Applied Statistics (Chapter 4)

Yasuhiro Omori

A1A2 term, 2019-2020

University of Tokyo

Chapter 4

```
library(LearnBayes)
data(marathontimes);attach(marathontimes)
# Draw a contour plot of the log posterior
density (normal-gamma) for time data
# Ranges:  mean (220-330), variance (500-9000).
# Point:Max, Lines:10%,1%,0.1% of Max
d = mycontour(normchi2post, c(220, 330, 500,
9000), time, xlab="mean",ylab="variance")
# S: sum of squares for time
S = sum((time - mean(time))^2)
n = length(time)
# Generate σ^{2(i)} ~ G((n-1)/2, S/2),  i = 1,...,1000
sigma2=1/rgamma(1000,shape=(n-1)/2,rate=S/2)
```

```
# Generate μ^(i)|σ^2(i) ~ N(time‾, σ^2/n),  i = 1,...,1000
mu = rnorm(1000, mean = mean(time), sd =
sqrt(sigma2)/sqrt(n))
# Add these points to the plot
points(mu, sigma2)
# 2.5th and 97.5-th percentiles to construct 95%
# credible intervals for μ and σ
quantile(mu, c(0.025, 0.975))
quantile(sqrt(sigma2), c(0.025, 0.975))
```

Section 4.3

```
library(LearnBayes)
# Parameter α = (α1, α2, α3) for Dirichlet dist
alpha = c(728, 584, 138)
# Generate (θ1^(i), θ2^(i), θ3^(i)) ~ D(α1, α2, α3), i = 1, ..., 1000
theta = rdirichlet(1000, alpha)
# Draw histogram for θ1 − θ2
hist(theta[, 1] - theta[, 2], main="")
# Use election data
data(election.2008);attach(election.2008)
# Define the function prob.obama(j) that estimates
# the winning probability of Obama in j-th State
```

```
prob.Obama=function(j)
{
# Generate $(\theta_1^{(i)}, \theta_2^{(i)}, \theta_3^{(i)}) \sim D(\alpha_1, \alpha_2, \alpha_3)$, $i = 1, \ldots, 5000$
# $\alpha_1 = 500 \times$ support rate for McCain $+ 1$
# $\alpha_2 = 500 \times$ support rate for Obama $+ 1$
# $\alpha_3 = 500 \times$ support rate for others $+ 1$
p=rdirichlet(5000,
500*c(M.pct[j],O.pct[j],100-M.pct[j]-O.pct[j])/100+1)
# Estimate Pr(Obama wins) by $\sum_{i=1}^{5000} I(\theta_2^{(i)} > \theta_1^{(i)})/5000$
mean(p[,2]>p[,1])
}
# Repeat **prob.Obama(j)** for $j = 1, \ldots, 51$ and
# save $\hat{\Pr}$(Obama wins) in **Obama.win.probs**
Obama.win.probs=sapply(1:51,prob.Obama)
```

Section 4.3 (continued)

```
# Define a fuction to simulate elections
sim.election=function()
{
# Generate winner(j)~Bernoulli(Obama.win.probs(j))
# j = 1,...,51.  Bin(1,p)=Bernoulli(p)
# winner(j)=1 if Obama wins, 0 otherwise
winner=rbinom(51,1,Obama.win.probs)
# Compute the total electoral votes by
# sum_{j=1}^{51} EV(j) * winner(j):(the winner takes all EVs)
sum(EV*winner)
}
# Simulate elections 1000 times
sim.EV=replicate(1000,sim.election())
```

```
# Draw a histogram for EV
hist(sim.EV,min(sim.EV):max(sim.EV),col="blue")
# Draw a vertical line at 365 since Obama
# received 365 votes
abline(v=365,lwd=3)
# Legend at coordinates (x,y)=(375,30)
# \n means a new line
text(375,30,"Actual \n Obama \n total")
```