Module 11: Solution (inspiration for sampler code)

Initial definition of samplers

Dirichlet sampler:

```
rdirichlet <- function(alpha){
  k <- length(alpha)
  # k gamma variables with k different shape parameters:
  gam <- rgamma(k, shape = alpha, scale = 1)
  return(gam/sum(gam))
}
```
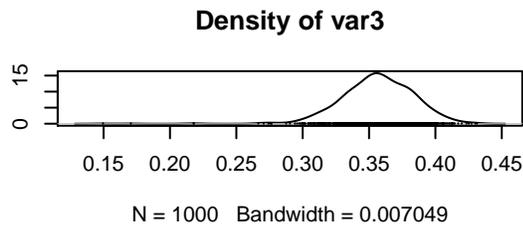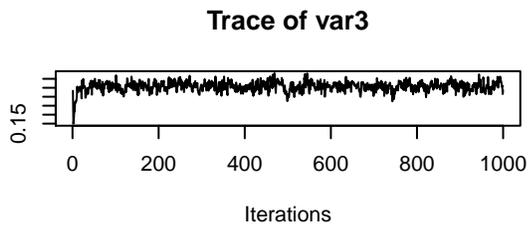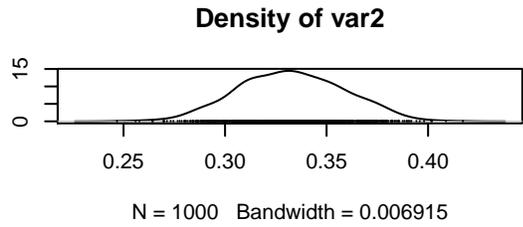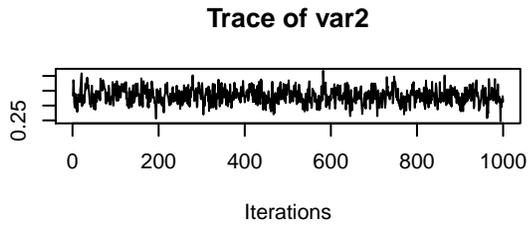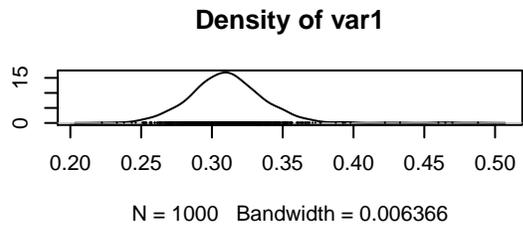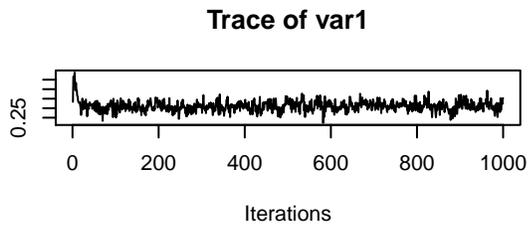
Samplers for full conditionals (in all these a bunch of global variables are assumed):

```
# Sampler for k-dimensional mu vector (mean of each component)
rmu <- function(z){
  mu <- rep(0, k)
  for(i in 1:k){
    x <- y[z==i] # y-values from i'the component
    n <- length(x)
    x_bar <- mean(x)
    # Hack to avoid NA if x is empty (no data with label i):
    if(n<1){x_bar <- 0}
    cond_mean <- (n*tau*x_bar + tau_prior*mu_prior)/(n*tau + tau_prior)
    cond_tau <- n*tau + tau_prior
    mu[i] <- rnorm(1, mean = cond_mean, sd = sqrt(1/cond_tau))
  }
  return(mu)
}
# Sampler for k-dimensional lambda vector (mixture weight/prob for each component)
rlambda <- function(z){
  nz <- rep(0, k)
  for(i in 1:k){
    nz[i] <- sum(z==i)
  }
  return(rdirichlet(alpha = nz + alpha_prior))
}
# Sampler for n-dimensional z vector (component label for each data point)
rz <- function(lambda, mu){
  z <- rep(0, n_data)
  for(i in 1:n_data){
    prob <- lambda * dnorm(y[i], mean = mu, sd = 1/sqrt(tau))
    z[i] <- sample(1:k, size = 1, prob = prob)
  }
  return(z)
}
```
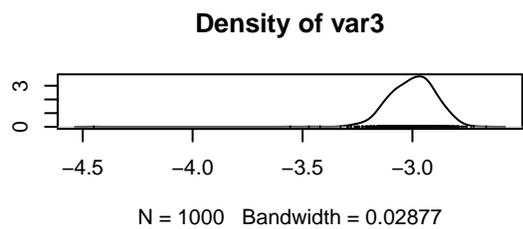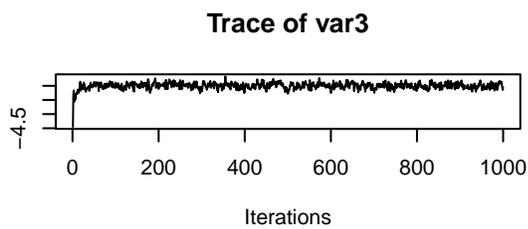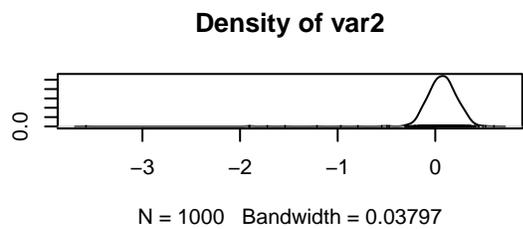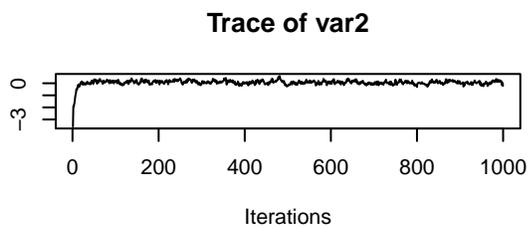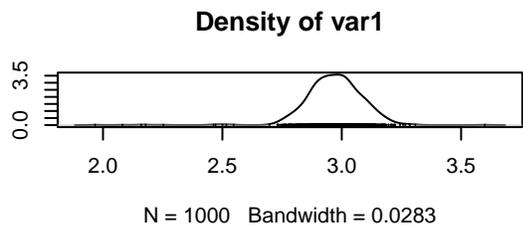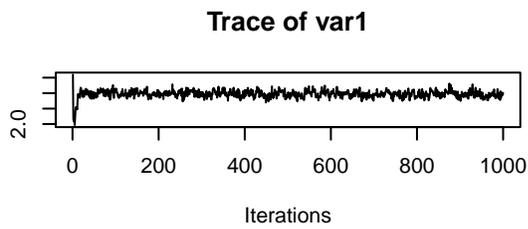
Exercise 1: Simulated mixture data

```r
# Constants and data:
k <- 3
tau_prior <- 1/3^2
mu_prior <- 0
alpha_prior <- rep(1,k)
tau <- 1
data_url <- "https://asta.math.aau.dk/course/bayes/2019/?file=simmix.csv"
mixdata <- read.csv(data_url, header = TRUE)
y <- mixdata$x
n_data <- length(y)
# Setup samplers
n_samples <- 1000
samples_mu <- samples_lambda <- matrix(0, nrow = n_samples, ncol = k)
samples_z <- matrix(0, nrow = n_samples, ncol = n_data)
samples_mu[1,] <- runif(k, -5, 5)
samples_lambda[1,] <- rep(1/k, k)
for(i in 2:n_samples){
  samples_z[i,] <- rz(samples_lambda[i-1,], samples_mu[i-1,])
  samples_lambda[i,] <- rlambda(samples_z[i,])
  samples_mu[i,] <- rmu(samples_z[i,])
}
```

```r
library(coda)
samples_lambda <- mcmc(samples_lambda)
samples_mu <- mcmc(samples_mu)
samples_z <- mcmc(samples_z)
plot(samples_lambda)
```
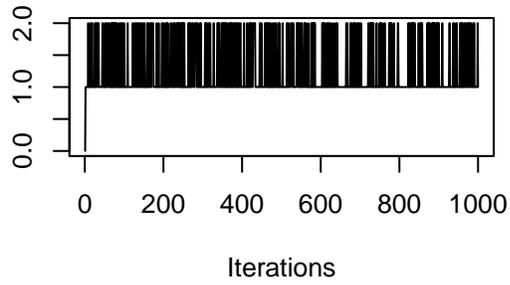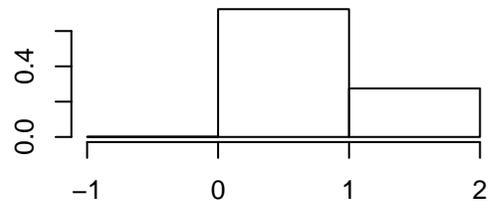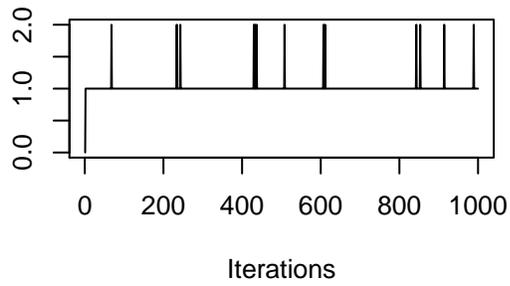
Trace of var1

Iterations

Density of var1

N = 1000   Bandwidth = 0.006366

Trace of var2

Iterations

Density of var2

N = 1000   Bandwidth = 0.006915

Trace of var3

Iterations

Density of var3

N = 1000   Bandwidth = 0.007049

```
plot(samples_mu)
```

Trace of var1

Iterations

Density of var1

N = 1000   Bandwidth = 0.0283

Trace of var2

Iterations

Density of var2

N = 1000   Bandwidth = 0.03797

Trace of var3

Iterations

Density of var3

N = 1000   Bandwidth = 0.02877

```r
plot(samples_z[,1:2])
```

Trace of var1

Density of var1

Trace of var2

Density of var2