Module 5: Solutions to exercises

Data

First recreate the same artificial dataset `x` as in the supplementary material (note the `set.seed()` command):

```
mu_true <- 250
tau_true <- 1/5^2
n <- 30
set.seed(42)
x <- rnorm(n, mean = mu_true, sd = sqrt(1/tau_true))
x_bar <- mean(x)
```

Exercise 1

Now, assume the researcher a priori (before seeing any data/people) is sure that she/he is in a land of tiny people, and chooses the following parameters for the priors for the mean and precision (still product of normal and gamma distribution):

```
tau_prior <- 1/10^2
mu_prior <- 50
alpha_prior <- 10
beta_prior <- .002
```

Rerun the Gibbs sampler from the supplementary material (with the same updating scheme: first μ then τ) with these prior parameters and comment on the marginal distribution of μ (and τ if you like).

From the plots it appears μ is far above the prior and close to the data mean.

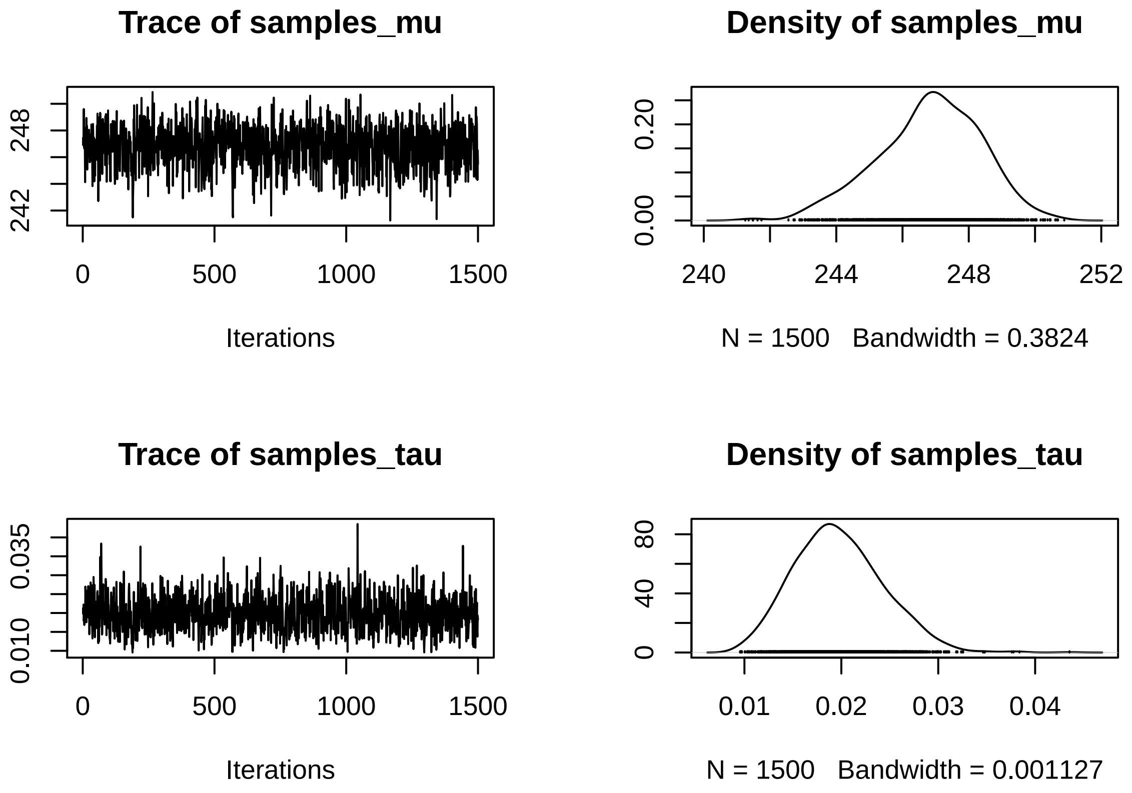
```
rmu <- function(tau){
  cond_mean <- (n*tau*x_bar + tau_prior*mu_prior)/(n*tau + tau_prior)
  cond_tau <- n*tau + tau_prior
  return(rnorm(1, mean = cond_mean, sd = sqrt(1/cond_tau)))
}
rtau <- function(mu){
  cond_alpha <- n/2 + alpha_prior
  cond_beta <- 1/(sum((x-mu)^2)/2 + 1/beta_prior)
  return(rgamma(1, shape = cond_alpha, scale = cond_beta))
}
```

```
n_samples <- 2000 # Number of samples
samples_tau <- samples_mu <- rep(0, n_samples) # Vectors to write results in
samples_tau[1] <- alpha_prior * beta_prior # Start in prior mean
samples_mu[1] <- mu_prior # Start in prior mean
for(i in 2:n_samples){
  samples_mu[i] <- rmu(samples_tau[i-1])
  samples_tau[i] <- rtau(samples_mu[i])
}
```

```r
library(coda)
samples <- cbind(samples_mu, samples_tau)
burn_in <- 1:500
chain <- samples[-burn_in,]
chain <- mcmc(chain)
summary(chain)
```

```
##
## Iterations = 1:1500
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 1500
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                   Mean       SD  Naive SE Time-series SE
## samples_mu  246.80644 1.557568 0.0402162       0.047516
## samples_tau   0.01972 0.004591 0.0001185       0.000144
##
## 2. Quantiles for each variable:
##
##                   2.5%       25%       50%       75%     97.5%
## samples_mu  243.48268 245.83297 246.92483 247.93844 249.53111
## samples_tau   0.01172   0.01642   0.01935   0.02267   0.02909
```

```r
plot(chain)
```


Trace of samples_mu

Density of samples_mu

Trace of samples_tau

Density of samples_tau

Exercise 2

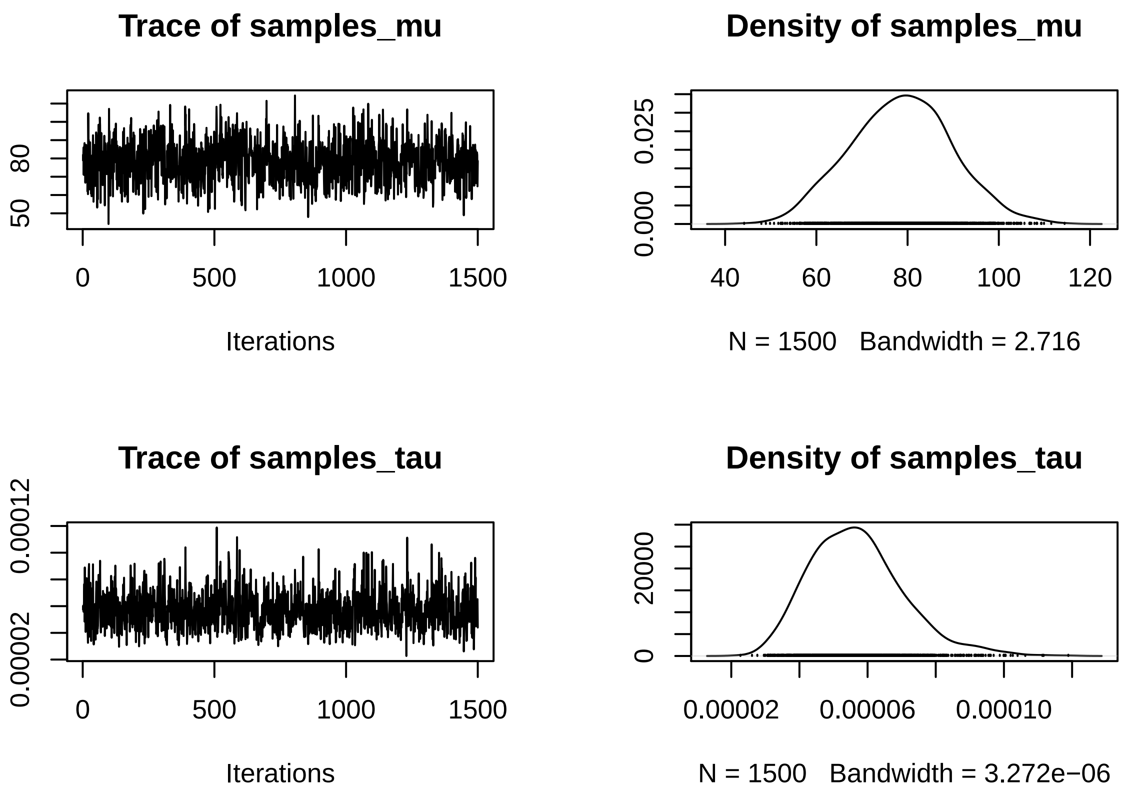
Now rerun the Gibbs sampler from the supplementary material with the opposite updating scheme – first τ then μ – and comment on the marginal distribution of μ (and τ if you like).

```
n_samples <- 2000 # Number of samples
samples_tau <- samples_mu <- rep(0, n_samples) # Vectors to write results in
samples_tau[1] <- alpha_prior * beta_prior # Start in prior mean
samples_mu[1] <- mu_prior # Start in prior mean
for(i in 2:n_samples){
  samples_tau[i] <- rtau(samples_mu[i-1])
  samples_mu[i] <- rmu(samples_tau[i])
}
```

```
samples <- cbind(samples_mu, samples_tau)
burn_in <- 1:500
chain <- samples[-burn_in,]
chain <- mcmc(chain)
summary(chain)
```

```
##
## Iterations = 1:1500
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 1500
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                  Mean        SD  Naive SE Time-series SE
## samples_mu  7.861e+01 1.106e+01 2.857e-01      3.728e-01
## samples_tau 5.686e-05 1.373e-05 3.545e-07      4.316e-07
##
## 2. Quantiles for each variable:
##
##                  2.5%       25%       50%       75%     97.5%
## samples_mu  5.770e+01 7.110e+01 7.879e+01 8.620e+01 9.981e+01
## samples_tau 3.409e-05 4.693e-05 5.574e-05 6.479e-05 8.941e-05
```

```
plot(chain)
```



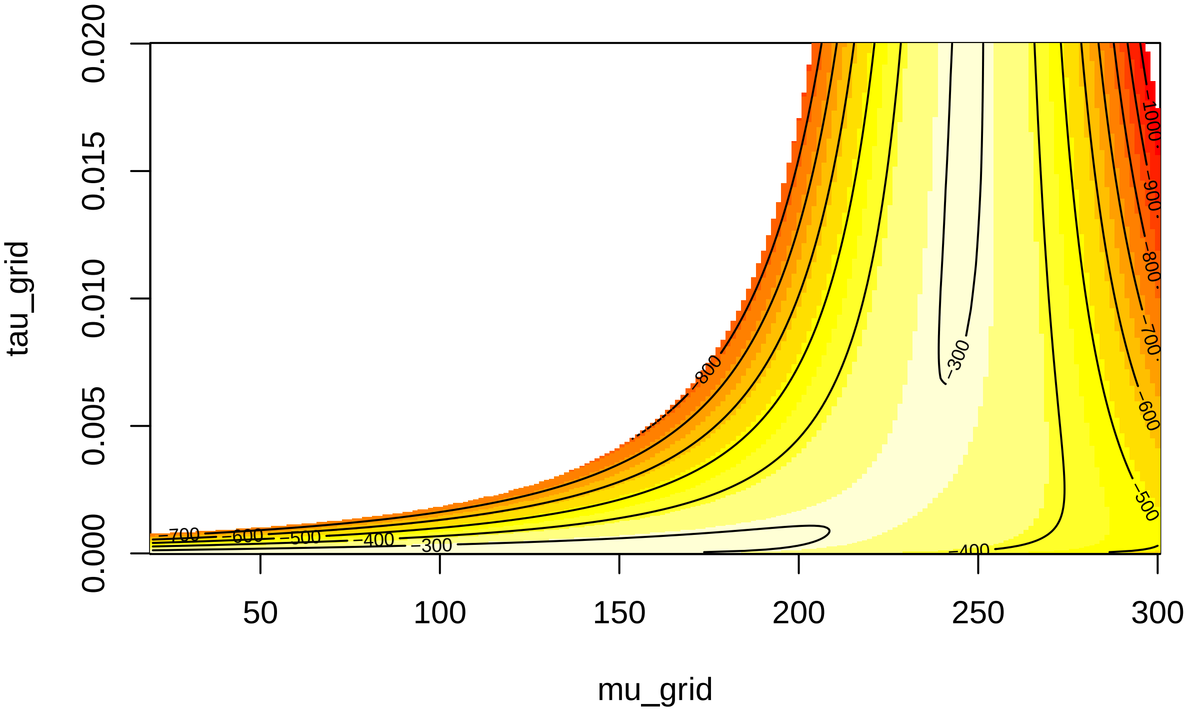
Exercise 3

Try to explain the different results you obtained in exercises 1 and 2 by looking at the posterior we are trying to sample from. **Don't spend too much time on this** – it is difficult, and you will probably have to consider the logarithm of the posterior to be able to identify the problem: Instead consult the solution if you find it too difficult.

Due to the conflicting prior and likelihood the posterior is bimodal: From the contour plot it appears there is one local maximum around $\mu = 80$ and $\tau = 0.00005$ and another local maximum around $\mu = 247$ and $\tau = 0.02$. The Gibbs sampler is not able to move between the two modes in this case and gets stuck in one. Which one depends on the updating scheme (and the initial values).

```
mu_grid <- seq(20, 300, length.out = 200)
tau_grid <- seq(0,.02,length.out=401)
# tau_grid <- exp(seq(log(1e-7), log(.08), length.out = 101))
# tau_grid <- (seq(sqrt(1e-7), sqrt(.08), length.out = 101))^2
prior <- function(mu, tau){
  x <- dnorm(mu, mean = mu_prior, sd = 1/sqrt(tau_prior))
  y <- dgamma(tau, shape = alpha_prior, scale = beta_prior)
  return(x*y)
}
post_grid <- matrix(0, nrow = length(mu_grid), ncol = length(tau_grid))
for(i in seq_along(mu_grid)){
  for(j in seq_along(tau_grid)){
    post_grid[i,j] <- log(prior(mu_grid[i], tau_grid[j])) +
      log(prod(dnorm(x, mean = mu_grid[i], sd = 1/sqrt(tau_grid[j]))))
  }
```

```r
}
image(mu_grid, tau_grid, post_grid)
contour(mu_grid, tau_grid, post_grid, add = TRUE)
```



Exercise 4

Now generate additional 70 data points (people) from the population (still N(250, 0.04)) and rerun the analysis from exercise 1 with this new dataset of 100 people (**remember** to update the global variables `n` and `x_bar`). Comment on the results. **Now the posterior is uni-modal and the Gibbs sampler correctly samples from the posterior distribution. We now have so much data that the prior doesn't matter much, and the distribution is uni-modal around the data mean.**

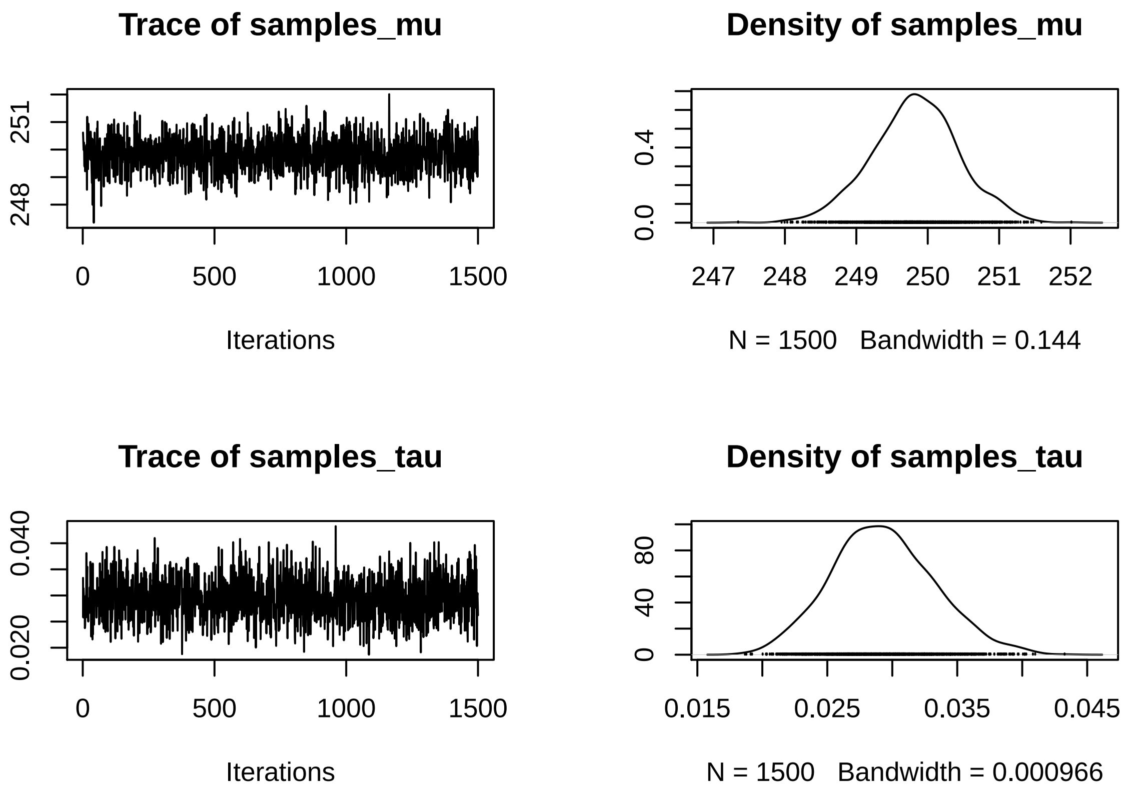
```r
y <- rnorm(70, mean = mu_true, sd = sqrt(1/tau_true))
x <- c(x,y) # Combine old 30 values, x, with new 70 values, y.
n <- length(x)
x_bar <- mean(x)
```

```r
n_samples <- 2000 # Number of samples
samples_tau <- samples_mu <- rep(0, n_samples) # Vectors to write results in
samples_tau[1] <- alpha_prior * beta_prior # Start in prior mean
samples_mu[1] <- mu_prior # Start in prior mean
for(i in 2:n_samples){
  samples_mu[i] <- rmu(samples_tau[i-1])
  samples_tau[i] <- rtau(samples_mu[i])
}
```

```r
samples <- cbind(samples_mu, samples_tau)
burn_in <- 1:500
chain <- samples[-burn_in,]
chain <- mcmc(chain)
summary(chain)
```

```
##
## Iterations = 1:1500
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 1500
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##                  Mean       SD  Naive SE Time-series SE
## samples_mu  249.82458 0.605219 0.0156267      0.0156267
## samples_tau   0.02929 0.003935 0.0001016      0.0001016
##
## 2. Quantiles for each variable:
##
##                  2.5%      25%       50%       75%     97.5%
## samples_mu  248.59724 249.4339 249.83587 250.21988 251.00814
## samples_tau   0.02184   0.0266   0.02908   0.03189   0.03734
```

```r
plot(chain)
```



Exercise 5

Based on the posterior simulations in exercise 4 estimate the probability of $\mu < 249$.

```r
mean(chain[,1]<249)
```

```
## [1] 0.09133333
```