



UNIVERSIDAD DE GRANADA

Inteligencia de Negocio

Práctica 2

Análisis relacional mediante
segmentación

David Carrasco Chicharro

Grupo de prácticas 2 (Jueves)

davidcch@correo.ugr.es

Índice

1. Introducción	1
2. Caso de estudio 1	1
2.1. K-means	2
2.2. Mean Shift	6
2.3. DBSCAN	9
2.4. Agglomerative Ward	12
2.5. Interpretación de la segmentación	13
3. Caso de estudio 2	14
3.1. K-means	14
3.2. Mean Shift	17
3.3. DBSCAN	19
3.4. Birch	21
3.5. Agglomerative Ward	23
3.6. Interpretación de la segmentación	24
4. Caso de estudio 3	25
4.1. K-means	25
4.2. Mean Shift	29
4.3. DBSCAN	31
4.4. Agglomerative Ward	34
4.5. Interpretación de la segmentación	35
5. Contenido adicional	35
6. Bibliografía	36

1. Introducción

El problema a tratar en esta práctica es la de utilizar algoritmos de aprendizaje no supervisado para realizar un análisis relacional mediante segmentación a partir de los datos proporcionados por la encuesta¹ del Instituto Nacional de Estadística (INE) en 2018 sobre fecundidad. El objetivo será definir tres casos de estudio de interés con el fin de encontrar relaciones entre distintas variables mediante algoritmos de clustering.

Para cada caso de estudio se definen las variables a utilizar, las condiciones aplicadas y los resultados obtenidos para cada algoritmo utilizado, detallando en algunos de ellos la configuración de parámetros establecida para su correcto funcionamiento y obtención de resultados óptimos cuando el algoritmo lo permita.

Los algoritmos elegidos para el análisis son: *K-means*, *Mean Shift*, *DBSCAN*, *Birch* y *Agglomerative Clustering*. Las medidas de rendimiento a utilizar serán *Calinski-Harabasz*, *Silhouette* y el tiempo de ejecución del algoritmo.

Por último, para cada caso de estudio, se realizará una interpretación global de los resultados obtenidos a fin de conseguir una conclusión clara sobre el problema.

2. Caso de estudio 1

En este caso de estudio se pretende analizar la satisfacción en el reparto de las tareas domésticas, la satisfacción de la relación, el número de hijos deseado y los estudios alcanzados en un conjunto de mujeres desempleadas donde estas tengan pareja que trabaje o no. Puede resultar interesante ver cómo, en mujeres sin empleo, según su formación académica, pueden influir la satisfacción en la relación y en el reparto de tareas en el número de hijos que desea tener. Para el estudio se ha limitado la selección a aquellas que desean un número de hijos menor que 8, pues considero que sin esta restricción se pueden tener objetos poco fidedignos con la realidad. Las variables utilizadas son:

- *EMPPAREJA*. Se han seleccionado aquellos objetos donde *EMPPAREJA=1* o *EMPPAREJA=3*, siendo 1 los casos donde ninguno tiene empleo y 3 aquellos donde la pareja tiene empleo pero la entrevistada no.
- *NDESEOHIJO*. Número de hijos deseados.
- *SATISTARDOM*. Grado de satisfacción en el reparto de las tareas domésticas.
- *SATISRELAC*. Grado de satisfacción que le proporciona la pareja.
- *ESTUDIOSA*. Nivel de estudios alcanzados.

¹https://www.ine.es/dyngs/INEbase/es/operacion.htm?c=Estadistica_C&cid=1254736177006&menu=ultiDatos&idp=1254735573002

Algoritmo	Clusters	Calinski-Harabasz	Silhouette	Tº ejec
K-means	3	2167.215	0.32094	0.20
Mean Shift	6	232.640	0.29614	1.35
DBSCAN	2	96.946	0.48320	0.35
AC	4			0.12
Birch	-	-	-	-

Tabla 1: Resultados de los algoritmos en el caso de estudio 1

En este caso de estudio el algoritmo Birch no ha conseguido ejecutarse, por lo que no ha podido realizar ninguna segmentación.

2.1. K-means

En primer lugar utilice “Elbow method”, que es un método que proporciona de manera visual cuál puede ser un buen número de clusters a elegir. Mide el valor de WCSS (“Within-Cluster-Sum-of-Squares”), el cual disminuye al aumentar el número de clusters, de manera que cuando se forma un “codo” –la disminución de un cluster a otro no es sustancial– se considera ese número clusters como un posible valor a utilizar para K-means.

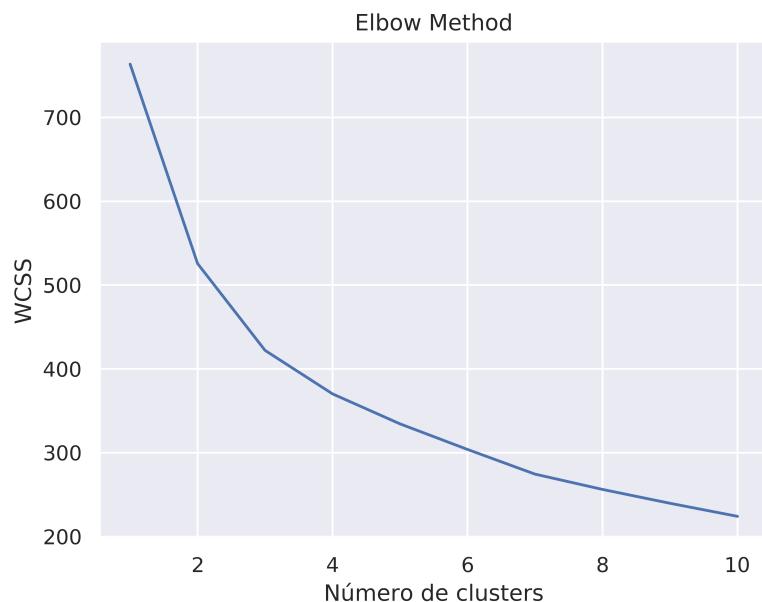


Figura 1: Elbow method

Se puede apreciar que con 2 y con 3 clusters se forman codos, por lo que ambos pueden ser adecuados. A continuación se muestran en una tabla comparativa los resultados que arroja K-means en las medidas de rendimiento *Calinski-Harabasz* y *Silhouette* según el número de clusters.

Clusters	Tamaño de cada cluster	Calinski-Harabasz	Silhouette	Tº ejec
2	0: 2683 (50.05 %) 1: 2678 (49.95 %)	2425.403	0.30570	0.10
3	0: 474 (8.84 %) 1: 2371 (44.23 %) 2: 2516 (46.93 %)	2167.215	0.32094	0.20
4	0: 920 (17.16 %) 1: 2205 (41.13 %) 2: 1776 (33.13 %) 3: 460 (8.58 %)	1916.847	0.28414	0.13
5	0: 1390 (25.93 %) 1: 1439 (26.84 %) 2: 879 (16.40 %) 3: 1212 (22.61 %) 4: 441 (8.23 %)	1724.043	0.23564	0.14
6	0: 1350 (25.18 %) 1: 1063 (19.83 %) 5: 1009 (18.82 %) 4: 807 (15.05 %) 2: 757 (14.12 %) 3: 375 (6.99 %)	1643.228	0.24169	0.20

Tabla 2: Resultados de K-means en el caso de estudio 1

Tanto con dos como con tres clusters los resultados son muy similares, tal y como se podía intuir que sucedería con la representación visual de *Elbow method*, así que me decanto por elegir tres, ya que ofrece mejor rendimiento con *Silhouette*.

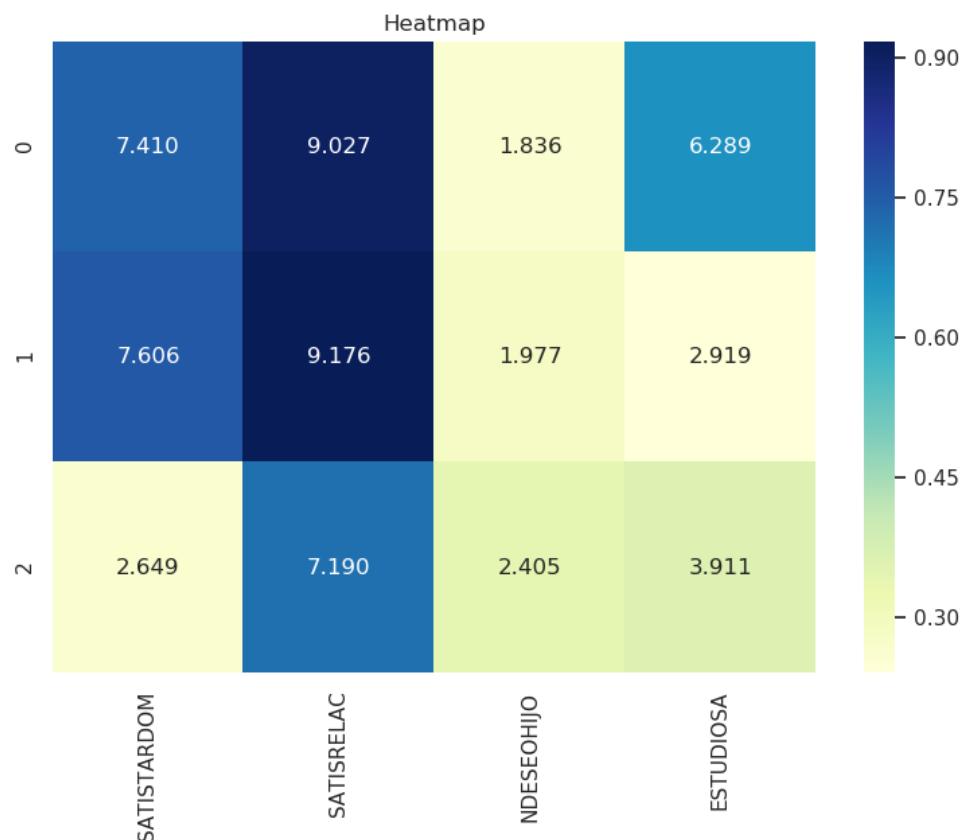


Figura 2: Heatmap para K-means

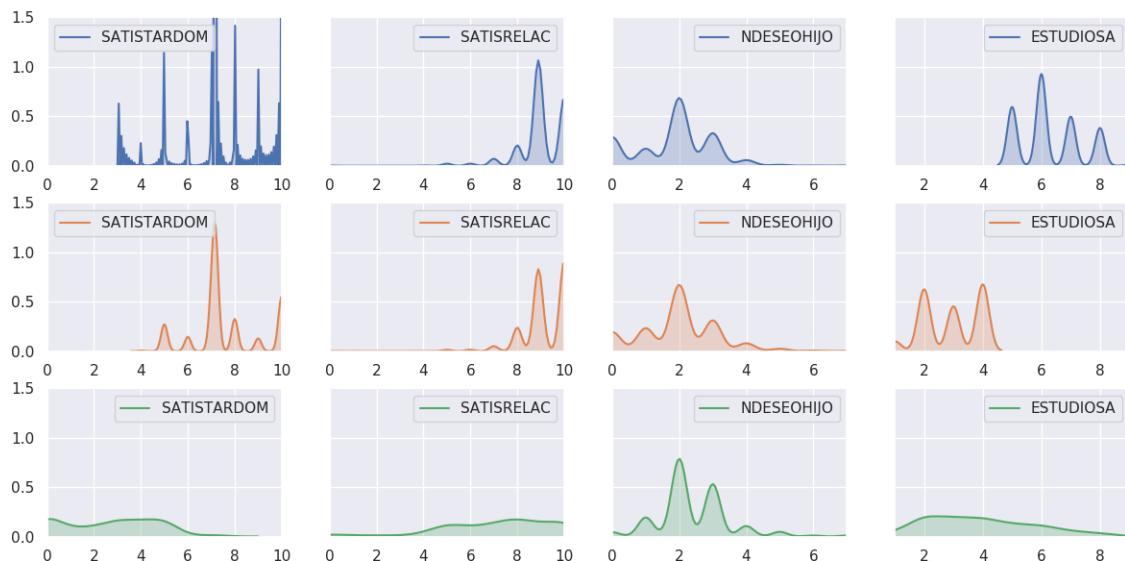


Figura 3: KDE para K-means

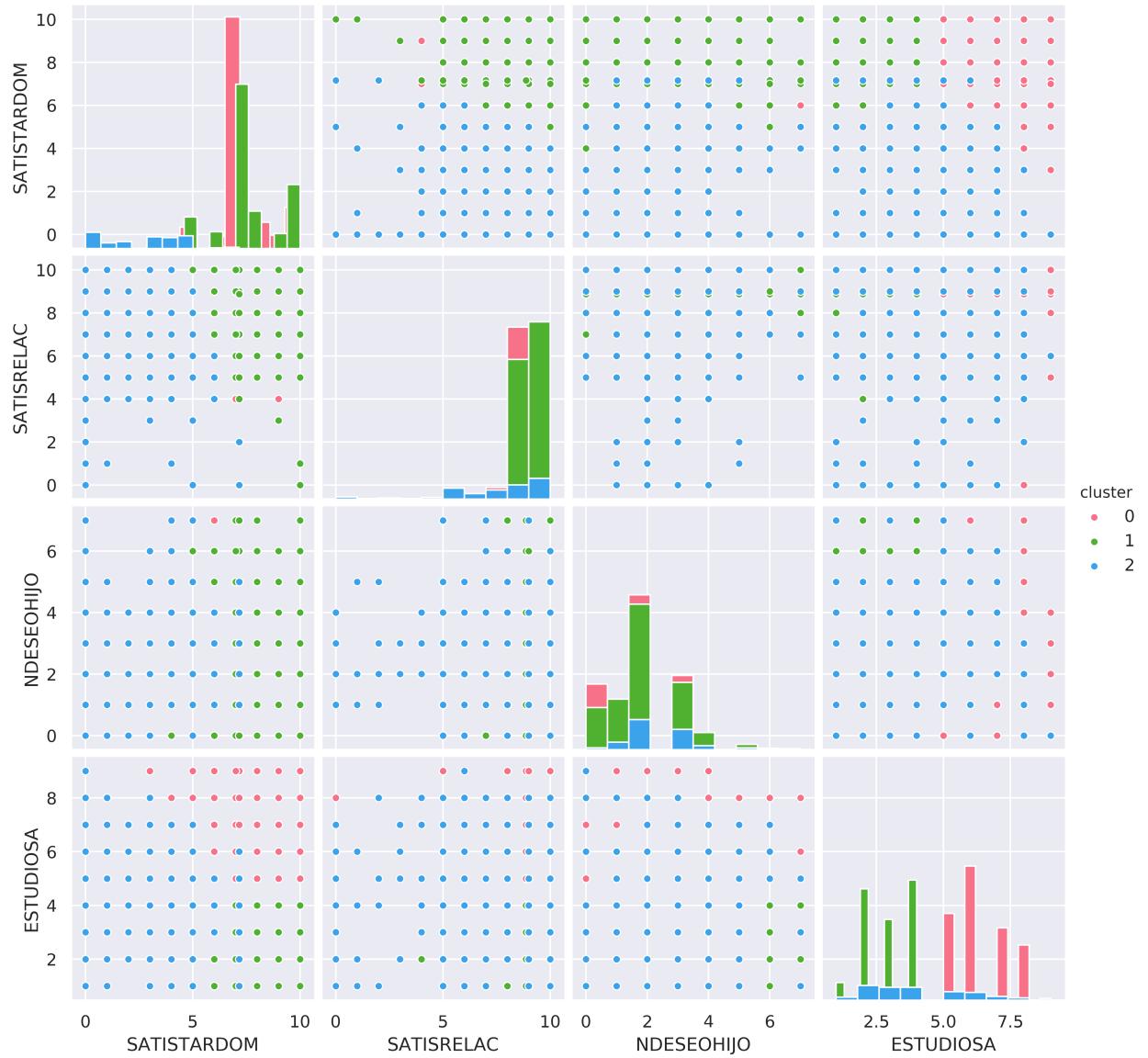


Figura 4: Scatter matrix para K-means

Hay agrupación por alta satisfacción en la relación en los clusters 0 y 1, con alta satisfacción en reparto de tareas domésticas, donde el cluster 0 destaca con mayor nivel de estudios.

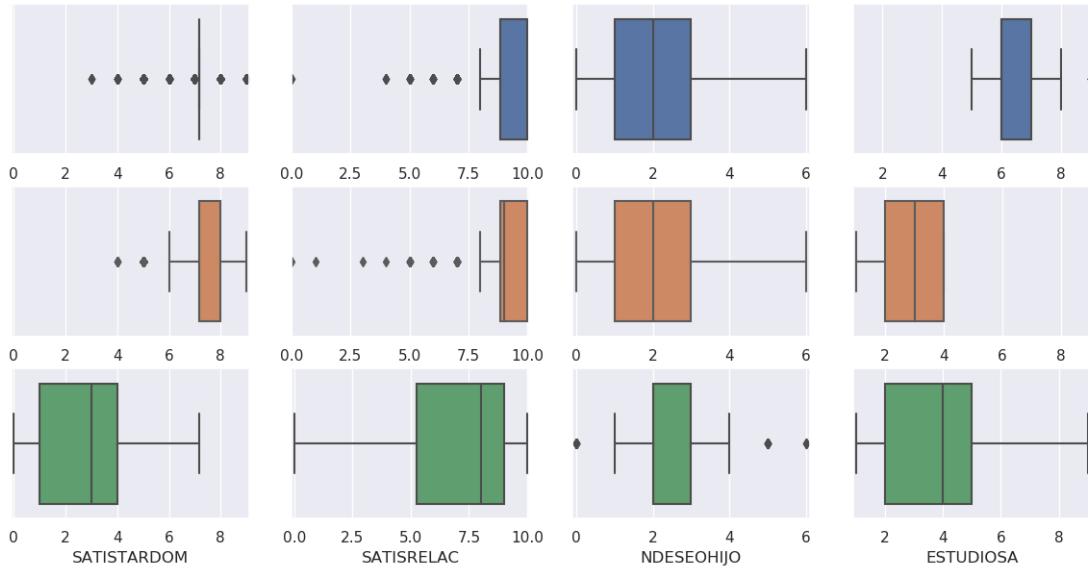


Figura 5: Boxplot para K-means

2.2. Mean Shift

Para el algoritmo Mean Shift el número de clusters lo calcula él mismo, aunque hay que establecer ciertos parámetros en la declaración de este en el código.

```

1 bw = estimate_bandwidth(X_normal, quantile=0.2, n_samples=500,
2   random_state=123456)
3 ms = MeanShift(bandwidth=bw, bin_seeding=True)

```

Los resultados obtenidos son:

Clusters	Tamaño de cada cluster	Calinski-Harabasz	Silhouette	Tº ejec
6	0: 5054 (94.27 %) 1: 91 (1.70 %) 2: 41 (0.76 %) 3: 38 (0.71 %) 4: 17 (0.32 %) 5: 120 (2.24 %)	232.640	0.29614	1.35

Tabla 3: Resultados de Mean Shift en el caso de estudio 1

El número de clusters formado es muy amplio, la mayoría de ellos con muy pocos objetos, lo cual hace que se formen grupos muy reducidos con características muy similares.

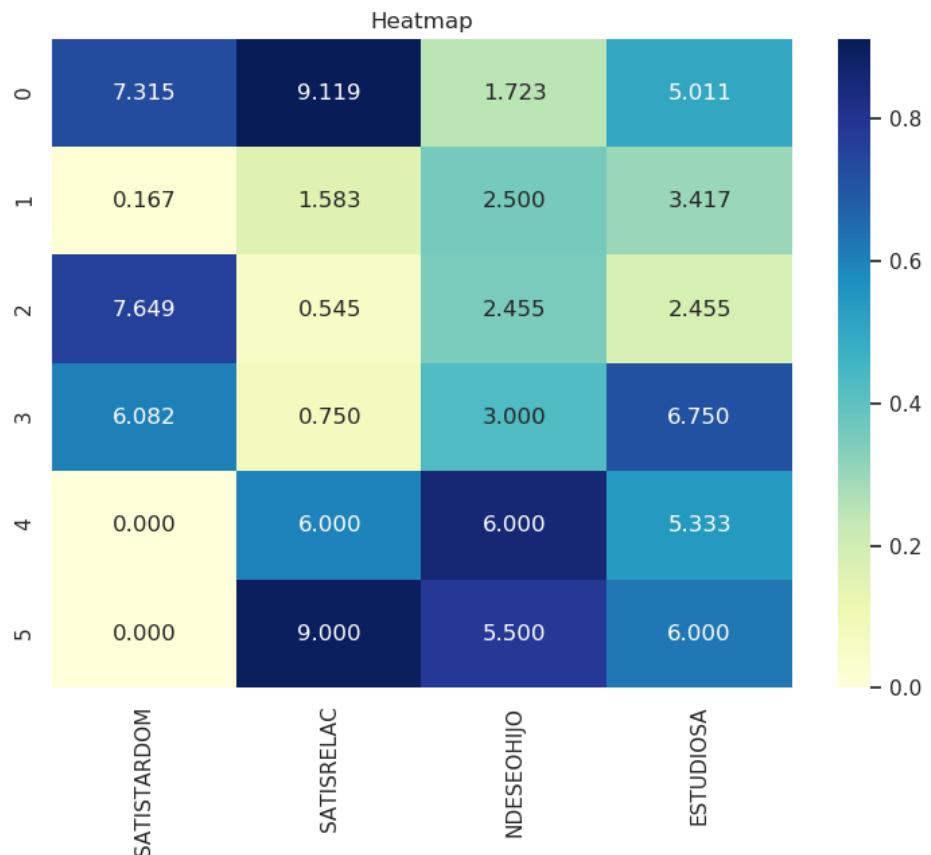


Figura 6: Heatmap para Mean Shift

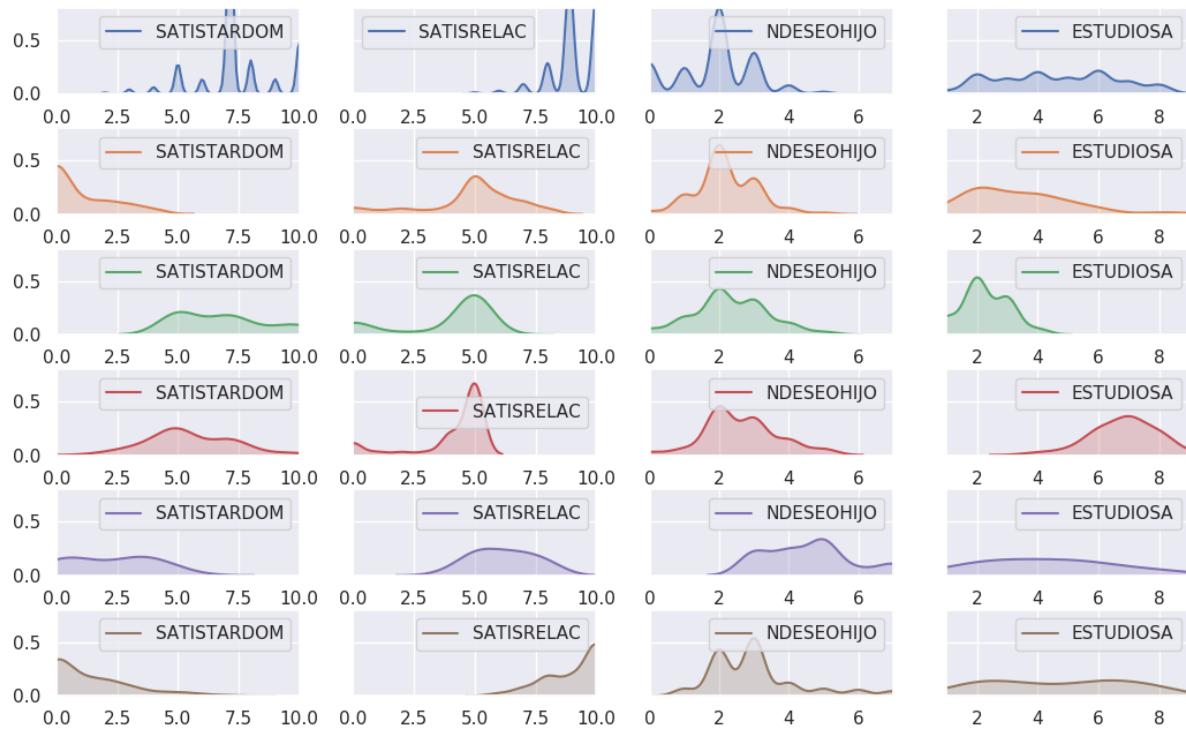


Figura 7: KDE para Mean Shift

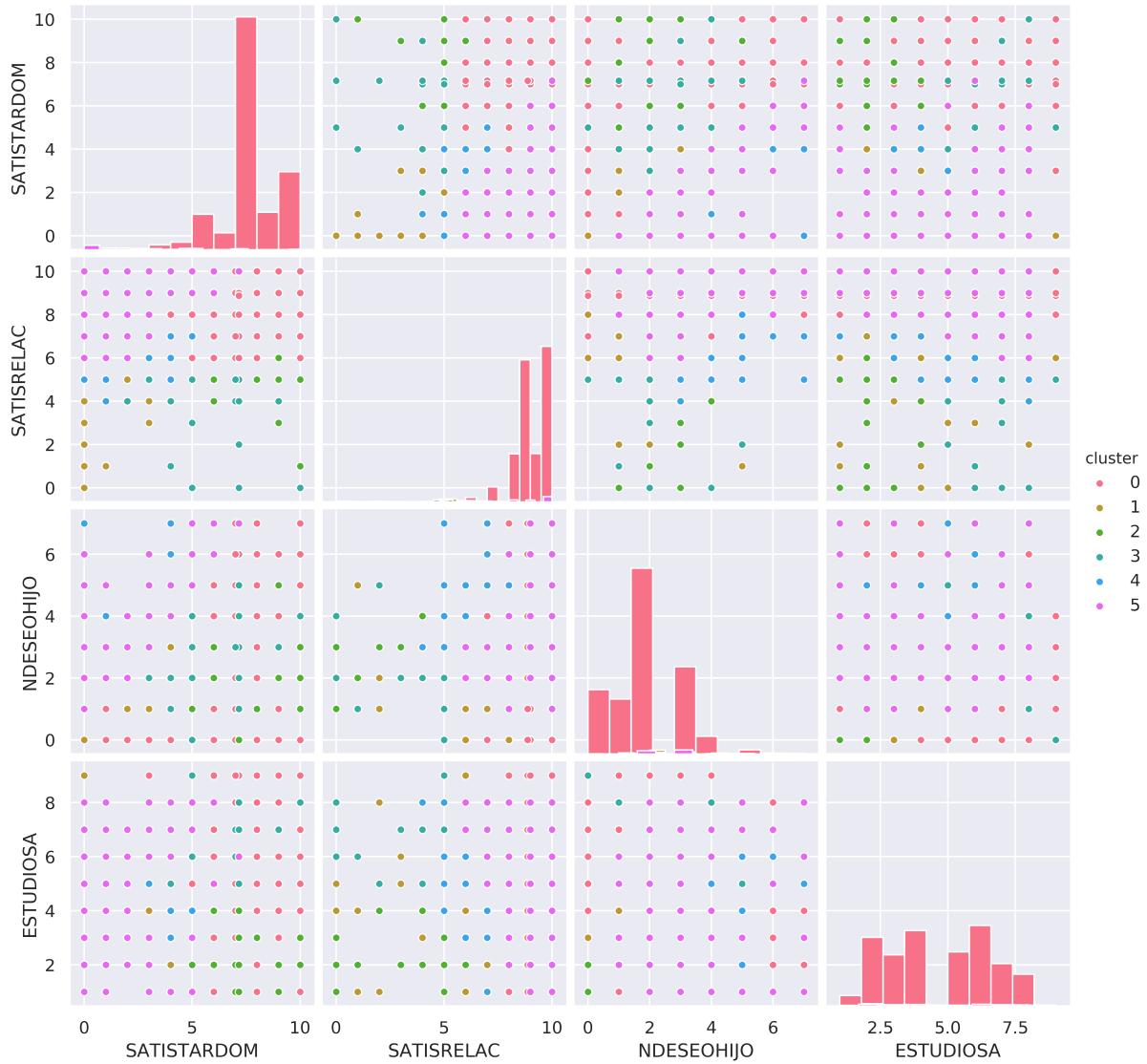


Figura 8: Scatter matrix para Mean Shift

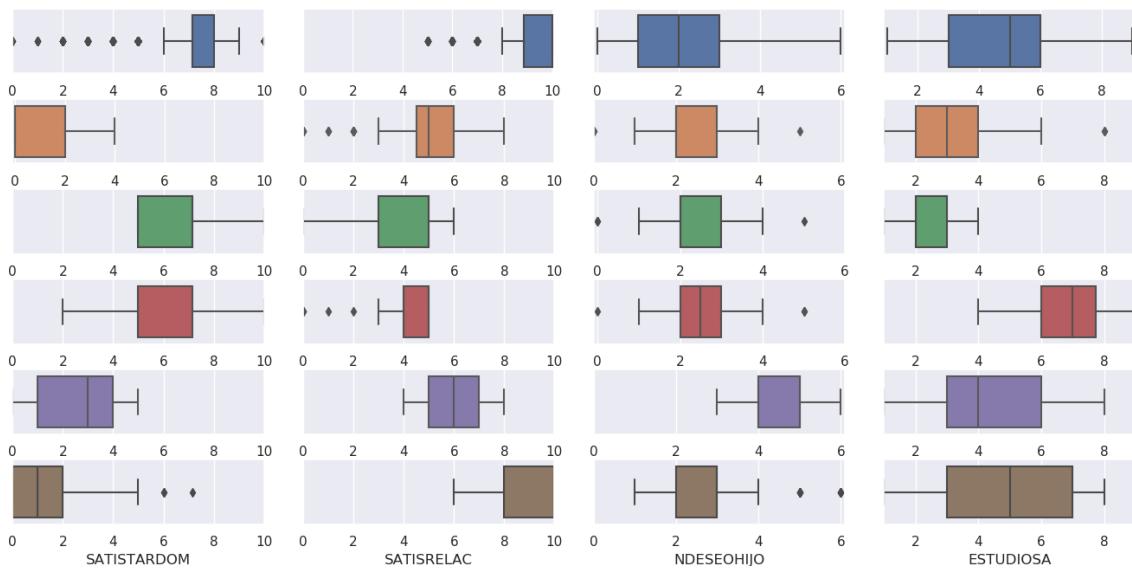


Figura 9: Boxplot para Mean Shift

2.3. DBSCAN

DBSCAN necesita un valor *epsilon* para ser configurado, el cual establece la distancia máxima entre dos muestras para que una sea considerada como vecina de la otra. Este es el parámetro más importante de DBSCAN para elegir apropiadamente el conjunto de datos y función de distancia. He probado con los siguientes valores de *epsilon*, cuyos resultados han sido los siguientes:

Epsilon	Clusters	Calinski-Harabasz	Silhouette	Tº ejec
0.15	0: 5213 (97.24 %) -1: 141 (2.63 %) 1: 7 (0.13 %)	126.824	0.38253	0.31
0.2	0: 5299 (98.84 %) -1: 56 (1.04 %) 1: 6 (0.11 %)	96.946	0.48320	0.35
0.25	0: 5331 (99.44 %) -1: 23 (0.43 %) 1: 7 (0.13 %)	57.608	0.49919	0.43
0.3	0: 5351 (99.81 %) -1: 10 (0.19 %)	41.326	0.53719	0.54

Tabla 4: Resultados de DBSCAN en el caso de estudio 1

Se puede apreciar que los índices de las medidas de rendimiento son bastante pobres y la clusterización muy similar; al aumentar el valor de *epsilon* aumentan el coste computacional (tiempo) y el valor de *Silhouette* pero disminuye el valor de *Calinski-Harabasz*, mientras que el número y tamaño de los clusters apenas varía. He decidido, por tanto, elegir un valor de *epsilon*=0.2, ya que tiene un buen equilibrio entre las métricas de rendimiento utilizadas.

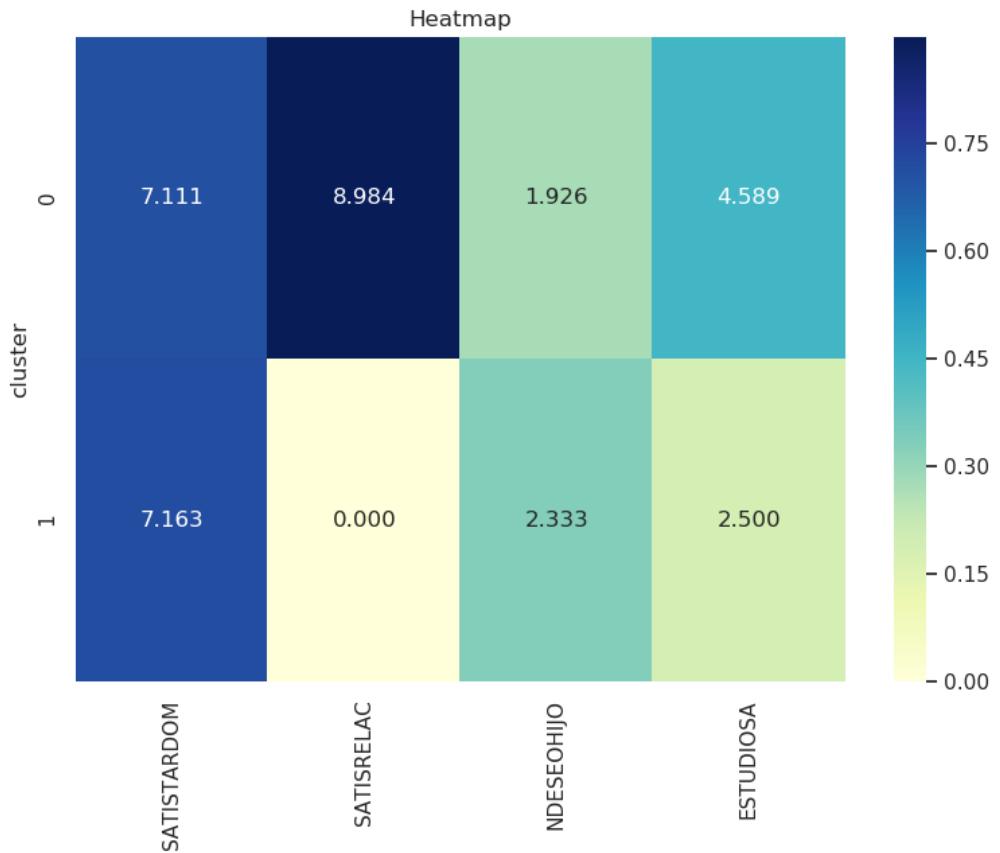


Figura 10: Heatmap para DBSCAN

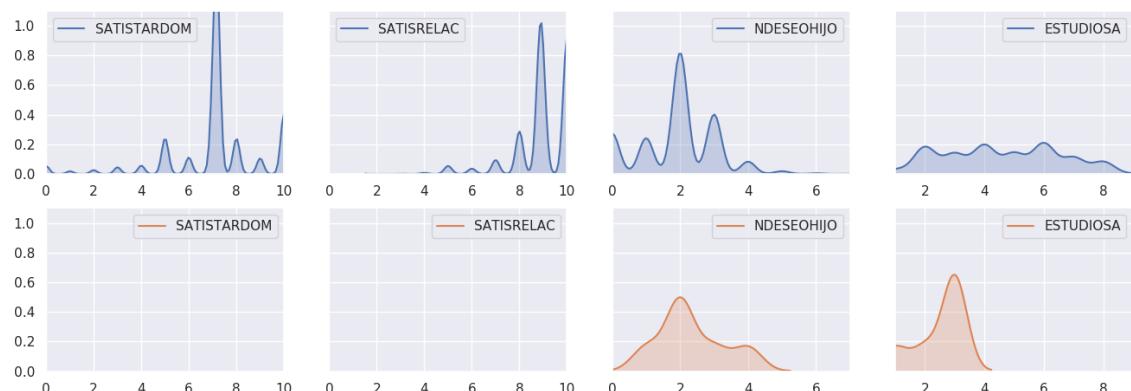


Figura 11: KDE para DBSCAN

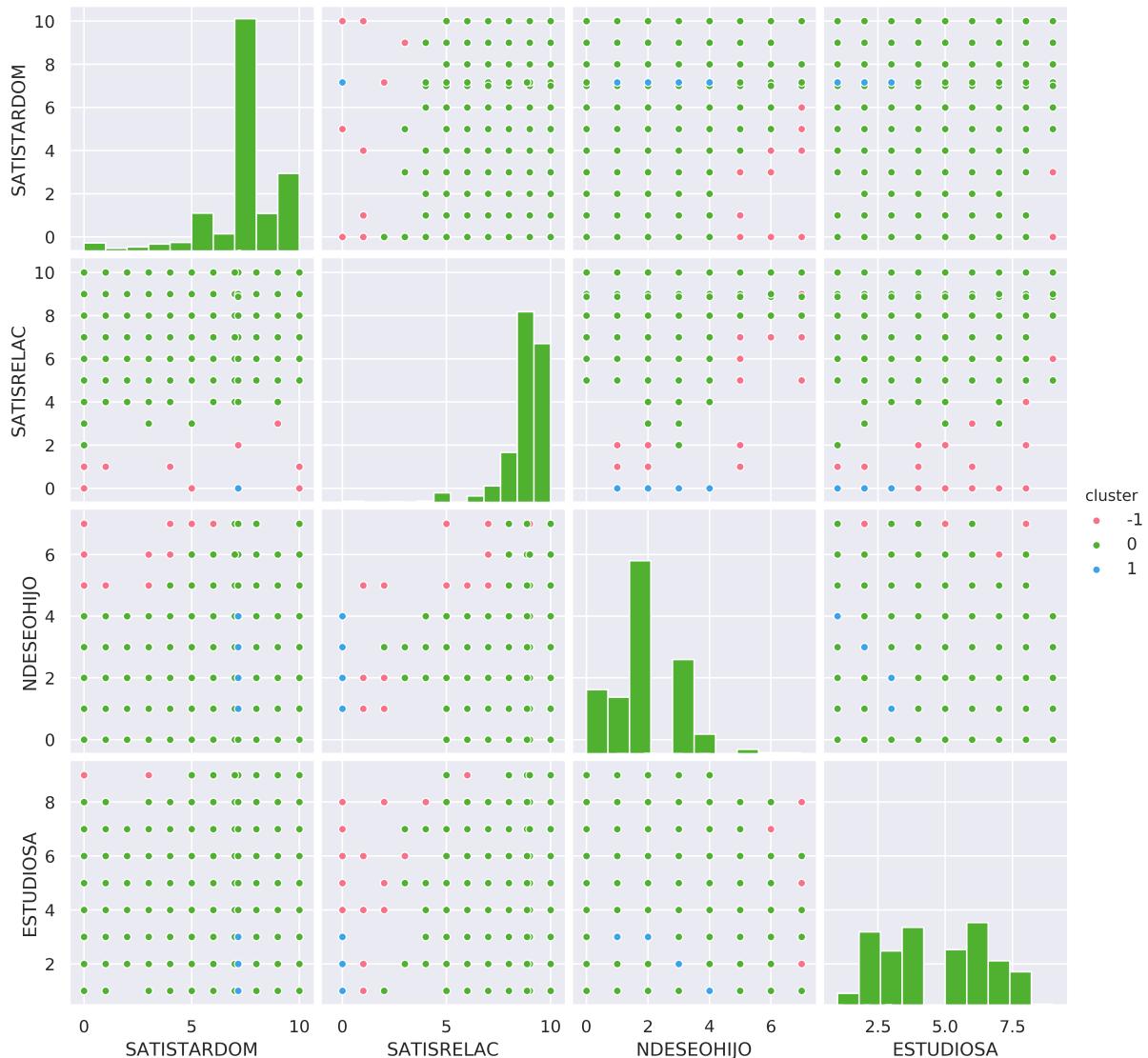


Figura 12: Scatter matrix para DBSCAN

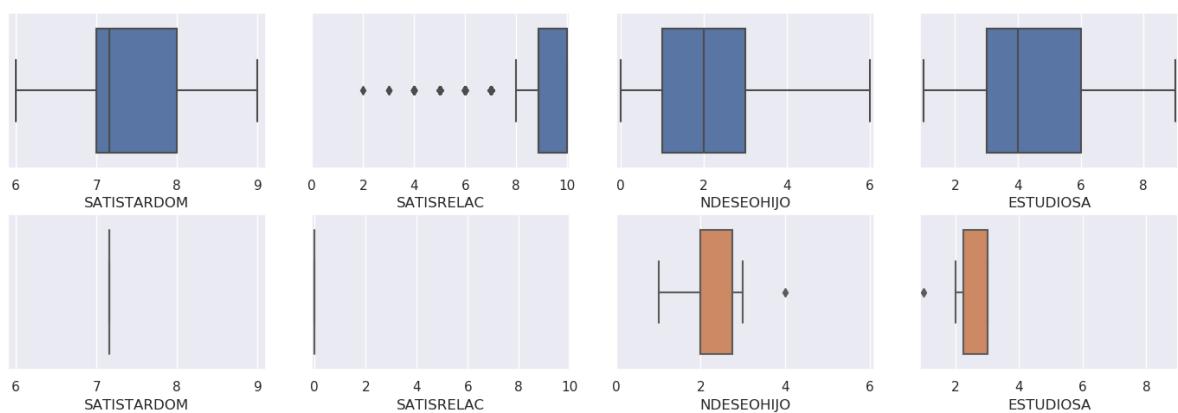


Figura 13: Boxplot para DBSCAN

2.4. Agglomerative Ward

Para el clustering jerárquico ejecuto 100 clusters, generando el siguiente dendograma:

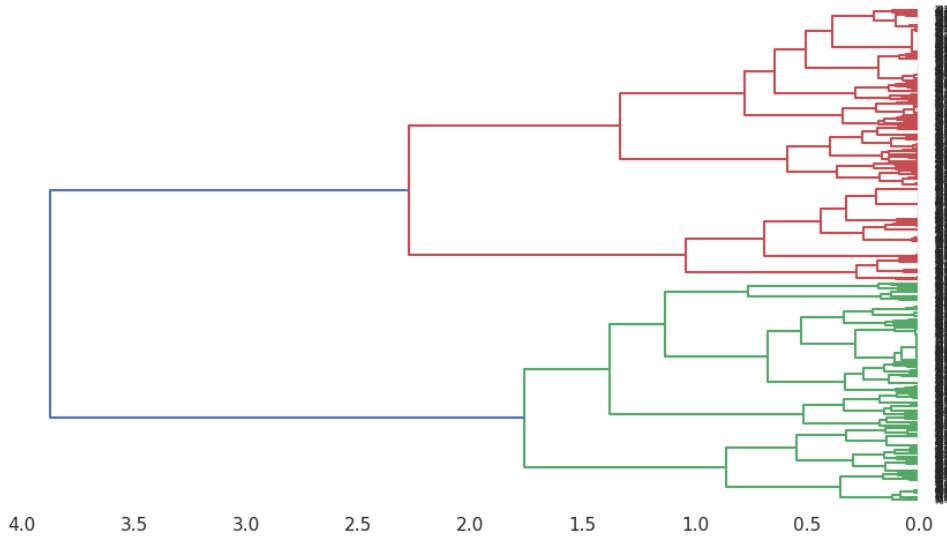


Figura 14: Dendograma

Ejecuto el algoritmo, pues, con 4 clusters, obteniendo así el siguiente dendograma incluyendo un heatmap:

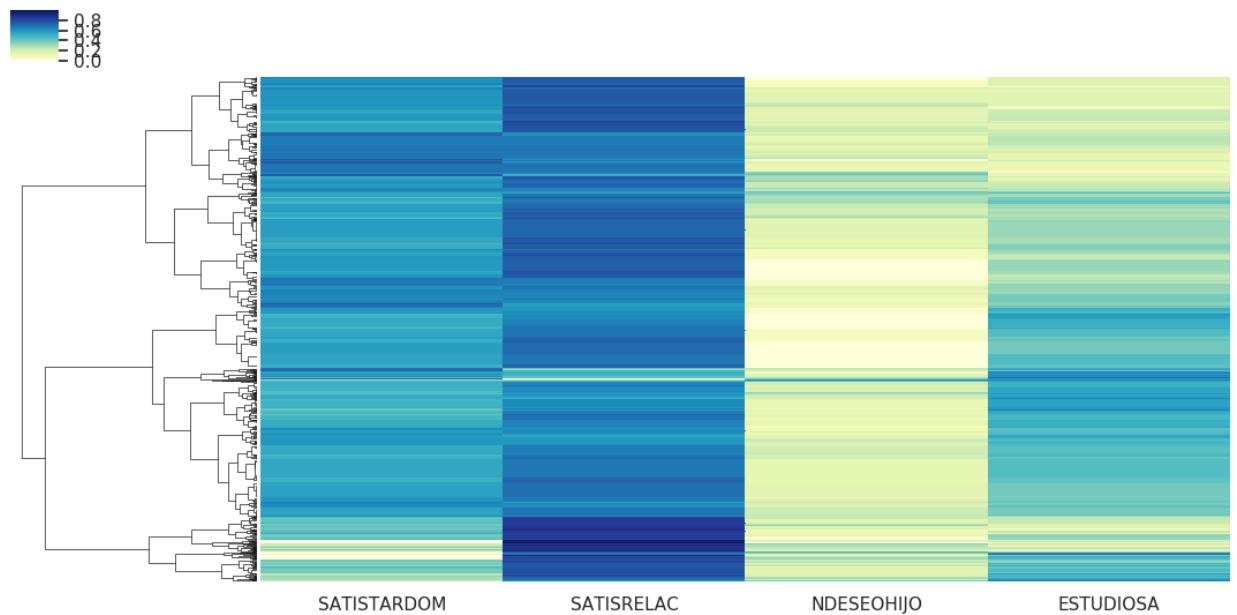


Figura 15: Dendograma con heatmap

2.5. Interpretación de la segmentación

La agrupación realizada por los distintos algoritmos difiere mucho entre ellos, teniendo desde un cluster que abarcaba casi todos los objetos con DBSCAN hasta multitud de clusters en Mean Shift. Ha sido K-means el que parece haber realizado una mejor segmentación, siendo además el que mejor índice de *Calinski-Harabasz* ha obtenido, con el segundo mejor índice de *Silhouette*, tal y como se muestra en la tabla 1. De los clusters formados se pueden destacar los siguientes grupos:

- Alto nivel de estudios –FP y universitarios en su mayoría– con una alta satisfacción tanto en el reparto de tareas domésticas como con la relación de pareja, donde el número de hijos deseado se sitúa entre 1 y 3 (predominando 2 mayoritariamente).
- Nivel de estudios medio-bajo –educación primaria o secundaria como máximo– con una satisfacción en el reparto de tareas aún mayor que en el grupo anterior y también una buena satisfacción con la pareja, haciendo que el número de hijos deseado sea consecuentemente algo mayor, aunque no muy distinto.
- El nivel de estudios está más repartido –de primaria a postsecundaria–, pero con excepciones, con una menor satisfacción en el reparto de tareas, más repartido sin importar mucho el nivel de estudios, aunque no destaca donde dicho nivel es más alto. En general muy disperso en cuanto a la satisfacción con la pareja, pero aún así se puede ver que hay muchas donde dicha satisfacción es baja. De media desean más hijos que en el resto de clusters, pero no suelen ser de más de 3 o 4.

Aunque a groso modo esos son los grupos formados, Mean Shift revela que si el nivel de estudios es mucho más alto que el resto la satisfacción con la pareja es menor, lo cual subdivide el primer grupo de la lista anterior.

3. Caso de estudio 2

Este caso de estudio analiza las mujeres con pareja según sus edades, el tipo de unión que tienen y el año desde el que viven juntos para ver cómo influye la edad de cada generación en el tipo de pareja que se forma en función de dichas variables. Se ha limitado el año de nacimiento de la pareja a, como máximo, 1950.

Las variables utilizadas son:

- *EDAD*. Edad de la persona entrevistada.
- *ANONPAR*. Año de nacimiento de la pareja.
- *TIPOUNION*. Tipo de unión que la entrevistada tiene con su pareja (1=matrimonio, 2=pareja de hecho registrada y 3=pareja de hecho sin registrar).
- *ANOVIVJUN*. Año en el que empezaron a vivir juntos.

Algoritmo	Clusters	Calinski-Harabasz	Silhouette	Tº ejec
K-means	2	15602.972	0.53150	0.11
Mean Shift	2	15601.695	0.53150	3.40
DBSCAN	2	15498.286	0.53398	2.09
Birch	2	15602.972	0.53150	0.46
AC	5			0.04

Tabla 5: Resultados de los algoritmos en el caso de estudio 1

Todos los algoritmos ofrecen resultados muy similares en las medidas de *Calinski-Harabasz* y *Silhouette* con dos clusters, ya sea porque eligen este número como óptimo o porque indican que es el adecuado cuando se realiza ajuste de parámetros.

3.1. K-means

Tal y como se hizo en el caso anterior, utilice “Elbow method”, para comprobar en qué número de clusters se forma un codo, y se vuelve a dar el caso en el cual este se forma tanto con dos clusters como con tres, así que veremos con las métricas de *Calinski-Harabasz* y *Silhouette* cuál es el número óptimo de clusters a escoger.

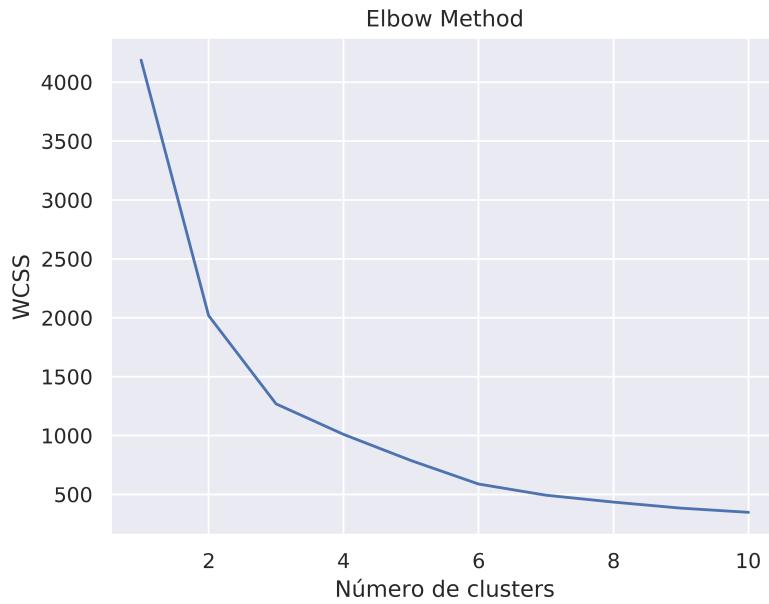


Figura 16: Elbow method

Clusters	Tamaño de cada cluster	Calinski-Harabasz	Silhouette	Tº ejec
2	0: 3260 (22.44 %) 1: 11265 (77.56 %)	15602.972	0.53150	0.11
3	0: 3178 (21.88 %) 1: 4802 (33.06 %) 2: 6545 (45.06 %)	16702.353	0.43182	0.17
4	0: 3110 (21.41 %) 1: 3178 (21.88 %) 2: 2460 (16.94 %) 3: 5777 (39.77 %)	15230.877	0.40341	0.24
5	0: 1395 (9.60 %) 1: 5776 (39.77 %) 2: 1783 (12.28 %) 3: 3111 (21.42 %) 4: 2460 (16.94 %)	15667.491	0.39706	0.32
6	0: 3661 (25.20 %) 1: 1768 (12.17 %) 2: 1411 (9.71 %) 3: 3501 (24.10 %) 4: 2100 (14.46 %) 5: 2084 (14.35 %)	17763.622	0.47281	0.35

Tabla 6: Resultados de K-means en el caso de estudio 2

La tabla 6 indica que existe un mejor número de clusters que tres, y es seis, pero aún así escojo dos clusters ya que tiene un mayor valor en *Silhouette* y la disminución de *Calinski-Harabasz* no es muy significativa.

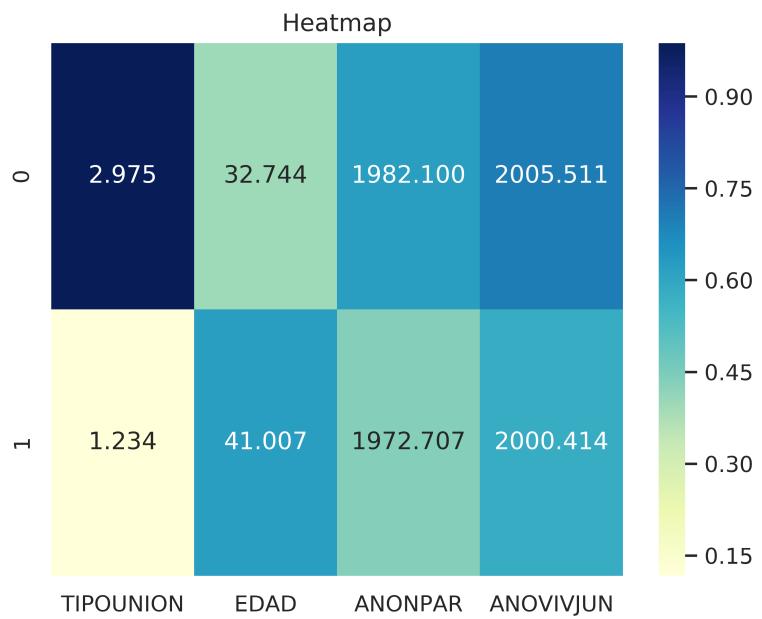


Figura 17: Heatmap para K-means

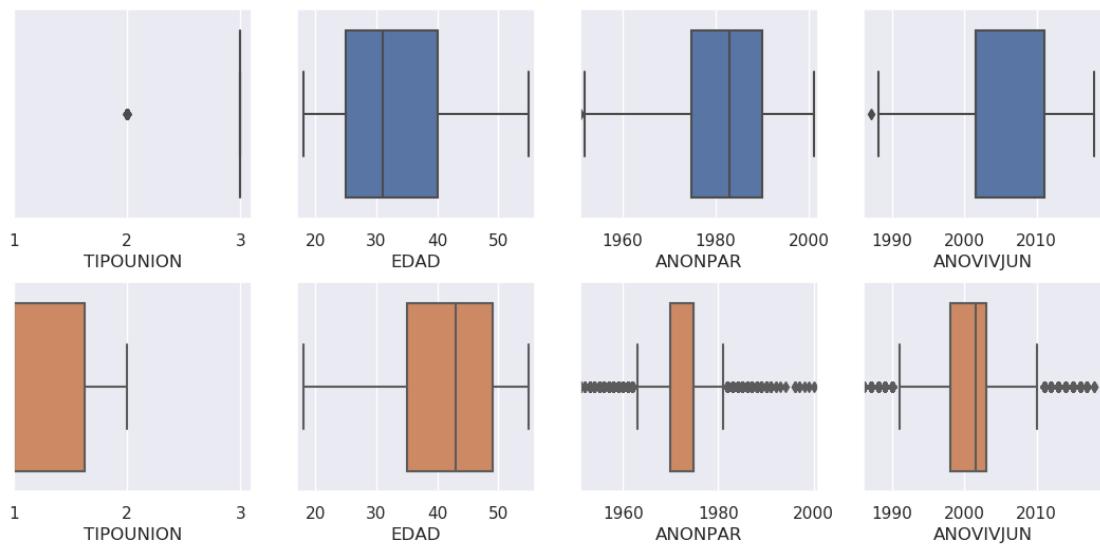


Figura 18: Boxplot para K-means

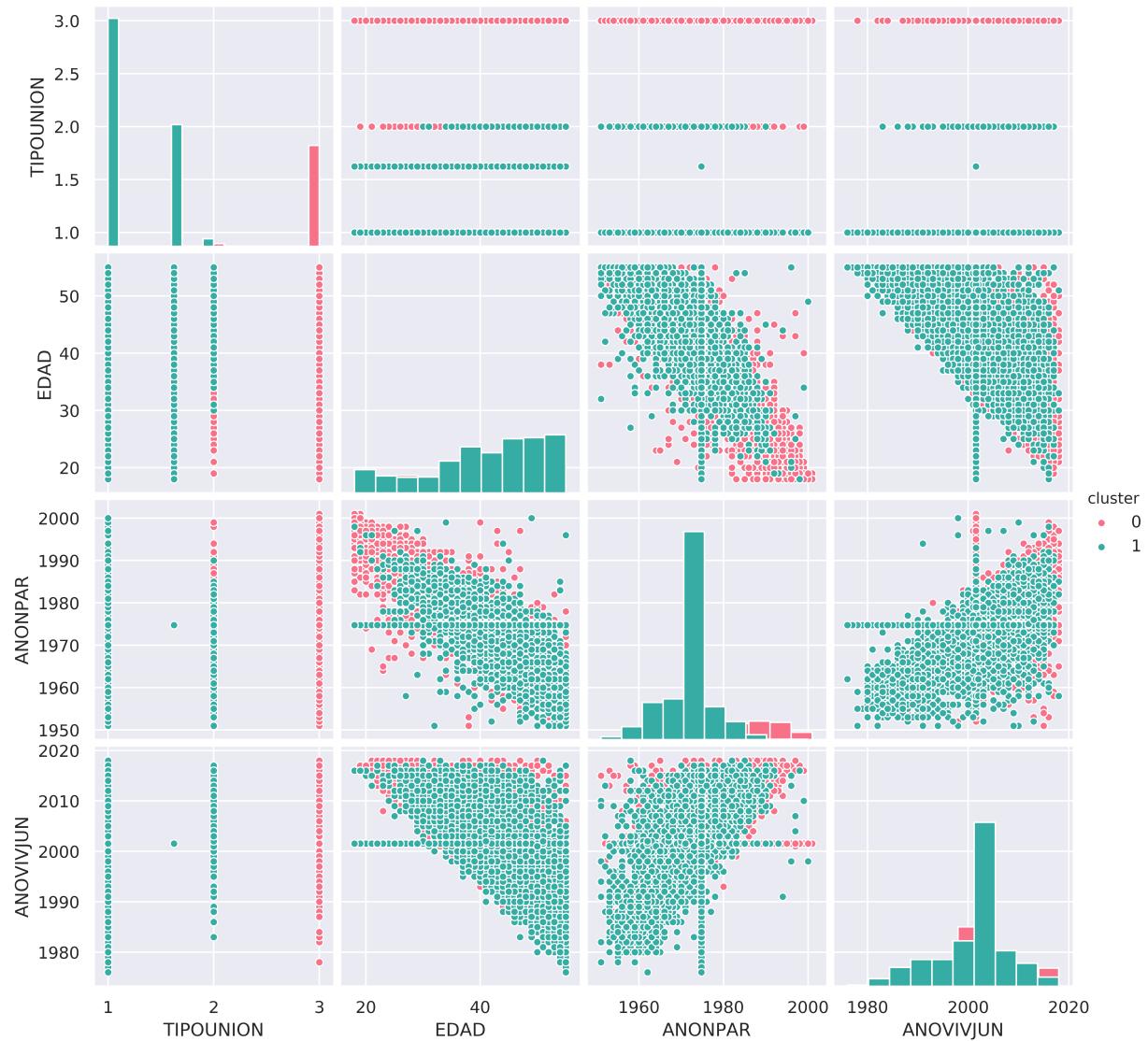


Figura 19: Scatter matrix para K-means

3.2. Mean Shift

Los resultados obtenidos con Mean Shift son:

Clusters	Tamaño de cada cluster	Calinski-Harabasz	Silhouette	Tº ejec
2	0: 11258 (77.51 %) 1: 3267 (22.49 %)	15601.695	0.53150	3.40

Tabla 7: Resultados de Mean Shift en el caso de estudio 2

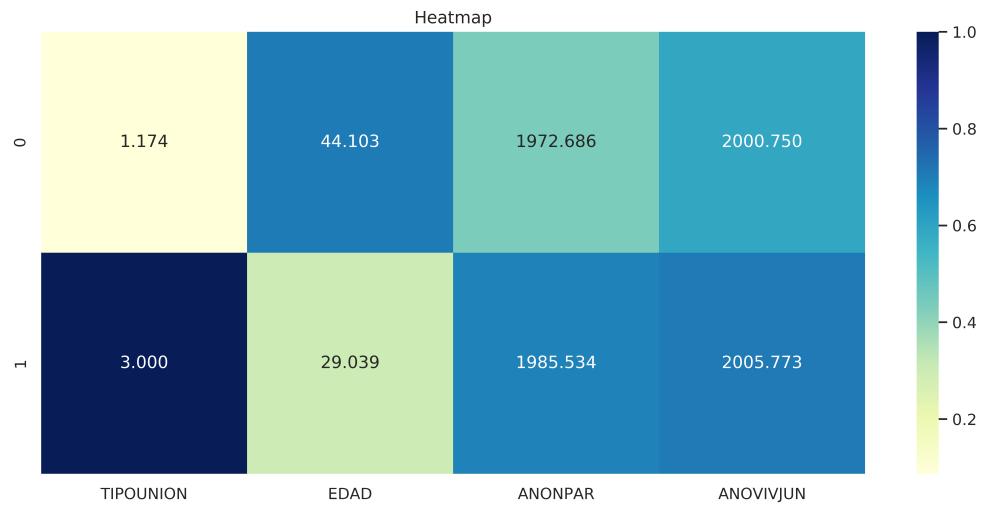


Figura 20: Heatmap para Mean Shift

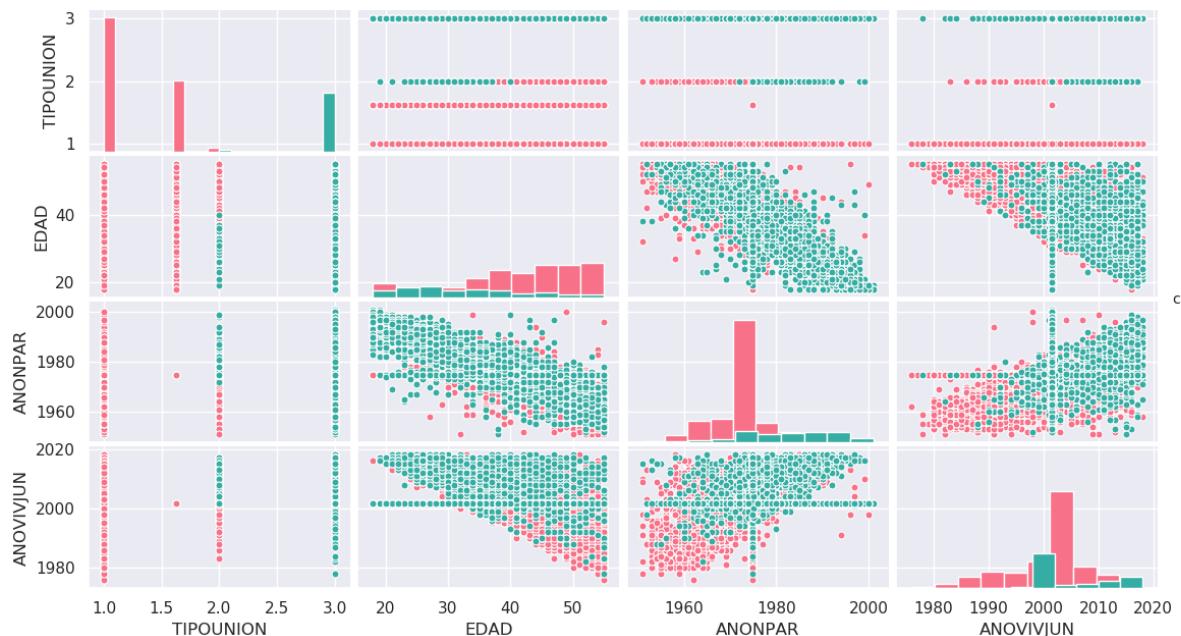


Figura 21: Scatter matrix para Mean Shift

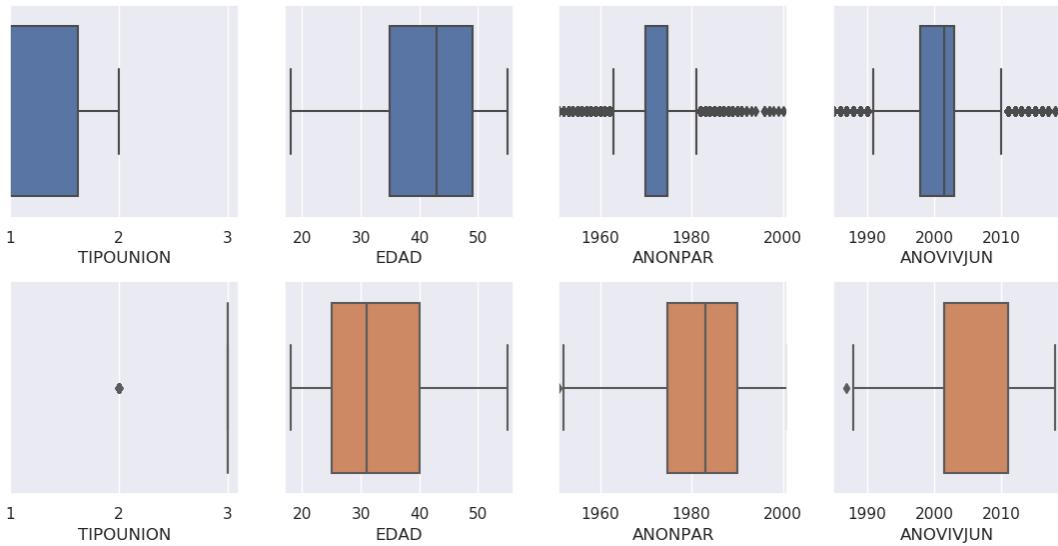


Figura 22: Boxplot para Mean Shift

3.3. DBSCAN

Los resultados de las medidas de rendimiento según el valor de *epsilon* para este caso de estudio son:

Epsilon	Clusters	Calinski-Harabasz	Silhouette	Tº ejec
0.15	0: 3839 (26.43 %) 1: 7179 (49.43 %) 2: 3174 (21.85 %) 3: 303 (2.09 %) -1: 30 (0.21 %)	6133.526	0.29929	0.82
0.2	0: 4153 (28.59 %) 1: 7187 (49.48 %) 2: 3176 (21.87 %) -1: 9 (0.06 %)	8035.610	0.32629	1.08
0.25	0: 4156 (28.61 %) 1: 7187 (49.48 %) 2: 3178 (21.88 %) -1: 4 (0.03 %)	8042.587	0.32989	1.24
0.3	0: 4157 (28.62 %) 1: 7187 (49.48 %) 2: 3178 (21.88 %) -1: 3 (0.02 %)	8041.765	0.36056	1.44
0.4	0: 11347 (78.12 %) 1: 3178 (21.88 %)	15498.286	0.53398	2.09

Tabla 8: Resultados de DBSCAN en el caso de estudio 2

Sin lugar a dudas el mejor valor es $\text{epsilon}=0.4$, aunque el tiempo de ejecución sea

superior al obtenido con el resto de valores.

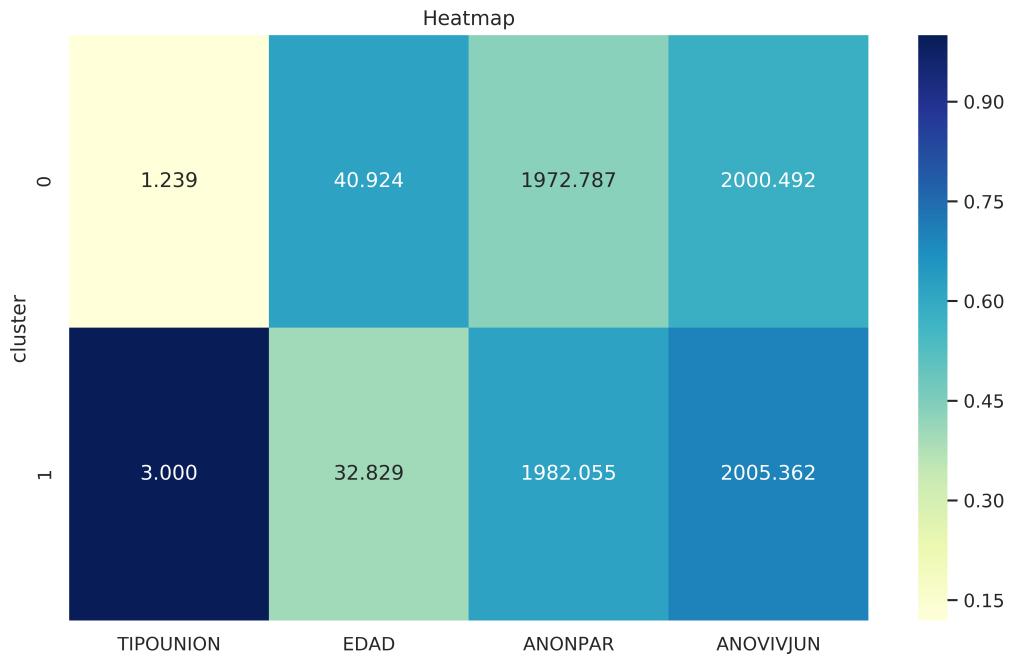


Figura 23: Heatmap para DBSCAN

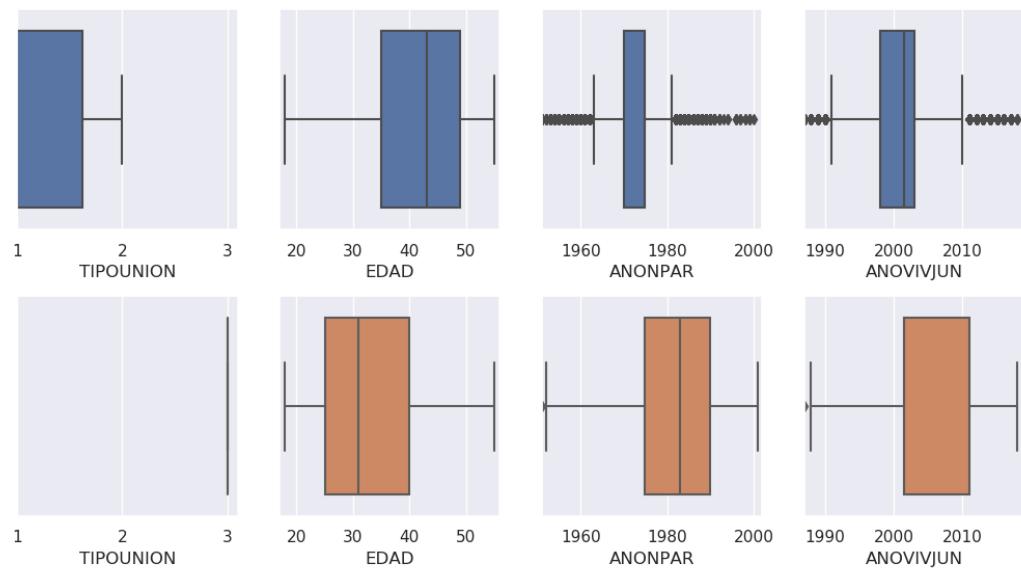


Figura 24: Boxplot para DBSCAN

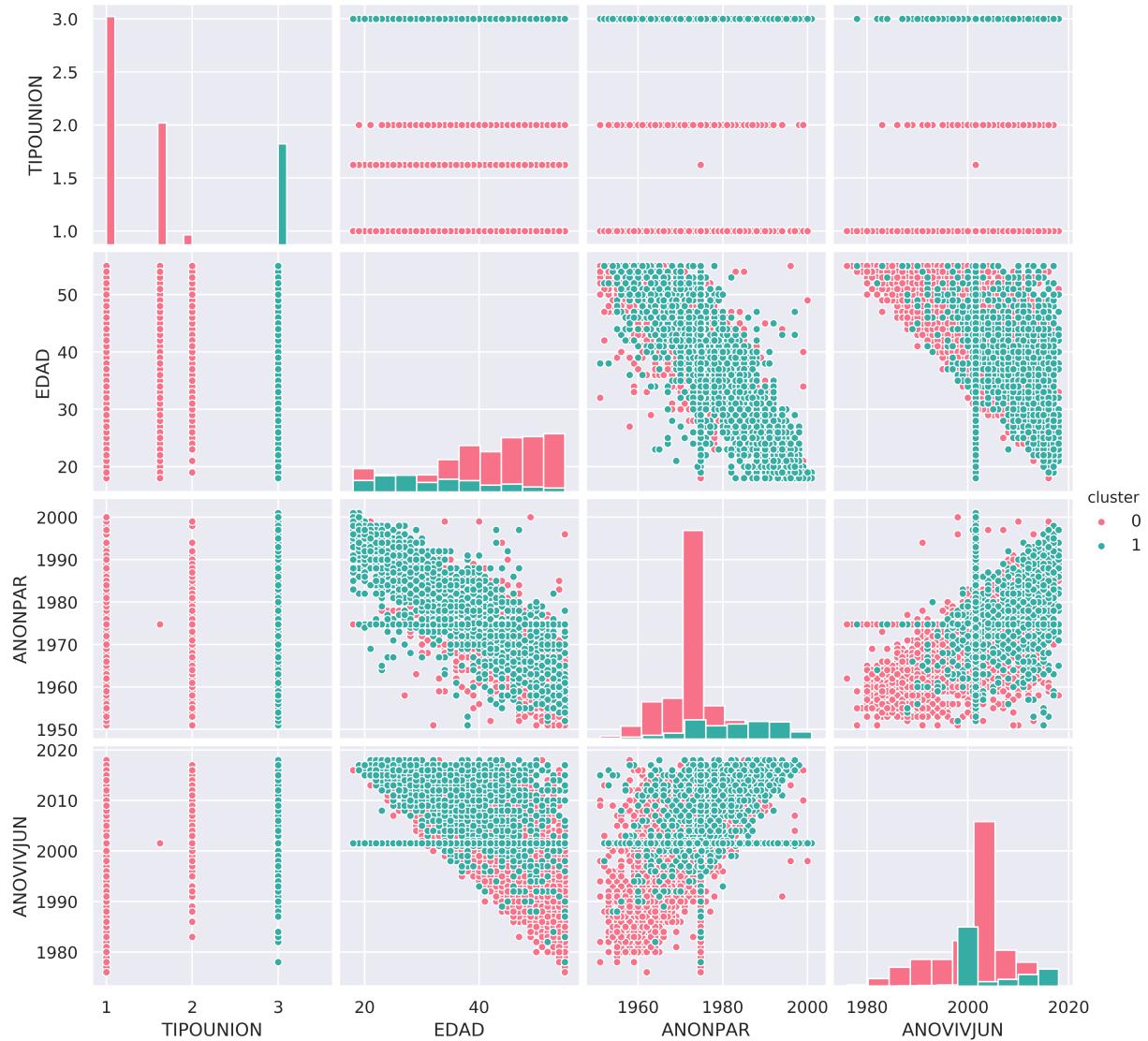


Figura 25: Scatter matrix para DBSCAN

3.4. Birch

Este algoritmo sólo ha funcionado estableciendo el número de clusters en dos:

Clusters	Tamaño de cada cluster	Calinski-Harabasz	Silhouette	Tº ejec
2	0: 3260 (22.44 %) 1: 11265 (77.56 %)	15602.972	0.53150	0.46

Tabla 9: Resultados de Birch en el caso de estudio 2

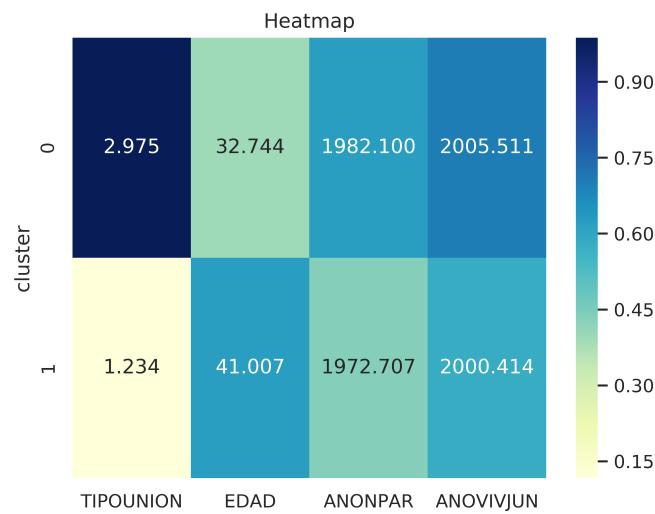


Figura 26: Heatmap para Birch

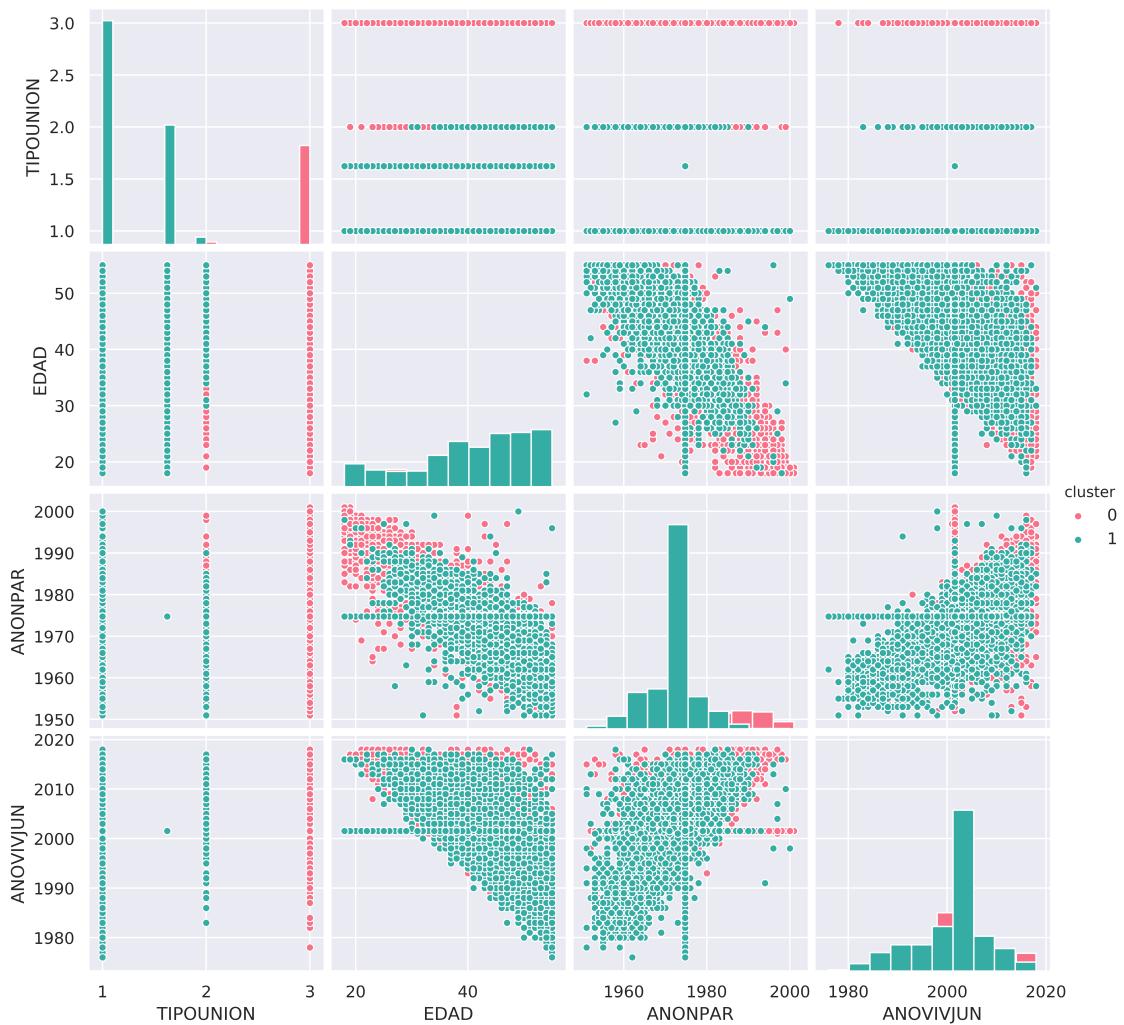


Figura 27: Scatter matrix para Birch

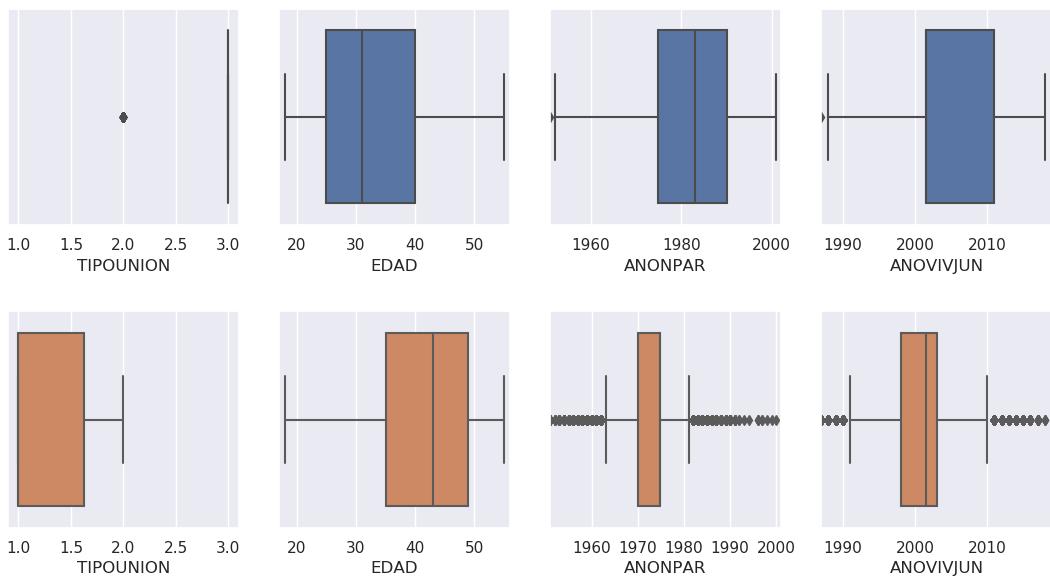


Figura 28: Boxplot para Birch

3.5. Agglomerative Ward

Para el clustering jerárquico ejecuto 100 clusters, generando el siguiente dendograma:

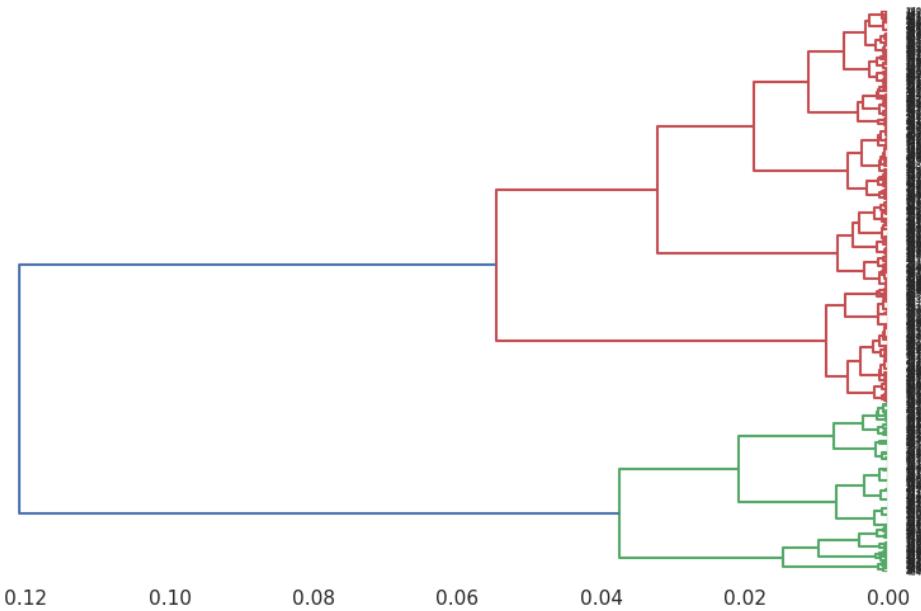


Figura 29: Dendrograma

En base al dendograma obtenido ejecuto el algoritmo con 5 clusters para generar el dendograma incluyendo un heatmap:

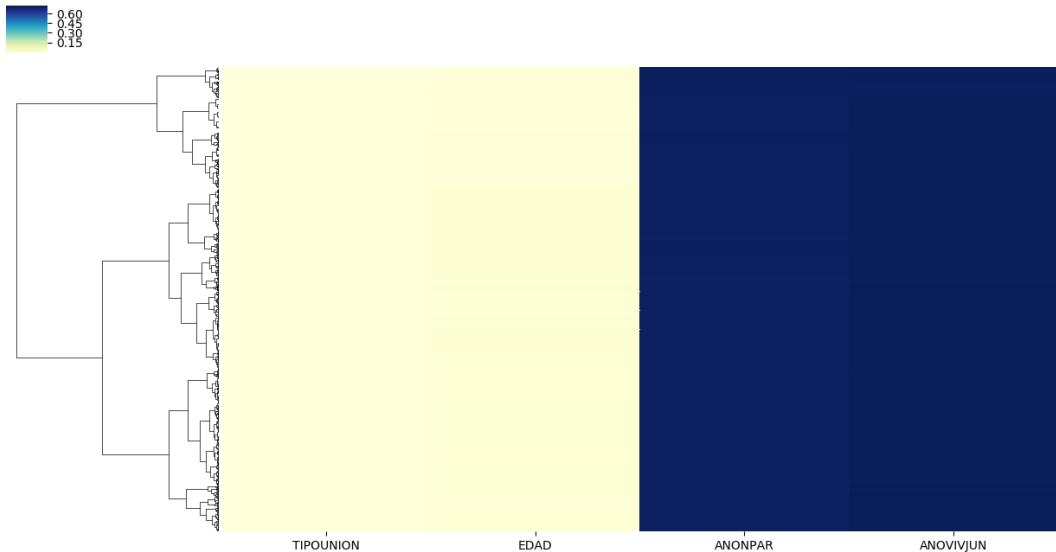


Figura 30: Dendograma con heatmap

3.6. Interpretación de la segmentación

La segmentación realizada por los diferentes algoritmos es muy similar, diferenciando dos grupos principales:

- Parejas de hecho sin registrar donde las mujeres entrevistadas son jóvenes –entre 25 y 40 años, 30 de media– y la pareja es de su misma edad, aunque puede variar, con cierta tendencia a que sea unos pocos años mayor (entre 3 y 5) y que llevan viviendo juntos entre 5 y 15 años, aunque hay cierta dispersión respecto al otro grupo.
- Parejas generalmente casadas, o parejas de hecho registradas en su defecto, donde la mujer entrevistada es mayor que en el grupo anterior, de mediana edad –entre 35 y 50 años, 43 de media–, su pareja tiene más o menos la misma edad, aunque con cierta tendencia a ser algo mayor, y que llevan viviendo juntos entre 15 y 20 años.

Estos grupos consolidan una idea que se suponía obvia, lo cual ratifica lo que cualquiera puede suponer al respecto con los datos escogidos.

4. Caso de estudio 3

En este último caso de estudio el análisis se centra en aquellas mujeres donde ella o su pareja usan anticonceptivos según la edad, ingresos en el hogar, número de hijos y número de hijos deseados. El interés se encuentra en conocer cómo influye la edad de estas, los hijos que han tenido y sus ingresos para ver si tiene algún efecto en los hijos que se desean a pesar de utilizar métodos de contracción. Para seleccionar los objetos de interés se ha filtrado por $USOANTICONCEP=1$, donde el número de hijos deseado sea menor que ocho, por el mismo motivo que en el caso anterior, y con unos ingresos inferiores a 50000 € mensuales, ya que considero que una cantidad mayor no es nada representativa ni habitual y por tanto escapa de mi interés para este caso.

Las variables utilizadas son:

- *EDAD*. Edad de la persona entrevistada.
- *USOANTICONCEP*. Si la entrevistada o la pareja usan método anticonceptivos.
- *NHIJOS*. Número de hijos de la entrevistada o de su pareja.
- *NDESEOHIGO*. Número de hijos deseados, tanto si se tienen como si no.
- *INGREHOG*. Ingresos mensuales netos en el hogar.

Al igual que ocurrió en el caso de estudio 1 el algoritmo Birch no se ha podido ejecutar con ningún número de clusters.

Algoritmo	Clusters	Calinski-Harabasz	Silhouette	Tº ejec
K-means	2	8051.672	0.44222	0.13
Mean Shift	6	1184.206	0.30107	1.25
DBSCAN	1	40.678	0.50233	0.92
AC	4			0.02
Birch	-	-	-	-

Tabla 10: Resultados de los algoritmos en el caso de estudio 3

4.1. K-means

Utilizo *Elbow method* para comprobar en qué número de clusters se forma un codo. El gráfico muestra con bastante claridad que el codo se forma dos clusters, por lo que con alta probabilidad será esta la opción escogida. A continuación de la gráfica se muestra una tabla comparativa de las medidas de rendimiento típicas –*Calinski-Harabasz* y *Silhouette*–.

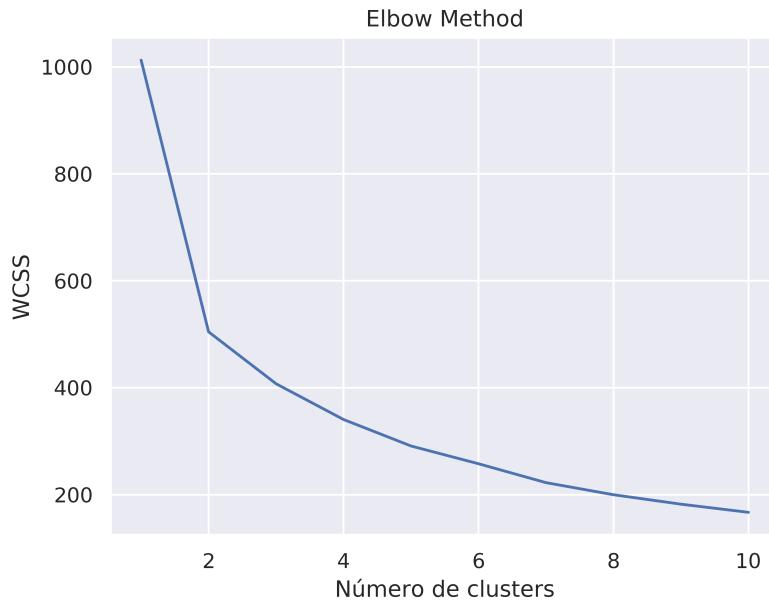


Figura 31: Elbow method

Clusters	Tamaño de cada cluster	Calinski-Harabasz	Silhouette	Tº ejec
2	0: 4709 (58.89 %) 1: 3287 (41.11 %)	8051.672	0.44222	0.13
3	0: 2931 (36.66 %) 1: 3165 (39.58 %) 2: 1900 (23.76 %)	5936.496	0.34686	0.15
4	0: 2569 (32.13 %) 1: 1977 (24.72 %) 2: 1276 (15.96 %) 3: 2174 (27.19 %)	5239.217	0.33647	0.25
5	0: 917 (11.47 %) 1: 1734 (21.69 %) 2: 1909 (23.87 %) 3: 1977 (24.72 %) 4: 1459 (18.25 %)	4948.249	0.32268	0.28
6	0: 1052 (13.16 %) 1: 1286 (16.08 %) 2: 703 (8.79 %) 3: 1766 (22.09 %) 4: 1547 (19.35 %) 5: 1642 (20.54 %)	4705.735	0.31849	0.42

Tabla 11: Resultados de K-means en el caso de estudio 3

En la tabla se confirma lo que se suponía con la gráfica de *Elbow method*, por lo que selecciono para K-means dos clusters.

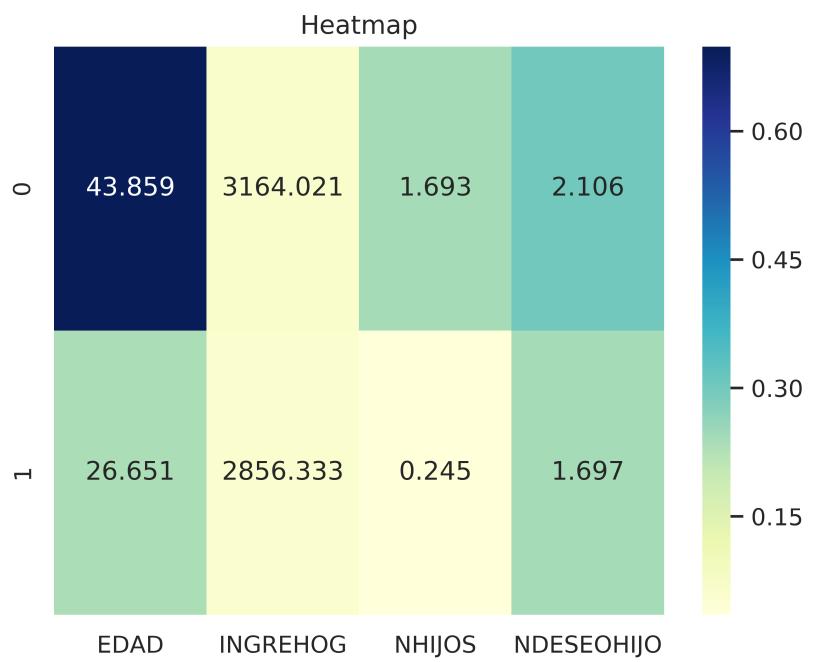


Figura 32: Heatmap para K-means

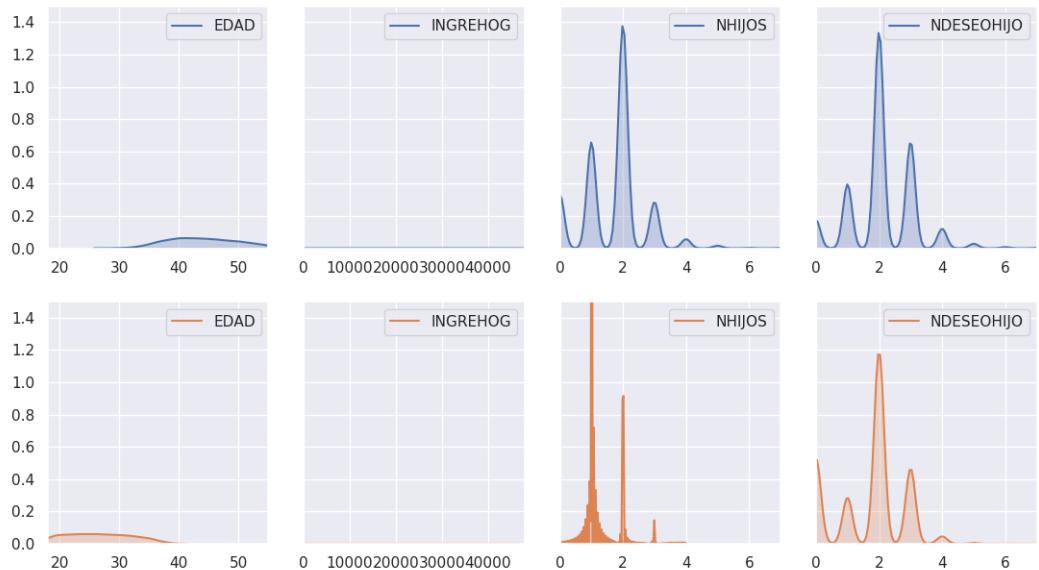


Figura 33: KDE para K-means

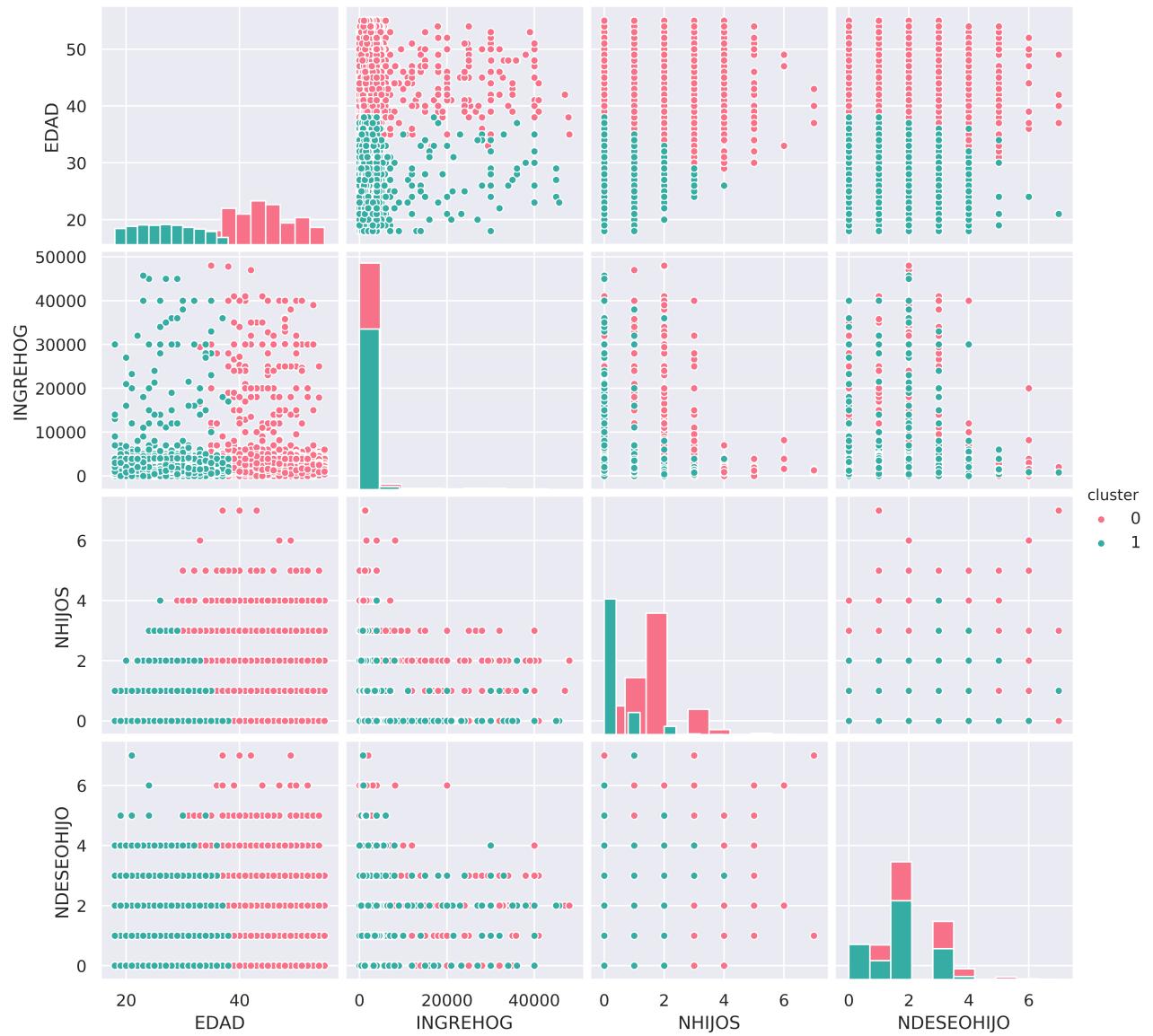


Figura 34: Scatter matrix para K-means

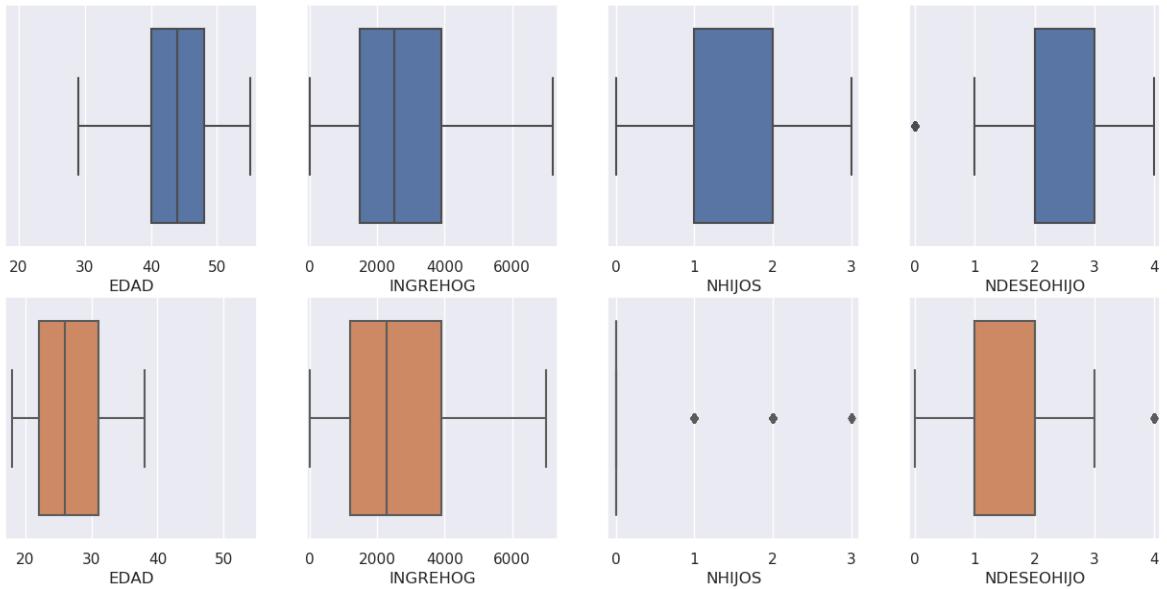


Figura 35: Boxplot para K-means

4.2. Mean Shift

Los resultados obtenidos con Mean Shift son:

Clusters	Tamaño de cada cluster	Calinski-Harabasz	Silhouette	Tº ejec
6	0: 4672 (58.43 %) 1: 3040 (38.02 %) 2: 77 (0.96 %) 3: 83 (1.04 %) 4: 42 (0.53 %) 5: 22 (0.28 %) 6: 23 (0.29 %) 7: 28 (0.35 %) 8: 1 (0.01 %) 9: 8 (0.10 %)	1184.206	0.30107	1.25

Tabla 12: Resultados de Mean Shift en el caso de estudio 3

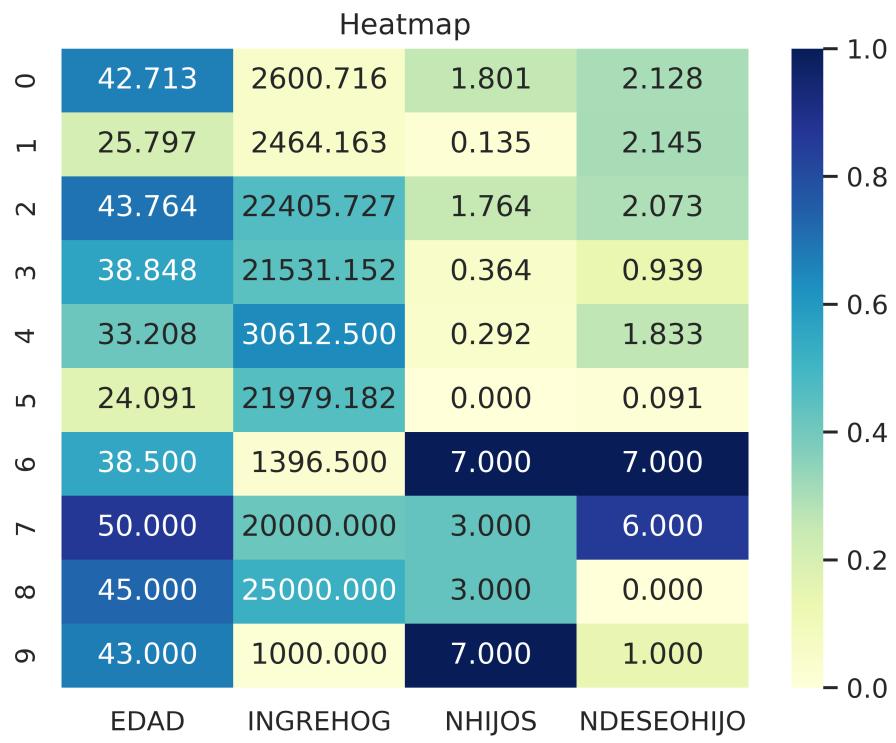


Figura 36: Heatmap para Mean Shift

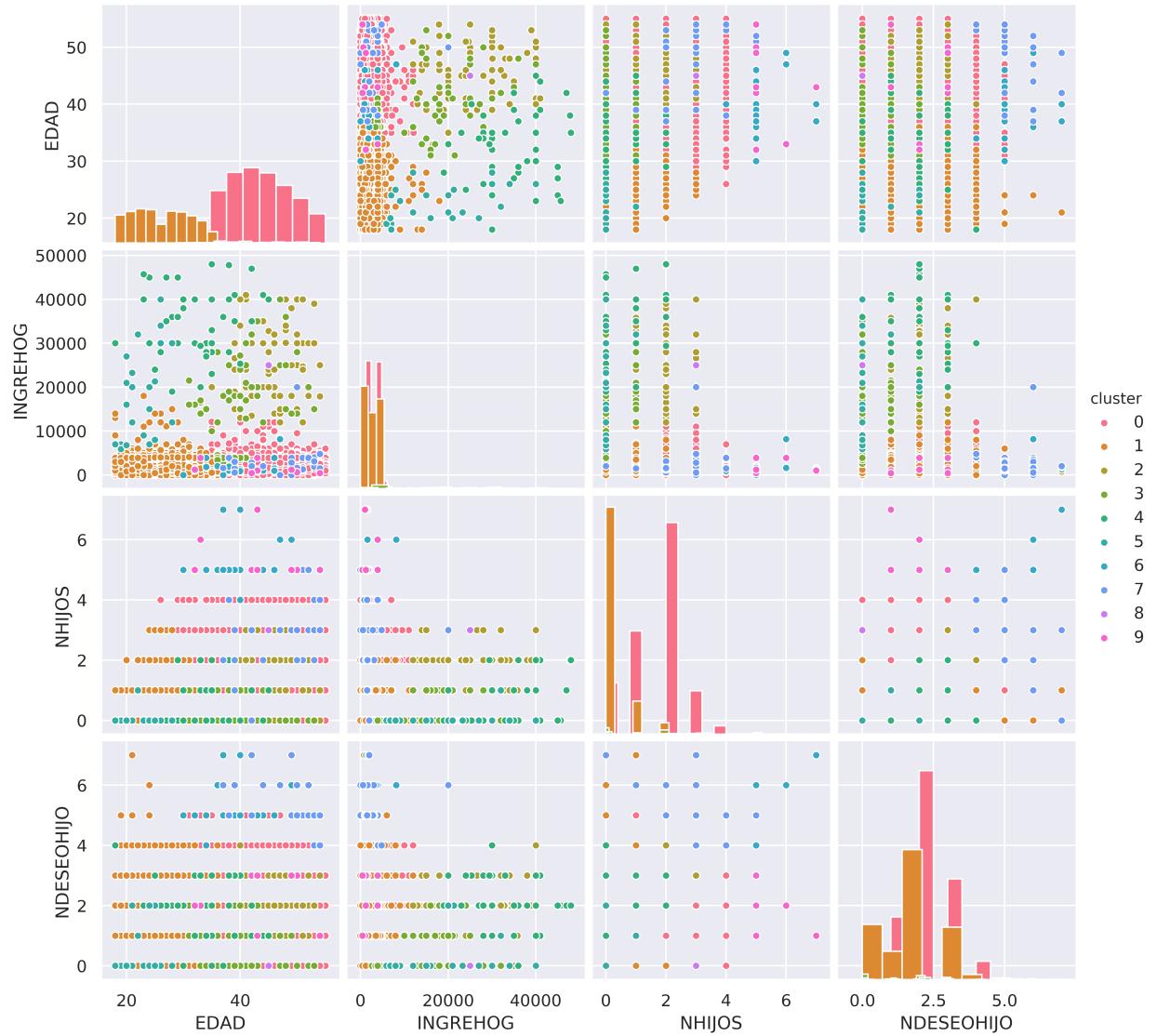


Figura 37: Scatter matrix para Mean Shift

Los dos primeros clusters vienen a reafirmar lo que se interpreta de los dos clusters formados por K-means, mientras que el resto son grupos minoritarios de casos particulares sin excepciones destacables.

4.3. DBSCAN

Los resultados de las medidas de rendimiento según el valor de ϵ para este caso de estudio son:

Epsilon	Clusters	Calinski-Harabasz	Silhouette	Tº ejec
0.15	0: 7884 (98.60 %) 1: 5 (0.06 %) 2: 5 (0.06 %) 3: 9 (0.11 %) 4: 5 (0.06 %) -1: 88 (1.10 %)	37.698	0.13992	0.74
0.2	0: 7978 (99.77 %) -1: 18 (0.23 %)	40.678	0.50233	0.92
0.25	0: 7990 (99.92 %) -1: 6 (0.08 %)	18.286	0.52197	1.12
0.3	0: 7991 (99.94 %) -1: 5 (0.06 %)	20.987	0.52300	1.35

Tabla 13: Resultados de DBSCAN en el caso de estudio 3

Los resultados son muy similares, al igual que malos, con pésimos índices de rendimiento y mala formación de clusters. Aunque ninguno tiene interés selecciono el caso con $\epsilon_{\text{epsilon}}=0.2$.

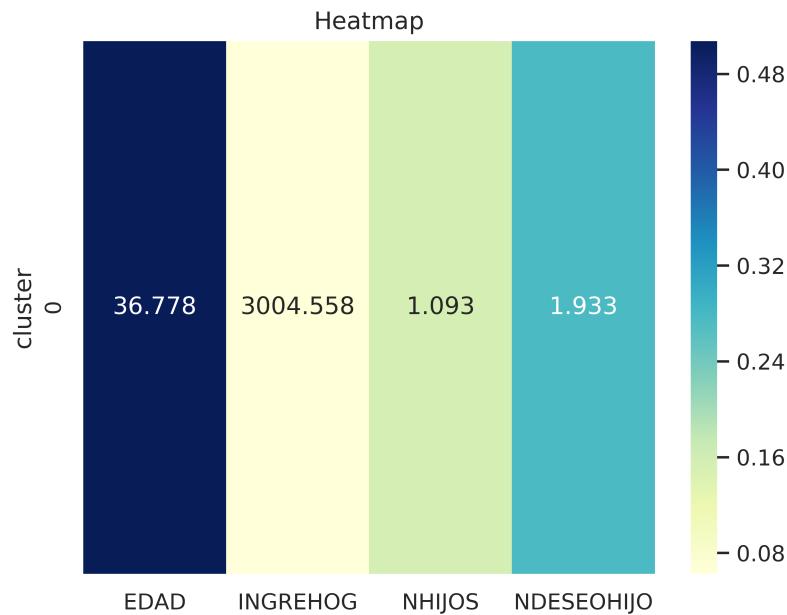


Figura 38: Heatmap para DBSCAN

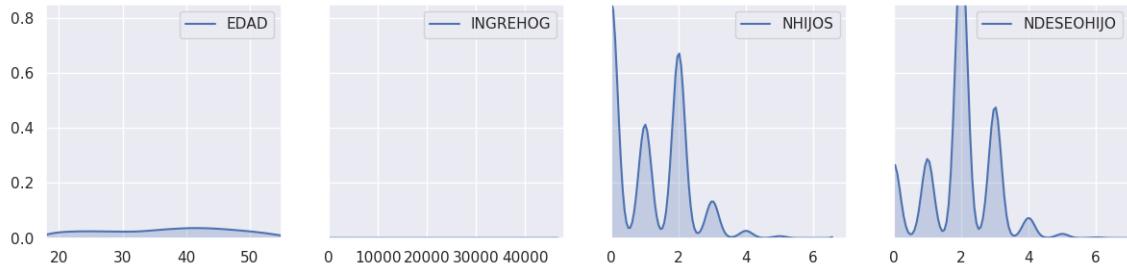


Figura 39: KDE para DBSCAN

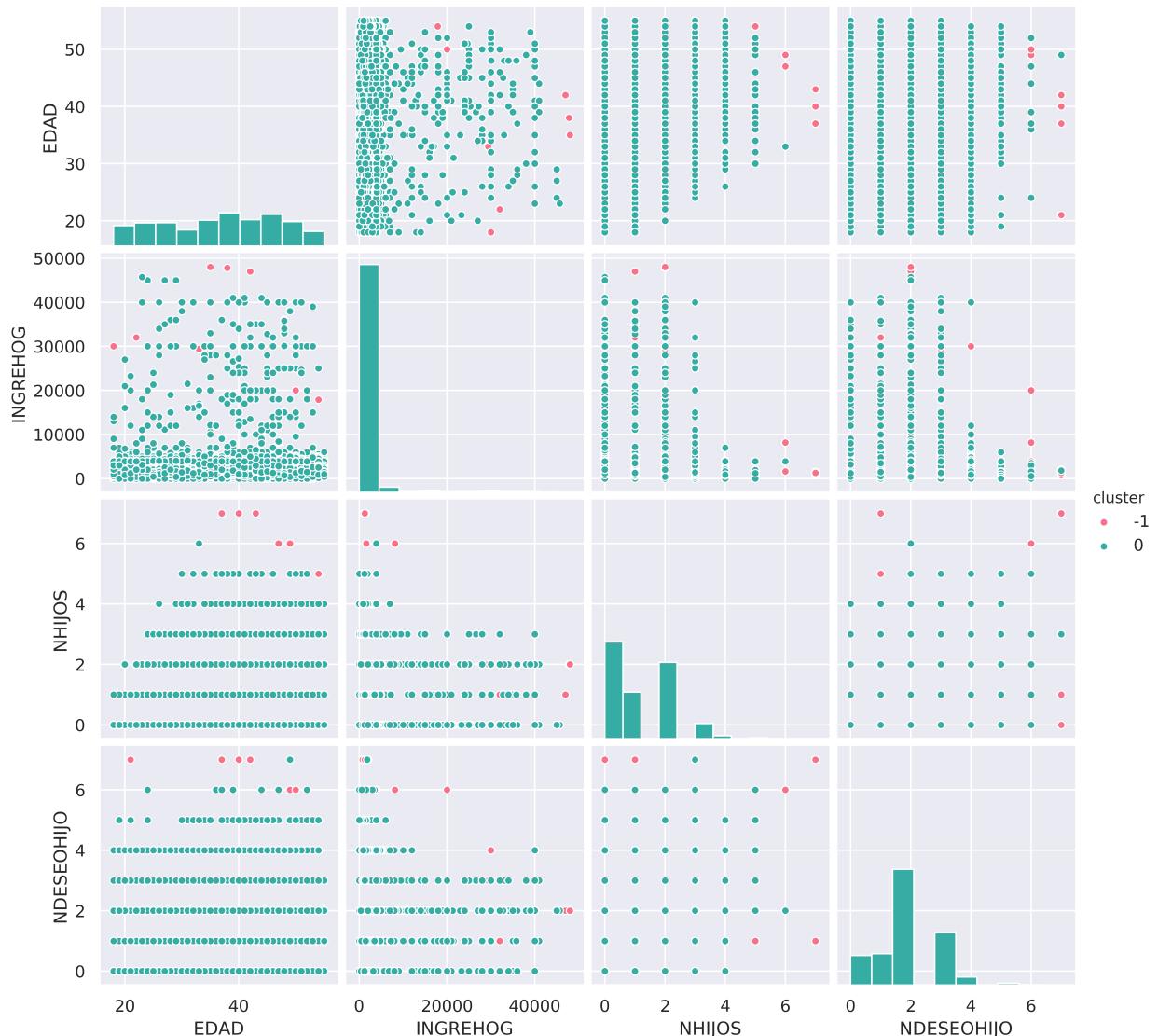


Figura 40: Scatter matrix para DBSCAN

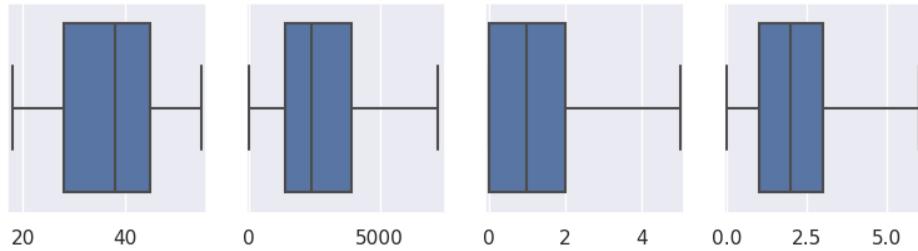


Figura 41: Boxplot para DBSCAN

4.4. Agglomerative Ward

Para el clustering jerárquico ejecuto 100 clusters, generando el siguiente dendograma:

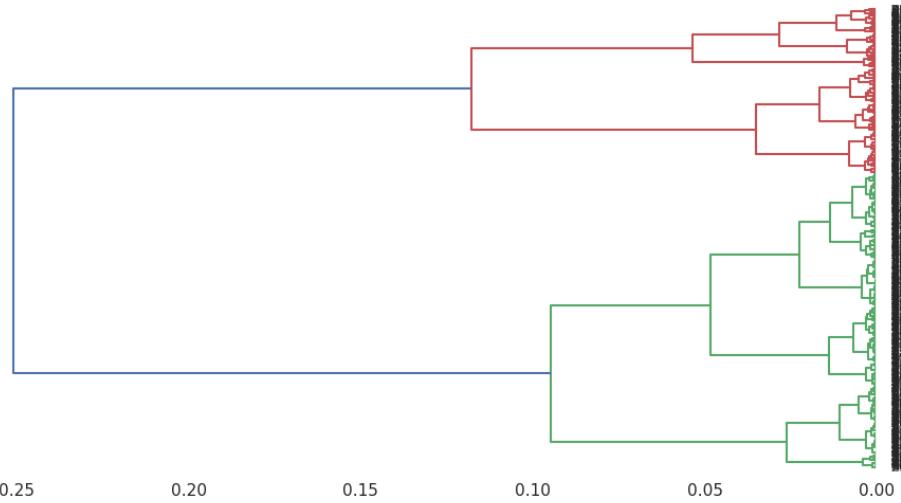


Figura 42: Dendograma

Ejecuto el algoritmo, pues, con 4 clusters, obteniendo así el siguiente dendograma incluyendo un heatmap:

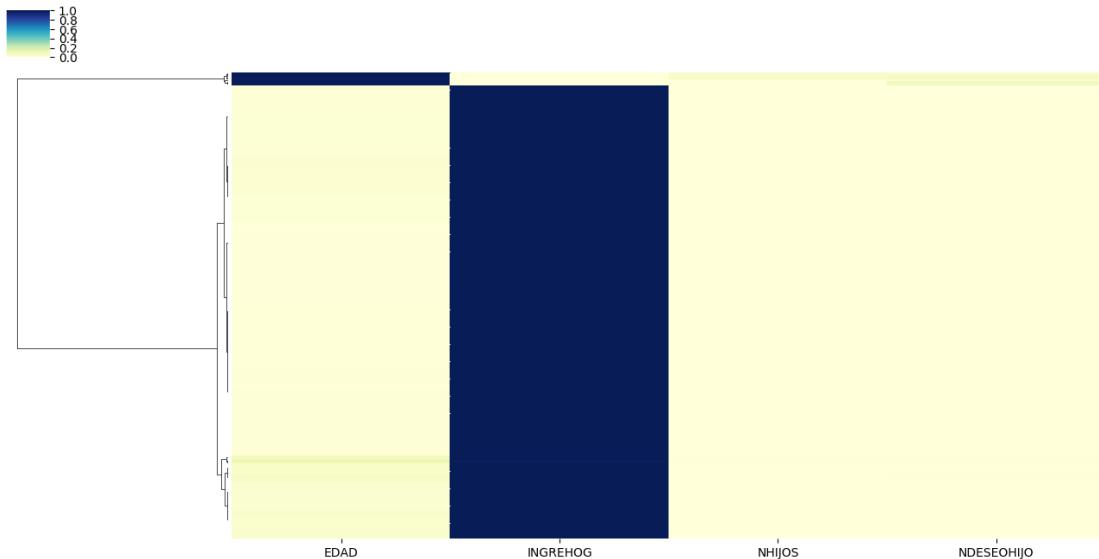


Figura 43: Dendograma con heatmap

4.5. Interpretación de la segmentación

Los algoritmos utilizados no han dado muy buenos resultados, exceptuando K-means y Mean Shift, que han arrojado resultados algo más esclarecedores. Se puede concluir que existen dos grupos más o menos bien diferenciados:

- Grupo de mediana edad y mayores con 1 o 2 hijos la mayoría, que desea los mismos que tiene (2) o alguno más. Dentro de este grupo tienen más hijos los del rango inferior de edad y menos las más mayores. El número de hijos deseado se decremente cuanto mayor es la edad. Los ingresos no revelan nada.
- Jóvenes sin hijos o con 1 hijo, que desearía quedarse con el mismo número de descendientes o tener dos hijos. Dentro de este grupo las más mayores tienen menos hijos por lo general.

Se puede concretar que la edad no influye mucho en los ingresos, y que el número de hijos que tienen y el deseado es menor entre el grupo de las más jóvenes.

5. Contenido adicional

6. Bibliografía

- [1] *Tips for Choosing the Optimal Number of Clusters* (<https://towardsdatascience.com/10-tips-for-choosing-the-optimal-number-of-clusters-277e93d72d92>)
- [2] *Elbow Method Python* (shorturl.at/cntAL)
- [3] *Determine k with Calinski-Harabasz and Silhouette* (<https://stackoverflow.com/questions/41561873/>)
- [4] *scikit-learn: Machine Learning in Python* (<https://scikit-learn.org/>)
- [5] *seaborn: statistical data visualization* (<https://seaborn.pydata.org/>)