```
#homework2
#Question1


sim_data<-read.csv("simulated.csv",header=TRUE)
simulated<-data.frame(sim_data)
sim_matrix<-data.matrix(simulated)
x1<-(simulated$x1)
x2<-(simulated$x2)
xs<-data.frame(x1,x2,y)
y<-(simulated$y)
valuesofx<-lm(y~x1+x2)


#########################################################1a
yhat <- ifelse( fitted(valuesofx)>0.5, 1, 0)
tab<-table(yhat, y)
tab[row(tab)!=col(tab)]
#[1] 30 24

length(yhat)
sum(yhat == y)
1 - sum(yhat==y)/length(y)
#[1] 0.27 <------------------- misclassification rate

x2_mis_rate <-((30/100))
#0.3<-------------------------blue misclassification rate
x1_mis_rate<-(24/100)
#0.24<------------------------orange misclassification rate

#########################################################1b
x2_lm<-lm(y~x2)
tabx1<-data.frame(yhat1,y,x1,x2)
subset(tabx1,y==0&yhat==1)
      #yhat1 y        x1       x2
#4      1 0  0.693436 0.777194
#5      1 0 -0.019837 0.867254
#8      1 0 -0.912620 1.216216
#10     1 0  1.897709 0.973755
#13     1 0  2.490979 1.148315
#17     1 0 -0.313007 1.273747
#18     1 0  2.813039 1.167760
#23     1 0  0.248728 0.855520
```

```
#25     1 0  1.467109 1.877591
#27     1 0  0.033318 1.930257
#30     1 0  2.126869 1.154180
#35     1 0  1.282879 1.276901
#37     1 0  2.489806 1.358510
#42     1 0  0.887559 0.883552
#45     1 0 -0.191272 2.493686
#47     1 0 -0.840369 1.881741
#54     1 0 -0.302275 0.886233
#56     1 0 -0.727077 1.457361
#57     1 0  0.134347 0.901676
#60     1 0  4.170746 1.079834
#62     1 0  0.205190 2.453888
#63     1 0  1.640157 1.608537
#66     1 0  0.310785 2.007982
#80     1 0  2.285265 1.008993
#85     1 0  2.892025 1.625783
#87     1 0 -0.074079 0.919702
#91     1 0  0.066332 1.580805
#93     1 0  2.551446 1.418180
#95     1 0 -0.715284 1.107884
#97     1 0  2.343731 0.910978

x2_lm<-lm(y~x1)
tabx1<-data.frame(yhat1,y,x1,x2)
dim(subset(tabx1,y==1&yhat==0))
#####################################################1c
valuesofx<-lm(y~x1+x2)
plot(x1,x2, col=ifelse(y==1,"orange", "blue"), xlab="x1", ylab="x2")
abline( (0.5-coef(valuesofx)[1])/coef(valuesofx)[3], -coef(valuesofx)[2]/coef(valuesofx)[3])

#####################################################1d
mod15 <- knn(x, xnew, g, k=15, prob=TRUE)
summary(mod15)
#Figure 2.2:
plot(x, col=ifelse(g==1,"red", "green"),xlab="x1", ylab="x2")
str(mod15)
prob <- attr(mod15, "prob")
prob <- ifelse( mod15=="1", prob, 1-prob)

px1 <- mixture.example$px1
px2 <- mixture.example$px2
prob15 <- matrix(prob, length(px1), length(px2))
```

```
contour(px1, px2, prob15, levels=0.5, labels="", xlab="x1", ylab="x2", main=
        "15-nearest neighbour")

points(x, col=ifelse(g==1, "red", "green"))
ghat15 <- ifelse(knn(x,x,k=15, cl=g)=="1", 1, 0)
sum(ghat15==g)
# [1] 169
1 - sum(ghat15==g)/length(g)
# [1] 0.155

############################################################1e
mod1 <- knn(x, xnew, k=1, cl=g, prob=TRUE)
prob <- attr(mod1, "prob")
prob <- ifelse( mod1=="1", prob, 1-prob) # prob now is voting
# fraction for "red"
prob1 <- matrix(prob, length(px1), length(px2) )
contour(px1, px2, prob1, level=0.5, labels="", xlab="x1", ylab="x2", main=
        "1-nearest neighbour")
# Adding the points to the plot:
points(x, col=ifelse(g==1, "red", "green"))


#########################################################
#question2
ortho.fun<-function(y,var1,var2) {
  z0<-rep(1,length(var1))
  fit0<-lm(z0~var1)
  z1<-var1-mean(resid(fit0))
  fit1<-lm(y~z1)
  z2<-var2-mean(resid(fit1))
  fit.final<-lm(y~z1+z2)
  fit.final$coefficients
}

ortho.fun(y,x1,x2)
#(Intercept)          z1          z2
#0.3290614  -0.0226360   0.2495983
#lm(y~x1+x2)#<-compared to OR

#########################################################################
###############
#quesion 3
```

```
stagewise.function<-function(y,xmatrix,eps,threshold){
  beta <- matrix(0,ncol=ncol(xmatrix),nrow=1)
  maxCorr = max(t(r)*xmatrix)
  r<-y-mean(y)
  while(r>maxCorr){
  co <- t(xmatrix)%*%r
  j <- (1:ncol(xmatrix))[abs(co)==max(abs(co))][1]
  delta <- eps*sign(co[j])
  b <- beta[nrow(beta),]
  b[j] <- b[j] + delta
  beta <- rbind(beta,b)
  r <- r - delta*xmatrix[,j]
  }

  coef<-solve(crossprod(xmatrix))%*%t(xmatrix)%*%y
  return (round(coef, 3))
}


p<-read.table("prostate.txt",header=TRUE)
xmatrix<-as.matrix(prostate[prostate$train, c("lcavol", "lweight", "lbph","svi")])

y<-prostate[prostate$train, c("lpsa")]

beta <- matrix(0,ncol=ncol(xmatrix),nrow=1)

y<-prostate$lpsa[prostate$train]


r<-y-mean(y)
eps <- 0.001
lots <- 10000
stagewise.function(y,xmatrix,eps,threshold)

##############################################################################
#############################
#question 4


stagewise.function<-function(y,xmatrix,eps,threshold){
  beta <- matrix(0,ncol=ncol(xmatrix),nrow=1)
```

```
  maxCorr = max(t(r)*xmatrix)
  r<-y-mean(y)
  while(r>maxCorr){
    co <- t(xmatrix)%*%r
    j <- (1:ncol(xmatrix))[abs(co)==max(abs(co))][1]
    delta <- eps*sign(co[j])
    b <- beta[nrow(beta),]
    b[j] <- b[j] + delta
    beta <- rbind(beta,b)
    r <- r - delta*xmatrix[,j]
  }
  coef<-solve(crossprod(xmatrix))%*%t(xmatrix)%*%y
  return (round(coef, 3))
}


p<-read.table("prostate.txt",header=TRUE)
xmatrix<-as.matrix(prostate[prostate$train, c("lcavol", "lweight", "lbph","svi")])

y<-prostate[prostate$train, c("lpsa")]

xmatrix <- xmatrix-xmatrix(apply(x,2,mean),ncol=ncol(x),nrow=nrow(x),byrow=T)
xmatrix <- xmatrix/xmatrix(apply(x,2,sd),ncol=ncol(x),nrow=nrow(x),byrow=T)
beta <- matrix(0,ncol=ncol(xmatrix),nrow=1)

y<-prostate$lpsa[prostate$train]


eps <- 0.001
lots <- 10000
stagewise.function(y,xmatrix,eps,threshold)


#[,1]
#lcavol  0.508
#lweight 0.448
#lbph    0.153
#svi     0.686

#The results compared to the table in the book is roughly "ballpark". Meaning that they are
roughly similar.
#I would not expect them to be totally similar. Lasso shrinks all the coefficients to zero, and will
#not include all of the variables.
```

#In the case of stagewise,the predictors star at zero, at each step, the variable most correlated with the current residual is computed,
#then adds the value nd then adds it to the current coe?cient for the current variable.
#This continues until none of the variables are correlated with the residuals.