is to carry out a comparative analysis on the characteristics of FIFA soccer players between the years 2020 - 2021 and real life. For the development of this research, two videogame datasets were used. Data visualization methods were applied with Python and an analysis was carried out for the detection of anomalies using unsupervised learning algorithms.

Key words

Pandas, Numpy, Seaborn, Scikit-Learn, Matplotlib, Dataset, Video Game, Clustering, Soccer, Python

Introducción

La Federación Internacional de Fútbol Asociación (FIFA) es la institución que gobierna a todas las federaciones de futbol del planeta, esta institución cada año brinda un informe acerca del rendimiento de los jugadores registrados, estos Datasets son de carácter público y ayudan al desarrollo del Big Data del futbol. El aprendizaje no supervisado es un método de aprendizaje automático donde un modelo se ajusta a las observaciones.

El siguiente proyecto presenta un dataset que contiene datos relevantes y específicos acerca de los jugadores de futbol registrados en la FIFA hasta el año 2021, también se visualizara el club al que pertenecen, de esta manera estaremos realizando análisis en profundidad y visualización de estos datos, además de aplicar aprendizaje supervisado para la regresión lineal y para la detección de anomalías mediante tareas de Clustering.

Problemática

Los equipos de hoy tienen jugadores excelentes que benefician a su equipo en cada partido que se juega y de esta manera muchas veces algunos de

estos jugadores son infravalorados y algunos otros sobrevalorados. Por esta razón al momento de contratar a un nuevo integrante existe la probabilidad de que el jugador beneficie o no al club al que se está integrando.

Objetivo General

Analizar los Datasets completos, para determinar cuáles son los jugadores infravalorados y sobrevalorados en base a distintas características que se obtendrá a través de un análisis profundo utilizando herramientas de machine Learning.

Los objetivos específicos son: Mediante un análisis exploratorio realizar gráficos de visualización. Detectar anomalías. Analizar y explicar los distintos gráficos obtenidos.

Referentes conceptuales

FIFA ha sido un videojuego al que atrae muchas personas en todo el mundo y ha aparecido en muchas ediciones. Su último lanzamiento será en octubre de este año, es decir, FIFA 2021. Este conjunto de datos contiene las características de los jugadores en el juego.

El conjunto de datos contiene datos de la clasificación de los jugadores, su edad, su nacionalidad, la posición en la que juegan y su potencial de crecimiento en el juego. Es posible que los datos de algunos jugadores y sus clubes no sean muy precisos, ya que la ventana de transferencia aún está abierta y es posible que se realicen cambios en etapas posteriores. (Mishra, 2020).

DATASETS

Para este proyecto se utilizaron tres Dataset en los cuales se trabajó y se creó un dataset único ya normalizado listo para ser analizado y visualizado de una manera correcta aplicando los distintos algoritmos.

Dataset 1

fifa-FUT-Data fue el primer dataset que se utilizó y se normalizó para las funciones necesarias al principio de las pruebas del proyecto. (Kafagy, 2019).

Dataset 2

players_21 fue el segundo dataset que se utilizós para poder realizar una actualización mas reciente respecto a las características de los jugadores. (Leone, 2020).

Dataset 3

AprendizajeAutomatico dataset combinado de los dos anteriores dataset el que fue normalizado e implementado en el proyecto. (AlbertAC, 2020).

Análisis y Visualización de los datos de los jugadores del FIFA 2021

Actualmente cada equipo fútbol tiene a un jugador estrella que según los entrenadores son considerados el mejor, entonces los espectadores comienzan a pensar el ¿por qué? estos jugadores se consideran los mejores.

Surgen varias preguntas al respecto, para analizar detalles de muchos de estos jugadores, se comienza a formar varios conceptos, analizando elementos como: si es el potencial, la edad o es el salario que sustentan, hace que estos jugadores sean considerados los mejores.

De esta manera se ha propuesto a analizar a varios jugadores junto a sus equipos y ver que tanto es el aporte de estos según la posición en la que juega, el potencial que tiene, el valor que tiene en el mercado y el pago actual que recibe por temporadas de juego.

Métodos.

Importación de las librerías

Para realizar el análisis y la parte visual gráfica se necesita de algunas librerías necesarias como **numpy**, **pandas**, **matplotlib, seaborn** y **scikit-learn**.

Figura 1. Importación de Librerías

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.cluster import KMeans
import seaborn as sns
from TeamClass import Team
```

Fuente: Elaboración propia

Importación del Dataset ya normalizado

Después de importar las librerías es necesario importar el Dataset ya normalizado para que de esta manera se tenga una visualización en tablas, así como en filas y columnas.

Tabla 1. Dataset normalizado

	sofifa_id	first_position	second_position	third_position	hits	team	height_cm	weight_kg	value_eur	wage_eur	..
0	158023	ST	CF	RW	299	FC Barcelona	170	72	95.5	565.0	..
1	20801	ST	LW	NaN	276	Juventus	187	83	58.5	405.0	..
2	190871	CAM	LW	NaN	186	Paris Saint-Germain	175	68	105.5	290.0	..
3	203376	CB	NaN	NaN	127	Liverpool	193	92	78.0	200.0	..
4	200389	GK	NaN	NaN	47	AtlÃ□Â©tico Madrid	188	87	77.5	125.0	..

.	shooting	passing	dribbling	defending	physic	name	age	nationality	potential	overall
.	92.0	92.0	96.0	39.0	66.0	L. Messi	33	Argentina	94	94
.	93.0	82.0	89.0	35.0	78.0	Cristiano Ronaldo	35	Portugal	93	93
.	85.0	87.0	95.0	32.0	58.0	Neymar Jr	28	Brazil	92	92
.	60.0	70.0	71.0	90.0	86.0	V. van Dijk	29	Netherlands	92	91
.	NaN	NaN	NaN	NaN	NaN	J. Oblak	27	Slovenia	93	91

Fuente: (Elaboración propia)

En la presente tabla se obsera algunas caracterisitcas que se encontró en el Dataset

Definiendo funciones para encontrar al peor y al mejor jugador

Figura 2. Funciones para encontrar al peor y al mejor jugador

```
def searchWorstTeam(team,pos):
    Worst = team[team.first_position == pos][team.potential==team.potential[team.first_position == pos].min()]
    return Worst
def searchBestPlayer(team,pos):
    BestPlayer = team[team.first_position == pos][team.potential==team.potential[team.first_position == pos].max()]
    return BestPlayer
```

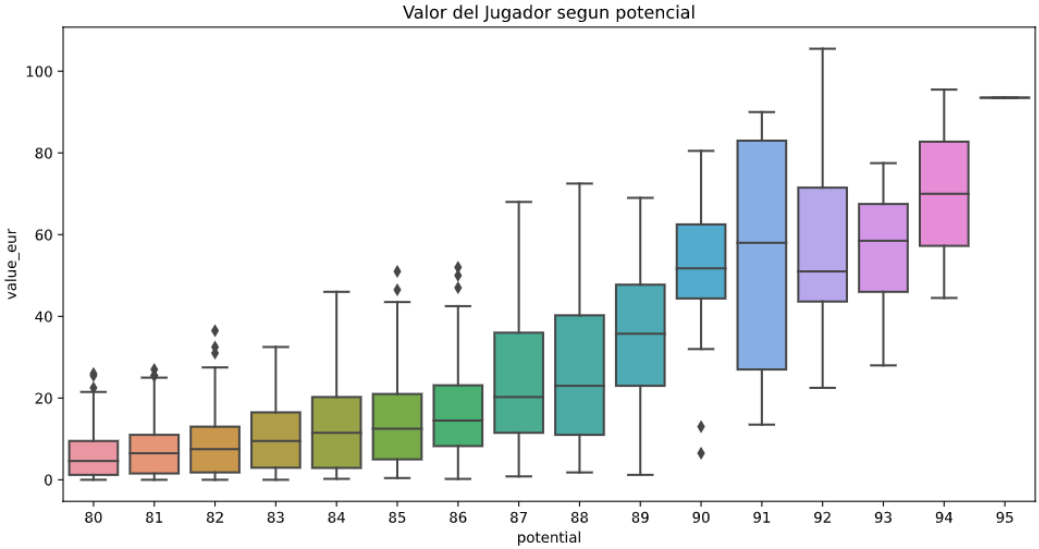
Fuente: Elaboración propia

Visualizaciones del Dataset

Gráfico de comparación del potencial y el valor del mercado del jugador en Euros

Seleccionamos los jugadores que tengan un potencial arriba de 80 y vemos el valor que tienen según su potencial.

Figura 3. Gráfico de comparación del potencial y el valor del mercado del jugador en Euros.



Fuente: (Elaboración propia)

Observaciones

En el gráfico podemos observamos que existen jugadores que están evaluados de una manera errónea ya que algunos valen más, pero no tienen el mayor potencial.

Gráfico de comparación entre el potencial y el sueldo del jugador en Euros

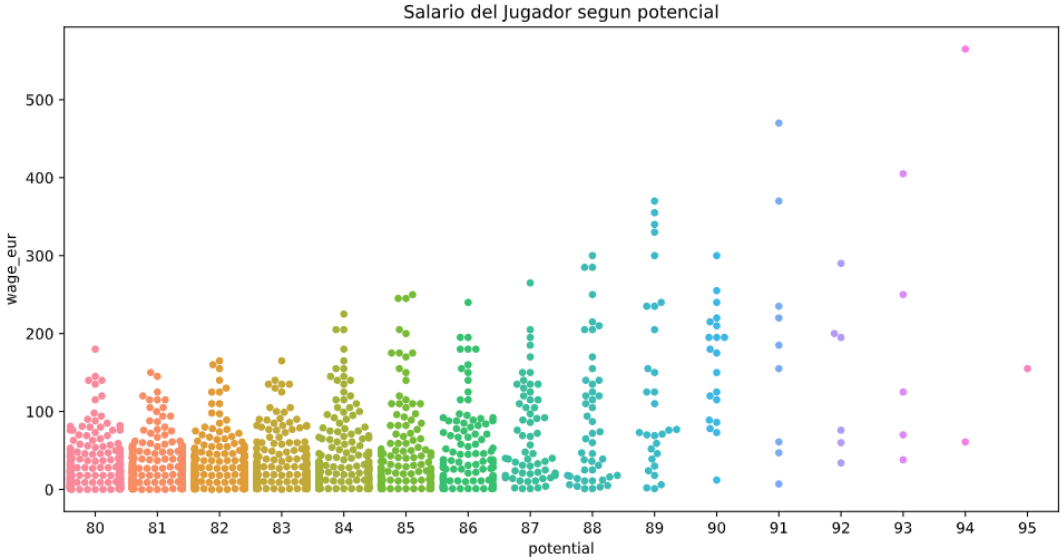
Seleccionamos los jugadores que tengan un potencial arriba de 80 y vemos el valor que tienen según su potencial

Figura 4. Vemos el valor que tienen según su potencial.

```
x = players_dataset.potential[players_dataset.potential>=80]
y = players_dataset.wage_eur[players_dataset.potential>=80]
plt.figure(figsize=(12,6))
sns.swarmplot(x,y)
plt.title('Salario del Jugador segun potencial')
```

Fuente: Elaboración propia

Figura 5. Gráfico de comparación entre el potencial y el salario del jugador en Euros.



Fuente: Elaboración propia

Observaciones

Se puede observar que existen algunas anomalías en la visualización de los datos, en primer lugar, existen jugadores que perciben un sueldo mayor teniendo un potencial menor que el de otros jugadores. Estos datos reflejan que existen jugadores sobrevalorados en el mercado.

Gráfico de comparación entre la posición el sueldo de los jugadores

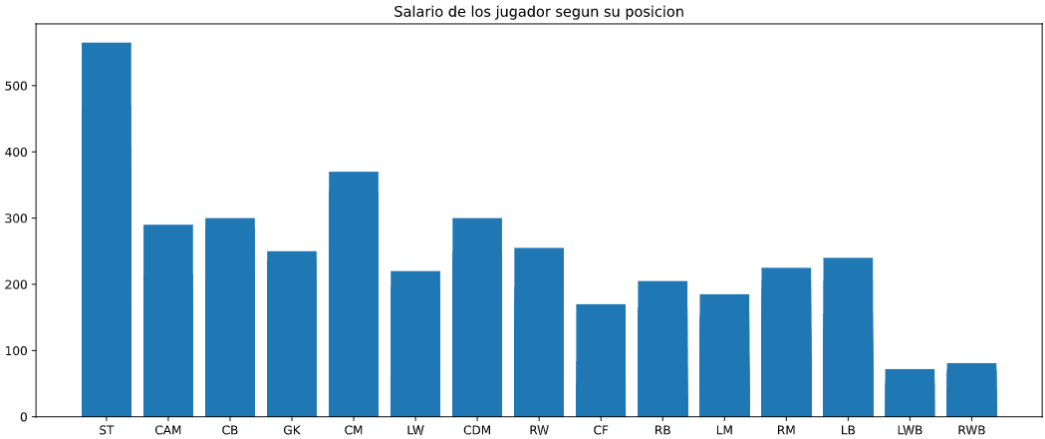
Seleccionamos los jugadores que tengan un potencial arriba de 80 y vemos el valor que tienen según su posición.

Figura 6. Vemos el valor que tienen según su posición.

```
y = players_dataset.wage_eur[players_dataset.potential>=80]
x = players_dataset.first_position[players_dataset.potential>=80]
plt.figure(figsize=(15,6))
plt.bar(x,y)
plt.title('Salario de los jugador segun su posicion')
```

Fuente: (Elaboración propia)

Figura 7. Gráfico de comparación entre la posición el salario de los jugadores



Fuente: Elaboración propia

Observaciones

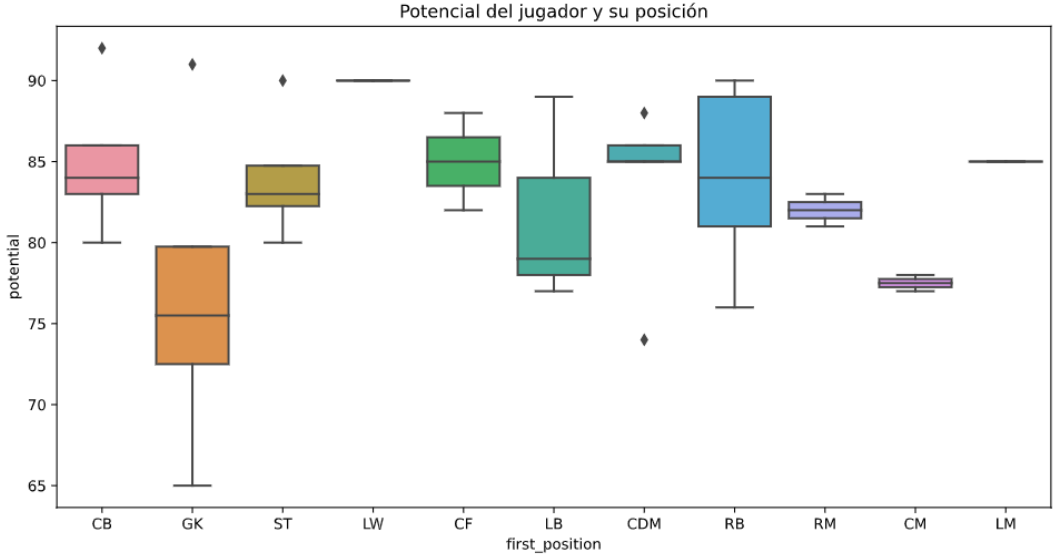
Según el gráfico podemos ver que los delanteros son los mejores pagados en las plantillas seguidos de los volantes de creación.

Figura 8. Análisis de la composición de un equipo de futbol

```
name = 'Liverpool'
players_team = players_dataset[players_dataset.team == name]
y = players_team.potential
x = players_team.first_position
plt.figure(figsize=(12,6))
sns.boxplot(x,y)
plt.title('Potencial del jugador y su posición')
```

Fuente: Elaboración propia

Figura 9. Gráfico de comparación entre el potencial y la posición del jugador.



Fuente: Elaboración propia

Observaciones

Según el gráfico se puede observar que existen jugadores que no aportan

mucho en el equipo, caso en el que encontramos a un GK (Portero) que tiene un rendimiento demasiado bajo.

Analizando cual es jugador con el potencial bajo

Figura 10. Analizando cual es jugador con el potencial bajo

```
pos = 'GK'
worst = searchWorstTeam(players_team,pos)
worst[['name','nationality','first_position','potential','value_eur','wage_eur']]
```

Fuente: Elaboración propia

Tabla 2. Extracción del peor jugador.

	name	nationality	first_position	potential	value_eur	wage_eur
10562	A. Lonergan	England	GK	65	0.15	8.0

Fuente: Elaboración propia

Figura 11. A. Lonergan según el análisis de datos el peor jugador del dataset y esta muy infravalorado

Fuente: (BeSoccer, 2020)

Aplicación de la Regresión Lineal

Declaración de hipótesis

Figura 12. Funcion de la hipótesis.

```
def h(X,theta):
    return np.dot(np.transpose(theta),X)
```

Fuente: Elaboración propia

Declaración del costo

Figura 13. Función consto

```
def J(X,y,theta):
    m = X.shape[1]
    return (1/(2*m))*np.sum(np.square(h(X,theta)-y))
```

Fuente: Elaboración propia

Declaración de la derivada de la función de costo

Figura 14. Derivada de la función

```
def dJ(X,y,theta):
    m=X.shape[1]
    return (1/m)*np.dot((h(X,theta)-y),np.transpose(X)).reshape((-1,1))
```

Fuente: Elaboración propia

Declaración de x como la columna que almacenara los valores de los jugadores y del potencial de los mismos

Figura 15. Almacenamiento de valores de los jugadores.

```
x = players_dataset[['value_eur']]
y = players_dataset.potential
x.to_numpy()
y.to_numpy()
y = np.array(y).reshape(1,-1)
x = np.array(x).reshape(1,-1)
```

Fuente: Elaboración propia

Declaración de Alpha y las iteraciones del modelo

Figura 16. Pruebas para determinar su almacenamiento.

```
alpha = 0.01
iterations = 400
```

Fuente: Elaboración propia

Declaración theta

Figura 17. Theta como random. Fuente: (Elaboración propia)

```
theta = np.random.random((X.shape[0],1))
theta
```

```
array([[0.58673882],
       [0.00666722]])
```

Fuente: Elaboración propia

Obtención del costo inicial

Figura 18. Obtención del costo inicial.

```
registro_grads = []
registro_J = []
J_init = J(X,y,theta)
print(f'Costo inicial:{J_init}')
```

```
Costo inicial:2573.490966374446
```

Fuente: Elaboración propia

Obtención del costo final

Figura 19. Obtención del costo final

```
for i in range(iterations):
    dtheta = dJ(X,y,theta)
    theta = theta-alpha*dtheta
    registro_J.append(J(X,y,theta))
print(f'Costo Final:{J(X,y,theta)}')
```

```
Costo Final:14.968570365001174
```

Fuente: Elaboración propia

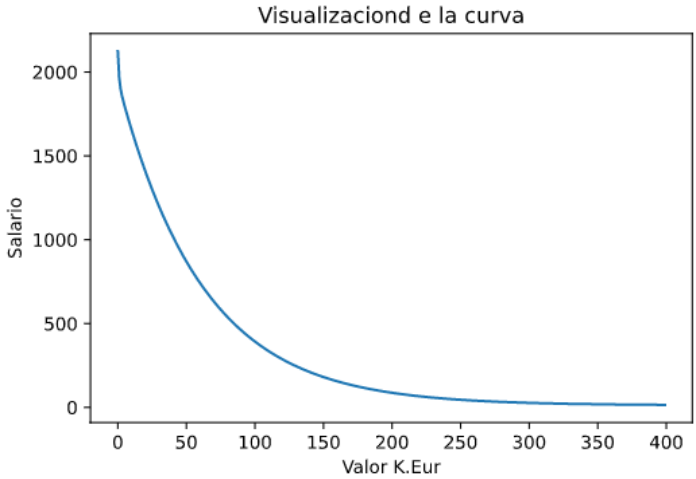
Visualización de la curva

Figura 20. Impresión de la curva.

```
it = list(range(iterations))
plt.plot(it,registro_J)
plt.title('Visualizaciond e la curva')
plt.xlabel('Valor K.Eur')
plt.ylabel('Salario')
plt.show()
```

Fuente: Elaboración propia

Figura 21. Gráfica de una curva que haciende de 0 2000.



Fuente: Elaboración propia

Observaciones

Se puede observar que existe una constante entre el valor del jugador y el salario que persive.

Visualización de datos para una tarea de detección de anomalías

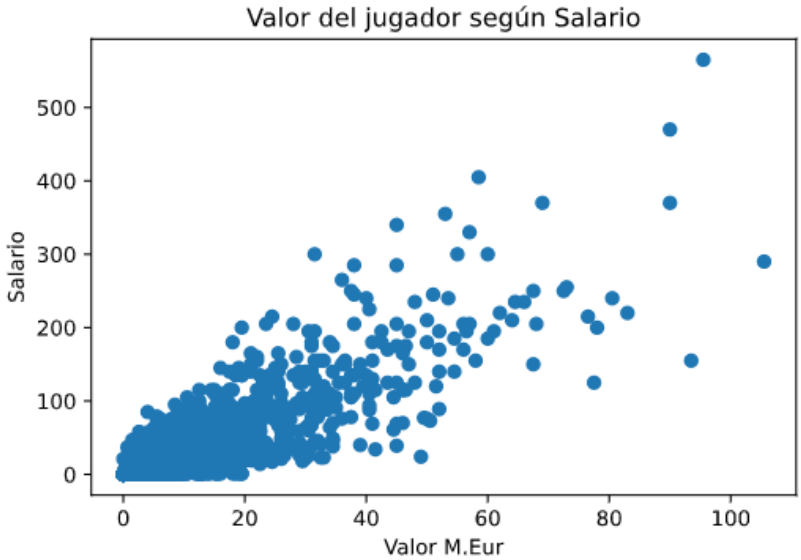
Gráfico del valor del mercado y el sueldo de los jugadores

Figura 22. Imprimimos las anomalias que existen.

```
x = players_dataset.value_eur[players_dataset.potential>=75]
y = players_dataset.wage_eur[players_dataset.potential>=75]
plt.title('Valor del jugador según Salario')
plt.xlabel('Valor M.Eur')
plt.ylabel('Salario')
plt.scatter(x,y)
```

Fuente: Elaboración propia

Figura 23. Valor del jugador según salario.



Fuente: Elaboración propia

Observaciones

Podemos observar que existen un jugadores que no cumplen con la constante, por lo cual realizaremos una tara de clustering para ver cual es la tendencia.

Aplicando Clustering al gráfico del valor del mercado y el sueldo de los jugadores

Figura 24. Imprimimos el clustering del grafico.

```
X = pd.concat([x,y], axis=1)
kmeans = KMeans(n_clusters=1, init='k-means++', max_iter=300, random_state=0)
pred_y = kmeans.fit_predict(X)
plt.scatter(x, y)
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s=300, c='red')
plt.title('Valor del jugador según Salario')
plt.xlabel('Valor M.Eur')
plt.ylabel('Salario')
plt.show()
```

Fuente: Elaboración propia

Observaciones

Despues de realizar una tarea de Clustering vemos que la mayoria de jugadores valen menos de 20 millones y persiben sueldos menores de 100 mil euros. Se podria decir que esto es una constante y es el salario promedio de los jugadores de futbol.

Aplicando al Clustering de 4 etapas sueldos y su valor de los jugadores

Figura 25. Codigo de implementación para graficar con Matplotlib.

```
from matplotlib.patches import Ellipse

def draw_ellipse(position, covariance, ax=None, **kwargs):
    ax = ax or plt.gca()

    # Convert covariance to principal axes
    if covariance.shape == (2, 2):
        U, s, Vt = np.linalg.svd(covariance)
        angle = np.degrees(np.arctan2(U[1, 0], U[0, 0]))
        width, height = 10 * np.sqrt(s)
    else:
        angle = 0
        width, height = 10 * np.sqrt(covariance)

    # Draw the Ellipse
    for nsig in range(1, 4):
        ax.add_patch(Ellipse(position, nsig * width, nsig * height,
                             angle, **kwargs))

def plot_gmm(gmm, X, label=True, ax=None):
    ax = ax or plt.gca()
    labels = gmm.fit(X).predict(X)
    if label:
        ax.scatter(X[:, 0], X[:, 1], c=labels, s=40, cmap='viridis', zorder=2)
    else:
        ax.scatter(X[:, 0], X[:, 1], s=40, zorder=2)
    ax.axis('equal')

    w_factor = 0.2 / gmm.weights_.max()
    for pos, covar, w in zip(gmm.means_, gmm.covariances_, gmm.weights_):
        draw_ellipse(pos, covar, alpha=w * w_factor)
```

Fuente: Elaboración propia

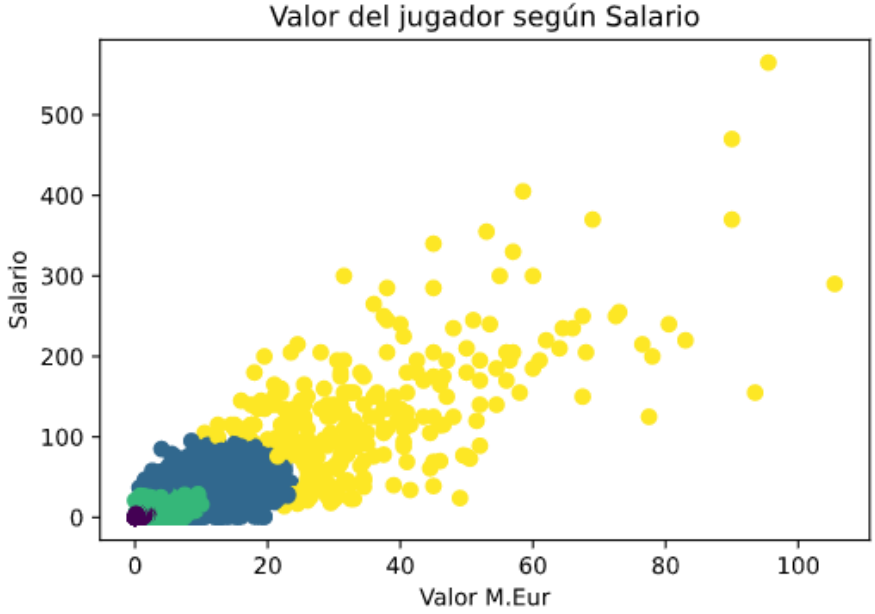
Figura 26. Imprimir los graficos.

```
X = pd.concat([x,y], axis=1)
X = X.to_numpy()
gmm = GaussianMixture(n_components=4).fit(X)
labels = gmm.predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels, s=40, cmap='viridis');
plt.title('Valor del jugador según Salario')
plt.xlabel('Valor M.Eur')
plt.ylabel('Salario')
plt.show()
```

Fuente: Elaboración propia

Visualziación grafica de las 4 etapas

Figura 27. grafica de las 4 etapas.



Fuente: Elaboración propia

Observaciones

Se puede observar que exisisten 4 segmentos marcados, donde existen jugadores que ganan lo que valen, pero en la cuarta etapa existen jugadores que valen más pero no se les paga lo que corresponde, con excepción de algunos que valen menos en el mercado pero ganan mas que otros.

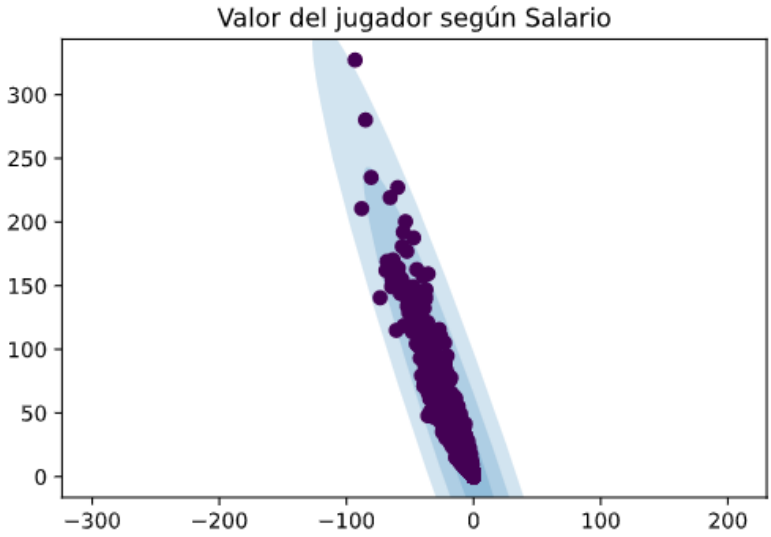
Visualización de jugadores fuera del rango de con los mejores sueldos y el valor del mercado

Figura 28. Utilizando las librerias de sckityleran para imprimir la grafica.

```
from sklearn.mixture import GaussianMixture
X = pd.concat([x,y], axis=1)
X = X.to_numpy()
rng = np.random.RandomState(13)
X_stretched = np.dot(X, rng.randn(2, 2))
gmm = GaussianMixture(n_components=1, covariance_type='full', random_state=42)
plot_gmm(gmm, X_stretched)
plt.title('Valor del jugador según Salario')
plt.show()
```

Fuente: Elaboración propia

Figura 29. jugadores fuera del rango de con los mejores sueldos y el valor del mercado.



Fuente: Elaboración propia

Observaciones

Al igual que el gráfico de 4 etapas podemos observar que exisisten 4 secmentos marcados, en este grafico vemos que solo existe un jugadopr que según la constante vale menos de lo que le pagan, este jugador es Lionel Messi que según los ultimos tiempos vajo su valor en el mercado.

Como se ha podido observar a lo largo del documento se exite un pequeño circulo rojo interlineado en donde existe un jugador que ha sido considerado el mejor de la historia hasta esta fechas y es:

Figura 30. Lionel Messi el jugador mas sobrevalorado en los utlimos tiempos.

Fuente: (As, 2020)

El jugador está evaluado por los 100,00 euros y es el mejor ya que es conocido por su larga travesía de partidos jugados y ganados en su largo recorrido. Por el momento sigue manteniendo la mejor racha jugando en el Barcelona.

Conclusión

Después de realizar este análisis se detectaron varias anomalías. En primer lugar, existen jugadores que no deberían ser centrados para el club puesto que según estadísticas y características de los jugadores estos no aportaran en mucho al equipo.

En segundo lugar, existen jugadores que actualmente valen menos de lo que solían valer, esto puede deberse al rendimiento que ahora tienen en sus respectivos equipos, en vista de ello, estos jugadores reciben salarios exorbitantes, los cuales consiguen la constante de salario vs valor del jugador, de igual manera existen jugadores que según sus estadísticas valen más de lo que perciben salarialmente. En conclusión, si bien existen jugadores que se les paga lo que valen, existen jugadores que no perciben el sueldo que merecen, esto se pude dar por varios motivos que pueden ser ajenos a las

estadísticas del jugador. Estos datos fueron extraídos de las características que tienen los jugadores en un videojuego, si bien estos datos son realizados por la FIFA no son datos exactos porque cuando se habla de futbol todo puede pasar.

Proyecto

Repositorio para ver el proyecto completo.
https://github.com/AlbertAC/ProyectoFinal/blob/master/SystemFifa.ipynb

Referencias Bibliográficas

- AlbertAC. (22 de 10 de 2020). *GitHub*. Obtenido de AprendizajeAutomatico: https://github.com/AlbertAC/ProyectoFinal/blob/master/SystemFifa.ipynb
- As. (2020). *As.* Obtenido de https://resultados.as.com/resultados/ficha/deportista/messi/15167/
- BeSoccer. (2020). *BeSoccer.* Obtenido de https://www.besoccer.com/player/a-lonergan-12304
- Kafagy, A. (26 de 11 de 2019). *GitHub.* Obtenido de fifa-FUT-Data: https://github.com/kafagy/fifa-FUT-Data
- Leone, S. (26 de 10 de 2020). *Kaggle*. Obtenido de FIFA 21 complete player dataset: https://www.kaggle.com/stefanoleone992/fifa-21-complete-player-dataset?select=players_21.csv
- Mishra, A. (15 de 09 de 2020). *Kaggle.* Obtenido de Conjunto de datos de jugador completo de FIFA 2021: https://www.kaggle.com/aayushmishra1512/fifa-2021-complete-player-data

Artículo Recibido: 23-09-2020
Artículo Aceptado: 11-10-2020