6- Evaluación.

Trabajamos con la base de datos del jugo de fútbol FIFA 2020 que contiene los datos individuales de cada jugador. Para estudiar la estructura y detectar posibles separaciones de los datos nos restringimos a las columnas que puntúan las habilidades de los jugadores. Algunas columnas que podrían haber sumando información interesante que no consideramos son por ejemplo el ingreso de los jugadores y el número en sus camisetas, que suele estar relacionado a su posición.

La división en dos clústers muestra una separación muy satisfactoria en términos del coeficiente silhouette, ambos grupos están distanciados en el espacio de características que evaluamos. La desventaja de esta clusterización es que es muy desbalanceada. Tenemos un grupo un clúster con coeficiente silhouette promedio mayor a 0.8, que es varias veces menor en términos de cantidad de elementos que el otro.

El desbalance de los clústers es un indicio de que el clúster mayor puede tener una estructura de clústers internos. Efectivamente, al considerar más clústers, siempre aparece uno de similares características al menor de los 2 iniciales, con un perfil silhouette idéntico independientemente de la cantidad de clusters usados, entre 2 y 8.

A medida que incrementamos el número de clústers, los perfiles silhouette se vuelven más desbalanceados internamente. Los clústers en los que se divide el mayor, están cerca por lo que al subdividirlo más y más aparecen en cada clúster más elementos con coeficientes silhouette bajo, e incluso negativo.

Utilizando el criterio del codo seleccionamos 4 clústers para subdividir la muestra ya que en el gráfico de inercia versus clústers tiene el mayor cambio de pendiente en este caso.

Al sacar a los arqueros, con k-means y 3 clusters encontramos clústers muy similares a los anteriores sin tener en cuenta el clúster que contenía a todos los arqueros. En cambio, con MeanShift por defecto se encuentran 2 clústers, uno simular a uno de los encontrados con k-means y en otro a los dos restantes.

Nuevamente, MeanShift que es más costoso, no ofrece mejores resultados que k-means.

Embeddings y técnicas de clustering

En la última actividad, trabajamos con PCA y tSNE, técnicas para la reducción de la dimensionalidad. A continuación, aplicamos los algoritmos de cluster Kmeans y Mean Shift con las componentes resultantes del análisis de PCA.

A partir del gráfico del codo determinamos que un buen número de clústers podría ser 3. Además este número coincide con las posibles posiciones de los jugadores dentro del campo de juego (sin tener en cuenta a los arqueros).

Debido a que K-Means se ve obligado a encontrar la cantidad de clústers que se le seteó previamente, en el gráfico podemos ver los 3 que encontró. Si bien tiene sentido el agrupamiento, de acuerdo a nuestro entendimiento de la visualización, el clúster que se encuentra en el medio (amarillo), no es tan evidente, como si quizás lo son los otros 2.

En la aplicación de Mean Shift, experimentamos con diferentes valores para el parámetro bandwidth, encontramos uno que permitió que el modelo encuentre 2 clústers. El valor utilizado de dicho parámetro es 3.

Visualizando en el gráfico los clústers encontrados por el modelo, creemos que la separación propuesta es adecuada de acuerdo a la dispersión de los datos en este espacio dimensional (generado por PCA).

Si bien podemos ver que la separación es similar a la propuesta por K-Means, creemos que el número de clústers encontrados por MeanShift tiene más sentido (al menos en este espacio dimensional).

7- Pregunta: ¿Se realizó alguna normalización de la base? ¿Por qué ?

Estos algoritmos usan la distancia en el espacio de todas las variables seleccionadas. Es necesario normalizarlas para que las variables que tienen rangos distintos, no tengan distinta relevancia a la hora de calcular la distancia euclídea (multidimensional). Por ello es óptima la normalización que garantiza que todas las variables tengan dominio [0,1]. Una alternativa es la estandarización que lleva las variables al dominio con media 0 y varianza 1, para cada variable. En general, esto puede no ser óptimo, pero en nuestro casa particular en que todas las variables por definición tienen dominio [0,100], esto puede no representar un problema y garantizar que todas las variables tengan varianza 1, puede ser una ventaja para equilibrar el peso de las variables.

La descomposición PCA del sklearn centra los datos pero no los normaliza. Además para generar la proyección a las nuevas componentes principales, utiliza como peso la desviación estándar en las unidades de esa componente real. Por eso, **estandarizamos** para que todas las variables tengan media 0 y desviación 1. Utilizamos StandardScaler de scikit-learn.