STA 602. HW08

Yicheng Shen

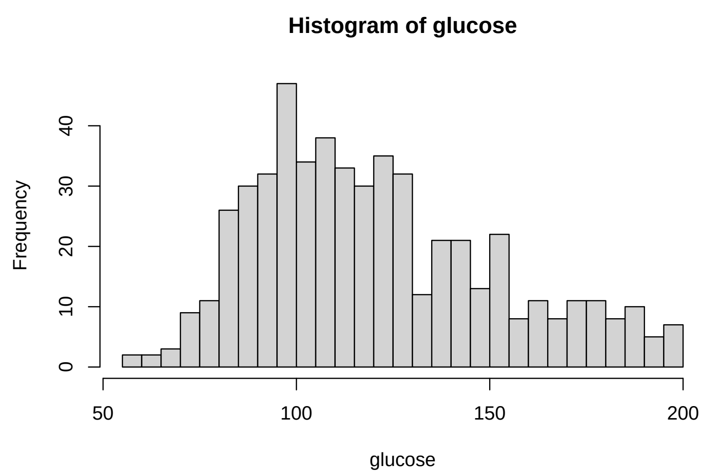
10/28/2022

[1] PH 6.2

(a)

```
glucose <- scan("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/glucose.dat")
```

```
hist(glucose, breaks = 50)
```



This histogram shows that the distribution is right skewed and not perfectly normal.

(b) First we find the full conditional distributions of X_i:

$$
\begin{aligned}
P(X_i = 1 \mid y_i, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) &= \frac{P(X_i = 1 \mid p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) \times p(y_i \mid X_i = 1, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2)}{P(y_i \mid p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2)} \\
&= \frac{P(X_i = 1 \mid p) \times p(y_i \mid X_i = 1, \theta_1, \sigma_1^2)}{P(X_i = 1 \mid p) \times p(y_i \mid X_i = 1, \theta_1, \sigma_1^2) + P(X_i = 0 \mid p) \times p(y_i \mid X_i = 0, \theta_2, \sigma_2^2)} \\
&= \frac{p \times \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(y_i-\theta_1)^2}{2\sigma_1^2})}{p \times \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(y_i-\theta_1)}{2\sigma_1^2}) + (1-p) \times \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp(-\frac{(y_i-\theta_2)^2}{2\sigma_2^2})} \\
P(X_i = 2 \mid y_i, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) &= \frac{(1-p) \times \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp(-\frac{(y_i-\theta_2)^2}{2\sigma_2^2})}{p \times \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(y_i-\theta_1)}{2\sigma_1^2}) + (1-p) \times \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp(-\frac{(y_i-\theta_2)^2}{2\sigma_2^2})} \\
x_i \mid y_i, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2 \; &= 1 \;\text{ if Bernoulli}(\frac{p \times \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(y_i-\theta_1)^2}{2\sigma_1^2})}{p \times \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(y_i-\theta_1)}{2\sigma_1^2}) + (1-p) \times \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp(-\frac{(y_i-\theta_2)^2}{2\sigma_2^2})}) = 1 \\
x_i \mid y_i, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2 \; &= 2 \;\text{ if Bernoulli}(\frac{p \times \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(y_i-\theta_1)^2}{2\sigma_1^2})}{p \times \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp(-\frac{(y_i-\theta_1)}{2\sigma_1^2}) + (1-p) \times \frac{1}{\sqrt{2\pi\sigma_2^2}} \exp(-\frac{(y_i-\theta_2)^2}{2\sigma_2^2})}) = 0
\end{aligned}
$$

Then we derive the full conditional of p.

For total n = 532, we can denote $\boldsymbol{x} = (x_1, \cdots, x_n)$ and $\boldsymbol{y} = (y_1, \cdots, y_n)$:

$$
\begin{aligned}
p(p \mid \boldsymbol{x}, \boldsymbol{y}, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) &\propto p(p) \times p(\boldsymbol{x}, \boldsymbol{y}, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2 \mid p) \\
&\propto p(p) \times p(\boldsymbol{x} \mid p) p(\boldsymbol{y} \mid \boldsymbol{x}, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) p(\theta_1, \theta_2, \sigma_1^2, \sigma_2^2) \\
\text{Only need to be related to p } &\propto p(p) \times p(\boldsymbol{x} \mid p) \\
\text{Remember that } p &\sim Beta(a, b), \quad p(x_i|p) = p^{2-x_i}(1-p)^{x_i-1} \text{ for } x_i = 1, 2 \\
&\propto p^{a-1}(1-p)^{b-1} \times \prod_{i=1}^{n} p^{2-x_i}(1-p)^{x_i-1} \\
&\propto p^{a-1}(1-p)^{b-1} \times p^{2n-\sum x_i}(1-p)^{\sum x_i - n} \\
&= p^{a+2n-\sum x_i - 1}(1-p)^{b+\sum x_i - n - 1} \\
\text{Recognize the kernel } &\sim \text{Beta}(a + 2n - \sum x_i, b + \sum x_i - n)
\end{aligned}
$$

Then we compute full conditionals of two θ:

We first define two sets as $\boldsymbol{y}_1 = \{y_i \in \boldsymbol{y} \text{ when } x_i = 1\}$ and $\boldsymbol{y}_2 = \{y_i \in \boldsymbol{y} \text{ when } x_i = 2\}$.

We also define size $n_1 = \sum_{i=1}^{n} I_{(x_i=1)}$ and $n_2 = \sum_{i=1}^{n} I_{(x_i=2)}$, where I is indicator, and $\bar{y}_j = \frac{1}{n_j}\sum_{y_i \in \boldsymbol{y}_j} y_i$:

$$
\begin{aligned}
p(\theta_1 \mid \boldsymbol{x}, \boldsymbol{y}, p, \theta_2, \sigma_1^2, \sigma_2^2) &\propto p(\theta_1) \times \prod_{i=1}^{n} p(y_i \mid x_i, p, \theta_1, \theta_2, \sigma_1^2, \sigma_2^2) \\
\text{We consider the case when } X_i = 1 \qquad &\propto \exp\left(-\frac{(\theta_1-\mu_0)^2}{2\tau_0^2}\right) \times \prod_{y_i \in \boldsymbol{y}_1} \exp\left(-\frac{(y_i-\theta_1)^2}{2\sigma_1^2}\right) \\
\text{Everything is know except } \theta \qquad &\propto \exp\left(-\frac{(\theta_1-\mu_0)^2}{2\tau_0^2}\right) \exp\left(-\frac{\sum_{y_i \in \boldsymbol{y}_1}(y_i-\theta_1)^2}{2\sigma_1^2}\right) \\
\text{Completion of square} \qquad &\propto N(\mu_{n1}, \tau_{n1}^2) \\
\text{where} \qquad \mu_{n1} &= (\frac{1}{\tau_0^2}\mu_0 + \frac{n_1}{\sigma_1^2}\bar{y}_1) \times \tau_{n1}^2 \\
\tau_{n1}^2 &= \frac{1}{\frac{1}{\tau_0^2} + \frac{n_1}{\sigma_1^2}} \\
\text{Similarly } p(\theta_2 \mid \boldsymbol{x}, \boldsymbol{y}, p, \theta_1, \sigma_1^2, \sigma_2^2) &\propto N(\mu_{n2}, \tau_{n2}^2) \\
\text{where} \qquad \mu_{n2} &= (\frac{1}{\tau_0^2}\mu_0 + \frac{n_2}{\sigma_2^2}\bar{y}_2) \times \tau_{n2}^2 \\
\tau_{n2}^2 &= \frac{1}{\frac{1}{\tau_0^2} + \frac{n_2}{\sigma_2^2}}
\end{aligned}
$$

Then we compute full conditionals of two σ^2 or the inverse $1/\sigma^2$:

$$
\begin{aligned}
p(1/\sigma_1^2 | x_1, ..., x_n, y_1, ..., y_n, \theta_1, \theta_2, \sigma_2^2, p) &= p(1/\sigma_1^2 | \boldsymbol{y}_1, \theta_1) \\
&\propto p(1/\sigma_1^2) \prod_{y_i \in \boldsymbol{y}_1}^{n_1} p(y_i | \theta_1, \sigma_1^2) \\
\text{Remeber IG for } 1/\sigma^2 \qquad &\propto (1/\sigma_1^2)^{\nu_0/2-1} \exp(-\frac{\nu_0\sigma_0^2}{2\sigma_1^2}) \times (1/\sigma_1^2)^{n_1/2} \exp(-\frac{\sum_{k=1}^{n_1}(y_i-\theta_1)^2}{2\sigma_1^2}) \\
&= (1/\sigma_1^2)^{(\nu_0+n_1)/2-1} \exp(-\frac{\nu_0\sigma_0^2 + \sum(y_i-\theta_1)^2}{2\sigma_1^2}) \\
&\sim Gamma(\frac{\nu_0+n_1}{2}, \frac{\nu_0\sigma_0^2 + \sum_{y_i \in \boldsymbol{y}_1}^{n_1}(y_i-\theta_1)^2}{2}) \\
p(\sigma_1^2 | x_1, ..., x_n, y_1, ..., y_n, \theta_1, \theta_2, \sigma_2^2, p) &\sim IG(\frac{\nu_0+n_1}{2}, \frac{\nu_0\sigma_0^2 + \sum_{y_i \in \boldsymbol{y}_1}^{n_1}(y_i-\theta_1)^2}{2}) \\
\text{Similarly } p(\sigma_2^2 | x_1, ..., x_n, y_1, ..., y_n, \theta_1, \theta_2, \sigma_1^2, p) &\sim IG(\frac{\nu_0+n_2}{2}, \frac{\nu_0\sigma_0^2 + \sum_{y_i \in \boldsymbol{y}_2}^{n_2}(y_i-\theta_2)^2}{2})
\end{aligned}
$$

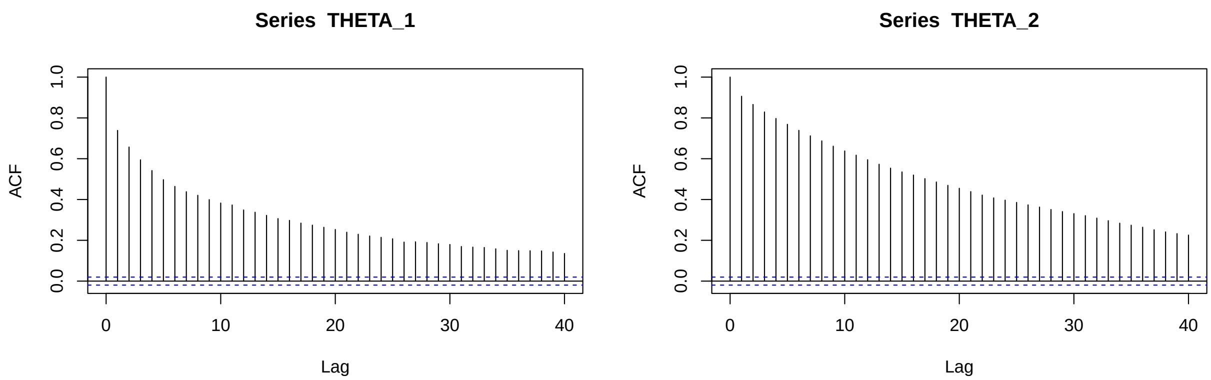
(c) Gibbs sampling is done below:

```
set.seed(8848)

# prior
a = b = 1
mu0 = 120
t20 = 200
s20 = 1000
nu0 = 10

S = 10000
```

```r
y = glucose
n = length(y)
# initialize
p = 1/2
theta1 = theta2 = mean(y)
s21 = s22 = var(y)

THETA1 = THETA2 = numeric(S)
THETA_1 = THETA_2 = numeric(S)
Empirical = numeric(S)

# Gibbs sampling
for (t in 1:S) {
  # draw X
  p1 = p * dnorm(y, theta1, sqrt(s21))
  p2 = (1 - p) * dnorm(y, theta2, sqrt(s22))
  bernoulli_p = p1 / (p1 + p2)
  X = rbinom(n, 1, bernoulli_p)

  # Classify Y based on X
  n1 = sum(X)
  n2 = n - n1
  y1 = y[X == 1] # bernoulli give 1 equals to X = 1
  y2 = y[X == 0] # bernoulli give 0 equals to X = 2
  ybar1 = mean(y1)
  ybar2 = mean(y2)
  yvar1 = var(y1)
  yvar2 = var(y2)

  # draw p
  p = rbeta(1, a + n1, b + n2)

  # draw thetas
  t2n1 = 1 / (1 / t20 + n1 / s21)
  mun1 = (mu0 / t20 + n1 * ybar1 / s21) / (1 / t20 + n1 / s21)
  theta1 = rnorm(1, mun1, sqrt(t2n1))

  t2n2 = 1 / (1 / t20 + n2 / s22)
  mun2 = (mu0 / t20 + n2 * ybar2 / s22) / (1 / t20 + n2 / s22)
  theta2 = rnorm(1, mun2, sqrt(t2n2))

  # draw sigma^2s
  nun1 = nu0 + n1
  s2n1 = (nu0 * s20 + (n1 - 1) * yvar1 + n1 * (ybar1 - theta1)^2) / nun1
  s21 = 1 / rgamma(1, nun1 / 2, s2n1 * nun1 / 2)

  nun2 = nu0 + n2
  s2n2 = (nu0 * s20 + (n2 - 1) * yvar2 + n2 * (ybar2 - theta2)^2) / nun2
  s22 = 1 / rgamma(1, nun2 / 2, s2n2 * nun2 / 2)

  # draws for part d
  x_draw = runif(1) < p # binary based on p
  y_draw = ifelse(x_draw, rnorm(1, theta1, sqrt(s21)), rnorm(1, theta2, sqrt(s22)))
```

```
  # Store values
  THETA1[t] = theta1; THETA2[t] = theta2
  THETA_1[t] = min(theta1,theta2); THETA_2[t] = max(theta1,theta2)
  Empirical[t] = y_draw
}
```

```
par(mfrow=c(1,2))
acf(THETA_1)
acf(THETA_2)
```



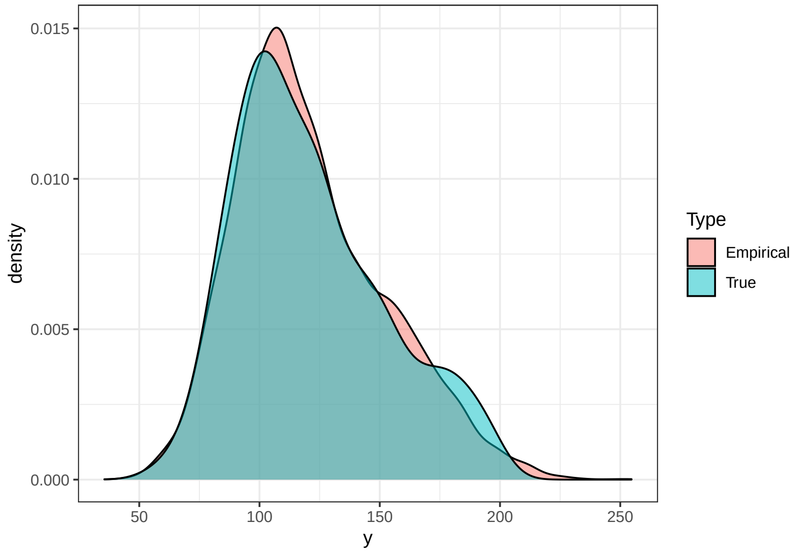

```
c(effectiveSize(THETA_1), effectiveSize(THETA_2))
```

```
##     var1     var1
## 391.6257 211.5284
```

The effective sample size for $\theta_1^{(s)}$ is 391.6257, and ess for $\theta_2^{(s)}$ is 211.5284. From the ACF plot, the latter one (max) decays more slowly than the former one (min).

(d) This two component mixture model is a good fit for the glucose data since the empirical distribution is very much coherent with the true one.

```
compare <- rbind(data.frame(y = Empirical, Type = 'Empirical'),
                 data.frame(y = glucose, Type = 'True'))
ggplot(compare, aes(x = y, fill = Type)) + geom_density(alpha = 0.5)
```



[2] PH 6.3

The model is:

$$Z_i = \beta x_i + \epsilon_i$$
$$Y_i = \delta_{(c,\infty)}(Z_i),$$

where β and c are unknown coefficients, $\epsilon_1, ..., \epsilon_n \sim i.i.d.normal(0, 1)$ and $\delta_{(c,1)}(z) = 1$ if $z > c$ and equals zero otherwise.

(a) Since β only depends on z and x through the first equation:

$$\begin{aligned}
p(\beta|y, x, z, c) &\propto p(\beta) \times p(z|\beta, x) \\
&\propto \exp(-\frac{\beta^2}{2\tau_\beta^2}) \times \exp(-\frac{\sum(z_i - \beta x_i)^2}{2}) \\
&\propto \exp(-\frac{\beta^2 + \tau_\beta^2 \sum(z_i - \beta x_i)^2}{2\tau_\beta^2}) \\
&\propto \exp(-\frac{\beta^2 + \tau_\beta^2 \sum(z_i^2 + \beta^2 x_i^2 + 2z_i x_i \beta)}{2\tau_\beta^2}) \\
&\propto \exp(-\frac{\beta^2 + \tau_\beta^2 \sum z_i^2 + \beta^2 \tau_\beta^2 \sum x_i^2 + \beta\tau_\beta^2 \sum 2z_i x_i)}{2\tau_\beta^2}) \\
&\propto \exp(-\frac{\beta^2(1 + \tau_\beta^2 \sum x_i^2) + \tau_\beta^2 \sum z_i^2 + 2\beta\tau_\beta^2 \sum z_i x_i)}{2\tau_\beta^2}) \\
&\propto \exp[-\frac{(\beta - \frac{\tau_\beta^2 \sum z_i x_i}{1+\tau_\beta^2 \sum x_i^2})^2}{2\tau_\beta^2/(1 + \tau_\beta^2 \sum x_i^2)}] \\
\text{Recognize the kernel } &\sim N(\frac{\tau_\beta^2 \sum z_i x_i}{1 + \tau_\beta^2 \sum x_i^2}, \frac{\tau_\beta^2}{1 + \tau_\beta^2 \sum x_i^2})
\end{aligned}$$

(b) From the 2nd equation, it is noticed that c only has dependence on y and z.

In specific, c should be higher than any z_i when $y_i = 0$, and lower than any z_i when $y_i = 1$.

Now denote $a = \max\{z_i : y_i = 0\}, b = \min\{z_i : y_i = 1\}$.

$$\begin{aligned} p(c|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{z}, \beta) &\propto p(c|\boldsymbol{y}, \boldsymbol{z}) \\ &\propto p(c) \times p(y|z, c) \\ &\propto N(0, \tau_c) \times \delta_{(a,b)}(c) \end{aligned}$$

The full conditional of c is thus proportional to this $p(c)$ but constrained by a and b.

In other words, this full conditional is a constrained normal density and lie in the interval (a, b).

Then we compute the full conditional distribution of z:

The model suggests that $Z_i \sim N(\beta x_i, 1)$. If we are given c and $Y_i = y_i$, we can trace back about the interval of Z_i that gives y_i. For example, if $y_i = 0$, Z_i should be in $(-\infty, c)$, and f $y_i = 1$, Z_i should be in (c, ∞):

$$p(z_i|\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{z}_{-i}, \beta, c) \propto \begin{cases} N(\beta x_i, 1) \times \delta_{(c,\infty)}(z_i) & y_i = 1 \\ N(\beta x_i, 1) \times \delta_{(-\infty,c)}(z_i) & y_i = 0 \end{cases}$$

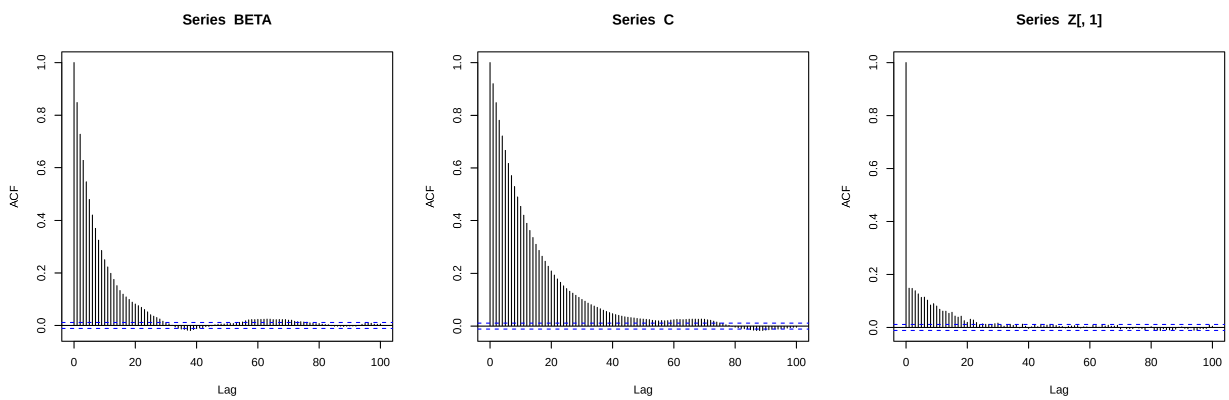
(c) Use the full conditionals before to do Gibbs sampling:

```
divorce <- read.table("http://www2.stat.duke.edu/~pdh10/FCBS/Exercises/divorce.dat")
```

```
n = nrow(divorce)
x = divorce[, 1]
y = divorce[, 2]
tau_c_sq = tau_beta_sq = 16

S <- 30000
BETA = NULL
C = NULL
Z = matrix(NA, nrow = S, ncol = n)

# initialize
beta = 1
c = 1
z = rep(0, n)

for (t in 1:S) {
  # draw beta
  Mu = tau_beta_sq * sum(z * x) / (1 + tau_beta_sq * sum(x^2))
  Var = tau_beta_sq / (1 + tau_beta_sq * sum(x ^ 2))
  beta = (rnorm(1, Mu, sqrt(Var)))

  # draw c
  z0 = subset(z, y == 0)  # get subset
  z1 = subset(z, y == 1)
  a = max(z0)
  b = min(z1)
  u = runif(1, pnorm((a-0)/sqrt(tau_c_sq)), pnorm((b-0)/sqrt(tau_c_sq)))
  c = 0 + sqrt(tau_beta_sq) * qnorm(u) # method from 12.1.1

  # draw z
  u0 = runif(n, 0, pnorm(c-x*beta))
  u1 = runif(n, pnorm(c-x*beta), 1)
  z0 = x*beta + qnorm(u0) # ez + qnorm(u)
```

```
  z1 = x*beta + qnorm(u1)
  z = z0*(as.numeric(!y))+z1*y

  BETA[t] = beta
  C[t] = c
  Z[t, ] = z
}
```

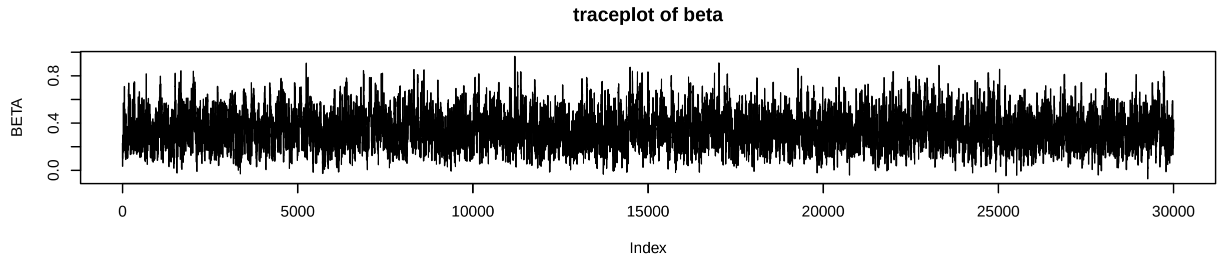
```
par(mfrow=c(1,3))
acf(BETA, lag.max = 100)
acf(C, lag.max = 100)
acf(Z[,1], lag.max = 100)
```

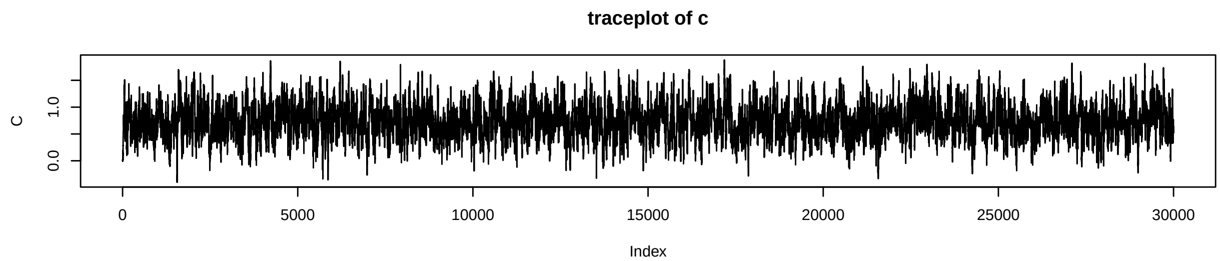


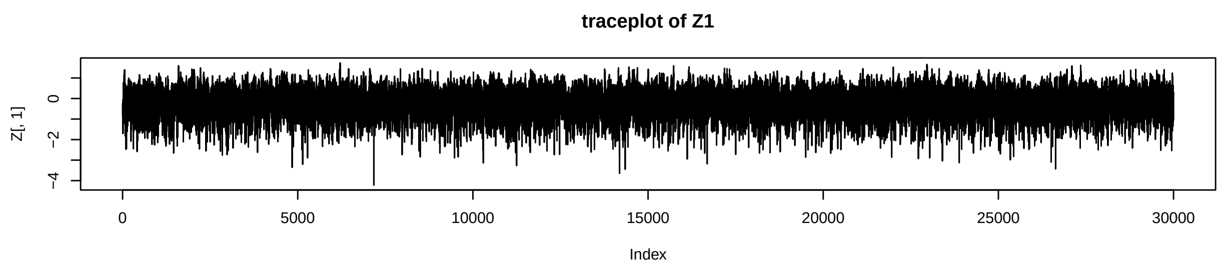

```
c(effectiveSize(BETA), effectiveSize(C), effectiveSize(Z[,1]))
```

```
##      var1     var1     var1
## 2122.837 1226.260 6263.318
```

```
par(mfrow=c(3,1))
plot(BETA, main = 'traceplot of beta', type = "l")
plot(C, main = 'traceplot of c', type = "l")
plot(Z[,1], main = 'traceplot of Z1', type = "l")
```







We would need around 30000 iterations for at least 1000 effective sample sizes for every parameter. ACF is good enough after 40 or 50 lags for β and c, so these two are less efficient than z. The mixing seems good enough considering the diagnostic plots above.

(d) A 95% posterior confidence interval for β and posterior and $Pr(\beta > 0|\mathbf{y}, \mathbf{x})$, which is a very high probability, are given below.

```
# 95% CI for beta
quantile(BETA, c(0.025, 0.975))
```

```
##      2.5%     97.5%
## 0.1083823 0.6505118
```

```
# Pr(beta > 0|y, x)
mean(BETA > 0)
```

```
## [1] 0.9989
```