Exercise 3.1.4

Saurabh Steixner-Kumar
(social): **in** - **Q** - **𝕏**

Contents

1 Question

Exercise 3.1.4 For both the k = 50, n = 100 and k = 5, n = 10 cases just considered, re-run the analyses with many more samples (e.g., ten times as many) by changing the nsamples variable in Matlab, or the n.iter variable in R. This will take some time, but there is an important point to understand. What controls the width of the posterior distribution (i.e., the expression of uncertainty in the rate parameter θ)? What controls the quality of the estimate of the posterior (i.e., the smoothness of the histograms in the figures)?

2 Comments/Solution

Here we compare the two datasets (k=5, n=10 and k=50, n=100) from the previous exercise 3.1.3 with different number of iterations. To get a good idea of the difference we compare here the posteriors with 2000 and 2000 x 10 iterations side by side (in the posterior plots section below).

To answer the two raised questions: The width of the posterior distribution is controlled by the number of data points and not the number of iteration, while the smoothness of the posterior estimate is dependent on the number of iterations.

The model used to calculate the required values and the plots is scripted below. Copy/pasting the given code will generate the same result on your own machine.

3 Code

3.1 libraries

The libraries required for the script and the plots.

```
# clears workspace
rm(list=ls())
#load libraries
library(rstan)
library(ggplot2)
library(patchwork)
```

3.2   Data

The data required for this particular stan model.

```
# data   initialization
k <- 5
n <- 10
k_1 <- 50
n_1 <- 100
# to be passed on to Stan
stan_data <- list(k = k, n = n)
stan_data_1 <- list(k = k_1, n = n_1)
```

3.3   Stan code

Stan code, that can be written in R as such or in a seperate new file with stan extension.

```
write("// Stan code here in this section

// Inferring theta
data {
  int<lower=1> n;
  int<lower=0> k;
}
parameters {
  real<lower=0,upper=1> theta;
}
model {
  // Prior Distribution for theta
  theta ~ beta(1, 1);

  // Observed Counts
  k ~ binomial(n, theta);
} // ",

"3_1_4.stan")
```

3.4   code in R to run stan

Running stan through R (with the required input parameters).

```
myinits <- list(
  list(theta=.1),   # chain 1 starting value
  list(theta=.9))   # chain 2 starting value

# parameters to be monitored:
parameters <- c("theta")
```

```r
# The following command calls Stan with specific options.
# For a detailed description type "?stan".
mod_fit <- stan(file="3_1_4.stan",
                data=stan_data,
                init=myinits,  # If not specified, gives random inits
                pars=parameters,
                iter=2000,
                chains=2,
                thin=1,
                warmup=100,  # Stands for burn-in; Default = iter/2
                seed=123  # Setting seed; Default is random seed
)
mod_fita <- stan(file="3_1_4.stan",
                data=stan_data,
                init=myinits,  # If not specified, gives random inits
                pars=parameters,
                iter=2000*10,
                chains=2,
                thin=1,
                warmup=100,  # Stands for burn-in; Default = iter/2
                seed=123  # Setting seed; Default is random seed
)
mod_fit_1 <- stan(file="3_1_4.stan",
                data=stan_data_1,
                init=myinits,  # If not specified, gives random inits
                pars=parameters,
                iter=2000,
                chains=2,
                thin=1,
                warmup=100,  # Stands for burn-in; Default = iter/2
                seed=123  # Setting seed; Default is random seed
)
mod_fit_1a <- stan(file="3_1_4.stan",
                data=stan_data_1,
                init=myinits,  # If not specified, gives random inits
                pars=parameters,
                iter=2000*10,
                chains=2,
                thin=1,
                warmup=100,  # Stands for burn-in; Default = iter/2
                seed=123  # Setting seed; Default is random seed
)
```

4 Outputs

4.1 Model summary

For dataset k=5, n=10 with 2000 iterations.

```
## Inference for Stan model: 3_1_4.
## 2 chains, each with iter=2000; warmup=100; thin=1;
## post-warmup draws per chain=1900, total post-warmup draws=3800.
##
##         mean se_mean   sd   2.5%   25%   50%   75% 97.5% n_eff Rhat
```

3

```
## theta  0.49     0.00 0.14   0.23  0.40  0.49  0.59  0.76  1464      1
## lp__   -8.83    0.02 0.73 -10.88 -8.98 -8.55 -8.37 -8.32  1475      1
##
## Samples were drawn using NUTS(diag_e) at Mon Oct 26 18:48:14 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).
```

For dataset k=5, n=10 with 2000 x 10 iterations.

```
## Inference for Stan model: 3_1_4.
## 2 chains, each with iter=20000; warmup=100; thin=1;
## post-warmup draws per chain=19900, total post-warmup draws=39800.
##
##         mean se_mean   sd   2.5%   25%   50%    75% 97.5% n_eff Rhat
## theta   0.50    0.00 0.14   0.24   0.4  0.50   0.60  0.76 14845    1
## lp__   -8.83    0.01 0.72 -10.85  -9.0 -8.56  -8.37 -8.32 18846    1
##
## Samples were drawn using NUTS(diag_e) at Mon Oct 26 18:48:15 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).
```

For dataset k=50, n=100 with 2000 iterations.

```
## Inference for Stan model: 3_1_4.
## 2 chains, each with iter=2000; warmup=100; thin=1;
## post-warmup draws per chain=1900, total post-warmup draws=3800.
##
##          mean se_mean   sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
## theta    0.50    0.00 0.05   0.40   0.46   0.50   0.53   0.59  1413    1
## lp__   -71.22    0.02 0.74 -73.22 -71.38 -70.94 -70.75 -70.70  1762    1
##
## Samples were drawn using NUTS(diag_e) at Mon Oct 26 18:48:15 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).
```

For dataset k=50, n=100 with 2000 x 10 iterations.

```
## Inference for Stan model: 3_1_4.
## 2 chains, each with iter=20000; warmup=100; thin=1;
## post-warmup draws per chain=19900, total post-warmup draws=39800.
##
##          mean se_mean   sd   2.5%    25%    50%    75%  97.5% n_eff Rhat
## theta    0.5       0 0.05   0.40   0.47   0.50   0.53   0.59 15154    1
## lp__   -71.2       0 0.69 -73.17 -71.36 -70.93 -70.75 -70.70 20995    1
##
## Samples were drawn using NUTS(diag_e) at Mon Oct 26 18:48:16 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).
```

4.2 Plots

4.2.1 Plot (chains)

The initial movement of the chains are shown here (including the warmup phase). The two chains begin from the initial starting points of as defined in the input parameters of the stan model. In A.- B. (row 1) and respectively in C.- D. (row 2), only the iteration number is different and hence the start of the chains will look similar. Also to note is the same seed.

comparison of different number of iterations
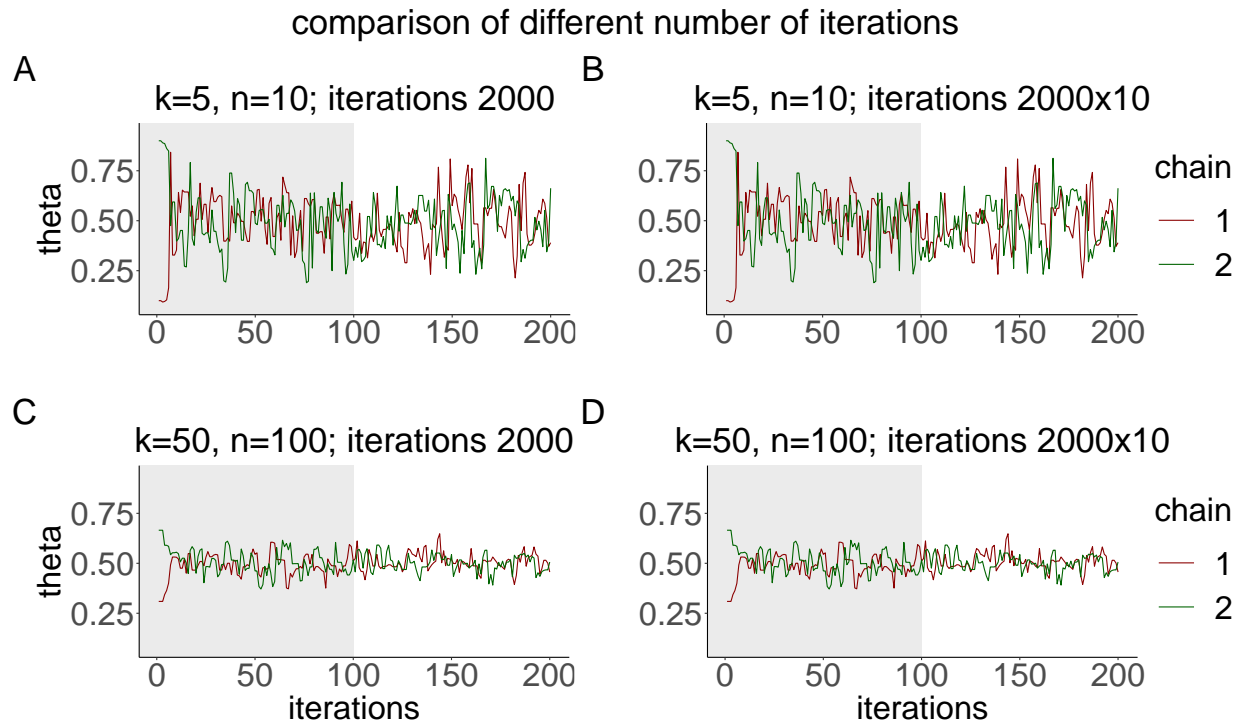
4.2.2 Plot (posterior)

The plot of the θ values per chain superimposed on each other. The plots on the right side (B. and D. are smoother than on the left due to a larger number of iterations, while the plots in the second row - c. and D. are narrower with reduced uncertainity owing to the larger number of data points.)

comparison of different number of iterations