Multivariate Statistics EX2

203832050 and 312584204

Question 1- Finding MLE for expectations and covariance matrices

We know that the MLE for the expectation is the average of the population:

$$\hat{\mu} = \bar{x}$$

Therefore,

$$\hat{\mu}_1 = \bar{x}_1 = (-0.06353216 \quad 0.06518073 \quad 0.027293872 \quad -0.132167697 \quad -0.042634169 \quad -0.029530654 \quad -0.018220567 \quad -0.26990334 \quad 0.08015979 \quad -0.006592466)^T$$

$$\hat{\mu}_2 = \bar{x}_2 = (1.0351219 \quad 1.2751972 \quad 1.2116342 \quad 1.1544727 \quad 1.0645012 \quad 1.0716831 \quad 1.0354589 \quad 0.9724618 \quad 0.9781849 \quad 0.9427868)^T$$

We know that the MLE for the covariance matrix is:

$$S = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(x_i - \bar{x})^T}{n}$$

Therefore,

$\widehat{\Sigma_1} = S_1 =$

1.134965	0.753485	0.650307	0.539305	0.519686	0.329146	0.42198	0.386192	0.277335	0.030267
0.753485	1.356093	0.902231	0.775878	0.658456	0.566827	0.298274	0.440495	0.380571	-0.01993
0.650307	0.902231	1.305247	0.836957	0.75169	0.646328	0.443579	0.636685	0.51891	0.182702
0.539305	0.775878	0.836957	1.659405	0.903098	0.802456	0.599076	0.792706	0.575605	0.144316
0.519686	0.658456	0.75169	0.903098	1.448899	0.95766	0.558394	0.750305	0.353998	0.122255
0.329146	0.566827	0.646328	0.802456	0.95766	1.524905	0.756505	0.940254	0.661778	0.381155
0.42198	0.298274	0.443579	0.599076	0.558394	0.756505	1.133939	0.863625	0.622104	0.39089
0.386192	0.440495	0.636685	0.792706	0.750305	0.940254	0.863625	1.633487	0.875284	0.607882
0.277335	0.380571	0.51891	0.575605	0.353998	0.661778	0.622104	0.875284	1.358474	0.664475
0.030267	-0.01993	0.182702	0.144316	0.122255	0.381155	0.39089	0.607882	0.664475	1.087838

$\widehat{\Sigma_2} = S_2 =$

1.263771	0.756064	0.533027	0.590776	0.375342	0.472173	0.458107	0.344952	0.279325	0.288713
0.756064	1.791631	0.933295	0.690999	0.573134	0.303116	0.341106	0.291649	0.397177	0.537872
0.533027	0.933295	1.873406	1.072807	0.912265	0.549956	0.228449	0.121987	0.29449	0.447743
0.590776	0.690999	1.072807	1.785938	0.984934	0.717232	0.487386	0.402052	0.310195	0.148486
0.375342	0.573134	0.912265	0.984934	1.589735	0.564792	0.60187	0.363469	0.416247	0.366799
0.472173	0.303116	0.549956	0.717232	0.564792	1.971469	0.861779	0.624498	0.424662	0.392307
0.458107	0.341106	0.228449	0.487386	0.60187	0.861779	1.73127	1.122097	0.469106	0.311183
0.344952	0.291649	0.121987	0.402052	0.363469	0.624498	1.122097	1.691234	0.632751	0.535469
0.279325	0.397177	0.29449	0.310195	0.416247	0.424662	0.469106	0.632751	1.843124	0.731213
0.288713	0.537872	0.447743	0.148486	0.366799	0.392307	0.311183	0.535469	0.731213	1.484362

R code for this question:

```
(mu1<-colMeans(group1))  #estimate expectation of group 1

(mu2<-colMeans(group2)) #estimate expectation of group 2


s1<-matrix(0,10,10)

for (i in 1:115) {

  r<-(group1[i,]-mu1)

  s1<-s1+r%*%t(r)

}

(s1<-s1/115)  #estimate covariance matrix for group 1

write.csv(s1, file = "cov_matrix1.csv") #for nice display from excel


s2<-matrix(0,10,10)

for (i in 1:115) {

  r<-(group2[i,]-mu2)

  s2<-s2+r%*%t(r)

}

(s2<-s2/115)  #estimate covariance matrix for group 1

write.csv(s2, file = "cov_matrix2.csv")

sigma1<-cov(group1)*114/115    #estimate covariance of group 1 - for checking
ourselves

sigma2<-cov(group2)*114/115   #estimate covariance of group 2
```

Question 2- Hypothesis testing

$$\begin{cases} H_0: \mu_1 = \mu_2 \\ H_1: \quad else \end{cases}$$

We saw in class that the statistic for this hypothesis testing is:

$$T_{st}^2 = (\overline{x_1} - \overline{x_2})^T S_{pooled}^{-1} (\overline{x_1} - \overline{x_2}) * c(n_1, n_2) \sim T_{p,n_1+n_2-p-2}^2,$$

Where:

$$S_{pooled} = \frac{(n_1 - 1)S_1 + (n_2 - 1)S_2}{n_1 + n_2 - 2}, \quad c(n_1, n_2) = \frac{n_1 * n_2}{n_1 + n_2}$$

Where S_i is the unbiased estimate for the covariance matrix in this case:

$$S_i = \frac{\sum_{j=1}^{n} (x_{ij} - \bar{x})(x_{ij} - \bar{x})^T}{n_i - 1}$$

And there is a relation between T^2 $distribution\ to\ F\ distribution$:

$$\frac{n_1 + n_2 - p - 1}{(n_1 + n_2 - 2) * p} * T_{st}^2 \sim F_{p,n_1+n_2-p\_1}$$

All we need to do is to calculate these values:

$$T_{st}^2 = 119.7124 \to F_{st} = 11.49869, F_{cr} = F_{p,n_1+n_2-p-2}^{0.95} = F_{10,218}^{0.95} = 1.874322$$

$$F_{st} > F_{cr}$$
$$\to \boldsymbol{we\ reject\ H_0\ in\ significance\ level\ of\ 5\%\ and\ say\ that\ the\ expectations}$$
$$\boldsymbol{of\ the\ groups\ are\ different}$$

R code for this question:

```
n1<-115

n2<-115

p<-10

spooled<-((n1-1)*s1(n2-1)*s2)/(n1+n2-2)

(c<-n1*n2/(n1+n2))

(tSquareSt<-t(mu1-mu2)%*%solve(spooled)%*%(mu1-mu2)*c)

(fSt<-tSquareSt*(n1+n2-p-1)/((n1+n2-2)*p))

(Fcritical<-qf(0.95,p,n1+n2-p-2))

fSt>Fcritical
```

Question 3:

a) The estimate will be S_{pooled}- a weighted average of two covariance matrices estimates for both group when the expectation is known:

$$S_{pool} = \sum_{i \in A_1} \frac{(x_i - \mu_1)(x_i - \mu_1)^T}{n_1 + n_2 - 2} + \sum_{i \in A_2} \frac{(x_i - \mu_2)(x_i - \mu_2)^T}{n_1 + n_2 - 2},$$

$$where, \quad \mu_1 = (0,0,0,0,0,0,0,0,0,0)^T, \mu_2 = (1,1,1,1,1,1,1,1,1,1)^T$$

Therefore,

$S =$

1.212546	0.764181	0.599732	0.576969	0.453948	0.40639	0.445116	0.37694	0.277816	0.160087
0.764181	1.62801	0.956087	0.756969	0.628748	0.447767	0.326817	0.356587	0.391892	0.253087
0.599732	0.956087	1.626234	0.977928	0.845574	0.610634	0.342495	0.376008	0.409042	0.31179
0.576969	0.756969	0.977928	1.758629	0.960165	0.774063	0.551974	0.618466	0.439741	0.143667
0.453948	0.628748	0.845574	0.960165	1.53566	0.770871	0.586766	0.56668	0.386068	0.244952
0.40639	0.447767	0.610634	0.774063	0.770871	1.766554	0.817793	0.792263	0.546003	0.388153
0.445116	0.326817	0.342495	0.551974	0.586766	0.817793	1.445973	1.003558	0.549264	0.353153
0.37694	0.356587	0.376008	0.618466	0.56668	0.792263	1.003558	1.714068	0.750022	0.578383
0.277816	0.391892	0.409042	0.439741	0.386068	0.546003	0.549264	0.750022	1.618322	0.704328
0.160087	0.253087	0.31179	0.143667	0.244952	0.388153	0.353153	0.578383	0.704328	1.299054

b) We saw in class that the classification rule is: $classify\ to\ group1\ if\ \frac{f_1(x)}{f_2(x)} > 1$

$$\frac{f_1(x)}{f_2(x)} = \frac{e^{-\frac{1}{2}(x-\mu_1)^T S^{-1}(x-\mu_1)}}{e^{-\frac{1}{2}(x-\mu_2)^T S^{-1}(x-\mu_2)}} > 1$$

We'll take logarithm of both side (log function keeping order):

$$\rightarrow -\frac{1}{2}\left[(x-\mu_1)^T S^{-1}(x-\mu_1) + (x-\mu_2)^T S^{-1}(x-\mu_2)\right] > 0$$

After some algebra that we saw in class we get, for each x classify to group 1 if: $(\mu_1 - \mu_2)^T S^{-1} x > \frac{1}{2}(\mu_1 - \mu_2)^T S^{-1}(\mu_1 + \mu_2)$

We noticed that for the classification rule it doesn't matter whether we use S_{pooled} or an average of both MLE for the covariances matrices because it's just a multiplication in a constant in both sides of the equation.

c) Now we classify all 230 samples by the classification rule above.

The rate of right classification is: 76.08%

R code for this question:

```
mu1<-rep(0,10)
mu2<-rep(1,10)
s1<-matrix(0,10,10)
s2<-matrix(0,10,10)
for (i in 1:115) {
  r<-(group1[i,]-mu1)
  s1<-s1+r%*%t(r)
```

```
}
for (i in 1:115) {
  r<-(group2[i,]-mu2)
  s2<-s2+r%*%t(r)
}
s<-(s1+s2)/(n1+n2-2)
write.csv(s, file = "cov_matrix3.csv") #for nice display from excel

#----Task c-------
classify<-function(x, mu1, mu2, S){
  side1<-t(mu1-mu2)%*%solve(S)%*%x
  side2<-0.5*t(mu1+mu2)%*%solve(S)%*%(mu1-mu2)
  return(side1>side2) #function that clasify to group 1
}
ans<-rep(NA,230)
for (i in 1:230) {
  ans[i]<-classify(data[i,],mu1,mu2,s)
}
((sum(ans[1:115])+115-sum(ans[116:230]))/230) #rate of right classifications
```

Question 4:

The first step is to find the eigenvalues and the eigenvectors of the estimated covariance matrix. Our estimate for the covariance matrix of group 1 was the unbiased estimate that calculated by cov(group1).

The following step is to take the eigenvectors that corresponds to the eigenvalues and can define 80% of the variance, i.e.:

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i} > 0.8$$

We define a matrix T with those eigenvectors.

In our case we found out that k=5, we need 5 components with the following eigenvectors:

```
> eigenvectors
[[1]]
       V1        V2        V3        V4        V5        V6        V7        V8        V9       V10 
0.2347344 0.2955730 0.3328556 0.3807961 0.3505760 0.3734984 0.2899505 0.3842206 0.2916548 0.1502463 

[[2]]
        V1         V2         V3         V4         V5         V6         V7         V8         V9        V10 
-0.3165811 -0.4314380 -0.2631811 -0.2006590 -0.1858589  0.1295474  0.2260329  0.3474645  0.3781172  0.4878403 

[[3]]
         V1          V2          V3          V4          V5          V6          V7          V8          V9 
 0.37175263  0.32312843  0.25686315 -0.16210708 -0.47790359 -0.43072307 -0.07047414 -0.04735821  0.40364276 
        V10 
 0.28474467 

[[4]]
         V1          V2          V3          V4          V5          V6          V7          V8          V9 
-0.48176763 -0.02172604  0.07283077  0.71581540 -0.15432126 -0.16101801 -0.34842986 -0.10623511  0.25843525 
        V10 
 0.02262772 

[[5]]
        V1         V2         V3         V4         V5         V6         V7         V8         V9        V10 
 0.3258561 -0.2709295 -0.2688083  0.4186296 -0.1950224 -0.4121847  0.4697666  0.2436768 -0.1154107 -0.2675774 
```

The final step in the PCA method is recast the data along the principal components axes:

$$FinalDataSet = T^T * DataSet^T$$

R code:

```
PCA<-prcomp(group1) #finding eigenvalues and eigenvectors of covariance
eigenValues<-as.vector(PCA$sdev^2) #eigenvalues
totEigenValue<-sum(eigenValues) #total sum of eigenvalues
p<-0
val<-0
eigenVectors<-rep(NA,10)
while(val/totEigenValue<0.8) { #adding components until we define 80%
  p<-p+1
  val<-val+eigenValues[p]
  eigenVectors[p]<-list(PCA$rotation[,p])}
eigenVectors1<-matrix(NA,10,5)
for (i in 1:5) {  eigenVectors1[,i]<-eigenVectors[[i]]} #matrix T
newdataset<-t(eigenVectors1)%*%t(group1) #new data set
```

Question 5:

Let,

$$f_i(x) = (2\pi)^{-p/2} |\Sigma|^{-1/2} \exp\left\{-\frac{1}{2}(x-\mu_i)^T \Sigma^{-1}(x-\mu_i)\right\} \text{ for i=1,…,k}$$

And define: $W_{ij}(x) = \log\left(\frac{f_i(x)}{f_j(x)}\right)$, $W_{ij}(x) = x^T \Sigma^{-1}(\mu_i - \mu_j) - \frac{1}{2}(\mu_i - \mu_j)^T \Sigma^{-1}(\mu_i + \mu_j)$

a) $-W_{ji} = -x^T\Sigma^{-1}(\mu_j - \mu_i) + \frac{1}{2}(\mu_j - \mu_i)^T\Sigma^{-1}(\mu_j + \mu_i)$

We look on the first summand, by linear algebra rules:

$$-x^T\Sigma^{-1}(\mu_j - \mu_i) = x^T\Sigma^{-1} * (-1)(\mu_j - \mu_i) = x^T\Sigma^{-1}(\mu_i - \mu_j) \qquad (1)$$

Now, we look on the second summand:

$$\frac{1}{2}(\mu_j - m_i)^T \Sigma^{-1}(\mu_j + \mu_i)$$
$$= \frac{1}{2}(\mu_j^T\Sigma^{-1}\mu_j + \mu_j^T\Sigma^{-1}\mu_i - \mu_i^T\Sigma^{-1}\mu_j - \mu_i^T\Sigma^{-1}\mu_i)$$

Because $\mu_j^T\Sigma^{-1}\mu_i$ is a scalar we know that $\mu_j^T\Sigma^{-1}\mu_i = (\mu_j^T\Sigma^{-1}\mu_i)^T$ and by transpose rules:

$$(\mu_j^T\Sigma^{-1}\mu_i)^T = \mu_i^T(\Sigma^{-1})^T\mu_j = \mu_i^T\Sigma^{-1}\mu_j$$

The last identity is because Σ^{-1} is a symmetric matrix.

So we get:

$$\frac{1}{2}(\mu_j^T\Sigma^{-1}\mu_j + \mu_j^T\Sigma^{-1}\mu_i - \mu_i^T\Sigma^{-1}\mu_j - \mu_i^T\Sigma^{-1}\mu_i) =$$
$$\frac{1}{2}(-\mu_i^T\Sigma^{-1}\mu_i + \mu_j^T\Sigma^{-1}\mu_j) \qquad (2)$$

We notice also in the following:

$$\frac{1}{2}(\mu_i - \mu_j)^T\Sigma^{-1}(\mu_i + \mu_j) = \frac{1}{2}(\mu_i^T\Sigma^{-1}\mu_i - \mu_j^T\Sigma^{-1}\mu_j + \mu_i^T\Sigma^{-1}\mu_j - \mu_j^T\Sigma^{-1}\mu_i)$$

And we already showed that the last two summands are equal.

So by (2) we get:

$$-\frac{1}{2}(\mu_i^T\Sigma^{-1}\mu_i + \mu_j^T\Sigma^{-1}\mu_j) = -\frac{1}{2}(\mu_i - \mu_j)^T\Sigma^{-1}(\mu_i + \mu_j) \; (3)$$

And by (1) and (3) we get the desired result:

$$-W_{ji} = x^T\Sigma^{-1}(\mu_i - \mu_j) - \frac{1}{2}(\mu_i - \mu_j)^T\Sigma^{-1}(\mu_i + \mu_j) = W_{ij} \quad \blacksquare$$

b) $W_{ij} + W_{jk} = x^T\Sigma^{-1}((\mu_i - \mu_j) + (\mu_j - \mu_k))$
$$-\frac{1}{2}[(\mu_i - \mu_j)^T\Sigma^{-1}(\mu_i + \mu_j) + (\mu_j - \mu_k)^T\Sigma^{-1}(\mu_j + \mu_k)]$$

First, look at first summand:

$$x^T\Sigma^{-1}((\mu_i - \mu_j) + (\mu_j - \mu_k)) = x^T\Sigma^{-1}(\mu_i - \mu_k)$$

The second summand:

$$-\frac{1}{2}[(\mu_i-\mu_j)^T\Sigma^{-1}(\mu_i+\mu_j)+(\mu_j-\mu_k)^T\Sigma^{-1}(\mu_j+\mu_k)]=$$

$$-\frac{1}{2}[\mu_i^T\Sigma^{-1}\mu_i-\mu_j^T\Sigma^{-1}\mu_i+\mu_i^T\Sigma^{-1}\mu_j-\mu_j^T\Sigma^{-1}\mu_j+\mu_j^T\Sigma^{-1}\mu_j-\mu_k^T\Sigma^{-1}\mu_j \\ +\mu_j^T\Sigma^{-1}\mu_k-\mu_k^T\Sigma^{-1}\mu_k]$$

As we showed before: $\mu_j^T\Sigma^{-1}\mu_i=\mu_i^T\Sigma^{-1}\mu_j \; and \; \mu_k^T\Sigma^{-1}\mu_j=\mu_j^T\Sigma^{-1}\mu_k$, so we get:

$$-\frac{1}{2}[\mu_i^T\Sigma^{-1}\mu_i-\mu_k^T\Sigma^{-1}\mu_k]=-\frac{1}{2}(\mu_i-\mu_k)^T\Sigma^{-1}(\mu_i+\mu_k)$$

The last identity was showed in the previous task (see equation (3)).

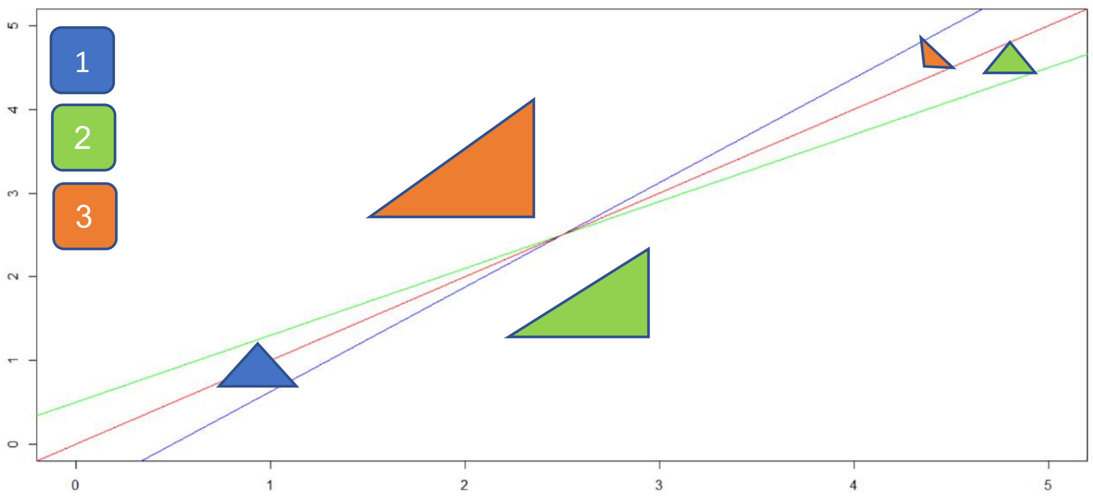
Finally we can see:

$$W_{ij}+W_{jk}=x^T\Sigma^{-1}(\mu_i-\mu_k)-\frac{1}{2}(\mu_i-\mu_k)^T\Sigma^{-1}(\mu_i+\mu_k)=W_{ik} \quad \blacksquare$$

c) We need to look what are the curves that dividing the space, according to question 3 and 5 we need to find the curves for them $W_{ij}>0 \; for \; i\neq j$

$$\mu_1=\begin{pmatrix}0\\0\end{pmatrix}, \mu_2=\begin{pmatrix}1\\0\end{pmatrix}, \mu_3=\begin{pmatrix}0\\1\end{pmatrix}, \Sigma=\begin{pmatrix}1 & .8\\ .8 & 1\end{pmatrix}$$

$$W_{12}>0\rightarrow \; x^T\Sigma^{-1}(\mu_1-\mu_2)>\frac{1}{2}(\mu_1-\mu_2)^T\Sigma^{-1}(\mu_1+\mu_2)$$
$$\rightarrow (x_1x_2)\begin{bmatrix}1 & .8\\ .8 & 1\end{bmatrix}^{-1}\begin{pmatrix}-1\\0\end{pmatrix}>\frac{1}{2}(-1 \quad 0)\begin{bmatrix}1 & .8\\ .8 & 1\end{bmatrix}^{-1}\begin{pmatrix}1\\0\end{pmatrix}$$
$$\rightarrow -2.7778x_1+2.2222x_2>-1.3889\rightarrow$$
$$x_1<\frac{2.2222x_2+1.3889}{2.7778}$$
$$W_{13}>0\rightarrow \; x^T\Sigma^{-1}(\mu_1-\mu_3)>\frac{1}{2}(\mu_1-\mu_3)^T\Sigma^{-1}(\mu_1+\mu_3)$$
$$\rightarrow (x_1x_2)\begin{bmatrix}1 & .8\\ .8 & 1\end{bmatrix}^{-1}\begin{pmatrix}0\\-1\end{pmatrix}>\frac{1}{2}(0 \quad -1)\begin{bmatrix}1 & .8\\ .8 & 1\end{bmatrix}^{-1}\begin{pmatrix}0\\1\end{pmatrix}$$
$$\rightarrow 2.2222x_1-2.7778x_2>-1.3889\rightarrow$$
$$x_1>\frac{2.7778x_2-1.3889}{2.2222}$$
$$W_{23}>0\rightarrow \; x^T\Sigma^{-1}(\mu_2-\mu_3)>\frac{1}{2}(\mu_2-\mu_3)^T\Sigma^{-1}(\mu_2+\mu_3)$$
$$\rightarrow (x_1x_2)\begin{bmatrix}1 & .8\\ .8 & 1\end{bmatrix}^{-1}\begin{pmatrix}1\\-1\end{pmatrix}>\frac{1}{2}(1 \quad -1)\begin{bmatrix}1 & .8\\ .8 & 1\end{bmatrix}^{-1}\begin{pmatrix}1\\1\end{pmatrix}$$
$$\rightarrow 5x_1-5x_2>0\rightarrow x_1>x_2$$



R code:

```
sig5<-matrix(c(1,0.8,0.8,1), 2,2)

mu1<-c(0,0)

mu2<-c(1,0)

mu3<-c(0,1)

a1<-solve(sig5)%*%(mu1-mu2)

b1<-(0.5*t(mu1-mu2)%*%solve(sig5)%*%(mu1+mu2))/a1[2]

a2<-solve(sig5)%*%(mu1-mu3)

b2<-(0.5*t(mu1-mu3)%*%solve(sig5)%*%(mu1+mu3))/a2[2]

a3<-solve(sig5)%*%(mu2-mu3)

b3<-(0.5*t(mu2-mu3)%*%solve(sig5)%*%(mu2+mu3))/a3[2]

plot(0:5,0:5, type = "n")

abline(b1,-a1[1]/a1[2],col='blue')

abline(b2,-a2[1]/a2[2],col='green')

abline(b3,-a3[1]/a3[2],col='red')
```

Question 6:

We can see from the last question the classification rules:

$$classify\ to\ group\ 1\ iff\ x_1 < \frac{2.2222x_2 + 1.3889}{2.7778}\ and\ x_1 > \frac{2.7778x_2 - 1.3889}{2.2222}$$

$$classify\ to\ group\ 2\ iff\ x_1 > \frac{2.2222x_2 - 1.3889}{2.7778}\ and\ x_1 > x_2$$

$$classify\ to\ group\ 3\ iff\ x_1 < x_2\ and\ x_1 < \frac{2.7778x_2 - 1.3889}{2.2222}$$

So we declare a function that will follow this rules and classify the samples.

The rate of the right classifications is: 78.889%.

R code:

```
classifyFunc <- function(x1,x2){ #Function that classify
  if(x1 < (2.2222*x2+1.3889)/2.7778 & x1 >(2.7778*x2-1.3889)/2.2222)
    return(1)
  if(x1>=(2.2222*x2+1.3889)/2.7778 & x1> x2)
    return(2)
  if(x1<= x2 &  x1 <= (2.7778*x2-1.3889)/2.2222 )
    return(3)}
set.seed(11)
data1<- rmvnorm(30,mu1,sig5)
data2<- rmvnorm(30,mu2,sig5)
data3<- rmvnorm(30,mu3,sig5)
sum<- 0
for (i in 1:30) {
  if(classifyFunc(data1[i,1],data1[i,2])==1)
     sum<- sum+1}
for (i in 1:30) {
  if(classifyFunc(data2[i,1],data2[i,2])==2)
     sum<- sum+1}
for (i in 1:30) {
  if(classifyFunc(data3[i,1],data3[i,2])==3)
    sum<- sum+1}
(sum/90)
```

Question 7:

The estimates for the parameters are:

$\widehat{\mu_1} = \ (-0.2805 \quad -0.2304)^T$

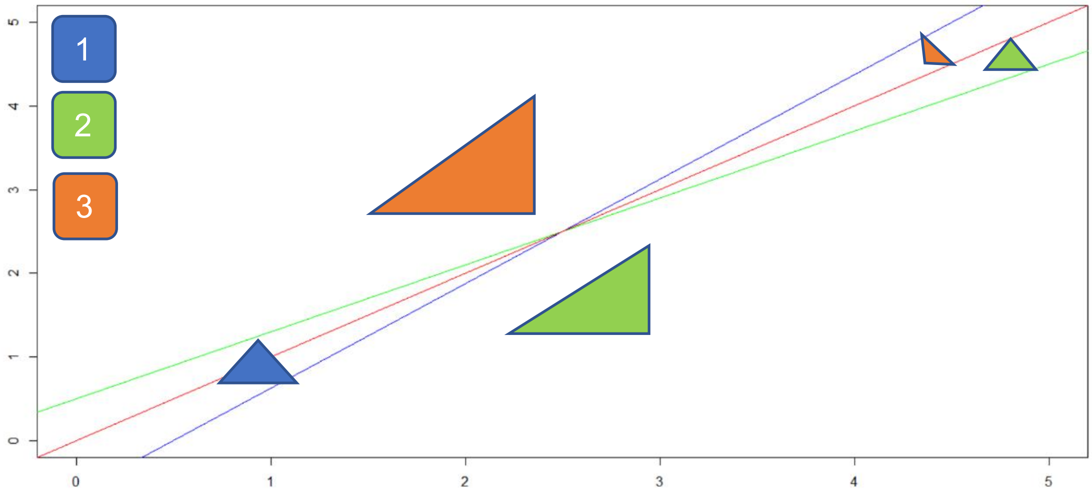
$\widehat{\mu_2} = (1.03545 \quad -0.04818)^T$

$\widehat{\mu_3} = (0.06217 \quad 1.1752)^T$

$\hat{\Sigma} = \begin{pmatrix} 0.87849 & 0.63588 \\ 0.63588 & 0.78331 \end{pmatrix}$

$got\ by\ a\ simple\ average\ of\ 3\ unbiased\ estimates\ for\ each\ group$

The discriminant curves are:



$$W_{12} > 0 \to \ x^T\Sigma^{-1}(\mu_1 - \mu_2) > \frac{1}{2}(\mu_1 - \mu_2)^T\Sigma^{-1}(\mu_1 + \mu_2)$$
$$\to -3.2239x_1 + 2.3844x_2 > -1.5491 \to$$
$$x_1 < \frac{2..3844x_2 + 1.5491}{3.2239}$$
$$W_{13} > 0 \to \ x^T\Sigma^{-1}(\mu_1 - \mu_3) > \frac{1}{2}(\mu_1 - \mu_3)^T\Sigma^{-1}(\mu_1 + \mu_3)$$
$$\to 2.2036x_1 - 3.5833x_2 > -1.9332 \to$$
$$x_1 > \frac{3.5833x_2 - 1.9332}{2.2036}$$

$$W_{23} > 0 \to \ x^T\Sigma^{-1}(\mu_2 - \mu_3) > \frac{1}{2}(\mu_2 - \mu_3)^T\Sigma^{-1}(\mu_2 + \mu_3) \to$$

$$5.4276x_1 - 5.9678x_2 > -0.3841 \to x_1 > \frac{5.9678x_2 - 0.3841}{5.4276}$$

R code:

```
mu1<-colMeans(data1)

mu2<-colMeans(data2)

mu3<-colMeans(data3)

sig<-(cov(data1)+cov(data2)+cov(data3))/3

a1<-solve(sig)%*%(mu1-mu2)

b1<-(0.5*t(mu1-mu2)%*%solve(sig)%*%(mu1+mu2))/a1[2]

a2<-solve(sig)%*%(mu1-mu3)

b2<-(0.5*t(mu1-mu3)%*%solve(sig)%*%(mu1+mu3))/a2[2]

a3<-solve(sig)%*%(mu2-mu3)

b3<-(0.5*t(mu2-mu3)%*%solve(sig)%*%(mu2+mu3))/a3[2]

plot(-1:5,-1:5, type = "n")

abline(b1,-a1[1]/a1[2],col='blue')

abline(b2,-a2[1]/a2[2],col='green')

abline(b3,-a3[1]/a3[2],col='red')
```

Question 8:

a) $group\ 1: X_1 \sim Exp(\lambda_1),\ X_2 \sim Exp(\lambda_2)$
$group\ 2: X_1 \sim Exp(\lambda_2),\ X_2 \sim Exp(\lambda_1)$

$$\frac{f_1(x)}{f_2(x)} = \frac{\lambda_1 e^{-\lambda_1 x_1} \lambda_2 e^{-\lambda_2 x_2}}{\lambda_2 e^{-\lambda_2 x_1} \lambda_1 e^{-\lambda_1 x_2}} > 1$$

We'll take logarithm:

$$\frac{f_1(x)}{f_2(x)} > 1 \leftrightarrow \log\left(\frac{f_1(x)}{f_2(x)}\right) > 0$$
$$\log\left(\frac{f_1(x)}{f_2(x)}\right) = -\lambda_1 x_1 - \lambda_2 x_2 + \lambda_2 x_1 + \lambda_1 x_2 = (x_1 - x_2)(\lambda_2 - \lambda_1) > 0$$

	$(\lambda_2 > \lambda_1)$	$(\lambda_2 < \lambda_1)$
$(x_1 > x_2)$	Group1	Group2
$(x_1 < x_2)$	Group2	Group1

b) $\alpha = \beta\ from\ symmetry$
Without loss of generality we assume $(\lambda_2 > \lambda_1)$

$$P_{group_2}(x_1 > x_2) = \int_0^\infty p[x_1 - x_2 > 0|x_2] * f(x_2)dx_2$$
$$= \int_{x_2}^\infty f_{x_1}(x_1)dx_1 \int_0^\infty f_{x_2}(x_2)dx_2$$
$$= \int_{x_2}^\infty \lambda_2 e^{-\lambda_2 x_1} dx_1 \int_0^\infty \lambda_1 e^{-\lambda_1 x_2} dx_2$$
$$= \int_0^\infty \int_{x_2}^\infty \lambda_2 \lambda_1 e^{-\lambda_2 x_1} e^{-\lambda_1 x_2}\, dx_1 dx_2$$
$$= \int_0^\infty \lambda_1 e^{-\lambda_1 x_2} dx_2 \int_{x_2}^\infty \lambda_2 e^{-\lambda_2 x_1} dx_1$$
$$= \int_0^\infty \lambda_1 e^{-\lambda_1 x_2} dx_2 \left[1 - F_{x_1}(x_2)\right]$$
$$= \int_0^\infty \lambda_1 e^{-\lambda_1 x_2} dx_2 \left[1 - (1 - e^{-\lambda_2 x_2})\right]$$
$$= \int_0^\infty \lambda_1 e^{-\lambda_1 x_2} e^{-\lambda_2 x_2} dx_2$$
$$= \frac{\lambda_1}{\lambda_1 + \lambda_2} \int_0^\infty (\lambda_1 + \lambda_2) e^{(\lambda_2 + \lambda_1) - x_2} dx_2$$

Since $\int_0^\infty (\lambda_1 + \lambda_2)e^{(\lambda_2+\lambda_1)-x_2}dx_2 = 1$, because it's a distribution function of $Exp(\lambda_1 + \lambda_2)\ random\ variable$ we get $\alpha = \beta = \frac{\lambda_1}{\lambda_1+\lambda_2}$

If $(\lambda_2 < \lambda_1)$, all we need to do is to switch indices, so we get $\alpha = \beta = \frac{\lambda_2}{\lambda_1+\lambda_2}$

In conclusion the result is:

$$\alpha = \beta = \frac{\min\{\lambda_1, \lambda_2\}}{\lambda_1 + \lambda_2}$$

c) $\lambda_1 = 1, \lambda_2 = 3$

As we can see $\lambda_2 > \lambda_1$ therefore we classify to group 1 if $x_1 > x_2$ and to group 2 otherwise.

When we sampled the data we "flipped a fair coin" to determine whether the observation is from group 1 or 2.

The rate of right classification is: 78%, which means 22% of mistakes, very close to what we found in the previous question:

$$\frac{\lambda_1}{\lambda_1 + \lambda_2} = 0.25$$

d) As we saw in class the rule that classify to group 1 is:

$$\frac{f_1(x)}{f_2(x)} = \frac{\lambda_1 e^{-\lambda_1 x_1}\lambda_2 e^{-\lambda_2 x_2}}{\lambda_2 e^{-\lambda_2 x_1}\lambda_1 e^{-\lambda_1 x_2}} > \frac{1-\pi}{\pi} = \frac{\frac{3}{4}}{\frac{1}{4}} = 3$$

Again. We'll take logarithm and get:

$(x_1 - x_2)(\lambda_2 - \lambda_1) > \log(3)$

So the classification rule will be:

	$(\lambda_2 > \lambda_1)$	$(\lambda_2 < \lambda_1)$
$\left(x_1 - x_2 > \frac{\log(3)}{(\lambda_2 - \lambda_1)}\right)$	Group1	Group2
$\left(x_1 - x_2 < \frac{\log(3)}{(\lambda_2 - \lambda_1)}\right)$	Group2	Group1

e) As we can see $\lambda_2 > \lambda_1$ therefore we classify to group 1 if $x_1 - x_2 > \frac{\log(3)}{(\lambda_2-\lambda_1)}$ and to group 2 otherwise.

When we sampled the data we "flipped an unfair coin" with probability 0.25 and 0.75 to determine whether the observation is from group 1 or 2.

The rate of right classification is: 86%, which means 14% of mistakes.

<u>R code:</u>

```
data8<-matrix(NA,100,3)
lambda1<-1
lambda2<-3
for (i in 1:100) {
  if(runif(1)<0.25)
    data8[i,]<-c(1,rexp(1,lambda1),rexp(1, lambda2))
  else
    data8[i,]<-c(2,rexp(1,lambda2),rexp(1, lambda1))}
ans<-rep(NA,100)
for (i in 1:100) {
  if((data8[i,2]-data8[i,3])>log(3,base = exp(1))/(lambda2-lambda1))
    ans[i]<-1
  else
    ans[i]<-2}
sum(ans==data8[,1])
```