# Computational Statistics
# Group Assessment

Group 11:
Kyriacos Xanthos 01389741
Xinhua Jia 01517829
Yiheng Jia 02035952
Hins Qi 02140607
Wanchen Yue 02080856
everyone contributed equally

December 3, 2021

## Question.1

Under the assumption that $\epsilon \sim \mathcal{N}(0, \omega)$ for $i = 1, ...30$, we have $Y_i$ is normally distributed conditioning on $X_{1i}$, $X_{2i}$, $\beta_1$, $\beta_2$ and $\omega$.

The likelihood of each $Y_i$ given by $X_{1i}, X_{2i}$ and $\omega$ is given by

$$P\left(Y_i \mid X_{1i}, X_{2i}, \beta_1, \beta_2, \omega\right) = \frac{1}{\sqrt{2\pi w}} \exp\left[-\frac{\left(Y_i - \left(\beta_1 X_{1i} + \beta_2 X_{2i}\right)\right)^2}{2w}\right]$$

Also, by independence, we can have the joint density for the prior which is

$$p\left(\beta_1, \beta_2, w\right) = \frac{1}{(5 \cdot \sqrt{2\pi})^2} \exp\left[-\frac{1}{2}\left(\frac{\beta_1^2 + \beta_2^2}{10}\right)\right] \exp(-w)$$

We apply the Bayes' rule to derive the following posterior distribution which is propotional to the product of the likelihood and the joint prior distribution:

$$\begin{aligned} P\left(\beta_1, \beta_2, \omega \mid \mathcal{D}\right) &\propto \left[\prod_i^n P\left(Y_i \mid X_{ji}, \beta_1, \beta_2, \omega\right)\right] p\left(\beta_1, \beta_2, \omega\right) \\ &\propto \frac{1}{\omega^{\frac{n}{2}}} \exp\left[-\frac{\sum_i \left[Y_i - \left(\beta_1 X_{1i} + \beta_2 X_{2i}\right)\right]^2}{2\omega}\right] \exp\left(-\frac{1}{2}\left(\frac{\beta_1^2 + \beta_2^2}{10}\right)\right) \exp\left(-\omega\right) \end{aligned}$$

We will use this posterior distribution for our density $f$ in question 2.

## Question.2

---

**Algorithm 1** Metropolis-Hastings algorithm

---

**Input:** Unnormalised density f; proposal distribution q; starting value $X^0$
**Output:** Markov Chain.
**for** *t=1,2,...* **do**
  Let $Y^t \sim q(X^{t-1}, .)$ Let $U \sim U(0,1)$
  **if** $U \leq \min\left(\frac{f(Y^t)q(Y^t, X^{t-1})}{f(X^{t-1})q(X^{t-1}, Y^t)}, 1\right)$ **then**
    $X^t = Y^t$
  **else**
    $X^t = Y^{t-1}$
  **end**
**end**

---

For $\omega$, We used the proposal in the Metropolis-Hastings algorithm $Y = X\epsilon$, $\epsilon \sim g$, where $g$ is a density with positive support for the $\omega$ since $\omega$ follows a Gamma distribution which is non negative: We have: $E[h(Y) \mid X = x] = E[h(X\epsilon) \mid X = x] = \int_0^\infty h(x\epsilon)g(\epsilon)d\epsilon$, subsititute in $y = x\epsilon$ gives $\mathrm{E}[h(Y) \mid X = x] = \int_0^\infty h(y)g(y/x)\frac{1}{x}dy$. Thus we have the corresponding accceptance probability for the Metropolis-Hasting algorithm:

$$\frac{f(y)q(y,x)}{f(x)q(x,y)} = \frac{f(\epsilon x)}{f(x)}\frac{g(1/\epsilon)/(x\epsilon)}{g(\epsilon)/x} = \frac{f(\epsilon x)}{f(x)}\frac{g(1/\epsilon)}{g(\epsilon)}\frac{1}{\epsilon}.$$

For $\beta_1$ and $\beta_2$, we have $\beta_i \sim N(0,5)$ with mean 0 and standard deviation 5.Then we deduce our proposal distribution $q(X,Y) = \operatorname{dexp}\left(\frac{Y_3}{X_3}, \text{ rate } = 1\right)/X_3$. Where *dexp* is the exponential distribution. Conduct the above Metropolis-Hasting algorithm, we can generate the Markov Chain: We have the effective sample size as follows:

| | $\beta_1$ | $\beta_2$ | $\omega$ |
|---|---|---|---|
| **Effective Size** | 25.8 | 28.1 | 37.9 |

It can be clearly seen from 1 that our algorithm has not converged, mainly because our implementation might have errors. We display 2 different chains with different starting values but we tried many more and they all give the same result.

We can see from 2 that our distributions at least have the expected distribution space since $\omega$ can only take positive values, which again was the reason we used the gamma distribution as the proposal for $\omega$.

Figure 3 shows the ACF plots for the model. As discussed above the model has not converged so we cannot interpret meaningful results from this figure other than seeing that the ACF decreases as expected.

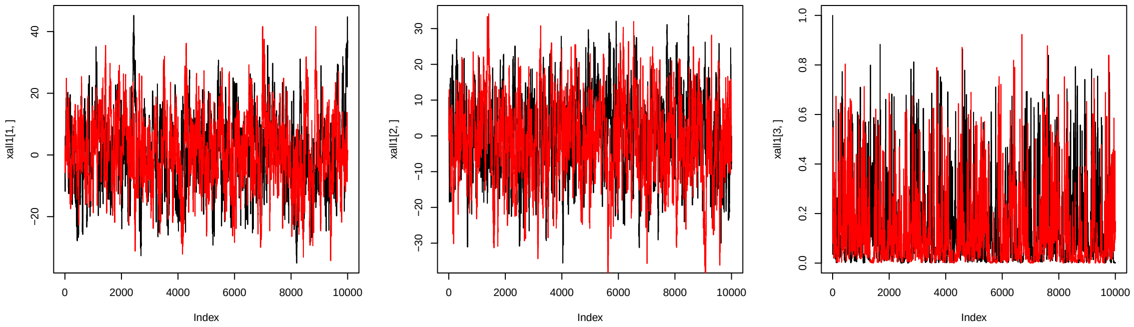


Figure 1: Traceplots for 2 different chains (red and black) for $\beta_1$, $\beta_2$ and $\omega$. It can be clearly seen that our algorithm has not converged.

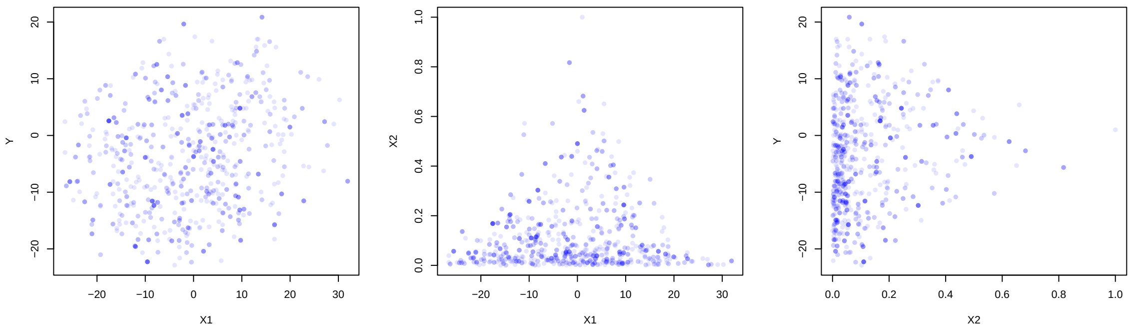


Figure 2: Graphical interpretation of the distributions for $\beta_1$, $\beta_2$ and $\omega$

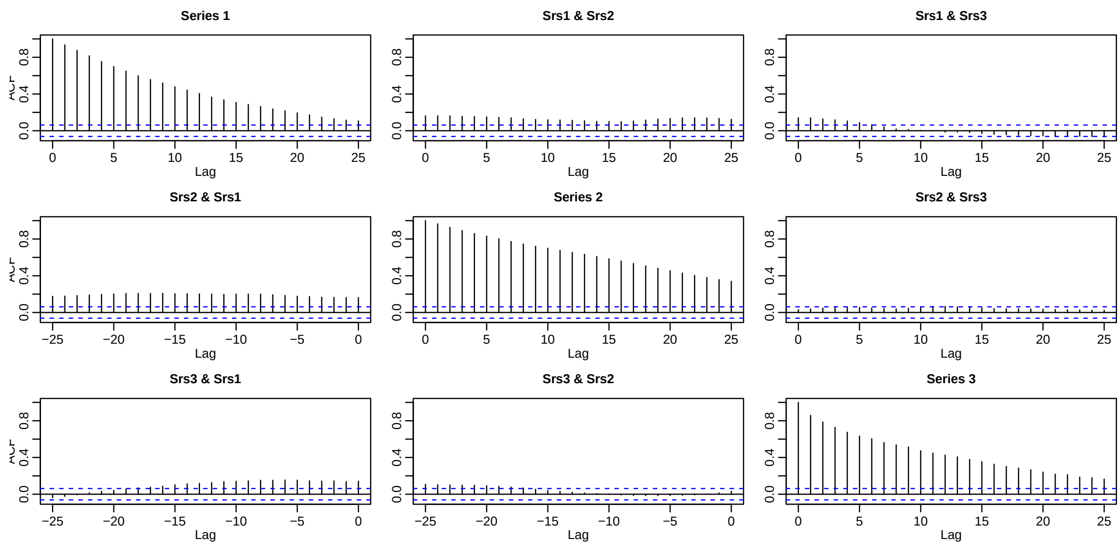


Figure 3: ACF plots for the 3 different parameters in the model.

## Question.3

For calculating the posterior mean of parameter space $\theta = \{\beta_1, \beta_2, \omega\}$, we need

$$E(\theta|y) = \int \theta\pi(\theta|y)d\theta$$

For each parameter we give other parameter a estimated value, then we can estimate the posterior mean:
$E(\beta_1 \mid D, \beta_2, w) = \int \beta_1 \cdot p(\beta_1 \mid D, \beta_2, w)\, d\beta_2$
$E(\beta_2 \mid D, \beta_1, w) = \int \beta_2 \cdot p(\beta_2 \mid D, \beta_1, w)\, d\beta_2$
$E(w \mid D, \beta_1, \beta_2) = \int w \cdot p(w \mid D, \beta_1, \beta_2)\, dw$
Also, to have a better estimation, we shall omit the first 100 terms.

# A Code for Q2

```
## 2) Three-dimensional target density
#########################

f<-function(x) -30/2*log(x[3])-sum(dataset$Y-
 (x[1]*dataset$X1+x[2]*dataset$X2))/2*x[3]-0.5*(x[1]^2+x[2]^2)
 /10-x[3]

rq <- function(x) c(x[1]+rnorm(1,mean=0,sd=5),x[2]+rnorm(1,mean=0,sd=5),

x[3]*rexp(1,rate=1))
q <- function(x,y) dexp(y[3]/x[3], rate=1)/x[3]


x <- c(1,1,1);
n <- 1e3
xall <- matrix(NA,nrow=3,ncol=n+1)
xall[,1] <- x

for(i in 1:n){
  y <- rq(x)
  if (runif(1)<f(y)/f(x)*(q(y,x)/q(x,y))){
    x <- y
  }
  xall[,i+1] <- x
}

## plot of the resulting joint samples
plot(xall[1,],xall[2,],col = rgb(red = 0, green = 0, blue = 1, alpha = 0.1),
     pch = 16, xlab="x", ylab="y")
plot(xall[2,],xall[3,],col = rgb(red = 0, green = 0, blue = 1, alpha = 0.1),
     pch = 16, xlab="x", ylab="y")
plot(xall[1,],xall[3,],col = rgb(red = 0, green = 0, blue = 1, alpha = 0.1),
     pch = 16, xlab="x", ylab="y")
## trace plots
par(mfrow=c(1,3))
plot(xall[1,], type="l")
plot(xall[2,], type="l")
plot(xall[3,], type="l")

# auto-correlation and acf
effectiveSize(t(xall))
acf(t(xall))
# 3D PLOT
persp(xall[1,], xall[2,], xall[3,], xlab='X Variable', ylab='Y Variable',

zlab='Z Variable',
      main='3D Plot', col='pink', shade=.4)
```