hw8

2024-03-29

2a

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select
```

```
Y <- read.table("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/azdiabetes.dat", header=TRUE)
diabetics <- Y %>%
  filter(diabetes=="Yes")
Y_d <- diabetics[, 1:7]
nondiabetics <- Y %>%
  filter(diabetes=="No")
Y_n <- nondiabetics[, 1:7]
```

```
mu0_d <- colMeans(diabetics[, 1:7])
samp_cov_d <- cov(diabetics[, 1:7])
lambda0_d <- cov(diabetics[, 1:7])
S0_d <- samp_cov_d
nu0_d <- 9

mu0_n <- colMeans(nondiabetics[, 1:7])
samp_cov_n <- cov(nondiabetics[, 1:7])
lambda0_n <- cov(nondiabetics[, 1:7])
S0_n <- samp_cov_n
nu0_n <- 9
```

```
### Simulate multivariate normal vector
rmvnorm<-function(n,mu,Sigma)
{
  p<-length(mu)
```

```
  res<-matrix(0,nrow=n,ncol=p)
  if(n>0 & p>0)
  {
    E<-matrix(rnorm(n*p),n,p)
    res<-t(  t(E%*%chol(Sigma)) +c(mu))
  }
  res
}

### Simulate from the Wishart distribution
rwish<-function(n,nu0,S0)
{
  sS0 <- chol(S0)
  S<-array( dim=c( dim(S0),n ) )
  for(i in 1:n)
  {
     Z <- matrix(rnorm(nu0 * dim(S0)[1]), nu0, dim(S0)[1]) %*% sS0
     S[,,i]<- t(Z)%*%Z
  }
  S[,,1:n]
}
```

```
### Gibbs sampler

Sigma_d <- samp_cov_d
n_d<-dim(Y_d)[1]
S0_d <- samp_cov_d
THETA_d <- NULL
SIGMA_d <- NULL

Sigma_n <- samp_cov_n
n_n<-dim(Y_n)[1]
S0_n <- samp_cov_n
THETA_n <- NULL
SIGMA_n <- NULL

S <- 10000

for (s in 1:S){
    ###update theta_d
  Ln_d<-solve( solve(lambda0_d) + n_d*solve(Sigma_d) )
  mun_d<-Ln_d%*%( solve(lambda0_d)%*%mu0_d + n_d*solve(Sigma_d)%*%mu0_d )
  theta_d<-rmvnorm(1,mun_d,Ln_d)
  ###

    ###update Sigma_d
  Sn_d<- S0_d + ( t(Y_d)-c(theta_d) )%*%t( t(Y_d)-c(theta_d) )
  Sigma_d<-solve( rwish(1, nu0_d+n_d, solve(Sn_d)) )
  ###

    ### save results
  THETA_d<-rbind(THETA_d,theta_d) ; SIGMA_d<-rbind(SIGMA_d,c(Sigma_d))
  ###
```

```r
    ###update theta_n
  Ln_n<-solve( solve(lambda0_n) + n_n*solve(Sigma_n) )
  mun_n<-Ln_n%*%( solve(lambda0_n)%*%mu0_n + n_n*solve(Sigma_n)%*%mu0_n )
  theta_n<-rmvnorm(1,mun_n,Ln_n)
  ###

    ###update Sigma_n
  Sn_n<- S0_n + ( t(Y_n)-c(theta_n) )%*%t( t(Y_n)-c(theta_n) )
  Sigma_n<-solve( rwish(1, nu0_d+n_n, solve(Sn_n)) )
  ###

    ### save results
  THETA_n<-rbind(THETA_n,theta_n) ; SIGMA_n<-rbind(SIGMA_n,c(Sigma_n))
  ###

  if (s %% 100 == 0){
    print(s)
  }

}
```

```
## [1] 100
## [1] 200
## [1] 300
## [1] 400
## [1] 500
## [1] 600
## [1] 700
## [1] 800
## [1] 900
## [1] 1000
## [1] 1100
## [1] 1200
## [1] 1300
## [1] 1400
## [1] 1500
## [1] 1600
## [1] 1700
## [1] 1800
## [1] 1900
## [1] 2000
## [1] 2100
## [1] 2200
## [1] 2300
## [1] 2400
## [1] 2500
## [1] 2600
## [1] 2700
## [1] 2800
## [1] 2900
```

```
## [1] 3000
## [1] 3100
## [1] 3200
## [1] 3300
## [1] 3400
## [1] 3500
## [1] 3600
## [1] 3700
## [1] 3800
## [1] 3900
## [1] 4000
## [1] 4100
## [1] 4200
## [1] 4300
## [1] 4400
## [1] 4500
## [1] 4600
## [1] 4700
## [1] 4800
## [1] 4900
## [1] 5000
## [1] 5100
## [1] 5200
## [1] 5300
## [1] 5400
## [1] 5500
## [1] 5600
## [1] 5700
## [1] 5800
## [1] 5900
## [1] 6000
## [1] 6100
## [1] 6200
## [1] 6300
## [1] 6400
## [1] 6500
## [1] 6600
## [1] 6700
## [1] 6800
## [1] 6900
## [1] 7000
## [1] 7100
## [1] 7200
## [1] 7300
## [1] 7400
## [1] 7500
## [1] 7600
## [1] 7700
## [1] 7800
## [1] 7900
## [1] 8000
## [1] 8100
## [1] 8200
## [1] 8300
```

```
## [1] 8400
## [1] 8500
## [1] 8600
## [1] 8700
## [1] 8800
## [1] 8900
## [1] 9000
## [1] 9100
## [1] 9200
## [1] 9300
## [1] 9400
## [1] 9500
## [1] 9600
## [1] 9700
## [1] 9800
## [1] 9900
## [1] 10000
```

```r
colMeans(THETA_n) - colMeans(THETA_d)
```

```
##       npreg         glu          bp        skin         bmi         ped
##  -1.7756579 -33.0601339  -4.8013461  -5.6817419  -4.3842703  -0.1699392
##         age
##  -7.2096250
```

```r
apply(THETA_n, 2, var) - apply(THETA_d, 2, var)
```

```
##         npreg           glu            bp          skin           bmi
## -0.0653479875 -3.9133928337 -0.4587850935 -0.3142579553 -0.1214909829
##           ped           age
## -0.0006571537 -0.3856434809
```

The glucose of diabetics seems to be, on average, much higher than that of non-diabetics. Additionally, the variance of glucose is much greater for diabetics than it is for non-diabetics.

```r
colMeans(THETA_d > THETA_n)
```

```
## npreg   glu    bp  skin   bmi   ped   age
##     1     1     1     1     1     1     1
```

$Pr(\theta_{d,j} > \theta_{n,j}|Y) = 1$ for all j $\in \{1, 2, 3, 4, 5, 6, 7\}$.

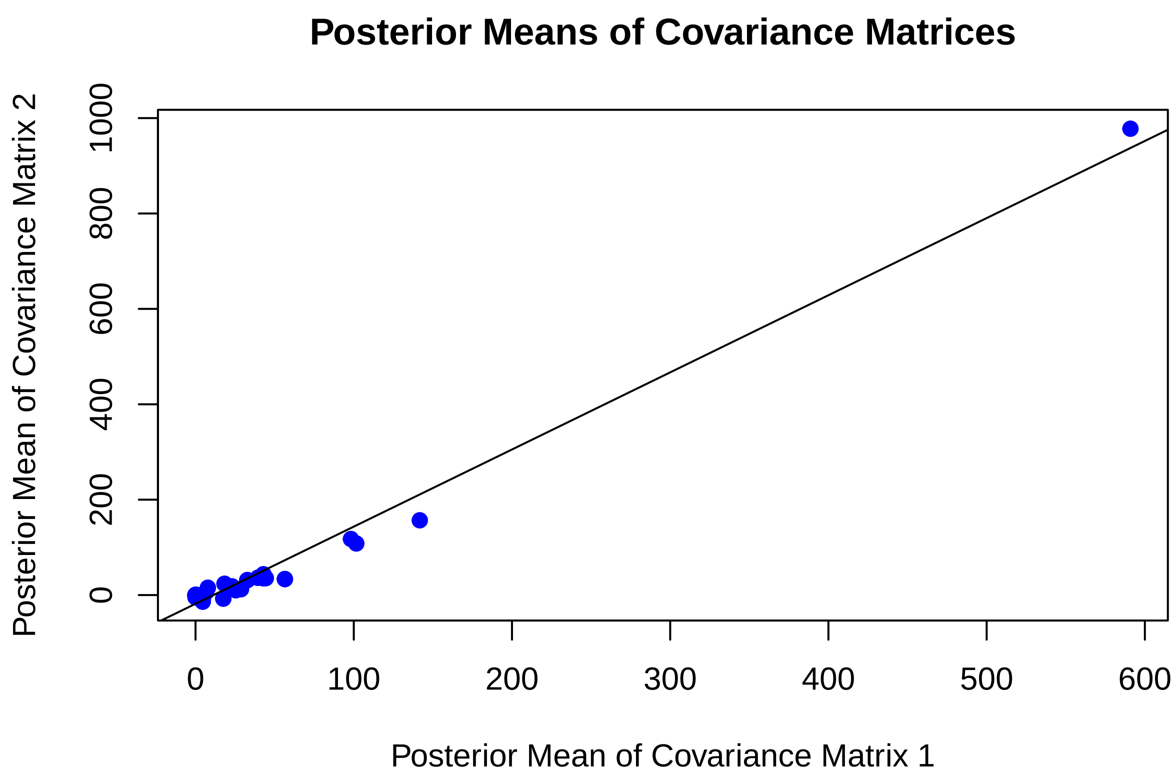
2b

```r
colMeans(SIGMA_n) - colMeans(SIGMA_d)
```

```
##  [1]   -7.60154621   14.38438901    0.51952251    7.77626691    4.65362092
##  [6]    0.05342474   -5.23881308   14.38438901 -386.84365430   22.92976838
## [11]    1.36213178   15.15711759    0.41321728    7.73697657    0.51952251
## [16]   22.92976838  -15.05849430   16.29929566    5.05912658    0.04328221
## [21]    3.17189992    7.77626691    1.36213178   16.29929566   -6.54805383
## [26]    8.81520427   -0.48356095   24.91181706    4.65362092   15.15711759
## [31]    5.05912658    8.81520427   -0.76124061   -0.29514813   18.24530299
## [36]    0.05342474    0.41321728    0.04328221   -0.48356095   -0.29514813
## [41]   -0.06985385    0.21283050   -5.23881308    7.73697657    3.17189992
## [46]   24.91181706   18.24530299    0.21283050  -19.30694013
```

The 2,2 entry corresponds to glucose, so this supports our observation that the variability in glucose is especially high in diabetics.

```
plot(colMeans(SIGMA_n), colMeans(SIGMA_d),
     xlab = "Posterior Mean of Covariance Matrix 1",
     ylab = "Posterior Mean of Covariance Matrix 2",
     main = "Posterior Means of Covariance Matrices",
     pch = 19, col = "blue")
line <- lm(colMeans(SIGMA_d)~colMeans(SIGMA_n))
abline(line)
```



The entries of the two covariance matrices are positively correlated. This visualization makes it apparent that the variance of glucose is high for both groups when compared to that of the other covariates.

4b

```
Y<-dget(url("http://www2.stat.duke.edu/~pdh10/FCBS/Inline/Y.pima.miss"))
```

```
Y=Y%>%
  filter(!is.na(glu))%>%
  filter(!is.na(bp))%>%
  filter(!is.na(skin))%>%
  filter(!is.na(bmi))
colMeans(Y)
```

```
##       glu        bp      skin       bmi
## 121.85039  70.61417  28.83465  31.67874
```

```
###
Y <- readRDS("hw8train.rds")
```

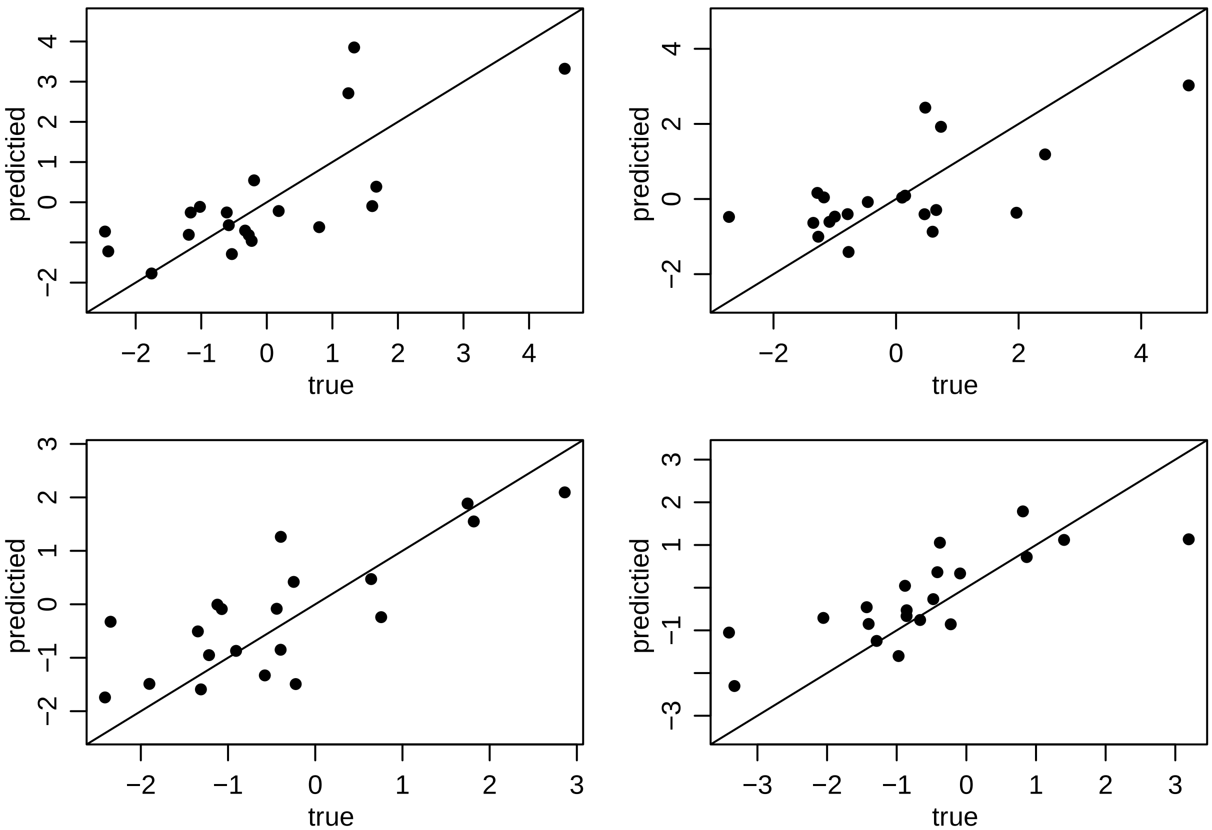
```
### prior parameters
n<-dim(Y)[1] ; p<-dim(Y)[2]
mu0<-c(rep(0,14))
sd0<-(mu0/2)
L0<-matrix(0,p,p) ; diag(L0)<-1 # ; L0<-L0*outer(sd0,sd0)
nu0<-p+2 ; S0<-L0
###

### starting values
Sigma<-S0
Y.full<-Y
O<-1*(!is.na(Y))
for(j in 1:p)
{
  Y.full[is.na(Y.full[,j]),j]<-mean(Y.full[,j],na.rm=TRUE)
}
###

### Gibbs sampler
THETA<-SIGMA<-Y.MISS<-NULL
set.seed(1)
S<-1000
for(s in 1:S)
{

  ###update theta
  ybar<-apply(Y.full,2,mean)
  Ln<-solve( solve(L0) + n*solve(Sigma) )
  mun<-Ln%*%( solve(L0)%*%mu0 + n*solve(Sigma)%*%ybar )
  theta<-rmvnorm(1,mun,Ln)
  ###

  ###update Sigma
  Sn<- S0 + ( t(Y.full)-c(theta) )%*%t( t(Y.full)-c(theta) )
  Sigma<-solve( rwish(1, nu0+n, solve(Sn)) )
  ###

  ###update missing data
  for(i in 60:n)
  {
    b <- ( O[i,]==0 )
    #print(b)
    a <- ( O[i,]==1 )
    #print(a)
    iSa<- solve(Sigma[a,a])
    beta.j <- Sigma[b,a]%*%iSa
    Sigma.j   <- Sigma[b,b] - Sigma[b,a]%*%iSa%*%Sigma[a,b]
    #print(dim(beta.j))
    #print(dim(t(Y.full[i,a])))
    #print(theta)
    #print(dim(theta))
```

```
    #print(theta[a])
    #print(dim(theta[a]))
    #print(matrix(theta[a]))
    theta.j<- matrix(theta[b]) + beta.j%*%matrix((t(Y.full[i,a])-theta[a]))
    Y.full[i,b] <- rmvnorm(1,theta.j,Sigma.j )
  }

  ### save results
  THETA<-rbind(THETA,theta) ; SIGMA<-rbind(SIGMA,c(Sigma))
  Y.MISS<-rbind(Y.MISS, Y.full[O==0] )
  ###
}
```

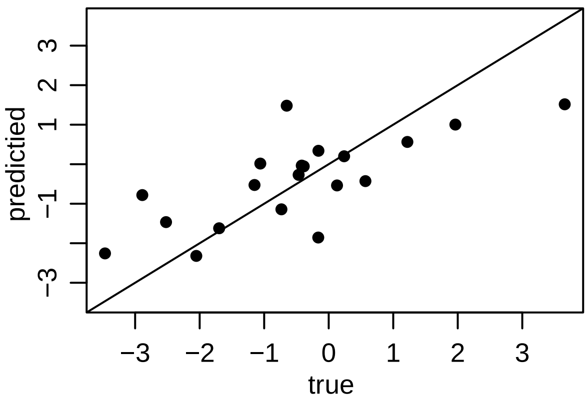
4c

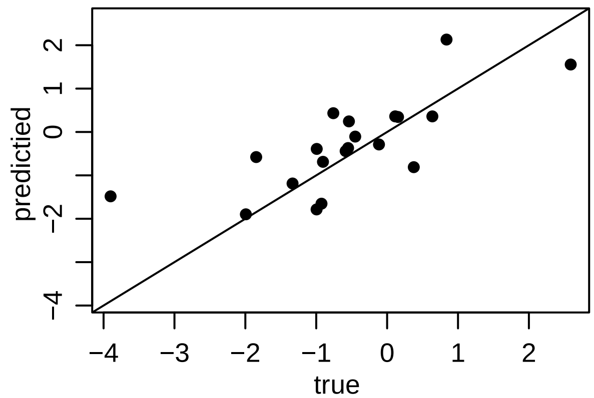
```
## compare to test dataset
Y.true<-readRDS("hw8test.rds")

V<-matrix(1:p,nrow=n,ncol=p,byrow=TRUE)

v.miss<-V[O==0]
y.pred<-apply(Y.MISS,2,mean)
y.true<-Y.true#[O==0]
par(mfrow=c(2,2),mar=c(3,3,1,1),mgp=c(1.75,.75,0))
for(j in 8:p){
  #print(y.pred[v.miss==j])
  #print(y.true[v.miss==j])
  plot(y.true[v.miss==j], y.pred[v.miss==j],
          xlab=paste("true", colnames(Y.true)[j]),
          ylab=paste("predictied", colnames(Y.true)[j]),pch=16,
          xlim=range(c(y.pred[v.miss==j],y.true[v.miss==j])),
          ylim=range(c(y.pred[v.miss==j],y.true[v.miss==j])))
          abline(0,1)
  cat(j, mean( (y.true[v.miss==j]- y.pred[v.miss==j])^2),
         mean( (y.true[v.miss==j]- mean(Y[,j],na.rm=TRUE))^2),"\n")
}
```

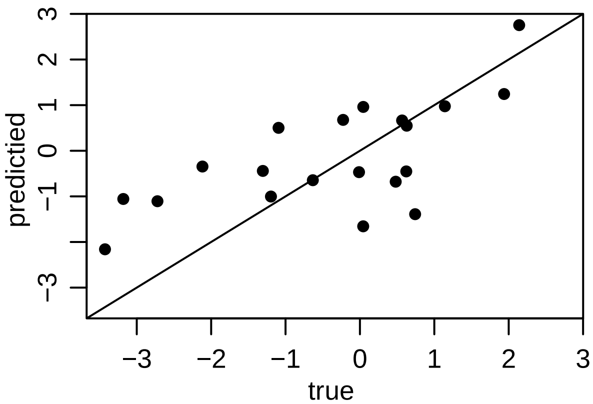
```
## 8 1.25844 2.521915

## 9 1.476817 2.600428

## 10 0.7580454 1.963907
```




```
## 11 0.9825413 2.7985

## 12 1.207481 2.759375

## 13 0.780527 1.715279

## 14 1.394759 2.389352
```







The error of our predicted values is less than the error of $\hat{\theta}_B$.