STA 602 Lab 8

Yicheng Shen

07 November, 2022

Ex.1

$$
\begin{aligned}
\mu_n &= (I + n\Sigma^{-1})^{-1}(n\Sigma^{-1}\bar{X}) \\
\Lambda_n &= (I + n\Sigma^{-1})^{-1}
\end{aligned}
$$

$$
\begin{aligned}
\theta_1|\theta_2, X_1 \cdots X_n, \rho \sim N(\mu_{1n}, \lambda_{1n}) \\
\mu_{1n} &= \mu_n^1 + \Lambda_n^{12}(\Lambda_n^{22})^{-1}(\theta_2 - \mu_n^2) \\
\Lambda_{1n} &= \Lambda_n^{12}(\Lambda_n^{22})^{-1}\Lambda_n^{21}
\end{aligned}
$$

Ex.2

$$
\begin{aligned}
\theta_2|\theta_1, X_1 \cdots X_n, \rho \sim N(\mu_{2n}, \Sigma_{2n}) \\
\mu_{2n} &= \mu_n^2 + \Lambda_n^{21}(\Lambda_n^{11})^{-1}(\theta_1 - \mu_n^1) \\
\Sigma_{2n} &= \Lambda_n^{21}(\Lambda_n^{11})^{-1}\Lambda_n^{12}
\end{aligned}
$$

Ex.3

A high correlation between θ, namely a large value for ρ, would make the posterior mean of one θ to be highly dependent on the other.

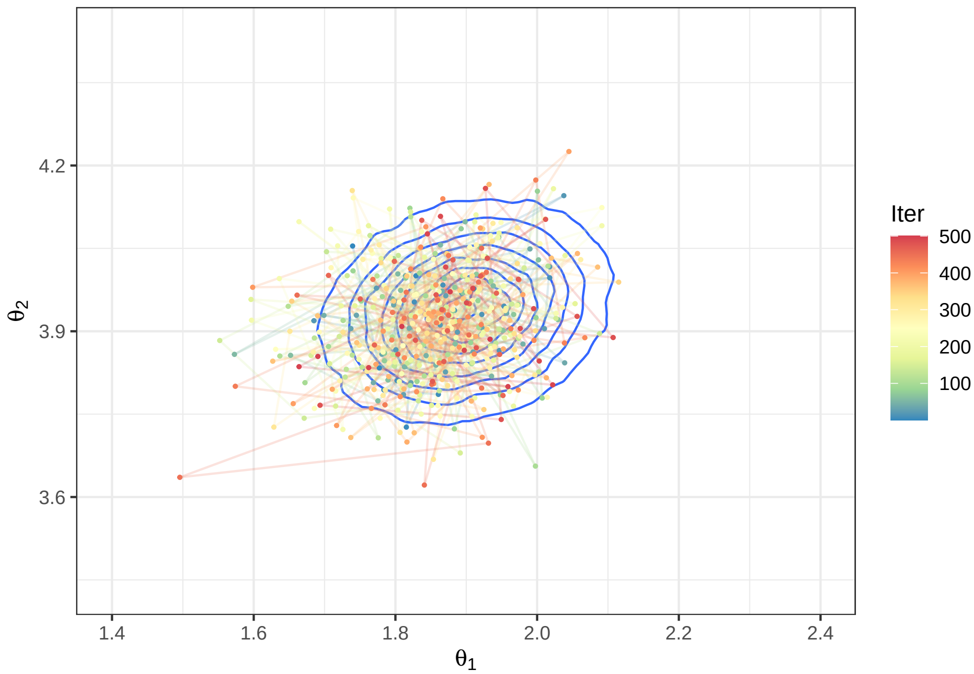
Ex.4

```
normal_gibbs_sampler <- function(S, X, rho) {

  result = matrix(NA, nrow = S, ncol = ncol(X))

  theta_1 = theta_2 = 0

  Sigma = matrix(c(1,rho,rho,1), 2, 2)

  for (t in 1:S)
  {
    mu_n = solve(matrix(c(1, 1, 1, 1), 2, 2) + n * solve(Sigma)) %*% (n*solve(Sigma) %*% matrix(colMean
    lambda_n = solve(matrix(c(1, 1, 1, 1), 2, 2) + n * solve(Sigma))
```

```
    theta_1 = rnorm(
      1,
      mean = mu_n[1, 1] + lambda_n[1, 2] %*% solve(lambda_n[2, 2]) %*% (theta_2 - mu_n[2, 1]) ,
      sd = sqrt(lambda_n[1, 1] - lambda_n[1, 2]  %*% solve(lambda_n[2, 2]) %*% lambda_n[2, 1])
    )

    theta_2 = rnorm(
      1,
      mean = mu_n[2, 1] + lambda_n[2, 1] %*% solve(lambda_n[1, 1]) %*% (theta_1 - mu_n[1, 1]) ,
      sd = sqrt(lambda_n[2, 2] - lambda_n[2, 1]  %*% solve(lambda_n[1, 1]) %*% lambda_n[1, 2])
    )

    result[t,] = c(theta_1, theta_2)
  }

  return (result)  # return a matrix of dimension S*2 containing Gibbs samples for theta1 and theta2
}
```

Draw samples with both Gibbs and HMC

```
n <- 100
rho <- 0.2
X <- MASS::mvrnorm(n = n, mu = c(2, 4), Sigma = matrix(c(1, rho, rho, 1), nrow = 2))
Sigma_post <- matrix(((1-rho^2)/((n+1-rho^2)^2 - (n^2)*(rho^2)))*c(n+1-rho^2, n*rho, n*rho, n+1-rho^2),
mu_post <- n*Sigma_post%*%matrix(c(1/(1-rho^2), -rho/(1-rho^2),
                                        -rho/(1-rho^2), 1/(1-rho^2)),
                                        nrow = 2)%*%colMeans(X)
norm_gibbs_samps <- normal_gibbs_sampler(600, X, rho)
#
true_post <- MASS::mvrnorm(n = 100000,
                           mu = mu_post,
                           Sigma = Sigma_post)
data.frame(norm_gibbs_samps) %>%
  magrittr::set_colnames(c("theta_1", "theta_2")) %>%
  dplyr::mutate(iter = 1:n()) %>%
  dplyr::filter(iter > 100) %>%
  dplyr::mutate(iter = 1:n()) %>%
  ggplot2::ggplot() +
  geom_density2d(data = data.frame(true_post) %>%
                        magrittr::set_colnames(c("true_1", "true_2")),
                 aes(x = true_1, y = true_2)) +
  geom_path(aes(x = theta_1, y = theta_2, colour = iter), alpha = 0.2, size = 0.5) +
  geom_point(aes(x = theta_1, y = theta_2, colour = iter), size = 0.5) +
  scale_color_distiller(palette = "Spectral", name = "Iter") +
  labs(x = expression(theta[1]), y = expression(theta[2])) +
  xlim(c(mu_post[1] - 0.5, mu_post[1] + 0.5)) +
  ylim(c(mu_post[2] - 0.5, mu_post[2] + 0.5))
```

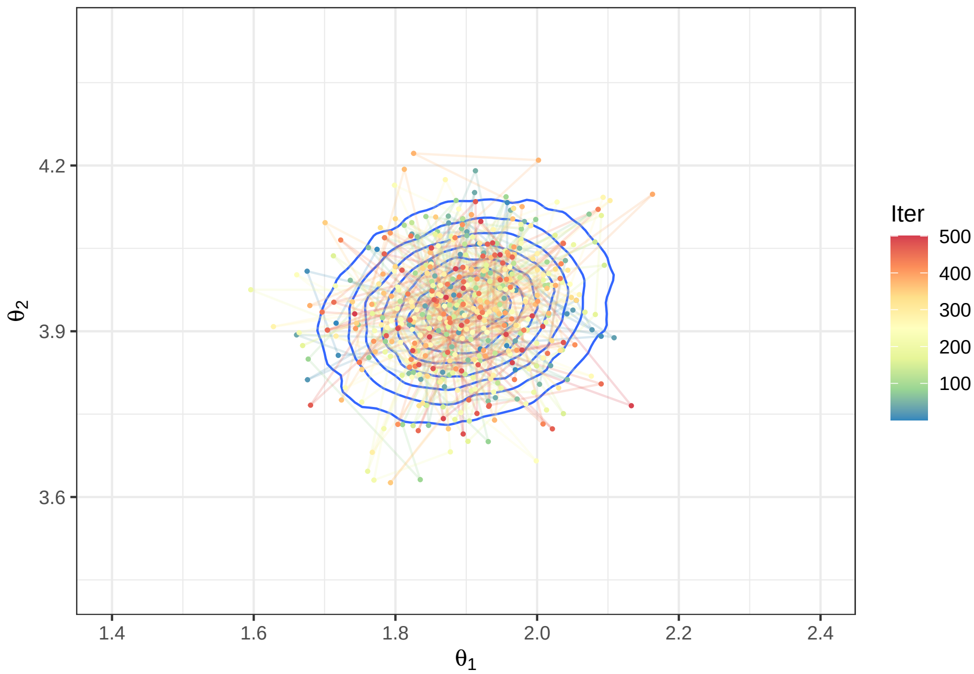



```
stan_res <- rstan::stan("lab-08-hmc_norm_example.stan", data = list(X = X,
                                                                     N = nrow(X),
                                                                     Sigma = matrix(c(1, rho, rho, 1), nrow = 2
                        chains = 1, iter = 600, warmup = 100, verbose = F, refresh = 0) %>%
            rstan::extract()
```

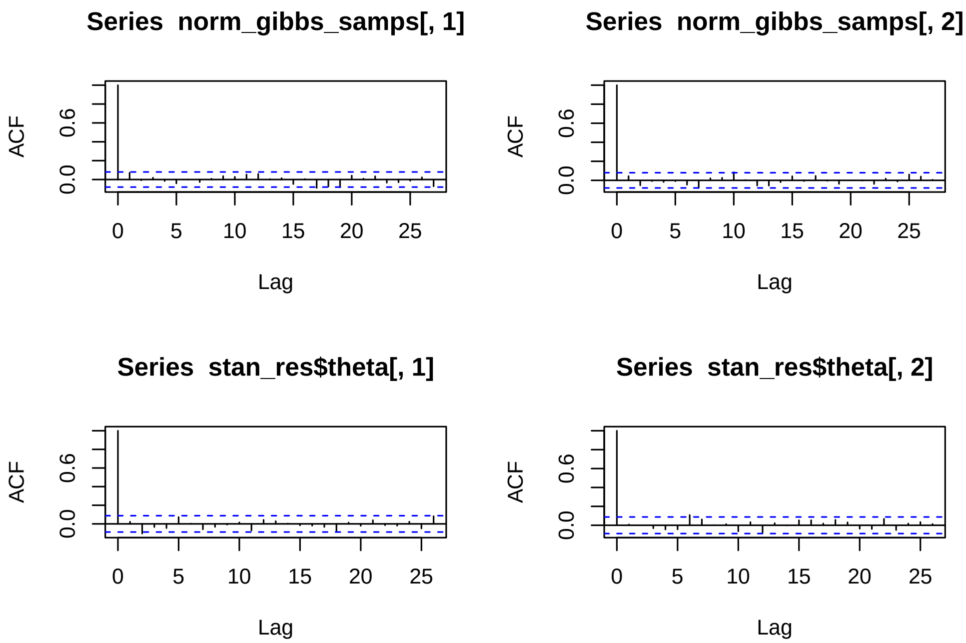
```
## Trying to compile a simple C file
```

```
## Running /Library/Frameworks/R.framework/Resources/bin/R CMD SHLIB foo.c
## clang -mmacosx-version-min=10.13 -I"/Library/Frameworks/R.framework/Resources/include" -DNDEBUG   -I
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/library/StanHeaders/incl
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/inclu
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/inclu
## /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/include/Eigen/src/Core/util/
## namespace Eigen {
## ^
## /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/include/Eigen/src/Core/util/
## namespace Eigen {
##                ^
##                ;
## In file included from <built-in>:1:
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/library/StanHeaders/incl
## In file included from /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/inclu
## /Library/Frameworks/R.framework/Versions/4.2/Resources/library/RcppEigen/include/Eigen/Core:96:10: fa
## #include <complex>
##          ^~~~~~~~~
## 3 errors generated.
## make: *** [foo.o] Error 1
```

```r
#
data.frame(stan_res$theta) %>%
  magrittr::set_colnames(c("theta_1", "theta_2")) %>%
  dplyr::mutate(iter = 1:n()) %>%
  ggplot2::ggplot() +
  geom_density2d(data = data.frame(true_post) %>%
                        magrittr::set_colnames(c("true_1", "true_2")),
                 aes(x = true_1, y = true_2)) +
  geom_path(aes(x = theta_1, y = theta_2, colour = iter), alpha = 0.2, size = 0.5) +
  geom_point(aes(x = theta_1, y = theta_2, colour = iter), size = 0.5) +
  scale_color_distiller(palette = "Spectral", name = "Iter") +
  labs(x = expression(theta[1]), y = expression(theta[2])) +
  xlim(c(mu_post[1] - 0.5, mu_post[1] + 0.5)) +
  ylim(c(mu_post[2] - 0.5, mu_post[2] + 0.5))
```




```r
par(mfrow = c(2,2))
acf(norm_gibbs_samps[,1])
acf(norm_gibbs_samps[,2])
acf(stan_res$theta[,1])
acf(stan_res$theta[,2])
```

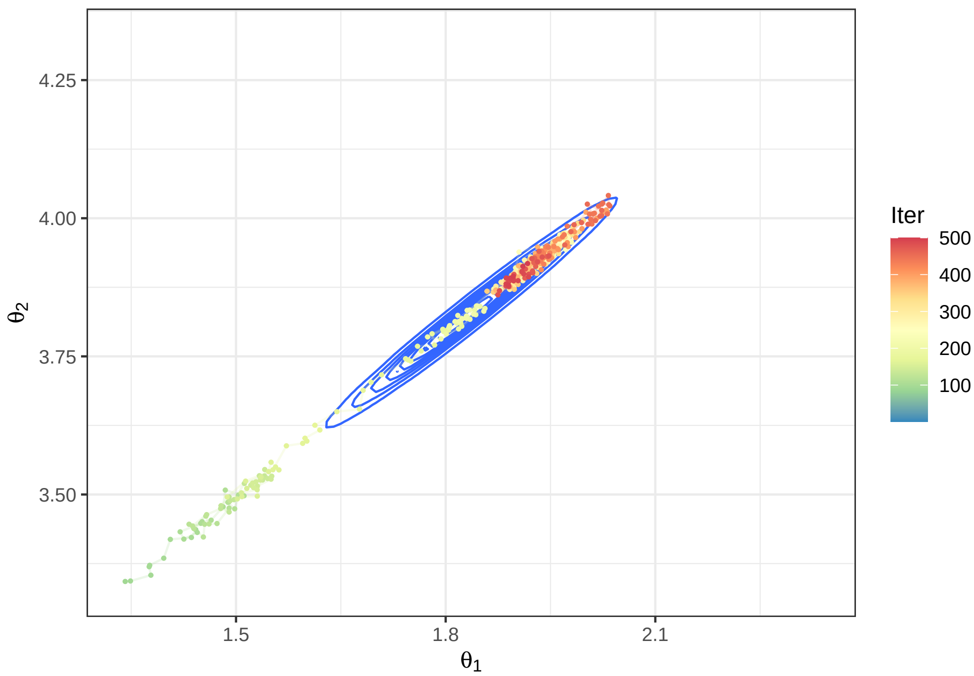


Large correlation

```
n <- 100
rho <- 0.995
X <- MASS::mvrnorm(n = n, mu = c(2, 4), Sigma = matrix(c(1, rho, rho, 1), nrow = 2))
Sigma_post <- matrix(((1-rho^2)/((n+1-rho^2)^2 - (n^2)*(rho^2)))*c(n+1-rho^2, n*rho, n*rho, n+1-rho^2),
mu_post <- n*Sigma_post%*%matrix(c(1/(1-rho^2), -rho/(1-rho^2),
                                    -rho/(1-rho^2), 1/(1-rho^2)),
                                    nrow = 2)%*%colMeans(X)
norm_gibbs_samps <- normal_gibbs_sampler(600, X, rho)
#
true_post <- MASS::mvrnorm(n = 100000,
                           mu = n*Sigma_post%*%(matrix(c(1/(1-rho^2), -rho/(1-rho^2),
                                                          -rho/(1-rho^2), 1/(1-rho^2)),
                                                          nrow = 2)%*%colMeans(X)),
                           Sigma = Sigma_post)
#
data.frame(norm_gibbs_samps) %>%
  magrittr::set_colnames(c("theta_1", "theta_2")) %>%
  dplyr::mutate(iter = 1:n()) %>%
  dplyr::filter(iter > 100) %>%
  dplyr::mutate(iter = 1:n()) %>%
  ggplot2::ggplot() +
  geom_density2d(data = data.frame(true_post) %>%
                   magrittr::set_colnames(c("true_1", "true_2")),
                 aes(x = true_1, y = true_2)) +
  geom_path(aes(x = theta_1, y = theta_2, colour = iter), alpha = 0.2, size = 0.5) +
  geom_point(aes(x = theta_1, y = theta_2, colour = iter), size = 0.5) +
  scale_color_distiller(palette = "Spectral", name = "Iter") +
```

```
  labs(x = expression(theta[1]), y = expression(theta[2])) +
  xlim(c(mu_post[1] - 0.5, mu_post[1] + 0.5)) +
  ylim(c(mu_post[2] - 0.5, mu_post[2] + 0.5))
```

```
## Warning: Removed 92 row(s) containing missing values (geom_path).

## Warning: Removed 92 rows containing missing values (geom_point).
```

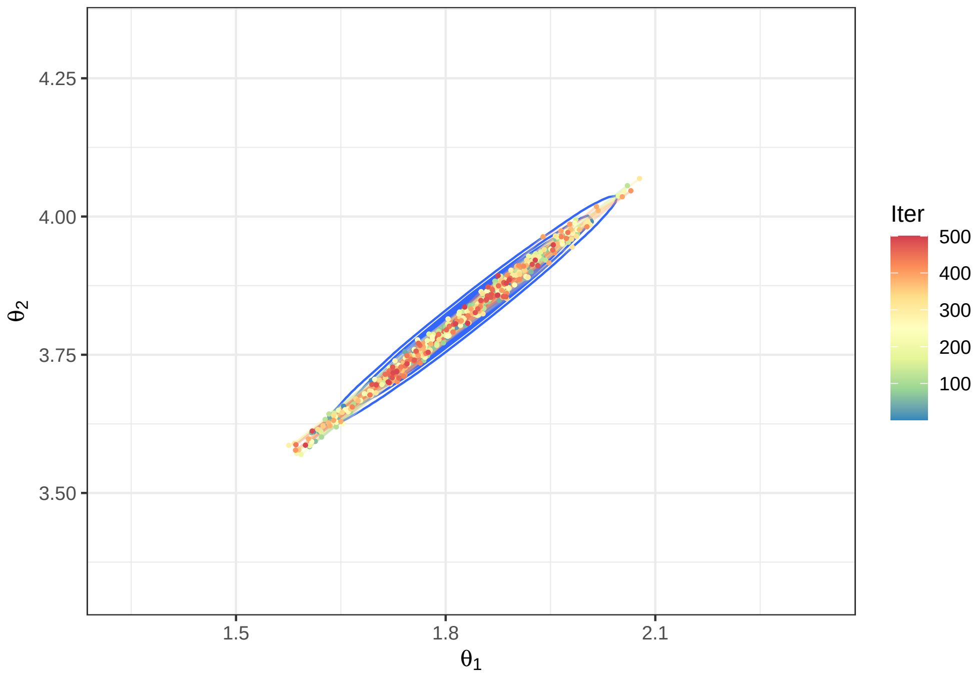



```
stan_res <- rstan::stan("lab-08-hmc_norm_example.stan", data = list(X = X,
                                                                    N = nrow(X),
                                                                    Sigma = matrix(c(1, rho, rho, 1), nrow = 2
                        chains = 1, iter = 600, warmup = 100, verbose = F, refresh = 0) %>%
            rstan::extract()
```

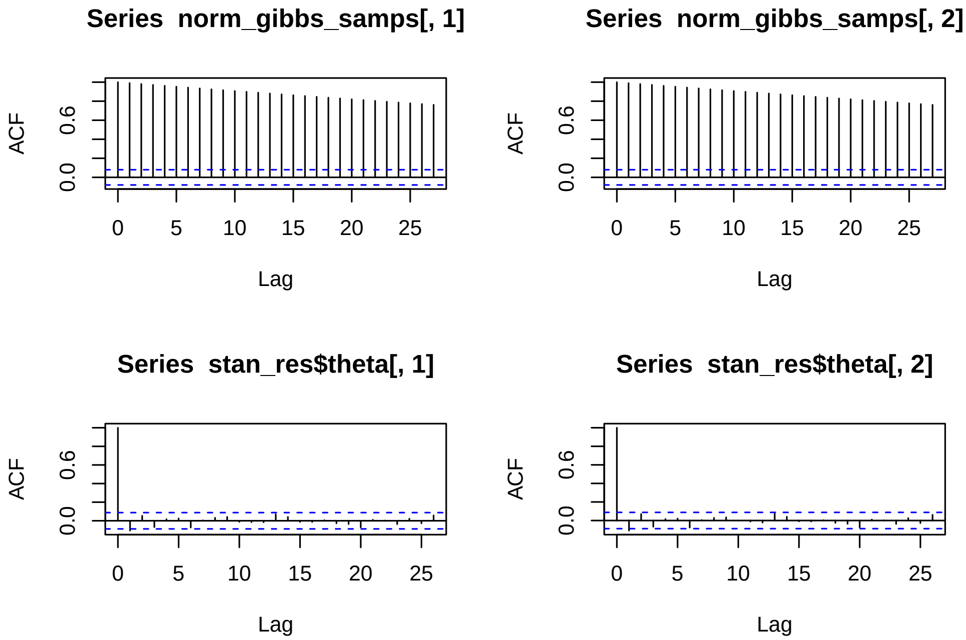
```
## Warning: Bulk Effective Samples Size (ESS) is too low, indicating posterior means and medians may be
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#bulk-ess

## Warning: Tail Effective Samples Size (ESS) is too low, indicating posterior variances and tail quant
## Running the chains for more iterations may help. See
## https://mc-stan.org/misc/warnings.html#tail-ess
```

```
data.frame(stan_res$theta) %>%
  magrittr::set_colnames(c("theta_1", "theta_2")) %>%
  dplyr::mutate(iter = 1:n()) %>%
  ggplot2::ggplot() +
  geom_density2d(data = data.frame(true_post) %>%
                        magrittr::set_colnames(c("true_1", "true_2")),
```

```
                  aes(x = true_1, y = true_2)) +
  geom_path(aes(x = theta_1, y = theta_2, colour = iter), alpha = 0.2, size = 0.5) +
  geom_point(aes(x = theta_1, y = theta_2, colour = iter), size = 0.5) +
  scale_color_distiller(palette = "Spectral", name = "Iter") +
  labs(x = expression(theta[1]), y = expression(theta[2])) +
  xlim(c(mu_post[1] - 0.5, mu_post[1] + 0.5)) +
  ylim(c(mu_post[2] - 0.5, mu_post[2] + 0.5))
```




```
par(mfrow = c(2,2))
acf(norm_gibbs_samps[,1])
acf(norm_gibbs_samps[,2])

acf(stan_res$theta[,1])
acf(stan_res$theta[,2])
```



Ex.5

The draws from Gibbs sampler are very inefficient in terms of mixing and fail to capture the true posterior density. HMC is efficient and scattered nicely across the true posterior distribution.

Ex.6

Again, high correlation value affects the conditional posterior density of θ. As the true posterior shows, θ_1 and θ_2 are very positively correlated. Even after lots of iterations, the auto-corrrelation does not diminish quickly and thus we don't see efficienct convergence.