Bases de Datos Masivas (11088)
Departamento de Ciencias Básicas

Calidad del agrupamiento: Coeficiente de Silueta

Banchero, Santiago – Octubre de 2015

Coeficiente de Silueta

El coeficiente de Silueta es una métrica para evaluar la calidad del agrupamiento obtenido con algoritmos de *clustering*. El objetivo de Silueta es identificar cuál es el número óptimo de agrupamientos.

En los algoritmos de aprendizaje no supervisado, la cantidad de grupos puede ser un parámetro de entrada del algoritmo o puede ser determinado automáticamente por el algoritmo. En el primer caso, como ocurre con el algoritmo de *K-Mean*, la determinación del número óptimo de *clusters* tiene que ser realizado mediante alguna medida externa al algoritmo. El coeficiente de silueta es indicador del número ideal de *clusters*. Un valor más alto de este índice indica un caso más deseable del número de *clusters*.

El coeficiente de Silueta para una observación *i* se denota como *s(i)* y se define como:

$$s(i)=\frac{b-a}{max(a,b)}$$

Donde:

- ***a*** es el promedio de las disimilitudes (o distancias) de la observación *i* con las demás observaciones del *cluster* al que pertenece *i*

- ***b*** es la distancia mínima a otro *cluster* que no es el mismo en el que está la observación *i*. Ese *cluster* es la segunda mejor opción para *i* y se lo denomina vecindad de *i.*

El valor de *s(i)* puede ser obtenido combinando los valores de ***a*** y ***b*** como se muestra a continuación:

$$s(i) = \begin{cases} 1-\frac{a}{b}, & si\ a<b \\ 0, & si\ a=b \\ \frac{b}{a}-1, & si\ a>b \end{cases}$$

El coeficiente de Silueta es un valor comprendido entre -1 y 1.

$$-1 \le s(i) \le 1$$

Analicemos las posibles soluciones, para que el coeficiente de Silueta sea cercano a 1 el valor de ***b***

tiene que ser mayor al de ***a***. Esto significa que la distancia de la observación *i* a los *clusters* vecinos es suficientemente grande para que su pertenencia al *cluster* actual sea la correcta. Es decir, no es similar a sus vecinos.

Un valor de *s(i)* que sea cercano a cero nos va a indicar que la observación *i* está en la frontera de dos *clusters*.

Y si el valor de *s(i)* es negativo, entonces la observación *i* debería ser asignada al *cluster* más cercano.

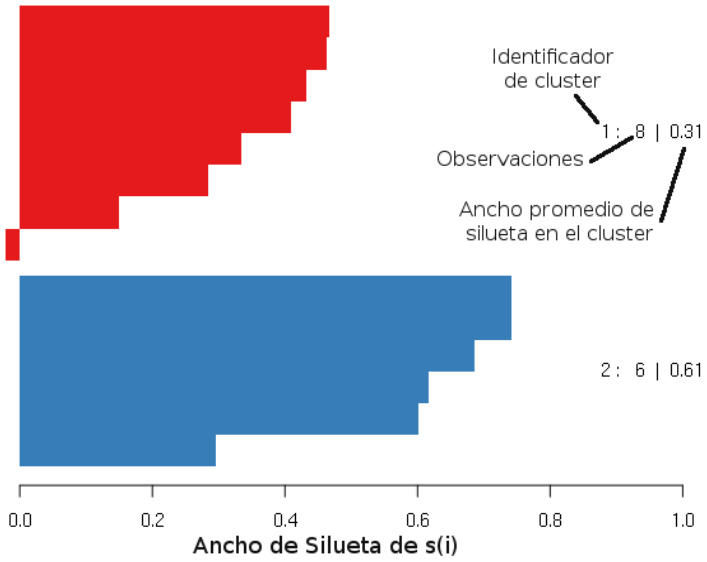
Resumiendo:

- *s(i)* ≈ 1, la observación *i* está bien asignada a su *cluster*
- *s(i)* ≈ 0, la observación *i* está entre dos *cluster*
- *s(i)* ≈ −1, la observación *i* está mal asignada a su *cluster*

Podemos calcular el coeficiente de Silueta como el promedio de todos los *s(i)* para todas las observaciones del conjunto de datos.

Interpretación del Gráfico de Silueta

Es posible realizar una interpretación visual del cálculo de Silueta como se muestra en el siguiente gráfico:



A partir del análisis visual de este gráfico es posible determinar cuál es el número correcto de

Calidad del agrupamiento: Coeficiente de Silueta

Banchero, Santiago – Octubre de 2015

clusters en el conjunto de datos analizado. El gráfico de silueta puede ser analizado como un gráfico de barras horizontales donde cada una de las barras representa una observación *i* para la cual se calculó *s(i)* que se muestra en el eje horizontal.

En este gráfico de ejemplo realizado para un K = 2 se muestra el identificador de cluster, la cantidad de observaciones que lo componen y el ancho de silueta promedio dentro del cluster.

Podemos observar que para el *cluster* superior (en rojo) una de las instancias tiene un valor de *s(i)* menor a cero, es decir, está mal asignado a ese *cluster.* Esto permite intuir que K = 2 no es el número correcto de *clusters* para este conjunto de datos, y vamos a tener que analizar con otros K diferentes.

Ejemplo utilizando R

Para el ejemplo utilizaremos los siguientes datos:

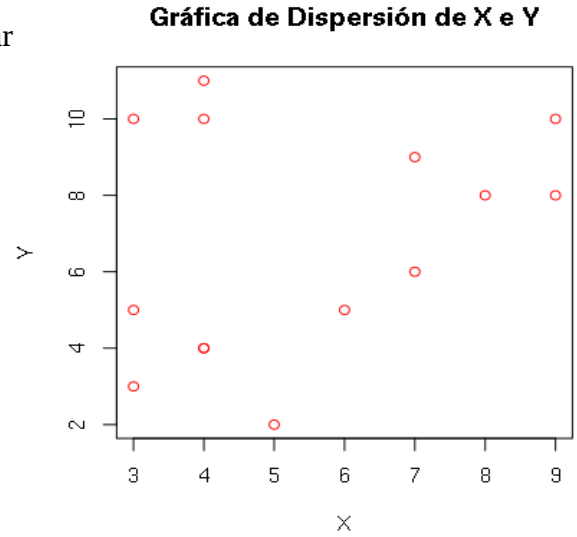
X	3	8	5	7	9	3	4	9	4	3	4	4	6	7
Y	3	8	2	9	8	5	4	10	11	10	10	4	5	6

Leemos los datos desde un archivo csv

```
> plot(ds, col="red", main="Gráfica de Dispersión de X e Y")
```

Podemos hacer una gráfica de dispersión para realizar un análisis visual de la distribución de los datos:

```
> plot(ds)
```



Clustering con K-Means

Vamos a realizar un agrupamiento utilizando el método de kmeans. Antes de realizar el

agrupamiento debemos estandarizar las observaciones, entonces:

```
> ds.est <- scale(ds)
```

Verificamos el escalado con *summary*:

```
> summary(ds.est)
       X                   Y
 Min.   :-1.0995   Min.   :-1.59964
 1st Qu.:-0.6467   1st Qu.:-0.84757
 Median :-0.4204   Median : 0.07163
 Mean   : 0.0000   Mean   : 0.00000
 3rd Qu.: 0.7114   3rd Qu.: 0.99082
 Max.   : 1.6169   Max.   : 1.40864
```

Ahora si podemos realizar el agrupamiento, vamos a realizar 4 *clusters* con valores de K de 2 a 5

```
> k2 <- kmeans(ds.est, 2)
> k3 <- kmeans(ds.est, 3)
> k4 <- kmeans(ds.est, 4)
> k5 <- kmeans(ds.est, 5)
```

Vamos a verificar cuál de los cuatro K es el correcto para nuestros datos. Para esto utilizaremos la evaluación con el coeficiente de Silueta, para esto necesitamos usar la librería **cluster**. Entonces:

```
> library(cluster)
```

Para calcular Silueta es necesario tener la matriz de distancias de nuestro conjunto de datos, en R se calcula así:

```
> distancias.ds=dist(ds.est, method="euclidean")
```

Ahora si podemos calcular el coeficiente de Silueta para cada uno de nuestros clusters:

```
> coef.silueta.k2 <- silhouette(k2$cluster, distancias.ds)
> coef.silueta.k3 <- silhouette(k3$cluster, distancias.ds)
> coef.silueta.k4 <- silhouette(k4$cluster, distancias.ds)
> coef.silueta.k5 <- silhouette(k5$cluster, distancias.ds)
```

Calidad del agrupamiento: Coeficiente de Silueta
Banchero, Santiago – Octubre de 2015

Si queremos ver los valores de s(i) para cada una de las observaciones:

```
> coef.silueta.k3
      cluster neighbor sil_width
 [1,]       1        2 0.6551780
 [2,]       3        2 0.6898605
 [3,]       1        3 0.5967739
 [4,]       3        2 0.4484816
 [5,]       3        2 0.6844245
 [6,]       1        2 0.5050283
 [7,]       1        2 0.7283637
 [8,]       3        2 0.5814772
 [9,]       2        3 0.7864536
[10,]       2        1 0.7674458
[11,]       2        3 0.8004370
[12,]       1        2 0.7283637
[13,]       1        3 0.1764532
[14,]       3        1 0.2512944
attr(,"Ordered")
[1] FALSE
attr(,"call")
silhouette.default(x = k3$cluster, dist = distancias.ds)
attr(,"class")
[1] "silhouette"
```

Si queremos ver un resumen por cluster, con la cantidad de observaciones por cada uno y el ancho de la silueta podemos utilizar *summary*:

```
> summary(coef.silueta.k3)
Silhouette of 14 units in 3 clusters from
silhouette.default(x = k3$cluster, dist = distancias.ds) :
 Cluster sizes and average silhouette widths:
        6         3         5
0.5650268 0.7847788 0.5311076
Individual silhouette widths:
```

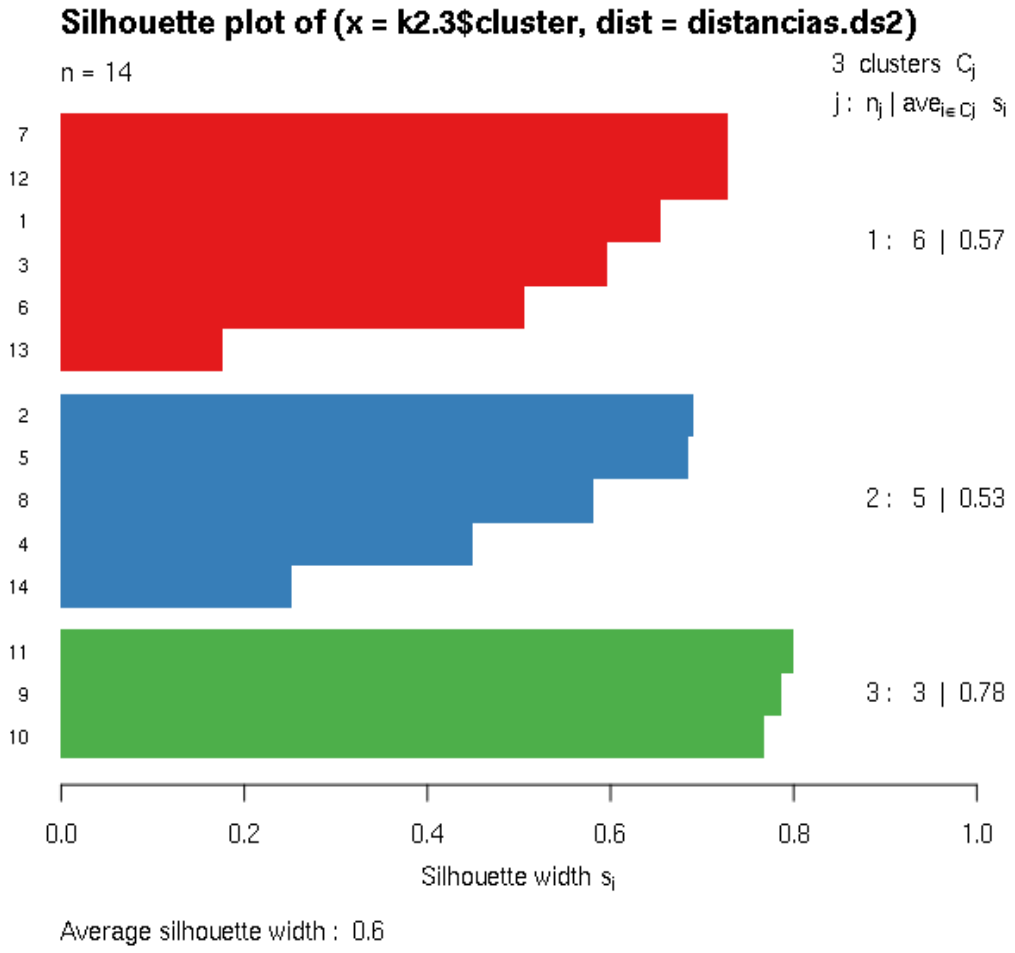
Calidad del agrupamiento: Coeficiente de Silueta

Banchero, Santiago – Octubre de 2015

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.1765  0.5241  0.6698  0.6000  0.7284  0.8004
```

Y por último, podemos obtener una representación gráfica de los coeficientes de Silueta, se utiliza el paquete *RColorBrewer*[1] para colorear cada cluster.

```
> plot(coef.silueta.k2.3, col=brewer.pal(3,"Set1"),
cex.names=0.7)
```



En el gráfico se obtiene tanto información de los clusters por separado como el ancho promedio de Silueta del agrupamiento, la cantidad de observaciones y abajo de todo el ancho de Silueta promedio. En este caso el mejor agrupamiento se consiguió con un K=3 y un ancho promedio de Silueta de 0.6. Podemos observar que solo el cluster 3 tiene una estructura sólida con valor de

1 https://cran.r-project.org/web/packages/RColorBrewer/index.html

silueta 0.78, mientras que el 1 y 2 son razonables: 0.57 y 0.53 respectivamente.

Para comparar los cuatro ajustes de silueta podemos graficar los cuatro juntos con:

```
> par(mfrow=c(2,2))
> plot(coef.silueta.k2)
> plot(coef.silueta.k3)
> plot(coef.silueta.k4)
> plot(coef.silueta.k5)
```

Referencias

Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, *20*, 53-65.

Santiago Carmona. (s. f.). TP Data mining - Introducción al clustering en bioinformática. Recuperado 23 de octubre de 2015, a partir de http://genoma.unsam.edu.ar/trac/docencia/wiki/Bioinformatica/Guias/DataMining