

Aprendizaje Automático - Trabajo Práctico 4

Gonzalo Castiglione - 49138

May 30, 2012

Objetivo: Aprender a agrupar datos mediante análisis de clusters

1 Determinación de clases. Clustering.

1. (TO DO)
2. Lirios de Fisher
 - (a) Clasificación de las mediciones según el algoritmo *kmeans*

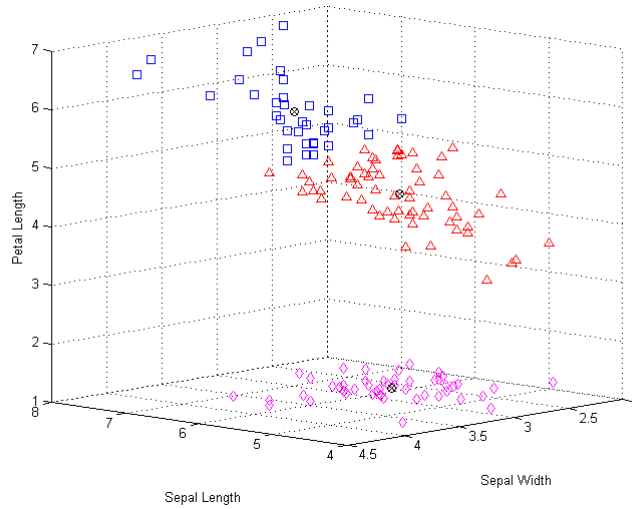


Figure 1: Agrupamiento de los lirios de Fisher

Matriz de confusión

| | Clase Predicha | | | |
|------------|----------------|--------|------------|-----------|
| | | Setosa | Versicolor | Virginica |
| Clase Real | Setosa | 50 | 0 | 0 |
| | Versicolor | 0 | 42 | 8 |
| | Virginica | 0 | 14 | 36 |

$$error = \frac{8 + 14}{150} \simeq 0.15$$

- (b) Calsificación segun algoritmo de clusters jerárquicos aglomerativos

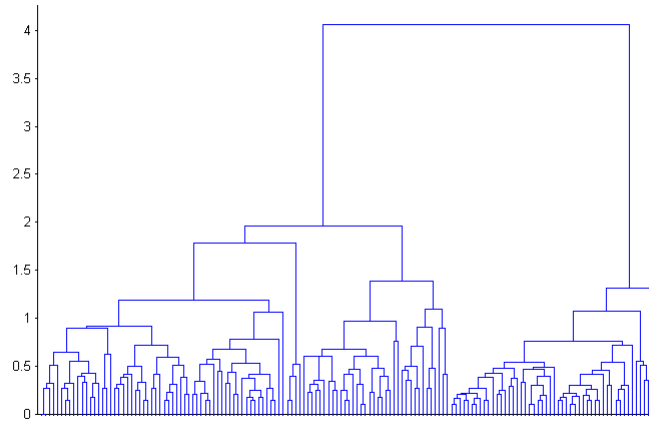


Figure 2: Agrupamiento de los lirios de Fisher

Matriz de confusión

| Clase Real | Clase Predicha | | | |
|------------|----------------|--------|------------|-----------|
| | | Setosa | Versicolor | Virginica |
| | Setosa | 50 | 0 | 0 |
| | Versicolor | 0 | 49 | 1 |
| | Virginica | 0 | 15 | 35 |

$$error = \frac{1 + 15}{150} \simeq 0.11$$

- (c) Luego de comprados los resultados obtenidos por cada algoritmo en los puntos (a) y (b) se pudo concluir que para este problema, el algoritmo que mejor clasifica a los los lirios recogidos es el de clasificación jerárquica ya que obtuvo un error menor al momento de realizar las clasificaciones.

Matriz obtenida en el ejercicio 5 del tp 2:

| Clase Real | Clase Predicha | | | |
|------------|----------------|--------|------------|-----------|
| | | Setosa | Versicolor | Virginica |
| | Setosa | 50 | 0 | 0 |
| | Versicolor | 0 | 47 | 3 |
| | Virginica | 0 | 3 | 47 |

$$error = \frac{3 + 3}{150} \simeq 0.04$$

Como se pudo observar, este algoritmo clasificó aun mejor que los anteriores a los lirios, en especial para los de la clase *Virginica*, que es donde mayor cantidad de errores tuvieron los dos anteriores.

3. (TO DO)

4. Congressistas

- (a) Luego de realizadas pruebas de clasificación variando el tipo de distancia entre *euclidean* y *cityblock*. Y el tipo de metodo de creación del árbol jerárquico entre *single*, *complete* y *average*. Se pudo observar que todos pudieron calificar correctamente a todos los congresistas de acuerdo a si es Demócrata o Republicano (excepto por uno que siempre es clasificado mal) a excepción del método *single*, el cual terminó por clasificar a todos como Demócratas excepto por el número 2, que fue clasificado correctamente con Republicano.

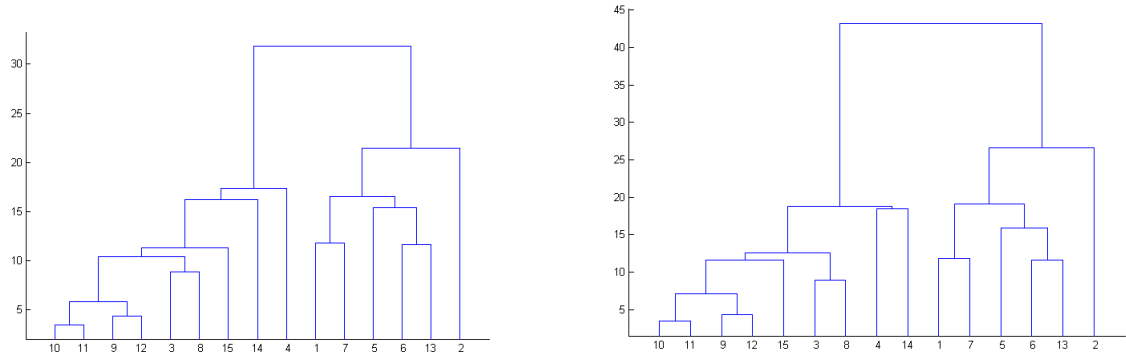


Figure 3: Árbol de jerarquía creados utilizando la distancia *euclidean* con el método *average*(izquierda) y el método *complete*(derecha)

Para los métodos aplicados en los gráficos anteriores, se puede observar que si bien se obtuvieron distintas categorías para varios conjuntos de congresistas, tomando clusters de 2 (por los grupos demócrata o republicano) ambos algoritmos clasificaron correctamente a todos los congresistas a excepción del número 12 que en realidad pertenece al partido republicano.

A continuación se muestra la matriz de confusión para ambos árboles jerárquicos (notar que ambos presentan la misma matriz):

| Clase Real | Clase Predicha | |
|-------------|----------------|-------------|
| | | |
| | Demócrata | Republicano |
| Demócrata | 8 | 1 |
| Republicano | 0 | 6 |

$$error = \frac{1}{15} \simeq 0.07$$

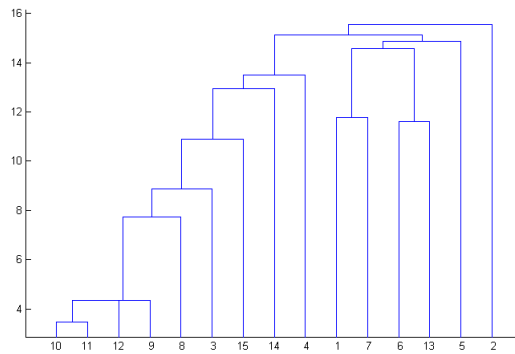


Figure 4: Árbol de jerarquía creado utilizando la distancia *euclidean* con el método *single*

Como se puede ver en la figura 4, no se obtuvieron buenos resultados al tratar de utilizar el método *single* (minimas distancias) para agrupar los datos ya que fueron clasificados todos dentro de la misma categoría, a excepción del número 2.

2 Código

1. (TO DO)

2. Lirios de Fisher

```
// a)
ptsymb = {'bs','r^','md','go','c+'};
load fisheriris;
k = 3;
[clust,cmeans,sumd] = kmeans(meas,k);
for i = 1:k
    clust = find(clust==i);
    plot3(meas(clust,1), meas(clust,2), meas(clust,3), ptsymb{i});
    hold on
end
plot3(cmeans(:,1), cmeans(:,2), cmeans(:,3),'ko');
plot3(cmeans(:,1), cmeans(:,2), cmeans(:,3),'kx');
hold off
xlabel('Sepal Length');
ylabel('Sepal Width');
zlabel('Petal Length');
view(-137,10);
grid on
types = [ones(1,50)*2 ones(1,50) ones(1,50)*3];
cMat = confusionmat(types,clust)
// b)
dist = pdist(meas, 'euclidean');
clustTree = linkage(dist, 'average');
[h, nodes] = dendrogram(clustTree);
set(gca, 'TickDir', 'out', 'TickLength', [.002 0], 'XTickLabel', []);
cluster_labels = cluster(clustTree, 'maxclust', 3 );
confusion_matrix = crosstab( cluster_labels, class )
```

3. (TO DO)

4. Congresistas

```
Y = pdist(X, 'cityblock');
Z = linkage(Y, 'average');
H = dendrogram(Z);
```