Diego Garcia-Olano **Assignment1** **UPC-MVA-Spring2014**

Part 1: Preprocess and handle missing data.
1. Load file and make demo column explicitly categorical with different levels.

```
Russet <- read.csv("Russet_ineqdata.csv", header = TRUE, quote = "\"", dec = ".", check.names=TRUE)
Russet$demo <- as.factor(Russet$demo)
levels(Russet$demo) <- c("stable","instable","dictatorship")
dim(Russet)  #  47 10
names(Russet)   [1] "pais"    "Gini"    "farm"    "Rent"    "Gnpr"    "Laboagr" "Instab"  "ecks"    "Death"   "demo"
```

#find all columns with zero or NA or extraneous data.

table(Russet[,4]==0)	table(is.na(Russet[,4]))	table(Russet[,7]==0)	table(is.na(Russet[,8]))	table(Russet[,8]==0)	table(Russet[,9]==0)
FALSE TRUE	FALSE TRUE	FALSE TRUE	FALSE TRUE	FALSE TRUE	FALSE TRUE
42 2	44 3	44 3	46 1	40 6	29 18

Column 4(Rent) represents the percentage of farmers renting their farm.
There are 3 NAs and 2 zeros. Its hard to tell whether the 0s are true zeros or outliers.

> summary(Russet$Rent[which(!is.na(Russet$Rent))])	>sd(Russet$Rent[which(!is.na(Russet$Rent))])
Min. 1st Qu. Median Mean 3rd Qu. Max. 0.00 8.95 18.25 21.75 27.72 75.00	17.84613

Thus the InterQuartileRange is 27.72 - 8.98 = 18.74, and hence the zero values are within the acceptable lower range for a mild outlier (Q1 - 1.5 IQR), and we can keep the zeros. For the NA's, we run K-nearest neighbors on the data as follows and give the NA values those of their nearest neighbor.

```
# Imputation of 'Rent'
library(class)
aux = Russet[,-4]
aux=aux[,2:8]
aux1 = aux[!is.na(Russet$Rent),]
aux2 = aux[is.na(Russet$Rent),]
knn.rent = knn(aux1,aux2,Russet$Rent[!is.na(Russet$Rent)])
Russet$Rent[is.na(Russet$Rent)] = as.numeric(as.character(knn.rent))
```

Column 7 (Instab) is the total number of prime ministers between during 1945 - 1961. There are 3 countries (Espagne, Taiwan, and Yougoslavie) with 0 values, but this seems reasonable as they were all living under dictators at the time which can be seen as follows: > Russet$demo[which(Russet[,7]==0)] [1] dictatorship dictatorship dictatorship

Column 8 (ecks) is the index of violent conflicts during 1946 - 1961. There are 6 which have 0 values, but this seems reasonable as we can see they are also each in "stable" countries as follows:
> Russet$demo[which(Russet[,8]==0)] [1] stable stable stable stable stable stable
There is one row (Norvege) which has a NA value, but as it also has 0 deaths, and all rows with 0 deaths also have a zero "ecks" value, we assign it 0.

```
> Russet$Death[which(Russet[,8]==0)]   # [1] 0 0 0 0 0 0
> which(is.na(Russet[,8]))       #31
> Russet[31,]
pais Gini farm Rent Gnpr Laboagr Instab ecks Death   demo
31 Norvege 66.9 87.5  7.5  969      26   12.8   NA      0 stable
> Russet[31,8]  <- 0
```

We could have similarly derived that finding using K-nearest neigbhors or the mice library.

Column 9 (Death) is the number of deaths during demonstrations. This variable has a high percentage of 0 values which seems acceptable so we can leave the 0 values as they are.

Part 2: Detect outliers with mahalanobis distance metric

```
x <- Russet[,2:9]                    # only use numeric columns
G = as.matrix(colMeans(x))    #  G is  8 x 1  # as.matrix(colMeans(x))
V = cov(x)                          #   V is 8 x 8
computeDistances <- function(x,G,V)
{
        distances = seq(0,by=0, length = nrow(x))
        for(i in 1:nrow(x)){
                xi_minus_g = as.matrix(x[i,] - G)
                maha_dist = (xi_minus_g %*% solve(V)) %*% t(xi_minus_g)
```

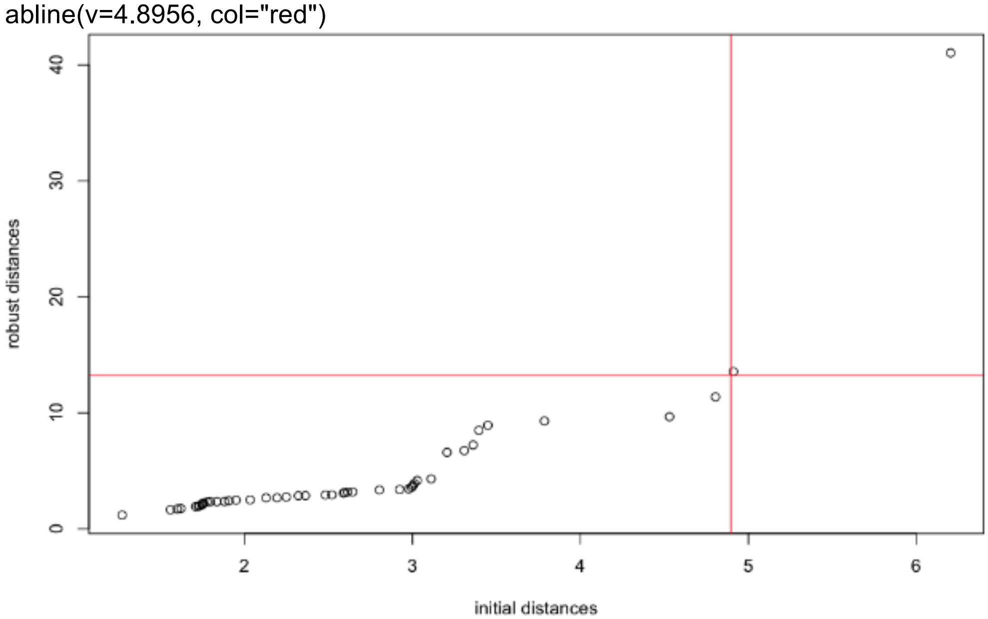
```
            distances[i] = sqrt(maha_dist)
        }
        distances
}

distances = computeDistances(x,G,V)                              #1st iteration
initial_distance = sort(distances,decreasing=TRUE,index.return=TRUE)  #rank top to bottom
lower_indexes = initial_distance$ix[11:47] # take bottom 75%
new_x = x[lower_indexes,]
new_G =  as.matrix(colMeans(new_x))
new_V = cov(new_x)
sum(new_G - G) # -73

new_distances = computeDistances(x, new_G, new_V)            #2nd iteration
sorted_distance2 = sort(new_distances,decreasing=TRUE,index.return=TRUE)
lower_indexes = sorted_distance2$ix[11:47]
n2_x = x[lower_indexes,]
n2_G = as.matrix(colMeans(n2_x))
n2_V = cov(n2_x)
sum(n2_G - new_G )   #-4

new_distances = computeDistances(x, n2_G, n2_V)               #3rd iteration
sorted_distance3 = sort(new_distances,decreasing=TRUE,index.return=TRUE
lower_indexes = sorted_distance3$ix[11:47]
n3_x = x[lower_indexes,]
n3_G = as.matrix(colMeans(n3_x))
n3_V = cov(n2_x)
sum(n3_G - n2_G )   #0  G has converged.  The variable sorted_distance3#x contains the final robust distances.

library(MASS)  # we now plot the initial distances vs the final robust mahalanobis distances.
plot(initial_distance$x,sorted_distance3$x, xlab="initial distances", ylab="robust distances)
quantile(sorted_distance3$x,c(.975))     #  97.5%   ->13.22975
quantile(initial_distance$x,c(.975))       #   97.5%   -> 4.895641
abline(h=13.22975, col="red")
abline(v=4.8956, col="red")
```



From the graph, we can see clearly that the upper right point is an outlier. Its distance is 41.036 away from the center of gravity while the mean and standard deviation of the robust distances are 4.73 and 6.08 respectively. It pertains to index 11 which is Cuba so it makes reasonable sense as an outlier.

```
> Russet[11,]
pais    Gini   farm  Rent  Gnpr Laboagr Instab ecks Death         demo
Cuba  79.2  97.8  53.8   361        42      13.6   100  2900          dictatorship
```

The other individual point right on the intersection of the two red lines from above pertains to South Vietnam. Its distance is 13.55 wherefor 97.5% our cut off distance is 13.23. This also makes reasonable sense as an outlier, but depends again on the confidence level chosen.