MA 578 HW4 Solutions

1 (BDA 2.11(a) and 4.1)

(2.11a)

We have $y_i \,|\, \theta \overset{\text{iid}}{\sim} \mathsf{Cauchy}(\theta, 1)$, that is, $\mathbb{P}(y_i \,|\, \theta) \propto (1 + (y_i - \theta)^2)^{-1}$, and $\theta \sim U(-4, 4)$. For the approximate (normalized) posterior, we can just proceed as usual with grid integration:

```
min_theta <- -4
max_theta <- 4
y <- c(-2, -1, 0, 1.5, 2.5)

lprior <- function (theta) 0 # uniform
lhood <- function (theta) -sum(log(1 + (y - theta) ^ 2))

log_normalize <- function (x) {
  x <- x - max(x) # avoid underflows
  exp(x - log(sum(exp(x))))
}
t <- seq(min_theta, max_theta, length = 100)
ptheta <- log_normalize(sapply(t, lprior) + sapply(t, lhood))
```

(4.1)

Since $l(\theta) = \log \mathbb{P}(\theta \,|\, y) = -\sum_i \log(1 + (y_i - \theta)^2)$, we can differentiate to find

$$l'(\theta) = 2 \sum_i \frac{y_i - \theta}{1 + (y_i - \theta)^2} \quad \text{and} \quad l''(\theta) = -2 \sum_i \frac{1 - (y_i - \theta)^2}{[1 + (y_i - \theta)^2]^2}.$$

To find the mode θ^*, we can iteratively solve $l'(\theta) = 0$ by fixing the denominator at the current iteration $\theta^{(t)}$,

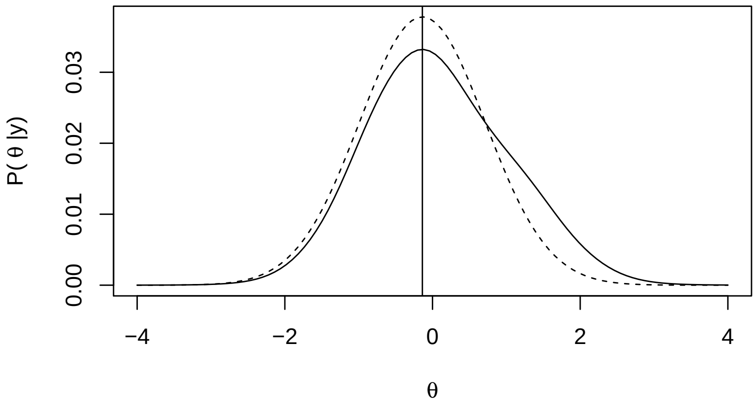
$$2 \sum_i \frac{y_i - \theta^{(t+1)}}{1 + (y_i - \theta^{(t)})^2} = 0 \quad \Rightarrow \quad \theta^{(t+1)} = \frac{\sum_i y_i / [1 + (y_i - \theta^{(t)})^2]}{\sum_i 1 / [1 + (y_i - \theta^{(t)})^2]}.$$

```
theta_m <- mean(y)
eps <- 1e-4 # tolerance
repeat {
  theta_c <- sum(y / (1 + (y - theta_m) ^ 2)) /
               (sum(1 / (1 + (y - theta_m) ^ 2)))
  if (abs(theta_c - theta_m) / abs(theta_m) < eps) break # converged?
  theta_m <- theta_c
}
```

After convergence, we find $\theta^* = -0.138$. The Laplace approximation compares well with the numerical approximation:

```
I_m <- 1 / (2 * sum((1 - (y - theta_m) ^ 2) / (1 + (y - theta_m) ^ 2) ^ 2))
plaplace <- log_normalize(dnorm(t, theta_m, sqrt(I_m), log = TRUE))
plot(t, ptheta, type = "n", xlab = expression(theta),
```

```
     ylab = expression("P("~theta~"|y)"),
     ylim = c(0, max(ptheta, plaplace)))
lines(t, ptheta); lines(t, plaplace, lty = 2); abline(v = theta_m)
```



2 (BDA 5.13)

(a)

Here $y_j \mid \theta_j \overset{\text{ind}}{\sim} \mathsf{Binom}(n_j, \theta_j)$ and $\theta_j \mid \alpha, \beta \overset{\text{iid}}{\sim} \mathsf{Beta}(\alpha, \beta)$ for $j = 1, \ldots, 10$. We adopt the same "weakly informative" prior from the rat tumor example, $\mathbb{P}(\alpha, \beta) \propto (\alpha + \beta)^{-5/2}$. Thus,

$$\mathbb{P}(\theta, \alpha, \beta \mid y) \propto \prod_j \theta_j^{y_j}(1-\theta_j)^{n_j - y_j} \prod_j \theta_j^{\alpha-1}(1-\theta_j)^{\beta-1}(\alpha+\beta)^{-5/2}.$$

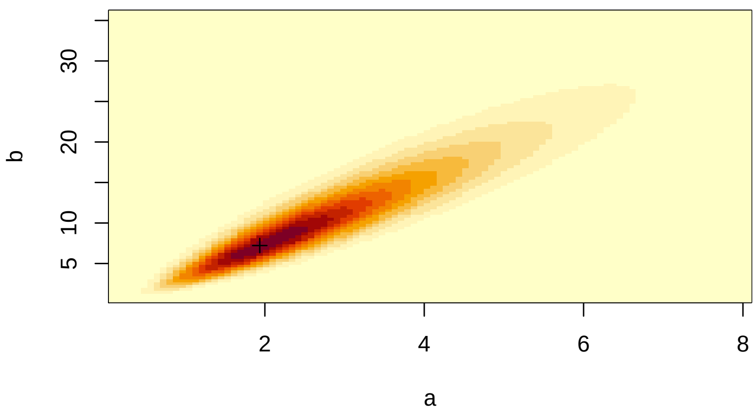
Note that $\theta_j \mid \alpha, \beta, y \overset{\text{ind}}{\sim} \mathsf{Beta}(\alpha + y_j, \beta + n_j - y_j)$.

(b)

Let us proceed as in the rat tumor example:

```
bicycle <- read.csv("data/bicycle.csv", comment = "#")
bicycle <- bicycle[bicycle$streettype == "residential",]

y <- with(bicycle, bicycles[streettype == "residential" & bikeroute == "yes"])
n <- y + with(bicycle, other[streettype == "residential" & bikeroute == "yes"])

lprior <- function (a, b) -2.5 * log(a + b) # non-informative
lhood_ab <- function (a, b)
  sum(lgamma(a + y) - lgamma(a) + lgamma(b + n - y) - lgamma(b) -
      (lgamma(a + b + n) - lgamma(a + b)))

# find approximate MLE: with lgamma(a + y) - lgamma(a) ~= y * log(a), define
r <- sum(y) / sum(n)
# and then am ~= r * ks and bm ~= (1 - r) * ks
# 'ks' can be found numerically:
m <- 100
k <- seq(20, 25, length = m)
lk <- sapply(k, function (x) lhood_ab(r * x, (1 - r) * x))
```

```r
ks <- k[which.max(lk)]
# thus, we can define rough grid boundaries:
am <- r * ks; bm <- (1 - r) * ks
a <- seq(0, 2 * am, length=m + 1)[-1]
b <- seq(0, 2 * bm, length=m + 1)[-1]

lab <- matrix(nrow = m, ncol = m)
for (ia in 1:m)
  for (ib in 1:m)
    lab[ia, ib] <- lprior(a[ia], b[ib]) + lhood_ab(a[ia], b[ib])
pab <- log_normalize(lab)
# posterior mode
im <- which(pab == max(pab), arr = TRUE)
as <- a[im[1]]; bs <- b[im[2]]
image(a, b, pab); points(as, bs, pch = 3)
```



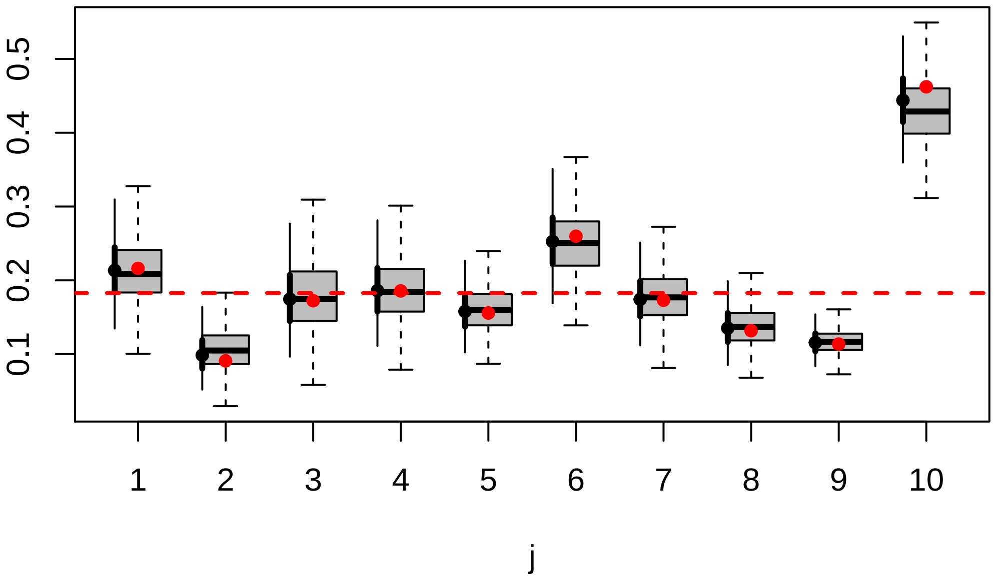

```r
# sampling
sample_ab <- function (ns) {
  is <- arrayInd(sample.int(m ^ 2, ns, replace = TRUE, prob = pab), dim(pab))
  cbind(a[is[,1]], b[is[,2]])
}

ns <- 1000
ab_s <- sample_ab(ns) # alpha, beta | y
# conditional posteriors: theta_j | alpha, beta, y
theta_s <- matrix(nrow = ns, ncol = length(y))
for (j in 1:length(y))
  theta_s[,j] <- rbeta(ns, ab_s[,1] + y[j], ab_s[,2] + n[j] - y[j])
```

(c)

The θ posteriors are in good agreement with the data, as the figure below shows. Red points are data proportions, dark reference lines are conditional posterior summaries (dot is median, thicker line marks quartiles, finer line marks 95% conditional credible intervals), and dashed red line is pooled data proportion.

```r
at <- seq(1, length = length(y), by = 1.5)
boxplot(theta_s, outline = FALSE, col = "gray", at = at, xlab = "j")
points(at, y / n, pch = 19, cex = .8, col = "red") # data proportions

# for comparison, let's plot conditional posterior summaries
```

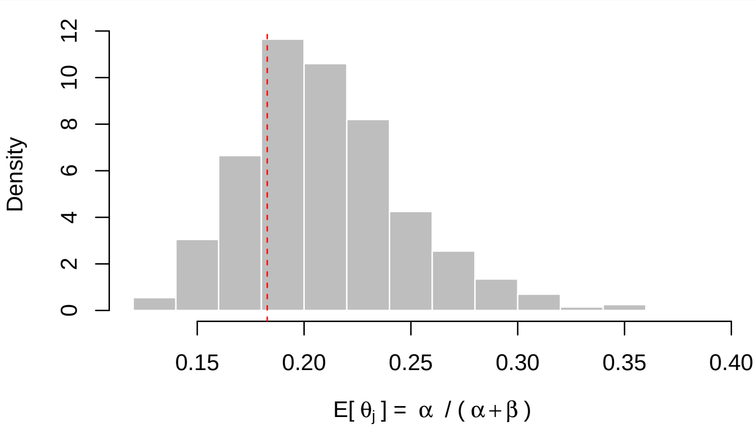
```r
delta <- .4
points(at - delta, pch = 19, cex = .8,
       qbeta(.5, as + y, bs + n - y)) # posterior medians
for (j in 1:length(y)) {
  lines(c(at[j] - delta, at[j] - delta),
        qbeta(c(.25, .75), as + y[j], bs + n[j] - y[j]), lwd = 3)
  lines(c(at[j] - delta, at[j] - delta),
        qbeta(c(.025, .975), as + y[j], bs + n[j] - y[j]))
}
abline(h = sum(y) / sum(n), col = "red", lwd = 2, lty = 2)
```



(d)

The average underlying proportion is $\mathbb{E}[\theta_j] = \alpha/(\alpha+\beta)$.

```r
etheta_s <- ab_s[,1] / rowSums(ab_s)
hist(etheta_s, col = "gray", border = "white", freq = FALSE, main = "",
     xlab = expression("E["~theta[j]~"] = "~alpha~" / ("~alpha+beta~")"))
abline(v = sum(y) / sum(n), col = "red", lty = 2)
```



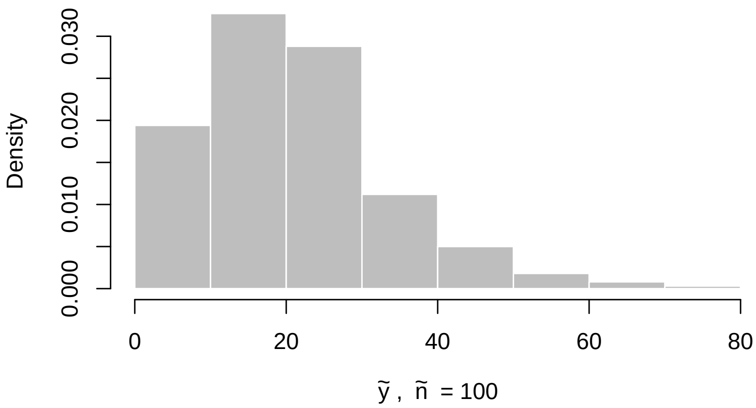
```r
round(quantile(etheta_s, c(.025, .975)), 3) # 95% credible interval
```

```
 2.5% 97.5% 
0.147 0.297 
```

(e)

A straightforward predictive sample yields a large interval, so it might not be very informative:

```r
ntilde <- 100
ytilde_s <- rbinom(ns, ntilde, rbeta(ns, ab_s[,1], ab_s[,2]))
hist(ytilde_s, col = "gray", border = "white", freq = FALSE, main = "",
     xlab = expression(tilde(y)~", "~tilde(n)~" = 100"))
```




```r
round(quantile(ytilde_s, c(.025, .975))) # 95% predictive interval
```

```
 2.5% 97.5% 
    2    52 
```

(f)

The last block ($j = 10$) has a much larger proportion of bikes, which leads us to believe that a multimodal (at least bimodal, say "calm" and "busy" blocks) prior for θ_j would be a more reasonable choice.