

MINERÍA DE DATOS

Clasificación no supervisada.

Rubén Ibáñez Redondo w140170

Grado en Ingeniería Informática – UPM

ÍNDICE

- 1) Introducción
- 2) Descripción del problema y del Dataset
- 3) Metodología, Resultados y Conclusiones

1. Introducción

En este trabajo de minería de datos nuestro principal objetivo será el de intentar encontrar grupos naturales en un conjunto de datos muy grande no etiquetados y sin una clase. En estos grupos se intentará que exista homogeneidad dentro de las clases y heterogeneidad entre las distintas clases, es decir, que las instancias dentro de un mismo grupo sean muy parecidas entre sí, y a su vez, que difiera mucho la semejanza entre las instancias de cada distinto grupo.

Para poder llevar a cabo este proceso, deberemos hacer uso de modelos computacionales, y en esta entrega, al estar hablando de clasificación no supervisada usaremos un algoritmo representativo de cada una de las tres técnicas que hemos visto en clase: jerárquico, particional y probabilista.

2. Descripción del problema y del Dataset

En los últimos años, la obesidad ha dejado de ser un problema exclusivamente estético y ha traspasado la barrera de la salud y la sanidad, al convertirse en una auténtica epidemia a escala mundial que requiere ingentes recursos humanos, técnicos y económicos para combatirla. A pesar del despliegue de toda una artillería preventiva y terapéutica por parte de las autoridades político-médico-científicas, lejos de detenerse, la obesidad se ha multiplicado peligrosamente. Su proliferación en la sociedad ha llegado a tal punto que los especialistas han pasado a denominarla “globesidad”, una especie de globalización del sobrepeso, al margen de su condición de país desarrollado o subdesarrollado.

Recientes estudios epidemiológicos demuestran que un elevado porcentaje de personas presenta algún tipo de patología asociada al sobrepeso, que alcanza cifras que superan los 300 millones en todo el mundo. Algo que hace saltar todavía más las alarmas cuando manejamos datos sobre obesidad infantil y juvenil.

Estos trastornos de carácter nutricional repercuten en una serie de dolencias asociadas al sobrepeso, algunas de ellas crónicas, tales como, la diabetes tipo 2, enfermedades cardíacas, hipertensión arterial e incluso diversos tipos de cáncer.

Con relación a todo esto anteriormente explicado, en este dataset se ha recopilado información de 252 hombres (el número de instancias de nuestro dataset) con relación a diferentes aspectos mayormente relacionados con la obesidad, en este caso, para cada muestra de nuestro conjunto, tendremos 15 distintos atributos que lo definirán (número de variables de nuestro dataset).

A continuación, mostraré el nombre, el tipo y el significado de cada una de las variables para una mejor comprensión de todo el proyecto:

1. Density determined from underwater weighing. Tipo: REAL
Significado: Densidad determinada a partir del pesaje subacuático.
2. Percent body fat. Tipo: REAL
Significado: Porcentaje de grasa corporal.
3. Age (years). Tipo: INTEGER
Significado: Edad de la persona.
4. Weight (lbs). Tipo: REAL
Significado: Peso en libras.
5. Height (inches). Tipo: REAL
Significado: Altura en pulgadas.
6. Neck circumference (cm). Tipo: REAL
Significado: Circunferencia del cuello en centímetros.
7. Chest circumference (cm). Tipo: REAL
Significado: Circunferencia del pecho en centímetros.
8. Abdomen circumference (cm). Tipo: REAL
Significado: Circunferencia del abdomen en centímetros.
9. Hip circumference (cm). Tipo: REAL
Significado: Circunferencia de la cadera en centímetros.
10. Thigh circumference (cm). Tipo: REAL
Significado: Circunferencia del muslo en centímetros.
11. Knee circumference (cm). Tipo: REAL
Significado: Circunferencia de la rodilla en centímetros.
12. Ankle circumference (cm). Tipo: REAL
Significado: Circunferencia del tobillo en centímetros.
13. Biceps (extended) circumference (cm). Tipo: REAL
Significado: Circunferencia del bíceps extendido en centímetros.
14. Forearm circumference (cm). Tipo: REAL
Significado: Circunferencia del antebrazo en centímetros.
15. Wrist circumference (cm). Tipo: REAL
Significado: Circunferencia de la muñeca en centímetros.

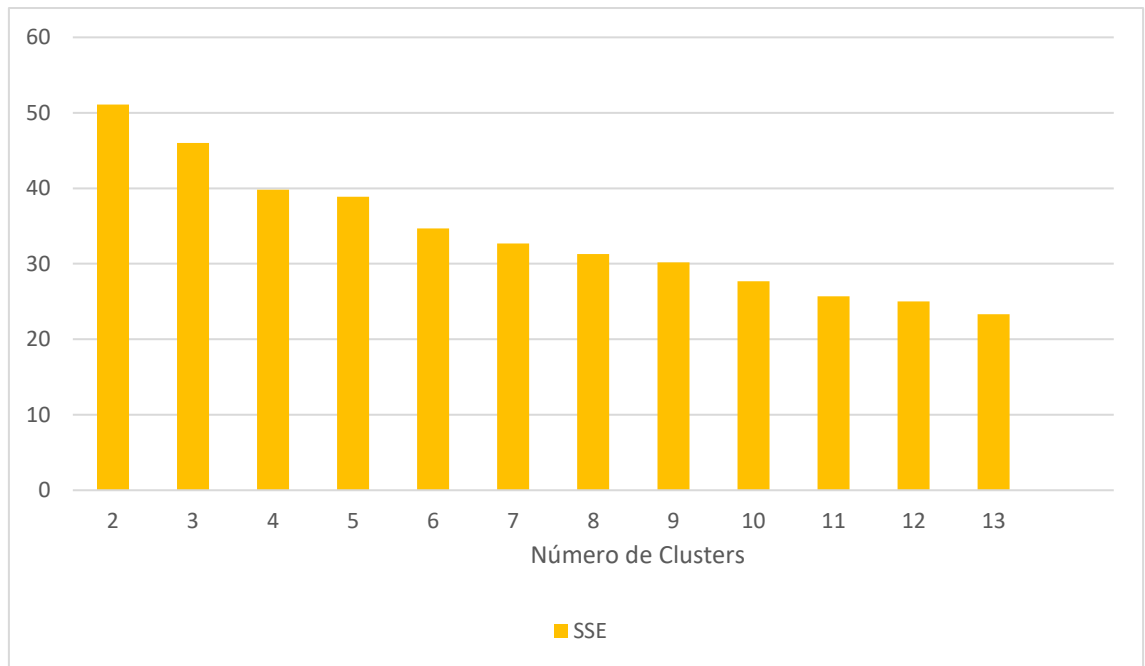
3. Metodología, Resultados y Conclusiones

En primer lugar, para analizar este conjunto de datos utilizaremos un algoritmo representativo de una de las tres técnicas vistas en clase, en este caso, la particional que consiste en formar particiones naturales de los datos en un número de grupos (clusters) posiblemente prefijado. Para esta técnica, usaremos el algoritmo que nos han recomendado que es **SimpleKMeans**. Como modo de agrupamiento utilizaremos 'Use Training Set' y sobre los parámetros del algoritmo usaremos los que vienen por defecto menos el del número de clusters que iremos probando varios hasta decidir cuál es que nos da un SSE (sum of squared errors) más adecuado y óptimo respecto a las demás.

Esto son los resultados obtenidos del valor SSE:

- 2 Clusters: 51.08785708213378
- 3 Clusters: 46.03443469484976
- 4 Clusters: 39.77286179685367
- 5 Clusters: 38.898756364928914
- 6 Clusters: 34.713708295847134
- 7 Clusters: 32.720353458495374
- 8 Clusters: 31.324989206483224
- 9 Clusters: 30.203327167547293
- 10 Clusters: 27.694207311226837
- 11 Clusters: 25.70377428183355
- 12 Clusters: 24.97331760298401
- 13 Clusters: 23.314544088681753

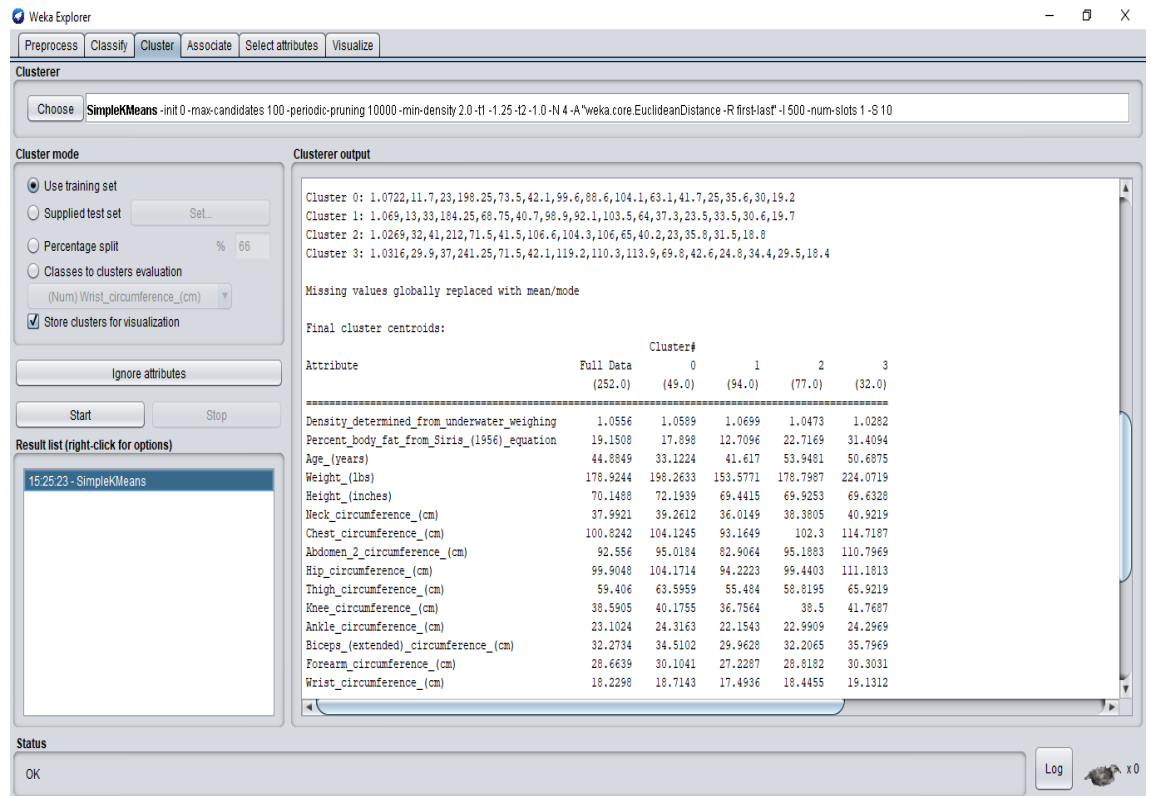
A continuación, se muestra un gráfico con estos valores para intentar ver, como se nos enseñó en la clase práctica, donde se producen los "codos" en este gráfico y decidir qué número de clusters usaremos en este algoritmo para analizar sus resultados.



Analizando los datos de este gráfico, podemos observar tres “codos”, uno de ellos entre los valores de 3 y 4 clusters, otro entre 5 y 6 y el último entre 9 y 10. El más significativo de todos estos es el primero, así que usaremos un número de 4 clusters en nuestro análisis, ya que a partir de ahí el SSE empieza a disminuir muy de poco a poco y aumentar el número de clusters sería poco efectivo.

Aunque un valor como 39.77 de SSE se puede considerar grande, intentaremos sacar cosas en claro de los 4 clusters que se forman, ya que esta es una decisión subjetiva, y en una muestra de 252 instancias, no nos merecería la pena tener un número muy grande de clusters como 20 ya que se empezarían a formar grupos de solamente 2 o 3 instancias y nos haría un análisis más complicado. Además, con este dataset, lo que queremos es hacer una clasificación de distintos cuerpos humanos dependiendo de sus medidas, y no queremos formar un número de grupos mayor de cinco en un principio.

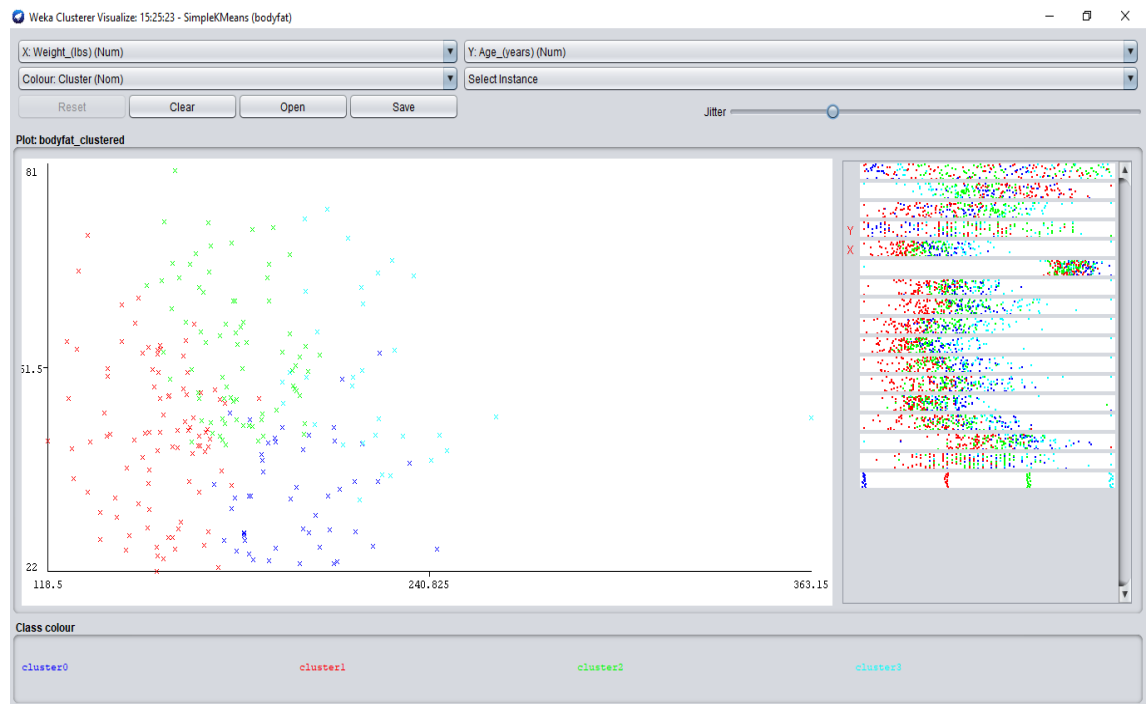
Estos son los resultados que nos desprende el software Weka para este modelo y con los parámetros anteriormente explicados:



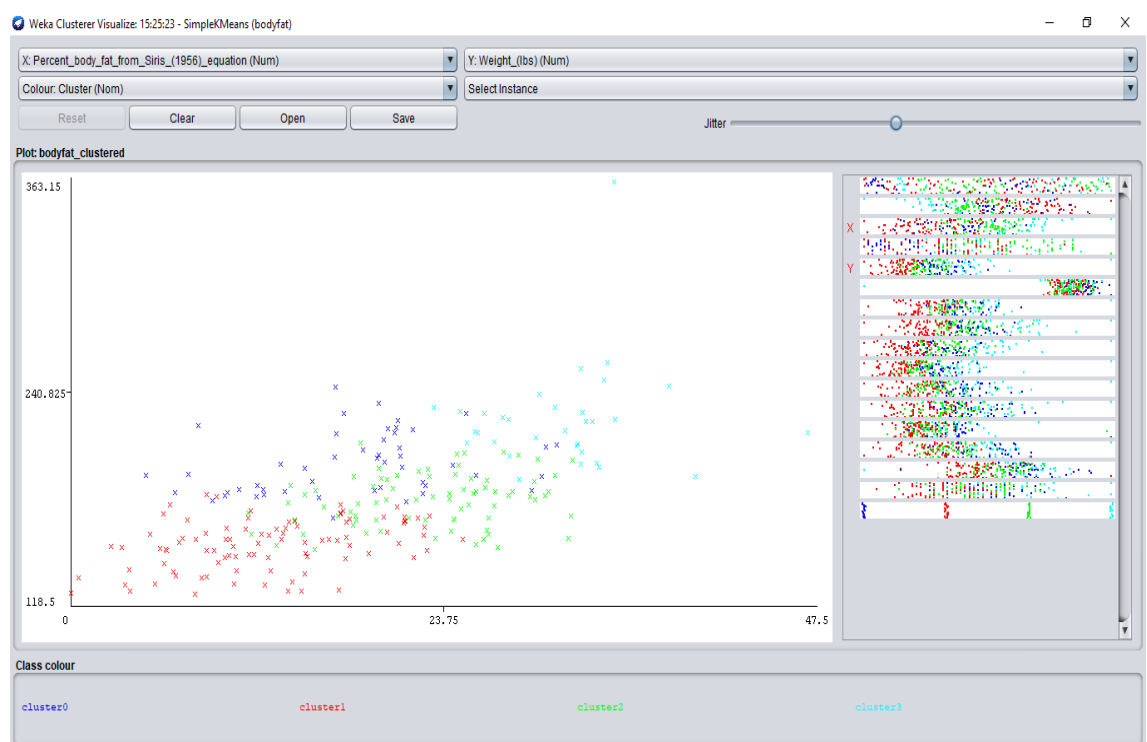
Se realizan 11 iteraciones para llegar a estos resultados. Como vemos, las 252 instancias han sido repartidas en 4 grupos distintos. En el primero de ellos, hay 49 instancias, un 19% del total. El segundo grupo está formado por 94 instancias, un 37% de ellas. El tercero por 77, un 31% del total y el cuarto y último grupo lo forman 32 instancias, el más minoritario, con solo de 13% de la cantidad total de hombres del dataset.

Ahora bien, llega el momento de analizar los centroides de cada cluster para cada atributo, y visualizarlos para intentar comprender qué caracteriza a cada uno de ellos.

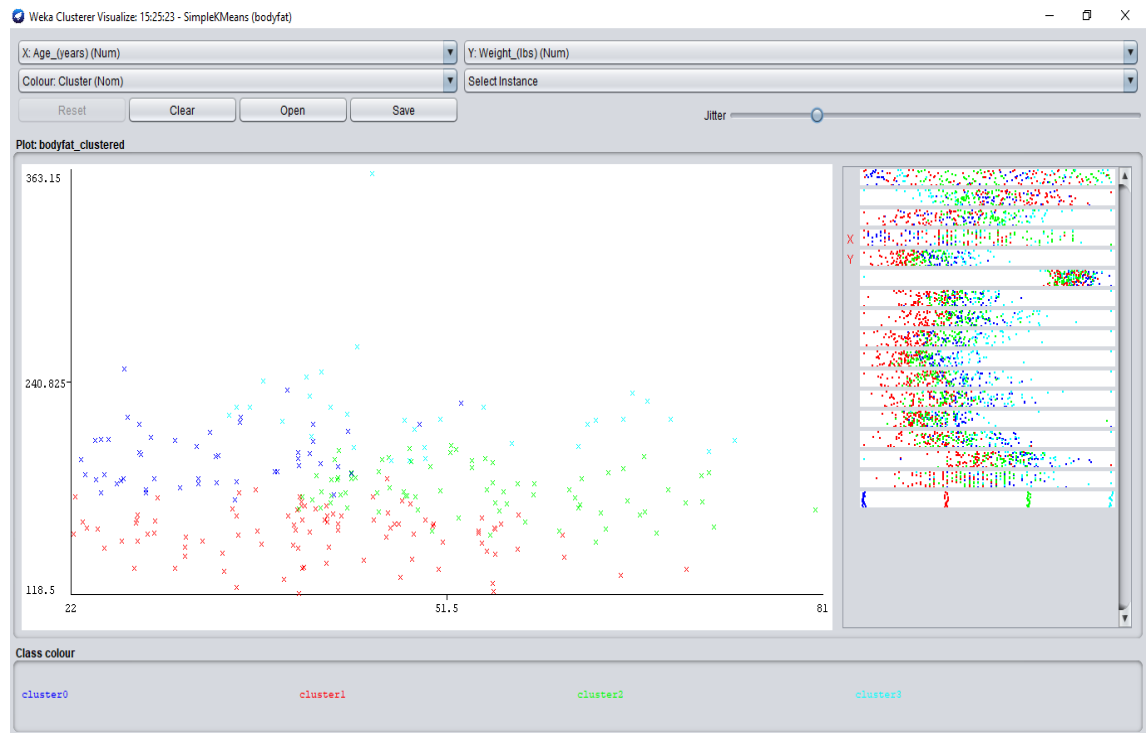
Cluster 0: Revisando y comparando los centroides de este grupo, lo más destacado y lo que más difiere del resto de grupos es que los hombres pertenecientes a él tienen una edad muy inferior al resto de grupos y además una altura superior. También se puede observar que en cuanto a las diez partes del cuerpo que se miden en este dataset, los hombres de este grupo superan a la media en cada una de ellas, por lo que deben tener un cuerpo voluminoso. En este gráfico se puede observar lo anteriormente explicado:



Cluster 1: En este grupo se puede observar claramente las diferencias respecto a los demás, y es que se diferencia por tener el peso de media más bajo al igual que el porcentaje medio de grasa corporal que el resto de los grupos. En cuanto al resto de medidas, todas ellas se encuentran por debajo de la media del dataset, por lo que deben ser cuerpo poco voluminosos y delgados. En el siguiente gráfico se puede apreciar perfectamente.



Cluster 2: En cuánto al tercer grupo que nos encontramos, lo que podemos decir es que es como el grupo más neutro y no destaca en nada especial ni en ninguno atributo en concreto. Todos los centroides son muy parecidos a los de la media del dataset y no tiene valores extremos. Lo único quizás destacable es que la media de edad es la mayor de los cuatro grupos, pero tampoco muy destacada. En este grupo, los cuerpos deben ser de una talla media. En la siguiente imagen se puede ver como están a mitad del gráfico la mayoría, pero tirando un poco hacia edades altas.



Cluster 3: En este último grupo queda muy claro desde que empiezas a analizar los datos en lo que difiere al resto. Los hombres que pertenecen a este grupo son los que más porcentaje de grasa corporal, menor densidad determinada a partir del pesaje subacuático y más peso tienen de todos los grupos. Además, en cuánto los diez atributos de medidas de parte del cuerpo del dataset, en todos ellos son lo valores más altos de todos los grupos. Se puede observar perfectamente en el siguiente gráfico.



Finalmente, después de todo este análisis a este algoritmo, podríamos poner unas etiquetas o características en común que distinguen a cada uno de los cuatro grupos que se han formado:

Cluster 0: Hombres jóvenes, altos y con cuerpos voluminosos (algo de obesidad).

Cluster 1: Hombres de mediana edad con cuerpos poco voluminosos y delgados.

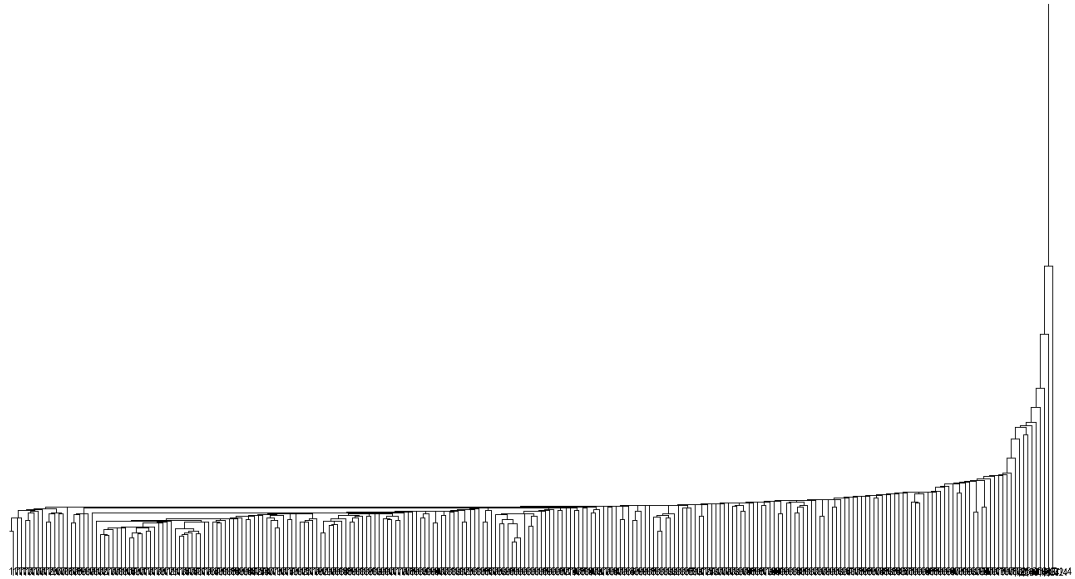
Cluster 2: Hombres algo mayores de edad con cuerpo de talla media.

Cluster 3: Hombres con los cuerpos más voluminosos y con un muy alto porcentaje de obesidad.

Ahora bien, como el profesor nos explicó en la clase práctica para este proyecto, con este algoritmo y utilizando el software WEKA, no queda muy claro cómo identificar instancias en el Dendrograma o árbol jerárquico, y, además, si se pone como parámetro un número diferente a uno como número de clusters, los efectos tampoco están claros, por lo que en el parámetro para definir el número de clusters, lo pondremos a uno.

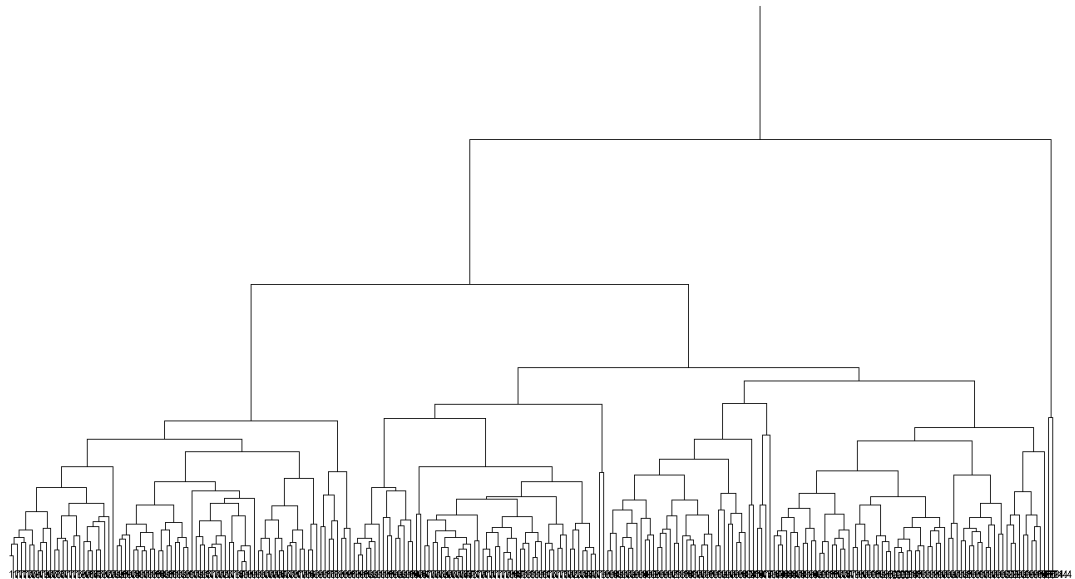
[illegible]

Como hemos dicho anteriormente, no se puede sacar mucha información de esta salida, por lo que nuestro objetivo con este algoritmo será el de analizar sus dendrogramas. A continuación, el correspondiente a los parámetros mencionados anteriormente:



Con este dendrogramas no se puede sacar casi nada de información en claro, puesto que no es posible identificar las instancias. Aún así, podemos decir que con este método (SINGLE) se aprecia muchas diferencias y uniones entre instancias al principio, y no se crean unan cantidad pequeña de clusters con un número grande de instancias cada uno, si no que se van uniendo finalmente instancia a instancia predominando un único cluster.

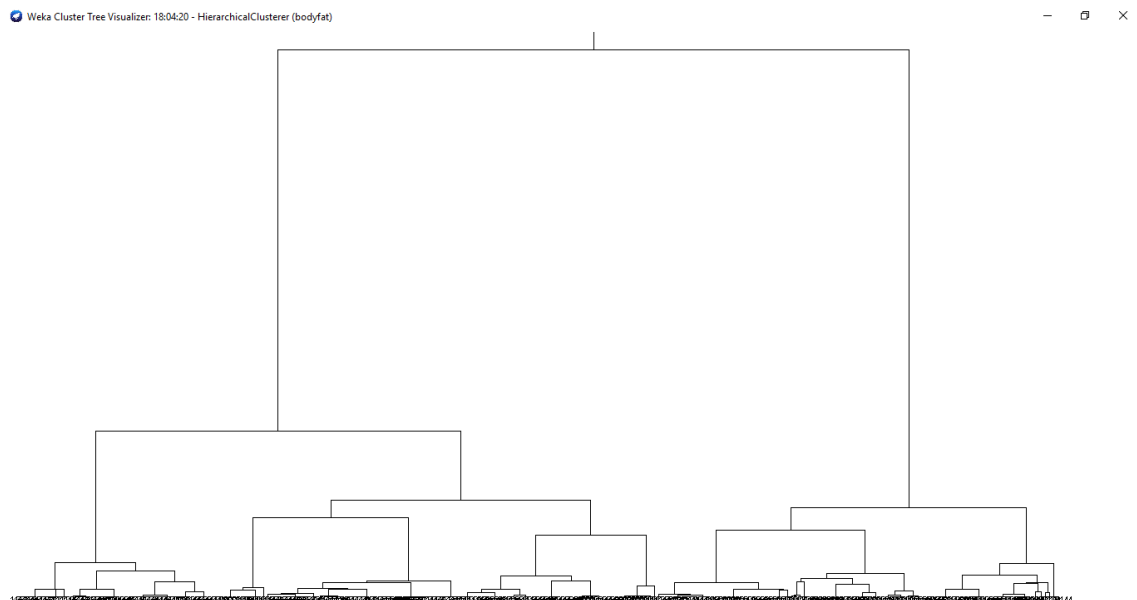
Por lo tanto, decidimos probar con otros métodos de unión distintos como el de enlace completo o vecino más lejano (COMPLETE) que consiste en que la distancia entre dos clusters es la máxima distancia entre todos los posibles pares de objetos en ambos clusters.



En este dendrogramas ya se puede apreciar más variedad que en el primero, puesto que aquí sí habrá lugares o momentos en los que podamos cortar y así

dividir nuestro conjunto de datos en un número de clusters determinados. El momento de corte más interesante podría resultar a mitad de altura del árbol donde se puede hacer un corte en cuatro clusters.

A posteriori, aplicamos a nuestro conjunto de datos de nuevo este algoritmo pero probando nuevo métodos de unión como sería el de enlace medio (AVERAGE) que consiste en que la distancia entre dos clusters es la media de las distancias entre todos los posibles pares de objetos en ambos clusters y también el método del centroide (CENTROID) que consiste en reemplazar cada cluster por su centroide (unitario de nuevo) y calcular la distancia entre los dos centroides pero con ambos métodos nos sale un resultado parecido al del enlace simple. Finalmente, también probamos con el método de WARD, que no calcula las distancias, si no que calcula la suma total de desviaciones (al cuadrado) de la media de un cluster y trata de minimizarla. Y nos imprime este dendrograma como resultado:

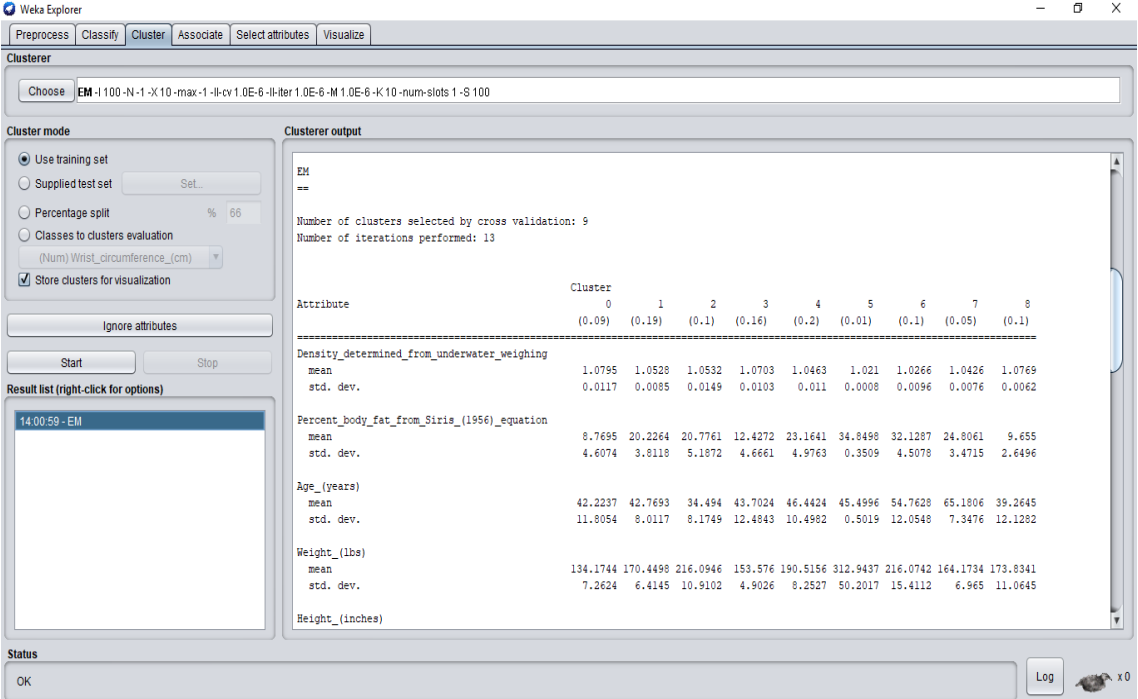


Este árbol jerárquico es algo diferente al resto, y podemos establecer también un punto de corte en el que poder dividir al conjunto de datos en un número determinado de clusters, en este caso, los mejores momentos de corte podrían ser a mitad de camino cuando se podría dividir en 5 clusters, o un poco más adelante donde se podría fraccionar en 3 o incluso 2 clusters.

Finalmente, como conclusión después de todo el análisis de este algoritmo, se podría afirmar que es el que menos información nos puede aportar de los tres, debido en gran parte a la poca claridad que nos aporta la propia salida de WEKA, pero aún así, dependiendo del método para enlazar clusters que se use, se puede observar una información u otra, y en algunos casos establecer un número preciso de número de clusters en los que poder dividir este conjunto de datos.

Para terminar, vamos a utilizar la tercera y última de las técnicas de clasificación no supervisada vistas en clase, que es la probabilística. Esta técnica suele asumir que las densidades condicionales de los clusters tienen cierta forma paramétrica conocida. Los datos como hipótesis provienen de una mixtura de k distribuciones de probabilidad, una para cada cluster y cada distribución da la probabilidad de que un objeto tenga ciertos valores en las variables si se supiera que pertenece a ese cluster porque realmente el objeto pertenece a un cluster, pero no sabemos a cuál, por lo que nuestro objetivo será el de encontrar el conjunto de clusters más probable dados los datos. Para esta técnica, como algoritmo representativo usaremos el **Expectation-maximization (EM)**, este algoritmo está compuesto por dos pasos, en el primero de ellos (E) se evalúan las responsabilidades usando los parámetros actuales y en el segundo (M) se reestiman los parámetros, pero ahora usando las responsabilidades actuales, es decir, este algoritmo nos indicará dada una muestra de datos la probabilidad de cada instancia de pertenecer a un determinado cluster. Como modo de agrupamiento utilizaremos 'Use Training Set' y sobre los parámetros del algoritmo usaremos los que vienen por defecto, dejando el número de cluster con un valor de -1 para que el algoritmo seleccione el número real de cluster mediante validación cruzada.

Estos son los resultados y la salida que nos proporciona WEKA al ejecutar este algoritmo con estos parámetros y nuestro conjunto de datos:



Clusterer

Choose **EM** -I 100 -N 1 -X 10 -max-1 -ll-cv 1.0E-6 -ll-iter 1.0E-6 -M 1.0E-6 -K 10 -num-slots 1 -S 100

Cluster mode

- ☒ Use training set
- ☐ Supplied test set
- ☐ Percentage split
- ☐ Classes to clusters evaluation
- ☒ Store clusters for visualization

Clusterer output

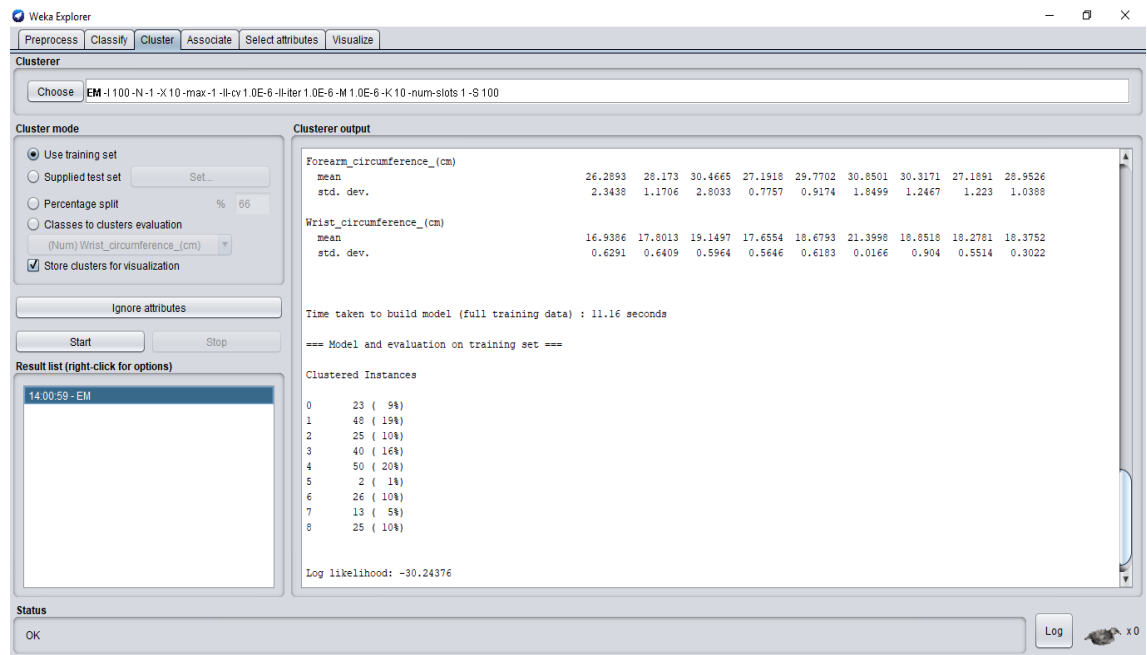
EM
==

Number of clusters selected by cross validation: 9
Number of iterations performed: 13

Attribute	Cluster 0 (0.09)	Cluster 1 (0.19)	Cluster 2 (0.1)	Cluster 3 (0.16)	Cluster 4 (0.2)	Cluster 5 (0.01)	Cluster 6 (0.1)	Cluster 7 (0.09)	Cluster 8 (0.1)
Density_determined_from_underwater_weighing									
mean	1.0795	1.0528	1.0532	1.0703	1.0463	1.021	1.0266	1.0426	1.0769
std. dev.	0.0117	0.0085	0.0149	0.0103	0.011	0.0008	0.0096	0.0076	0.0062
Percent_body_fat_from_Siris_(1956)_equation									
mean	8.7695	20.2264	20.7761	12.4272	23.1641	34.8498	32.1287	24.8061	9.655
std. dev.	4.6074	3.8118	5.1872	4.6661	4.9763	0.3509	4.5078	3.4715	2.6496
Age_(years)									
mean	42.2237	42.7693	34.494	43.7024	46.4424	45.4996	54.7628	65.1806	39.2645
std. dev.	11.8054	8.0117	8.1749	12.4843	10.4982	0.5019	12.0548	7.3476	12.1282
Weight_(lbs)									
mean	134.1744	170.4498	216.0946	153.576	190.5156	312.9437	216.0742	164.1734	173.8341
std. dev.	7.2624	6.4145	10.9102	4.9026	8.2527	50.2017	15.4112	6.965	11.0645
Height_(inches)									

Status

OK Log



Como podemos observar el número de clusters seleccionados automáticamente por el algoritmo mediante validación cruzada es de 9, el algoritmo ha llegado a esta conclusión tras realizar 13 iteraciones. Respecto a la medida de calidad de este algoritmo, WEKA la mide mediante el log-likelihood, que cuanto mayor sea, mayor verosimilitud tendrá el modelo, pero la cosa es que si se aumenta el número de clusters, obviamente aumentar el log-likelihood, pero esto quizás puede producir overfit, por lo que estableciendo el número de clusters a -1 como parámetros antes de correr el modelo, se deja que él lo seleccione automáticamente. Para este conjunto de datos y para un número de 9 clusters, nuestro log-likelihood es de -30.24376

Esta es la clasificación de instancias que ha hecho nuestro modelo en los nueve diferentes clusters:

- 0 23 (9%)
- 1 48 (19%)
- 2 25 (10%)
- 3 40 (16%)
- 4 50 (20%)
- 5 2 (1%)
- 6 26 (10%)
- 7 13 (5%)
- 8 25 (10%)

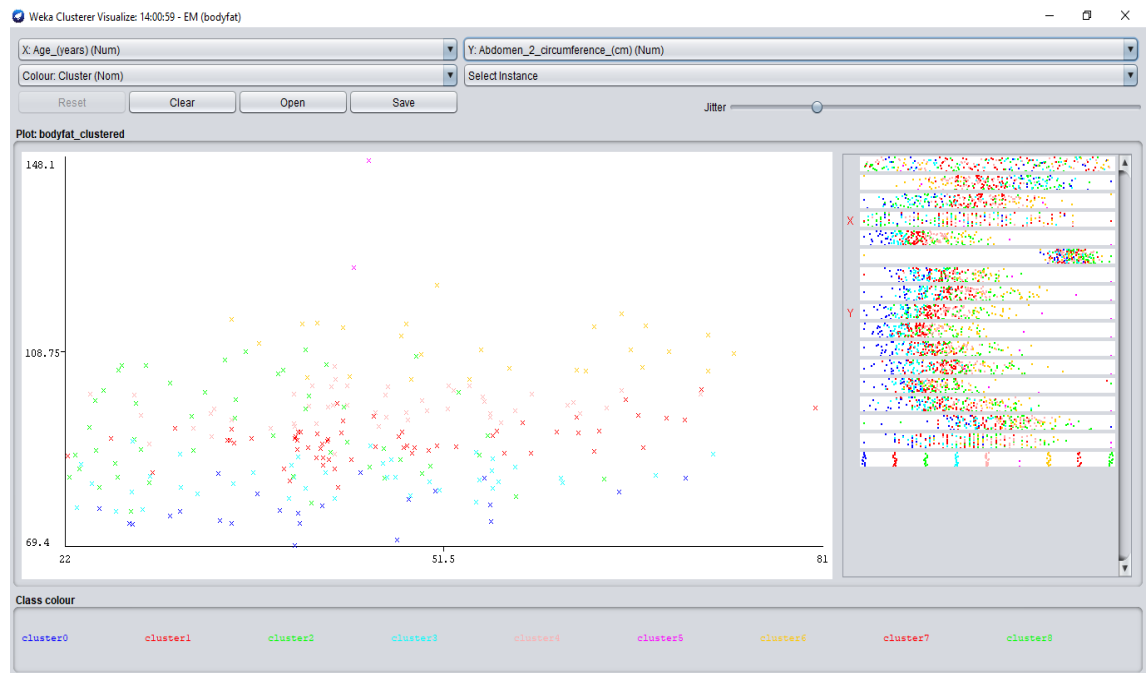
Esto mide las probabilidades de cada una de las 252 instancias de pertenecer a cada uno de los grupos, es decir, que según este modelo y esta división, cada instancia tiene una probabilidad del 9% de pertenecer al Cluster 0, un 19% de pertenecer al Cluster 1, un 10% de pertenecer al Cluster 2, un 16% de pertenecer al Cluster 3, un 20% de pertenecer al Cluster 4, un 1% de pertenecer al Cluster 5, un 10% de pertenecer al Cluster 6, un 5% de pertenecer al Cluster 7 y un 10% de pertenecer al Cluster 8.

Por lo que se puede ver según estos datos, tenemos 3 grupos mayoritarios a los que las instancias tendrán muchas más probabilidades de pertenecer que al resto que son los clusters 1, 3 y 4, después tenemos un grupo medio de 4 grupos que tienen un porcentaje intermedio que son los clusters 0, 2, 6 y 8, y finalmente, tenemos dos grupos situados a la cola que tiene un porcentaje muy bajo y las instancias tendrán muy pocas probabilidades de pertenecer a ellos que son el cluster 5 y el 7.

Ahora llega el turno de fijarnos en los valores de las medias de los atributos para cada posible grupo para intentar analizar las diferencias entre ellos y poder visualizarlas mediante los gráficos también. Este es un análisis más complicado que con el primer algoritmo que hemos utilizado en esta práctica puesto que hay más de el doble de grupos y cada uno con sus valores, por lo que resulta más complejo sacar unas posibles etiquetas como nombres de cada grupo, pero podemos observar que entre el cluster 1 y 2 aunque tengan un porcentaje de grasa corporal y una densidad determinada por el pesaje subacuático casi idénticas, después difieren mucho respecto al peso como se aprecia en la siguiente foto, ya que las instancias rojas (cluster 1) y las verdes (cluster 2) están repartidas equilibradamente en el eje de la densidad, pero después, en el eje del peso todas las verdes se encuentran más hacia la derecha ya que son instancias con una media de peso muy superior.



O que para los clusters 0 y 1 por ejemplo la media de la edad de sus instancias sea prácticamente idéntica, y luego difieran mucho en los atributos de circunferencia del abdomen en centímetros, el peso o el porcentaje de grasa corporal. En el siguiente gráfico se puede ver como las muestras de estos dos clusters (azules y rojas) se reparten a lo largo del eje de la edad de una manera parecida puesto que tienen la misma media, pero después, en el eje de la circunferencia del abdomen en centímetros, todas las instancias rojas están por encima de las azules:



Se pueden seguir viendo más diferencias como que los Clusters 2 y 6 tienen una media de peso casi igual, sin embargo, sus componentes difieren en edad una media de 20 años, o que por ejemplo las instancias de los clusters 1 y 7 tienen una media de la medida de la cadera en centímetros idéntica, y el peso también, pero después en cuanto al porcentaje de grasa corporal se diferencian de media un 10%.

Se pueden sacar muchas más conclusiones así, pero lo importante de este algoritmo basado en probabilidades es que considera que el número óptimo de clusters para dividir este conjunto de datos es de nueve, a diferencia del algoritmo usado para la técnica particional más orientado a las distancias entre las instancias que, aunque es más subjetivo y puede depender de unas necesidades prefijadas, mostraba como un número más óptimo, la división del dataset en cuatro o seis subconjuntos.