2024 ST440/540 Exam 1 Solution

(1) In this exam, I study NBA's Steph Curry's jump shots to guide defensive stratety. The data are from the R package `hoopR` for 2009-2024. There are n=8946 shot attempts and the variables are

X := Home (X=1) versus road (X=0) game
Y := Distance of the shot (ft)
Z := Score differential with Z = -1 if Curry's team is losing by 5+ points,
Z = 1 if Curry's team is winning by 5+ points and
Z = 0 if the score is within 5.

(2) I selected a normal distribution for the distances, $Y_i|\mu \sim \text{Normal}(\mu, \sigma^2)$, independent over i. The data are continuous and far enough above zero that a normal distribution is justified. I fixed the standard deviation at the sample standard deviation, $\sigma = 6.47$. The unknown parameter μ is a real number so I selected a conjugate normal prior $\mu \sim \text{Normal}(\theta, \sigma^2/m)$. To give an uninformative prior I set $\theta=m=0$. The posterior distribution is $\mu|Y_1,\ldots,Y_n \sim \text{Normal}(\bar{Y}, \sigma^2/n)$ where $\bar{Y}$ is the sample mean of $Y_1,\ldots,Y_n$.

$$p(\mu|Y_1,\ldots,Y_n) \propto \left[\prod_{i=1}^n f(y_i|\mu)\right]\pi(\mu) \propto \prod_{i=1}^n e^{-\frac{1}{2\sigma^2}\sum_{i=1}^n (y_i-\mu)^2}$$

$$\propto e^{-\frac{1}{2\sigma^2}\left(-2\left(\sum_{i=1}^n y_i\right)\mu + n\mu^2\right)} \propto e^{-\frac{n}{2\sigma^2}\left(-2\bar{Y}\mu + \mu^2\right)}$$

$$\propto e^{-\frac{n}{2\sigma^2}(\mu - \bar{Y})^2} \qquad \bar{Y} = \sum_{i=1}^n Y_i/n$$

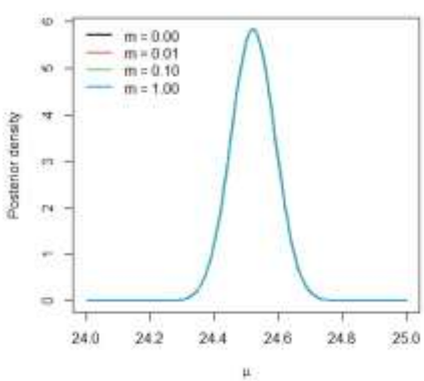
(3) Below is the posterior distribution for several values of m (all have $\theta = 0$). The posterior is virtually identical for all values of m and so the posterior is insensitive to the prior. For the uninformative prior with m=0 the mean estimate is 24.52 feet with 95% credible set (24.38, 24.66).

(4) The PDF (below) of the data (black) and fitted normal distribution (red) with mean 24.52 (posterior mean) and standard deviation 0.07 (the fixed value) show the fit is decent, but the real data are skewed.

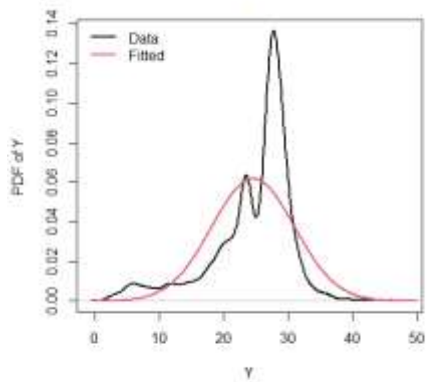
(5) Let Δ be the difference between the mean distance at home and the mean distance on the road. The posterior of Δ is plotted below. We test H_0: $\Delta < 0$ versus H_1: $\Delta > 0$, i.e., we test whether the average distance is longer at home (H_1) versus the road (H_0). The Monte Carlo approximation of the posterior probability of H_0 is 0.997, so there is strong evidence he shoots longer shots on average on the road.

(6) The analysis in (5) is repeated for the three levels of Z. Let Δ_L, Δ_C and Δ_W denote the difference between the average distance at home versus road for the three levels of Z. The posteriors of Δ_L, Δ_C and Δ_W are plotted below. The posterior probability that he shoots longer shots on the road is 0.990 when losing, 0.990 when the game is close, and 0.861 when winning, therefore the results are somewhat different when winning. The largest difference is between winning and losing conditions with the posterior probability that $\Delta_W > \Delta_L$ is 0.90, so there is some but not overwhelming evidence of a differential effect.

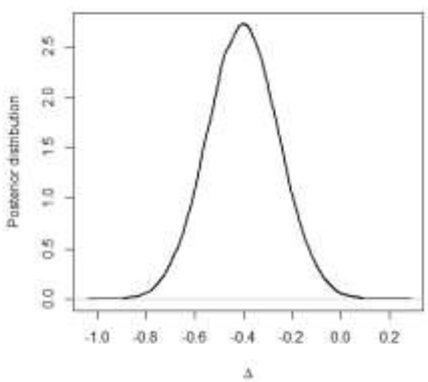
Problem (3)



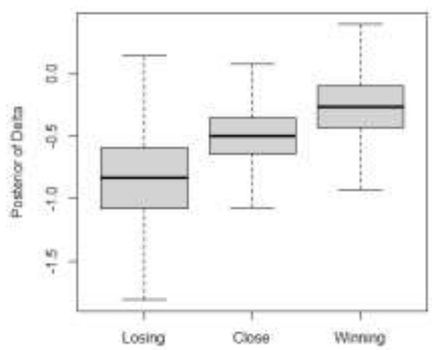
Problem (4)



Problem (5)



Problem (6)



```
# Code
rm(list=ls())

# Load the data
 library(hoopR)
 nba_pbp <- hoopR::load_nba_pbp(season=2009:2024)   # NBA play-by-play data
 keep    <- nba_pbp$type_text == "Jump Shot" &      # Extract jump shots
            nba_pbp$athlete_id_1 == 3975            # by Steph Curry
 keep    <- ifelse(is.na(keep),FALSE,keep)
 SC      <- nba_pbp[keep,]
 X       <- ifelse(SC$home_team_abbrev=="GS",1,0)   # X = 1 for a home game
 Z       <- ifelse(X==1,1,-1)*(SC$home_score - SC$away_score)
 Z       <- ifelse(Z< -5, -1,0) + ifelse(Z > 5,1,0) # Z gives the score diff
 loc     <- cbind(SC$coordinate_y,44-abs(SC$coordinate_x))
 plot(loc)
 Y       <- sqrt(loc[,1]^2 + loc[,2]^2)             # Shot distance
 sigma   <- sd(Y)                                   # Pretend sigma is known

# Function to compute the posterior mean and sd of a normal mean
#    Y is the data; sigma is the (known) sd of the data,
#    the prior is mu ~ Normal(theta,sigma/sqrt(m))
normal_normal <- function(Y,sigma,theta=0,m=0){
  n       <- length(Y)
  w       <- n/(n+m)
  out     <- list(post_mn = w*mean(Y) + (1-w)*theta,
                  post_sd = sigma/sqrt(n+m))
return(out)}

# Code for (3)

sigma <- sd(Y)

fit1 <- normal_normal(Y,sigma,m=0.00)
fit2 <- normal_normal(Y,sigma,m=0.01)
fit3 <- normal_normal(Y,sigma,m=0.10)
fit4 <- normal_normal(Y,sigma,m=1.00)

mu   <- seq(24,25,.01)
plot( mu,dnorm(mu,fit1$post_mn,fit1$post_sd),type="l",col=1,lwd=2,
      xlab=expression(mu),ylab="Posterior density",main="Problem (3)")
lines(mu,dnorm(mu,fit2$post_mn,fit2$post_sd),type="l",col=2,lwd=2)
lines(mu,dnorm(mu,fit3$post_mn,fit3$post_sd),type="l",col=3,lwd=2)
lines(mu,dnorm(mu,fit4$post_mn,fit4$post_sd),type="l",col=4,lwd=2)

legend("topleft",paste("m
=",c("0.00","0.01","0.10","1.00")),lwd=2,col=1:4,bty="n")

# Code for (4)

plot(density(Y),xlab="Y",ylab="PDF of Y",lwd=2 ,main="Problem (4)")
y <- seq(0,50,.1)
lines(y,dnorm(y,fit1$post_mn,sigma),col=2,lwd=2)
legend("topleft",c("Data","Fitted"),lwd=2,col=1:2,bty="n")
```

```
# Code for (5)
S     <- 100000
fitX0 <- normal_normal(Y[X==0],sigma)
fitX1 <- normal_normal(Y[X==1],sigma)
delta <- rnorm(S,fitX1$post_mn,fitX1$post_sd)-
         rnorm(S,fitX0$post_mn,fitX1$post_sd)
plot(density(delta),xlab=expression(Delta),ylab="Posterior
distribution",lwd=2,main="Problem (5)")
mean(delta>0)

# Code for (6)

 # When losing
 fitX0 <- normal_normal(Y[X==0 & Z==-1],sigma)
 fitX1 <- normal_normal(Y[X==1 & Z==-1],sigma)
 deltaL <- rnorm(S,fitX1$post_mn,fitX1$post_sd)-
           rnorm(S,fitX0$post_mn,fitX1$post_sd)

 # When close
 fitX0 <- normal_normal(Y[X==0 & Z==0],sigma)
 fitX1 <- normal_normal(Y[X==1 & Z==0],sigma)
 deltaC <- rnorm(S,fitX1$post_mn,fitX1$post_sd)-
           rnorm(S,fitX0$post_mn,fitX1$post_sd)

 # When winning
 fitX0  <- normal_normal(Y[X==0 & Z==1],sigma)
 fitX1  <- normal_normal(Y[X==1 & Z==1],sigma)
 deltaW <- rnorm(S,fitX1$post_mn,fitX1$post_sd)-
           rnorm(S,fitX0$post_mn,fitX1$post_sd)

 delta  <- cbind(deltaL,deltaC,deltaW)
 colnames(delta) <- c("Losing","Close","Winning")
 boxplot(delta,outline=FALSE,ylab="Posterior of Delta",main="Problem (6)")

 mean(deltaW>deltaC)
 mean(deltaW>deltaL)
 mean(deltaC>deltaL)
```