Clustering

Manuel Gamboa

May 2024

Resumen

Este trabajo trata sobre la técnica de clustering o agrupamiento y se divide en tres partes, primero se profundizará en algoritmos de agrupamiento de diferentes variantes para agrupar conjuntos de datos, una segunda parte profundizará sobre los índices de evaluación de agrupamiento para ver qué tan buenos son los resultados obtenidos tras haber aplicado K-Means, y una tercera parte incluirá la primera y la segunda aplicadas al objeto de estudio de este trabajo, el cual será el conjunto de municipios de cuba, teniendo en cuenta como parámetros la relación que hay entre las proporciones de edad en cuanto a edad pediátrica, adulto joven y adulto mayor.

Parte 1 Algoritmos de Agrupamiento:

Existen diferentes tipo de algoritmos de agrupamiento, algunos son más beneficiosos que otros debido a las características de los datos, en general los algoritmos de agrupamiento se pueden agrupar en 4 categorías distintas.

Basados en Densidad:

En los algoritmos basados en densidad los datos se agrupan por áreas de altas concentraciones de puntos de datos rodeadas por áreas de bajas concentraciones de puntos de datos, en general el algoritmo encuentra las áreas que tienen grandes concentraciones de puntos de datos y las nombra en un grupo, cabe resaltar que las áreas pueden tener cualquier forma mientras esta sea densa. Este tipo de algoritmos, al no tener en cuanta las áreas con poca densidad, ignora los casos aislados.

Un algoritmo ejemplo de este tipo es el conocido como DBSCAN(Density-based spatial clustering of applications with noise) o en español Agrupamiento Espacial Basado en Densidad de Aplicaciones con Ruido, a este algoritmo se le pasa como parámetros: un epsilon mayor que cero, o sea la distancia máxima que habrá entre un elemento y otro para que se consideren vecinos, y como segundo parámetro la cantidad M de puntos de datos mínimo que contendrá un conjunto para considerarse grupo. A lo largo de la ejecución del algoritmo se encontrarán distintos tipos de puntos de datos, el llamado punto central, es un punto que contenga como cantidad de vecinos M o mayor que M(un punto q es vecino de otro p si está a una distancia menor o igual que), el llamado punto borde, este es el punto que tiene menos vecinos que M pero son vecinos de un punto central, y el llamado punto de ruido, este es el punto que no entra en ninguna de las categorías anteriores.

Se dice que un punto p es directamente alcanzable por densidad desde q, si q es un punto central y p es vecino de q, se dice que un punto de alcanzable por densidad si existe una secuencia ordenada de puntos directamente alcanzables por densidad que los une, se dice que dos puntos p y q están densamente conectados si existe un punto p' tal que p y q se alcanzan por densidad a partir de o, luego sea un grupo G, entonces todos sus puntos están densamente conectados, y si C es un conjunto de puntos densamente conectados, entonces C es considerado un grupo. Ahora el algoritmo actúa de la siguiente manera: puede estar en dos estados distintos, un estado es cuando ya ha encontrado un punto central, entonces procede a recorrer todos sus vecinos hasta no encontrar más puntos centrales o puntos bordes, y el segundo estado es cuando aún no ha encontrado puntos centrales, cuando está en el primer estado, a medida que va recorriendo los puntos, los marca como visitados y como puntos de ruido, y si está en el segundo estado los marca, dependiendo de la cantidad de vecinos, como puntos borde o puntos centrales.

Para comprobar si un punto q está a una distancia menor o igual que del punto p, no es necesario verificar todos los puntos, pero para acotar dicha cantidad se agrupará todos los puntos en un Árbol Binario de Búsqueda de orden k dependiendo de . Este árbol se construirá de la siguiente forma: si estamos hablando de R2, entonces si p pertenece al nodo (i,j), solo basta verificar los nodos que están alrededor del mismo en R2, lo que significa verificar cada punto que está en (i + 1, j), que es el que está arriba, el nodo (i -1, j), que es el que está abajo, y así sucesivamente hasta cubrir las ocho direcciones.

Ventajas:

• No es necesario especificar un número inicial de clusters o grupos(luego se verá que en otros algoritmos como K-Means es necesario especificar dicha cantidad)

• Los grupos que forma no tendrán una forma en concreto, o sea no es una restricción para este algoritmo que los grupos sigan alguna forma para definirse

Véase en la figura 1 se han generado puntos aleatorios entre 0 y 1, y al correr el algoritmo con parámetros de = 0.1 y M = 20. Donde los puntos en rojo constituyen los valores atípicos, y el resto de los colores denotan los grupos formados. Aquí puede denotarse dos grupos importantes de puntos, véase que en el conjunto de los puntos rosa, hay un valor atípico muy cerca de un elemento del conjunto, esto se debe a que ese valor no tiene como vecino a un punto central, sin embargo tiene como vecino a dicho punto que es un punto borde.

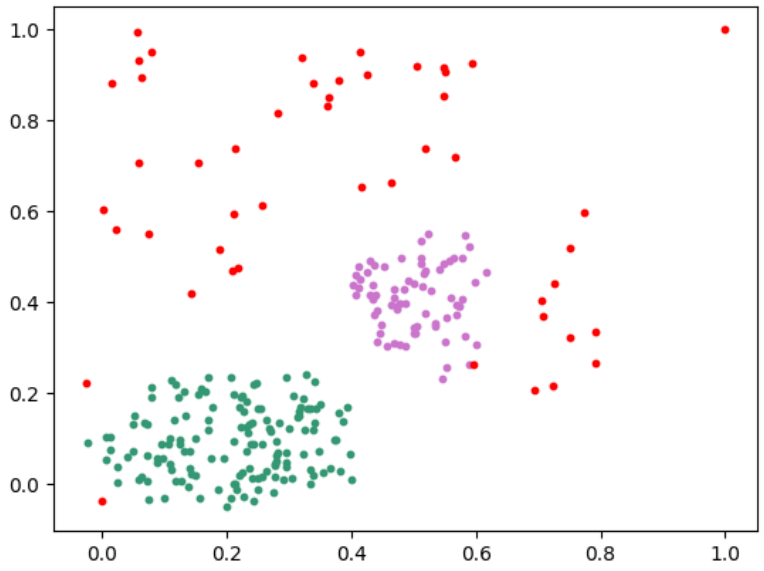


Figura 1

Ahora véase la figura 2 como el algoritmo agrupa correctamente los puntos siguiendo un patrón de densidad

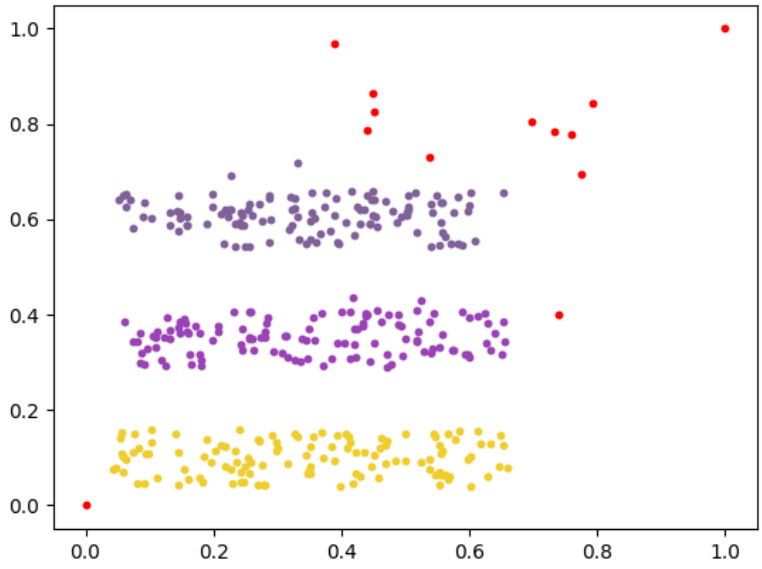


Figura 2

Desventajas:

• No tiene en cuenta los puntos atípicos o outliers, al no incorporarlos a los grupos más cercanos estos valores no aportan información al conjuntos

• En conjuntos de datos donde los elementos están muy dispersos no calcula muy bien los grupos. Ejemplo en la figura 3

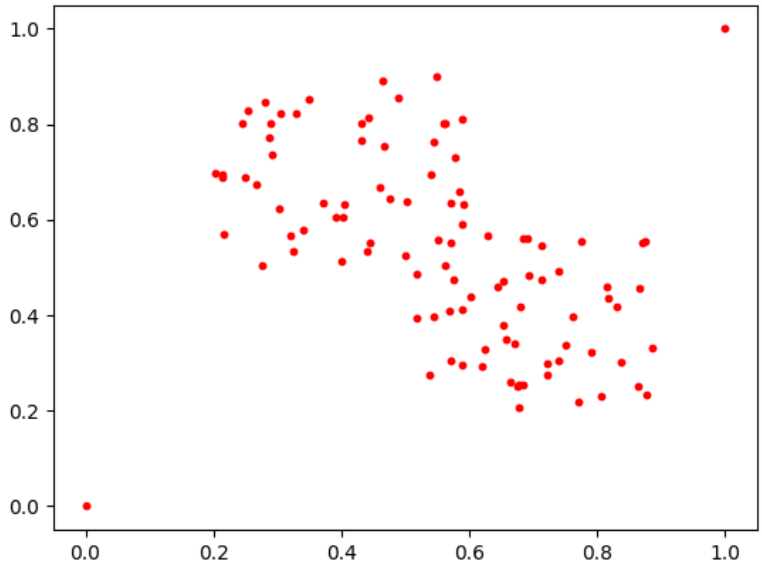


Figura 3

Véase que aunque visualmente hay claramente dos grupos en los que se puede dividir ese conjunto de datos, usando el algoritmo no los distingue, y si aumentamos el $= 0.1$ a $= 0.2$ entonces forma un solo grupo, que sería el de todos esos elementos, cuando debería formar 2, véase figura 4

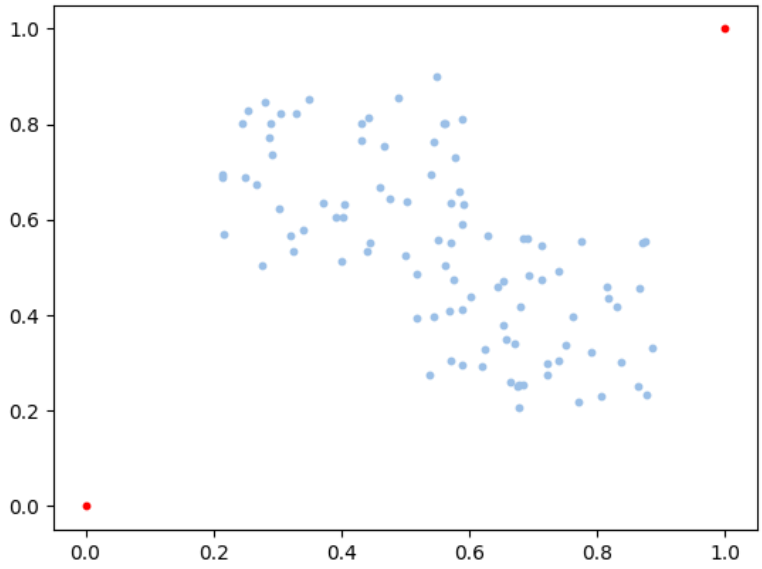


Figura 4

Resumiendo el algoritmo de DEBSCA agrupa bien cuando hay densidad de puntos independientemente de la forma del área que puedan ocupar, y agrupa mal los conjuntos de datos que, a pesar de tener un patrón de distribución circular no son muy densos y están cerca unos de otros los conjuntos posibles a formar. Cabe resaltar que dependiendo del conjunto de datos que se esté estudiando se puede emplear diferentes variantes de este algoritmo que mejoren el agrupamiento, como por ejemplo incluir los valores atípicos en el grupo correspondiente al borde al que es vecino, en caso de que lo sea, y si es vecino de bordes de diferentes grupos, entonces asignarlo al grupo del borde con el que tenga menor distancia, para saber si es un mejor agrupamiento se evalúa usando métricas o índices de evaluación que se verán más adelante en la segunda parte.

Basados en centroides:

Uno de los algoritmos más conocidos de agrupamiento basados en centroides es el algoritmo de K-Means. Este algoritmo es sencillo, se le pasa como parámetro la cantidad de clusters o grupos en los que se quiere dividir el conjunto de datos, y te devuelve los grupos que se formaron. Cada grupo está definido por su centroide, que es el valor promedio de todos los valores que pertenecen a dicho grupo, a cada elemento se le asigna como grupo, el grupo que esté caracterizado por el centroide más cercano a dicho elemento, y se recalculan los centroides de los grupos mientras haya algún cambio, o sea si algún elemento cambia de

grupo, o mientras que los centroides se desplacen por encima de un cierto umbral. Al iniciar la ejecución del algoritmo se toman la cantidad k de elementos igual a la cantidad de grupos a formar y se convierten en los centroides de los clusters. Como hay que proporcionarle al algoritmo una cantidad k de grupos a formar, una buena estrategia a seguir es ir probando con 2,3,. . . ,k grupos hasta obtener los mejores resultados, basándonos en los índices de evaluación que se verán más adelante, pero como aún no llegamos, a modo de ejemplo se pueden ver a simple vista tratándose de solo dos dimensiones Tomemos como primer ejemplo una distribución de varios puntos aleatorios alrededor de 6 puntos en específicos, los 6 distantes entre sí, al aplicar el algoritmo de K-Means pasandole como parámetro una cantidad de grupos igual a k = 2 obtenemos el resultado que se muestra en la figura 5.

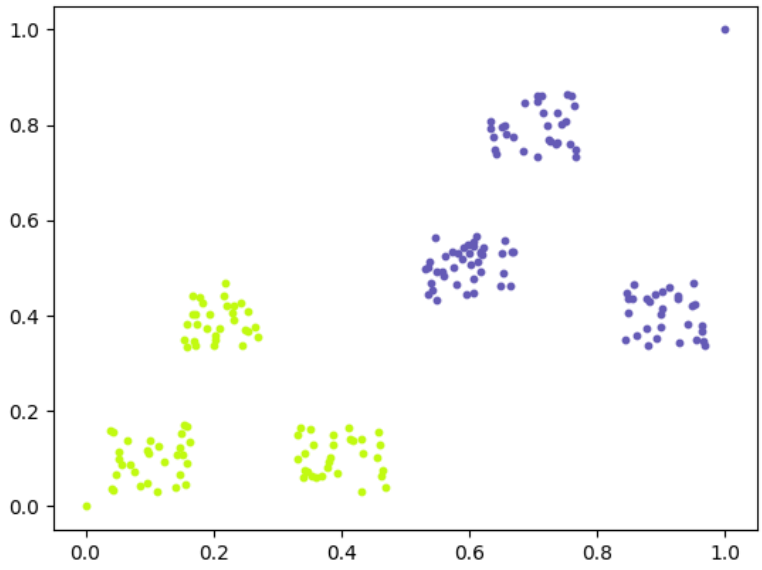


Figura 5

Véase que con dos grupos me devuelve un resultado válido, pero a simple vista se puede notar que la mejor agrupación es usando k = 6 como parámetro. Tras aplicar el algoritmo con el nuevo parámetro obtenemos el resultado plasmado en la figura 6

=9cm]KMeans2.png
Figura 6

Fue el resultado esperado, pero qué pasaría si se añadieran algunos valores atípicos, en este algoritmo estos valores si influyen, ya que siempre se ubican a

todos los elementos en algún grupo. Probemos agregando valores aleatorios por todo el espacio, véase los resultados en las figuras 7 y 8

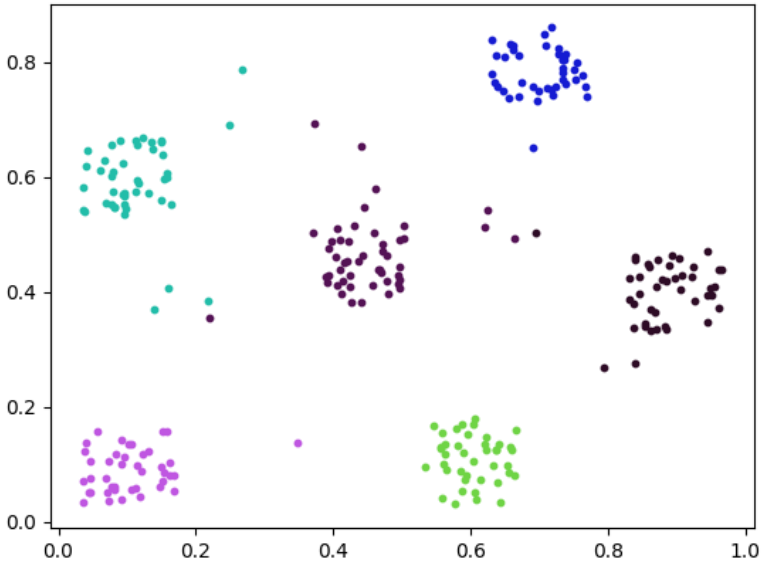


Figura 7

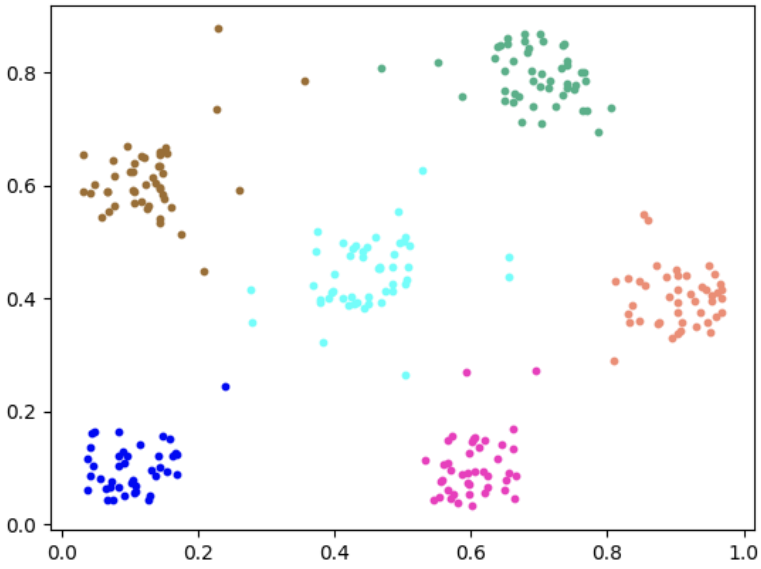


Figura 8

En ambos resultados, tanto en el primero que se añadieron 10 puntos como en el segundo en el cual se agregaron 20, podemos apreciar que, a pesar de mantenerse los grupos, hay un desplazamiento del centroide por parte de algunos grupos debido a la dispersión de los nuevos elementos añadidos, además de que de estos nuevos puntos, hay muchos que pueden pertenecer tanto a un cluster como a otro, ya que la diferencia entre la distancia hacia los centroides es mínima, por lo que incita a no tenerlos en cuenta. Este es el caso intuitivo para en el que el algoritmo funciona bien, ahora veamos cómo se comporta con uno de los casos en particular que vimos con el algoritmo de DEBSCA, en donde se le aplica K-Means pasandole como parámetros una cantidad k = 3, 4 y 6 respectivamente:(figura 9, figura 10, figura 11)

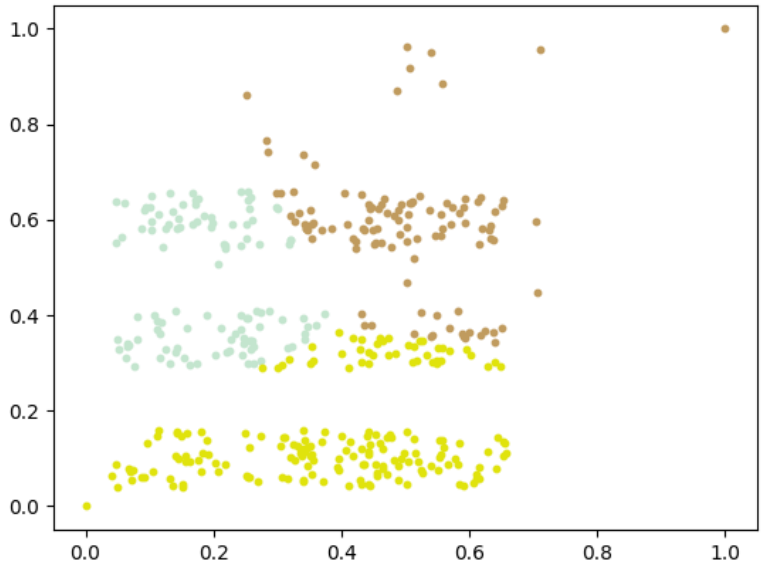


Figura 9

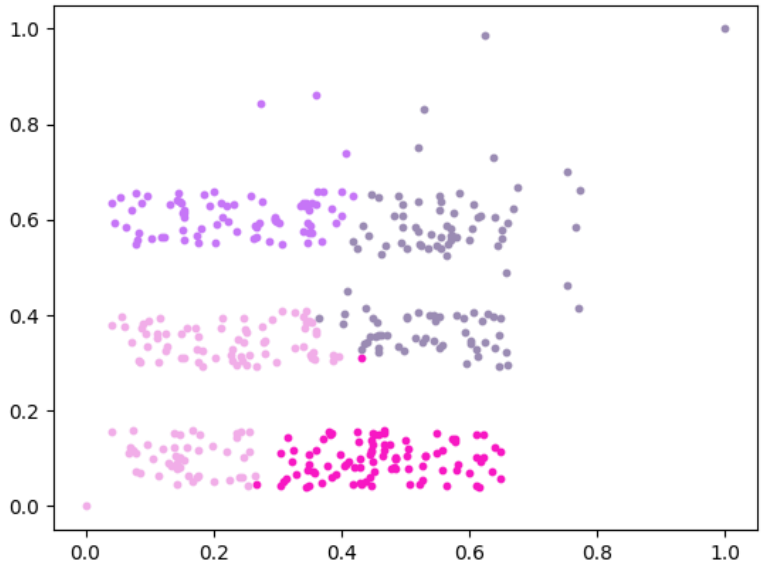


Figura 10

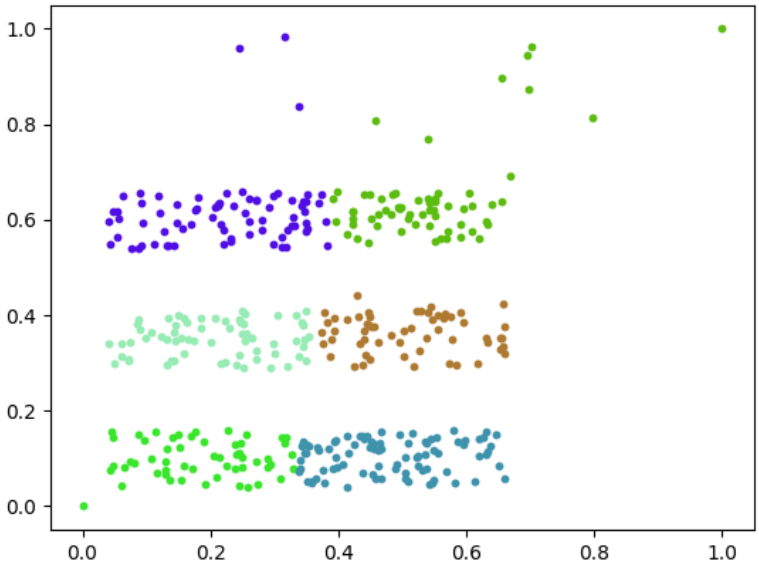


Figura 11

En estos ejemplos se puede ver con más claridad el comportamiento circular(en R2) que tiene el algoritmo a pesar de tratarse de un conjunto de datos que están distribuidos con patrones distintos.

Ventajas:

• A la hora de ubicar un nuevo elemento en un grupo ya creado es fácil encontrar cual es su más cercano, ya que solo debe medir su distancia a cada uno de los centroides(aunque no garantiza que pertenezca específicamente a un solo grupo) • Es rápido y sencillo • Tiene en cuenta los valores atípicos(no siempre es bueno)

Desventajas:

• Tiene un carácter circular, por lo que no agrupa de forma correcta distintos patrones reconocibles a simple vista(en R2) • Hay que especificar la cantidad de clusters o grupos desde un inicio(ya vimos que probando una cierta cantidad podíamos aproximarnos a una buena solución)

Parte 2 Índices de Evaluación de Agrupamientos

Los índices de evaluación de agrupamiento o clustering nos permiten distinguir una buena agrupación de una mala en resultados que no pueden distinguirse a simple vista, o sea cuando hay mas de 3 parámetros o dimensiones. Para cada uno de los índices que se verán en este documento no hay un número en específico que diga si el agrupamiento es bueno, sino que depende del conjunto de datos, por lo que de una serie de agrupamientos que se hagan, el mejor agrupamiento para ese conjunto de datos sería el que tiene mayor valor en el índice, o menor dependiendo del que se aplique.

Índice Calinski Harabasz:

Este índice propone una proporción entre la varianza global entre todos los grupos o clusters hallados y la varianza de los elementos con respecto al centroide del cluster al que pertenece. La varianza global la calcularemos como:

$$SS_B = \frac{\sum_{i=1}^{k} n_i ||m_i - m||^2}{k-1} \tag{1}$$

donde k es la cantidad de clusters, n_i es la cantidad de elementos en el cluster número i, m_i es el centroide número i y m es la media de todos los valores observados. La varianza de los elementos con respecto a sus centroides loa calcularemos como:

$$SS_W = \frac{\sum_{i=1}^{k} \sum_{x \in C_i} ||x - m_i||^2}{N-k} \tag{2}$$

donde k es la cantidad de clusters, C_i es el cluster i-ésimo, N es la cantidad total de elementos a estudiar. Luego el índice de Calinski Harabasz quedaría como:

$$CH = \frac{SS_B}{SS_W} = \frac{\sum_{i=1}^{k} n_i ||m_i - m||^2 (N-k)}{(\sum_{i=1}^{k} \sum_{x \in C_i} ||x - m_i||^2)(k-1)} \tag{3}$$

Véase que mientras mayor sea la distancia entre los clusters, y menor la distancia entre los elementos de cada cluster con respecto a su centroide, entonces el índice nos indica mejor resultado cuando se forman los grupos, en general, mientras mayor sea el índice de Calinski Harabasz, mejor será el resultado obtenido.

A continuación se generarán puntos en R^2 y luego se agruparán haciendo uso del algoritmo ***K-Means***, se harán varias iteraciones sobre el mismo conjunto de puntos generados, variando la cantidad de grupos a obtener, lo que se espera,

es que a mayor índice obtenido indique una mejor agrupación, basándonos en la observación a simple vista. Véase las figuras 12-15, donde el índice Calinski Harabasz está representado como CH.

Se Puede observar a simple vista que hay seis conjuntos de grupos, y según los índices obtenidos podemos notar que el de mayor valor es aproximadamente 286.84, que se obtuvo al pasarle como parámetro al algoritmo K-Means k = 6(figura 14). Respecto al primer caso(figura 12) puede notarse que el grupo de los puntos en azul tiene a los elementos alejados del centroide, lo cual hace que aumente el denominador, disminuyendo así el total, dando luga a un valor de aproximadamente 127.54.

Índice Davies-Bouldin

Este índice trabaja con las distancias entre los clusters y con la distancia promedio de los elementos en cada cluster, se define como:

$$DB = \frac{1}{k}\sum_{i=1}^{k} max_{i\neq j}(\frac{d_i + d_j}{d_{ij}}) \tag{4}$$

donde k es la cantidad de clusters, d_i es el promedio de distancia entre los elementos del cluster i y su respectivo centroide, d_j al igual que d_i es la distancia promedio entre los elementos del cluster j con su respectivo centroide, y d_{ij} es la distancia entre los centroides i y j.

Este índice se va quedando con el mayor valor de la relación del grupo i con respecto a los demás grupos, siendo la misma una relación entre la suma de cuán agrupados están los elementos de un mismo grupo con respecto a su centroide, con respecto a la distancia entre dichos centroides, véase que mientras más alejados estén dichos centroides, mayor será el denominador, disminuyendo el valor de $(\frac{d_i+d_j}{d_{ij}})$, y mientras menor sean los valors de d_i y d_j menor será el numerador, disminuyendo también el valor de $(\frac{d_i+d_j}{d_{ij}})$. En general, mientras menor sea el valor del índice Davies-Bouldin, mejor será el agrupamiento. Véase en las figuras 12-15 como el valor del índice(DB) va aumentando a medida que se acerca a $k = 6$, y cuando lo sobrepasa disminuye.

Índice Silhouette

El Índice de Silhouette es mide, para cada elemento del conjunto de datos, la similitud que tiene con su propio grupo en comparación con los demás grupos. Este valor varía entre -1 y 1, mientras más se acerque a 1, indica una mejor agrupación, caso contrario si se acerca a -1. Al promediar los valores de Silhouette de todos los elementos del conjunto de datos nos dará el valor general de Silhouette para el agrupamiento del conjunto de datos.

Sea $i \in$C$_I$ entonces definamos como:

$$a(i) = \frac{1}{|C_I| - 1} \sum_{j \in C_I, i \neq j} d(i,j) \tag{5}$$

o sea la distancia promedio del centroide al resto de los elementos del grupo, sin tener en cuenta $d(i,i)$ por eso se le resta 1 al denominador $|C_I|(d(i,j)$ es la distancia euclidiana entre i y $j)$.

Por cada punto i definimos como:

$$b(i) = min_{J \neq I} \frac{1}{C_J} \sum_{j \in C_J} d(i,j) \tag{6}$$

se escoge el más pequeño con la idea de quedarnoscon la distancia promedio al cluster más cercano a i que no es C_I. Ahora definamos como valor de Silhouette:

si $|C_I| > 1$:

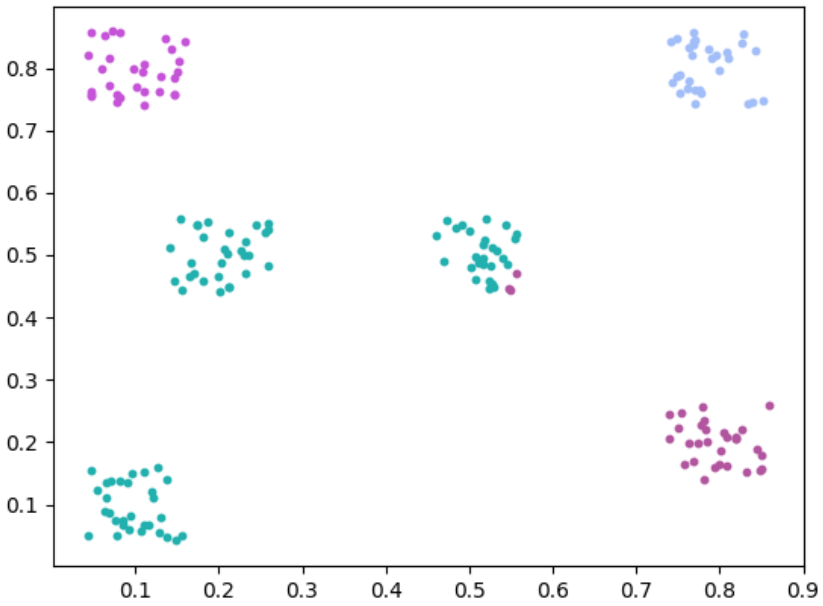
$$s(i) = \frac{b(i) - a(i)}{maxa(i), b(i)} \tag{7}$$

si $|C_I| = 1$:

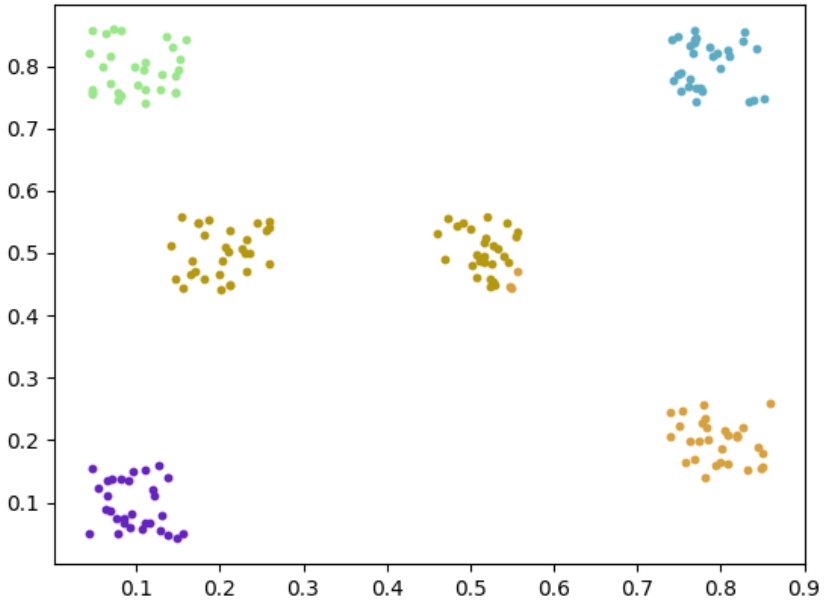
$$s(i) = 0 \tag{8}$$

Nótese que esto hace quq, como ya se dijo con anterioridad, los valores de Silhouette se encuentren en un rango de -1 y 1, donde mientras más se acerque a 1 indica un mejor emparejamiento, y mientras más se acerca a -1 indica un peor emparejamiento. Si obtenemos el promedio de los valores de Silhouette del conjunto de datos agrupados, entonces obtenemos una métrica de evalauación en general para la agrupación efectuada.

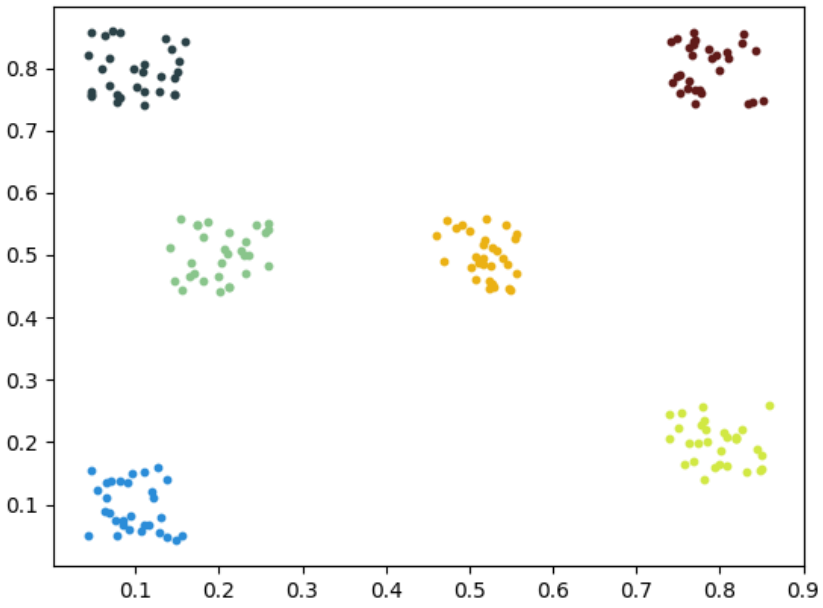
Véase en las figuras 12-15, como el valor más cercano a 1 lo obtenemos en la agrupación de k = 6, con un valor de aproximadamente 0.88.



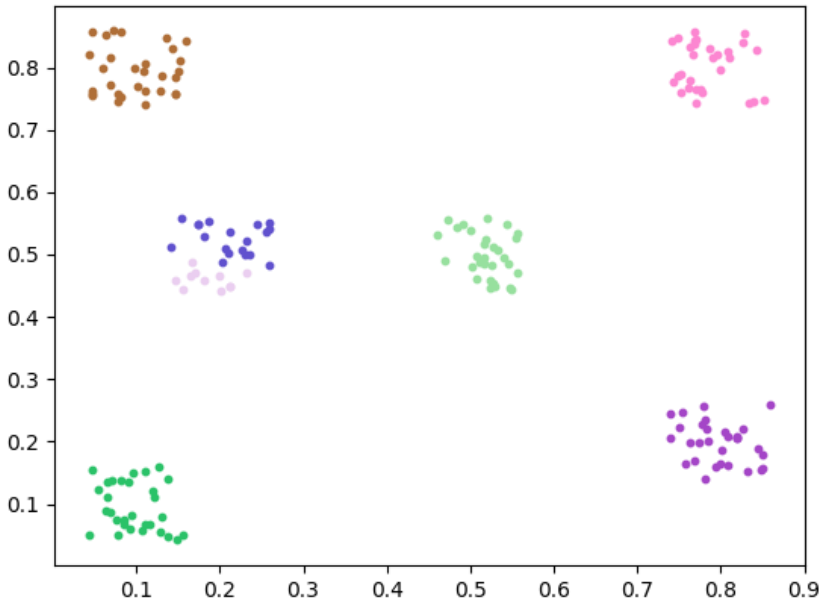
k = 4
CH = 127.54085335991431
DB = 0.5923484829040584
Silhouette = 0.6786615321033015
Figura 12



k = 5
CH = 170.1427035208753
DB = 0.4739354494398914
Silhouette = 0.8027884667961496
Figura 13



k = 6
CH = 286.8396406409764
DB = 0.2552039158314042
Silhouette = 0.8821940669696452
Figura 14



k = 7
CH = 249.6847688399773
DB = 0.43588009506966724
Silhouette = 0.832590792689287
Figura 15

. . . . . .

Parte 3 Analizando, Agrupando y Evaluando nuestro conjunto de datos(Pendiente)

. . . . .