M4L2\_Assignment

Xueyi Fan

June 13, 2016

# Assignment:

* First, go to the UC Irvine Machine Learning Repository at <https://archive.ics.uci.edu/ml/> and find a dataset for clustering. Note that every student MUST use a different dataset so you MUST get approved for which data you are going to use. You will use this dataset for this module on unsupervised learning and the next on supervised learning.
* Next, cluster some of your data using k-means, PAM and hierarchical clustering.
* Finally, answer the following questions:

1. How did you choose a k for k-means?
2. Evaluate the model performance. How do the clustering approaches compare on the same data?
3. Generate and plot confusion matrices for the k-means and PAM. What do they tell you?
4. Generate centroid plots against the 1st two discriminant functions for k-means and PAM. What do they tell you?
5. Generate silhouette plots for PAM. What do they tell you?
6. For the hierarchical clustering use all linkage methods (Single Link, Complete Link, Average Link, Centroid and Minimum energy clustering) and generate dendograms. How do they compare on the same data?
7. For the hierarchical clustering use both agglomerative and divisive clustering with a linkage method of your choice and generate dendograms. How do they compare on the same data?
8. For the hierarchical clustering use centroid clustering and squared Euclidean distance and generate dendograms. How do they compare on the same data?

Loading the packages:

library('ggplot2')  
library('cluster')

## Warning: package 'cluster' was built under R version 3.2.5

library('amap')  
library('useful')

Here, I choose the [Breast Cancer Wisconsin (Prognostic) data set](https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/)

Loading the data:

data\_url <- 'https://archive.ics.uci.edu/ml/machine-learning-databases/breast-cancer-wisconsin/wpbc.data'  
  
cancer\_data <- read.table(url**(**data\_url), sep = ',')  
  
  
names(cancer\_data) <- c('ID number', 'Outcome','Time','radius\_mean','texure\_mean','perimeter\_mean','area\_mean','smoothness\_mean','compactness\_mean','concavity\_mean','concave\_points\_mean','symmetry\_mean','fractal\_dimension\_mean', 'radius\_SE','texure\_SE','perimeter\_SE','area\_SE','smoothness\_SE','compactness\_SE','concavity\_SE','concave\_points\_SE','symmetry\_SE','fractal\_dimension\_SE','radius\_worst','texure\_worst','perimeter\_worst','area\_worst','smoothness\_worst','compactness\_worst','concavity\_worst','concave\_points\_worst','symmetry\_worst','fractal\_dimension\_worst','tumor\_size','lymph\_node\_status')   
cancer <- data.frame(cancer\_data[,3:34])  
head(cancer)

## Time radius\_mean texure\_mean perimeter\_mean area\_mean smoothness\_mean  
## 1 31 18.02 27.60 117.50 1013.0 0.09489  
## 2 61 17.99 10.38 122.80 1001.0 0.11840  
## 3 116 21.37 17.44 137.50 1373.0 0.08836  
## 4 123 11.42 20.38 77.58 386.1 0.14250  
## 5 27 20.29 14.34 135.10 1297.0 0.10030  
## 6 77 12.75 15.29 84.60 502.7 0.11890  
## compactness\_mean concavity\_mean concave\_points\_mean symmetry\_mean  
## 1 0.1036 0.1086 0.07055 0.1865  
## 2 0.2776 0.3001 0.14710 0.2419  
## 3 0.1189 0.1255 0.08180 0.2333  
## 4 0.2839 0.2414 0.10520 0.2597  
## 5 0.1328 0.1980 0.10430 0.1809  
## 6 0.1569 0.1664 0.07666 0.1995  
## fractal\_dimension\_mean radius\_SE texure\_SE perimeter\_SE area\_SE  
## 1 0.06333 0.6249 1.8900 3.972 71.55  
## 2 0.07871 1.0950 0.9053 8.589 153.40  
## 3 0.06010 0.5854 0.6105 3.928 82.15  
## 4 0.09744 0.4956 1.1560 3.445 27.23  
## 5 0.05883 0.7572 0.7813 5.438 94.44  
## 6 0.07164 0.3877 0.7402 2.999 30.85  
## smoothness\_SE compactness\_SE concavity\_SE concave\_points\_SE symmetry\_SE  
## 1 0.004433 0.01421 0.03233 0.009854 0.01694  
## 2 0.006399 0.04904 0.05373 0.015870 0.03003  
## 3 0.006167 0.03449 0.03300 0.018050 0.03094  
## 4 0.009110 0.07458 0.05661 0.018670 0.05963  
## 5 0.011490 0.02461 0.05688 0.018850 0.01756  
## 6 0.007775 0.02987 0.04561 0.013570 0.01774  
## fractal\_dimension\_SE radius\_worst texure\_worst perimeter\_worst  
## 1 0.003495 21.63 37.08 139.70  
## 2 0.006193 25.38 17.33 184.60  
## 3 0.005039 24.90 20.98 159.10  
## 4 0.009208 14.91 26.50 98.87  
## 5 0.005115 22.54 16.67 152.20  
## 6 0.005114 15.51 20.37 107.30  
## area\_worst smoothness\_worst compactness\_worst concavity\_worst  
## 1 1436.0 0.1195 0.1926 0.3140  
## 2 2019.0 0.1622 0.6656 0.7119  
## 3 1949.0 0.1188 0.3449 0.3414  
## 4 567.7 0.2098 0.8663 0.6869  
## 5 1575.0 0.1374 0.2050 0.4000  
## 6 733.2 0.1706 0.4196 0.5999  
## concave\_points\_worst symmetry\_worst fractal\_dimension\_worst tumor\_size  
## 1 0.1170 0.2677 0.08113 5.0  
## 2 0.2654 0.4601 0.11890 3.0  
## 3 0.2032 0.4334 0.09067 2.5  
## 4 0.2575 0.6638 0.17300 2.0  
## 5 0.1625 0.2364 0.07678 3.5  
## 6 0.1709 0.3485 0.11790 2.5

str(cancer)

## 'data.frame': 198 obs. of 32 variables:  
## $ Time : int 31 61 116 123 27 77 60 77 119 76 ...  
## $ radius\_mean : num 18 18 21.4 11.4 20.3 ...  
## $ texure\_mean : num 27.6 10.4 17.4 20.4 14.3 ...  
## $ perimeter\_mean : num 117.5 122.8 137.5 77.6 135.1 ...  
## $ area\_mean : num 1013 1001 1373 386 1297 ...  
## $ smoothness\_mean : num 0.0949 0.1184 0.0884 0.1425 0.1003 ...  
## $ compactness\_mean : num 0.104 0.278 0.119 0.284 0.133 ...  
## $ concavity\_mean : num 0.109 0.3 0.126 0.241 0.198 ...  
## $ concave\_points\_mean : num 0.0706 0.1471 0.0818 0.1052 0.1043 ...  
## $ symmetry\_mean : num 0.186 0.242 0.233 0.26 0.181 ...  
## $ fractal\_dimension\_mean : num 0.0633 0.0787 0.0601 0.0974 0.0588 ...  
## $ radius\_SE : num 0.625 1.095 0.585 0.496 0.757 ...  
## $ texure\_SE : num 1.89 0.905 0.611 1.156 0.781 ...  
## $ perimeter\_SE : num 3.97 8.59 3.93 3.44 5.44 ...  
## $ area\_SE : num 71.5 153.4 82.2 27.2 94.4 ...  
## $ smoothness\_SE : num 0.00443 0.0064 0.00617 0.00911 0.01149 ...  
## $ compactness\_SE : num 0.0142 0.049 0.0345 0.0746 0.0246 ...  
## $ concavity\_SE : num 0.0323 0.0537 0.033 0.0566 0.0569 ...  
## $ concave\_points\_SE : num 0.00985 0.01587 0.01805 0.01867 0.01885 ...  
## $ symmetry\_SE : num 0.0169 0.03 0.0309 0.0596 0.0176 ...  
## $ fractal\_dimension\_SE : num 0.00349 0.00619 0.00504 0.00921 0.00511 ...  
## $ radius\_worst : num 21.6 25.4 24.9 14.9 22.5 ...  
## $ texure\_worst : num 37.1 17.3 21 26.5 16.7 ...  
## $ perimeter\_worst : num 139.7 184.6 159.1 98.9 152.2 ...  
## $ area\_worst : num 1436 2019 1949 568 1575 ...  
## $ smoothness\_worst : num 0.119 0.162 0.119 0.21 0.137 ...  
## $ compactness\_worst : num 0.193 0.666 0.345 0.866 0.205 ...  
## $ concavity\_worst : num 0.314 0.712 0.341 0.687 0.4 ...  
## $ concave\_points\_worst : num 0.117 0.265 0.203 0.258 0.163 ...  
## $ symmetry\_worst : num 0.268 0.46 0.433 0.664 0.236 ...  
## $ fractal\_dimension\_worst: num 0.0811 0.1189 0.0907 0.173 0.0768 ...  
## $ tumor\_size : num 5 3 2.5 2 3.5 2.5 1.5 4 2 6 ...

In the breast cancer dataset, there are 35 columns. Two of them are factors and others are number. Here I use column 3-34 as the clustering data.

## Cluster data using k-means

k <- 6  
cancer.6.kmeans <- kmeans(cancer, centers = 6)  
cancer.6.kmeans

## K-means clustering with 6 clusters of sizes 41, 47, 25, 7, 41, 37  
##   
## Cluster means:  
## Time radius\_mean texure\_mean perimeter\_mean area\_mean  
## 1 53.90244 15.34317 22.15610 100.92195 730.8610  
## 2 44.02128 17.53426 22.72660 115.21277 954.9617  
## 3 33.12000 21.33120 23.14440 141.49200 1420.0640  
## 4 33.85714 24.18143 23.14714 160.81429 1840.2857  
## 5 39.70732 19.54659 22.00561 129.32683 1187.6463  
## 6 61.64865 13.25676 21.38459 87.11892 544.3919  
## smoothness\_mean compactness\_mean concavity\_mean concave\_points\_mean  
## 1 0.1018995 0.1324161 0.1352556 0.06850171  
## 2 0.1004704 0.1322753 0.1412600 0.08417064  
## 3 0.1026120 0.1629328 0.2092568 0.11751600  
## 4 0.1019086 0.1661000 0.2271571 0.14063714  
## 5 0.1011437 0.1501956 0.1814512 0.10238122  
## 6 0.1082535 0.1406546 0.1213608 0.06208081  
## symmetry\_mean fractal\_dimension\_mean radius\_SE texure\_SE perimeter\_SE  
## 1 0.1902366 0.06343878 0.4153951 1.148963 3.002098  
## 2 0.1902787 0.06065745 0.5993106 1.352626 4.146787  
## 3 0.1957320 0.06056000 0.9187800 1.267024 6.394760  
## 4 0.1830429 0.05899143 1.1130571 1.084400 7.587143  
## 5 0.1944561 0.06102293 0.7159878 1.282176 5.119390  
## 6 0.1966270 0.06851135 0.3823622 1.293097 2.748892  
## area\_SE smoothness\_SE compactness\_SE concavity\_SE concave\_points\_SE  
## 1 39.81244 0.006922707 0.03041685 0.04064439 0.01399839  
## 2 65.47362 0.006628426 0.02790338 0.03741404 0.01523134  
## 3 126.25840 0.006582000 0.03433880 0.04619840 0.01656240  
## 4 179.61429 0.005485286 0.02511857 0.03360143 0.01399857  
## 5 88.23463 0.006598854 0.03351561 0.04547537 0.01646507  
## 6 31.46865 0.007296811 0.03271541 0.03753838 0.01385749  
## symmetry\_SE fractal\_dimension\_SE radius\_worst texure\_worst  
## 1 0.01957212 0.003845878 18.10537 30.52829  
## 2 0.01968155 0.003809532 20.86319 30.33426  
## 3 0.02033200 0.003972520 26.69280 30.04440  
## 4 0.01691429 0.003087714 32.21000 31.30286  
## 5 0.02231756 0.004042976 23.58073 29.41707  
## 6 0.02163924 0.004486189 15.67108 30.10378  
## perimeter\_worst area\_worst smoothness\_worst compactness\_worst  
## 1 121.1317 1007.3707 0.1447805 0.3634271  
## 2 137.7681 1328.2553 0.1388538 0.3114383  
## 3 178.1560 2189.6400 0.1388444 0.3779480  
## 4 215.7714 3182.4286 0.1413571 0.3625571  
## 5 158.6244 1700.0732 0.1393780 0.3752951  
## 6 104.8503 749.4757 0.1583532 0.4156311  
## concavity\_worst concave\_points\_worst symmetry\_worst  
## 1 0.4260780 0.1597993 0.3280537  
## 2 0.3850896 0.1700191 0.3079447  
## 3 0.4910280 0.2120352 0.3152200  
## 4 0.4778143 0.2293571 0.2943286  
## 5 0.4641293 0.1933561 0.3230756  
## 6 0.4390697 0.1627378 0.3492838  
## fractal\_dimension\_worst tumor\_size  
## 1 0.09216610 2.019512  
## 2 0.08372064 3.080851  
## 3 0.08488680 3.164000  
## 4 0.08208571 2.928571  
## 5 0.08815317 3.397561  
## 6 0.10700649 2.629730  
##   
## Clustering vector:  
## [1] 2 5 3 6 5 6 5 6 6 6 1 2 1 1 1 2 1 3 3 2 1 2 2 2 6 2 5 2 2 6 6 6 6 6 5  
## [36] 6 6 5 6 1 3 1 1 6 1 5 6 5 2 1 5 6 1 6 1 2 2 5 3 5 5 1 3 6 1 2 6 2 5 3  
## [71] 3 2 2 1 1 4 3 2 1 2 6 6 6 6 5 1 3 1 1 5 1 6 6 3 3 1 6 2 5 4 2 2 2 5 6  
## [106] 5 1 2 1 5 2 1 2 4 3 2 5 5 1 6 3 5 2 5 5 1 2 5 4 5 1 4 1 5 5 3 3 6 6 2  
## [141] 3 2 1 5 2 5 5 6 2 2 2 3 2 2 1 5 1 2 5 5 6 4 1 1 2 5 4 2 1 5 3 5 1 6 2  
## [176] 3 2 2 3 1 1 2 1 5 2 6 3 3 6 5 5 5 6 3 1 2 3 2  
##   
## Within cluster sum of squares by cluster:  
## [1] 561985.9 1049920.7 2281247.0 1466252.3 1364867.2 693009.7  
## (between\_SS / total\_SS = 92.0 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

## Cluster data using PAM

k<-6  
cancer.6.pam <- pam(cancer,k,keep.diss=TRUE, keep.data= TRUE)  
cancer.6.pam

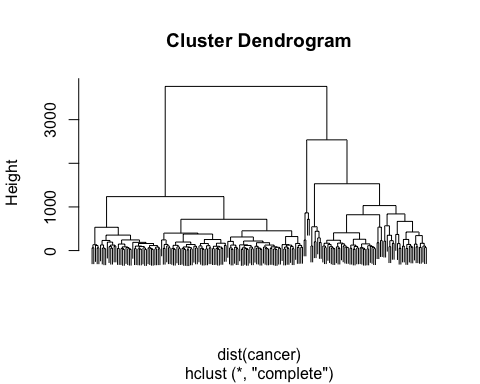
## Medoids:  
## ID Time radius\_mean texure\_mean perimeter\_mean area\_mean  
## [1,] 149 13 17.27 25.42 112.40 928.8  
## [2,] 77 53 21.09 26.57 142.70 1311.0  
## [3,] 32 76 13.48 20.82 88.40 559.2  
## [4,] 166 41 19.89 20.26 130.50 1214.0  
## [5,] 164 38 15.05 19.07 97.26 701.9  
## [6,] 129 24 23.51 24.27 155.10 1747.0  
## smoothness\_mean compactness\_mean concavity\_mean concave\_points\_mean  
## [1,] 0.08331 0.11090 0.12040 0.05736  
## [2,] 0.11410 0.28320 0.24870 0.14960  
## [3,] 0.10160 0.12550 0.10630 0.05439  
## [4,] 0.10370 0.13100 0.14110 0.09431  
## [5,] 0.09215 0.08597 0.07486 0.04335  
## [6,] 0.10690 0.12830 0.23080 0.14100  
## symmetry\_mean fractal\_dimension\_mean radius\_SE texure\_SE perimeter\_SE  
## [1,] 0.1467 0.05407 0.5100 1.6790 3.283  
## [2,] 0.2395 0.07398 0.6298 0.7629 4.414  
## [3,] 0.1720 0.06419 0.2130 0.5914 1.545  
## [4,] 0.1802 0.06188 0.5079 0.8737 3.654  
## [5,] 0.1561 0.05915 0.3860 1.1980 2.630  
## [6,] 0.1797 0.05506 1.0090 0.9245 6.462  
## area\_SE smoothness\_SE compactness\_SE concavity\_SE concave\_points\_SE  
## [1,] 58.38 0.008109 0.04308 0.04942 0.017420  
## [2,] 81.46 0.004253 0.04759 0.03872 0.015670  
## [3,] 18.52 0.005367 0.02239 0.03049 0.012620  
## [4,] 59.70 0.005089 0.02303 0.03052 0.011780  
## [5,] 38.49 0.004952 0.01630 0.02967 0.009423  
## [6,] 164.10 0.006292 0.01971 0.03582 0.013010  
## symmetry\_SE fractal\_dimension\_SE radius\_worst texure\_worst  
## [1,] 0.01594 0.003739 20.38 35.46  
## [2,] 0.01798 0.005295 26.68 33.48  
## [3,] 0.01377 0.003187 15.53 26.02  
## [4,] 0.01057 0.003391 23.73 25.23  
## [5,] 0.01152 0.001718 17.58 28.06  
## [6,] 0.01479 0.003118 30.67 30.73  
## perimeter\_worst area\_worst smoothness\_worst compactness\_worst  
## [1,] 132.8 1284.0 0.1436 0.4122  
## [2,] 176.5 2089.0 0.1491 0.7584  
## [3,] 107.3 740.4 0.1610 0.4225  
## [4,] 160.5 1646.0 0.1417 0.3309  
## [5,] 113.8 967.0 0.1246 0.2101  
## [6,] 202.4 2906.0 0.1515 0.2678  
## concavity\_worst concave\_points\_worst symmetry\_worst  
## [1,] 0.5036 0.1739 0.2500  
## [2,] 0.6780 0.2903 0.4098  
## [3,] 0.5030 0.2258 0.2807  
## [4,] 0.4185 0.1613 0.2549  
## [5,] 0.2866 0.1120 0.2282  
## [6,] 0.4819 0.2089 0.2593  
## fractal\_dimension\_worst tumor\_size  
## [1,] 0.07944 9.0  
## [2,] 0.12840 4.0  
## [3,] 0.10710 2.5  
## [4,] 0.09136 8.0  
## [5,] 0.06954 2.5  
## [6,] 0.07738 3.5  
## Clustering vector:  
## [1] 1 2 2 3 4 3 4 5 3 3 1 1 5 5 5 1 5 6 2 1 5 1 1 1 3 1 4 1 1 3 3 3 3 3 4  
## [36] 3 3 4 3 5 2 5 5 3 5 4 3 4 1 5 4 3 5 3 5 1 1 4 2 4 4 1 2 3 5 1 3 1 4 2  
## [71] 2 1 1 5 5 6 2 1 5 1 3 3 3 3 4 5 2 5 5 4 5 3 3 2 2 5 3 1 2 6 1 4 1 2 3  
## [106] 2 5 1 5 4 1 5 1 6 2 1 4 4 5 3 2 4 1 4 2 5 1 4 6 4 5 6 5 4 4 2 2 3 3 1  
## [141] 2 1 5 4 1 4 4 5 1 1 1 2 4 1 5 4 5 1 4 4 3 6 5 5 4 4 6 1 5 4 2 4 1 3 1  
## [176] 2 1 1 2 5 5 1 5 4 1 3 6 6 3 4 4 4 3 2 5 1 2 1  
## Objective function:  
## build swap   
## 163.9977 160.1383   
##   
## Available components:  
## [1] "medoids" "id.med" "clustering" "objective" "isolation"   
## [6] "clusinfo" "silinfo" "diss" "call" "data"

## Cluster data using hierarchical clustering

cancer.h.clust <- hclust(d=dist(cancer))  
cancer.h.clust

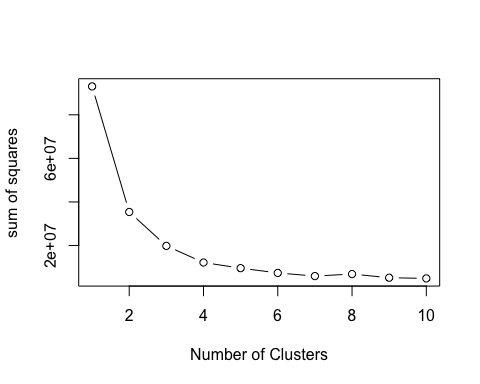
##   
## Call:  
## hclust(d = dist(cancer))  
##   
## Cluster method : complete   
## Distance : euclidean   
## Number of objects: 198

plot(cancer.h.clust,labels=F)

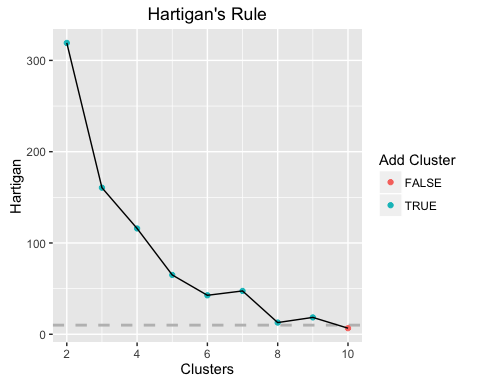


### 1. How did you choose a k for k-means?

#plot the sum of squares responding to each cluster  
sos <- rep(NA,10)  
for (i in 1:10){  
 sos[i] <- kmeans(cancer, centers=i)$tot.withinss  
}   
  
plot(sos[c(1:10)], type="b",xlab="Number of Clusters", ylab="sum of squares")



#Hartigan's rule  
best<- FitKMeans(cancer, max.cluster =10, seed = 66)  
PlotHartigan(best)



First, I plot the sum of squares against different k. The value of sos is decreasing as the k increast. Specially, the sos value show quickly decreasing at the range of 2 to 6.

Second, I also use Hartigan's rule to determine the k. The rule is that if the value of hartigan is greater than 10, it is justifiable to add the extra group. According to the Hartigan's Rule plot, it is accepable that k is smaller than 10. And the value is decreasing rapidly before k is 6.

So I chhose k=6 for k-means

### 2. Evaluate the model performance. How do the clustering approaches compare on the same data?

There are many ways evaluating the model performance. For example, using correlation to measure cluster validity, using similarity matrix like Davies-Bouldin index, Dunn index, Silhouette coefficient, purity, entropy, kullback-leibler divergence, jaccard similarity coefficientr and rand index.

Here, I use Davies-Bouldin index to evalute the model performance.

library("clusterSim")

## Loading required package: MASS

#evaluate the k-means model performance  
  
#look at the size  
cancer.6.kmeans$size

## [1] 41 47 25 7 41 37

#look at the cluster centers  
cancer.6.kmeans$centers

## Time radius\_mean texure\_mean perimeter\_mean area\_mean  
## 1 53.90244 15.34317 22.15610 100.92195 730.8610  
## 2 44.02128 17.53426 22.72660 115.21277 954.9617  
## 3 33.12000 21.33120 23.14440 141.49200 1420.0640  
## 4 33.85714 24.18143 23.14714 160.81429 1840.2857  
## 5 39.70732 19.54659 22.00561 129.32683 1187.6463  
## 6 61.64865 13.25676 21.38459 87.11892 544.3919  
## smoothness\_mean compactness\_mean concavity\_mean concave\_points\_mean  
## 1 0.1018995 0.1324161 0.1352556 0.06850171  
## 2 0.1004704 0.1322753 0.1412600 0.08417064  
## 3 0.1026120 0.1629328 0.2092568 0.11751600  
## 4 0.1019086 0.1661000 0.2271571 0.14063714  
## 5 0.1011437 0.1501956 0.1814512 0.10238122  
## 6 0.1082535 0.1406546 0.1213608 0.06208081  
## symmetry\_mean fractal\_dimension\_mean radius\_SE texure\_SE perimeter\_SE  
## 1 0.1902366 0.06343878 0.4153951 1.148963 3.002098  
## 2 0.1902787 0.06065745 0.5993106 1.352626 4.146787  
## 3 0.1957320 0.06056000 0.9187800 1.267024 6.394760  
## 4 0.1830429 0.05899143 1.1130571 1.084400 7.587143  
## 5 0.1944561 0.06102293 0.7159878 1.282176 5.119390  
## 6 0.1966270 0.06851135 0.3823622 1.293097 2.748892  
## area\_SE smoothness\_SE compactness\_SE concavity\_SE concave\_points\_SE  
## 1 39.81244 0.006922707 0.03041685 0.04064439 0.01399839  
## 2 65.47362 0.006628426 0.02790338 0.03741404 0.01523134  
## 3 126.25840 0.006582000 0.03433880 0.04619840 0.01656240  
## 4 179.61429 0.005485286 0.02511857 0.03360143 0.01399857  
## 5 88.23463 0.006598854 0.03351561 0.04547537 0.01646507  
## 6 31.46865 0.007296811 0.03271541 0.03753838 0.01385749  
## symmetry\_SE fractal\_dimension\_SE radius\_worst texure\_worst  
## 1 0.01957212 0.003845878 18.10537 30.52829  
## 2 0.01968155 0.003809532 20.86319 30.33426  
## 3 0.02033200 0.003972520 26.69280 30.04440  
## 4 0.01691429 0.003087714 32.21000 31.30286  
## 5 0.02231756 0.004042976 23.58073 29.41707  
## 6 0.02163924 0.004486189 15.67108 30.10378  
## perimeter\_worst area\_worst smoothness\_worst compactness\_worst  
## 1 121.1317 1007.3707 0.1447805 0.3634271  
## 2 137.7681 1328.2553 0.1388538 0.3114383  
## 3 178.1560 2189.6400 0.1388444 0.3779480  
## 4 215.7714 3182.4286 0.1413571 0.3625571  
## 5 158.6244 1700.0732 0.1393780 0.3752951  
## 6 104.8503 749.4757 0.1583532 0.4156311  
## concavity\_worst concave\_points\_worst symmetry\_worst  
## 1 0.4260780 0.1597993 0.3280537  
## 2 0.3850896 0.1700191 0.3079447  
## 3 0.4910280 0.2120352 0.3152200  
## 4 0.4778143 0.2293571 0.2943286  
## 5 0.4641293 0.1933561 0.3230756  
## 6 0.4390697 0.1627378 0.3492838  
## fractal\_dimension\_worst tumor\_size  
## 1 0.09216610 2.019512  
## 2 0.08372064 3.080851  
## 3 0.08488680 3.164000  
## 4 0.08208571 2.928571  
## 5 0.08815317 3.397561  
## 6 0.10700649 2.629730

print(index.DB(cancer, cancer.6.kmeans$cluster, centrotypes="centroids"))

## $DB  
## [1] 0.805144  
##   
## $r  
## [1] 0.7955779 0.7544012 0.8909922 0.7033227 0.8909922 0.7955779  
##   
## $R  
## [,1] [,2] [,3] [,4] [,5] [,6]  
## [1,] Inf 0.6782332 0.3052372 0.2347484 0.3597325 0.7955779  
## [2,] 0.6782332 Inf 0.4597738 0.2947380 0.7544012 0.4021340  
## [3,] 0.3052372 0.4597738 Inf 0.7033227 0.8909922 0.2595828  
## [4,] 0.2347484 0.2947380 0.7033227 Inf 0.3942649 0.2151010  
## [5,] 0.3597325 0.7544012 0.8909922 0.3942649 Inf 0.2773061  
## [6,] 0.7955779 0.4021340 0.2595828 0.2151010 0.2773061 Inf  
##   
## $d  
## 1 2 3 4 5 6  
## 1 0.0000 392.9891 1373.2035 2448.365 832.6486 319.1823  
## 2 392.9891 0.0000 982.0860 2059.912 439.9720 711.9991  
## 3 1373.2035 982.0860 0.0000 1080.228 543.8093 1690.9192  
## 4 2448.3650 2059.9123 1080.2281 0.000 1623.5962 2763.9602  
## 5 832.6486 439.9720 543.8093 1623.596 0.0000 1151.4765  
## 6 319.1823 711.9991 1690.9192 2763.960 1151.4765 0.0000  
##   
## $S  
## [1] 117.0768 149.4615 302.0760 457.6730 182.4540 136.8576  
##   
## $centers  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 53.90244 15.34317 22.15610 100.92195 730.8610 0.1018995 0.1324161  
## [2,] 44.02128 17.53426 22.72660 115.21277 954.9617 0.1004704 0.1322753  
## [3,] 33.12000 21.33120 23.14440 141.49200 1420.0640 0.1026120 0.1629328  
## [4,] 33.85714 24.18143 23.14714 160.81429 1840.2857 0.1019086 0.1661000  
## [5,] 39.70732 19.54659 22.00561 129.32683 1187.6463 0.1011437 0.1501956  
## [6,] 61.64865 13.25676 21.38459 87.11892 544.3919 0.1082535 0.1406546  
## [,8] [,9] [,10] [,11] [,12] [,13] [,14]  
## [1,] 0.1352556 0.06850171 0.1902366 0.06343878 0.4153951 1.148963 3.002098  
## [2,] 0.1412600 0.08417064 0.1902787 0.06065745 0.5993106 1.352626 4.146787  
## [3,] 0.2092568 0.11751600 0.1957320 0.06056000 0.9187800 1.267024 6.394760  
## [4,] 0.2271571 0.14063714 0.1830429 0.05899143 1.1130571 1.084400 7.587143  
## [5,] 0.1814512 0.10238122 0.1944561 0.06102293 0.7159878 1.282176 5.119390  
## [6,] 0.1213608 0.06208081 0.1966270 0.06851135 0.3823622 1.293097 2.748892  
## [,15] [,16] [,17] [,18] [,19] [,20]  
## [1,] 39.81244 0.006922707 0.03041685 0.04064439 0.01399839 0.01957212  
## [2,] 65.47362 0.006628426 0.02790338 0.03741404 0.01523134 0.01968155  
## [3,] 126.25840 0.006582000 0.03433880 0.04619840 0.01656240 0.02033200  
## [4,] 179.61429 0.005485286 0.02511857 0.03360143 0.01399857 0.01691429  
## [5,] 88.23463 0.006598854 0.03351561 0.04547537 0.01646507 0.02231756  
## [6,] 31.46865 0.007296811 0.03271541 0.03753838 0.01385749 0.02163924  
## [,21] [,22] [,23] [,24] [,25] [,26] [,27]  
## [1,] 0.003845878 18.10537 30.52829 121.1317 1007.3707 0.1447805 0.3634271  
## [2,] 0.003809532 20.86319 30.33426 137.7681 1328.2553 0.1388538 0.3114383  
## [3,] 0.003972520 26.69280 30.04440 178.1560 2189.6400 0.1388444 0.3779480  
## [4,] 0.003087714 32.21000 31.30286 215.7714 3182.4286 0.1413571 0.3625571  
## [5,] 0.004042976 23.58073 29.41707 158.6244 1700.0732 0.1393780 0.3752951  
## [6,] 0.004486189 15.67108 30.10378 104.8503 749.4757 0.1583532 0.4156311  
## [,28] [,29] [,30] [,31] [,32]  
## [1,] 0.4260780 0.1597993 0.3280537 0.09216610 2.019512  
## [2,] 0.3850896 0.1700191 0.3079447 0.08372064 3.080851  
## [3,] 0.4910280 0.2120352 0.3152200 0.08488680 3.164000  
## [4,] 0.4778143 0.2293571 0.2943286 0.08208571 2.928571  
## [5,] 0.4641293 0.1933561 0.3230756 0.08815317 3.397561  
## [6,] 0.4390697 0.1627378 0.3492838 0.10700649 2.629730

#evaluate the PAM model performance  
  
#look at the medoids  
cancer.6.pam$medoids

## Time radius\_mean texure\_mean perimeter\_mean area\_mean smoothness\_mean  
## [1,] 13 17.27 25.42 112.40 928.8 0.08331  
## [2,] 53 21.09 26.57 142.70 1311.0 0.11410  
## [3,] 76 13.48 20.82 88.40 559.2 0.10160  
## [4,] 41 19.89 20.26 130.50 1214.0 0.10370  
## [5,] 38 15.05 19.07 97.26 701.9 0.09215  
## [6,] 24 23.51 24.27 155.10 1747.0 0.10690  
## compactness\_mean concavity\_mean concave\_points\_mean symmetry\_mean  
## [1,] 0.11090 0.12040 0.05736 0.1467  
## [2,] 0.28320 0.24870 0.14960 0.2395  
## [3,] 0.12550 0.10630 0.05439 0.1720  
## [4,] 0.13100 0.14110 0.09431 0.1802  
## [5,] 0.08597 0.07486 0.04335 0.1561  
## [6,] 0.12830 0.23080 0.14100 0.1797  
## fractal\_dimension\_mean radius\_SE texure\_SE perimeter\_SE area\_SE  
## [1,] 0.05407 0.5100 1.6790 3.283 58.38  
## [2,] 0.07398 0.6298 0.7629 4.414 81.46  
## [3,] 0.06419 0.2130 0.5914 1.545 18.52  
## [4,] 0.06188 0.5079 0.8737 3.654 59.70  
## [5,] 0.05915 0.3860 1.1980 2.630 38.49  
## [6,] 0.05506 1.0090 0.9245 6.462 164.10  
## smoothness\_SE compactness\_SE concavity\_SE concave\_points\_SE  
## [1,] 0.008109 0.04308 0.04942 0.017420  
## [2,] 0.004253 0.04759 0.03872 0.015670  
## [3,] 0.005367 0.02239 0.03049 0.012620  
## [4,] 0.005089 0.02303 0.03052 0.011780  
## [5,] 0.004952 0.01630 0.02967 0.009423  
## [6,] 0.006292 0.01971 0.03582 0.013010  
## symmetry\_SE fractal\_dimension\_SE radius\_worst texure\_worst  
## [1,] 0.01594 0.003739 20.38 35.46  
## [2,] 0.01798 0.005295 26.68 33.48  
## [3,] 0.01377 0.003187 15.53 26.02  
## [4,] 0.01057 0.003391 23.73 25.23  
## [5,] 0.01152 0.001718 17.58 28.06  
## [6,] 0.01479 0.003118 30.67 30.73  
## perimeter\_worst area\_worst smoothness\_worst compactness\_worst  
## [1,] 132.8 1284.0 0.1436 0.4122  
## [2,] 176.5 2089.0 0.1491 0.7584  
## [3,] 107.3 740.4 0.1610 0.4225  
## [4,] 160.5 1646.0 0.1417 0.3309  
## [5,] 113.8 967.0 0.1246 0.2101  
## [6,] 202.4 2906.0 0.1515 0.2678  
## concavity\_worst concave\_points\_worst symmetry\_worst  
## [1,] 0.5036 0.1739 0.2500  
## [2,] 0.6780 0.2903 0.4098  
## [3,] 0.5030 0.2258 0.2807  
## [4,] 0.4185 0.1613 0.2549  
## [5,] 0.2866 0.1120 0.2282  
## [6,] 0.4819 0.2089 0.2593  
## fractal\_dimension\_worst tumor\_size  
## [1,] 0.07944 9.0  
## [2,] 0.12840 4.0  
## [3,] 0.10710 2.5  
## [4,] 0.09136 8.0  
## [5,] 0.06954 2.5  
## [6,] 0.07738 3.5

#look at the cluster information  
cancer.6.pam$clusinfo

## size max\_diss av\_diss diameter separation  
## [1,] 47 282.0766 140.3779 447.4666 44.06243  
## [2,] 27 489.6757 245.6313 883.5316 53.85825  
## [3,] 35 313.5637 117.2788 453.2551 54.28800  
## [4,] 39 440.2731 152.2938 708.0736 44.06243  
## [5,] 40 256.0667 104.2747 391.3393 52.73338  
## [6,] 10 1028.4046 426.2379 1444.0797 184.90906

print(index.DB(cancer, cancer.6.pam$clustering, centrotypes="centroids"))

## $DB  
## [1] 0.8274368  
##   
## $r  
## [1] 0.7500804 0.9011130 0.8191173 0.9011130 0.8191173 0.7740798  
##   
## $R  
## [,1] [,2] [,3] [,4] [,5] [,6]  
## [1,] Inf 0.4640749 0.4096676 0.7500804 0.6751185 0.3408416  
## [2,] 0.4640749 Inf 0.2552335 0.9011130 0.2984542 0.7740798  
## [3,] 0.4096676 0.2552335 Inf 0.2729968 0.8191173 0.2453418  
## [4,] 0.7500804 0.9011130 0.2729968 Inf 0.3487637 0.4498623  
## [5,] 0.6751185 0.2984542 0.8191173 0.3487637 Inf 0.2683090  
## [6,] 0.3408416 0.7740798 0.2453418 0.4498623 0.2683090 Inf  
##   
## $d  
## 1 2 3 4 5 6  
## 1 0.0000 901.2396 692.8629 421.5328 387.9522 1888.9664  
## 2 901.2396 0.0000 1591.7276 486.7339 1287.7435 989.8942  
## 3 692.8629 1591.7276 0.0000 1114.3100 305.1251 2575.4185  
## 4 421.5328 486.7339 1114.3100 0.0000 809.3538 1476.4473  
## 5 387.9522 1287.7435 305.1251 809.3538 0.0000 2273.2299  
## 6 1888.9664 989.8942 2575.4185 1476.4473 2273.2299 0.0000  
##   
## $S  
## [1] 147.9120 270.3307 135.9315 168.2715 114.0017 495.9264  
##   
## $centers  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 46.82979 17.35298 22.80702 113.92766 935.1255 0.09950681 0.1305323  
## [2,] 39.55556 20.90889 22.20556 138.92593 1363.8741 0.10246333 0.1668156  
## [3,] 62.71429 13.22314 21.45086 86.91657 541.8600 0.10801371 0.1407549  
## [4,] 37.69231 19.50949 22.14128 128.89231 1182.8590 0.10093590 0.1446085  
## [5,] 51.32500 15.19950 21.89975 100.04800 716.9250 0.10338425 0.1352367  
## [6,] 26.60000 23.58500 24.88900 156.52000 1751.9000 0.10352400 0.1629600  
## [,8] [,9] [,10] [,11] [,12] [,13] [,14]  
## [1,] 0.1349302 0.08137021 0.1898043 0.06063702 0.5936191 1.340445 4.116234  
## [2,] 0.2053526 0.11559963 0.1976704 0.06110778 0.8379074 1.150893 5.913704  
## [3,] 0.1224026 0.06207171 0.1968200 0.06842771 0.3812314 1.300409 2.750029  
## [4,] 0.1779103 0.09975436 0.1926231 0.06038821 0.6990872 1.326108 4.942128  
## [5,] 0.1380485 0.06976675 0.1914600 0.06421450 0.4141600 1.147770 2.995225  
## [6,] 0.2305300 0.13823800 0.1848000 0.05971500 1.1765100 1.314280 8.063200  
## [,15] [,16] [,17] [,18] [,19] [,20]  
## [1,] 64.38404 0.006333170 0.02780804 0.03663000 0.01504734 0.01945326  
## [2,] 112.77185 0.006503481 0.03443259 0.04434778 0.01599889 0.02050519  
## [3,] 31.27229 0.007310371 0.03320771 0.03834829 0.01396229 0.02210177  
## [4,] 85.42821 0.006957179 0.03295128 0.04603897 0.01660969 0.02218410  
## [5,] 39.23450 0.007078525 0.03072980 0.04081625 0.01413973 0.01966042  
## [6,] 183.87900 0.005526200 0.02642400 0.03791000 0.01484100 0.01767600  
## [,21] [,22] [,23] [,24] [,25] [,26] [,27]  
## [1,] 0.003827106 20.70766 30.58936 136.7745 1309.4894 0.1372198 0.3121851  
## [2,] 0.003999704 26.16259 28.99519 175.3222 2099.3333 0.1401559 0.3983667  
## [3,] 0.004506600 15.59229 30.14000 104.4274 741.8371 0.1580677 0.4186671  
## [4,] 0.003949128 23.22718 29.39436 155.6513 1648.8974 0.1395205 0.3559513  
## [5,] 0.003915800 17.96950 30.27300 120.2675 990.4875 0.1472800 0.3713878  
## [6,] 0.003316200 31.23000 33.47700 209.0700 3006.3000 0.1397900 0.3470600  
## [,28] [,29] [,30] [,31] [,32]  
## [1,] 0.3779832 0.1682904 0.3087085 0.08421489 3.117021  
## [2,] 0.4962963 0.2122585 0.3306407 0.08785778 3.611111  
## [3,] 0.4448823 0.1623714 0.3510286 0.10681829 2.565714  
## [4,] 0.4520769 0.1871128 0.3127051 0.08552256 2.997436  
## [5,] 0.4339625 0.1629330 0.3305775 0.09401250 2.090000  
## [6,] 0.4738100 0.2259600 0.2892800 0.08191900 2.950000

#evaluate the hierarchical clustering performance  
  
hcl6 <- cutree(cancer.h.clust,k=6)  
print(index.DB(cancer, hcl6, centrotypes="centroids"))

## $DB  
## [1] 0.8627514  
##   
## $r  
## [1] 0.7241327 1.0670008 1.0670008 0.7147370 0.8018184 0.8018184  
##   
## $R  
## [,1] [,2] [,3] [,4] [,5] [,6]  
## [1,] Inf 0.7241327 0.4854332 0.7147370 0.2791016 0.2606880  
## [2,] 0.7241327 Inf 1.0670008 0.3038359 0.3933387 0.3201106  
## [3,] 0.4854332 1.0670008 Inf 0.2628643 0.7380752 0.4548073  
## [4,] 0.7147370 0.3038359 0.2628643 Inf 0.1734572 0.1868859  
## [5,] 0.2791016 0.3933387 0.7380752 0.1734572 Inf 0.8018184  
## [6,] 0.2606880 0.3201106 0.4548073 0.1868859 0.8018184 Inf  
##   
## $d  
## 1 2 3 4 5 6  
## 1 0.0000 587.2289 1011.1064 525.8127 1682.9995 2492.6335  
## 2 587.2289 0.0000 424.5027 1112.2383 1097.9037 1911.5889  
## 3 1011.1064 424.5027 0.0000 1535.1299 673.9706 1489.6699  
## 4 525.8127 1112.2383 1535.1299 0.0000 2204.7760 3009.8898  
## 5 1682.9995 1097.9037 673.9706 2204.7760 0.0000 818.6593  
## 6 2492.6335 1911.5889 1489.6699 3009.8898 818.6593 0.0000  
##   
## $S  
## [1] 231.5558 193.6759 259.2688 144.2620 238.1722 418.2439  
##   
## $centers  
## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 46.64286 16.62143 22.44560 109.23643 861.1619 0.1009751 0.1317240  
## [2,] 41.91667 19.39556 21.63000 128.19444 1172.2250 0.1011006 0.1464117  
## [3,] 32.92308 21.02308 22.81769 139.73462 1372.0000 0.1025500 0.1684277  
## [4,] 64.43902 13.35317 21.52073 87.81317 552.5220 0.1078893 0.1411073  
## [5,] 33.85714 23.02857 25.96286 152.24286 1654.4286 0.1016086 0.1555000  
## [6,] 22.75000 24.48000 22.29750 162.90000 1906.0000 0.1020925 0.1639000  
## [,8] [,9] [,10] [,11] [,12] [,13] [,14]  
## [1,] 0.1387574 0.07707274 0.1901560 0.06182321 0.5196845 1.260163 3.630071  
## [2,] 0.1774889 0.10048861 0.1920944 0.06105083 0.7186833 1.249319 5.147833  
## [3,] 0.2089892 0.11626923 0.2001462 0.06070654 0.8390192 1.263892 5.858346  
## [4,] 0.1224251 0.06310902 0.1962171 0.06813805 0.3827488 1.284678 2.799220  
## [5,] 0.2149143 0.12953429 0.1880714 0.06075429 1.0994857 1.313786 7.554857  
## [6,] 0.2333250 0.14316500 0.1779000 0.05685000 1.1832250 1.200600 8.087750  
## [,15] [,16] [,17] [,18] [,19] [,20]  
## [1,] 54.35988 0.006794226 0.02889655 0.03911333 0.01473648 0.01982714  
## [2,] 88.70667 0.006632694 0.03329000 0.04570111 0.01648042 0.02222722  
## [3,] 111.58846 0.006713500 0.03539692 0.04623462 0.01650704 0.02097000  
## [4,] 31.89463 0.007186195 0.03272463 0.03727512 0.01377244 0.02104054  
## [5,] 162.73429 0.005640571 0.02775857 0.04123000 0.01611286 0.01820000  
## [6,] 199.37500 0.005172000 0.02384250 0.02967750 0.01296250 0.01723000  
## [,21] [,22] [,23] [,24] [,25] [,26] [,27]  
## [1,] 0.003836107 19.67774 30.30393 130.4893 1190.5857 0.1411027 0.3301370  
## [2,] 0.004049667 23.45944 29.08833 157.8917 1686.3056 0.1395639 0.3624167  
## [3,] 0.003999962 25.95192 29.46769 173.4192 2059.5385 0.1396196 0.4037615  
## [4,] 0.004402073 15.84854 30.38244 106.3380 767.1122 0.1577700 0.4191451  
## [5,] 0.003745571 29.49143 33.72000 196.9429 2668.7143 0.1378571 0.3414571  
## [6,] 0.002670750 33.46750 31.73750 224.0750 3446.2500 0.1389250 0.3596750  
## [,28] [,29] [,30] [,31] [,32]  
## [1,] 0.4024263 0.1644782 0.3168000 0.08711024 2.625000  
## [2,] 0.4600278 0.1910500 0.3171083 0.08731500 3.344444  
## [3,] 0.4983000 0.2100762 0.3276654 0.08679423 3.384615  
## [4,] 0.4392727 0.1645805 0.3472171 0.10623561 2.546341  
## [5,] 0.4739857 0.2292000 0.2966714 0.08533143 2.285714  
## [6,] 0.4537500 0.2224500 0.2937500 0.07843500 3.625000

Comparing the DB index of three methods, the index of kmeans is the smallest, which shows that the clusters produced by kmeans have relative low intra-cluster distance and high inter cluster distance. Kmeans performance better than the other two methods.

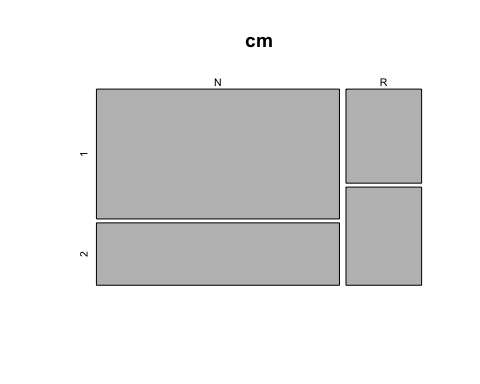
### 3. Generate and plot confusion matrices for the k-means and PAM. What do they tell you?

The confusion matrix is a 2\*2 table with counts of the true positive, true negative, false positive and false negative.  
  
Here, the dataset contains a column named "outcome", which represents the recur and nonrecur. I have to use k-means and PAM to get 2 clusters and make the confusion matrices.

#confusion matrics for k-means  
k<-2  
cancer.2.kmeans <- kmeans(cancer,centers=k)  
cm <- table(cancer\_data$Outcome,cancer.2.kmeans$cluster)  
cm

##   
## 1 2  
## N 102 49  
## R 23 24

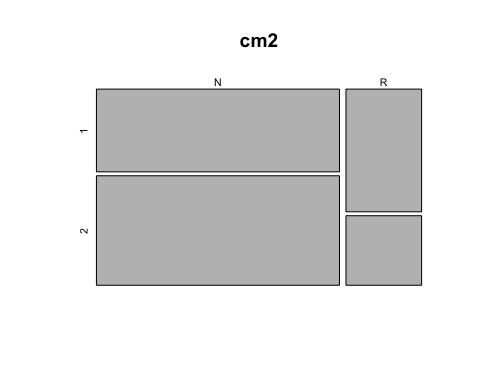
plot(cm)



#confusion matrics for PAM  
cancer.2.pam<- pam(cancer,k,keep.diss=TRUE, keep.data= TRUE)  
cm2 <- table(cancer\_data$Outcome,cancer.2.pam$clustering)  
cm2

##   
## 1 2  
## N 65 86  
## R 30 17

plot(cm2)

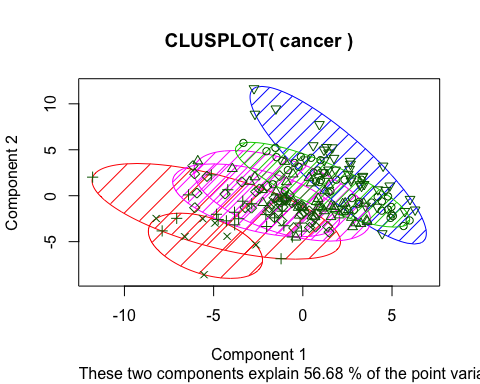


These two approaches have a large number of false positive and false negative. In k-means, although most of nonrecur patients are clustered togethe, the recur patients can't be clustered well. In PAM, recur patients can be clustered well but noncur patients can't.

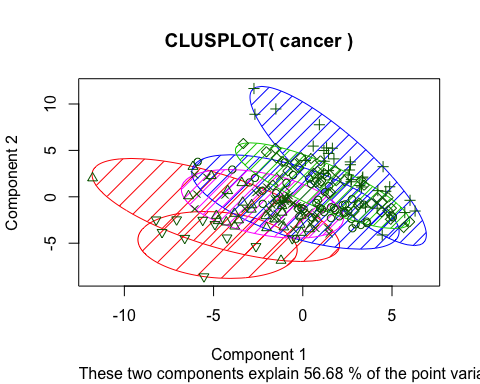
That means these two approaches don't have good performance on dividing the data into two clusters.

### 4. Generate centroid plots against the 1st two discriminant functions for k-means and PAM. What do they tell you?

#for k-means  
clusplot(cancer, cancer.6.kmeans$cluster, color=T, shade= T ,lines=0)



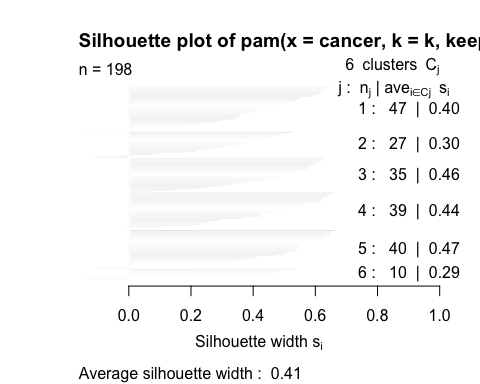
#for PAM  
clusplot(cancer,cancer.6.pam$cluster, color=T, shade= T,lines=0)



A centroid plot is to visualize the centroid for each of the clusters resulting from a partitional clustering experiment. In plots of k-means or PAM, four clusters show overlap but it has two culters seperate well.

### 5. Generate silhouette plots for PAM. What do they tell you?

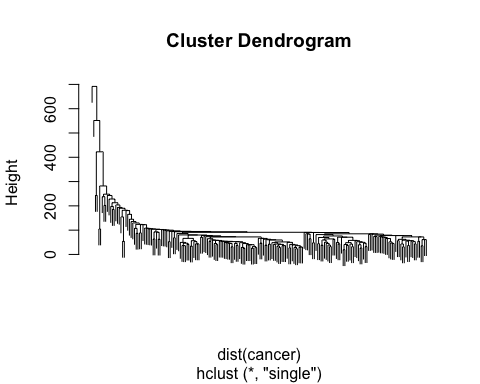
#dissC <- daisy(cancer)  
#sp<- silhouette(cancer.6.pam$clustering, dissC)  
#plot(sp)  
plot(cancer.6.pam,which.plot=2)



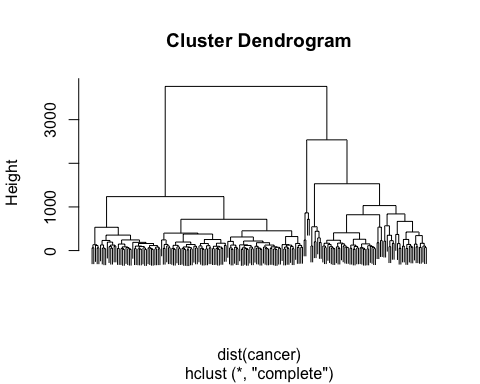
Silhouette value is a measure of how similar an object is to its own cluster compared to other clusters. In the plots of PAM, the value of cluster 5 and cluster 6 are close to one for each cluster, these two fits are good, but others are not fit so well.

### 6. For the hierarchical clustering use all linkage methods (Single Link, Complete Link, Average Link, Centroid and Minimum energy clustering) and generate dendograms. How do they compare on the same data?

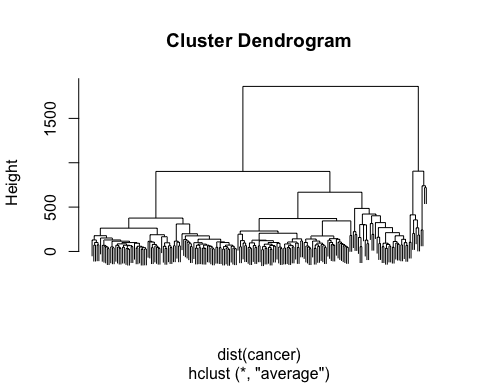
#use all linkage methods (Single Link, Complete Link, Average Link, Centroid and Minimum energy clustering) and generate dendograms  
  
cancer.h.clust.si <- hclust(d=dist(cancer),method="single")  
cancer.h.clust.co <- hclust(d=dist(cancer),method="complete")  
cancer.h.clust.av <- hclust(d=dist(cancer),method="average")  
cancer.h.clust.ce <- hclust(d=dist(cancer),method="centroid")  
  
library('energy')  
cancer.h.clust.mi <- energy.hclust(dist(cancer),alpha=1)  
plot(cancer.h.clust.si,labels=F)



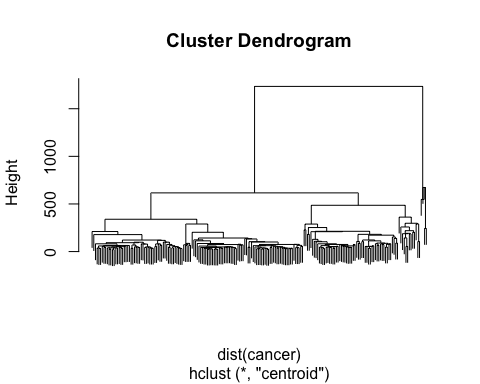
plot(cancer.h.clust.co,labels=F)



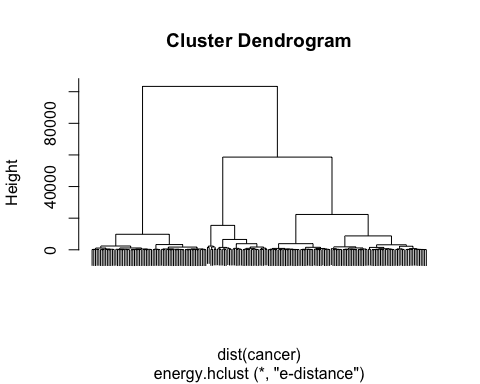
plot(cancer.h.clust.av,labels=F)



plot(cancer.h.clust.ce,labels=F)



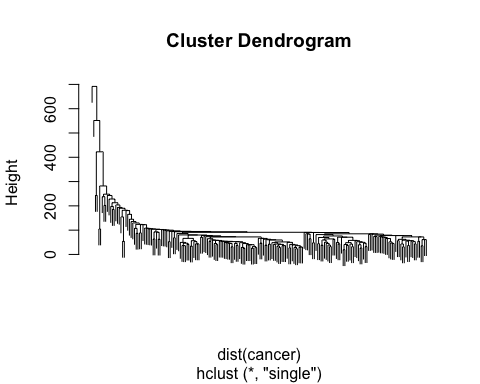
plot(cancer.h.clust.mi, labels=F)



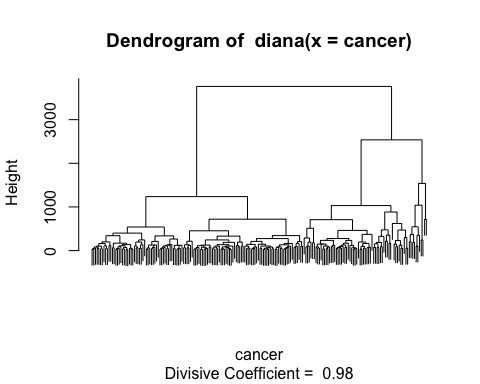
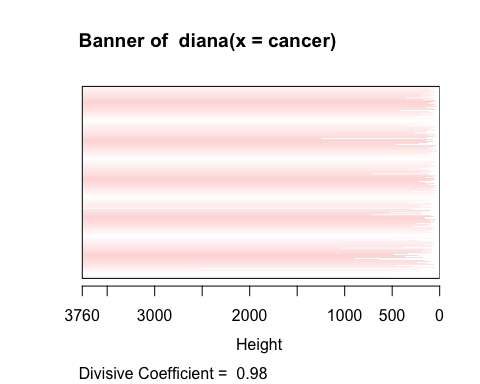
Using different linkage method will have different clustering result. the method "average" and method "centroid" have similar clustering pattern

### 7. For the hierarchical clustering use both agglomerative and divisive clustering with a linkage method of your choice and generate dendograms. How do they compare on the same data?

#use both agglomerative and divisive clustering with a linkage method of your choice and generate dendograms  
  
#agglomerative clustering  
cancer.h.clust.agg <- hclust(d=dist(cancer),method="single")  
   
   
#divisive clustering  
cancer.h.clust.div <- diana(cancer)  
  
plot(cancer.h.clust.agg,labels=F)



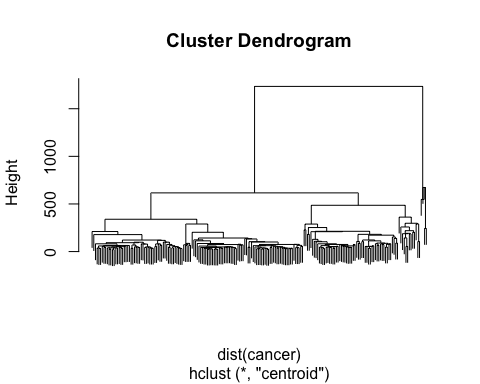
plot(cancer.h.clust.div,labels=F)



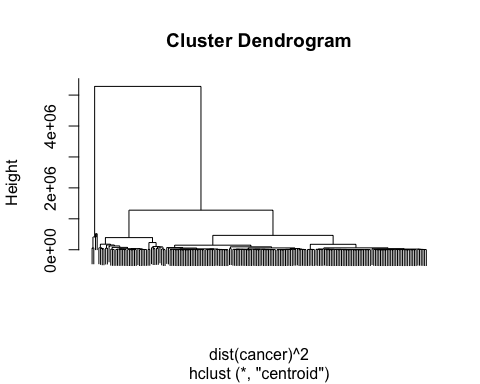
Agglomerative clustering is a "bottom-up" strategy, and the divisive clustering is a "top-down" strategy. The results of these two methods are totally different.

### 8. For the hierarchical clustering use centroid clustering and squared Euclidean distance and generate dendograms. How do they compare on the same data?

#use centroid clustering and squared Euclidean distance and generate dendograms.  
  
cancer.h.clust.ce <- hclust(d=dist(cancer),method="centroid")  
cancer.h.clust.cen.euc <- hclust(dist(cancer)^2, method="centroid" )  
  
plot(cancer.h.clust.ce, labels=F)



plot(cancer.h.clust.cen.euc, labels=F)



The result of clustering with squared Euclidean distance is different from clustering with Euclidean distance. However, it is simialr to the minimum energy.