Homework 9

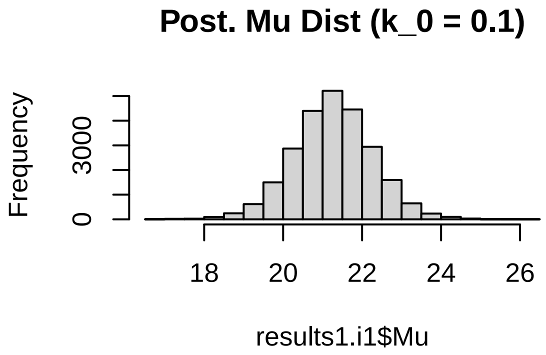
Zachary Lazerick

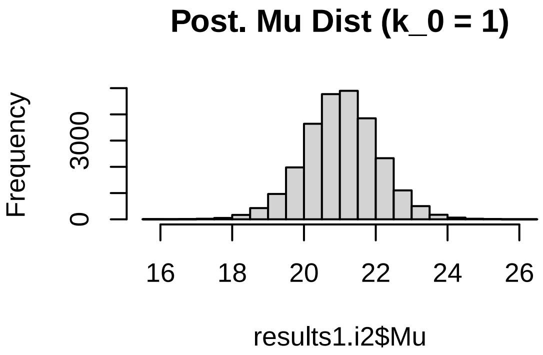
11 April 2023

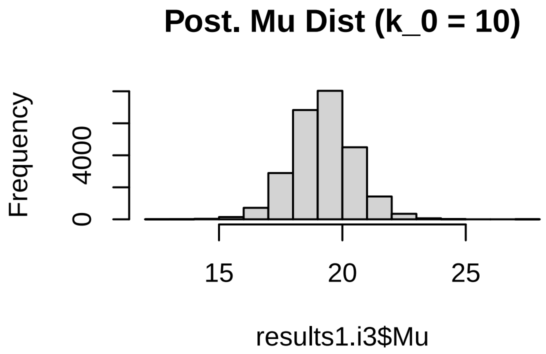
1) The Normal Distribution with Unknown Mean and Unknown Variance: Conjugate Priors – The data set on Canvas that accompanies this assignment contains 45 realizations of a normal distribution [it's the same one as last week – HW8_data.txt]. Again, your goal is to determine the mean and variance of the normal distribution that produced this data set, but this time using conjugate priors rather than the noninformative prior that we used previously. Ultimately, we are going to want to sample from the marginal posterior distributions for both μ and σ^2. Obtain 25,000 samples from $P(\mu|Y)$ and $P(\sigma^2|Y)$ using the full conditional for $\mu[P(\mu|\sigma^2, Y)]$ and marginal posterior for $\sigma^2[P(\sigma^2|Y)]$ sampler [not the Gibbs sampler]. Then:

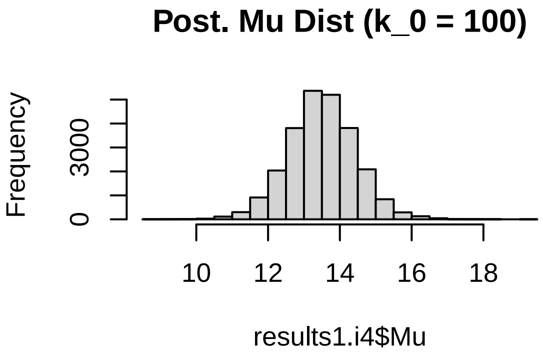
```r
target_df <- read.delim("HW8_data.txt", header = F)
attach(target_df)

ConjPrior <- function(iterations, Y, k_0, mu_0, v_0, sigma2_0) {
  ## Initialize Output Vectors
  output.mu <- rep(0, iterations); output.sigma2 <- rep(0, iterations)

  ## Initialize Posterior Values
  n = length(Y)
  k_n = k_0 + n; mu_n = (k_0/k_n) * mu_0 + (n/k_n) * mean(Y); v_n = v_0 + n
  scale = v_0 * sigma2_0 + (n - 1) * var(Y) + (k_0 * n)/k_n * (mean(Y) - mu_0)^2

  for (i in 1:iterations) {
    ## Generate Next Set of Variables
    chi <- rchisq(1, v_n); chi.inv <- 1/chi; next_sigma2 <- chi.inv * scale
    next_mu <- rnorm(1, mu_n, (next_sigma2 / k_n))

    ## Assign to Output Vector
    output.mu[i] <- next_mu; output.sigma2[i] <- next_sigma2
  }
  mylist <- list("Mu" = output.mu, "Sigma.Sq" = output.sigma2)
  return(mylist)
}
```

i) Describe how the value of k_0 impacts the marginal posterior distribution of μ. To do this, let $\mu_0 = 10$ and then choose various values for $k_0[= 0.1, 1, 10, 100]$. For now, fix the values of $v_0[= 2]$ and $\sigma_0^2[= 40]$. Your answer should read something like "as the value of k_0 increases, the posterior mean of μ increases/decreases/stays the same" or "as the value of k_0 increases, the posterior distribution for μ becomes more/less concentrated around the mean value."









```
mean(results1.i1$Mu); quantile(results1.i1$Mu, c(0.025, 0.975))
```

```
## [1] 21.26002
```

```
##     2.5%    97.5%
## 19.25699 23.24499
```

```
mean(results1.i2$Mu); quantile(results1.i2$Mu, c(0.025, 0.975))
```

```
## [1] 21.04341
```

```
##     2.5%    97.5%
## 18.94732 23.11362
```

```
mean(results1.i3$Mu); quantile(results1.i3$Mu, c(0.025, 0.975))
```

```
## [1] 19.22906
```

```
##     2.5%    97.5%
## 16.76260 21.76613
```

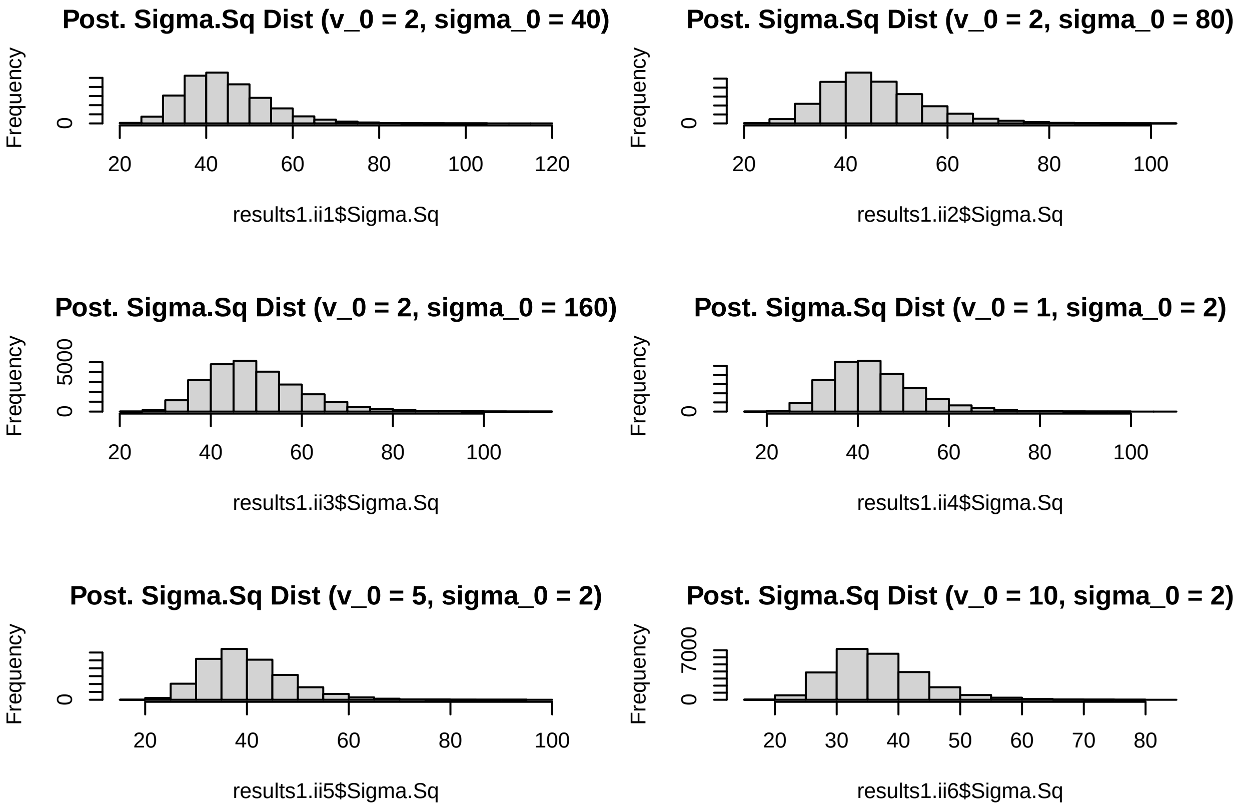
```
mean(results1.i4$Mu); quantile(results1.i4$Mu, c(0.025, 0.975))
```

```
## [1] 13.49869
```

```
##     2.5%    97.5%
## 11.63649 15.37913
```

As the value of k_0 increases, while mu_0, v_0, and σ_0^2 stay constant, the estimate of the posterior mean μ decreases. Thus, the 95% CI for μ also decreases. However, the width of the interval stays roughly the same.

ii) Describe how the values of v_0 and σ_0^2 impact the marginal posterior distribution of σ^2. Use (2,40), (2,80), (2,160), (1,2), (5,2), and (10,2) for (v_0, σ_0^2), and a fixed value for both μ_0[= 20] and k_0[= 1]. Your (written) description of the results should mirror what you wrote for (i), except now we are talking about σ^2.



```
mean(results1.ii1$Sigma.Sq); quantile(results1.ii1$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 44.26097
```

```
##     2.5%    97.5%
## 29.44979 66.42502
```

```r
mean(results1.ii2$Sigma.Sq); quantile(results1.ii2$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 45.9339
```

```
##    2.5%   97.5%
## 30.4834 69.2590
```

```r
mean(results1.ii3$Sigma.Sq); quantile(results1.ii3$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 49.47407
```

```
##     2.5%    97.5%
## 32.91296 74.32140
```

```r
mean(results1.ii4$Sigma.Sq); quantile(results1.ii4$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 43.45098
```

```
##     2.5%    97.5%
## 28.72881 65.67395
```

```r
mean(results1.ii5$Sigma.Sq); quantile(results1.ii5$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 39.90655
```

```
##     2.5%    97.5%
## 26.76693 58.82217
```

```r
mean(results1.ii6$Sigma.Sq); quantile(results1.ii6$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 36.42438
```

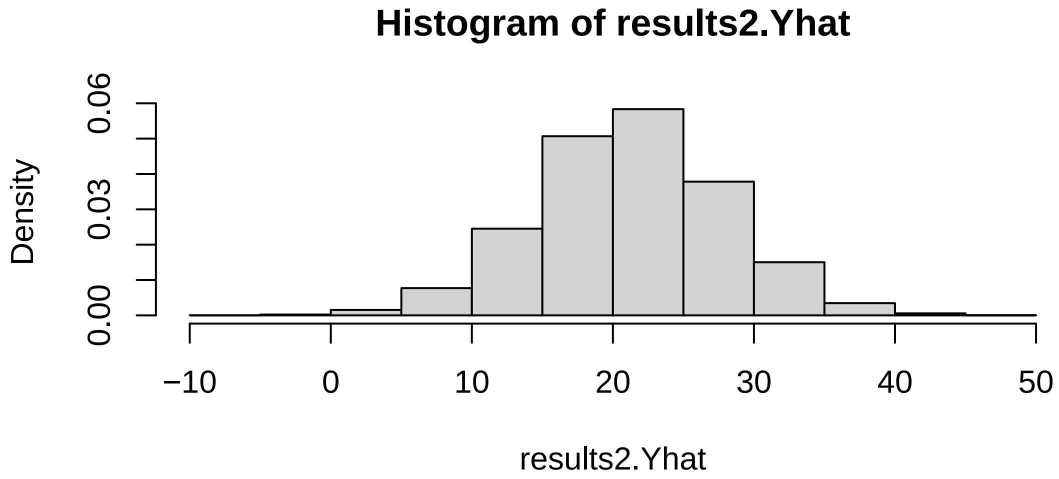
```
##     2.5%    97.5%
## 24.93878 53.10875
```

As the value of $sigma_0^2$ increases, while k_0, μ_0, and v_0 stay constant, the estimate for the posterior variance σ^2 increases. Thus, the 95% CI for σ^2 also increases. The width of this interval increases as well. As the value of v_0 increases, while k_0, μ_0, and σ_0^2 stay constant, the estimate for the posterior variance σ_2 decreases. Thus, the 95% CI for σ^2 also decreases. The width of this interval decreases as well.

2) Posterior Predictive Inference – Choose appropriate values for μ_0, k_0, v_0 and σ_0^2 in order to draw samples from the posterior predictive distribution of a future observation $P(\hat{Y}|Y)$. Do this by using the full conditional for μ and marginal posterior for σ^2 sampler from question #1. In other words, draw a value for σ^2, draw a value for $\mu|\sigma^2$, and then draw $\hat{Y}|\mu, \sigma^2 \sim N(\mu, \sigma^2)$ (then repeat!) Plot a histogram of your simulated values of $\hat{Y}$ and then determine the values C_1 and C_2 such that $PC_1 \leq \hat{Y} \leq C_2 = k$ for $k = 0.95$ and 0.99

For the prior distribution, I chose $k_0 = 9$ for 9 prior obs. of the mean, mu_0 equal to the mean of the dataset, $v_0 = 9$ for 9 prior obs. of the variance, and σ_0^2 equal to the variance of the dataset. Doing so yields:

```
set.seed(629)
results2 <- ConjPrior(25000, target_df$V1, k_0 = 9, mu_0 = mean(target_df$V1),
                      v_0 = 9, sigma2_0 = var(target_df$V1))
## Set New Vector for Y Pred.
results2.Yhat <- rep(0, length(results2$Mu)); set.seed(630)
for (i in 1:length(results2.Yhat)) {
  results2.Yhat[i] = rnorm(1, results2$Mu[i], sqrt(results2$Sigma.Sq[i]))
}
hist(results2.Yhat, freq = F)
```



```
mean(results2.Yhat)
```

```
## [1] 21.26796
```

```
quantile(results2.Yhat, c(0.025, 0.975)); quantile(results2.Yhat, c(0.005, 0.995))
```

```
##      2.5%     97.5%
##  7.978426 34.376441
```

```
##      0.5%     99.5%
##  3.738379 38.887790
```

3) Studying. The files "School1.txt", "School2.txt", and "School3.txt" contain data on the amount of time students from three high schools spent on studying or homework during an exam period. Analyze data from each of these schools separately, using the normal model with a conjugate prior distribution, in which $\{\mu 0 = 5, \sigma_0^2 = 4, k_0 = 1, v_0 = 2\}$ and compute the following:

```
## Read .txt Files
School1 <- read.delim("School1.txt", header = F)
School2 <- read.delim("School2.txt", header = F)
School3 <- read.delim("School3.txt", header = F)

## Simulate the Posterior Dist.
set.seed(317); results3.S1 <- ConjPrior(5000, School1$V1, 1, 5, 2, 4)
set.seed(318); results3.S2 <- ConjPrior(5000, School2$V1, 1, 5, 2, 4)
set.seed(319); results3.S3 <- ConjPrior(5000, School3$V1, 1, 5, 2, 4)

## Simulate Values from the Posterior
results3.S1.Yhat <- rep(0, length(results3.S1$Mu)); set.seed(329)
for (i in 1:length(results3.S1.Yhat)) {
  results3.S1.Yhat[i] = rnorm(1, results3.S1$Mu[i], sqrt(results3.S1$Sigma.Sq[i]))
}
results3.S2.Yhat <- rep(0, length(results3.S2$Mu)); set.seed(330)
for (i in 1:length(results3.S2.Yhat)) {
  results3.S2.Yhat[i] = rnorm(1, results3.S2$Mu[i], sqrt(results3.S2$Sigma.Sq[i]))
}
results3.S3.Yhat <- rep(0, length(results3.S3$Mu)); set.seed(331)
for (i in 1:length(results3.S3.Yhat)) {
  results3.S3.Yhat[i] = rnorm(1, results3.S3$Mu[i], sqrt(results3.S3$Sigma.Sq[i]))
}
```

a) posterior means and 95% confidence intervals for the mean and standard deviation from each school; Hint: You may want to create a function that takes the data for each school as its input. That way you don't have to copy and paste your code three times.

```
mean(results3.S1$Mu); quantile(results3.S1$Mu, c(0.025, 0.975)) ## School 1
```

```
## [1] 9.294906
```

```
##      2.5%     97.5%
##  8.039063 10.497905
```

```
mean(results3.S1$Sigma.Sq); quantile(results3.S1$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 15.49499
```

```
##      2.5%     97.5%
##  8.887052 26.944744
```

```
mean(results3.S2$Mu); quantile(results3.S2$Mu, c(0.025, 0.975)) ## School 2
```

```
## [1] 6.945665
##     2.5%    97.5%
## 5.204405 8.650701
```

```r
mean(results3.S2$Sigma.Sq); quantile(results3.S2$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 19.76251
```

```
##     2.5%    97.5%
## 10.99004 34.53539
```

```r
mean(results3.S3$Mu); quantile(results3.S3$Mu, c(0.025, 0.975)) ## School 3
```

```
## [1] 7.814963
```

```
##     2.5%    97.5%
## 6.317247 9.331346
```

```r
mean(results3.S3$Sigma.Sq); quantile(results3.S3$Sigma.Sq, c(0.025, 0.975))
```

```
## [1] 14.48433
```

```
##      2.5%     97.5%
##  7.927672 26.370183
```

b) the posterior probability that $\mu_i < \mu_j < \mu_k$ for all six permutations $\{i, j, k\}$ of $\{1, 2, 3\}$.

```r
mean((results3.S1$Mu < results3.S2$Mu) & (results3.S2$Mu < results3.S3$Mu))
```

```
## [1] 0.0034
```

```r
mean((results3.S1$Mu < results3.S3$Mu) & (results3.S3$Mu < results3.S2$Mu))
```

```
## [1] 0.001
```

```r
mean((results3.S2$Mu < results3.S1$Mu) & (results3.S1$Mu < results3.S3$Mu))
```

```
## [1] 0.053
```

```r
mean((results3.S2$Mu < results3.S3$Mu) & (results3.S3$Mu < results3.S1$Mu))
```

```
## [1] 0.7384
```

```r
mean((results3.S3$Mu < results3.S1$Mu) & (results3.S1$Mu < results3.S2$Mu))
```

```
## [1] 0.0136
```

```
mean((results3.S3$Mu < results3.S2$Mu) & (results3.S2$Mu < results3.S1$Mu))
```

```
## [1] 0.1906
```

c) the posterior probability that $\hat{Y}_i < \hat{Y}_j < \hat{Y}_k$ for all six permutations $\{i, j, k\}$ of $\{1, 2, 3\}$, where $\hat{Y}_i$s a sample from the posterior predictive distribution of school i.

```
mean((results3.S1.Yhat < results3.S2.Yhat) & (results3.S2.Yhat < results3.S3.Yhat))
```

```
## [1] 0.1036
```

```
mean((results3.S1.Yhat < results3.S3.Yhat) & (results3.S3.Yhat < results3.S2.Yhat))
```

```
## [1] 0.1068
```

```
mean((results3.S2.Yhat < results3.S1.Yhat) & (results3.S1.Yhat < results3.S3.Yhat))
```

```
## [1] 0.1836
```

```
mean((results3.S2.Yhat < results3.S3.Yhat) & (results3.S3.Yhat < results3.S1.Yhat))
```

```
## [1] 0.2666
```

```
mean((results3.S3.Yhat < results3.S1.Yhat) & (results3.S1.Yhat < results3.S2.Yhat))
```

```
## [1] 0.141
```

```
mean((results3.S3.Yhat < results3.S2.Yhat) & (results3.S2.Yhat < results3.S1.Yhat))
```

```
## [1] 0.1984
```

d) Compute the posterior probability that μ_1 is bigger than both μ_2 and μ_3, and the posterior probability that $\hat{Y}_1$ is bigger than both $\hat{Y}_2$ and $\hat{Y}_3$. Hint: This is simply a sum of two of the probabilities that you calculated in parts (b) and (c).

```
(mean((results3.S3$Mu < results3.S2$Mu) & (results3.S2$Mu < results3.S1$Mu))
  + mean((results3.S2$Mu < results3.S3$Mu) & (results3.S3$Mu < results3.S1$Mu)))
```

```
## [1] 0.929
```

```
(mean((results3.S2.Yhat < results3.S3.Yhat) & (results3.S3.Yhat < results3.S1.Yhat))
  + mean((results3.S3.Yhat < results3.S2.Yhat) & (results3.S2.Yhat < results3.S1.Yhat)))
```

```
## [1] 0.465
```

4) M&Ms – A young child opens up a two fun-sized bag of plain M&Ms and finds the following number of each colored candy:

Red	Orange	Yellow	Green	Blue	Brown
11	1	7	3	4	6

Let $\theta = (\theta_R, \theta_O, \theta_Y, \theta_G, \theta_{BL}, \theta_{BR})$ represent the true proportion of each color for plain M&Ms. For each prior distribution on θ given below, sample 1000 values of θ from the joint posterior distribution. Determine the posterior mean of θ_R and give a 95% Bayesian credible interval θ_R. Then, for part (d), use the code below as a template to plot the posterior density for each individual color of M&M.

```
EmInEm <- function(iterations, theta_1, theta_2, theta_3, theta_4, theta_5, theta_6) {
  theta1.out <- rep(0, iterations); theta2.out <- rep(0, iterations);
  theta3.out <- rep(0, iterations); theta4.out <- rep(0, iterations);
  theta5.out <- rep(0, iterations); theta6.out <- rep(0, iterations);

  for (i in 1:iterations) {
    ## Generate Xs
    x_1 <- rgamma(1, theta_1, rate = 1); x_2 <- rgamma(1, theta_2, rate = 1)
    x_3 <- rgamma(1, theta_3, rate = 1); x_4 <- rgamma(1, theta_4, rate = 1)
    x_5 <- rgamma(1, theta_5, rate = 1); x_6 <- rgamma(1, theta_6, rate = 1)
    x_sum = x_1 + x_2 + x_3 + x_4 + x_5 + x_6

    ## Generate Thetas
    t_1 = x_1/x_sum; t_2 = x_2/x_sum; t_3 = x_3/x_sum; t_4 = x_4/x_sum
    t_5 = x_5/x_sum; t_6 = x_6/x_sum

    ## Assign to Output Vector
    theta1.out[i] <- t_1; theta2.out[i] <- t_2; theta3.out[i] <- t_3
    theta4.out[i] <- t_4; theta5.out[i] <- t_5; theta6.out[i] <- t_6
  }
  mylist <- list("theta_red" = theta1.out, "theta_orange" = theta2.out,
                 "theta_yellow" = theta3.out, "theta_green" = theta4.out,
                 "theta_blue" = theta5.out, "theta_brown" = theta6.out)
  return(mylist)
}
```

a) $\theta \sim Dirichlet(0, 0, 0, 0, 0, 0)$ (Noninformative)

```
set.seed(449)
results4a <- EmInEm(1000, 11 + 0, 1 + 0, 7 + 0, 3 + 0, 4 + 0, 6 + 0)
mean(results4a$theta_red); quantile(results4a$theta_red, c(0.025, 0.975))
```

```
## [1] 0.3411612
```

```
##      2.5%     97.5%
## 0.1868621 0.5152472
```

b) $\theta \sim Dirichlet(1, 1, 1, 1, 1, 1)$ (Uniform)

```
set.seed(450)
results4b <- EmInEm(1000, 11 + 1, 1 + 1, 7 + 1, 3 + 1, 4 + 1, 6 + 1)
mean(results4b$theta_red); quantile(results4b$theta_red, c(0.025, 0.975))
```

```
## [1] 0.3154361
```

```
##      2.5%     97.5%
## 0.1871549 0.4560822
```

c) $\theta \sim Dirichlet(10, 10, 10, 10, 10, 10)$

```
set.seed(451)
results4c <- EmInEm(1000, 11 + 10, 1 + 10, 7 + 10, 3 + 10, 4 + 10, 6 + 10)
mean(results4c$theta_red); quantile(results4c$theta_red, c(0.025, 0.975))
```

```
## [1] 0.2300764
```

```
##      2.5%     97.5%
## 0.1473484 0.3193348
```

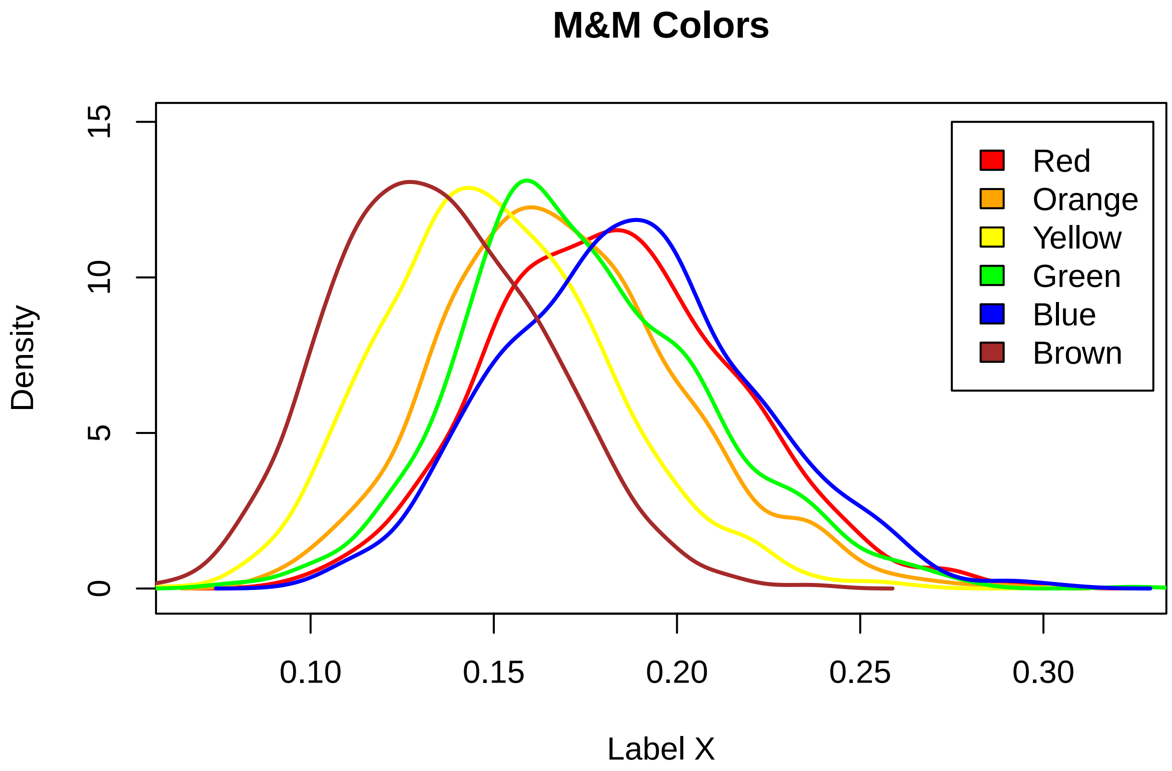
d) $\theta \sim Dirichlet(13, 21, 13, 20, 21, 12)$ (The actual distribution of colors)

```
set.seed(452)
results4d <- EmInEm(1000, 11 + 13, 1 + 21, 7 + 13, 3 + 20, 4 + 21, 6 + 12)
mean(results4d$theta_red); quantile(results4d$theta_red, c(0.025, 0.975))
```

```
## [1] 0.1828856
```

```
##      2.5%     97.5%
## 0.1201357 0.2503944
```

```
## Set Densities
red_density <- density(results4d$theta_red)
orange_density <- density(results4d$theta_orange)
yellow_density <- density(results4d$theta_yellow)
green_density <- density(results4d$theta_green)
blue_density <- density(results4d$theta_blue)
brown_density <- density(results4d$theta_brown)

## Construct Plot
plot(red_density$x, red_density$y, type = 'l', col = 'red', lwd = 2,
     ylim = c(-0.2, 15), xlab = "Label X", ylab = "Density", main = "M&M Colors")
lines(orange_density$x, orange_density$y, type = 'l', col = 'orange', lwd = 2)
lines(yellow_density$x, yellow_density$y, type = 'l', col = 'yellow', lwd = 2)
lines(green_density$x, green_density$y, type = 'l', col = 'green', lwd = 2)
lines(blue_density$x, blue_density$y, type = 'l', col = 'blue', lwd = 2)
lines(brown_density$x, brown_density$y, type = 'l', col = 'brown', lwd = 2)
legend(x = 0.275, y = 15,
       fill = c('red', 'orange', 'yellow', 'green', 'blue', 'brown'),
       legend = c("Red", "Orange", "Yellow", "Green", "Blue", "Brown"))
```


M&M Colors
Density
0
5
10
15
0.10
0.15
0.20
0.25
0.30
Label X
Red
Orange
Yellow
Green
Blue
Brown