PAM: Partitioning Around Medoids

- Unlike the hierarchical clustering methods, techniques like k-means cluster analysis (available through the `kmeans` function) or partitioning around ***mediods*** require that we specify the number of clusters that will be formed in advance.

- This method is available through the `pam` function in the ***cluster*** library.

- `pam` offers some additional diagnostic information about a clustering solution, and provides a nice example of an alternative technique to hierarchical clustering.

- You can pass `pam` a data frame or a distance matrix, as well as the number of clusters you wish to form.

- Let's look at the three cluster solution produced by `pam`:

```
#install.packages("cluster")
library(cluster)

cars.pam = pam(cars,3)
```

- Use the `summary()`, names() , `class()` and `str()` commands to analyse the output of the procedure.

```
> names(cars.pam)
 [1] "medoids"    "id.med"     "clustering" "objective"
 [5] "isolation"  "clusinfo"   "silinfo"    "diss"
 [9] "call"       "data"
```

Results of Procedure

- The medoids (centres in other words) of each of the three clusters

```
> cars.pam$medoids
   Country Car  MPG Weight Drive_Ratio Horsepower
30       7  20 18.2  3.830        2.45        135
37       5  26 29.5  2.135        3.05         68
19       5  16 22.0  2.815        3.70         97
   Displacement Cylinders
30          318         8
37           98         4
19          146         6
>
> cars.pam$id.med
[1] 30 37 19
```

- The cluster assignment for each of the 38 cases.

```
> cars.pam$clustering
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19
 1  2  2  2  2  2  2  2  2  2  3  3  3  3  3  3  3  3  3
20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38
 3  3  3  2  1  3  1  3  1  1  1  1  1  3  3  3  3  2  1
> table(cars.pam$clustering)

 1  2  3
 9 11 18
```

- Summary statistics on the dimensionality of each cluster.

```
> cars.pam$clusinfo
     size max_diss  av_diss  diameter separation
```

```
[1,]    9 64.44583 33.85620 106.90893   31.24501
[2,]   11 18.10320 13.25804  31.75348   23.44201
[3,]   18 86.00128 34.71620 113.97993   23.44201
```

- Information used to construct the silohouette plot (forthcoming) is contained in `cars.pam$silinfo`

Comparison of Results

- Let's see if this solution agrees with the `hclust` solution.
- Since pam only looks at one cluster solution at a time, we don't need to use the `cutree` function as we did with hclust
- The cluster memberships are stored in the `clustering` component of the `pam` object (see below).
- We can use `table()` to compare the results of the `hclust` and `pam` solutions:

```
    > cars.pam$clustering
     [1] 1 2 2 2 2 2 2 2 2 2 1 3 3 3 1 1 1 3 3 1 3 1 2
    [24] 1 3 1 3 1 1 1 1 1 1 3 3 3 2 1 1
    >
    > table(groups.3,cars.pam$clustering)
    groups.3  1  2  3
           1  8  0  0
           2  0 19  1
           3  0  0 10
```

- The solutions seem to agree, except for 1 observations that `hclust` put in group 2 and `pam` put in group 3. Which observations was it?

```
> cars$Car[groups.3 != cars.pam$clustering]
[1] Audi 5000
```

- One novel feature of `pam` is that it finds observations from the original data that are typical of each cluster in the sense that they are closest to the center of the cluster.

- The indexes of the medoids are stored in the `id.med` component of the `pam` object, so we can use that component as a subscript into the vector of car names to see which ones were selected:

```
> cars$Car[cars.pam$id.med]
> cars$Car[cars.pam$id.med]
[1] Dodge St Regis    Dodge Omni        Ford Mustang Ghia
```

Exercises

Try out the PAM procedure on the European Jobs Data Set, with a 3 and 4 cluster solution respectively.

- What are the medoids for the 3 cluster solution?

- What are the medoids for the 4 cluster solution?

- For both solutions, how many are in each cluster?

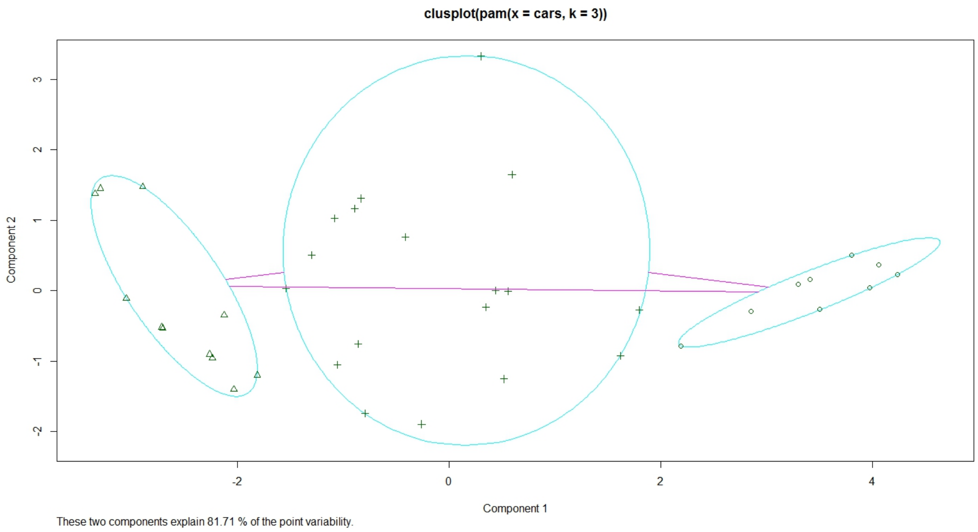
Silhouette Plot

- Another feature available with `pam` is a plot known as a ***silhouette plot***.
- First, a measure is calculated for each observation to see how well it fits into the cluster that it's been assigned to. This is done by comparing how close the object is to other objects in its own cluster with how close it is to objects in other clusters.
- Values near one mean that the observation is well placed in its cluster; values near 0 mean that it's likely that an observation might really belong in some other cluster. Within each cluster, the value for this measure is displayed from smallest to largest.
- If the silhouette plot shows values close to one for each observation, the fit was good; if there are many observations closer to zero, it's an indication that the fit was not good.
- The sihouette plot is very useful in locating groups in a cluster analysis that may not be doing a good job; in turn this information can be used to help select the proper number of clusters.
- For the current example, here's the silhouette plot for the three cluster `pam` solution, produced by the command

Summary
Silhouette Plot shows for each cluster:

- the number of plots per cluster = number of horizontal lines, also given in the right hand column,
- the means similarity of each plot to its own cluster minus the mean similarity to the next most similar cluster (given by the length of the lines) with the mean in the right hand column, and
- the average silhouette width

```
clusplot(cars.pam)
plot(cars.pam) # both plots Sequentially
```



The plot indicates that there is a good structure to the clusters, with most observations seeming to belong to the cluster that they're in.

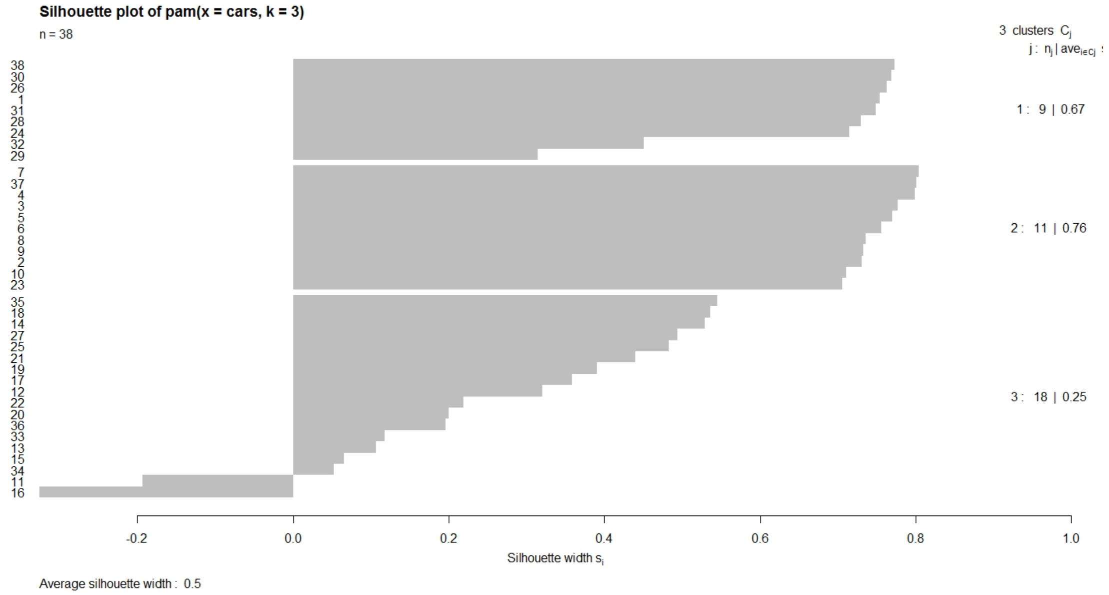
Average Silhouette Width

There is a summary measure at the bottom of the plot labeled ”***Average Silhouette Width***”. This table shows how to use the value:

Range	Interpretation
0.71-1.0	A strong structure has been found
0.51-0.70	A reasonable structure has been found
0.26-0.50	The structure is weak and could be artificial
$<$ 0.25	No substantial structure has been found

Exercises

Try out the PAM procedure on the ***cars*** Data Set, with a 4 and 5 cluster solution respectively.



- What is the average silouette width for the 4 cluster solution?
- What is the average silouette width for the 5 cluster solution?

Try out the PAM procedure on the ***European Jobs*** Data Set, with a 4 and 5 cluster solution respectively.

- What is the average silouette width for the 4 cluster solution?
- What is the average silouette width for the 5 cluster solution?

Creating Silhouette Plots for Other Solutions

- To create a silhouette plot for a particular solution derived from a hierarchical cluster analysis, the `silhouette` function can be used.

- This function takes the appropriate output from cutree along with the distance matrix used for the clustering.

- So to produce a silhouette plot for our 4 group hierarchical cluster (not shown), we could use the following statements:

```
plot(silhouette(cutree(cars.hclust,4),cars))
```