Curso 2018-2019

APRENDIZAJE AUTOMÁTICO: ALGORITMO K-MEDIAS

mrodrigue212@alumno.uned.es | Manuel Rodríguez Sánchez

Contenido

Actividad 2

Mostrar en una tabla las coordenadas de los centroides obtenidos usando todos los ejemplos de la base de datos frente a los obtenidos solo utilizando los ejemplos de entrenamiento. Analizar y comentar los resultados obtenidos en cada caso.

Centroides aleatorios para todos los ejemplos				
	sepallength	sepalwidth	petallength	petalwidth
Centroide 1	6,1	2,9	4,7	1,4
Centroide 2	6,2	2,9	4,3	1,3
Centroide 3	6,9	3,1	5,1	2,3

Tabla 1 - Coordenadas de los centroides obtenidos, usando todos los ejemplos de la base de datos.

Centroides aleatorios escogidos del 66%				
	sepallength	sepalwidth	petallength	petalwidth
Centroide 1	4,8	3	1,4	0,3
Centroide 2	6,7	3,1	4,7	1,5
Centroide 3	5,6	3	4,1	1,3

Tabla 2 - Coordenadas de los centroides obtenidos, usando solo el 66% de los ejemplos de la base de datos.

Tanto en la Tabla 1 como en la Tabla 2 se observan las semillas o centroides que han sido seleccionados de forma aleatoria de la base de datos, y que a partir de los cuales se establecerán los grupos ("cluster" en su término anglosajón).

Análisis y comentarios de los resultados obtenidos para la Tabla 1.

En el algoritmo K-medias se establecen K grupos para la clasificación de ejemplos. Es por ello que para los 150 ejemplos de entrenamiento de la base de datos, se han seleccionado tres semillas (K=3):

S_1 = (6'1, 2'9, 4'7, 1'4)

S_2 = (6'2, 2'9, 4'3, 1'3)

S_3 = (6'9, 3'1, 5'1, 2'3)

Ya tenemos inicializados los grupos y semillas para iniciar el proceso de clasificación (Paso 0).

El siguiente paso (Paso 1) es la asignación de cada uno de los 150 ejemplos al grupo más próximo (por centroides iniciales). Aquí el algoritmo calculará la distancia y posteriormente la similitud para clasificar en cada uno de los tres grupos los ejemplos:

$$Distancia(x_i, x_j) = \sqrt{\sum (x_{ik} - x_{jk})^2}$$

$$Similitud(x_i, x_j) = \frac{1}{1 + distancia(x_i, x_j)}$$

Una vez clasificados los ejemplos en base a los centroides iniciales, el algoritmo recalcula los centroides (Paso 2); se usa la siguiente ecuación:

$$Centroide = \left[\frac{\sum_{i=1}^{m} v_{i,1}}{m}, \frac{\sum_{i=1}^{m} v_{i,2}}{m}, \ldots \frac{\sum_{i=1}^{m} v_{i,j}}{m}\right]$$

Una vez recalculados los centroides, se vuelve a realizar los cálculos de distancia y similitud para reagrupar los 150 ejemplos en base a éstos nuevos centroides (Paso 3). Una vez finalizado el proceso (que es lo mismo que se hace en el paso 1), se ha de comprobar si este nuevo agrupamiento coincide con el del paso 1, si así fuera, el proceso finaliza, y si no, hemos de repetir (nueva iteración) el paso 2 y el 3. Y aquí es donde quería llegar por que en el ejercicio que nos ocupa, vemos que este proceso se realiza hasta seis veces. Lo podemos ver en la línea donde pone "Number of iterations".

La agrupación final nos la da el cuadro siguiente:

```
Final cluster centroids:
                            Cluster#
Attribute      Full Data    Cluster 0  Cluster 1  Cluster 2
                 (150.0)     (61.0)     (50.0)     (39.0)
============================================================
sepallength       5.8433     5.8885      5.006     6.8462
sepalwidth         3.054     2.7377      3.418     3.0821
petallength       3.7587     4.3967      1.464     5.7026
petalwidth        1.1987      1.418      0.244     2.0795
```

Es decir, después de 6 iteraciones (seis repeticiones del paso 2 y 3), se han clasificado los 150 ejemplos repartidos en:

- 61 en la primera agrupación (Cluster 0).
- 50 en la segunda agrupación (Cluster 1).
- 39 en la tercera agrupación (Cluster 2).

Análisis y comentarios de los resultados obtenidos para la Tabla 2.

Ahora vamos con la segunda parte de los resultados del experimento, donde aquí trabajamos con los 2/3 de los ejemplos de la base de datos, es decir cogemos 99 registros, y en base a estos se extraen aleatoriamente tres semillas o centroides (los de la tabla 2) y sobre estos, hacemos las diferentes iteraciones para la agrupación de los 99 ejemplos.

Aquí las iteraciones son solo tres, y los resultados de las agrupaciones son:

```
Final cluster centroids:
                            Cluster#
Attribute      Full Data    Cluster 0  Cluster 1  Cluster 2
                 (99.0)     (35.0)     (36.0)     (28.0)
============================================================
sepallength       5.8313     5.0514      6.725     5.6571
sepalwidth        3.0586     3.4543     3.0139     2.6214
petallength       3.6848     1.4771     5.4389     4.1893
petalwidth        1.1657     0.2571     1.9139     1.3393
```

De los 99 ejemplos y después de 3 iteraciones (tres repeticiones del paso 2 y 3), se han clasificado en:

- 35 en la primera agrupación (Cluster 0).
- 36 en la segunda agrupación (Cluster 1).
- 28 en la tercera agrupación (Cluster 2).

Como nos quedaban 51 ejemplos, que no se han usado para el entrenamiento, la clasificación de éstos una vez entrenado el algoritmo (con los 99 ejemplos), queda de la siguiente forma:

En Cluster 0 se clasifican 15 ejemplos (29%)

En Cluster 1 se clasifican 20 ejemplos (39%)

En Cluster 2 se clasifican 16 ejemplos (31%)

Como podemos ver, la diferencia de la tabla 1 a la 2 corresponde al número de registros que se han usado para la búsqueda aleatoria de los centroides (150 para la tabla 1, y 99 para la tabla 2). Esto también ha influido en el número de iteraciones para la búsqueda de nuevos centroides, a partir de los centroides iniciales.

Actividad 3

Representar cuatro gráficas usando distintos pares de atributos (eje X – eje Y) y la variable "cluster" como color. Representar también la variable "cluster" (eje Y) frente a la variable "Instance_number" (eje X). Analizar e interpretar los datos obtenidos en cada una de las gráficas.

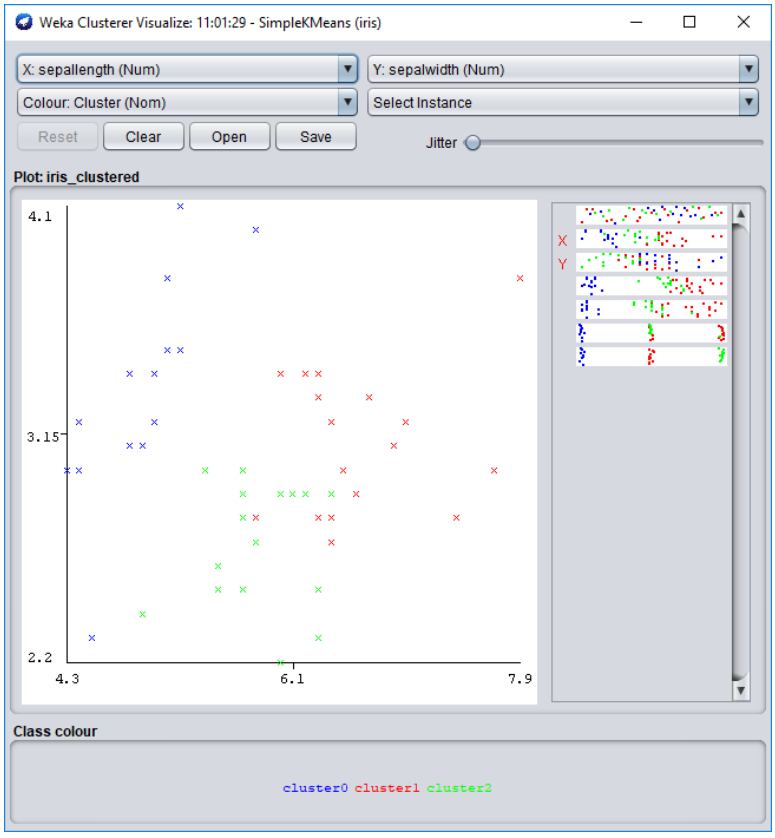


Ilustración 1 - Representación gráfica sepallength (eje X)-sepalwidth (eje Y)

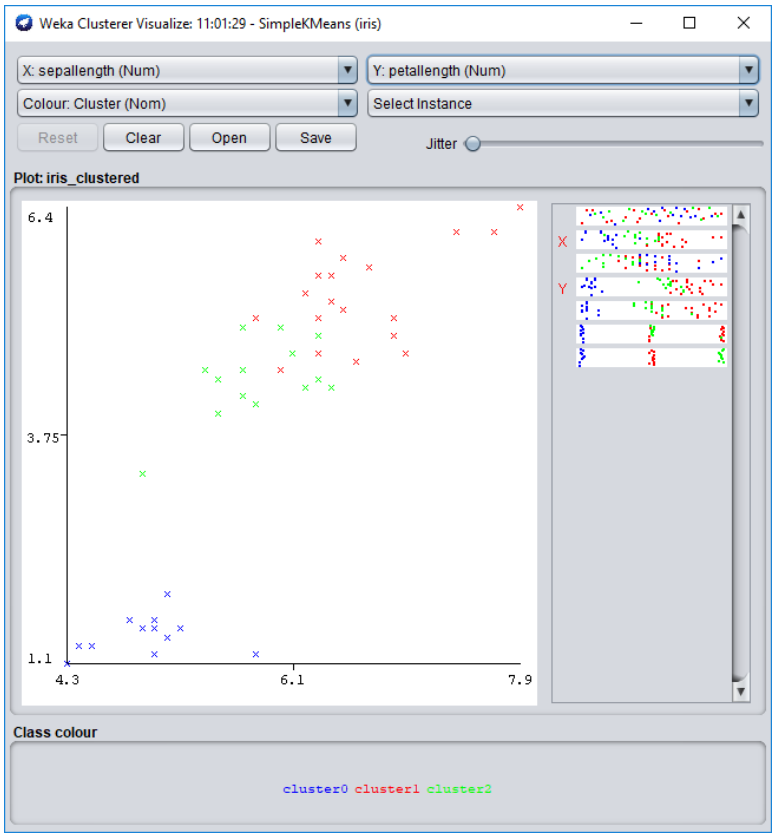


Ilustración 2 - Representación gráfica sepallength (eje X)-petallength (eje Y)

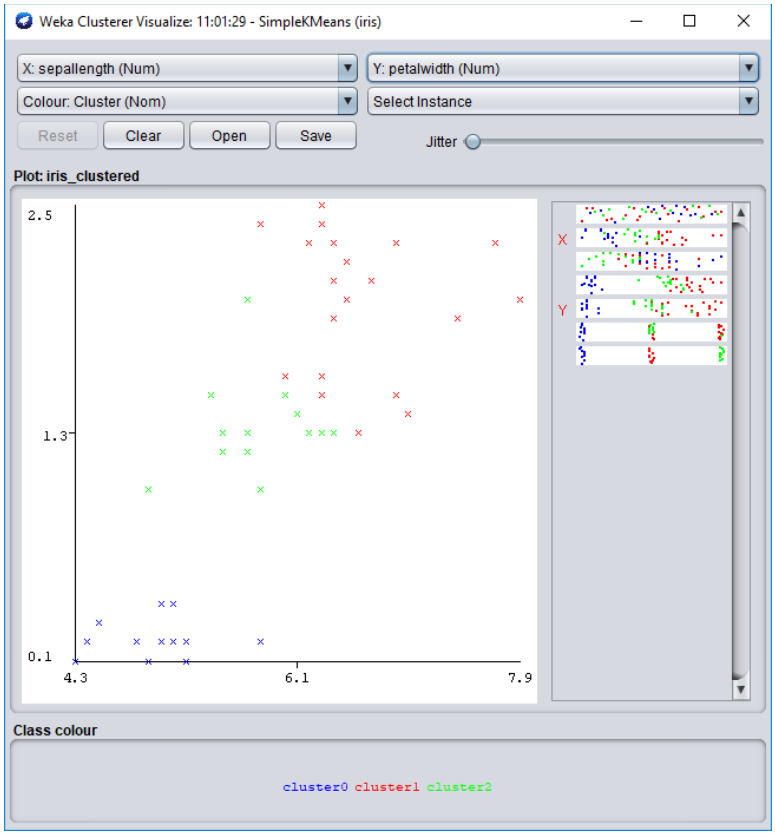


Ilustración 3 - Representación gráfica sepallength (eje X)-petalwidth (eje Y)

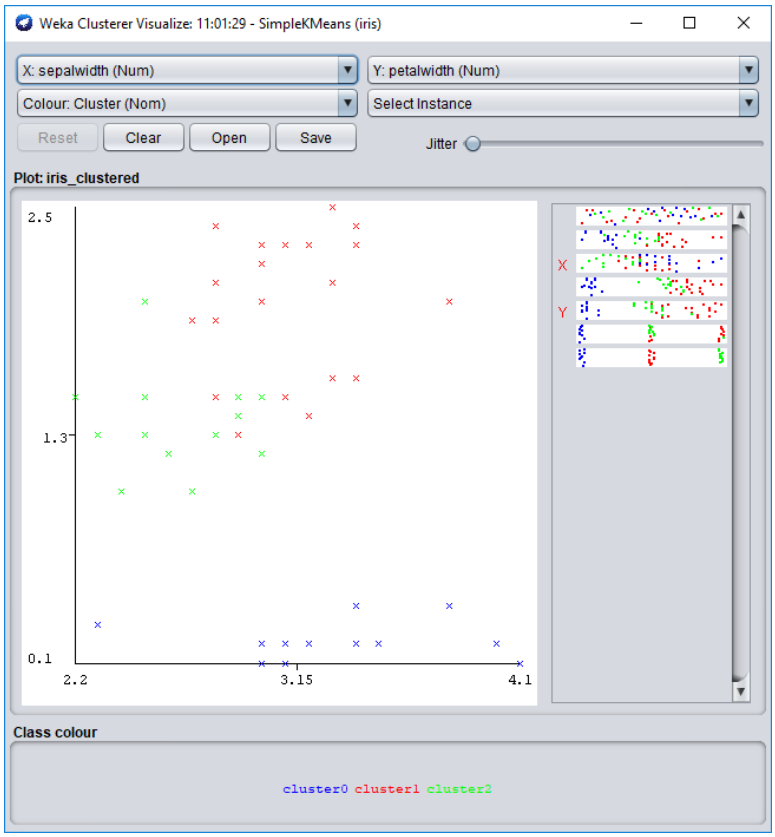


Ilustración 4 - Representación gráfica sepalwidth (eje X)-petalwidth (eje Y)

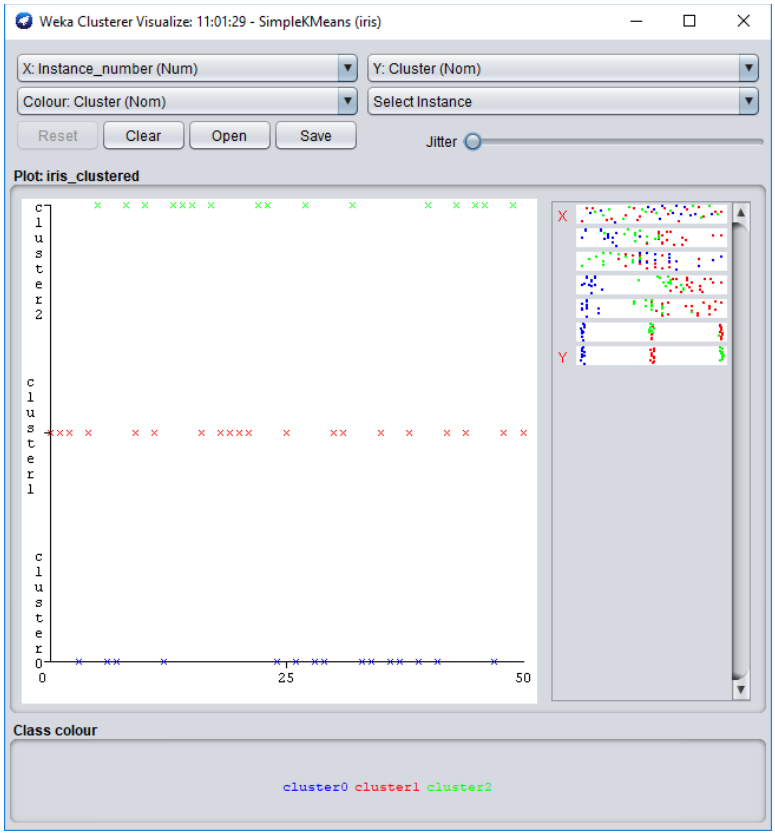


Ilustración 5 - Representación gráfica de la variable Cluster (eje Y) frente a la variable Instance_number (eje X)

Análisis e interpretación de los resultados obtenidos de cada una de las ilustraciones anteriores

He de destacar que desde la ilustración 1 a la ilustración 4, se observa que los grupos se identifican claramente, esto es que aunque hayamos ignorado la clase, al final las clasificaciones prácticamente coinciden con el valor de la clase (tipo de flor) de cada registro.

Por ejemplo, si cogemos varios puntos de cualquier gráfico de dispersión, vemos que casi siempre va a coincidir en el "cluster" y en clase (tipo de flor).

De esto podemos deducir, que el algoritmo K-medias funciona muy bien en la búsqueda de patrones, que permiten identificar a que clase o "cluster" pertenece cada ejemplo, sin utilizar la clase.

En la ilustración 5, observamos el número de ejemplos (de los 51 que se han usado, recordemos que los 99 restantes han sido para el entrenamiento del algoritmo) que se asignan a cada uno de los "cluster" o grupos, 15 para el "cluster" 0, 20 para el "cluster" 1 y 16 para el "cluster" 2.

Actividad 4

Realizar un análisis de la gráfica generada, comentando las bondades de las agrupaciones obtenidas. Mostrar la matriz de confusión, y analizar los resultados obtenidos.

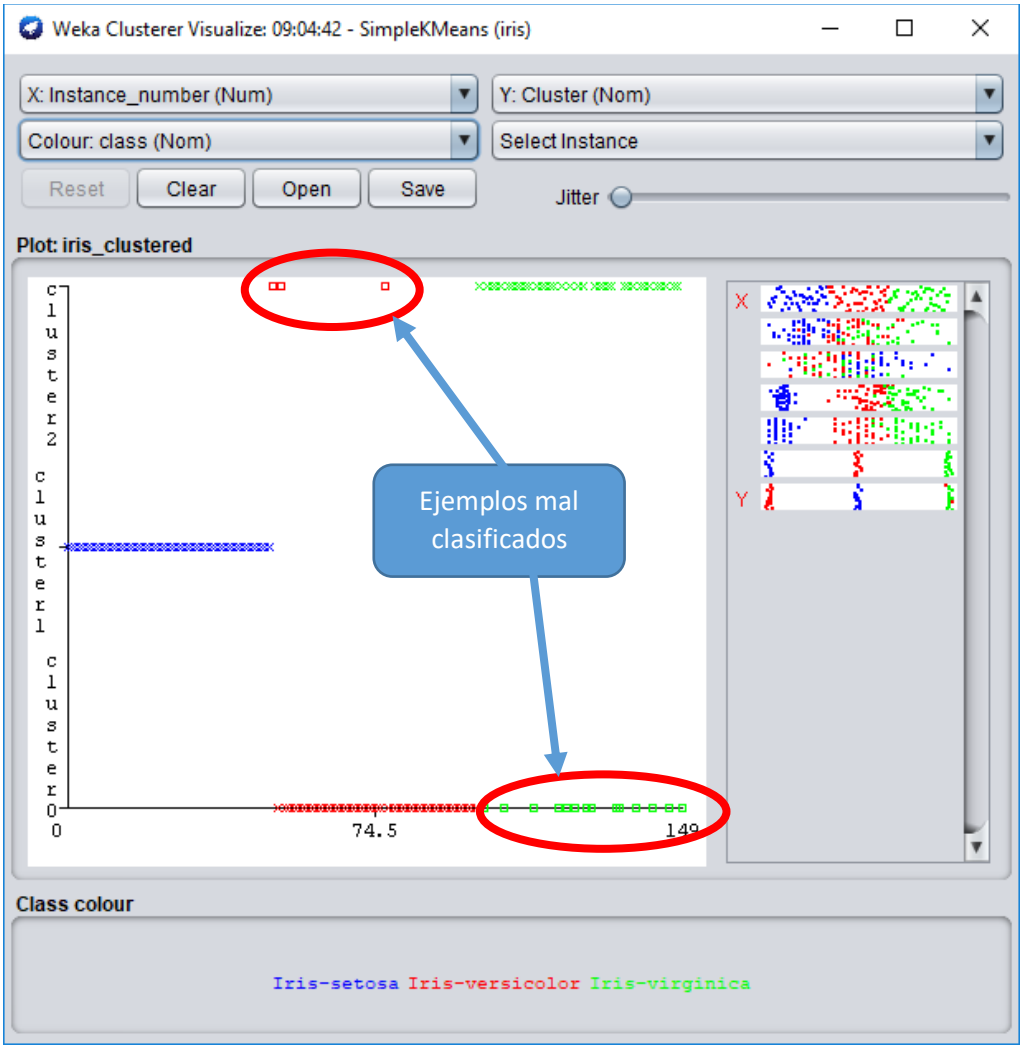


Ilustración 6 - Clasificación de ejemplos y sus errores

Análisis e interpretación de los resultados obtenidos

Se han agrupado 150 ejemplos:

Cluster 0 61 (41%)

Cluster 1 50 (33%)

Cluster 2 39 (26%)

Atributo de clase: class

Matriz de confusión:

Cluster asigando			
Cluster 0	Cluster 1	Cluster 2	Clase
0	50	0	Iris-Setosa
47	0	**3**	Iris-Versicolor
14	0	36	Iris-virginica

Cluster 0 <-- Se agrupan las flores del tipo o clase Iris-versicolor

Cluster 1 <-- Se agrupan las flores del tipo o clase Iris-setosa

Cluster 2 <-- Se agrupan las flores del tipo o clase Iris-virginica

Ejemplos mal agrupados: 17/150 11.3333 %

Observamos que los 50 ejemplos de la flor Iris-Setosa han sido perfectamente agrupados en el *cluster 1*, sin ningún error, es decir no se ha metido en este *cluster* ningún ejemplo de otra clase distinta a la Iris-Setosa.

Se han agrupado 61 ejemplos en el *cluster 0*, de los cuales 14 de ellos pertenecen a la clase Iris-Virginica, lo que quiere decir que estos últimos se han clasificado erróneamente (tenían que haber ido al *cluster 2*).

Por último, se han agrupado 39 ejemplos en el *cluster 2* de los cuales 3 de ellos pertenecen a la clase Iris-Versicolor; esto se toma como que estos han sido también clasificados erróneamente (tenían que haber ido al *cluster 0)*.

Por esta razón, nos dan la cantidad de 17 (14+3) ejemplos mal agrupados, que corresponden al 11’33% del total de los ejemplos. En el gráfico de dispersión de la ilustración 6 se puede observar los ejemplos clasificados y mal clasificados.

No debemos olvidarnos que se han utilizado tres centroides (K=3) para la clasificación de los 150 ejemplos.

Actividad 5

Mostrar en una tabla las coordenadas de los cluster obtenidos en cada semilla indicada. Representar en cada caso el tipo de gráfica indicada en el anterior apartado. Comentar y comparar las agrupaciones obtenidas en cada caso.

Tabla de coordenadas de los cluster obtenidos en cada caso.

Valores de semilla	Coordenadas de los *Cluster*
Semilla = 3	Cluster 0: 6.1,2.9,4.7,1.4 Cluster 1: 6.2,2.9,4.3,1.3 Cluster 2: 6.9,3.1,5.1,2.3
Semilla = 13	Cluster 0: 6.1,2.9,4.7,1.4 Cluster 1: 6.2,2.9,4.3,1.3 Cluster 2: 6.9,3.1,5.1,2.3 Cluster 3: 5.5,4.2,1.4,0.2 Cluster 4: 6.9,3.1,4.9,1.5 Cluster 5: 6.1,3,4.6,1.4 Cluster 6: 4.9,3.1,1.5,0.1 Cluster 7: 4.4,3,1.3,0.2 Cluster 8: 5.5,2.4,3.7,1 Cluster 9: 4.3,3,1.1,0.1 Cluster 10: 6,2.7,5.1,1.6 Cluster 11: 5.7,2.5,5,2 Cluster 12: 4.6,3.1,1.5,0.2
Semilla = 33	Cluster 0: 6.1,2.9,4.7,1.4 Cluster 1: 6.2,2.9,4.3,1.3 Cluster 2: 6.9,3.1,5.1,2.3 Cluster 3: 5.5,4.2,1.4,0.2 Cluster 4: 6.9,3.1,4.9,1.5 Cluster 5: 6.1,3,4.6,1.4 Cluster 6: 4.9,3.1,1.5,0.1 Cluster 7: 4.4,3,1.3,0.2 Cluster 8: 5.5,2.4,3.7,1 Cluster 9: 4.3,3,1.1,0.1 Cluster 10: 6,2.7,5.1,1.6 Cluster 11: 5.7,2.5,5,2 Cluster 12: 4.6,3.1,1.5,0.2 Cluster 13: 7.4,2.8,6.1,1.9 Cluster 14: 5.9,3,5.1,1.8 Cluster 15: 6.9,3.2,5.7,2.3 Cluster 16: 6.7,3.3,5.7,2.5 Cluster 17: 7.2,3.6,6.1,2.5 Cluster 18: 7.3,2.9,6.3,1.8 Cluster 19: 6.1,2.8,4.7,1.2 Cluster 20: 5,3.5,1.3,0.3 Cluster 21: 6.3,3.3,4.7,1.6 Cluster 22: 5.9,3,4.2,1.5 Cluster 23: 5.7,3,4.2,1.2 Cluster 24: 6.7,3.3,5.7,2.1 Cluster 25: 7.7,2.6,6.9,2.3 Cluster 26: 5,3.2,1.2,0.2 Cluster 27: 4.6,3.6,1,0.2 Cluster 28: 6.1,2.6,5.6,1.4 Cluster 29: 6.2,2.2,4.5,1.5 Cluster 30: 6.7,2.5,5.8,1.8 Cluster 31: 6.3,2.5,4.9,1.5 Cluster 32: 7.7,2.8,6.7,2

Semilla = 47	Cluster 0: 6.1,2.9,4.7,1.4 Cluster 1: 6.2,2.9,4.3,1.3 Cluster 2: 6.9,3.1,5.1,2.3 Cluster 3: 5.5,4.2,1.4,0.2 Cluster 4: 6.9,3.1,4.9,1.5 Cluster 5: 6.1,3,4.6,1.4 Cluster 6: 4.9,3.1,1.5,0.1 Cluster 7: 4.4,3,1.3,0.2 Cluster 8: 5.5,2.4,3.7,1 Cluster 9: 4.3,3,1.1,0.1 Cluster 10: 6,2.7,5.1,1.6 Cluster 11: 5.7,2.5,5,2 Cluster 12: 4.6,3.1,1.5,0.2 Cluster 13: 7.4,2.8,6.1,1.9 Cluster 14: 5.9,3,5.1,1.8 Cluster 15: 6.9,3.2,5.7,2.3 Cluster 16: 6.7,3.3,5.7,2.5 Cluster 17: 7.2,3.6,6.1,2.5 Cluster 18: 7.3,2.9,6.3,1.8 Cluster 19: 6.1,2.8,4.7,1.2 Cluster 20: 5,3.5,1.3,0.3 Cluster 21: 6.3,3.3,4.7,1.6 Cluster 22: 5.9,3,4.2,1.5 Cluster 23: 5.7,3,4.2,1.2 Cluster 24: 6.7,3.3,5.7,2.1 Cluster 25: 7.7,2.6,6.9,2.3 Cluster 26: 5,3.2,1.2,0.2 Cluster 27: 4.6,3.6,1,0.2 Cluster 28: 6.1,2.6,5.6,1.4 Cluster 29: 6.2,2.2,4.5,1.5 Cluster 30: 6.7,2.5,5.8,1.8 Cluster 31: 6.3,2.5,4.9,1.5 Cluster 32: 7.7,2.8,6.7,2 Cluster 33: 4.9,2.4,3.3,1 Cluster 34: 5.4,3.4,1.7,0.2 Cluster 35: 6.3,2.8,5.1,1.5 Cluster 36: 5.1,3.5,1.4,0.3 Cluster 37: 4.8,3.4,1.6,0.2 Cluster 38: 6.7,3.1,5.6,2.4 Cluster 39: 4.7,3.2,1.6,0.2 Cluster 40: 5,2,3.5,1 Cluster 41: 4.4,2.9,1.4,0.2 Cluster 42: 6.9,3.1,5.4,2.1 Cluster 43: 5.6,2.8,4.9,2 Cluster 44: 7.9,3.8,6.4,2 Cluster 45: 4.8,3.1,1.6,0.2 Cluster 46: 5.5,2.5,4,1.3

Representación gráfica para cada caso

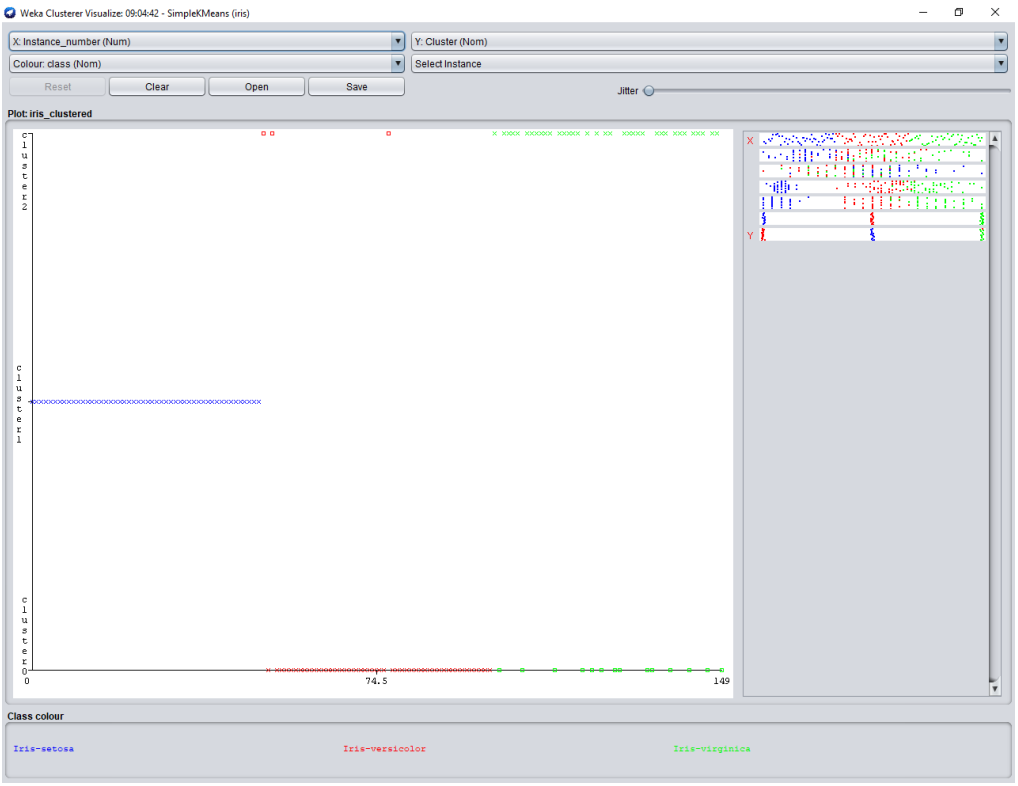


Ilustración 7 - Gráfica con 3 semillas (K=3)

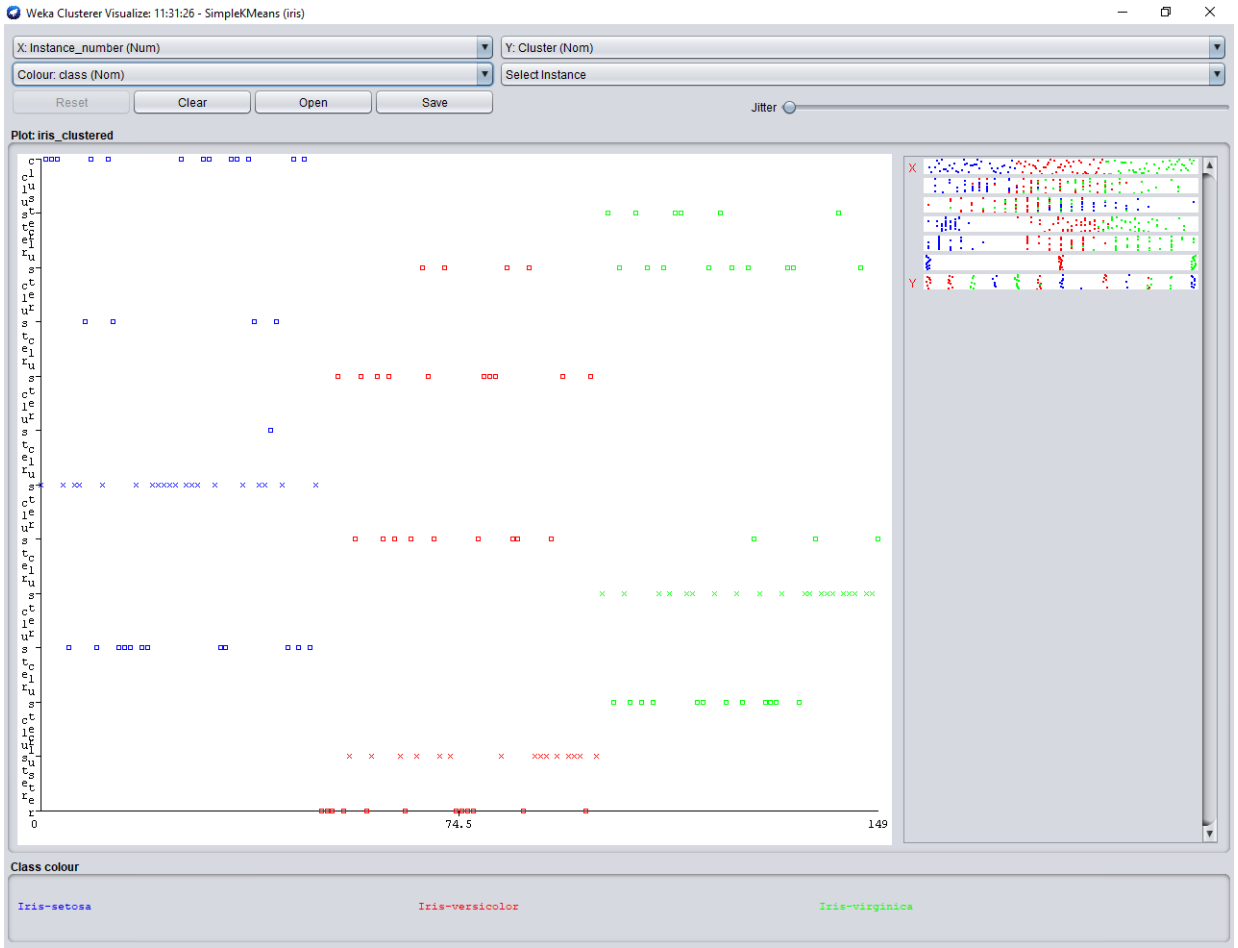


Ilustración 8 - Gráfica con 13 semillas (K=13)

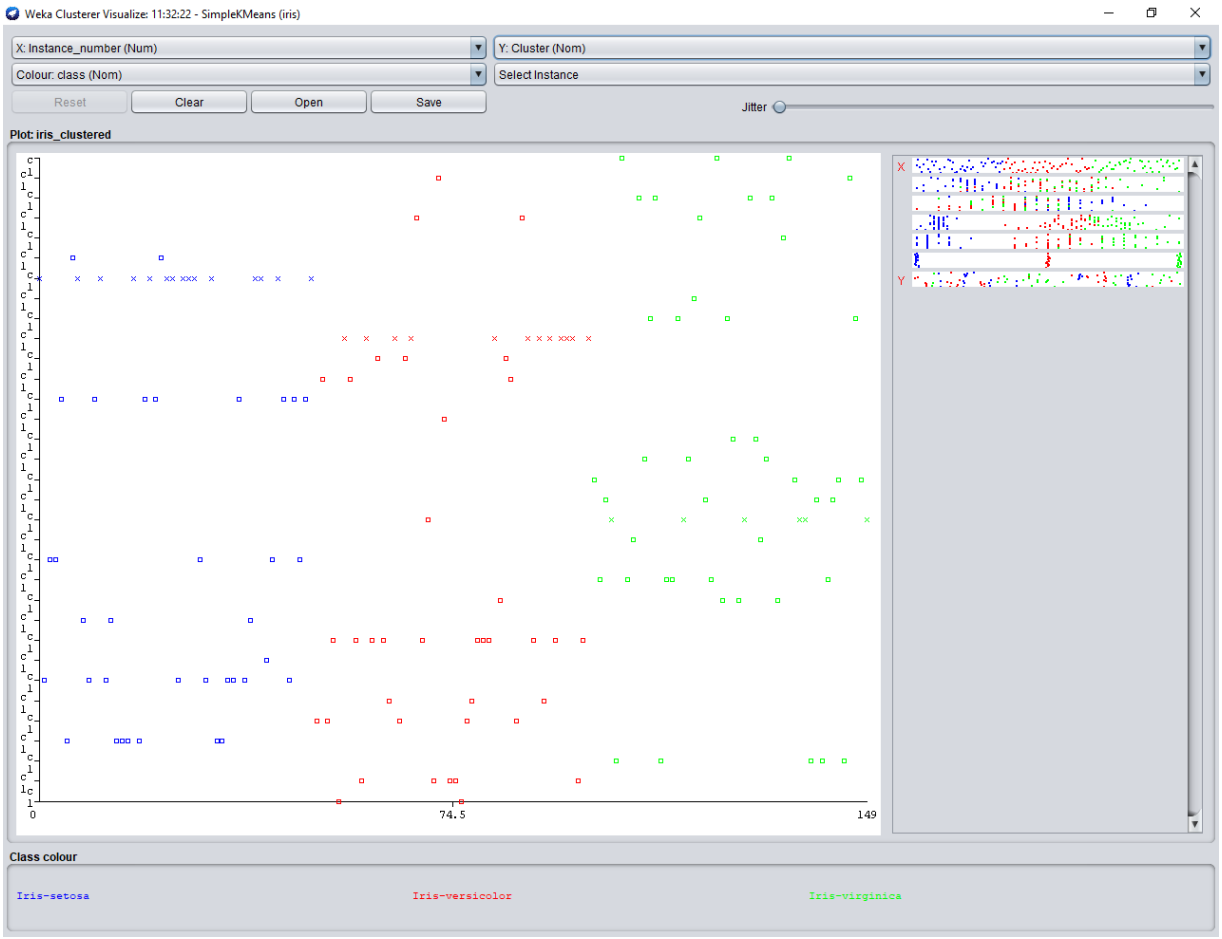


Ilustración 9 - Gráfico con 33 semillas (K=33)

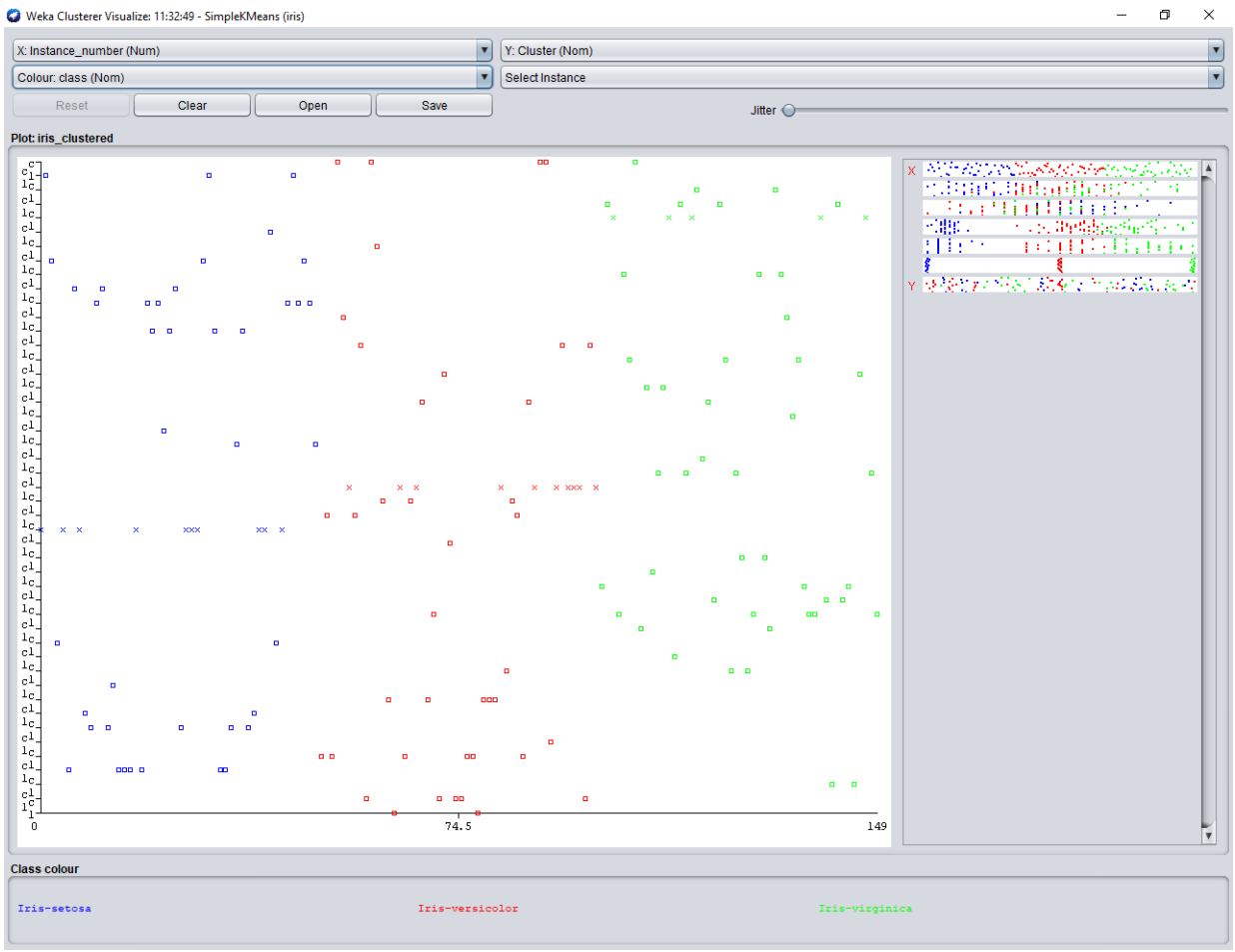


Ilustración 10 - Gráfico con 47 semillas (K=47)

Comentarios de las agrupaciones obtenidas en los distintos ejemplos

Uno de los detalles más significativos para este caso, es que conforme se aumenta el número de semillas, también aumenta el número de errores y la matriz de confusión se vuelve más confusa aún. De este dato se puede deducir que no por haber más semillas, la clasificación es de más calidad; podemos ver que, de los cuatro ejemplos con las distintas semillas, el mejor es el k=3, ya que realiza la mejor clasificación de ejemplos con el menor error, de aquí podemos concluir que esto es una limitación importante en el algoritmo, y que habremos de lanzarlo con varios k hasta encontrar el que mejor resultado dé. También demostramos con esto, la sensibilidad en la fijación que se hace de los centroides en el primer paso, variando de forma significativa los resultados según esta inicialización.

Actividad 6

Representar para cada caso, el tipo de gráfica indicada en el apartado 4, comentando y comparando la bondad de las agrupaciones obtenidas en cada caso. ¿Qué ocurre si solo se seleccionan los atributos "petallength" y "petalwidth"?

Se realizan las siguientes combinaciones de atributos[1]:

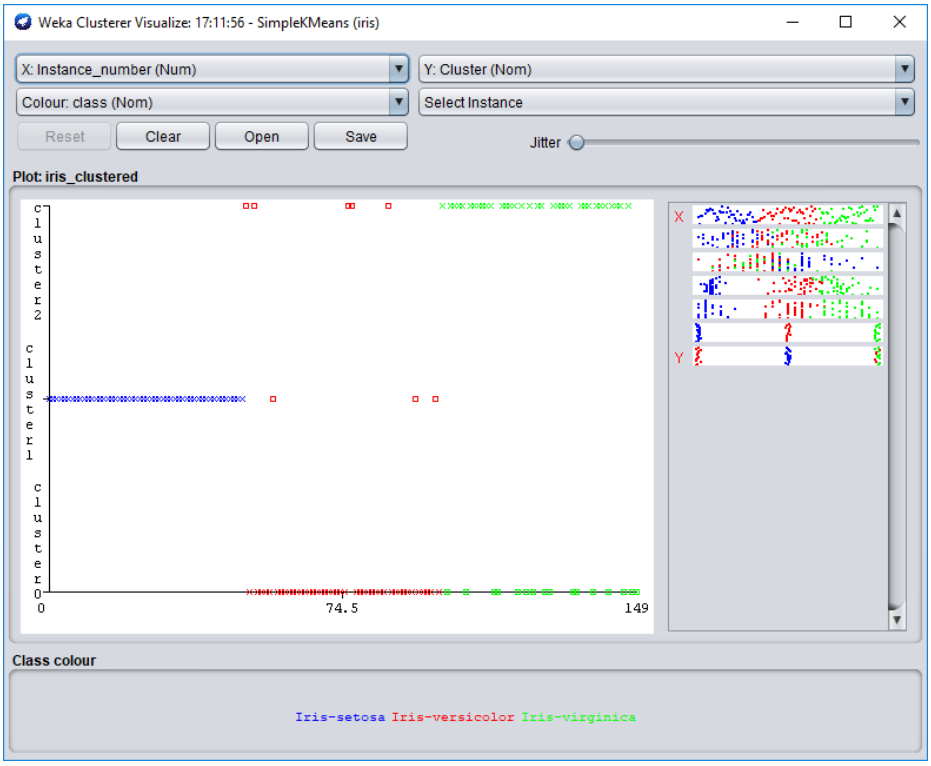
- Sepallength - Petallength
- Sepallength – Sepalwidth
- Sepallength – SepalWidth – Petalwidth
- Sepallength – SepalWidth – Petallength
- SepalWidth – Petallength – Petalwidth
- Sepallength – Petallength – Petalwidth
- Sepalwidth – Petalwidth

Gráficas generadas e instancias incorrectamente clasificadas.

a. Sepallength – Petallength

Número de iteraciones: 5

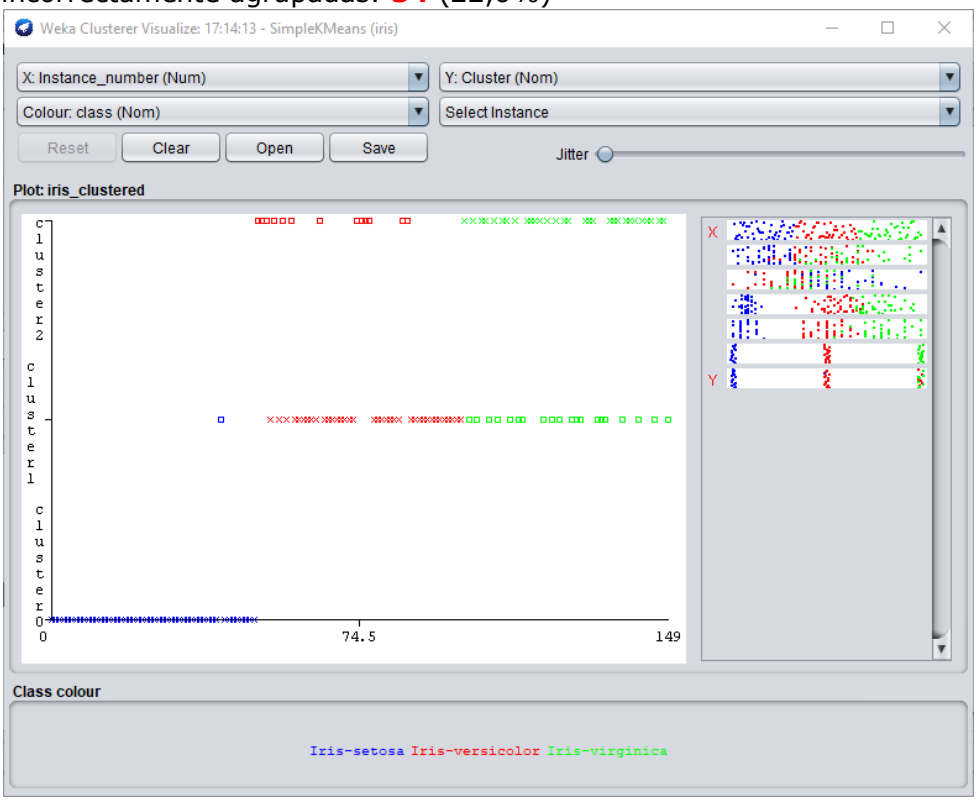
Instancias incorrectamente agrupadas: **24** (16%)



[1] Los atributos que no aparecen en cada caso, han sido ignorados.

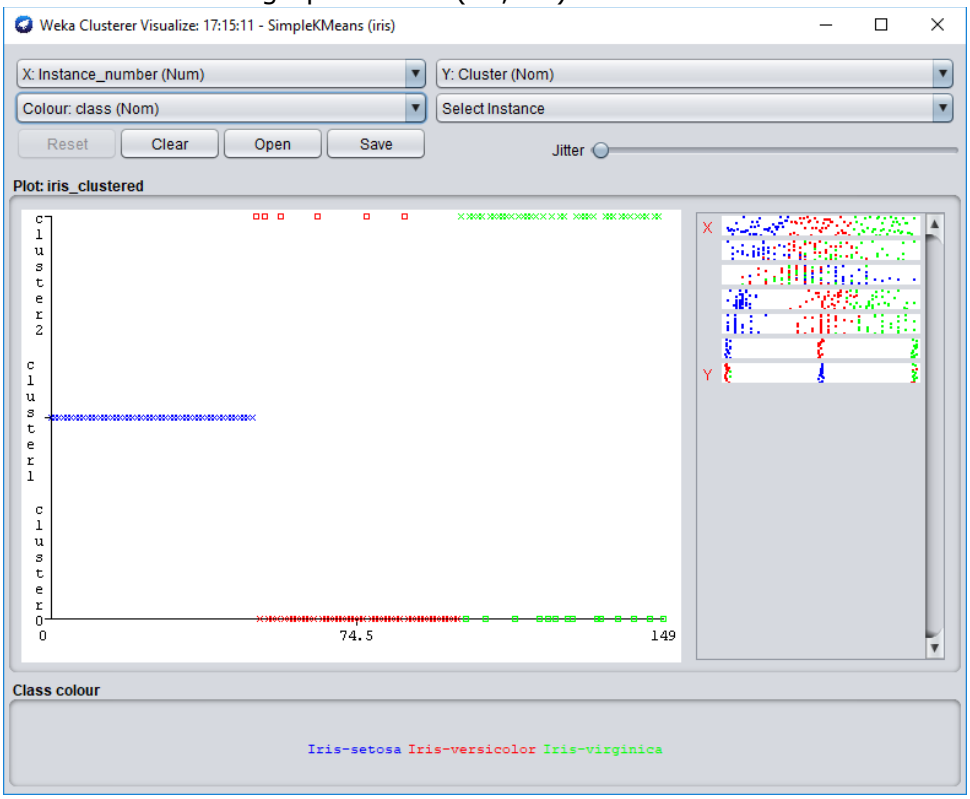
b. Sepallength – Sepalwidth

Número de iteraciones: 7
Instancias incorrectamente agrupadas: **34** (22,6%)



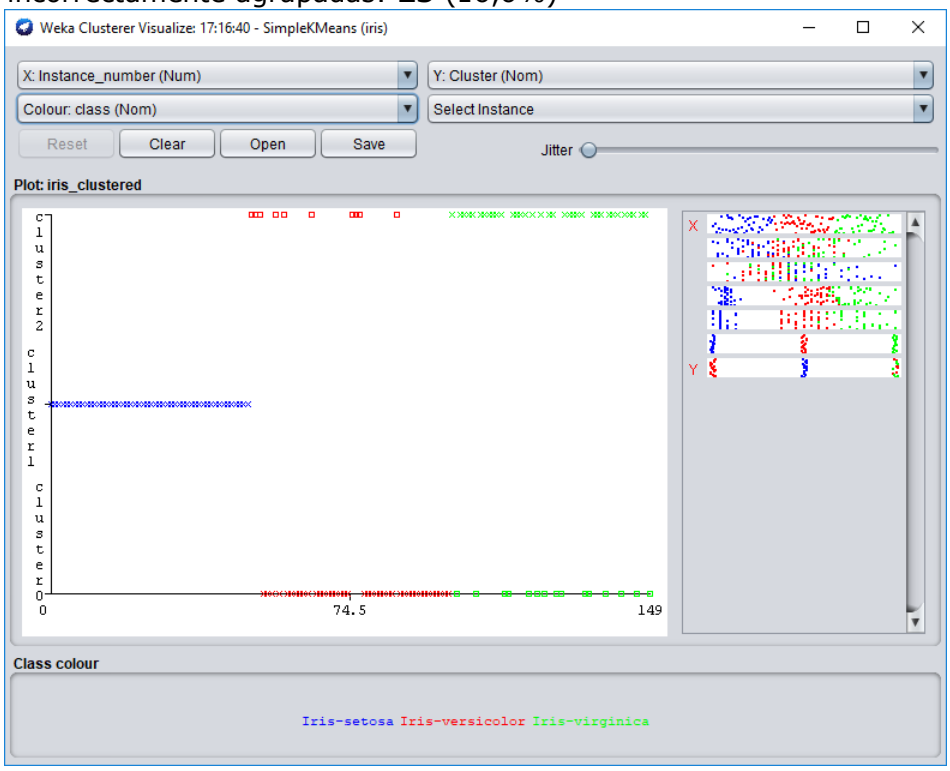
c. Sepallength – SepalWidth – Petalwidth

Número de iteraciones: 7
Instancias incorrectamente agrupadas: **20** (13,3%)



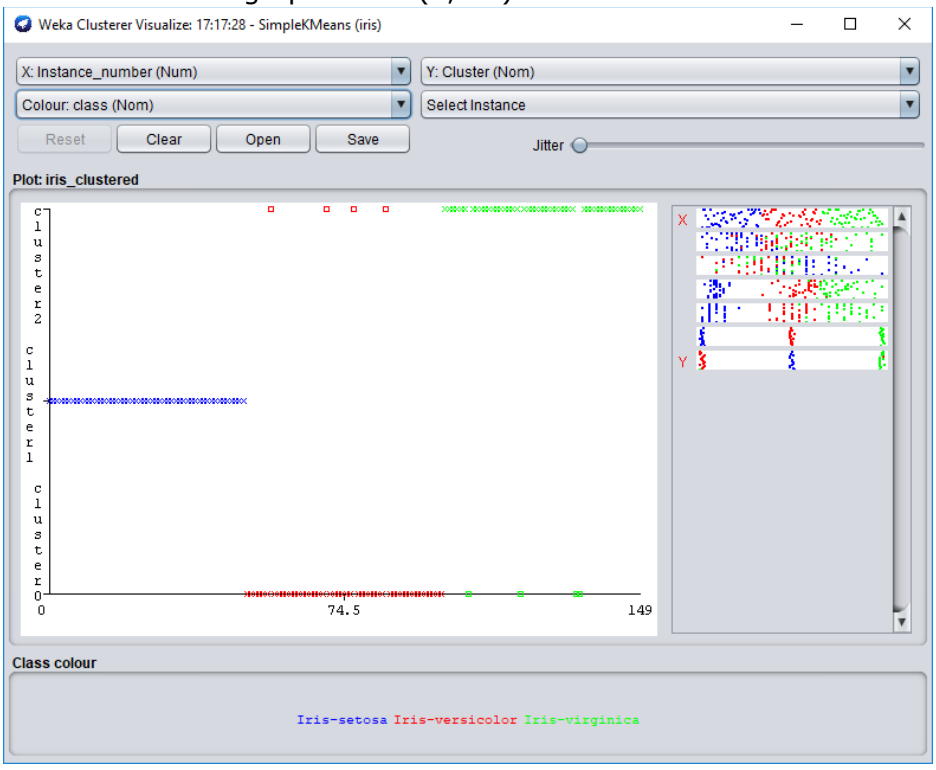
d. Sepallength – SepalWidth – Petallength

Número de iteraciones: 7
Instancias incorrectamente agrupadas: **25** (16,6%)



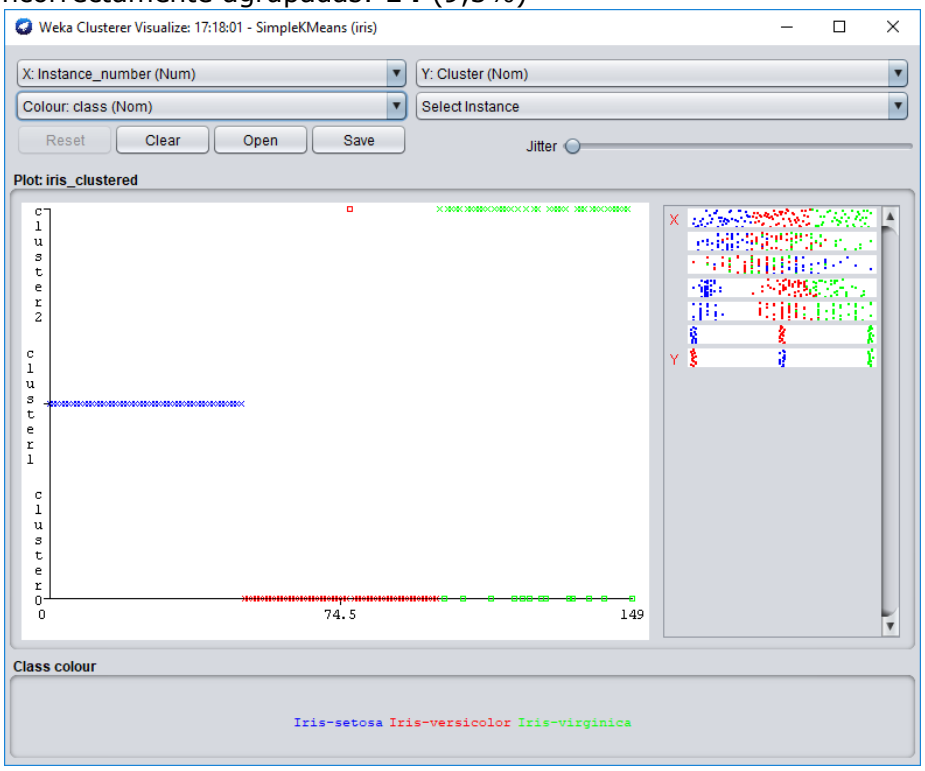
e. SepalWidth – Petallength – Petalwidth

Número de iteraciones: 10
Instancias incorrectamente agrupadas: **8** (5,3%)



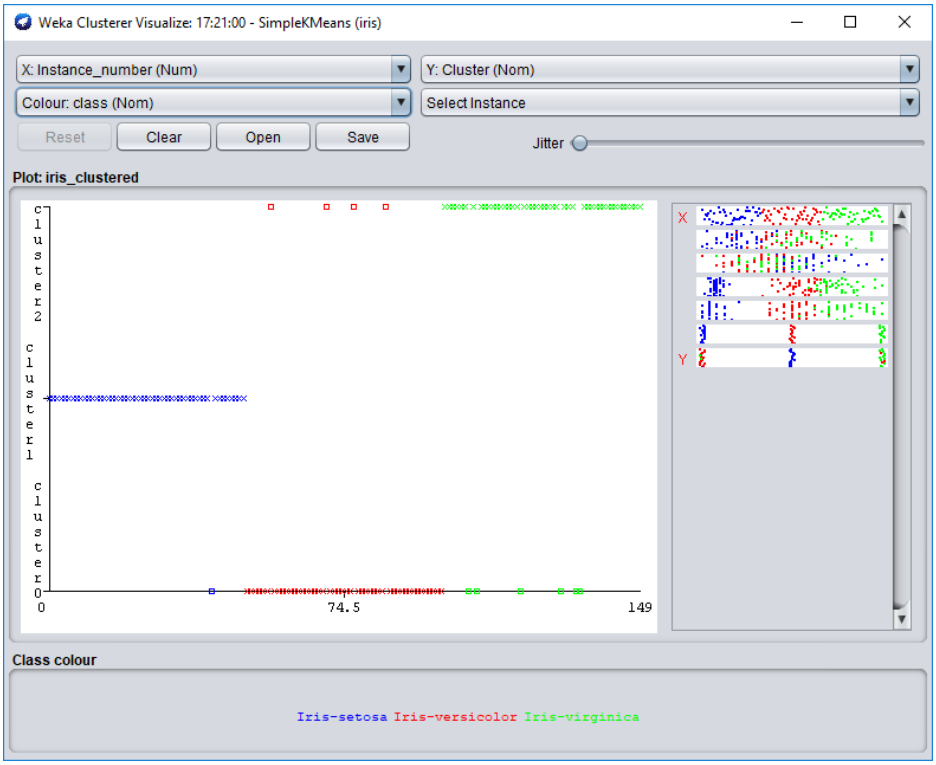
f. Sepallength – Petallength – Petalwidth

Número de iteraciones: 6
Instancias incorrectamente agrupadas: **14** (9,3%)



g. Sepalwidth – Petalwidth

Número de iteraciones: 9
Instancias incorrectamente agrupadas: **11** (7,3%)



Podemos observar que, si solo usamos atributos relacionados con los pétalos (petallength y petalwidth), el número de errores en la clasificación disminuye; incluso combinando con algún atributo relacionado con los sépalos el valor del error no sube mucho. La cosa cambia cuando utilizamos todos los atributos de los sépalos y tan solo uno de los pétalos, ahí vemos como el número de errores en la clasificación comienza a aumentar. Estos errores aumentan considerablemente cuando solo usamos los atributos relacionados con los sépalos (sepallength y sepalwidth) donde los casos clasificados incorrectamente se disparan a su valor más alto (apartado b).

¿Qué ocurre si solo se seleccionan los atributos Petallength y Petalwidth?

Si solo usamos los atributos de los pétalos, nos da el valor de errores de clasificación más pequeño; esto quiere decir que la calidad de la información que nos dan los atributos de los pétalos, es mejor que los datos que nos proporcionan los sépalos. Si comparamos las dos gráficas (apartado b e Ilustración 11), y las instancias incorrectamente agrupadas (34 y 6 respectivamente), corrobora que la información de los pétalos es más fiable que la de los sépalos.

Instancias incorrectamente agrupadas Petallength - Petalwidth: **6** (4%)

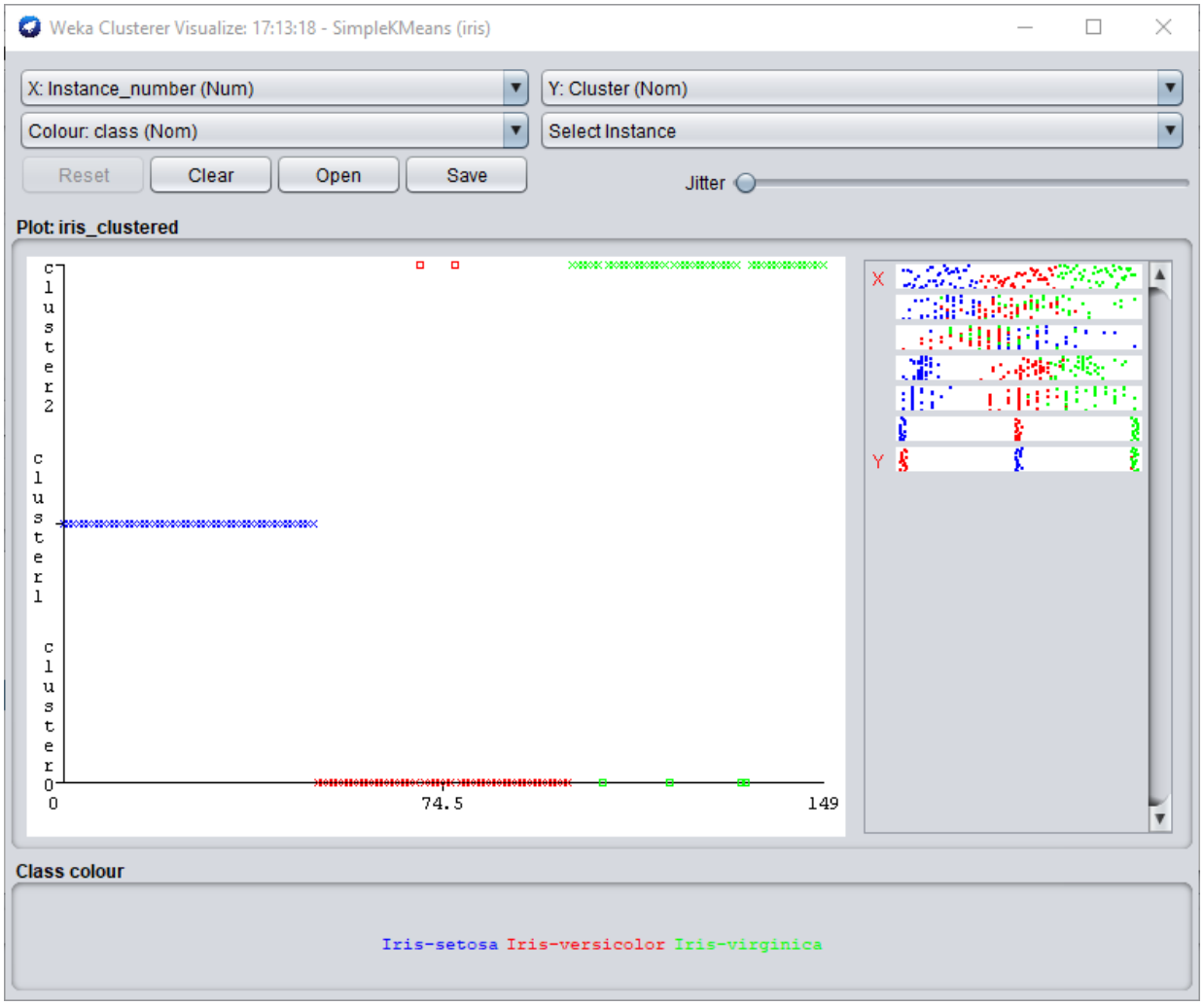
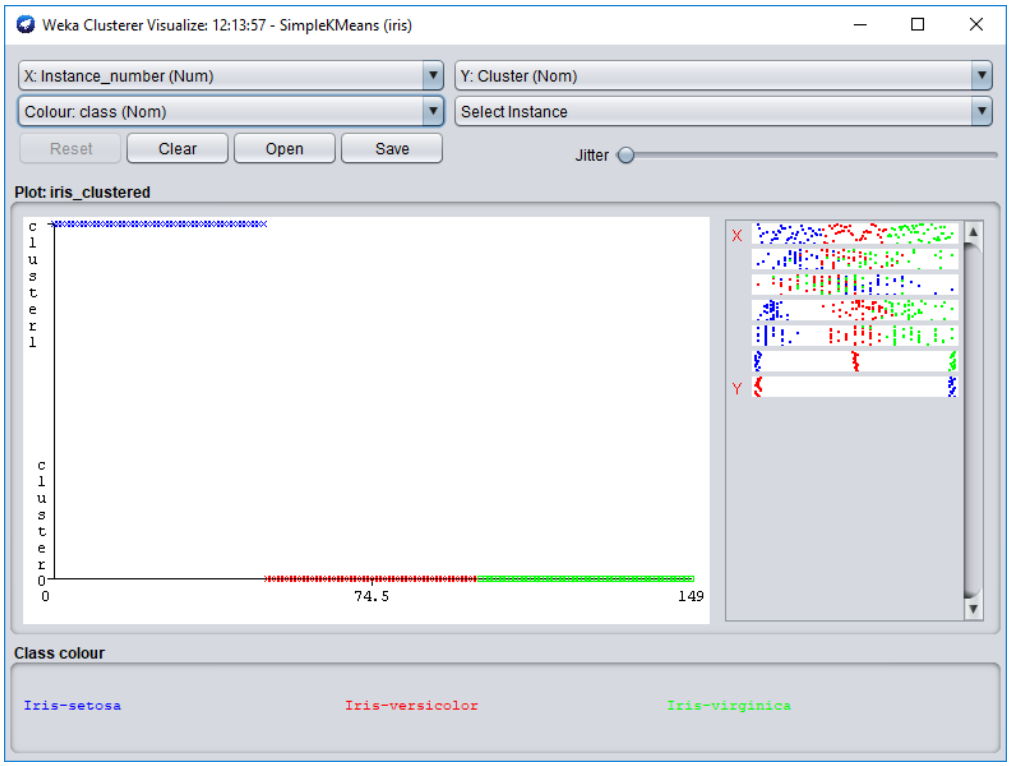


Ilustración 11 - Gráfico de la combinación Petallength-PetalWidth

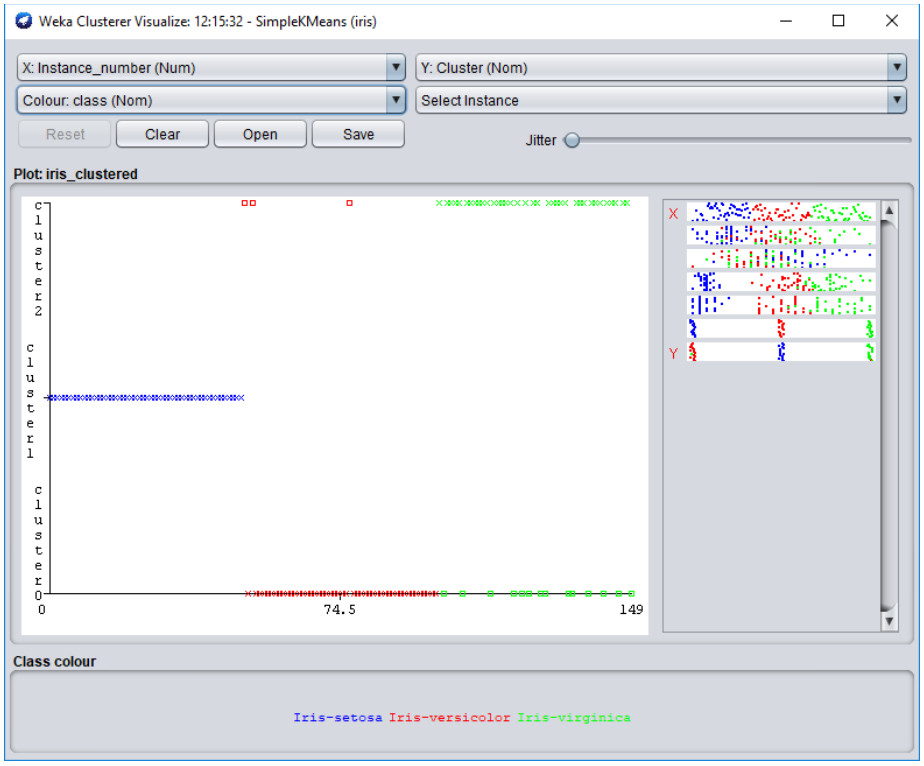
Actividad 7

Repetir el experimento realizado en el apartado 4, pero cambiando los valores del parámetro numCluster: 2, 3, 4 y 5. Representar para cada caso, el tipo de gráfica indicada en el apartado 4. Comentar y comparar la bondad de las agrupaciones obtenidas en cada caso.

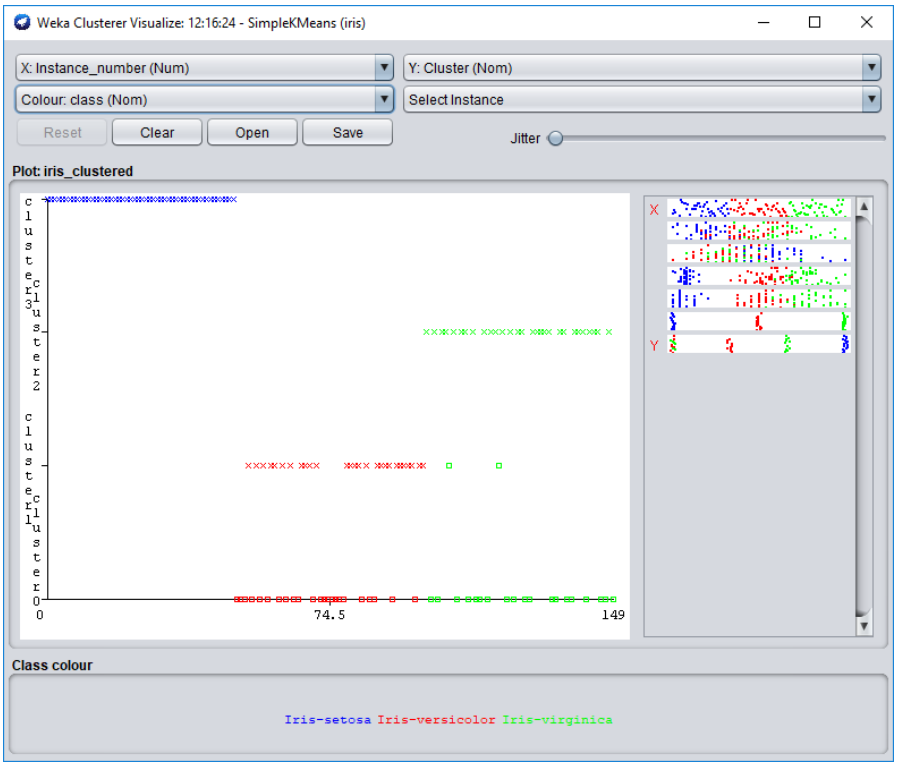
- numCluster = 2: 50 instancias incorrectamente clasificadas, dos agrupaciones.



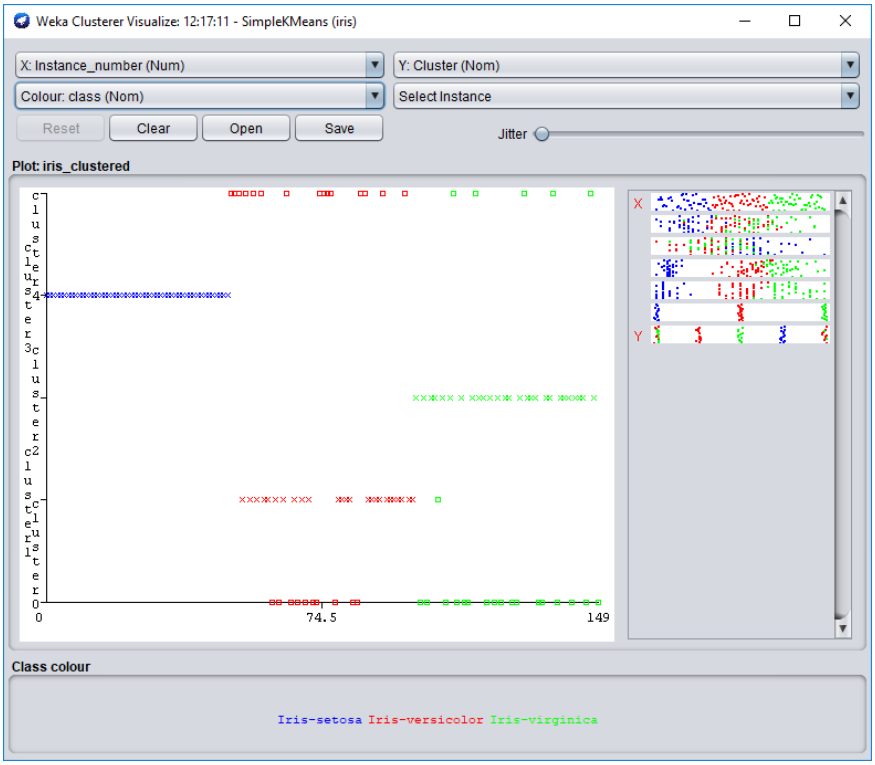
- numCluster = 3: 17 instancias incorrectamente clasificadas, 3 agrupaciones.



- numCluster = 4: 44 instancias incorrectamente clasificadas, 4 agrupaciones.



- numCluster = 5: 48 instancias incorrectamente clasificadas, 5 agrupaciones.



No hay duda que el k=3 es el más adecuado para la clasificación de los ejemplos de la base de datos. Los resultados obtenidos con los demás valores del número de agrupaciones (numCluster =2, 4 y 5) varían entre 40 y 50 instancias no clasificadas correctamente, sin embargo con numCluster=3, el número de instancias no clasificadas correctamente son solo 17, corroborando pues que el número adecuado de agrupaciones es 3.

Conclusiones

Las conclusiones que podemos sacar de este algoritmo en base a los experimentos realizados es que:

- El algoritmo depende de las semillas y de donde caigan al principio.
- A la hora de realizar las agrupaciones, tenemos que indicarle al algoritmo los grupos fijos, y esto puede ser bastante sencillo en unas ocasiones, pero no tan sencillo en otras, teniendo que probar en este caso con varias agrupaciones hasta dar con la más óptima. Esto de tener que fijar las clases de antemano, es una limitación importante del algoritmo.
- Los ejemplos solamente pueden pertenecer a una clase.
- Podemos ver que la agrupación es prácticamente automática.
- Se puede decir que los costes computacionales son bastante buenos: el coste espacial es lineal y el coste temporal es polinomial.