

BDA - Assignment 6

24/10/2021

1

We start by pre-calculating the mean μ and covariance metrics Σ

```
corr = 0.6
a_std = 2
b_std = 10

mu = c(0,10)
sigma = matrix( c(a_std^2, a_std*b_std*corr, a_std*b_std*corr, b_std^2),nrow = 2)
```

Here is the stan model used for the assignment

```
data {
  int<lower=0> N;
  vector[N] x;
  int<lower=0> n[N];
  int<lower=0> y[N];
  vector[2] mu;
  matrix[2,2] sigma;
}
parameters {
  vector[2] theta;
}
model {
  theta ~ multi_normal(mu, sigma);
  for (k in 1:N) {
    y[k] ~ binomial_logit(n[k],theta[1] + theta[2]*x[k]);
  }
}
```

Let us create the data used by stan

```
data <- list(
  N = length(bioassay$x), #Number of data points
  x = bioassay$x,         #Outcome
  n = bioassay$n,         #Total draws
  y = bioassay$y,         #Successes
  mu = mu,                #Mean vector
  sigma = sigma           #Covariance matrix
)
```

```
fit1 = sampling(stanmodel,
  data = data,             # named list of data
  chains = 4,              # number of Markov chains
  warmup = 1000,          # number of warmup iterations per chain
```

```

iter = 2000,          # total number of iterations per chain
cores = 1,            # number of cores (could use one per chain)
refresh = 0           # no progress shown
)

```

We have chosen to use a chain length of 2000 with a warmup of 1000. We are simulating 4 chains.

Let us print the model

```
print(fit1)
```

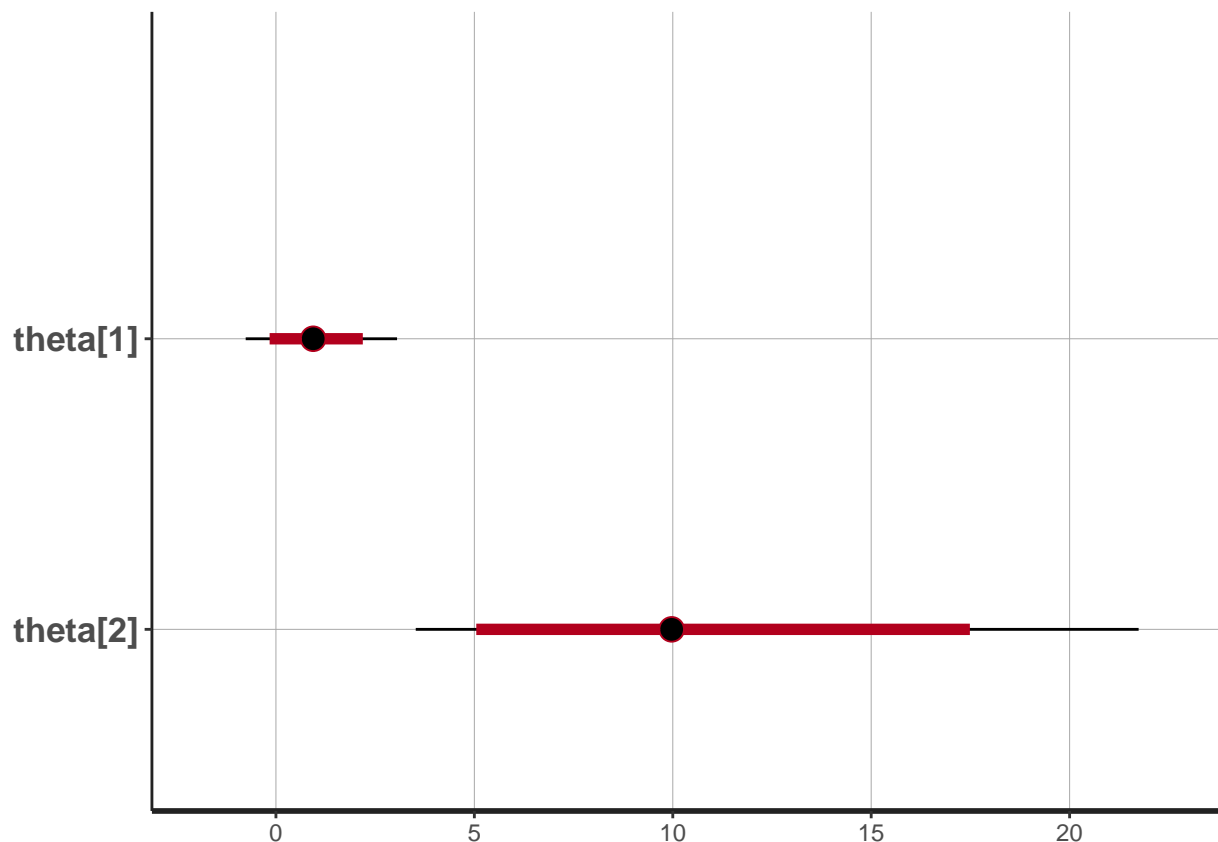
```

## Inference for Stan model: 2dc9fd89244b41be37b26fa255d63416.
## 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]  0.99    0.03 0.94  -0.76  0.35  0.94  1.58  3.05  1052   1
## theta[2] 10.74    0.15 4.81   3.52  7.11  9.97 13.72 21.74  1008   1
## lp__      -7.21    0.03 1.08 -10.17 -7.65 -6.88 -6.42 -6.14  1051   1
##
## Samples were drawn using NUTS(diag_e) at Fri Oct 22 12:37:56 2021.
## 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).

```

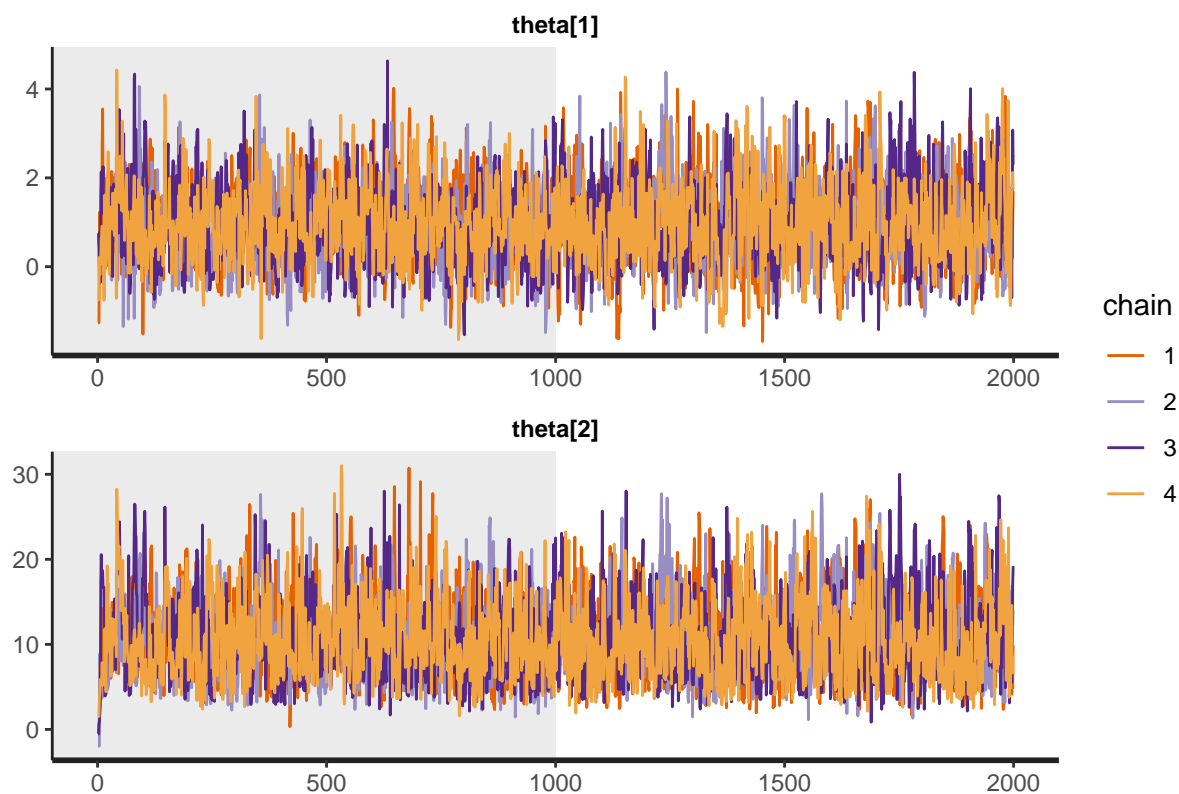
and here we can see how the parameters alpha and beta are distributed

```
plot(fit1, probs=c(.05,.5,.95))
```



We can now plot all chains to examine graphically if they converge.

```
traceplot(fit1, inc_warmup = TRUE, nrow = 2)
```



2

We use the built in function for Rhat from Rstan

```
diagnostics = monitor(fit1)
```

```
## 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] -0.5  0.9  2.6  1.0  0.9    1   1059   1437
```

```
## theta[2]  4.1 10.0 19.6 10.7  4.8    1   1081   1625
```

```
## lp__      -9.4 -6.9 -6.2 -7.2  1.1    1   1154   1680
```

```
##
```

```
## 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
```

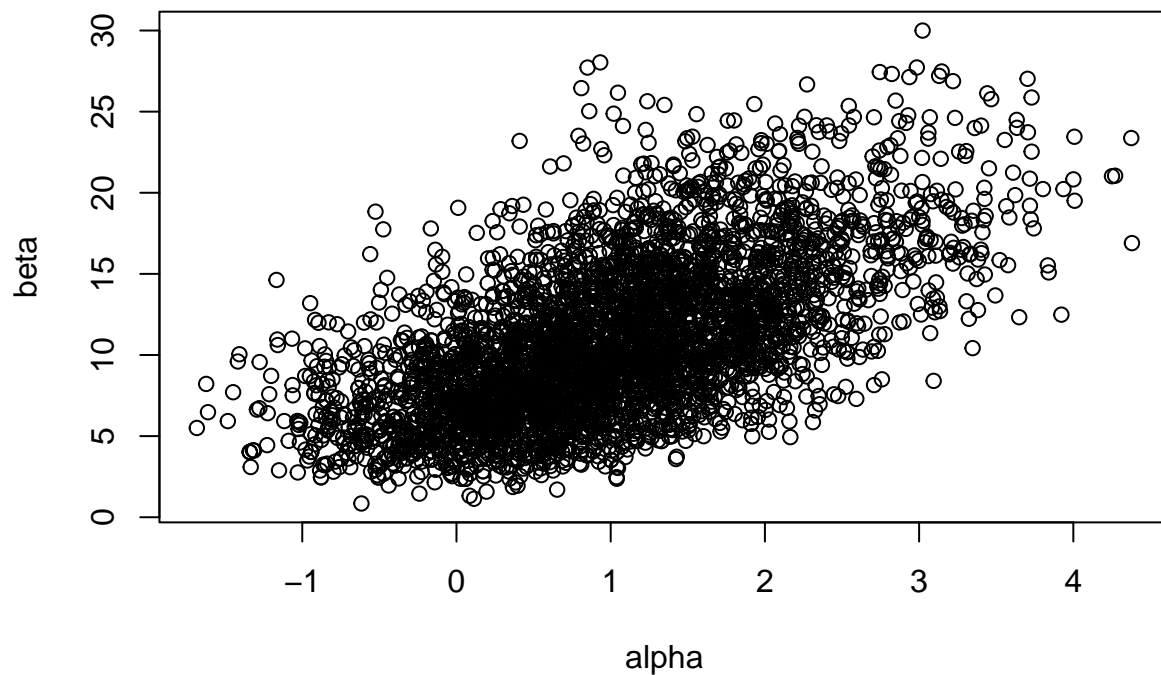
```
## factor on rank normalized split chains (at convergence, Rhat <= 1.05).
```

The \hat{R}_α is 1.00 and \hat{R}_β is 1.00. We can from this conclude that since the \hat{R} -values are $\hat{R} < 1.05$ that the chains have converged well.

3

Let us scatter plot the parameters

```
theta = extract(fit1)$theta  
alpha = theta[,1]  
beta = theta[,2]  
plot(alpha,beta)
```



We can conclude that the plot is very similar to last weeks plot and the plot from the course book.

4.

The following tech was used for the assignment:

1. OS: Windows 10
2. Language: R
3. Interface used: Rstan

I had no installation or compilation error on my laptop. I ran everything locally. I had however problems getting things to work on my desktop, the stanmodel woldn't work for some reason. I couldn't find the cause of the problem.

I found the documentation quite lacking, I found no explanation for the $y \sim \text{dist}(\text{para1}, \text{para2})$ kind of syntax and when to use it and when to use normal $y = \text{dist}(y, \text{para1}, \text{para2})$ syntax.