

# HW 7

Bingying Liu

April 8, 2020

## Question 1:

Part(a): Using the g-prior with  $g = n = 6$ , generate samples/realizations from the prior predictive distribution for a single swimmer over the 12 weeks ( $W=2,4,6,8,10,12$ ) and create a density plot of the predictive draws (one for each  $W$ ). Are the values plausible?

```
Y <- read.table("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/swim.dat")
Y <- t(Y)

# data summaries
n = nrow(Y)
n_swimmers = ncol(Y)
g = n
W = seq(2,12,length.out=n)
X = cbind(rep(1,n),(W-mean(W)))
p = ncol(X)

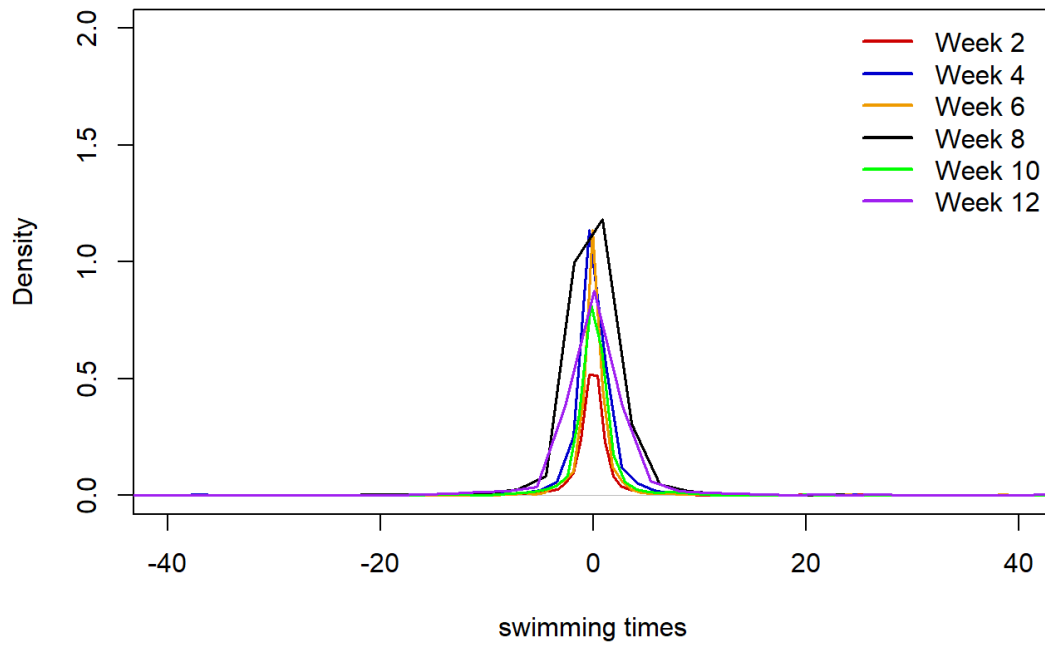
# Hyperparameters for the prior
nu_0 = 1
sigma_0_sq = 1/10
beta_0 = c(0,0)

n_iter = 1000
prior_predict = matrix(0, nrow=n_iter, ncol=nrow(X))
for (i in 1:n_iter){
  # Prior predictive distribution
  SIGMA_SQ = 1/rgamma(1,nu_0/2,nu_0*sigma_0_sq/2)
  Sigma_0 = g*SIGMA_SQ*solve(t(X)%*%X)
  BETA = rmvnorm(1, beta_0, Sigma_0)
  prior_predict[i,] = rmvnorm(1, X%*%t(BETA), SIGMA_SQ*diag(1,nrow=n)) # save different weeks
}

colnames(prior_predict) = c('Week 2','Week 4','Week 6','Week 8','Week 10','Week 12')

plot(density(prior_predict[, 'Week 2']),col="red3",xlim=c(-40,40),ylim=c(0,2),lwd=1.5,
      main="Prior Predictive Distributions",xlab="swimming times")
legend("topright",2,c('Week 2','Week 4','Week 6','Week 8','Week 10','Week 12'),col=c("red3","blue3","orange2","black",
      "green","purple"),lwd=2,bty="n")
lines(density(prior_predict[, "Week 4"]),col="blue3",lwd=1.5)
lines(density(prior_predict[, "Week 6"]),col="orange2",lwd=1.5)
lines(density(prior_predict[, "Week 8"]),col="black",lwd=1.5)
lines(density(prior_predict[, "Week 10"]),col="green",lwd=1.5)
lines(density(prior_predict[, "Week 12"]),col="purple",lwd=1.5)
```

### Prior Predictive Distributions



Yes, the values are plausible because since we're using a g prior, the prior predictive distribution all center around 0.

Part(b): Using the data, and the g-prior with  $g = n = 6$  for each swimmer, give the posterior distributions of  $\beta_0$ ,  $\beta_1$  and  $\sigma^2$  for each swimmer.

```

# set number of iterations
S <- 10000
BETA = array(0,c(n_swimmers, S, p))
SIGMA_SQ = matrix(0,n_swimmers,S)

# Gibbs sampler
for (j in 1:n_swimmers){
  beta_ols <- solve(t(X)%*%X)%*%t(X)%*%Y[,j]

  # sample sigma_sq
  nu_n <- nu_0 + n
  Hg <- (g/(g+1))* X%*%solve(t(X)%*%X)%*%t(X)
  SSRg <- t(Y[,j])%*(diag(1,nrow=n) - Hg)%*%Y[,j]
  nu_n_sigma_n_sq <- nu_0*sigma_0_sq + SSRg
  sigma_sq <- 1/rgamma(S,(nu_n/2),(nu_n_sigma_n_sq/2))

  # sample beta
  mu_n <- g*beta_ols/(g+1)
  beta <- matrix(nrow=S,ncol=p)
  for(s in 1:S){
    Sigma_n <- g*sigma_sq[s]*solve(t(X)%*%X)/(g+1)
    beta[s,] <- rmvnorm(1,mu_n,Sigma_n)
  }

  BETA[j,,] = beta
  SIGMA_SQ[j,] = sigma_sq
}

# posterior summaries: means
beta_postmean <- t(apply(BETA,c(1,3),mean))
colnames(beta_postmean) <- c("Swimmer 1","Swimmer 2","Swimmer 3","Swimmer 4")
rownames(beta_postmean) <- c("beta_0","beta_1")
beta_postmean

```

```

##           Swimmer 1   Swimmer 2   Swimmer 3   Swimmer 4
## beta_0 19.67429187 20.01411681 19.43440000 20.18126890
## beta_1 -0.04774097  0.01953296  0.01719867 -0.03140066

```

Part(c): For each swimmer  $j$ , plot their posterior predictive distributions for a future time  $T^*$  two weeks after the last recorded observation (overlay the 4 densities in a single plot).

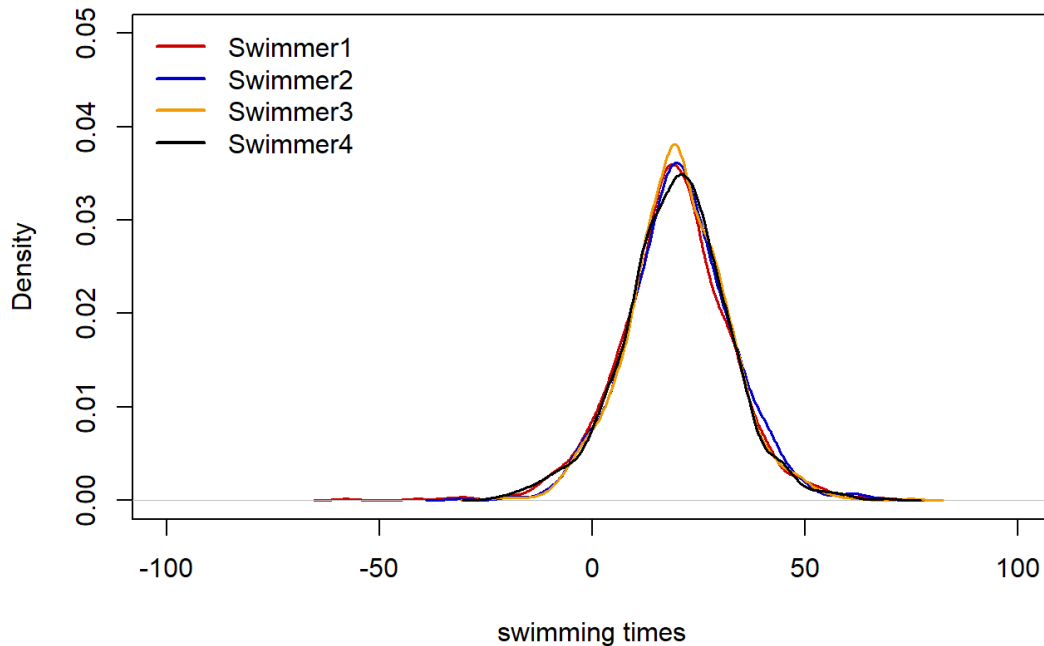
```

x_new <- matrix(c(1,(14-mean(W))),ncol=1)
post_pred <- matrix(0,nrow=n_iter,ncol=n_swimmers)
for(j in 1:n_swimmers){
  post_pred[,j] <- rnorm(n_iter,BETA[j,,]%*%x_new,sqrt(SIGMA_SQ[j,]))
}
colnames(post_pred) <- c("Swimmer 1","Swimmer 2","Swimmer 3","Swimmer 4")

plot(density(post_pred[, "Swimmer 1"]),col="red3",xlim=c(-100,100),ylim=c(0,0.05),lwd=1.5,
      main="Predictive Distributions",xlab="swimming times")
legend("topleft",2,c("Swimmer1", "Swimmer2", "Swimmer3", "Swimmer4"),col=c("red3", "blue3", "orange2", "black"),lwd=2,bty=
      "n")
lines(density(post_pred[, "Swimmer 2"]),col="blue3",lwd=1.5)
lines(density(post_pred[, "Swimmer 3"]),col="orange2",lwd=1.5)
lines(density(post_pred[, "Swimmer 4"]),lwd=1.5)

```

## Predictive Distributions



Part(d): The coach of the team has to recommend which of the swimmers to compete in a swim meet in two weeks time. Using draws from the predictive distributions, compute  $P(Y_j^* = \max(Y_1^*, Y_2^*, Y_3^*, Y_4^*))$  for each swimmer  $j$ , and based on this make a recommendation to the coach.

```
post_pred_min <- as.data.frame(apply(post_pred,1,function(x) which(x==max(x))))
colnames(post_pred_min) <- "Swimmers"
post_pred_min$Swimmers <- as.factor(post_pred_min$Swimmers)
levels(post_pred_min$Swimmers) <- c("Swimmer 1","Swimmer 2","Swimmer 3","Swimmer 4")
table(post_pred_min$Swimmers)/n_iter
```

```
##
## Swimmer 1 Swimmer 2 Swimmer 3 Swimmer 4
##      0.237      0.269      0.255      0.239
```

I recommend swimmer 1 to the coach since the maximum swimming time for swimmer 1 is lowest among all four swimmers.

## Hoff 9.2: Model selection

Part(a): Fit a regression model using the g-prior with  $g = n$ ,  $\nu_0 = 2$  and  $\sigma_0^2 = 1$ . Obtain the posterior confidence intervals for all of the parameters.

```

az = read.table("azdiabetes.dat.txt", header = TRUE)[-8]

# glu ~ 1 + npreg + bp + skin + bmi + ped + age
### data and priors
n = nrow(az)
intercept = as.matrix(rep(1, n), ncol = 1)
colnames(intercept) = c("intercept")
rownames(intercept) = 1:nrow(az)
X = cbind(intercept, as.matrix(az[, -2])) #delete glu col
Y = as.matrix(az[, 2], ncol = 1)
p = ncol(X)
nu_0 = 2
sigma_0_sq = 1
g = n

# Hyperparameters for the prior
beta_ols = solve(t(X)%*%X)%*%t(X)%*%Y

# MC sampling
S = 10000

# sample sigma_sq
nu_n <- nu_0 + n
Hg <- (g/(g+1))* X%*%solve(t(X)%*%X)%*%t(X)
SSRg <- t(Y)%*%(diag(1,nrow=n) - Hg)%*%Y
nu_n_sigma_n_sq <- nu_0*sigma_0_sq + SSRg
sigma_sq <- 1/rgamma(S,(nu_n/2),(nu_n_sigma_n_sq/2))

# sample beta
mu_n <- g*beta_ols/(g+1)
beta <- matrix(nrow=S,ncol=p)
for(s in 1:S){
  Sigma_n <- g*sigma_sq[s]*solve(t(X)%*%X)/(g+1)
  beta[s,] <- rmvnorm(1,mu_n,Sigma_n)
}

# posterior summaries
colnames(beta) <- colnames(X)
mean_beta <- apply(beta,2,mean)
round(mean_beta,4)

```

```

## intercept      npreg      bp      skin      bmi      ped      age
## 52.1656    -0.6509    0.2043    0.1906    0.6480    10.4850    0.7645

```

```

round(apply(beta,2,function(x) quantile(x,c(0.025,0.975))),4)

```

```

##      intercept      npreg      bp      skin      bmi      ped      age
## 2.5%      35.127 -1.6288 -0.0168 -0.1189 0.1520  3.3186 0.4546
## 97.5%      69.233  0.3238  0.4271  0.5021 1.1441 17.6413 1.0753

```

```

sigma_sq_mx <- as.matrix(sigma_sq,nrow=S)
colnames(sigma_sq_mx) <- 'sigma^2'
apply(sigma_sq_mx, 2, function(x) quantile(x,c(0.025,0.975)))

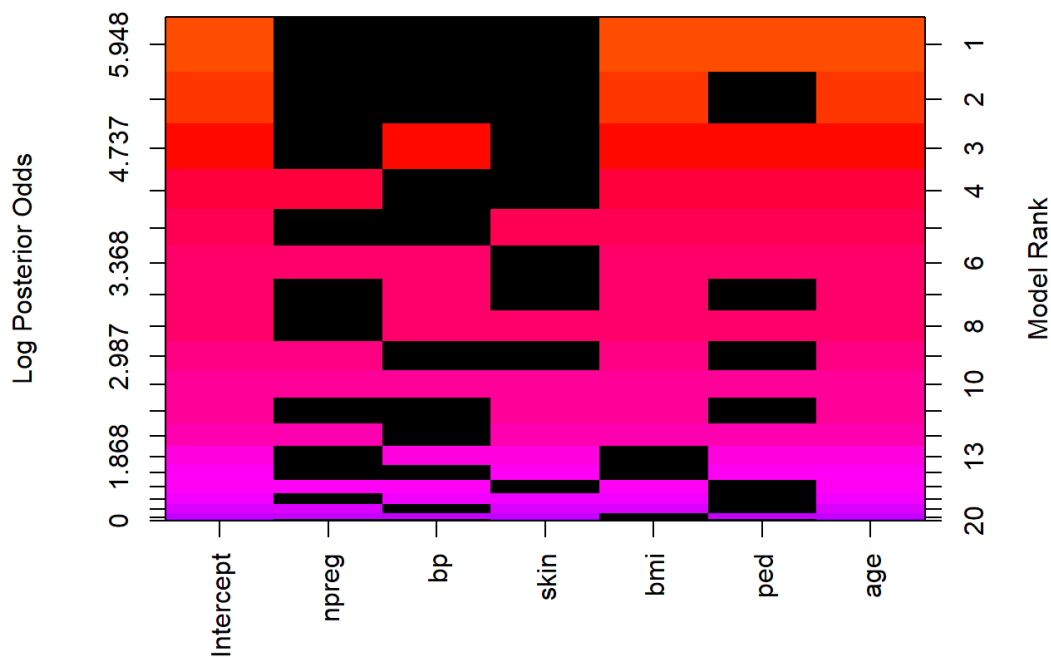
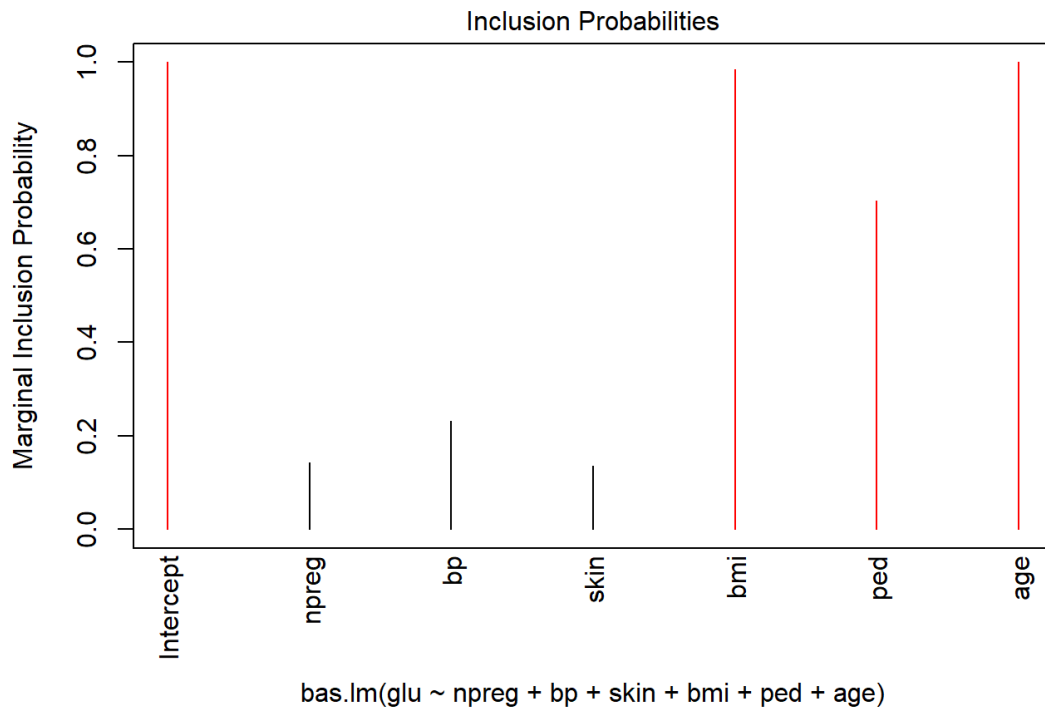
```

```

##      sigma^2
## 2.5% 745.8955
## 97.5% 948.3742

```

Part(b): Perform the model selection and averaging procedure described in Section 9.3. Obtain  $Pr(\beta_j \neq 0|y)$ , as well as posterior confidence intervals for all of the parameters. Compare to the results in part a).



```
coef(Data_bas) # Pr(Beta_j != 0)
```

```
##
## Marginal Posterior Summaries of Coefficients:
##
## Using BMA
##
## Based on the top 64 models
##      post mean  post SD  post p(B != 0)
## Intercept 121.03008  1.24993  1.00000
## npreg     -0.09522  0.29819  0.14362
## bp        0.04758  0.10266  0.23233
## skin      0.02998  0.10059  0.13595
## bmi       0.90862  0.24219  0.98400
## ped       7.38061  5.69703  0.70388
## age       0.73746  0.13594  0.99999
```

```
confint(coef(Data_bas)) # posterior confidence intervals
```

```
##           2.5%      97.5%      beta
## Intercept 118.563948735 123.4272764 121.03007519
## npreg     -0.975241884  0.0000000  -0.09522102
## bp        0.000000000  0.3045498  0.04757862
## skin      -0.001206578  0.3204308  0.02997921
## bmi       0.457790777  1.3744172  0.90861759
## ped       0.000000000 15.9225798  7.38061463
## age       0.464670284  0.9942756  0.73746007
## attr(,"Probability")
## [1] 0.95
## attr(,"class")
## [1] "confint.bas"
```

In part (b) since we have marginal inclusion probability (MIP) of each parameter through model averaging, we can see that the CIs for coefficients with low MIP are narrower and closer to 0 compared to CIs in part(a) with only a single model. The coefficients of low MIP parameters are almost 0 (which indicates we should not include those parameters, i.e. npreg, bp, skin). Also part(b)'s CI of intercept is wider than that of part(a)'s intercept.