

Chapter 8: Bayesian statistics

8.1 a) The prior distribution on the left of the figure assigns equal mass to every value of θ , and therefore does not incorporate relevant information about which values of θ are more likely to occur. The prior distribution in the middle reflects the belief that the probability of heads is either very close to zero, or very close to one. The prior distribution on the right reflects the information that it is most likely that the coin is a fair coin (highest probability at 0.5), but that there is some uncertainty about θ around this value.

8.1 b) Middle: Right

α : 0.5

α : 2

β : 0.5

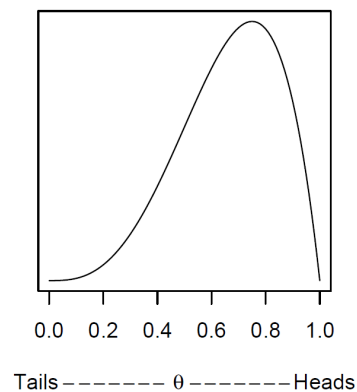
β : 2

```
alpha <- 0.5
beta <- 0.5
curve(dbeta(x, alpha, beta), xlab = expression(theta), ylab = '', yaxt = 'n')

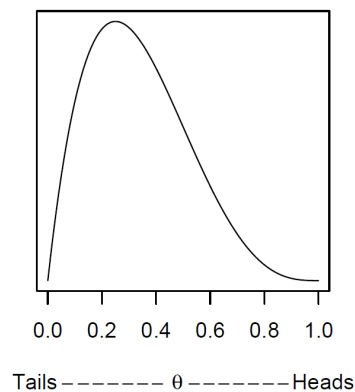
alpha <- 2
beta <- 2
curve(dbeta(x, alpha, beta), xlab = expression(theta), ylab = '', yaxt = 'n')
```

8.1 c)

Biased towards heads:



Biased towards tails:



8.2 a) The Greek letter θ represents the proportion of customers that leaves the store feeling satisfied in this case.

8.2 b) An example of prior parameters can be $\alpha = 7$ and $\beta = 5$. This combination yields a prior distribution that assigns most mass at the point $\theta = 0.6$, but expresses relatively much uncertainty about θ . However, other values of the parameters can also be correct, since the prior reflects your own beliefs. Higher values of α and β give prior distributions that are less wide, and this reflect more certain prior beliefs.

- 8.2 c) The answer depends on your own choices of α and β in assignment 8.2b. In the following answers, $\alpha = 7$ and $\beta = 5$ are used.

```
curve(dbeta(x, shape1 = 7, shape2 = 5),  
      xlab = expression(theta), ylab = '', yaxt = 'n')
```

- 8.2 d) α : $7 + 33 = 40$ β : $5 + 40 - 33 = 12$

- 8.2 e) Probability: 0.995

```
pbeta(0.6, shape1 = 40, shape2 = 12, lower.tail = FALSE) # 0.995
```

- 8.2 f) Probability: 0.875

```
diff(pbeta(c(0.7, 0.9), shape1 = 40, shape2 = 12)) # 0.875
```

- 8.2 g) The posterior distribution is fairly robust to changes in the prior distribution. In Bayesian inference, the data quickly overwhelm the prior distribution when it comes to parameter estimation. Therefore, the more data you have the more robust the posterior distribution generally is.

-
- 8.3 a) `samples <- extract(stanFit)`

- 8.3 b) The histogram approximates the analytical posterior pretty well. The more samples you draw, the better the histogram will resemble the actual posterior distribution.

```
hist(samples, breaks = 100, probability = TRUE)  
curve(dbeta(x, shape1 = 40, shape2 = 12), add = TRUE)
```

8.3 c)

```

modelCode <- '
data {
  int n;
  int k;
}
parameters {
  real<lower=0,upper=1> theta;
}
model {
  theta ~ normal(0.6, 0.1);
  k ~ binomial(n, theta);
}
'

# Note: The following line can take a while to execute
compiledModel <- stan_model(model_code = modelCode, model_name = 'model')

stanFit <- sampling(compiledModel, data = list(n = 40, k = 33),
  iter = 5000, warmup = 500, chains = 4)

```

8.3 d) Probability: 0.990

```
length(which(samples$theta > 0.6)) / length(samples$theta) # 0.990
```

8.3 e) Probability: 0.796

```
length(which(samples$theta >= 0.7 & samples$theta <= 0.9)) /
length(samples$theta) # 0.796
```

8.3 f) The answers of assignments 8.3d and 8.3e do not differ substantially from those of assignments 8.2e and 8.2f. Overall you can conclude that your posterior distribution is fairly robust to changes in the prior distribution. You can therefore be highly sure that the percentage of satisfied customers is higher than 60 percent, and reasonably sure that the percentage lies between 70 and 90 percent.

8.4 a) $H_0: \theta = 0.5$ $H_0: \theta \neq 0.5$

8.4 b) An example of a possible answer can be $\alpha = 3$ and $\beta = 3$. This prior distribution is fairly wide, and thus expresses not much confidence in your programming skills.

$$\theta \sim \text{Beta}(3, 3)$$

8.4 c) Assuming the values $\alpha = 2$ and $\beta = 2$:

$$\alpha: 3 + 22 = 25$$

$$\beta: 3 + 30 - 22 = 11$$

8.4 d)

```
alpha <- 3
beta <- 3
n <- 30
k <- 22

# Plot the posterior distribution first so no adjustment of the axes is
# needed
curve(dbeta(x, alpha + k, beta + n - k),
      bty = 'n', las = 1, lty = 1, yaxt = 'n', ylab = '', xlab = expression(theta))
curve(dbeta(x, alpha, beta), add = TRUE, lty = 2)
```

8.4 e)

```
heightPrior <- dbeta(0.5, alpha, beta)
```

8.4 f)

```
heightPosterior <- dbeta(0.5, alpha + k, beta + n - k)
```

8.4 g) The value of the Bayes factor **BF10** is 7.02. This implies that the data are 7.18 times more likely to occur under the alternative hypothesis $H_1: \theta \neq 0.5$ than under the null hypothesis $H_0: \theta = 0.5$.

8.4 h) The strength of the evidence is moderate, and so you cannot be very confident that your algorithm performs better than change. You will need to collect more data if you want a more decisive Bayes factor.

8.5 a) $M_1: \theta \leq 0.75$ $M_2: \theta \geq 0.75$

8.5 b)

```
model2code <- '
data {
  int n;
  int k;
}
parameters {
  real<lower=0.75,upper=1> theta;
}
model {
  theta ~ beta(1, 1)T[0.75, 1];
  k ~ binomial(n, theta);
}
'
```

Note: The following line can take a while to execute
`model2 <- stan_model(model_code = model2code, model_name = 'model2')`

8.5 c) The prior distribution for θ in M_1 is a $Beta(1,1)$ prior distribution truncated to the $[0, 0.75]$ interval (`beta(1, 1)T[0, 0.75]`). The prior distribution for θ in M_2 is a $Beta(1,1)$ prior distribution truncated to the $[0.75, 1]$ interval (`beta(1, 1)T[0.75, 1]`).

8.5 d)

```
stanFitM1 <- sampling(model1, data = list(n = 156, k = 123),
  iter = 5000, warmup = 500, chains = 4)

stanFitM2 <- sampling(model2, data = list(n = 156, k = 123),
  iter = 5000, warmup = 500, chains = 4)
```

8.5 e)

```
mLike1 <- bridge_sampler(stanFitM1)
mLike2 <- bridge_sampler(stanFitM2)
```

8.5 f)

```
BF12 <- bf(mLike2, mLike1)
```

8.5 g) The value of `BF12` is around 17.77. This indicates that the data is 17.77 times more likely to occur under model $M_2: \theta \geq 0.75$ than under model $M_2: \theta \leq 0.75$.

8.5 h) The evidence in favor of M_2 is moderate, and so additional testing might be required to come to a reasonable conclusion.

8.6 a) `# Be sure to set your working directory when providing a relative path`
`insurance <- read.csv('insurance.csv')`

8.6 b) $\text{charges} = \beta_0 + \beta_1 \times \text{age} + \beta_2 \times \text{bmi} + \beta_3 \times \text{neighbors}$

8.6 c) β_0 : -6670.25 β_1 : 333.92 β_2 : 241.56 β_3 : 45.84

```
regressionFit <- sampling(object = regressionModel,
                          data = list(N = nrow(insurance),
                                       x1 = insurance$bmi,
                                       x2 = insurance$age,
                                       x3 = insurance$neighbors,
                                       y = insurance$charges),
                          iter = 5000, warmup = 500, chains = 4)

regressionSamples <- extract(regressionFit)
summary(regressionSamples)
```

8.6 d) YES

```
summary(lm(charges ~ 1 + age + bmi + neighbors, data = insurance))
```

8.6 e)

```
layout(matrix(1:4, nrow = 2))

plot(density(regressionSamples$beta1), xlab = '', ylab = '',
     yaxt = 'n', bty = 'n', main = expression(beta[1]))
plot(density(regressionSamples$beta0), xlab = '', ylab = '',
     yaxt = 'n', bty = 'n', main = expression(beta[0]))
plot(density(regressionSamples$beta2), xlab = '', ylab = '',
     yaxt = 'n', bty = 'n', main = expression(beta[2]))
plot(density(regressionSamples$beta3), xlab = '', ylab = '',
     yaxt = 'n', bty = 'n', main = expression(beta[3]))

layout(1)
```

8.6 f) Probability: 0.966

```
length(which(regressionSamples$beta2 > 200)) / length(regressionSamples$beta2)
```

8.6 g) Probability: 0.052

```
length(which(regressionSamples$beta1 < 250)) / length(regressionSamples$beta1)
```

8.6 h) Lower bound: -79.98

Upper bound: 169.74

```
quantile(regressionSamples$beta3, probs = c(0.10, 0.90))
```

8.6 i)

```
regressionmodelcode2 <- '
data {
  int<lower=0> N;
  vector[N] x1;
  vector[N] x2;
  vector[N] y;
}
parameters {
  real beta0;
  real beta1;
  real beta2;
  real<lower=0> sigma;
}
model {
  y ~ normal(beta0 + beta1 * x1 + beta2 * x2, sigma);
}
'
```

Note: The following line can take a while to execute

```
regressionModel2 <- stan_model(model_code = regressionmodelcode2,
                                model_name = 'Regression2')
```

8.6 j)

```
regressionFit2 <- sampling(object = regressionModel2,
                           data = list(N = nrow(insurance),
                                         x1 = insurance$bmi,
                                         x2 = insurance$age,
                                         y = insurance$charges),
                           iter = 5000, warmup = 500, chains = 4)

mLike1 <- bridge_sampler(regressionFit)
mLike2 <- bridge_sampler(regressionFit2)

BF12 <- bf(mLike1, mLike2)
```

- 8.6 k) The value of **BF12** is 258.65. This implies that the data are 258.65 times as likely to be observed under the model where β_3 is not included in the regression equation under the model where β_3 is included. The evidence against the number of **neighbors** adding to the prediction accuracy is extreme.