BST 691 Homework 1

Name: Ruijie Yin

1. (a) Given that $C(0,1)=3$, $C(1,0)=2$

the risk function can be written as:

$R(f) = E_X[C(0,1)P(Y=0|X)I(f(X)=0) + C(1,0)P(Y=1|X)I(f(X)=0)]$

The Bayes rule is:

$$\phi_B = \begin{cases} 1, & \text{if } C(1,0)P(Y=1|X=x) > C(0,1)P(Y=0|X=x) \\ 0, & \text{if } C(1,0)P(Y=1|X=x) < C(0,1)P(Y=0|X=x) \end{cases}$$

$$\Rightarrow \begin{cases} 1, & \text{if } \frac{P(Y=1|X=x)}{P(Y=0|X=x)} > \frac{C(0,1)}{C(1,0)} \\ 0, & \text{otherwise} \end{cases} \Rightarrow \begin{cases} 1, & \text{if } P(Y=1|X=x) > \frac{C(0,1)}{C(1,0)+C(0,1)} = \frac{3}{5} \\ 0, & \text{otherwise} \end{cases}$$

, or can be expressed as: $\phi_B = \begin{cases} 1, & \text{if } \frac{g_1(X)}{g_0(X)} > \frac{\pi_0 C(0,1)}{\pi_1 C(1,0)} \\ 0, & \text{if } \frac{g_1(X)}{g_0(X)} < \frac{\pi_0 C(0,1)}{\pi_1 C(1,0)} \end{cases}$, given that $\pi_0=\pi_1=0.5$ in the example, $\phi_B = \begin{cases} 1, & \text{if } \frac{g_1(X)}{g_0(X)} > \frac{3}{2} \\ 0, & \text{if } \frac{g_1(X)}{g_0(X)} < \frac{3}{2} \end{cases}$, in which $g_1(x)=\phi(x;\mu=0,\sigma=1)$ and $g_0(X)=0.65\phi(X;\mu=1,\sigma=1)+0.35\phi(X;\mu=-1,\sigma=2)$

(b) the Bayes decision boundary is $\left\{x: \frac{g_1(X)}{g_0(X)} = \frac{3}{2}\right\} \Rightarrow \left\{x: \frac{\frac{1}{\sqrt{2\pi}}\exp\{-\frac{1}{2}x^2\}}{\frac{1}{\sqrt{2\pi}}\exp\{-\frac{1}{2}(x-1)^2\}\times 0.65 + \frac{1}{2\sqrt{2\pi}}\exp\{-\frac{1}{8}(x+1)^2\}\times 0.35} = \frac{3}{2}\right\}$

(c) The boundary is: $\left\{x: \frac{g_1(X)}{g_0(X)} = \frac{3}{2}\right\} = \{-1.58, 0.27\}$. Hence, the optimal classification regions are now: $\Omega_1^* = (-1.58, 0.27)$ and $\Omega_0^* = (-\infty, -1.58) \cup (0.27, \infty)$

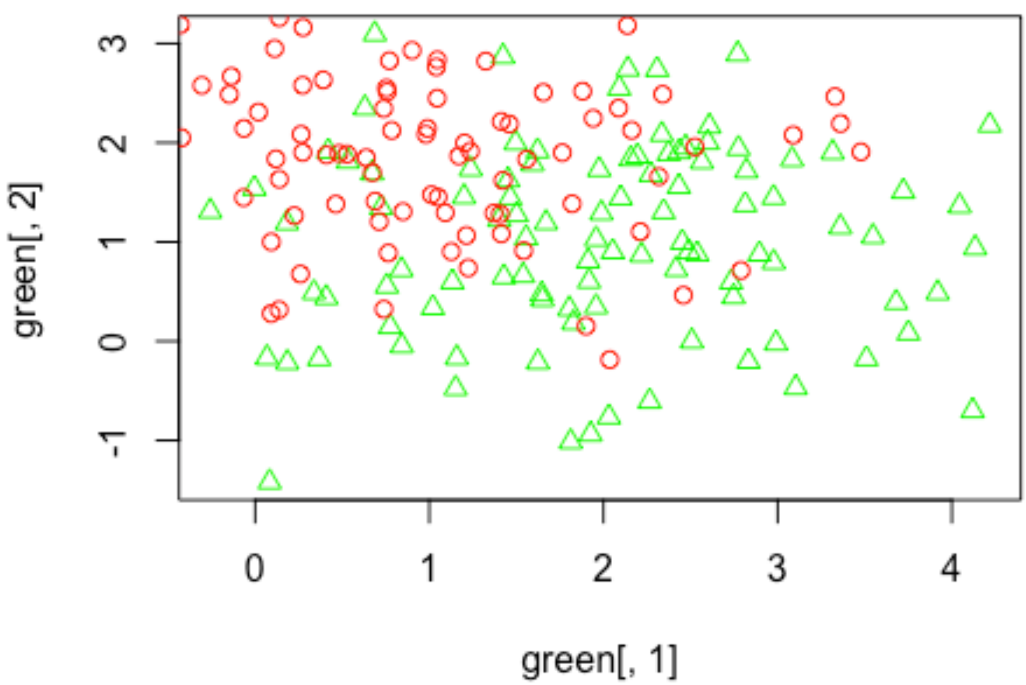
HW1.q2

(a)Generate a training dataset

```
#generate training dataset
library(MASS)
set.seed(2000)
miu1<-c(2,1)
miu2<-c(1,2)
sigma<-matrix(c(1,0,0,1),2,2,byrow = T)
green<-mvrnorm(100,miu1,sigma)
red<-mvrnorm(100,miu2,sigma)
#training dataset
label<-matrix(c(rep(1,100),rep(2,100)),200,1)
train_data<-rbind(green,red)
train_data<-cbind(train_data,label)
```

(b)Draw the scatter plot of the training data: The red circles are in class 'red', while the green diamonds are in class 'green'.

```
#Show green points and red points on the same scatterplot
plot(green[,1],green[,2],col="green",pch=2)
points(red[,1],red[,2],col="red")
```



(c)Generate a testing set and save to local drive.

```
#generate a test set
library(MASS)
set.seed(2014)
miu1<-c(2,1)
miu2<-c(1,2)
sigma<-matrix(c(1,0,0,1),2,2,byrow = T)
green2<-mvrnorm(500,miu1,sigma)
red2<-mvrnorm(500,miu2,sigma)
label2<-matrix(c(rep(1,500),rep(2,500)),1000,1)
test_set<-rbind(green2,red2)
test_set<-cbind(test_set,label2)
write.table(test_set,file = '/Users/ruijieyin/Dropbox/UM Biostatistics/BST691
High Dimensional and Complex Data/hw1q2testdataset.txt')
```

3. (a) 0-1 loss $L = I(Y \neq f(x))$ is equivalent to equal cost, which is $C(\text{green}, \text{red}) = C(\text{red}, \text{green})$, given that two classes have the same prior probabilities and denote $g_1(X) = \phi(X; \mu_1 = \binom{2}{1}, \Sigma_1 = \underline{I})$ (green) and $g_2(X) = \phi(X; \mu_2 = \binom{1}{2}, \Sigma_2 = \underline{I})$ (red)

The Bayes classifier can be expressed as:

$$f^*(X) = \begin{cases} \text{green}, & \text{if } \frac{g_1(X)}{g_2(X)} > \frac{\pi_2}{\pi_1} \\ \text{red}, & \text{if } \frac{g_1(X)}{g_2(X)} < \frac{\pi_2}{\pi_1} \end{cases}$$

$\Rightarrow$ We classify a point to green if $\frac{g_1(X)}{g_2(X)} > 1$ and to red if $\frac{g_1(X)}{g_2(X)} < 1$, where:

$$\frac{g_1(X)}{g_2(X)} = \frac{(2\pi)^{-p_1/2} |\Sigma_1|^{-\frac{1}{2}} \exp\{-(X-\mu_1)^T \Sigma_1^{-1} (X-\mu_1)/2\}}{(2\pi)^{-p_2/2} |\Sigma_2|^{-\frac{1}{2}} \exp\{-(X-\mu_2)^T \Sigma_2^{-1} (X-\mu_2)/2\}}$$

in which $p_1 = p_2 = 100$, $\Sigma_1 = \Sigma_2 = \underline{I} = \Sigma$

and
$$\frac{g_1(X)}{g_2(X)} = \exp\{-\tfrac{1}{2}(X-\mu_1)^T \Sigma^{-1} (X-\mu_1) + \tfrac{1}{2}(X-\mu_2)^T \Sigma^{-1} (X-\mu_2)\}$$
$$= \exp\{\tfrac{1}{2}(X-\mu_2)^T \Sigma^{-1} (X-\mu_2) - \tfrac{1}{2}(X-\mu_1)^T \Sigma^{-1} (X-\mu_1)\}$$

the Bayes decision boundary is $\{X : \frac{g_1(X)}{g_2(X)} = 1\}$. that is when

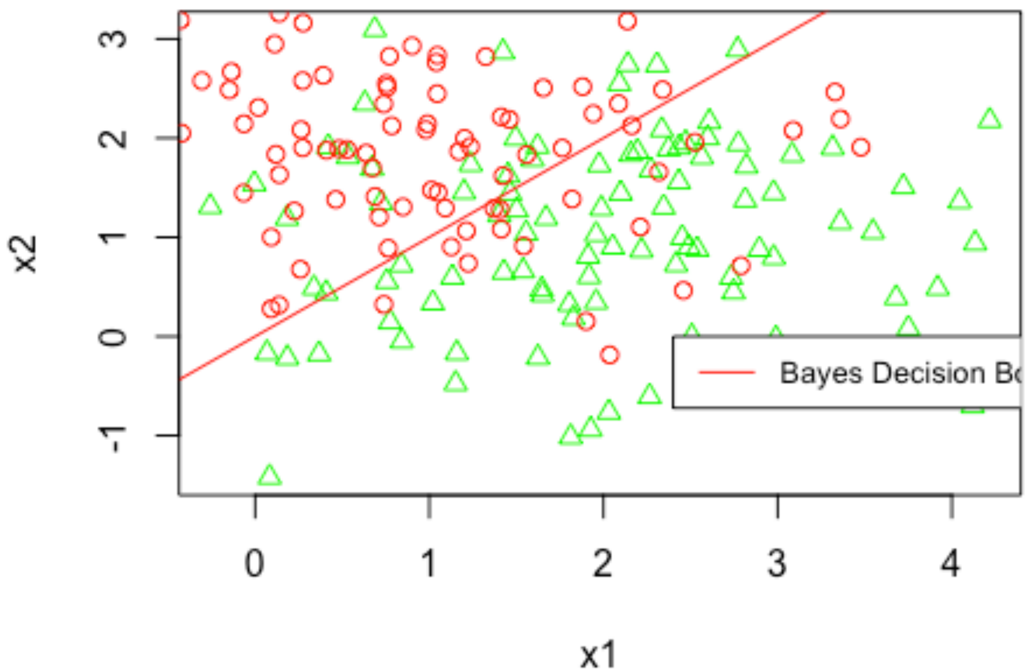
$$\tfrac{1}{2}(X-\mu_2)^T \Sigma^{-1} (X-\mu_2) - \tfrac{1}{2}(X-\mu_1)^T \Sigma^{-1} (X-\mu_1) = \tfrac{1}{2}[(X_1-\mu_{21}), (X_2-\mu_{22})]\begin{pmatrix} X_1-\mu_{21} \\ X_2-\mu_{22} \end{pmatrix} -$$
$$\tfrac{1}{2}[(X_1-\mu_{11})^2 + (X_2-\mu_{12})^2] = \tfrac{1}{2}(X_1-\mu_{21})^2 + \tfrac{1}{2}(X_2-\mu_{22})^2 - \tfrac{1}{2}(X_1-\mu_{11})^2 - \tfrac{1}{2}(X_2-\mu_{12})^2$$
$$= \tfrac{1}{2}(X_1-1)^2 + \tfrac{1}{2}(X_2-2)^2 - \tfrac{1}{2}(X_1-2)^2 - \tfrac{1}{2}(X_2-1)^2 = 0$$

, given that $\underline{X} = \binom{X_1}{X_2}$, $\mu_1 = \binom{\mu_{11}}{\mu_{12}} = \binom{2}{1}$ and $\mu_2 = \binom{\mu_{21}}{\mu_{22}} = \binom{1}{2}$. This shows that the boundary is $\{X : X_1 = X_2\}$

Hw1.q3

(b) Add the Bayes decision boundary to the scatterplot. The Bayes decision boundary is x1=x2

```
#The Bayes decision boundary is x1=x2
#plot a scatterplot and the Bayes decision boundary
library(MASS)
set.seed(2000)
miu1<-c(2,1)
miu2<-c(1,2)
sigma<-matrix(c(1,0,0,1),2,2,byrow = T)
green<-mvrnorm(100,miu1,sigma)
red<-mvrnorm(100,miu2,sigma)
#training dataset
label<-matrix(c(rep(1,100),rep(2,100)),200,1)
train_data<-rbind(green,red)
train_data<-cbind(train_data,label)
plot(green[,1],green[,2],col="green",xlab = "x1",ylab = "x2",pch=2)
points(red[,1],red[,2],col="red")
abline(a=0,b=1,col="red")
legend(2.4,0,legend="Bayes Decision Boundary",
       col="red", lty=1:2, cex=0.8)
```



(c) The training error is 0.215 and the testing error is 0.239

```
#calculate lda training error
train_data[,3]<-train_data[,3]-1
bayes.result<-matrix(NA,200,1)
for (i in 1:200) {
  if(train_data[i,1]>train_data[i,2]) {
    bayes.result[i,]=0
  } else {
    bayes.result[i,]=1
  }
}
bayes.result<-as.numeric(bayes.result)
bayes.error<-1-sum(as.numeric(bayes.result==train_data[,3]))/200
bayes.error

## [1] 0.215

#The training error rate=0.215

#calculate lda testing error
test_set<-read.table(file = "/Users/ruijieyin/Dropbox/UM Biostatistics/BST691
High Dimensional and Complex Data/hw1q2testdataset.txt")
test_set[,3]<-test_set[,3]-1
```

```r
bayes.result.test<-matrix(NA,1000,1)
for (i in 1:1000) {
  if(test_set[i,1]>test_set[i,2]) {
    bayes.result.test[i,]=0
  } else {
    bayes.result.test[i,]=1
  }
}
bayes.result.test<-as.numeric(bayes.result.test)
bayes.error.test<-1-sum(as.numeric(bayes.result.test==test_set[,3]))/1000
bayes.error.test
```

```
## [1] 0.239
```

```r
#The testing error rate=0.239
```

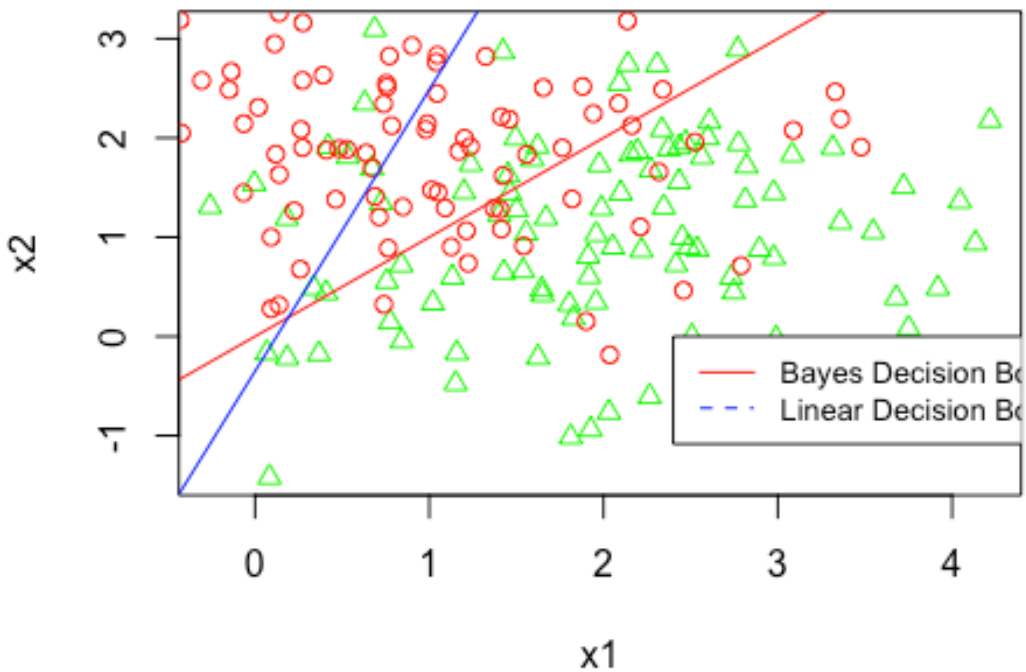
hw1.q4

(a)Train the linear regression model

```
library(MASS)
set.seed(2000)
miu1<-c(2,1)
miu2<-c(1,2)
sigma<-matrix(c(1,0,0,1),2,2,byrow = T)
green<-mvrnorm(100,miu1,sigma)
red<-mvrnorm(100,miu2,sigma)
#training dataset
label<-matrix(c(rep(1,100),rep(2,100)),200,1)
train_data<-rbind(green,red)
train_data<-cbind(train_data,label)
train<-as.data.frame(train_data)
colnames(train)<-c("x1","x2","y")
train[,3]<-train[,3]-1
#train a linear model
linear_model<-lm(y~x1+x2,data = train)
coef<-coefficients(linear_model)
```

(b)add the linear boundary to the graph

```
library(MASS)
set.seed(2000)
miu1<-c(2,1)
miu2<-c(1,2)
sigma<-matrix(c(1,0,0,1),2,2,byrow = T)
green<-mvrnorm(100,miu1,sigma)
red<-mvrnorm(100,miu2,sigma)
#training dataset
label<-matrix(c(rep(1,100),rep(2,100)),200,1)
train_data<-rbind(green,red)
train_data<-cbind(train_data,label)
plot(green[,1],green[,2],col="green",xlab = "x1",ylab = "x2",pch=2)
points(red[,1],red[,2],col="red")
abline(a=0,b=1,col="red")
#copied from Q3
#generate a linear regression boundary on this same graph
abline(a=(0.5-coef[[1]])/coef[[2]],b=-coef[[1]]/coef[[2]],col="blue")
legend(2.4,0,legend=c("Bayes Decision Boundary", "Linear Decision Boundary"),
       col=c("red", "blue"), lty=1:2, cex=0.8)
```



(c) The testing error is 0.22 and the training error is 0.235

```
#training error
linear.pred<-coef[[1]]+coef[[2]]*train[,1]+coef[[3]]*train[,2]-0.5
linear.pred<-as.matrix(linear.pred,200,1)
linear.result<-matrix(NA,200,1)
for(i in 1:200) {
  if(linear.pred[i,]>0) {
    linear.result[i,]=1
  }else {
    linear.result[i,]=0
  }
}
linear.result<-as.numeric(linear.result)
linear.train.error<-as.numeric(train[,3]==linear.result)
linear.train.error<-1-sum(linear.train.error)/200
linear.train.error

## [1] 0.22

#error rate=0.22


#testing error
```

```r
test_set<-read.table(file = "/Users/ruijieyin/Dropbox/UM Biostatistics/BST691 High Dimensional and Complex Data/hw1q2testdataset.txt")
test_set<-as.data.frame(test_set)
test_set[,3]<-test_set[,3]-1
linear.pred.test<-coef[[1]]+coef[[2]]*test_set[,1]+coef[[3]]*test_set[,2]-0.5
linear.pred.test<-as.matrix(linear.pred.test,1000,1)
linear.result.test<-matrix(NA,1000,1)
for(i in 1:1000) {
  if(linear.pred.test[i,]>0) {
    linear.result.test[i,]=1
  }else {
    linear.result.test[i,]=0
  }
}
linear.result.test<-as.numeric(linear.result.test)
linear.test.error<-as.numeric(test_set[,3]==linear.result.test)
linear.test.error<-1-sum(linear.test.error)/1000
linear.test.error
```

```
## [1] 0.235
```

```r
#error rate=0.235
```