

Introduction to Bayesian Statistics with R

4: Exercise solutions

Jack Kuipers

28 November 2022

First we load the tidyverse and set a seed.

```
library(tidyverse); set.seed(42)
```

Exercise 4.1 - Credible intervals

Update the `basicMCMC` code from Exercise 3.2 to sample μ from its posterior distribution, depending on the data x , and the prior choices t_s and ν .

For data we will use Gosset's data on barley yields from his 1908 paper which introduced the Student- t distribution. The yields (in US cwt per acre) for 11 plots of land for normal seed and seed which has been dried in a kiln are in `seed_data.csv`.

- Read in the data, and extract the relevant quantity as x
- Choose prior parameters t_s and ν
- Sample from the posterior distribution of μ
- Visualise this distribution
- Obtain estimates for the 95% credible interval of μ

First let's rewrite our `basicMCMC` function, renaming the main variable to μ for clarity:

```
# simple MCMC function in terms of the variable mu
# n_its is the number of iterations
# start_mu the initial position
# rw_sd is the sd of the Gaussian random walk
basicMCMCmu <- function(n_its = 1e3, start_mu = 0, rw_sd = 1, ...) {
  mus <- rep(NA, n_its) # to store all the sampled values
  mu <- start_mu # starting point
  mus[1] <- mu # first value
  p_mu <- target_density(mu, ...) # probability density at current value of mu
  for (ii in 2:n_its) { # MCMC iterations
    mu_prop <- mu + rnorm(1, mean = 0, sd = rw_sd) # Gaussian random walk to propose next mu
    p_mu_prop <- target_density(mu_prop, ...) # probability density at proposed mu
    if (runif(1) < p_mu_prop/p_mu) { # MH acceptance probability
      mu <- mu_prop # accept move
      p_mu <- p_mu_prop # update density
    }
    mus[ii] <- mu # store current position, even when move rejected
  }
  return(mus)
}
```

For the target density, we want the product of the likelihood and the prior

```
g_like <- function(mu, sigma, x2, x1, x0) {  
  exp(-(x2 - 2*mu*x1 + x0*mu^2)/(2*sigma^2))/(sigma^x0)  
}  
prior_t <- function(mu, t_s, nu) { # prior  
  dt(mu/t_s, nu)/t_s # scaled student-t density  
}  
target_density <- function(mu, x2, x1, x0, t_s, nu) { # product  
  g_like(mu, sigma=1, x2, x1, x0)*prior_t(mu, t_s, nu)  
}
```

Next we can read in the data

```
seed_df <- read.csv("./data/seed_data.csv")  
head(seed_df)
```

```
##   Plot Regular_seed Kiln_dried_seed  
## 1     1         19.03          20.09  
## 2     2         19.35          19.15  
## 3     3         19.10          20.11  
## 4     4         24.96          24.63  
## 5     5         21.08          21.80  
## 6     6         19.61          19.25
```

The quantity we care about is the difference in yield, especially given the wide variability in yields over the different plots. Let's extract this and compute the statistics we need for the likelihood

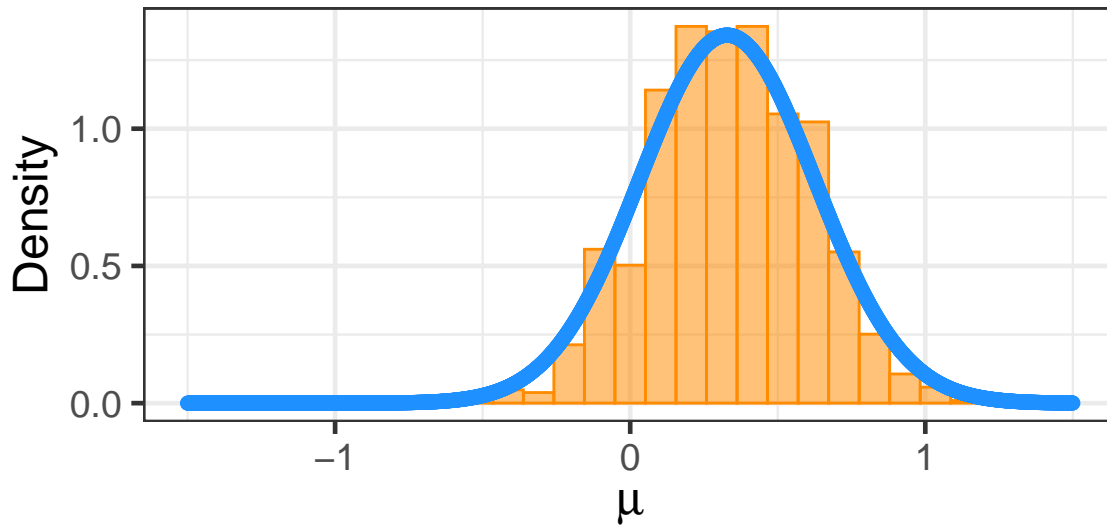
```
x <- seed_df$Kiln_dried_seed - seed_df$Regular_seed  
x2 <- sum(x^2)  
x1 <- sum(x)  
x0 <- length(x) # number of observations
```

We also need to choose our prior distribution. Knowing very little about yields, a low degree of freedom gives fatter tails so let's choose $\nu = 5$. On yields of around 20 a difference of ± 2 (or 10%) would seem quite plausible so let's rescale our t -distribution by the factor $t_s = 2$.

Now we can quickly run a short chain with the default length of 1000 iterations:

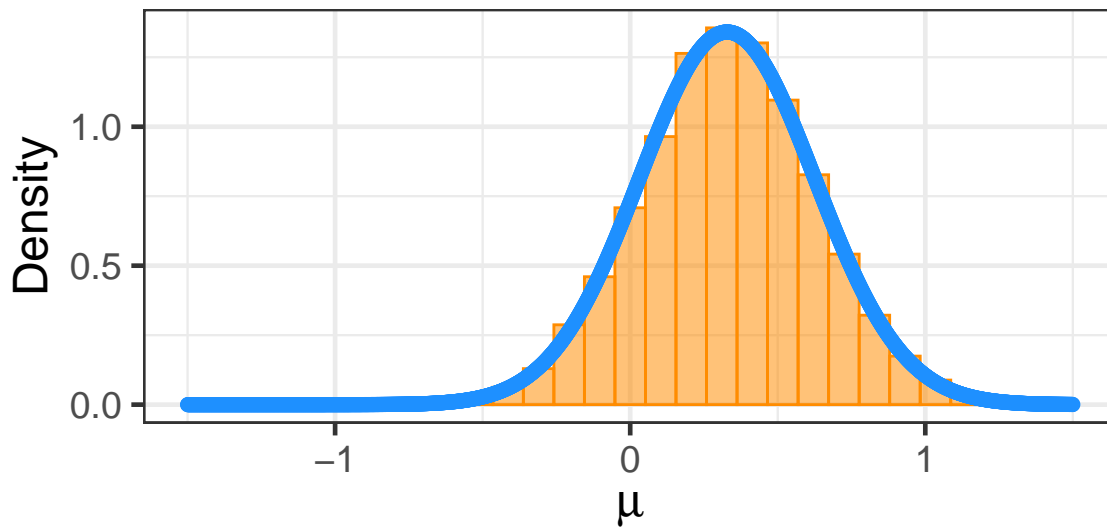
```
short_chain <- basicMCMCmu(x2=x2, x1=x1, x0=x0, t_s=2, nu=5)
```

On the histogram we overlay the target density and see ok agreement



To get better estimates we can try a longer chain

```
long_chain <- basicMCMCmu(n_its=1e5, x2=x2, x1=x1, x0=x0, t_s=2, nu=5)
```



From the samples in the chain, we can easily extract estimates for the credible intervals

```
round(quantile(long_chain, c(0.025, 0.975)), 2)
```

```
## 2.5% 97.5%
## -0.25 0.92
```