BDA - Assignment 6

Anonymous

```
library(tidyverse)
library(aaltobda)
library(rstan)
# stan settings:
source('stan_utility.R') # diagnosis of rhats
options(mc.cores = parallel::detectCores()) #for local computer
rstan_options(auto_write = TRUE) # autosave Stan
# bay settings:
library(loo) #pred. error of MCMC log likelihood
library(gridExtra)
library(bayesplot) #plots of posterior draws (mcmc_hist etc)
library(shinystan) # model paramteres & MCMC simulations
bayesplot_theme_set() #default
SEED <- 48927 # random seed for reproducability
data("bioassay")
```

Q1

With the bioassay data from the aaltobda package, we will create a stan model, that replicate the computations in section 3.7 in BDA3. we base it on the Gaussian prior from assignment 4 and 5, where:

$$\begin{bmatrix}\alpha\\ \beta\end{bmatrix} \sim \mathcal{N}\{\mu_{\mathbf{0}}, \mathbf{\Sigma_0}\},\ \mu_{\mathbf{0}} = \begin{pmatrix}\mu_\alpha\\ \mu_\beta\end{pmatrix} = \begin{pmatrix}0\\ 10\end{pmatrix} \text{ and } \mathbf{\Sigma_0} = \begin{pmatrix}\sigma_\alpha^2 & \rho\sigma_\alpha\sigma_\beta\\ \rho\sigma_\alpha\sigma_\beta & \sigma_\beta^2\end{pmatrix} = \begin{pmatrix}4 & 10\\ 10 & 100\end{pmatrix}.$$

The created stanmodel:

```
# display the stanmodel "ex6.stan"
writeLines(readLines("ex6.stan"))
```

```
## // stan model for Bioassay data
## data {
##   int<lower=0> N;    // dose levels
##   int<lower=0> n [N];    // number of animals
##   int<lower=0> y [N];    // number of deaths
##   vector[N] x;
##   vector[2] mu_theta;
##   matrix[2, 2] sigma_theta;
## }
##
## parameters {
##   vector[2] theta;
## }
##
## model {
```

```
## //prior
##   theta ~ multi_normal(mu_theta, sigma_theta);
##   for (i in 1:N) {
## //likelihood
##     y[i] ~ binomial_logit(n[i],theta[1]+theta[2]*x[i]);
##   }
## }
```

Before calling our model, we need to summarize the data into a vector:

```
N = length(bioassay$x)
x = bioassay$x # dose
y = bioassay$y # deaths
n = bioassay$n # animals
# values
mu_alpha = 0
mu_beta = 10
sigma_alpha = 2
sigma_beta = 10
corr = 0.5
# the sigma matrix:
sigma <-  matrix(c(sigma_alpha^2, corr*sigma_alpha*sigma_beta,
                   corr*sigma_alpha*sigma_beta, sigma_beta^2 ),
                 ncol=2)
mu = c(mu_alpha, mu_beta)
# binomial data list
d_bin <- list(N = N,
              n = n,
              x = x,
              y = y,
              sigma_theta = sigma,
              mu_theta = mu)
```

We can now fit the data to our stan model:

```
fit_bioassay <- stan(file="ex6.stan", data = d_bin, seed = SEED)
```

Q2

For convergence analysis, we can use the build-in $\hat{R}$ analysis in rstan:

```
# Monitor takes an array of simulations as it argument
# probs: specifying quantiles of interest
monitor(fit_bioassay, probs = c(0.1, 0.5, 0.9))
```

```
## Inference for the input samples (4 chains: each with iter = 2000; warmup = 0):
##
##            Q5  Q50  Q95 Mean  SD  Rhat Bulk_ESS Tail_ESS
## theta[1] -0.4  0.9  2.4  0.9 0.9     1     1470     2052
## theta[2]  4.2  9.6 18.6 10.3 4.4     1     1515     1901
## lp__     -8.9 -6.8 -6.1 -7.1 1.0     1     1781     2323
##
```

```
## For each parameter, Bulk_ESS and Tail_ESS are crude measures of
## effective sample size for bulk and tail quantities respectively (an ESS > 100
## per chain is considered good), and Rhat is the potential scale reduction
## factor on rank normalized split chains (at convergence, Rhat <= 1.05).
```

```
print(fit_bioassay) #returned by Stan's sampling function
```

```
## Inference for Stan model: ex6.
## 4 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=4000.
##
##           mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
## theta[1]  0.93    0.02 0.87 -0.68  0.32  0.90  1.49  2.72  1478    1
## theta[2] 10.26    0.12 4.39  3.61  6.99  9.56 12.99 20.14  1437    1
## lp__     -7.06    0.02 0.97 -9.57 -7.45 -6.77 -6.36 -6.11  1656    1
##
## Samples were drawn using NUTS(diag_e) at Fri Apr 03 13:28:42 2020.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

Above we can find the $\hat{R}$ values to:

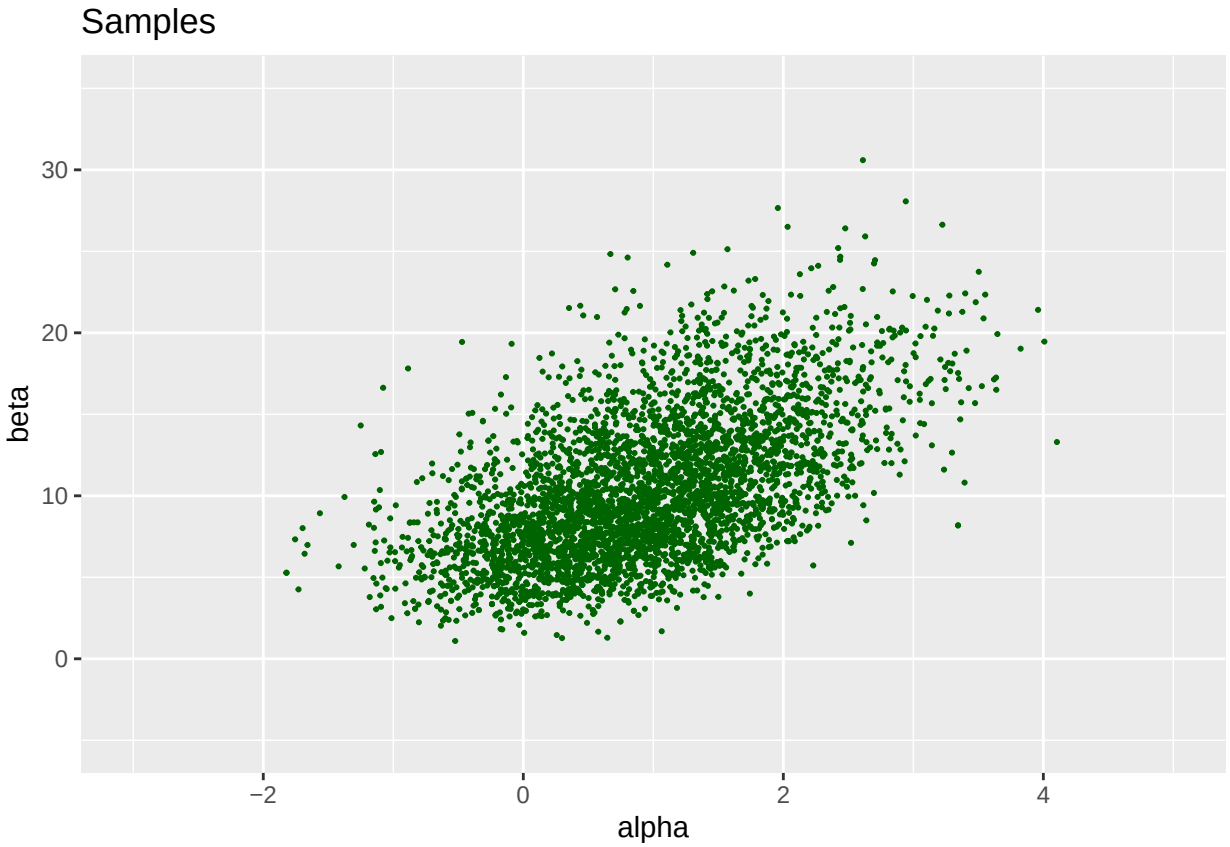
- α (=theta[1]) = 1
- β (=theta[2]) = 1.

Since the $\hat{R}$ values for α and β are both 1 we conclude that the model has converged. This means that our estimates are acceptable as they get closer and closer to the real value as the iterations proceeds.

Q3

Scatter plot of α and β with ggplot:

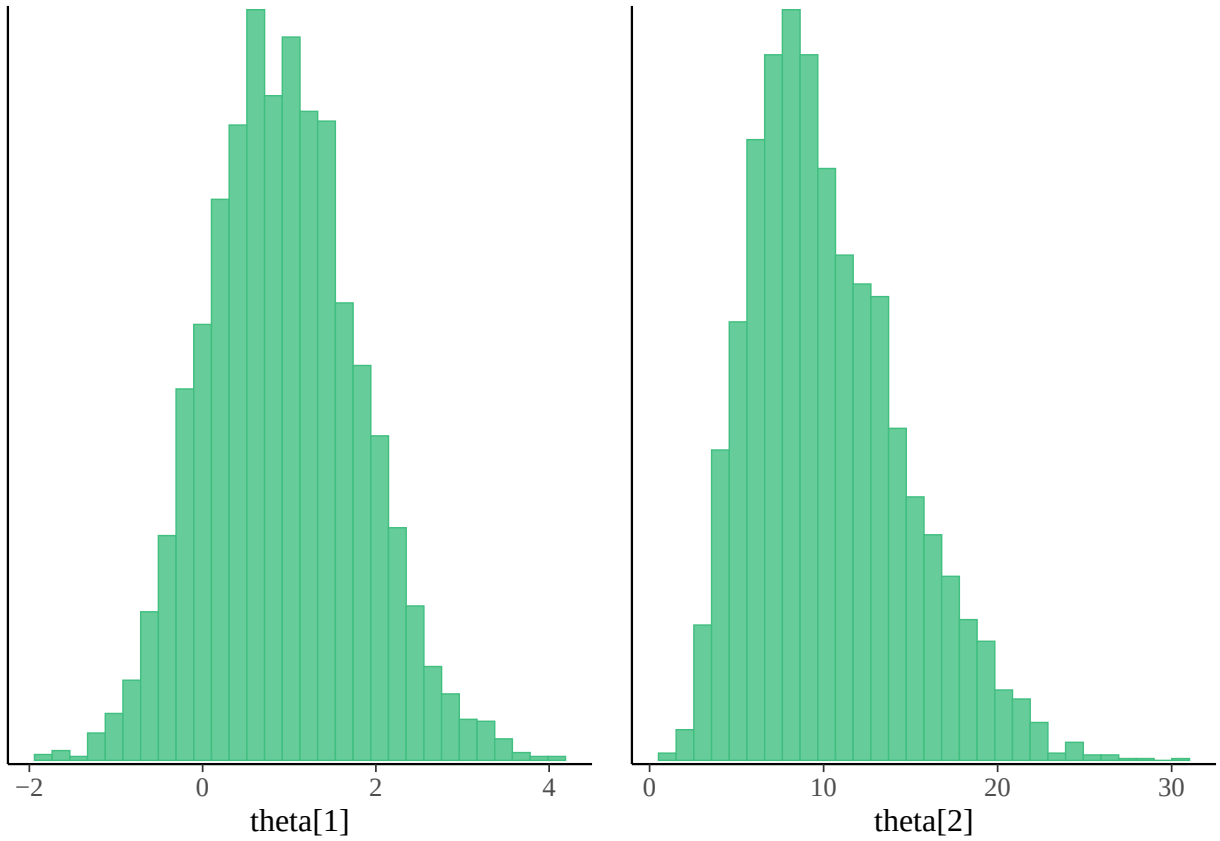
```
draws <- as.data.frame(fit_bioassay)
xl <- c(-3, 5)
yl <- c(-5, 35)
ggplot(data = data.frame(draws$`theta[1]`, draws$`theta[2]`)) +
  geom_point(aes(draws$`theta[1]`, draws$`theta[2]`), color = 'darkgreen', size = 0.4) +
  coord_cartesian(xlim = xl, ylim = yl) +
  labs(x = 'alpha', y = 'beta') +
  ggtitle("Samples")
```



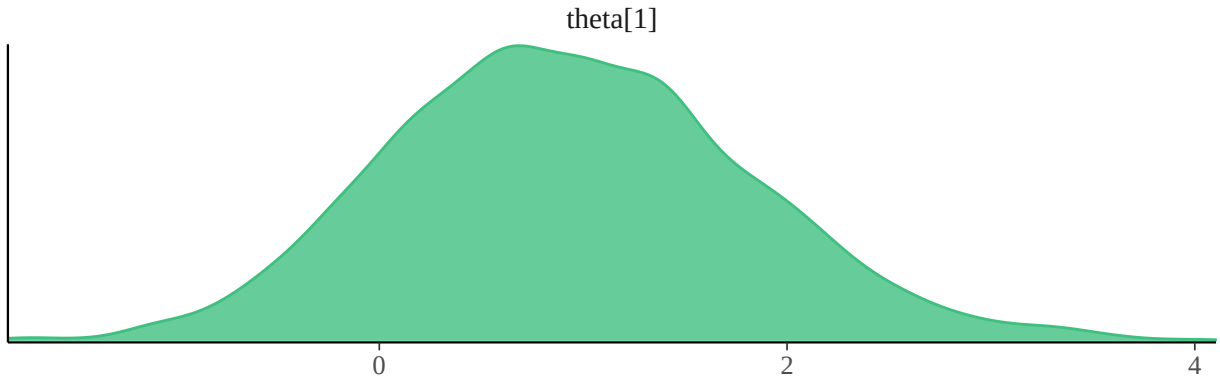
We can also use some of the mcmc plot functions to visualize α (theta[1]) and β (theta[2]):

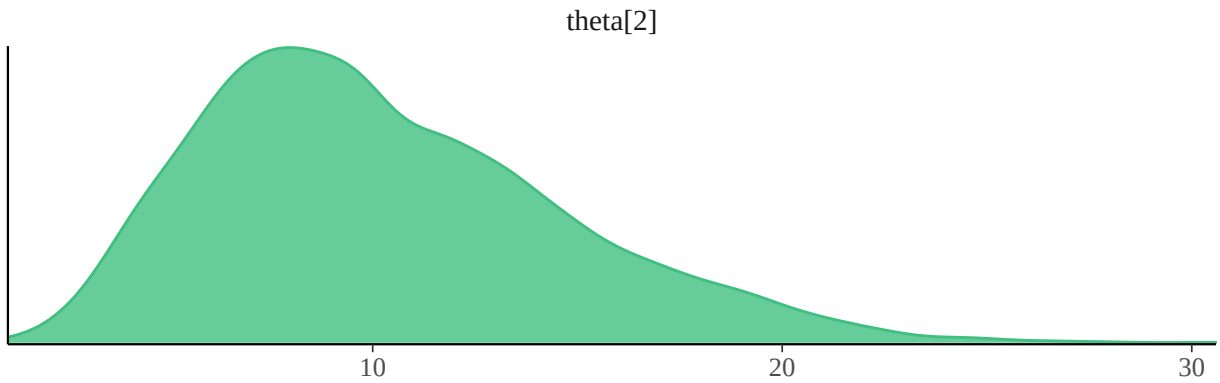
```
# histogram of alpha and beta,
# plots marginal posterior distributions combining all chains
color_scheme_set("green")
p1 <- mcmc_hist(draws, pars = 'theta[1]', binwidth = NULL)
p2 <- mcmc_hist(draws, pars = 'theta[2]', binwidth = NULL)
grid.arrange(p1, p2, ncol=2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# density
mcmc_dens(draws, pars = c("theta[1]", "theta[2]"),
          facet_args = list(nrow = 2))
```





References:

Based on code examples from:
https://github.com/avehtari/BDA_R_demos