HW4

```r
library(readr)
library(invgamma)
```

```r
cookies_dat <- read.csv("C:/Users/isaac/Downloads/cookies_dat.csv")
```

Part B

```r
k <- .1
n <- 25
mu <- 130
y_bar <- mean(cookies_dat$x)
v <- .1
y_2 <- 3.5^2
v_star <- v + n
s_2 <- var(cookies_dat$x)
y_star_2 <- 1/v_star * ((v*y_2) + (n-1)*s_2 + k*n/(k+n)*(y_bar - mu)^2)
samp_var <- rinvgamma(100000, v_star/2, v_star * y_star_2 / 2)
#Expected value of inverse gamma (sampling dist for variance)
ex_invgamma <- (v_star * y_star_2 / 2) / (v_star/2 - 1)
mu_star <- (k/(k+n))*mu + (n/(n+k))*y_bar
tao_star_2 <- samp_var/(k+n)
#Expected variance for posterior sampling dist for theta
ex_tao_star_2 <- ex_invgamma / (k+n)

theta.mc1000 <- rnorm(1000, mu_star, sqrt(tao_star_2))
theta.mc1000_ex <- rnorm(1000, mu_star, sqrt(ex_tao_star_2))

#randomly sampled theta < 135
mean(theta.mc1000 <= 135)
```

```
## [1] 0.89
```

```r
#expected theta < 135
mean(theta.mc1000_ex <= 135)
```

```
## [1] 0.873
```

Part C

```r
B <- rt(10000, 25.1)
#mean(pnorm(135, mu_star, sqrt(y_star_2 * B^2/(k+n))))
mean(mu_star + sqrt(y_star_2/(k+n)) * B <= 135)
```

```
## [1] 0.8915
```

Part D

```r
theta_post <- rnorm(10000, mu_star, sqrt(tao_star_2/25.1))
sig_2_post <- rinvgamma(10000, v_star/2, v_star * y_star_2 / 2)
z <- rnorm(10000, theta_post, sqrt(sig_2_post))
mean(z <= 135)
```

```
## [1] 0.5958
```

Part E

```r
a <- rnorm(10000, 0, sqrt(sig_2_post))
mean(mu_star + sqrt(y_star_2/(k+n)) * B + a <= 135)
```

```
## [1] 0.5983
```

Question 2

Part B

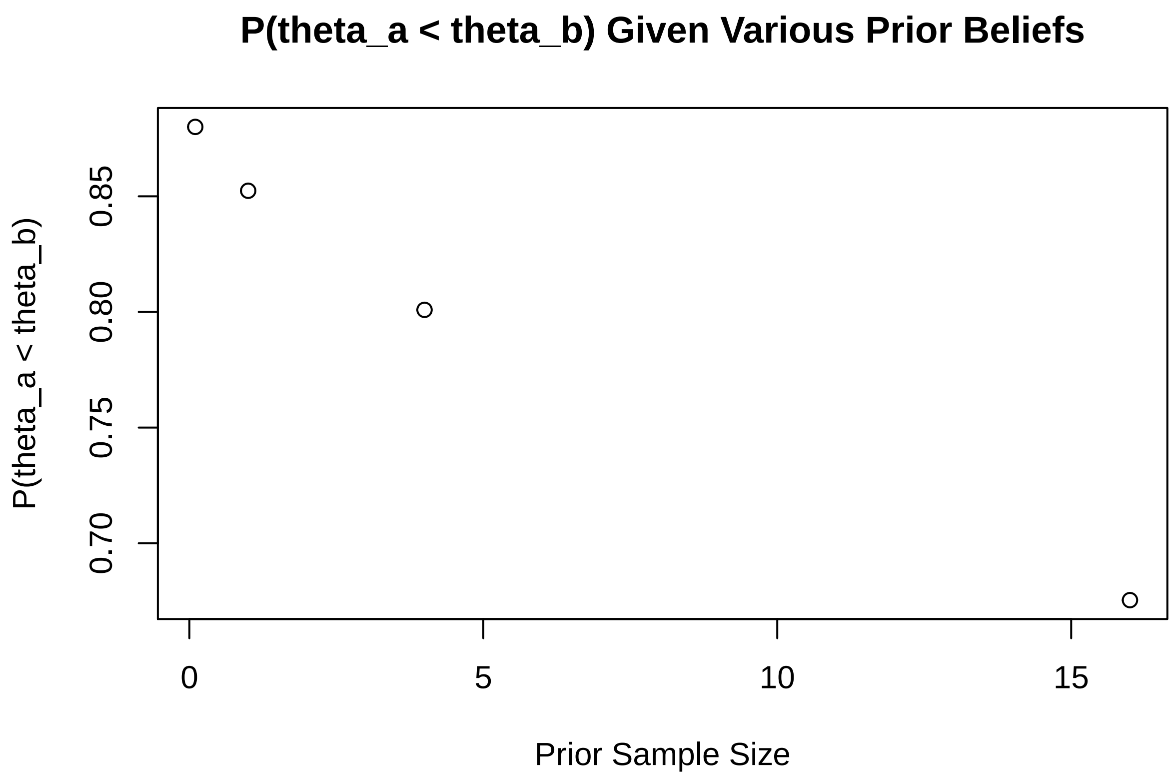
```r
kv <- c(.1, 1, 4, 16)
for (x in kv) {
mu_star_a <- x/(16 + x) * 75 + 16/(16 + x) * 75.2
y_star_a <- 1/(16+x) * (100*x + 15*32.49 + (16*x/(16+x)) * .2^2)
v_star_a <- 16 + x
var_a <- rinvgamma(10000, v_star_a/2, v_star_a * y_star_a / 2)
theta_a <- rnorm(10000, mu_star_a, sqrt(var_a/(k+n)))

mu_star_b <- x/(16 + x) * 75 + 16/(16 + x) * 77.5
y_star_b <- 1/(16+x) * (100*x + 15*65.61 + (16*x/(16+x)) * 2.5^2)
v_star_b <- 16 + x
var_b <- rinvgamma(10000, v_star_b/2, v_star_b * y_star_b / 2)
theta_b <- rnorm(10000, mu_star_b, sqrt(var_b/(k+n)))

print(mean(theta_b > theta_a))
}
```

```
## [1] 0.8698
## [1] 0.8528
## [1] 0.7957
## [1] 0.6694
```

```r
p_vec <- c(.88, .8524, .8009, .6754)
plot(x = kv, y = p_vec, main = "P(theta_a < theta_b) Given Various Prior Beliefs", xlab = "Prior Sample
```



Part C

```r
x <- .1
mu_star_a <- x/(16 + x) * 75 + 16/(16 + x) * 75.2
y_star_a <- 1/(16+x) * (100*x + 15*32.49 + (16*x/(16+x)) * .2^2)
v_star_a <- 16 + x
var_a <- rinvgamma(170000, v_star_a/2, v_star_a * y_star_a / 2)
theta_a <- rnorm(170000, mu_star_a, sqrt(var_a/16.1))
a_dist <- rnorm(170000, theta_a, sqrt(var_a))
mean(a_dist >= 65)
```

```
## [1] 0.9529412
```

```r
mu_star_b <- x/(16 + x) * 75 + 16/(16 + x) * 77.5
y_star_b <- 1/(16+x) * (100*x + 15*65.61 + (16*x/(16+x)) * 2.5^2)
v_star_b <- 16 + x
var_b <- rinvgamma(235000, v_star_b/2, v_star_b * y_star_b / 2)
theta_b <- rnorm(235000, mu_star_b, sqrt(y_star_b/16.1))
b_dist <- rnorm(235000, theta_b, sqrt(var_b))
mean(b_dist >= 65)
```

```
## [1] 0.9284085
```

```r
sd(a_dist >= 65)
```

```
## [1] 0.2117653
```

```r
sd(b_dist >= 65)
```

```
## [1] 0.2578108
```

```r
c(mean(a_dist >= 65) - 1.96 * sd(a_dist >= 65)/sqrt(170000), mean(a_dist >= 65) + 1.96 * sd(a_dist >= 65
```

```
## [1] 0.9519345 0.9539478
```

```r
c(mean(b_dist >= 65) - 1.96 * sd(b_dist >= 65)/sqrt(235000), mean(b_dist >= 65) + 1.96 * sd(b_dist >= 65
```

```
## [1] 0.9273661 0.9294509
```

For both distributions, the abs(probability estimate - bound of confidence interval) is < .001.

Question 3

Part A

```r
win_vec <- replicate(10000, 0)
rounds <- replicate(10000, 0)
win_rd_total = 0
for (i in 1:10000) {
  pennies = 0
  rounds_count = 0

  while (pennies < 49) {
  rounds_count = rounds_count + 1
  pennies = pennies + sample(x=3,size=1,prob=c(0.25,0.25,0.5))
  if (pennies >= 49){
    win_vec[i] = 1
    rounds[i] = rounds_count
    win_rd_total = win_rd_total + rounds_count
    break
  }
  pennies = pennies + sample(x=3,size=1,prob=c(0.5,0.25,0.25))
  if (pennies >= 49){
    rounds[i] = rounds_count
  }
  }
}
#My win probability
mean(win_vec)
```

```
## [1] 0.5638
```

```r
#Expected number of rounds in a game
mean(rounds)
```

```
## [1] 12.6641
```

```r
#Expected number of rounds | win
win_rd_total / sum(win_vec)
```

```
## [1] 12.86981
```

Part B

```r
win_vec_2 <- replicate(10000, 0)
rounds_2 <- replicate(10000, 0)
win_rd_total_2 = 0
for (i in 1:10000) {
  pennies = 0
  rounds_count = 0
  while (pennies < 49) {
  rounds_count = rounds_count + 1
  x <- sample(x=3,size=1,prob=c(0.25,0.25,0.5))
  pennies = pennies + x
  if (pennies >= 49){
    win_vec_2[i] = 1
    rounds_2[i] = rounds_count
    win_rd_total_2 = win_rd_total_2 + rounds_count
    break
  }
  pennies = pennies + sample(x=3,size=1,prob=c(0.5 - .15*x ,0.25 + .05*x, 0.25 + .1*x))
  if (pennies >= 49){
    rounds_2[i] = rounds_count
  }
  }
}
#My win probability
mean(win_vec_2)
```

```
## [1] 0.5052
```

```r
#Expected number of rounds in a game
mean(rounds_2)
```

```
## [1] 11.1714
```

```r
#Expected number of rounds | win
win_rd_total_2 / sum(win_vec_2)
```

```
## [1] 11.41924
```

Part C

```r
win_vec_3 <- replicate(10000, 0)
rounds_3 <- replicate(10000, 0)
win_rd_total_3 = 0
for (i in 1:10000) {
  pennies = 0
  rounds_count = 0
  while (pennies < 49) {
  rounds_count = rounds_count + 1
  x <- sample(x=3,size=1,prob=c(0.25,0.25,0.5))
  pennies = pennies + x
  if (pennies >= 49){
    win_vec_3[i] = 1
    rounds_3[i] = rounds_count
    win_rd_total_3 = win_rd_total_3 + rounds_count
    break
  }
  if (x == 1){
  pennies = pennies + 3
  }
  if (x == 2){
  pennies = pennies + 2

  }
  if (x == 3){
  pennies = pennies + 1
  }
  }
  }


#My win probability
mean(win_vec_3)
```

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

```r
#Expected number of rounds in a game
mean(rounds_3)
```

```
## [1] 13
```

```r
#Expected number of rounds | win
win_rd_total_3 / sum(win_vec_3)
```

```
## [1] 13
```