

# Clustering en Weka

Laura Rodríguez Navas  
rodrigueznava@posgrado.uimp.es

July 10, 2020

En esta práctica se realiza un estudio acerca de la base de datos Iris. Esta BD se distribuye junto a la herramienta Weka.

1. Ejecuta el algoritmo SimpleKMeans usando la herramienta Weka con las distancias Euclídea y Manhattan.

La BD está formada por 4 variables descriptivas y una variable clase. No se aplica preprocesamiento de datos ya que las variables descriptivas son numéricas y no existen valores perdidos en la BD. Además, como la variable de clase puede tomar tres valores, elegimos que el valor de  $k$  sea igual a 3.

```
Clusterer output

Number of iterations: 3
Within cluster sum of squared errors: 7.817456892309574

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4,Iris-versicolor
Cluster 1: 6.2,2.9,4.3,1.3,Iris-versicolor
Cluster 2: 6.9,3.1,5.1,2.3,Iris-virginica

Missing values globally replaced with mean/mode

Final cluster centroids:

Attribute          Full Data          Cluster#
                   (150.0)            0
                   (50.0)            1
                   (50.0)            2
                   (50.0)
=====
sepalength          5.8433             5.936             5.006             6.588
sepalwidth          3.054              2.77              3.418             2.974
petallength         3.7587             4.26              1.464             5.552
petalwidth          1.1987             1.326            0.244             2.026
class               Iris-setosa Iris-versicolor  Iris-setosa  Iris-virginica

Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      50 ( 33%)
1      50 ( 33%)
2      50 ( 33%)
```

Figura 1: KMeans con distancia Euclídea.

- (a) ¿Cuántas instancias contiene cada grupo?

En la ejecución del algoritmo KMeans con distancia Euclídea (ver Figura 1) se han formado tres grupos: 0, 1 y 2. Los 3 grupos contienen 50 instancias cada uno. En la ejecución del algoritmo KMeans con distancia Manhattan (ver Figura 2) también se han formado los grupos 0, 1 y 2, con el mismo número de instancias cada uno.

```

Clusterer output

Number of iterations: 3
Sum of within cluster distances: 49.87499999999999

Initial starting points (random):

Cluster 0: 6.1,2.9,4.7,1.4,Iris-versicolor
Cluster 1: 6.2,2.9,4.3,1.3,Iris-versicolor
Cluster 2: 6.9,3.1,5.1,2.3,Iris-virginica

Missing values globally replaced with mean/mode

Final cluster centroids:

Attribute      Full Data      Cluster#
              (150.0)      0          1          2
              (50.0)      (50.0)      (50.0)
=====
sepal.length    5.8           5.9           5           6.5
sepal.width     3             2.8           3.4           3
petal.length    4.35          4.35          1.5           5.55
petal.width     1.3           1.3           0.2           2
class           Iris-setosa  Iris-versicolor  Iris-setosa  Iris-virginica

Time taken to build model (full training data) : 0 seconds

=== Model and evaluation on training set ===

Clustered Instances

0      50 ( 33%)
1      50 ( 33%)
2      50 ( 33%)

```

Figura 2: KMeans con distancia Manhattan.

(b) ¿Cuáles son los centroides?

Si nos volvemos a fijar en la figura 1, podemos observar los centroides de la ejecución de Kmeans con distancia Euclídea, y si nos volvemos a fijar en la figura 2, podemos observar los centroides de la ejecución de Kmeans con distancia Manhattan. Se muestran más detalladamente en las siguientes figuras:

```

Final cluster centroids:

Attribute      Full Data      Cluster#
              (150.0)      0          1          2
              (50.0)      (50.0)      (50.0)
=====
sepal.length    5.8433        5.936         5.006         6.588
sepal.width     3.854         2.77          3.418         2.974
petal.length    3.7587        4.26          1.464         5.552
petal.width     1.1987        1.326         0.244         2.026
class           Iris-setosa  Iris-versicolor  Iris-setosa  Iris-virginica

```

Figura 3: KMeans centroides con distancia Euclídea.

```

Final cluster centroids:

Attribute      Full Data      Cluster#
              (150.0)      0          1          2
              (50.0)      (50.0)      (50.0)
=====
sepal.length    5.8           5.9           5           6.5
sepal.width     3             2.8           3.4           3
petal.length    4.35          4.35          1.5           5.55
petal.width     1.3           1.3           0.2           2
class           Iris-setosa  Iris-versicolor  Iris-setosa  Iris-virginica

```

Figura 4: KMeans centroides con distancia Manhattan.

(c) Analiza los centroides. ¿Hay algo destacable en esos centroides? ¿Están los centroides separados en el espacio? ¿Tienen componentes similares?

Los centroides resultantes de la ejecución del algoritmo KMeans con distancia Euclídea y los centroides resultantes de la ejecución del algoritmo KMeans con distancia Manhattan son muy parecidos. Tienen componentes muy similares. Este comportamiento nos podría indicar que en la BD no existen *outliers*, la distancia Manhattan se ve menos afectada por

ellos (es más robusta), y al no presentar diferencias significantes con la distancia Euclídea, este podría ser el motivo de tanta similitud.

En la figura 5 se puede observar que los centroides están separados en el espacio. Aunque los centroides de los grupos 0 y 1 están más cercanos entre ellos. El centroide del grupo 2 es el que está más alejado.

Nota: Solo se muestra el gráfico de los centroides resultantes de la ejecución del algoritmo KMeans con distancia Euclídea, porque casi es igual al gráfico de los centroides resultantes de la ejecución del algoritmo KMeans con distancia Manhattan.



Figura 5: Representación de los centroides de KMeans con distancia Euclídea.

2. Ejecuta el algoritmo HierarchicalClusterer con tipo de enlace completo y métrica de distancia euclídea, y visualice las gráficas de los puntos agrupados. ¿Alguno de ellas produce grupos bien diferenciados y con fronteras claras?

Nota: Compara que el eje X instance\_number y el eje Y vaya variando y muestra cada una de las variables (debes adjuntar las imágenes).

Como se ha comentado anteriormente, la variable de clase puede tomar tres valores, así que volvemos a elegir que el valor de  $k$  sea igual a 3 para la ejecución del algoritmo HierarchicalClusterer (ver Figura ).

Las gráficas donde el eje Y es igual a las variables *sepalwidth* i *sepalwidth* muestran grupos bien diferenciados y con las fronteras claras.

```

Clusterer output

=== Run information ===

Scheme:      weka.clusterers.HierarchicalClusterer -N 3 -L COMPLETE -P -A "weka.core.EuclideanDistance -R first-last"
Relation:    iris
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              petalwidth
              class
Test mode:    evaluate on training data

=== Clustering model (full training set) ===

Cluster 0
(((((((0.0:0.03254,0.0:0.03254):0.01754,0.0:0.05008):0.05008,(0.0:0.06514,(0.0:0.03254,0.0:0.03254):0.0326):0.03501):0.03501,
Cluster 1
(((((((1.0:0.10206,(1.0:0.06508,1.0:0.06508):0.03698):0.0771,1.0:0.17916):0.0424,(((1.0:0.09914,(1.0:0.08779,1.0:0.08779):0.08779,
Cluster 2
(((((((2.0:0.15339,(2.0:0.10972,(2.0:0.06047,2.0:0.06047):0.04924):0.04367):0.10143,(((2.0:0.04383,2.0:0.04383):0.03891,

Time taken to build model (full training data) : 0.04 seconds

=== Model and evaluation on training set ===

Clustered Instances
0      50 ( 33%)
1      50 ( 33%)
2      50 ( 33%)

```

Figura 6: Hierarchical Clustering con distancia Euclídea.

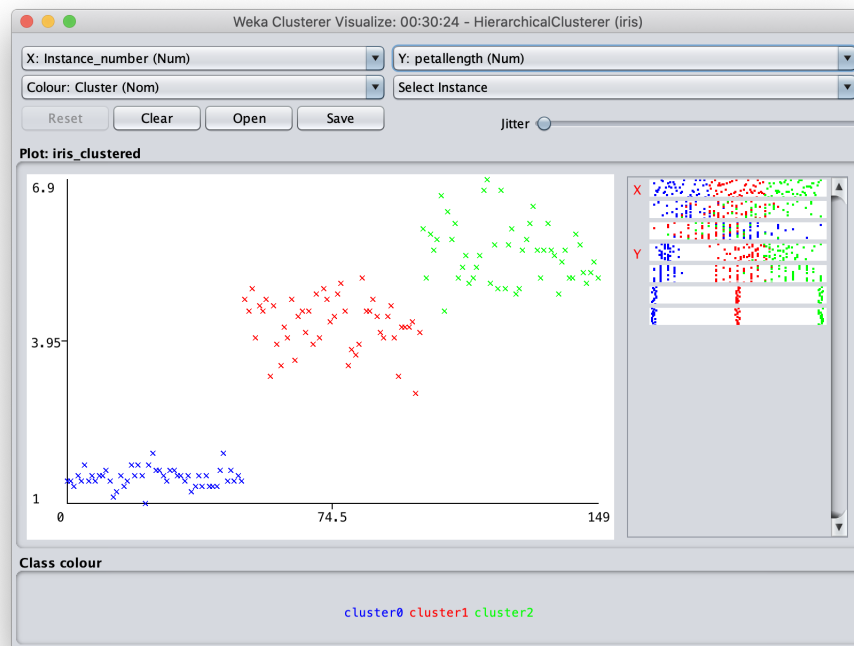


Figura 7: X instance\_number con distancia Manhattan.

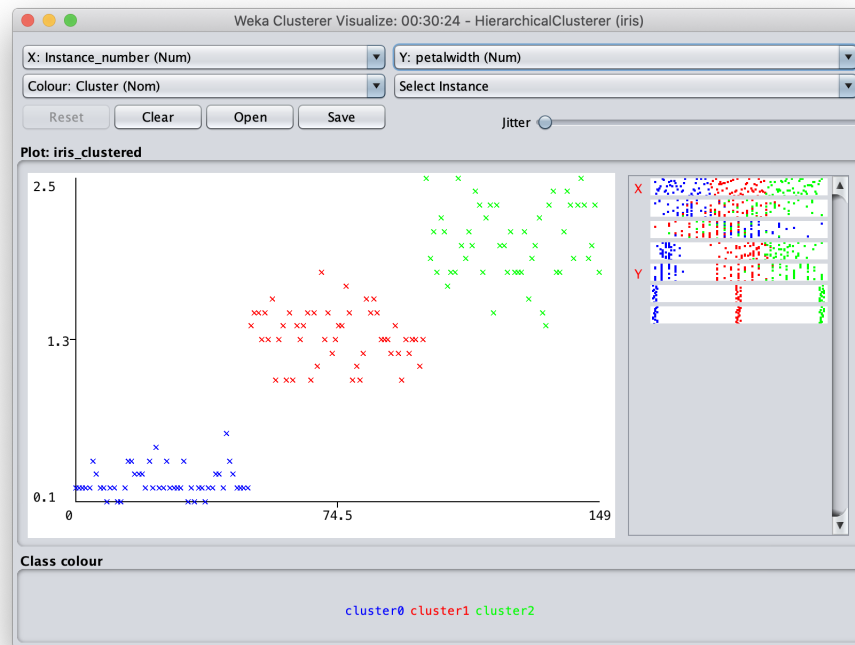


Figura 8: X instance\_number con distancia Manhattan.

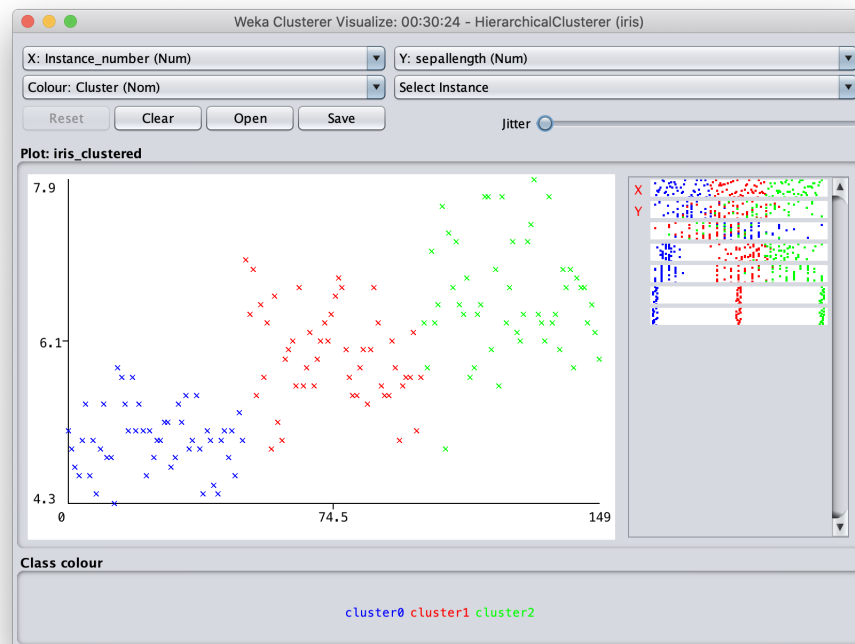


Figura 9: X instance\_number con distancia Manhattan.

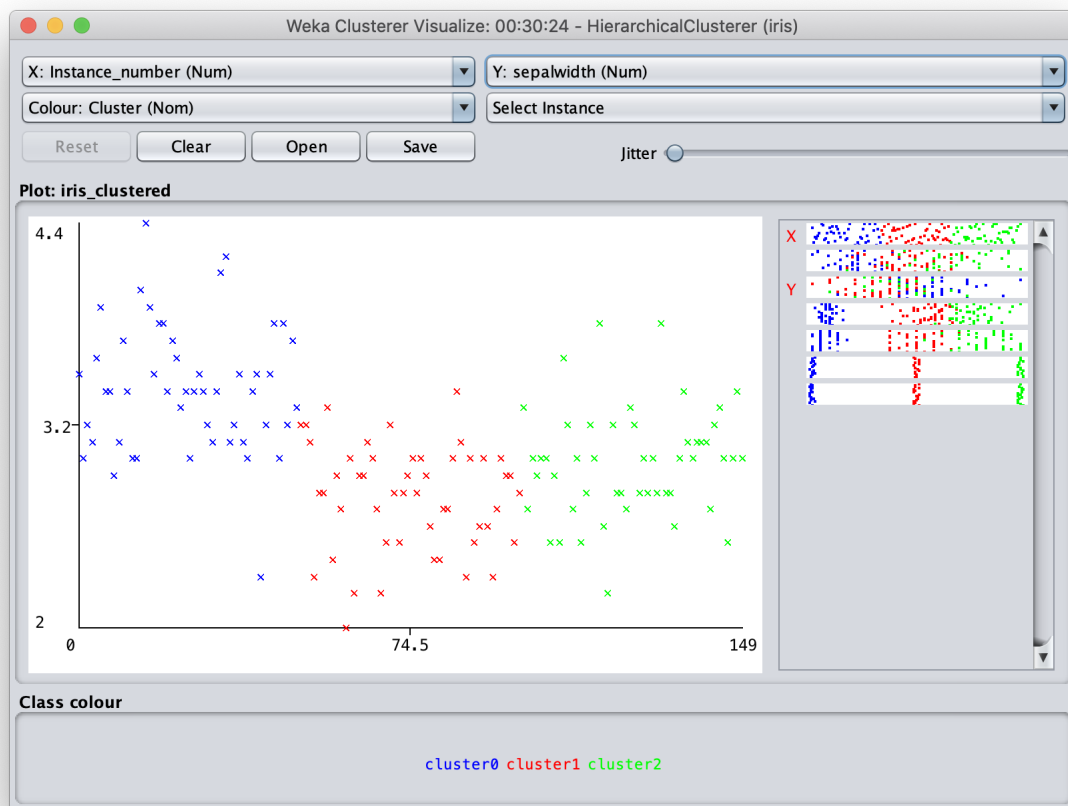


Figura 10: X instance\_number con distancia Manhattan.