s23 P-set6 (For you use only. Please do not circulate or post)

1. (a) All three linkage methods produce the same final groupings. The vertical scales are different, and the order of branching for the single linkage is different from the other two.

   (b) For $k = 3$ clusters, Single linkage has clusters $\{(1), (2), (3, 4, 5)\}$, the other two methods have $\{(1, 2), (3, 4), (5)\}$.

2. $A = (5, -4), B = (1, -2), C = (-1, 1), D = (3, 1)$.

$K = 2$ initial clusters (AB) and (CD), by hand (or one-step Lloyd), the clusters will not change:

	$\bar{x}_1$	$\bar{x}_2$
(AB)	3	-3
(CD)	1	1

Each point is already in the cluster of the closest center.

Cluster	dist^2 to grp centroids			
	A	B	C	D
(AB)	5	5	(32)	(16)
(CD)	(41)	(9)	4	4

The within cluster sum of squares = 18.

It is sensible to check other possible clusters with $K = 2$. By R,

```
kmeans(cbind(c(5,1,-1,3),c(-4,-2,1,1)),2,centers=matrix(c(3,1,-3,1),2,2))
```

The clusters chosen by the algorithm are (A) and (BCD):

with centroids

	$\bar{x}_1$	$\bar{x}_2$
(A)	5	-4
(BCD)	1	0

and distances to cluster centroids

Cluster	dist^2 to grp centroids			
	A	B	C	D
(A)	0	(20)	(61)	(29)
(BCD)	(32)	4	5	5

The within cluster sum of squares for clusters {(A),(BCD)} is = 14 < 18, that for clusters {(AB), (CD)}. Therefore optimal clusters should be {(A),(BCD)}.
This exercise shows the effects of initialization (now handled in software by multiple random starts).

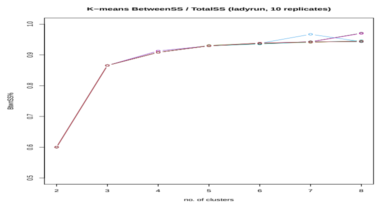
3. Note: You may use either normalized or original data (ok when variables are at comparable scales already).

   (a) distance(Kenya, Papua New Guinea) = 87.18 is the maximum distance,
   distance(Spain, Canada) = 0.51 is the minimum.
   The following R commands produce the distance matrix (too large to print), a grayscale heatmap of the distance values, ranging 0 (dark) to 86 (light). (not required), max and min distances.

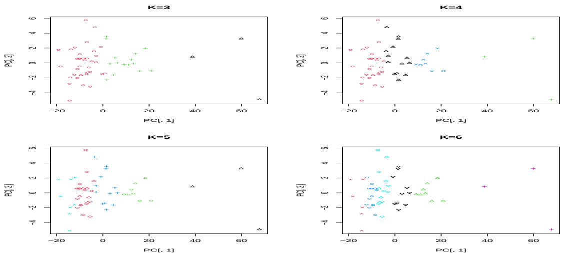
```
ladyrun = read.table("ladyrun23.dat")
     colnames(ladyrun)=c("Country","100m","200m","400m","800m","1500m","3000m","Marathon")
dmat = dist(ladyrun[,2:8]); # 54 by 54

Dmat = as.matrix(dmat)
rownames(Dmat)=ladyrun[,1]; colnames(Dmat)=ladyrun[,1]
heatmap(as.matrix(dmat),symm=T,col=gray.colors(100),Rowv=NA,Colv = "Rowv")
max(Dmat); which(Dmat == max(Dmat), arr.ind = TRUE)
min(Dmat[Dmat>0]); which(Dmat == min(Dmat[Dmat>0]), arr.ind = TRUE)
```

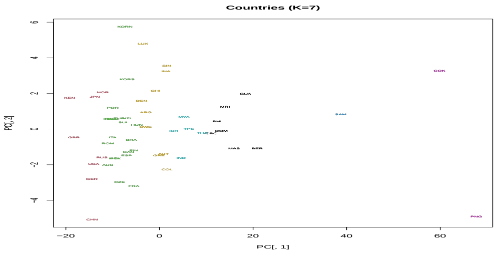
   (b) Dendrogram of average linkage differs from others at 2-cluster level. Single linkage and complete linkage differ at three cluster level. When $k = 8$ (or 7), the three smallest clusters using complete linkage are {SAM(46)}, {COK(11} , {PNG(40)}. (Dendrogram plots omitted.)

   (c) The K-means for K=3 to 8 results are reasonable and consistent with the results for the linkage procedures. The following plots (not required) show 10 replicates of the percentages of sums of squares explained by the clustering, for cluster k=2 to 8. k=3 and 4 gives substantial improvements, k=5 to 8 has marginal improvements.



The plots show cluster results K=4 to 7 on the first two principle components, with cluster K=7 showing country names. Even K=3 shows a reasonable grouping. (some output omitted)





4. The likelihood function $L = L(p_1, p_2|x_1, x_2, x_3, x_4, x_5) = \prod_{i=1}^{5}(2x_ip_1 + 2(1 - x_i)p_2)$

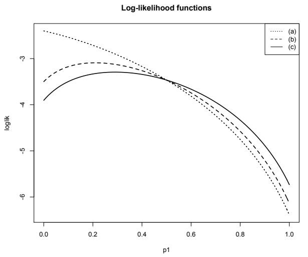
   (a) $L = 2^5(0.1p_1 + 0.9p_2)(0.2p_1 + 0.8p_2)(0.3p_1 + 0.7p_2)(0.4p_1 + 0.6p_2)(0.7p_1 + 0.3p_2)$

   (b) $L = 2^5(0.1p_1 + 0.9p_2)(0.2p_1 + 0.8p_2)(0.3p_1 + 0.7p_2)(0.4p_1 + 0.6p_2)(0.9p_1 + 0.1p_2)$

   (c) $L = 2^5(0.1p_1 + 0.9p_2)(0.2p_1 + 0.8p_2)(0.3p_1 + 0.7p_2)(0.6p_1 + 0.4p_2)(0.9p_1 + 0.1p_2)$

   (d)

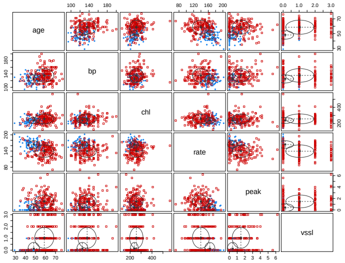
```
====== R code for Q4(d) plot =====
pts = 0:100/100
dat=c(.1,.2,.3,.4,.7)
loglik = log(cbind(pts,1-pts)%*%rbind(dat,1-dat))%*%c(1,1,1,1,1)
plot(pts,loglik,type="l",lty=3,lwd=2,xlab="p1")
dat=c(.1,.2,.3,.4,.9)
loglik = log(cbind(pts,1-pts)%*%rbind(dat,1-dat))%*%c(1,1,1,1,1)
points(pts,loglik,type="l",lty=2,lwd=2)
dat=c(.1,.2,.3,.6,.9)
loglik = log(cbind(pts,1-pts)%*%rbind(dat,1-dat))%*%c(1,1,1,1,1)
points(pts,loglik,type="l",lwd=2)
legend("topright", c("(a)","(b)","(c)"),lty=c(3,2,1))
title(main="Log-likelihood functions")
```



   (e) The estimates are about $(\hat{p}_1, \hat{p}_2) = (0, 1)$, $(0.2, 0.8)$, and $(0.3, 0.7)$ for cases (a), (b), and (c) respectively. Case (a) estimates of $p_1 = 1$ is not that reasonable (partially due to limitation of MLE at boundary points).

5. Only real valued variables should be included in this analysis. The scatter plots (omitted) shows the six real valued variables labeled by their heart disease status (heart disease cases are crosses). The two classes are quite mixed.

A quick mixture analysis of two mixtures using the 6 variables obtains the classification (illness cases are squares),



which is not ideal but ok, compared with the original data with cases and controls (heart disease free) highly mixed. Further, you may consider 4 or 5 variable models. Whichever model you choose, comment on the fit, etc. (omitted).

Q5 R code:

```
heart=read.table("heart.dat.txt")  # 270 obs
colnames(heart)=c("age","sex","chest","bp","chl","sugar","ecg","rate","angina","peak","slope","vssl","thal","ill")

# Consider real var's c(1,4,5,8,10,12) or a subset of 5 or 4 variables
pairs(heart[,c(1,4,5,8,10,12)],pch=heart$ill+1,col=heart$ill,cex=.5)
Hdata=heart[,c(1,4,5,8,10,12)]
# plot(Mclust(Hdata),what=c("BIC"))
# plot(Mclust(Hdata,G=2),what=c("density"))
plot(Mclust(Hdata,G=2),what=c("classification"))
# Note: If you get error code, reduce the number of variables to 5 or 4
```

6. The data consisting of 4 observations from trivariate normal $N_3(\boldsymbol{\mu}, \Sigma)$ are given with missing components:

$$\boldsymbol{X} = [x_{jk}] = \begin{bmatrix} 3 & 6 & 0 \\ 4 & 4 & 3 \\ - & 8 & 3 \\ 5 & - & - \end{bmatrix} \begin{matrix} obs1 \\ obs2 \\ obs3 \\ obs4 \end{matrix}$$

Use the Expectation (*a.k.a.* prediction) – Maximization (*a.k.a.* estimation) algorithm to estimate $\boldsymbol{\mu}$ and Σ.

(a) Impute values to obtain $\tilde{\boldsymbol{X}}$, the data matrix with missing values filled by initial estimates.

(b) Find $\tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$ estimated from $\tilde{\boldsymbol{X}}$.

(c) Iterate to find the first revised estimates:

i. Find the revised estimate for the missing value x_{31} in observation 3, using the estimated $\tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$ in step (b). What is the updated estimates of $\tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$?

ii. Find the revised estimates for the missing observations x_{42} and x_{43} in observation 4, using the estimated $\tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$ in step (i).

Write out the updated estimates of $\tilde{\boldsymbol{\mu}}$ and $\tilde{\Sigma}$, the revised estimates after the first iteration.

The expectation step uses $E(X_{(1)} \mid X_{(2)} = x_{(2)}) = \mu_{(1)} + \Sigma_{(12)}\Sigma_{(22)}^{-1}(x_{(2)} - \mu_{(2)})$

(a) Based on the observed column means $(4, 6, 2)$, the Initial estimates of the data are $\tilde{\boldsymbol{X}} = \begin{bmatrix} 3 & 6 & 0 \\ 4 & 4 & 3 \\ (4) & 8 & 3 \\ 5 & (6) & (2) \end{bmatrix}$.

(b) Based on (a), the maximum likelihood estimates of mean and variance are $\tilde{\boldsymbol{\mu}} = \begin{bmatrix} 4 \\ 6 \\ 2 \end{bmatrix}, \tilde{\Sigma} = \begin{bmatrix} 0.5 & 0.0 & 0.5 \\ 0.0 & 2.0 & 0.0 \\ 0.5 & 0.0 & 1.5 \end{bmatrix}$

(c) i. Using the conditional expectation to update

$$\tilde{x}_{31} = E\left(X_{(1)} = X_1 \mid X_{(2)} = \begin{bmatrix} X_2 \\ X_3 \end{bmatrix} = \begin{bmatrix} 8 \\ 3 \end{bmatrix}\right) = 4 + \begin{bmatrix} 0.0 & 0.5 \end{bmatrix} \begin{bmatrix} 2.0 & 0.0 \\ 0.0 & 1.5 \end{bmatrix}^{-1} \left(\begin{bmatrix} 8 \\ 3 \end{bmatrix} - \begin{bmatrix} 6 \\ 2 \end{bmatrix}\right) = \frac{13}{3} \approx 4.33$$

The updated mean and covariance of $\boldsymbol{X}$ are

$$\tilde{\mu} = \begin{bmatrix} 4.08 \\ 6.00 \\ 2.00 \end{bmatrix}, \quad \tilde{\Sigma} = \begin{bmatrix} 0.52 & 0.17 & 0.58 \\ 0.17 & 2.00 & 0.00 \\ 0.58 & 0.00 & 1.50 \end{bmatrix}$$

ii. Note that this extra update step is different from the example in the demo in class. To update x_{42}, x_{43} using $\begin{bmatrix} \tilde{x}_{42} \\ \tilde{x}_{43} \end{bmatrix} = E\left(X_{(1)} = \begin{bmatrix} X_2 \\ X_3 \end{bmatrix} \middle| X_{(2)} = X_1 = 5\right)$ we reorder the current mean and covariance matrix as

$$\tilde{\mu}^* = \begin{bmatrix} 6.00 \\ 2.00 \\ 4.08 \end{bmatrix}, \quad \tilde{\Sigma}^* = \begin{bmatrix} 2.00 & 0.00 & 0.17 \\ 0.00 & 1.50 & 0.58 \\ 0.17 & 0.58 & 0.52 \end{bmatrix}$$

The revised estimated

$$\begin{bmatrix} \tilde{x}_{42} \\ \tilde{x}_{43} \end{bmatrix} = E\left(X_{(1)} = \begin{bmatrix} X_2 \\ X_3 \end{bmatrix} \middle| X_{(2)} = X_1 = 5\right) = \begin{bmatrix} 6 \\ 2 \end{bmatrix} + \begin{bmatrix} 0.17 \\ 0.58 \end{bmatrix} 0.52^{-1}(5 - 4.08) \approx \begin{bmatrix} 6.30 \\ 3.03 \end{bmatrix}$$

The imputed data, the revised estimates of the population mean and variance after the first iteration are

$$\tilde{\boldsymbol{X}} = \begin{bmatrix} 3 & 6 & 0 \\ 4 & 4 & 3 \\ (4.33) & 8 & 3 \\ 5 & (6.30) & (3.03) \end{bmatrix}, \quad \tilde{\mu} \approx \begin{bmatrix} 4.08 \\ 6.07 \\ 2.26 \end{bmatrix}, \tilde{\Sigma} \approx \begin{bmatrix} 0.52 & 0.23 & 0.82 \\ 0.23 & 2.02 & 0.06 \\ 0.82 & 0.06 & 1.70 \end{bmatrix}$$

```
X=matrix(c(3,4,4,5, 6,4,8,6, 0,3,3,2),4,3)
round(3*var(X)/4,2)
X1=matrix(c(3,4,4.333333,5, 6,4,8,6, 0,3,3,2),4,3)
c(1,1,1,1)%*%X1/4    # = 4.083333    6    2
round(3*var(X1)/4,2)
c(6,2) +c(0.1666666,0.5833333)*(5-4.083333)/0.5208333    # = 6.293333 3.026667
X2=matrix(c(3,4,4.333333,5,6,4,8, 6.293333,0,3,3,3.026667),4,3)
c(1,1,1,1)%*%X2/4   # = 4.083333 6.073333 2.256667
round(3*var(X2)/4,2)
```

7. (a) In this case $d_2 = d_1^2$,

$$d_2(x,y) - d_2(z,w) = d_1(x,y)^2 - d_1(z,w)^2 = (d_1(x,y) + d_1(z,w))(d_1(x,y) - d_1(z,w))$$

Thus

$$d_2(x,y) - d_2(z,w) = c(d_1(x,y) - d_1(z,w))$$

where (in non-degenerating cases)

$$c = (d_1(x,y) + d_1(z,w)) > 0$$

which means $d_2(x,y) - d_2(z,w)$ and $c(d_1(x,y) - d_1(z,w))$ always have the same sign, or both zero. so

$$d_2(x,y) - d_2(z,w) \le 0 \quad \text{if and only if} \quad c(d_1(x,y) - d_1(z,w)) \le 0$$

or equivalently,

$$d_2(x,y) \le d_2(z,w) \quad \text{if and only if} \quad c(d_1(x,y) \le d_1(z,w))$$

which implies global-order equivalence.

(b) Many counterexamples. E.g., in 2-d plane, $x = z = (0,0), y = (2,2), w = (0,3)$.