

**EA**

**ARGENTINA**



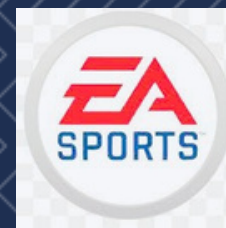
**FIFA 19**



# ***¿PODREMOS PREDECIR QUE POSICIÓN OCUPA UN JUGADOR EN BASE A SUS ESTADÍSTICAS DE EL VIDEOJUEGO FIFA 19***

EQUIPO DE ANALISIS Y  
DESARROLLO EA ARGENTINA

***VAMOS A VERLO !!***



# ***PRESENTACIÓN DE LOS DATOS***

El archivo FIFA 19.csv contiene los datos de los jugadores de el videojuego con todas sus características.

Fue extraído de la página Kaggle.com

# ***EQUIPO DE TRABAJO***

- Ignacio Kalbermatten
- Javier Gómez
- Josue Pozo
- Yamil Merlo

# *VEAMOS LOS PASOS REALIZADOS*



# 01 *DATA WRANGLING*

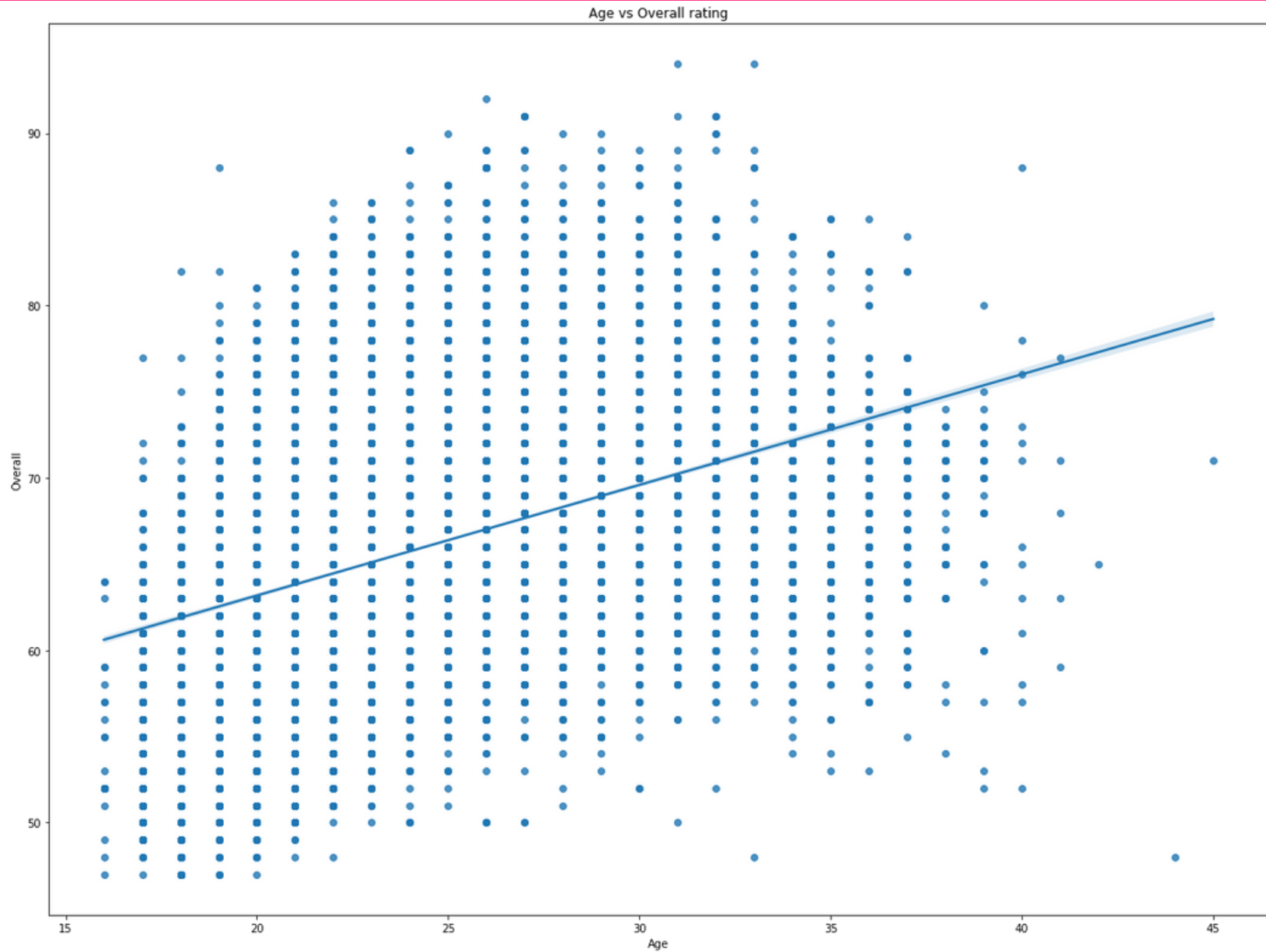
- Verificamos las características de nuestras variables con un `df.info()`
- Revisamos si existían registros Nulos
- Analizamos que la consistencia de nuestro dataset esté lo mas prolija posible

# 02

## ***ANÁLISIS EXPLORATORIO DE DATOS***

- Realizamos un HeadMap de las variables para ver la relación entre cada una de las variables
- Con ello nos percatamos algunos casos evidentes de análisis univariado y bivariado
- ejecutamos una correlación Lineal para dejar en evidencia nuestro análisis

# Análisis de Edad vs Overall Rating





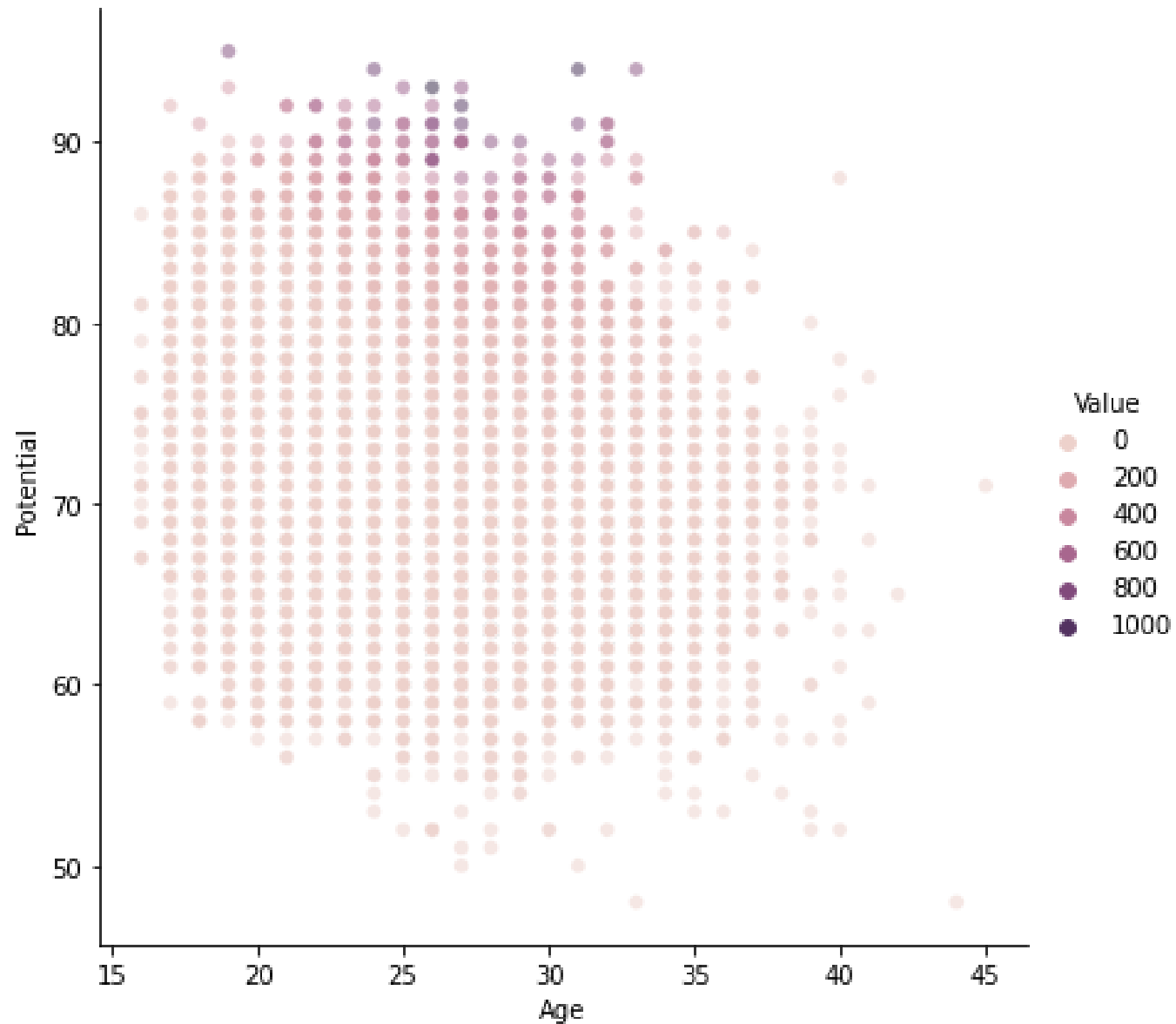
# 03 *SEGMENTACIÓN DE DATOS*

En esta oportunidad segmentamos o separamos las variables que estaban demás para así dejar solamente las variables que vamos a trabajar para ello creamos un dataframe nuevo seleccionando así las variables mas relevantes

# 04 *ANALISIS MULTIVARIADO*

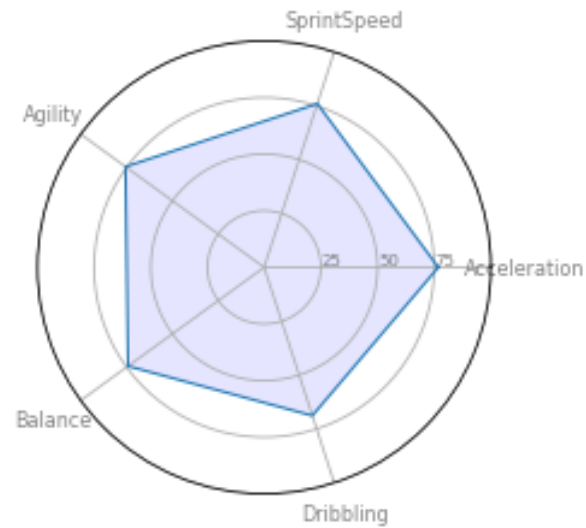
Luego de identificar las variables que target que tienen correlación decidimos graficar la distribución por area de cada característica dependiendo de la posición del jugador

# Análisis de Edad vs Potencial

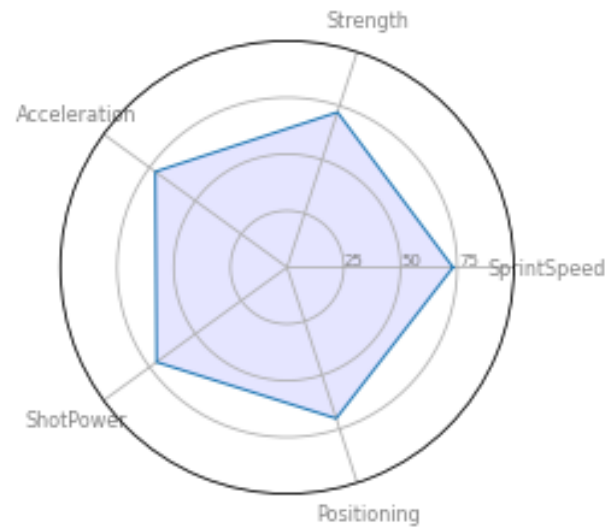


# Distribución por Area de cada característica por posición de jugador

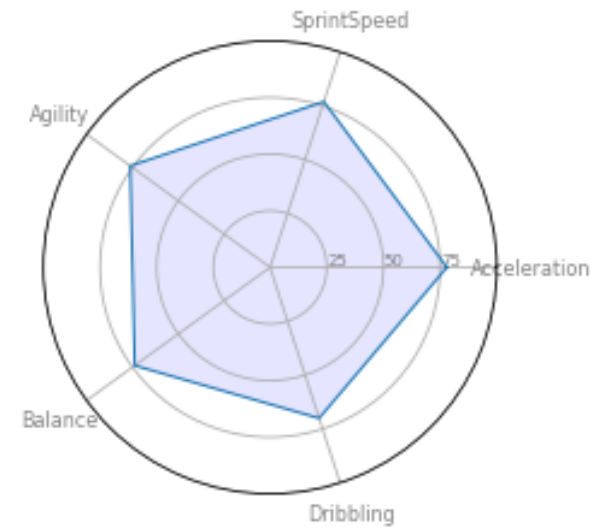
**LM**



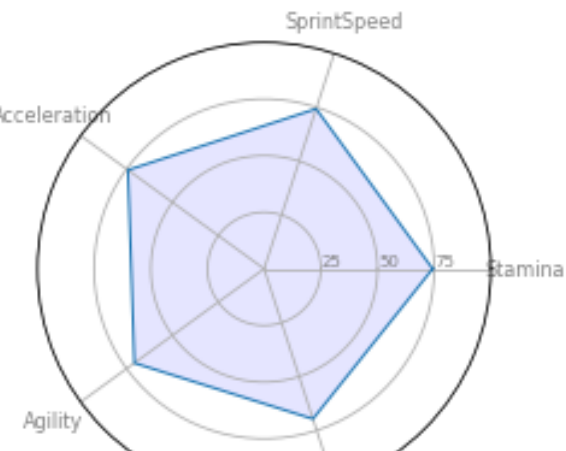
**LS**



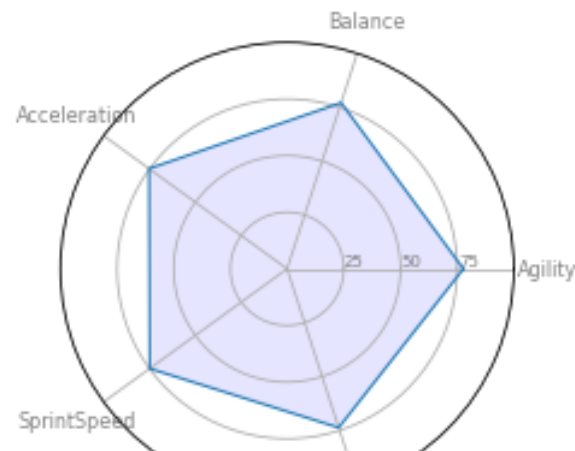
**LW**



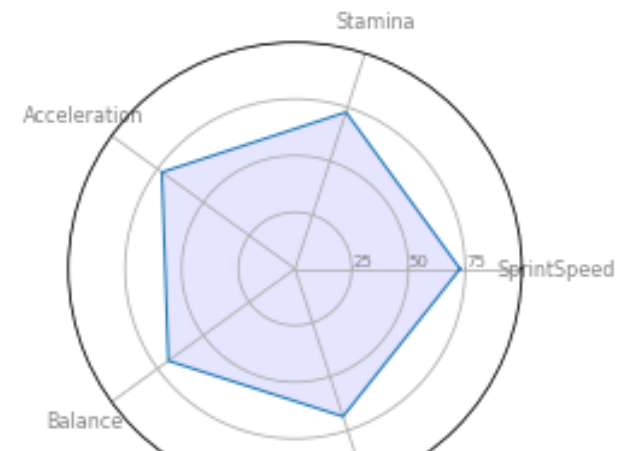
**LWB**



**RAM**



**RB**



# 05

## ***PREDICCIÓN DE POSICIONES DE LOS JUGADORES***

Ya seleccionadas las variables target y ordenadas vamos por nuestro objetivo que es Predecir la posición de un jugador dado una serie de atributos distribuidos entre :

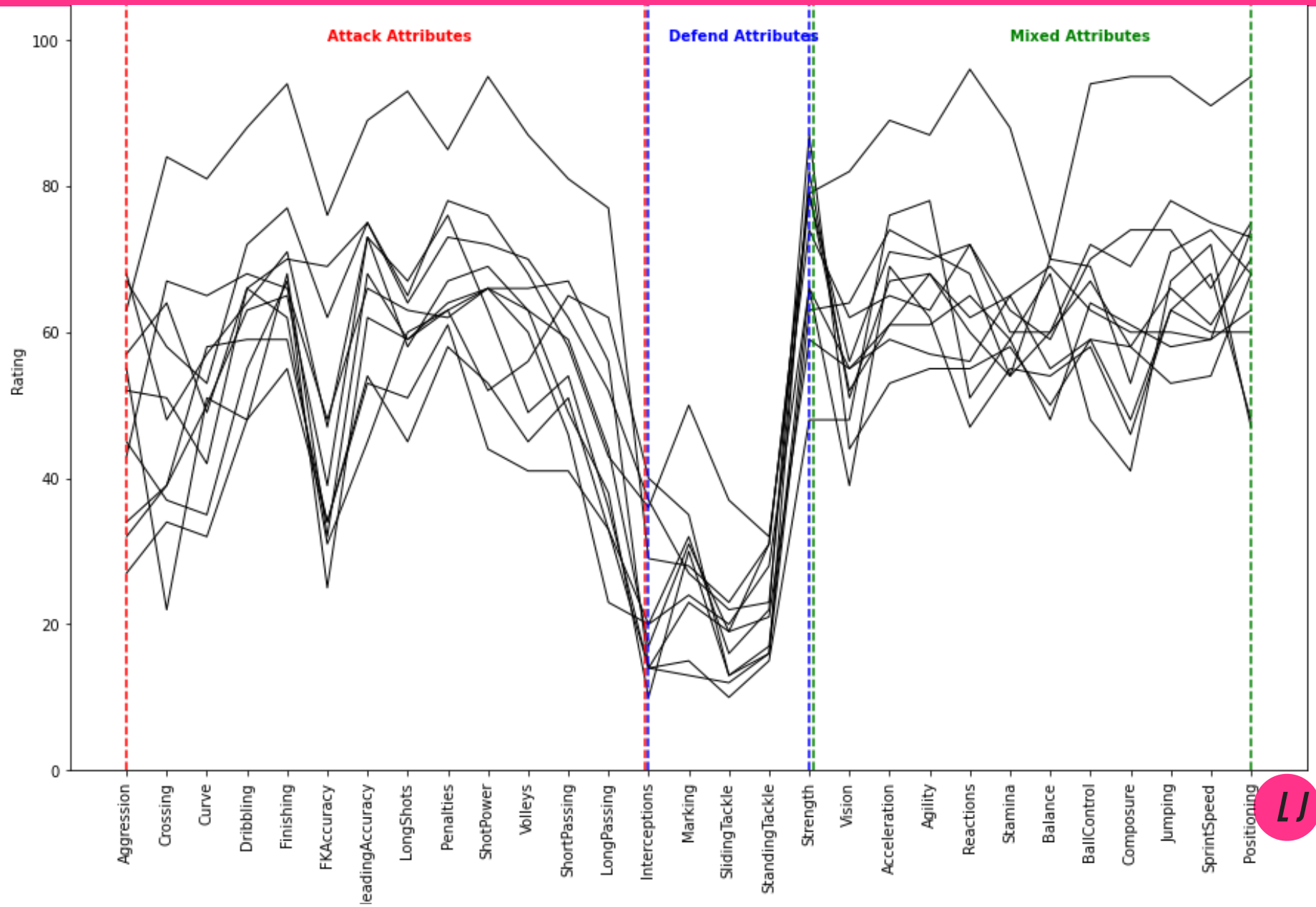
- Ofensivos
- Defensivos
- Mixtos



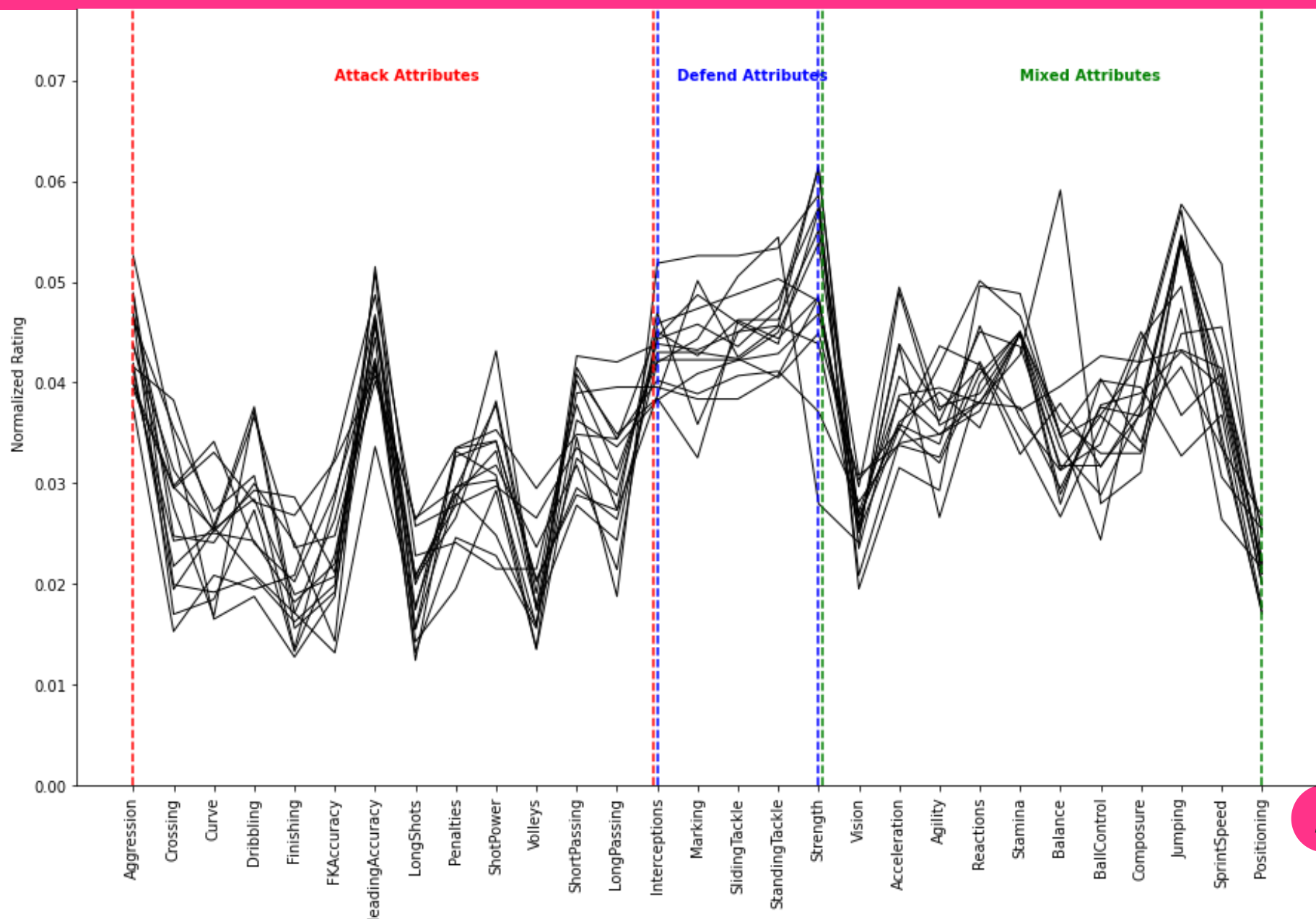
# 06 *NORMALIZACIÓN DE DATOS*

Verificaremos el patrón de atributos tomando como muestra 200 columnas de  
nuestro Dataframe

# Distribución de Atributos (sin normalizar)



# Distribución de Atributos (Ya normalizados)



# 07

## ***PREDICCIÓN POR REGRESIÓN LOGÍSTICA***

Los datos, después de la normalización, se verifican mas acorde a las posiciones de los jugadores como se demostro.

Vamos a normalizar todo el conjunto de datos

Reclasificaremos el valor target (posiciones) en grupos binarios como se muestra a continuación:

1 = posiciones de ataque = RF, ST, RW, LW, RM, CM, LM, CAM, CF, RCM, LF, RS

0 = posiciones defendidas = CDM, CB, LB, RB, RWB, LWB, RCB, LCM, LS, LCB, LAM, RDM, RAM

# 08 *IMPORTAMOS LIBRERÍAS*

```
from sklearn.model_selection import train_test_split  
from sklearn.linear_model import LogisticRegression  
from sklearn.dummy import DummyClassifier
```



# 09 *ENTRENAMOS EL MODELO*

```
#Entrenamos el modelo
```

```
X_train, X_test, y_train, y_test = train_test_split(df_new_normalized.iloc[:, :-1], df_new_normalized.iloc[:, -1], random_state=0)

print('X train shape: {}'.format(X_train.shape))
print('X test shape: {}'.format(X_test.shape))
print('y train shape: {}'.format(y_train.shape))
print('y test shape: {}'.format(y_test.shape))
```

# 10 *APLICANDO REGRESIÓN LOGISTICA AL MODELO*

```
clf_d = DummyClassifier(strategy = 'most_frequent').fit(X_train, y_train)
acc_d = clf_d.score(X_test, y_test)
print ('Dummy Classifier (most frequent class): {}'.format(acc_d))

clf = LogisticRegression().fit(X_train, y_train)
acc = clf.score(X_test, y_test)
print ('Logistic Regression Accuracy: {}'.format(acc))
```

```
Dummy Classifier (most frequent class): 0.4963386727688787
Logistic Regression Accuracy: 0.8725400457665904
```

***¿PODREMOS MEJORAR  
AÚN MÁS LA PRESIÓN  
DEL ALGORITMO?***

***¡¡¡VEAMOSLO!!!***

# 11

## ***LIMITANDO LOS ATRIBUTOS***

Limitaremos los atributos de 28 a 15 ya que consideramos que son los principales así mejoraremos la exactitud de nuestro algoritmo

# RESULTADO FINAL

La precisión ha mejorado poco. esto se debe a que la distribución de ciertos atributos es bastante uniforme entre atacante, defensa, mixtos.

-Los atributos más significativos son en su mayoría defensivos, lo que significa que son más útiles para ayudarnos a identificar el rol de un jugador. En otras palabras, es extremadamente raro que un jugador en un rol ofensivo como ST sea bueno en esas habilidades defensivas, mientras que es relativamente más fácil para un jugador en un rol defensivo ser bueno en algunas habilidades de ataque como el centro y la precisión de los tiros libres. , etc.