Introduction to Bayesian Data Analysis

Contents

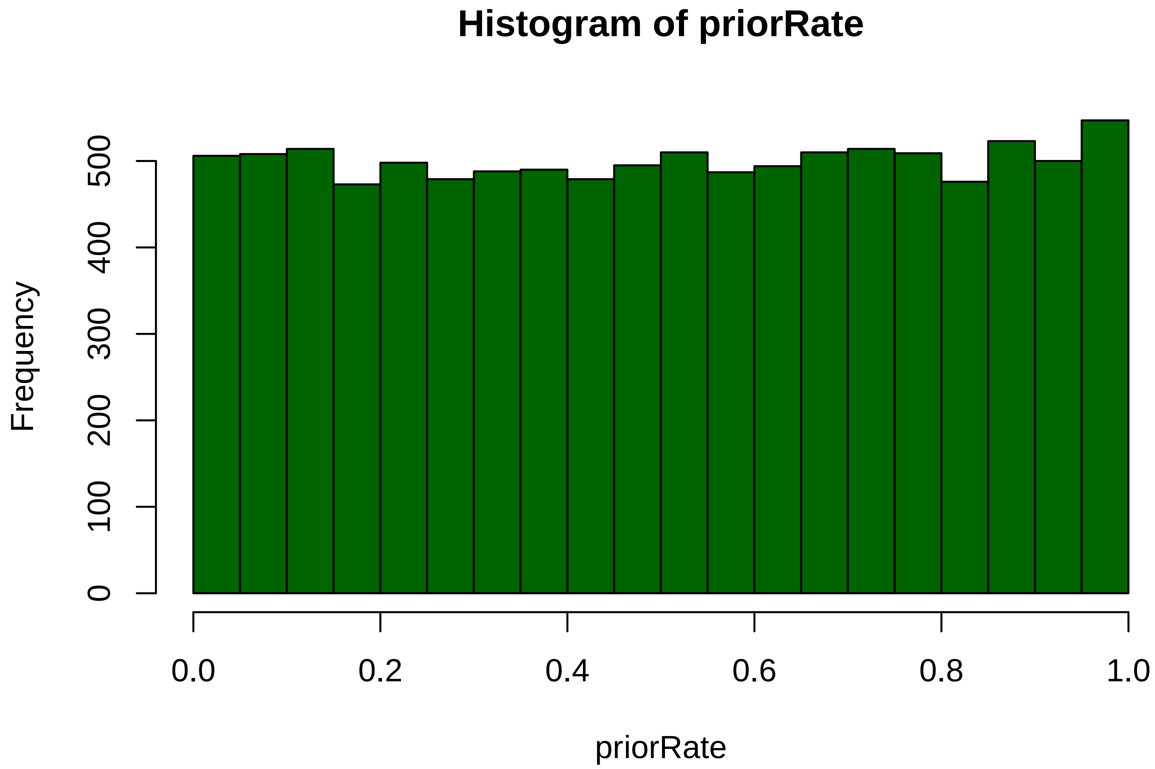
Approximate Bayesian Computation: Exercise 1

Number of random draws from the prior:

```
nDraws <- 10000
```

Defining and drawing from the prior distribution:

```
priorRate <- runif(nDraws, 0, 1)
hist(priorRate, col = "darkgreen")
```



Defining the generative model:

```
genModel <- function(rate) {
 subscribers <- rbinom(1, size = 16, prob = rate)
 subscribers
}
```

Simulating the data:

```
subscribers <- rep(NA, nDraws)
for (i in 1:nDraws) {
 subscribers[i] <- genModel(priorRate[i])
}
```

Filtering out the parameter values that did not result in the data that we actually observed:

```
postRate <- priorRate[subscribers == 6]
```

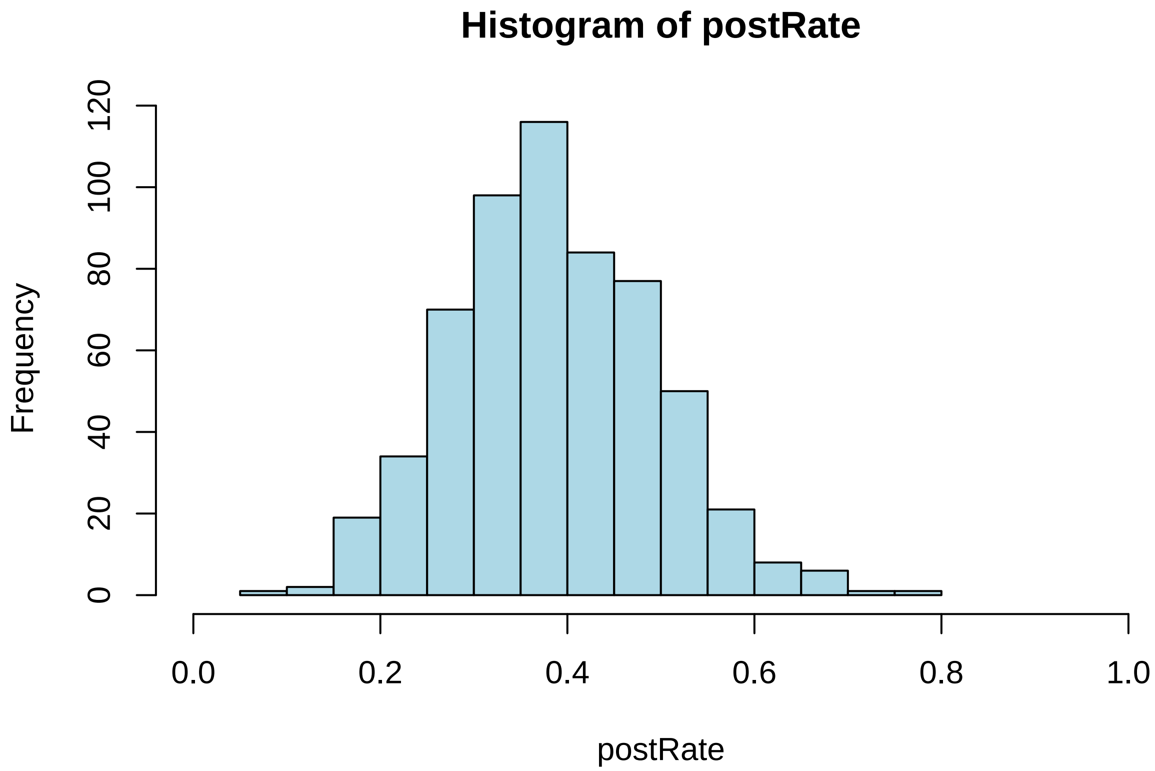
Checking that there are enough samples left:

```
length(postRate)
```

```
## [1] 588
```

Plotting and summarising the posterior:

```
hist(postRate, xlim = c(0, 1),
     col = "lightblue")
```



```
mean(postRate)
```

```
## [1] 0.3870232
```

```
quantile(postRate, c(.025, .975))
```

```
##      2.5%     97.5%
## 0.1860915 0.6030391
```

What's the *probability* that method A has a higher rate of sign-up than telemarketing?

```
sum(postRate> .2) / length(postRate)
```

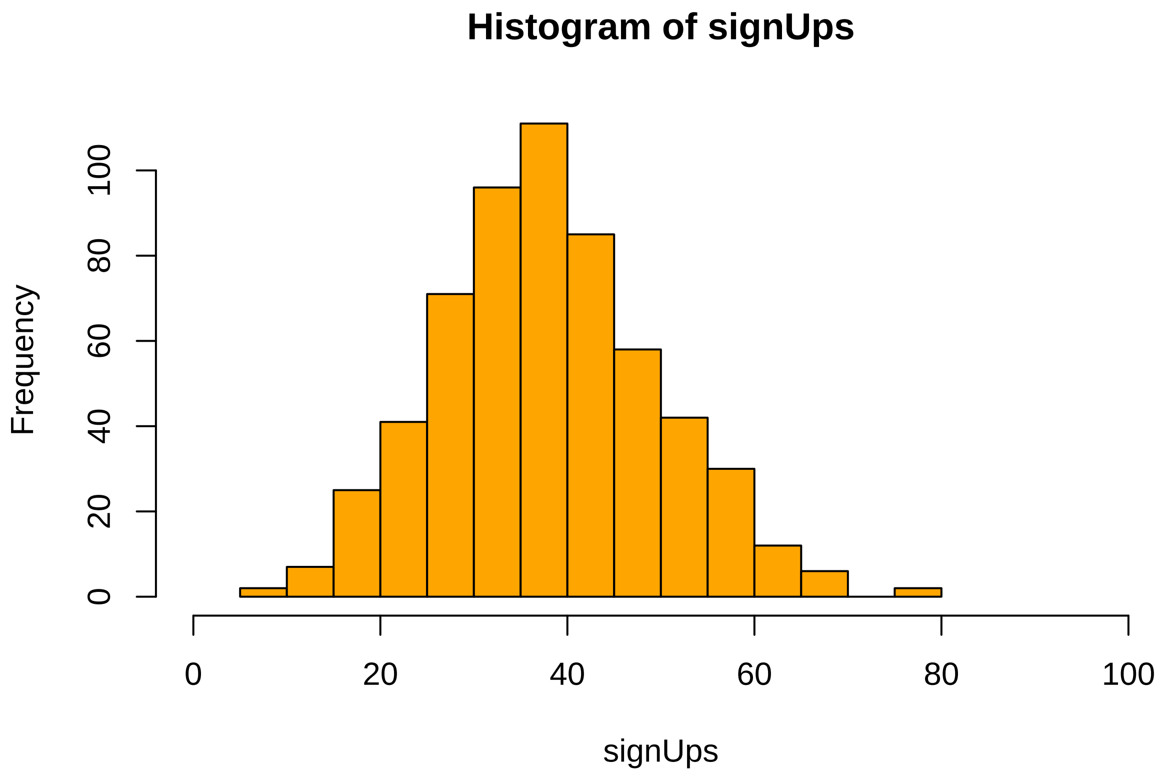
```
## [1] 0.962585
```

If method A was used on 100 people what would be number of sign-ups?

```
signUps <- rep(NA, length(postRate))
for (i in 1:length(postRate)) {
 signUps[i] <- rbinom(n = 1, size = 100, prob = postRate[i])
}
```

But since rbinom is vectorized we can simply write it like this:

```r
signUps <- rbinom(n = length(postRate), size = 100, prob = postRate)
hist(signUps, xlim = c(0, 100),
     col = "orange")
```



```r
quantile(signUps, c(.025, .975))
```

```
##   2.5%  97.5% 
## 18.675 63.000
```

So a decent guess is that is would be between 20 and 60 sign-ups

Hamiltonian Monte Carlo: Exercise 2

Setup options for this section:

```r
library(rstan)
```

```
## Loading required package: StanHeaders
```

```
## Loading required package: ggplot2
```

```
## rstan (Version 2.19.2, GitRev: 2e1f913d3ca3)
## For execution on a local, multicore CPU with excess RAM we recommend calling
## options(mc.cores = parallel::detectCores()).
## To avoid recompilation of unchanged Stan programs, we recommend calling
## rstan_options(auto_write = TRUE)
```

```
rstan_options(auto_write = TRUE)
```

Generate data the data:

```
dataList <- list(nA = 16, nB = 16, sA = 6, sB = 10)
```

You can also generate a modelString from within R. This is done by making the stan model code a **string**, directly in the R script:

```
modelString <- "
// Data:
data {
  // Number of trials
  int nA;
  int nB;
  // Number of successes
  int sA;
  int sB;
}
// Parameters:
parameters {
  real<lower=0, upper=1> rateA;
  real<lower=0, upper=1> rateB;
}
// Model:
model {
  rateA ~ uniform(0, 1); // prior for rateA
  rateB ~ uniform(0, 1); // prior for rateB
  sA ~ binomial(nA, rateA); // likelihood for sA
  sB ~ binomial(nB, rateB); // likelihood for sB
}
// Generated Quantities:
generated quantities {
real rateDiff; // rateDiff is a real value
rateDiff = rateB - rateA; // generate difference probabilities....
}
"
fishModel <- stan(model_code = modelString, data = dataList)
```

Let's inspect the model:

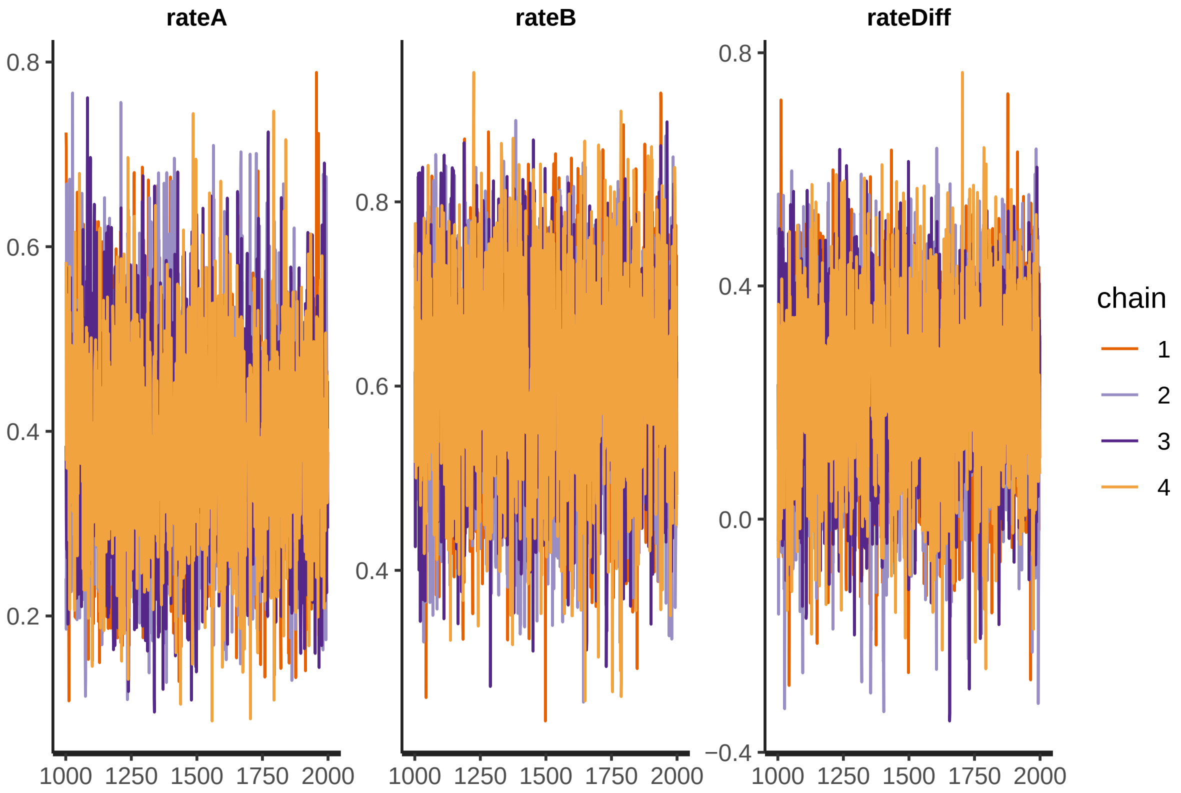
```
fishModel
```

```
## Inference for Stan model: baf0ffe29dca4ec0dcec0007feadfeae.
## 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
## rateA       0.39    0.00 0.11   0.18   0.31   0.39   0.46   0.62  3910    1
## rateB       0.61    0.00 0.11   0.39   0.53   0.62   0.69   0.81  3458    1
## rateDiff    0.22    0.00 0.16  -0.10   0.11   0.23   0.34   0.51  3756    1
## lp__      -25.08    0.02 1.03 -27.83 -25.45 -24.77 -24.36 -24.08  1976    1
##
## Samples were drawn using NUTS(diag_e) at Tue Nov 19 12:42:53 2019.
## 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).
```

We can also plot the model to visually inspect it:

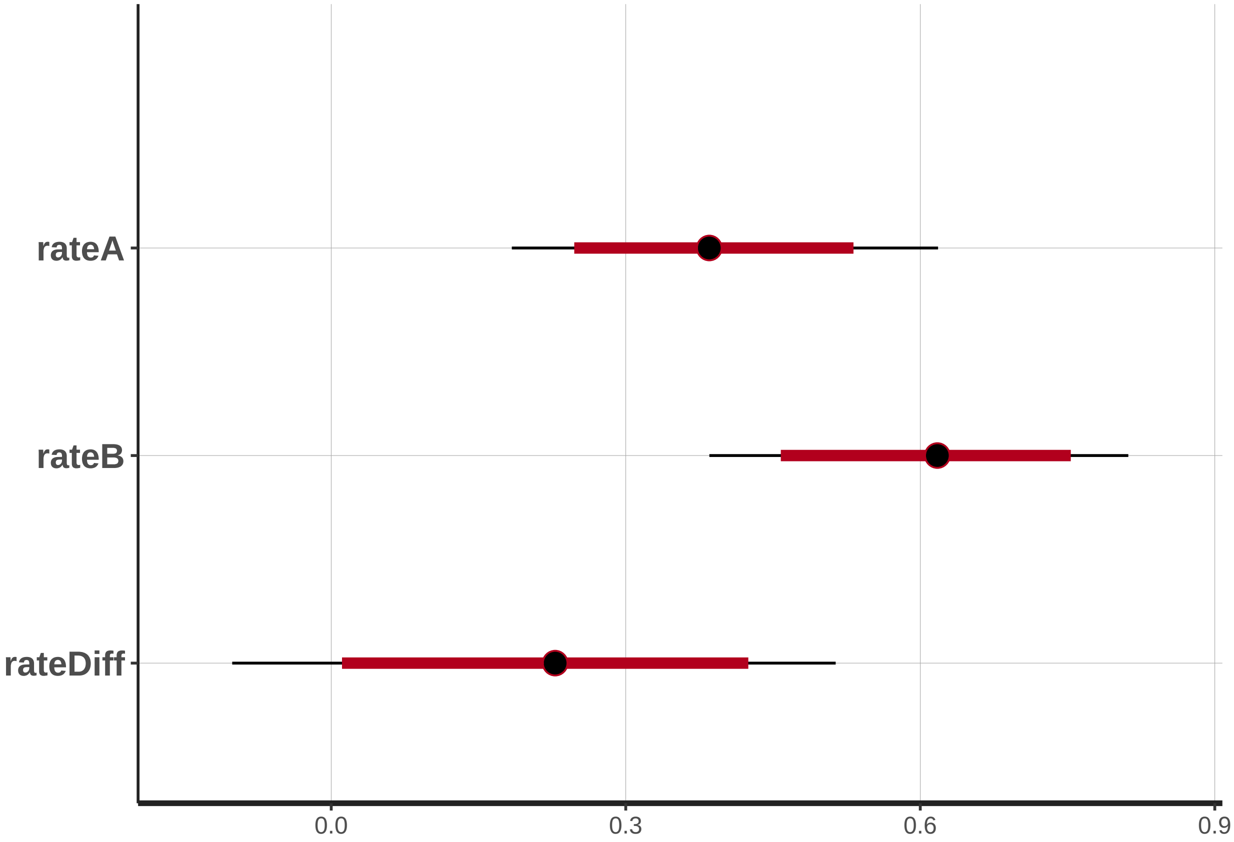
```
traceplot(fishModel)
```



```
plot(fishModel,
     col = "red")
```

```
## ci_level: 0.8 (80% intervals)
```

```
## outer_level: 0.95 (95% intervals)
```



So, which rate is likely higher? A or B?

We can export samples to a data.frame for easier handling:

```
posteriorSamples <- as.data.frame(fishModel)
sum(posteriorSamples$rateDiff > 0) / length(posteriorSamples$rateDiff)
```

```
## [1] 0.911
```

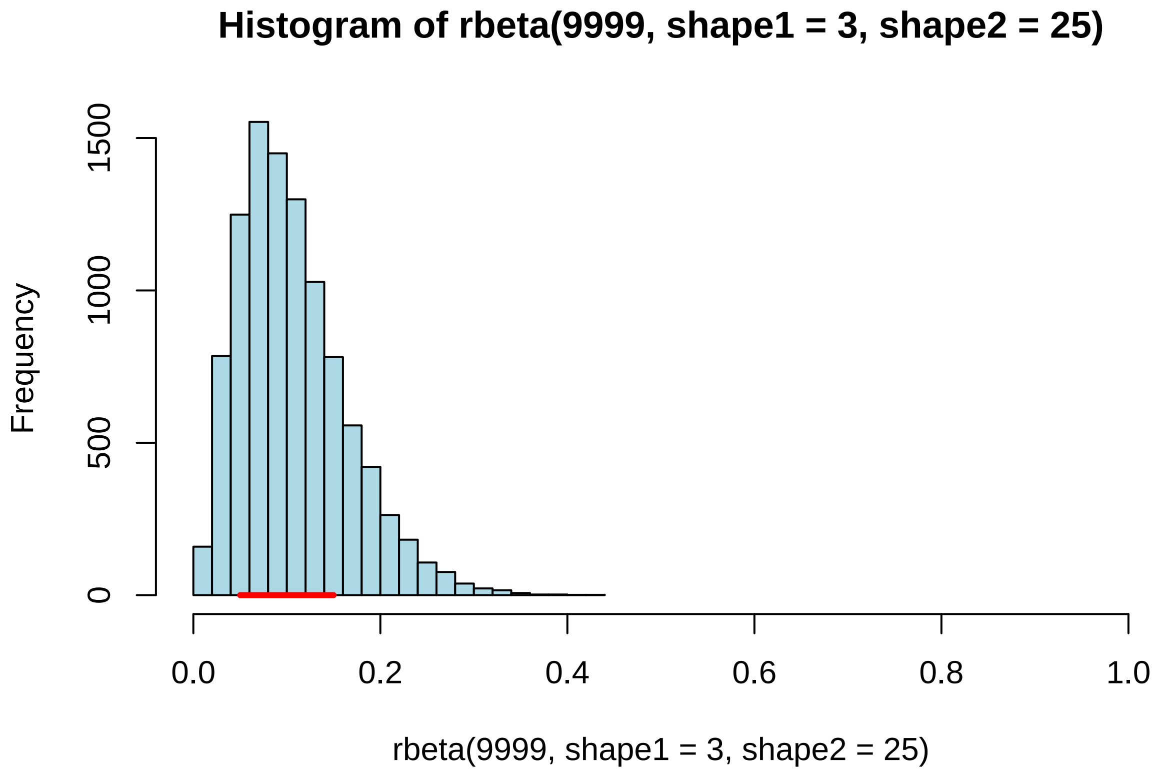
There is roughly 91% probability that rate B is higher than rate A.

Part II

The marketing department are starting to believe that it was a fluke that such a large proportion of the Danes signed up. In all other European markets the proportion that signs up for a year of salmon is around 5% to 15%, even when given a free salmon. Use this information and make the priors in your model more informative.

We will represent the background knowledge using the following beta distribution which is mostly focused on the region 0.05-0.15, i.e. **this is between 5-15%**:

```
hist(rbeta(9999, shape1 = 3, shape2 = 25),
     xlim = c(0, 1), 30,
     col = "lightblue")
lines(c(.05, .15), c(0, 0),
      col = "red",
      lwd = 3)
```



Except for the prior, the model below is exactly the same as in question I:

```
modelString <- "
// Data:
data {
  // number of trials:
  int nA;
  int nB;
  // number of successes:
  int sA;
  int sB;
}
// Parameters:
parameters {
  real <lower = 0, upper = 1> rateA;
  real <lower = 0, upper = 1> rateB;
}
// Model:
model {
  rateA ~ beta(3, 25);
  rateB ~ beta(3, 25);
  sA ~ binomial(nA, rateA);
  sB ~ binomial(nB, rateB);
}
// Generated Quantities:
generated quantities {
  real rateDiff;
  rateDiff = rateB - rateA;
}
"
```

```
fishModel <- stan(model_code = modelString, data = dataList)
```

Now we can inspect the model:

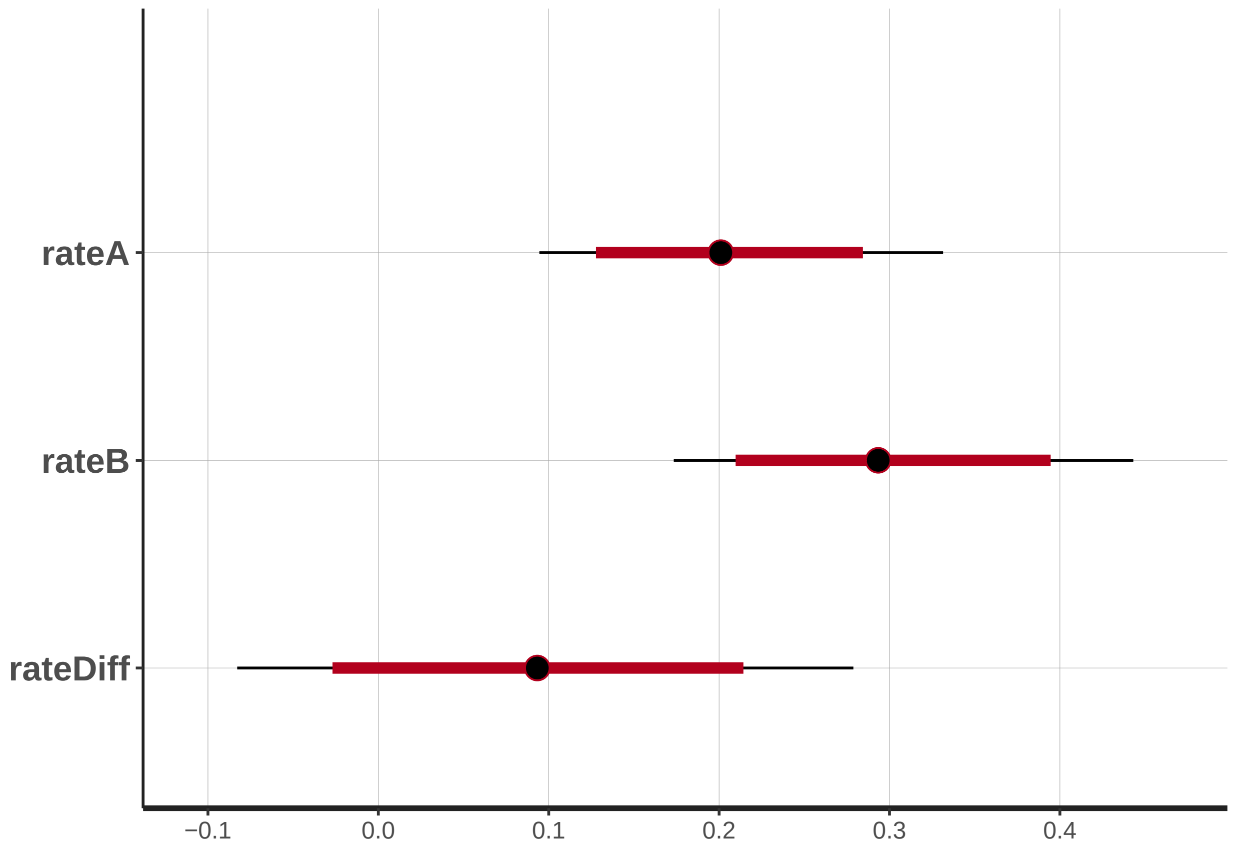
```
fishModel
```

```
## Inference for Stan model: 8a16920648e24c6404bb2ba83eab5bb4.
## 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
## rateA      0.20    0.00 0.06   0.09   0.16   0.20   0.24   0.33  3142    1
## rateB      0.30    0.00 0.07   0.17   0.25   0.29   0.35   0.44  3807    1
## rateDiff   0.09    0.00 0.09  -0.08   0.03   0.09   0.16   0.28  3620    1
## lp__     -50.07    0.03 1.04 -52.87 -50.48 -49.77 -49.31 -49.02  1581    1
##
## Samples were drawn using NUTS(diag_e) at Tue Nov 19 12:43:41 2019.
## 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).
```

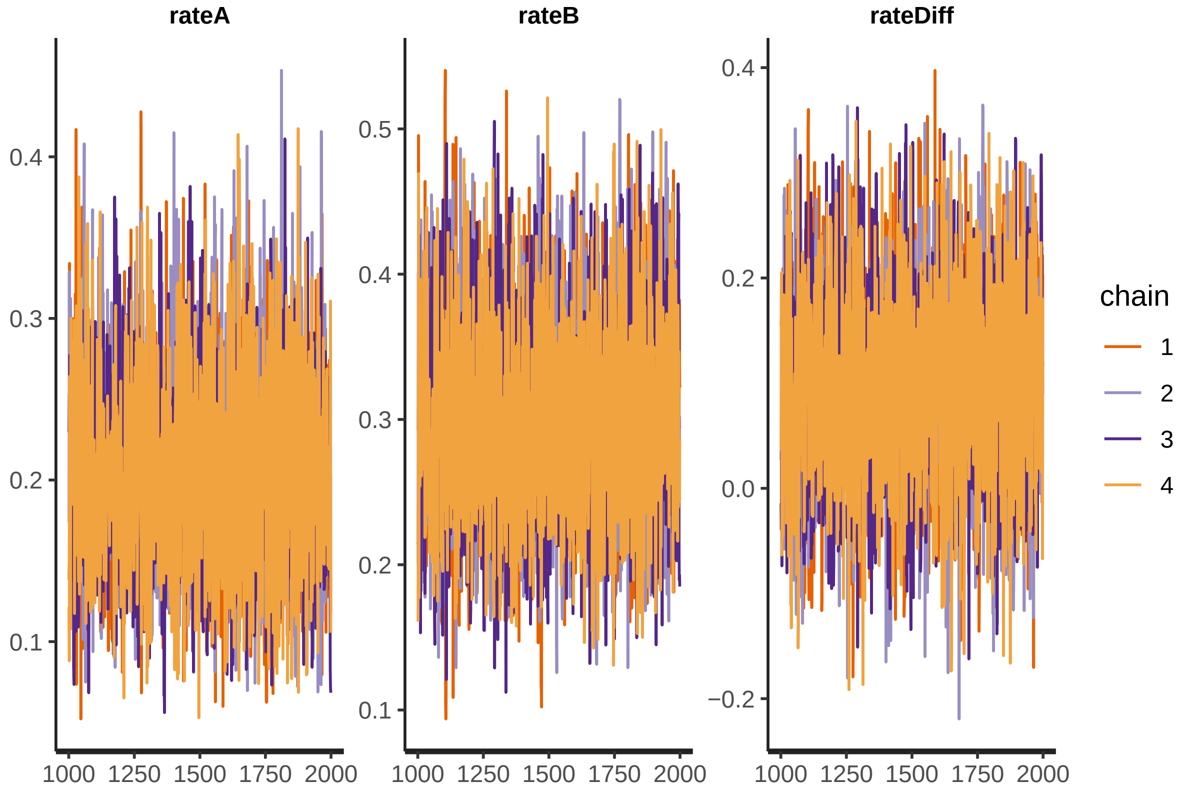
```
plot(fishModel,
     col = "grey")
```

```
## ci_level: 0.8 (80% intervals)
```

```
## outer_level: 0.95 (95% intervals)
```



```
traceplot(fishModel)
```



Here are some further summary Stats:

```
posteriorSamples <- as.data.frame(fishModel)
sum(posteriorSamples$rateDiff > 0) / length(posteriorSamples$rateDiff)
```

```
## [1] 0.84575
```

So rate B is still estimated to be higher than A with around 83% probability, but both rates are estimated to be much lower.

Part III

The economy department gives you the following information:

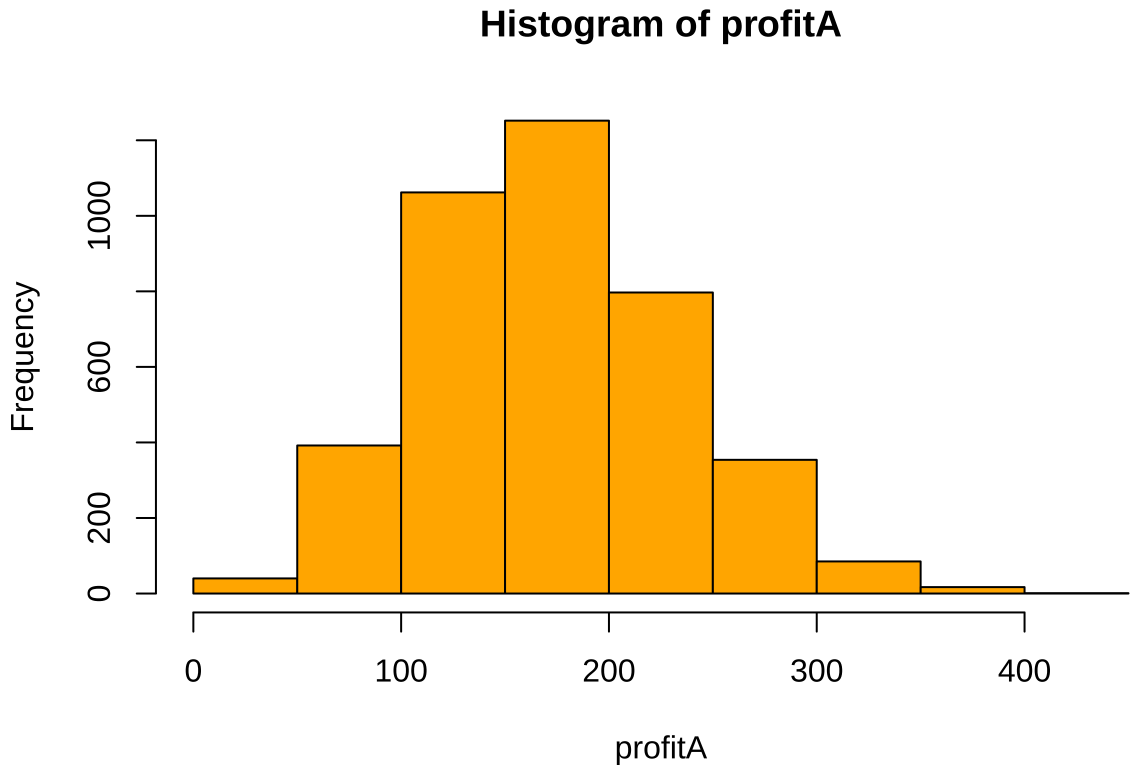
- A mail of type A costs 30 kr to send out.
- A mail of type B costs 300 kr to send out (due to the cost of the free salmon).
- A salmon subscription brings in 1000 kr in revenue.

Which method, A or B, is *most* likely to make Swedish Fish Incorporated the most money? Note: we don't have to make any changes to the model, it is enough to "post-process" the posterior distribution in posterior.

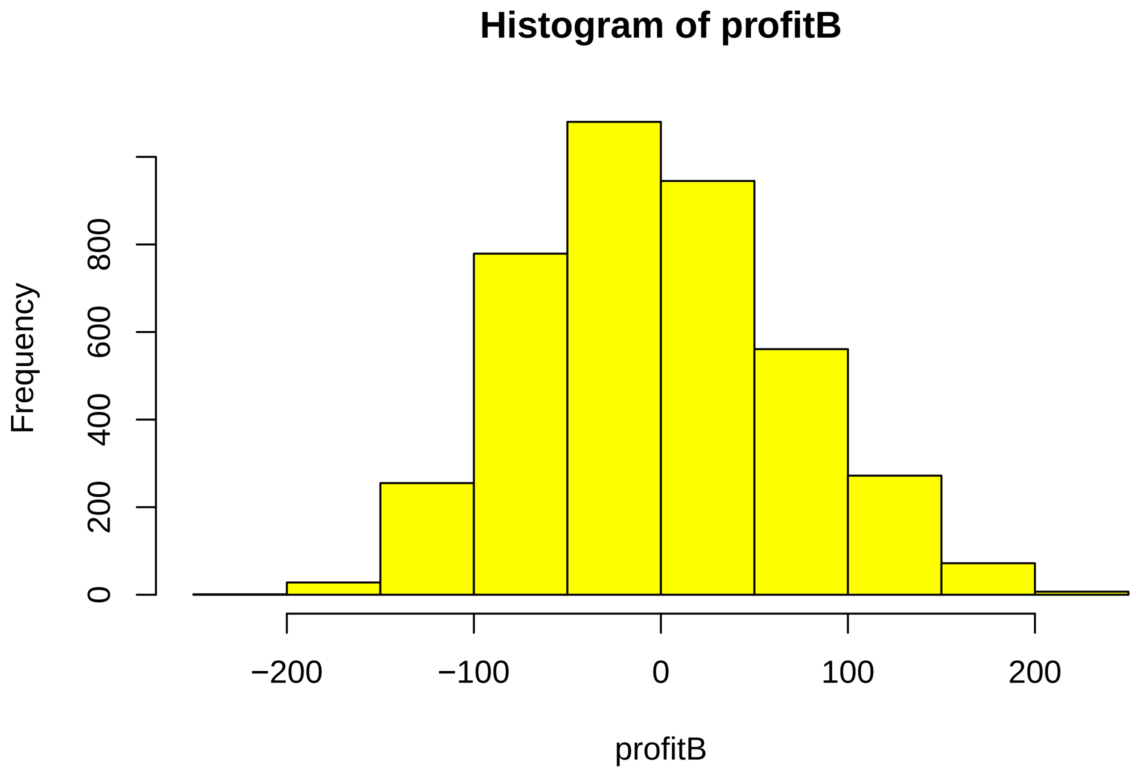
```
profitA <- -30 + posteriorSamples$rateA * 1000
profitB <- -300 + posteriorSamples$rateB * 1000

expectedProfitDiff <- mean(profitA - profitB)
```

Now we inspect the ***distributional differences*** between the two advertising options:

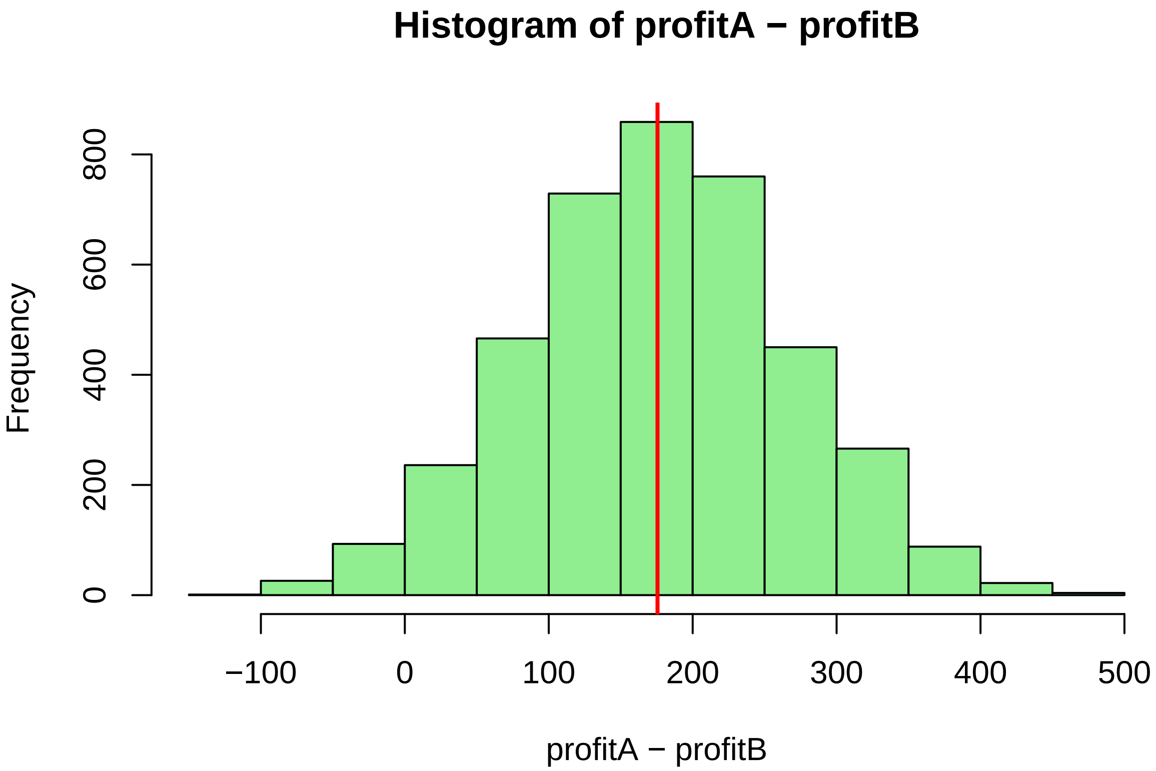
```r
hist(profitA,
     col = "orange")
```



```r
hist(profitB,
     col = "yellow")
```



```
hist(profitA - profitB,
     col = "lightgreen")
abline(v = expectedProfitDiff,
       col = "red",
       lwd = 2)
```



The expected profit when using method A is around 190 kr higher than for method B (which actually has a negative expected profit). So I guess sending free salmon to people isn't the best idea. But note that we got this result after having made the decision analysis based on the model with the informative priors. If we use the non-informative priors we get a different result, and it's up to you, the analyst, to decide which version of the model you decide to use.