

Homework4

Alexander Grunenwald

2024-10-28

```
library(ggplot2)
library(ggthemes)
theme_set(theme_economist())
```

Problem 1

We want to express the posterior distribution:

$$\begin{aligned} P(\theta \mid y^A, y^B, \gamma) &\propto P(y^A \mid \theta) P(y^B \mid \theta, \gamma) P(\theta) \quad \text{where } y^A, y^B \in \mathbb{R}^n, \quad e = 1_n \\ &\propto \theta^{e^T y^A} \exp(-n_A \theta) \theta \gamma^{e^T y^B} \exp(-n_B \theta \gamma) \theta^{\alpha_\theta - 1} \exp(-\beta_\theta \theta) \\ &\propto \theta^{e^T y^A + e^T y^B + \alpha_\theta - 1} \exp(-\theta(n_A + n_B \gamma + \beta_\theta)) \end{aligned}$$

We can express the posterior as:

$$P(\theta \mid y^A, y^B, \gamma) \sim \gamma(\hat{\alpha}_\theta, \hat{\beta}_\theta)$$

Where the parameters are:

$$\hat{\alpha}_\theta = e^T y^A + e^T y^B + \alpha_\theta, \quad \hat{\beta}_\theta = n_A + n_B \gamma + \beta_\theta$$

1b.)

We want to express the posterior distribution:

$$\begin{aligned} P(\gamma \mid y^A, y^B, \theta) &\propto P(y^B \mid \theta, \gamma) P(\gamma) \quad \text{where } y^A, y^B \in \mathbb{R}^n, \quad e = 1_n \\ &\propto \theta \gamma^{e^T y^B} \exp(-n_B \theta \gamma) \gamma^{\alpha_\gamma - 1} \exp(-\beta_\gamma \gamma) \\ &\propto \theta \gamma^{e^T y^B + \alpha_\gamma - 1} \exp(-\gamma(n_B \theta + \beta_\gamma)) \end{aligned}$$

We can express the posterior distribution as:

$$P(\gamma \mid y^A, y^B, \theta) \sim \gamma(\hat{\alpha}_\gamma, \hat{\beta}_\gamma)$$

The parameters are:

$$\hat{\alpha}_\gamma = e^T y^B + \alpha_\gamma \quad \hat{\beta}_\gamma = n_B \theta + \beta_\gamma$$

Part 1c.)

```

# read in files
yA <- scan("menchild30bach.txt"); yB <- scan("menchild30nobach.txt")
a_t <- 2; b_t<-1; a_g <- 8; b_g <- 8; S <- 5000

theta.post.params <- function(yA, yB, gam){
  list(alpha = a_t + sum(yA) + sum(yB), beta = b_t + length(yA) + length(yB)*gam)
}

gam.post.params <- function(yB, theta){
  list(alpha = a_g + sum(yB), beta = b_g + length(yB)*theta)
}

sample.matrix.1 <- data.frame(theta=rep(0, S), gam= rep(0,S))

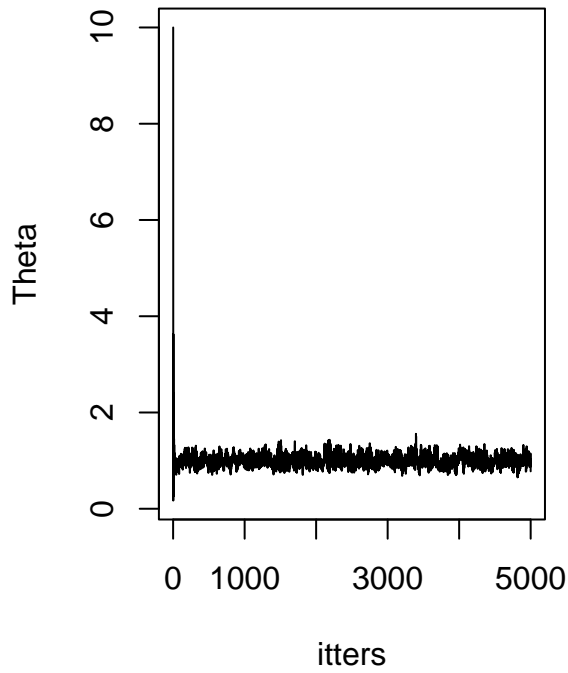
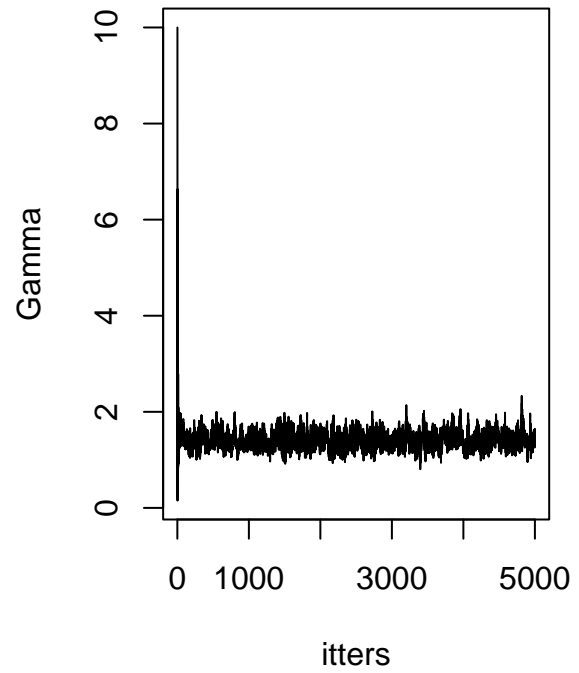
# intail start
gam<-10;theta<-10
sample.matrix.1[1, ] <- cbind(theta, gam)
for (itter in 2:S){
  # compute the parameters
  theta.post.hyp <- theta.post.params(yA, yB, gam)
  gamma.post.hyp <- gam.post.params(yB, theta)

  # Sample from the Posterior
  gam <- rgamma(1, shape= gamma.post.hyp$alpha, rate = gamma.post.hyp$beta)
  theta <- rgamma(1, shape= theta.post.hyp$alpha, rate = theta.post.hyp$beta)

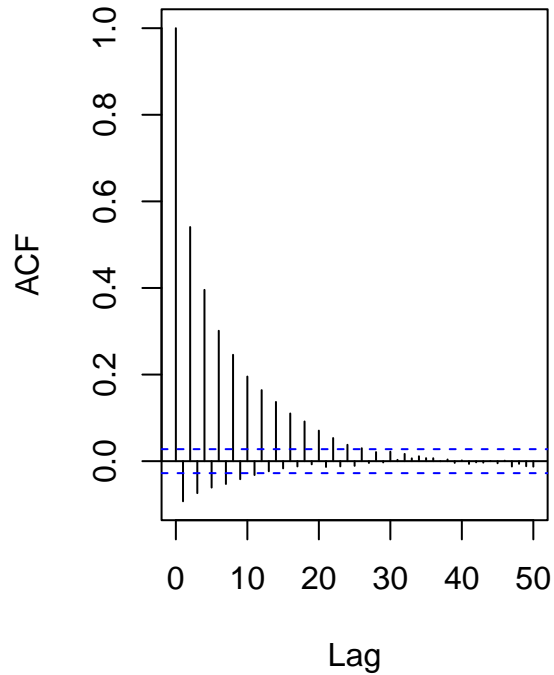
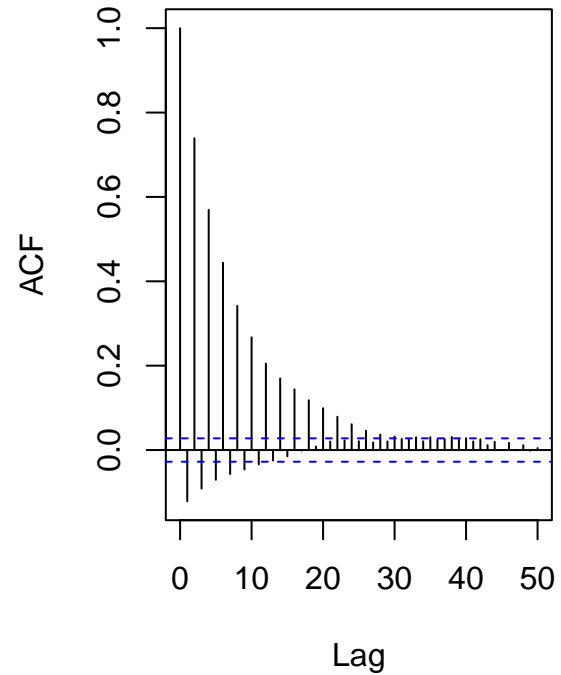
  # Store obtained values
  sample.matrix.1[itter, ] <- cbind(theta, gam)
}

mcmc.samples <- sample.matrix.1
par(mfrow=c(1, 2))
plot(mcmc.samples[, "theta"], type="l", main="Traceplot of Theta", ylab="Theta", xlab="itters")
plot(mcmc.samples[, "gam"], type="l", main="Traceplot of Gamma", ylab="Gamma", xlab="itters")

```

Traceplot of Theta**Traceplot of Gamma**

```
par(mfrow=c(1, 2))
acf(sample.matrix.1$theta, main="Autocorrelation of Theta", lag.max=50)
acf(sample.matrix.1$gam, main="Autocorrelation of Gamma", lag.max=50)
```

Autocorrelation of Theta**Autocorrelation of Gamma**

From these two plots, I see that the sampler converges quickly at the first couple of iterations, and that the parameters show a significant reduction in correlation after 30 lags. In order to estimate $\mathbb{E}(\theta_B - \theta_A \mid y^A, y^B)$ we will only use independent samples, I will apply a burn-in period of 100 iterations to remove the early, potentially non-converged samples. Additionally, I will thin the chain by keeping every 30th sample after the burn-in. From there, we know that $\theta_A = \theta$ and $\theta_B = \theta\gamma$ and putting it all

together is just

$$\mathbb{E}(\theta\gamma - \theta \mid y^A, y^B)$$

which we can easily calculate using the mean function in R and just subtracting θ from the scaled $\theta\gamma$ column.

```
post_burn_in <- sample.matrix.1[-(1:100), ]
thinned_samples.1 <- post_burn_in[seq(1, nrow(post_burn_in), by = 30), ]
mean_diff.1 <- mean(thinned_samples.1$theta*thinned_samples.1$gam - thinned_samples.1$theta)
```

1d.)

```
a_t <- 2; b_t<-1; a_g <- 128; b_g <- 128; S <- 5000

theta.post.params <- function(yA, yB, gam){
  list(alpha = a_t + sum(yA) + sum(yB), beta = b_t + length(yA) + length(yB)*gam)
}

gam.post.params <- function(yB, theta){
  list(alpha = a_g + sum(yB), beta = b_g + length(yB)*theta)
}

sample.matrix.2 <- data.frame(theta=rep(0, S), gam= rep(0,S))

# intail start
gam<-10;theta<-10
sample.matrix.2[1, ] <- cbind(theta, gam)
for (itter in 2:S){
  # compute the parameters
  theta.post.hyp <- theta.post.params(yA, yB, gam)
  gamma.post.hyp <- gam.post.params(yB, theta)

  # Sample from the Posterior
  gam <- rgamma(1, shape= gamma.post.hyp$alpha, rate = gamma.post.hyp$beta)
  theta <- rgamma(1, shape= theta.post.hyp$alpha, rate = theta.post.hyp$beta)

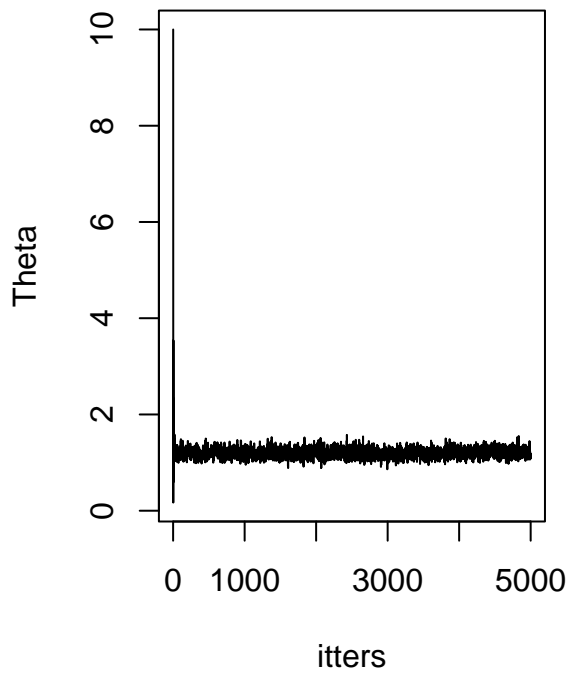
  # Store obtained values
  sample.matrix.2[itter, ] <- cbind(theta, gam)
}
post_burn_in <- sample.matrix.2[-(1:100), ]

thinned_samples.2 <- post_burn_in[seq(1, nrow(post_burn_in), by = 10), ]

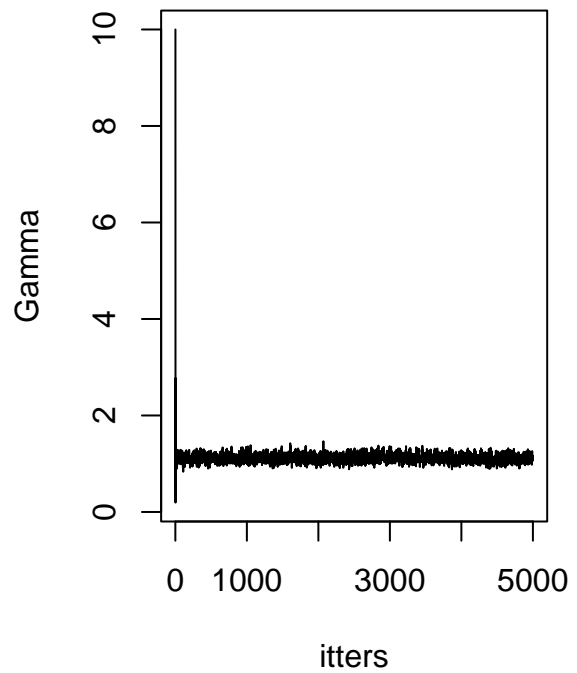
mean_diff.2 <- mean(thinned_samples.2$theta*thinned_samples.2$gam - thinned_samples.2$theta)

mcmc.samples <- sample.matrix.2
par(mfrow=c(1, 2))
plot(mcmc.samples[, "theta"], type="l", main="Traceplot of Theta", ylab="Theta", xlab="itters")
plot(mcmc.samples[, "gam"], type="l", main="Traceplot of Gamma", ylab="Gamma", xlab="itters")
```

Traceplot of Theta

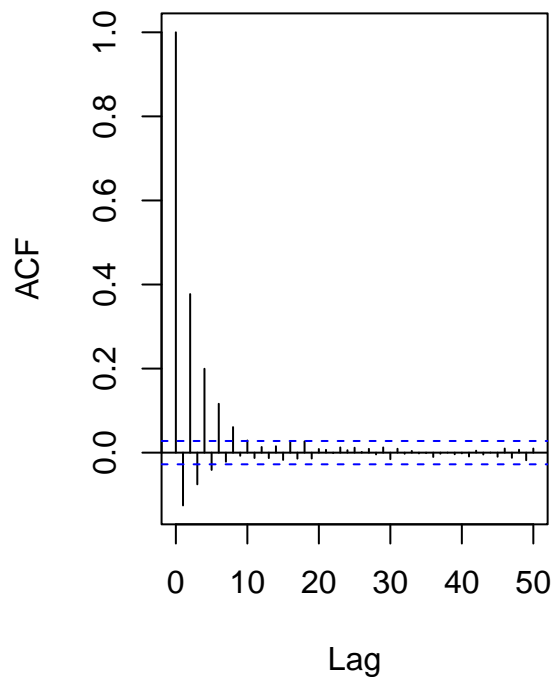


Traceplot of Gamma

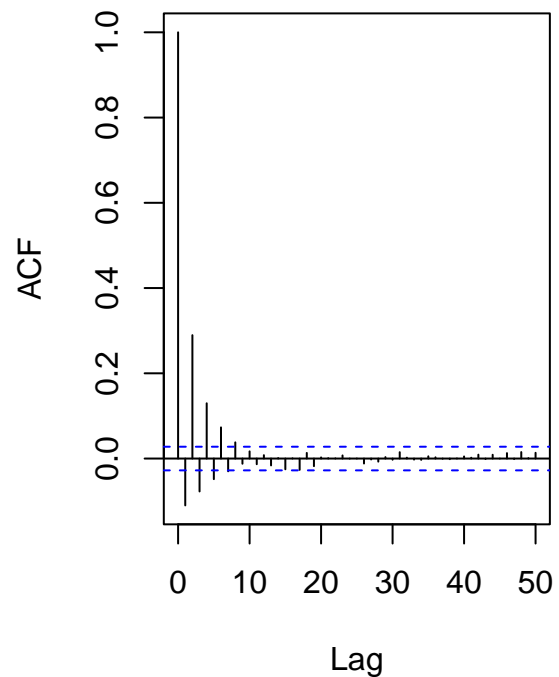


```
par(mfrow=c(1, 2))
acf(sample.matrix.2$theta, main="Autocorrelation of Theta", lag.max=50)
acf(sample.matrix.2$gam, main="Autocorrelation of Gamma", lag.max=50)
```

Autocorrelation of Theta



Autocorrelation of Gamma



I would say it converges to the sample mean at the same rate as the one from part c; however, the autocorrelation is much lower and shows independence at around 10 lags.

```
cat("Mean Difference from part c: ", mean_diff.1, "\nMean Difference from part d: ", mean_diff.2)
```

```
## Mean Difference from part c: 0.3894848
```

```
## Mean Difference from part d: 0.1351838
```

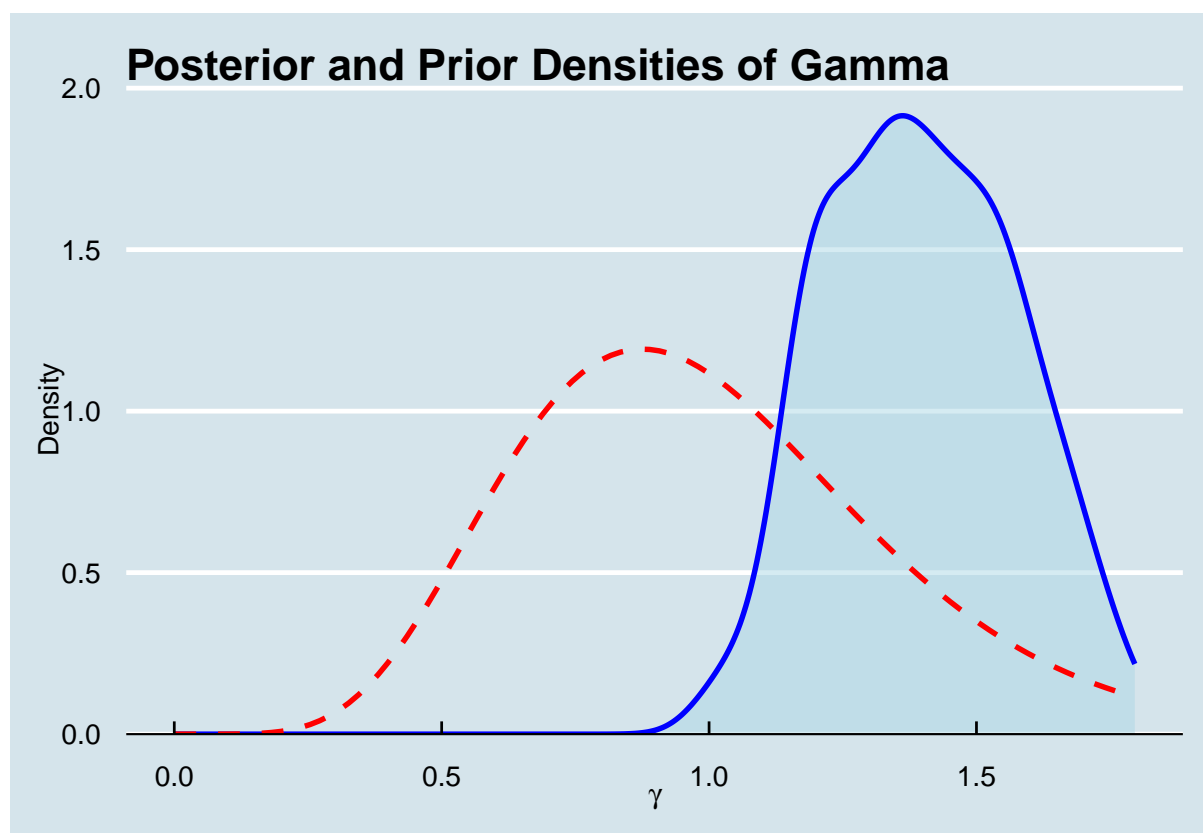
The results of changing the parameters in the gamma prior, made it so that there was less variance within the mcmc samples as well as reducing the overall autocorrelation within the samples.

```
a_g <- 8
b_g <- 8

gamma_vals <- seq(0, max(thinned_samples.1$gam), length.out = 100)

prior_density <- dgamma(gamma_vals, shape = a_g, rate = b_g)

ggplot() +
  geom_density(data = thinned_samples.1, aes(x = gam),
    fill = "lightblue", alpha = 0.5,
    color = "blue", linewidth = 1, linetype = "solid") +
  geom_line(aes(x = gamma_vals, y = prior_density),
    color = "red", linewidth = 1, linetype = "dashed") +
  labs(title = "Posterior and Prior Densities of Gamma",
    x = expression(gamma),
    y = "Density") +
  scale_y_continuous(expand = expansion(mult = c(0, 0.05)))
```



The prior distribution shows our prior beliefs of the parameter ratio of $\frac{\theta_B}{\theta_A}$ and the posterior is our updated beliefs after observing the data. According to the distributions, we tend to see that men in Group B tend to have 1.4 times more children than those in Group A.