Homework 9

Megan Robertson

Wednesday, November 04, 2015

Defining the Model

The model is defined as:

$$Y_{ij} \sim Poisson(\lambda_i \theta t_{ij})$$
$$\theta \sim Gamma(a, b)$$
$$\lambda_i \sim Gamma(\phi, \phi)$$
$$\phi \sim Gamma(c, d)$$

where Y_{ij} is the count of candy, θ is the overall effect that accounts for variation between the individuals (age, personality types, other differences, etc.), λ_i is the costume effect, and ϕ is used to get different parameters for λ for each costume type.

Likelihood and Full Conditionals

$L(Y_{ij}|\lambda_i, \theta, \phi) = \prod_i \prod_j \frac{(\lambda_i \theta t_{ij})^{(y_{ij})}}{y_{ij}}$
$= \frac{(\lambda_i \theta t_{ij})^{\Sigma_i \Sigma_j y_{ij}}}{\prod_i \prod_j y_{ij}!}$

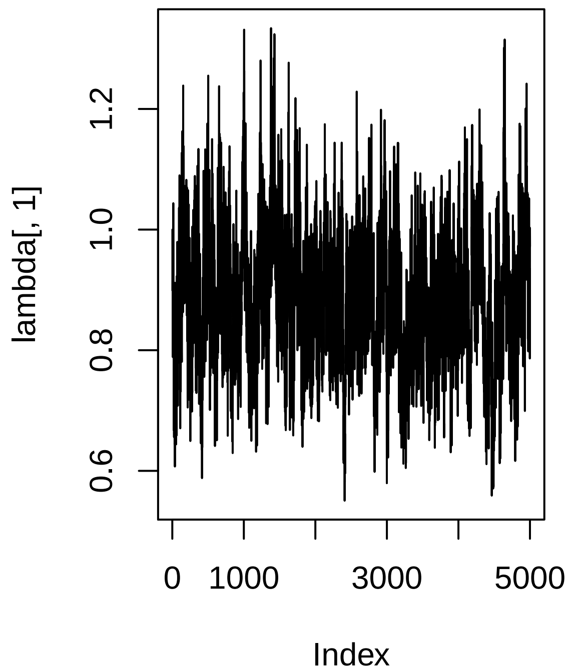
$\pi(\theta) \sim (\theta)^{\Sigma_i \Sigma_j y_{ij}} exp(\Sigma_i \Sigma_j (-\lambda_i \theta t_{ij}))\theta^{a-1} exp(-b\theta)$
$= \theta^{\Sigma_i \Sigma_j y_{ij} + a - 1} exp(-\theta(\Sigma_i \Sigma_j \lambda_i t_{ij} + b))$
$\theta \sim Gamma(\Sigma_i \Sigma_j y_{ij} + a, \Sigma_i \Sigma_j \lambda_i t_{ij} + b)$

$\pi(\lambda_i) \sim \lambda_i^{\Sigma_j y_{ij}} exp(\Sigma_j \lambda_i \theta t_{ij})\lambda_i^{\phi-1} exp(-b\phi)$
$= (\lambda_i)^{\Sigma_j y_{ij} + \phi - 1} exp(-\lambda_i(\Sigma_j \theta t_{ij} + \phi))$
$\pi(\lambda_i) \sim Gamma(\Sigma_j y_{ij} + \phi, \Sigma_j \theta t_{ij} + \phi)$

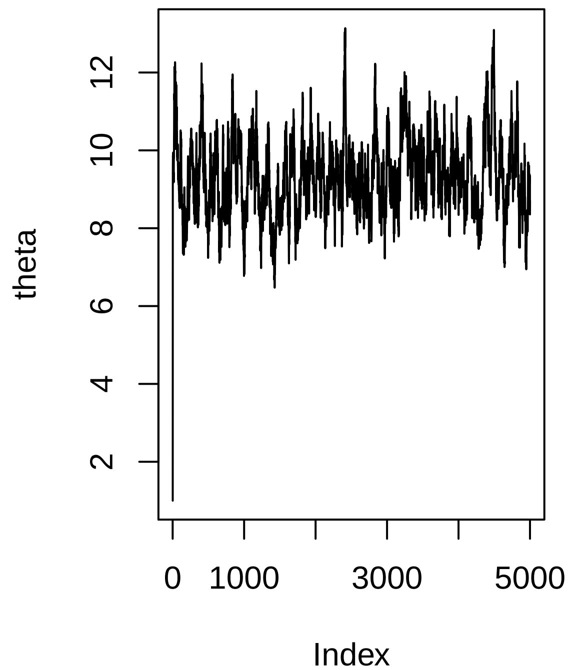
$\pi(\phi) \sim \lambda_i^{\phi-1} exp(-b\phi)\phi^{c-1} exp(-d\phi)$
$= \lambda_i^{\phi-1} exp(-\phi(b + d))\phi^{c-1}$

The full conditional for ϕ is not conjugate, so it is necessary to use the Metropolis algorithm in the code together with the Gibbs sampler.

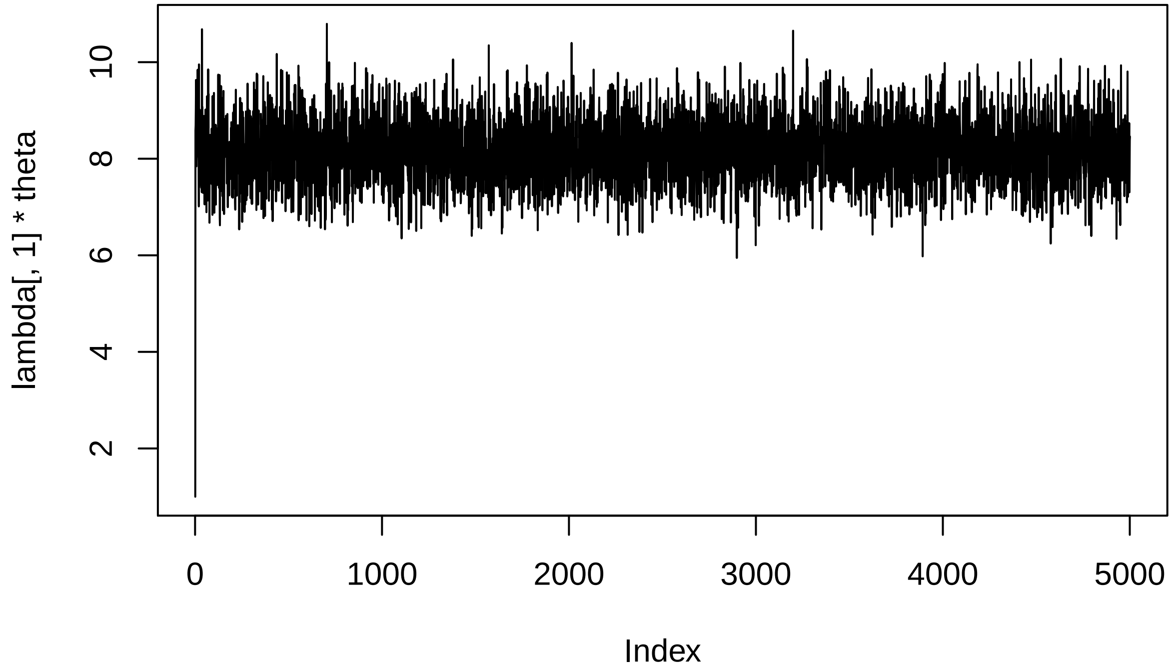
Results



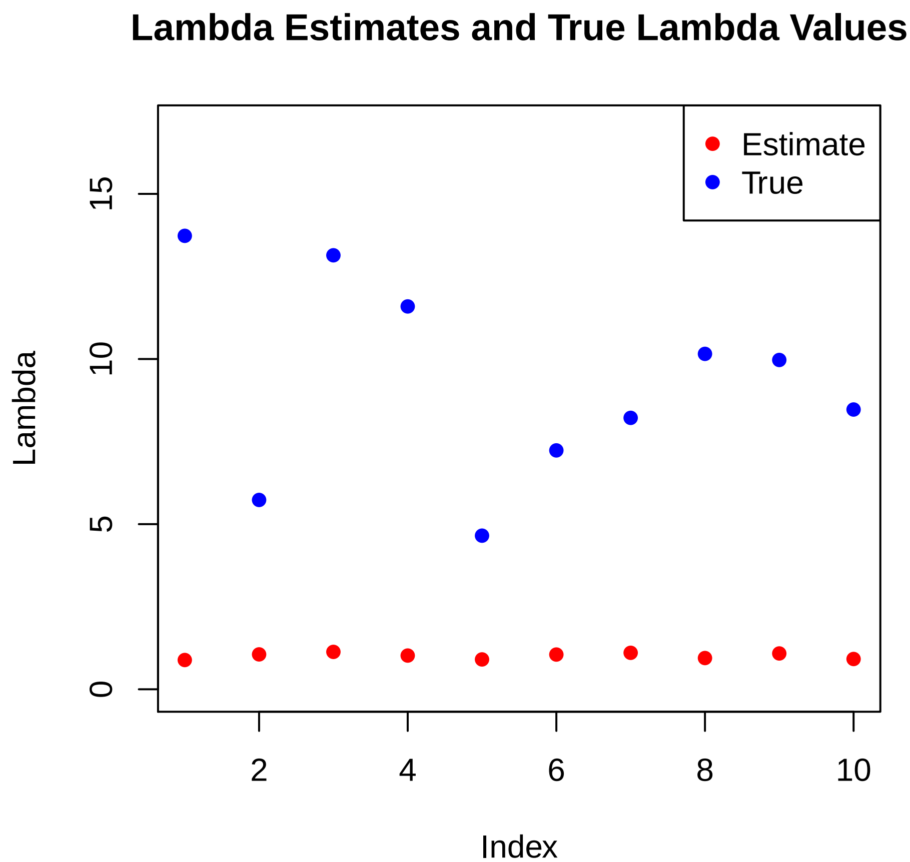
lambda[, 1]
0.6
0.8
1.0
1.2
0
1000
3000
5000
Index



theta
2
4
6
8
10
12
0
1000
3000
5000
Index



The first pair of plots shows that the posteriors for the θ and λ_1 values resulting from the MCMC method implemented do not converge. However, the trace plot of $\lambda_1\theta$ shows that the product of these two parameter distributions converge. This occurs since introducing more parameters into the model can reduce autocorrelation.



The MSE for the λ_i values was found to be large at 76.65104. The plot above shows that the MCMC method implemented results in estimates for the λ_i values that are significantly smaller than the values of λ_i that were used to generate the data. This suggests that the Metropolis algorithm may not be appropriate to use in the MCMC method. The Metropolis algorithm assumes that ϕ^* is drawn from a symmetric distribution, and this may not be an appropriate assumption to make. It is also possible that the parameter values selected for θ and ϕ can be improved. The MSE did not change very much when I changed the value for kappa in the Metropolis portion of the code.

Code Section

```
#simulate data
set.seed(17)
time.vector = c();
for (i in 1:100){
  time.vector[i] = rpois(n=1, lambda=2)
  while (time.vector[i]==0){
    time.vector[i] = rpois(n=1, lambda=2)
  }
}

lambda.1 = rgamma(n=10, shape = 10, rate =1) #generating random rates
lambda.vector2 = rep(lambda.1, each = 10)
```

```r
possible.lambda = seq(0.5, 20, length.out=40)

sample.lambda = time.vector*lambda.vector2
candy.count=c()
for (j in (1:length(sample.lambda))){
  candy.count[j] = rpois(n=1, lambda = sample.lambda[j])
}
data = matrix(candy.count, nrow = 10, byrow=T)
time = matrix(time.vector, nrow = 10, byrow=T)

#defining objects to be used in the Gibbs sampler
R = 5000 #number of iterations for Gibbs
lambda = matrix(0, nrow = R+1, ncol=10); phi=c(); theta=c()
mu = matrix(0, nrow=R+1, ncol=10)
lambda[1,] = rep(1,10); phi[1] = 1; theta[1] = 1

reject = 0; R =5000
lambda = matrix(0, nrow = R+1, ncol=10); phi=c(); theta=c()
lambda[1,] = rep(1,10); phi[1]=1; theta[1] = 1; c = d=1/2; a = b= 1
kappa = 10; phi.star = 0

for (i in (1:R)){
  #Metropolis for phi
  phi.star = 0
  while (phi.star<=0){
    phi.star = rnorm(n = 1, mean = phi[i], sd = kappa)
  }
  numerator = prod(dgamma(lambda[i,], phi.star, phi.star))*dgamma(phi.star,
    c, d)
  denominator = prod(dgamma(lambda[i,], phi[i], phi[i]))*dgamma(phi[i], c, d)
  alpha = min(log(1), log(numerator/denominator))
  test.prob = runif(n=1, 0, 1)
  if (log(test.prob)<alpha){
    phi[i+1] = phi.star
    } else {
    phi[i+1] = phi[i]
    reject = reject + 1
  }

  #getting the new theta
  theta[i+1] = rgamma(n=1, shape = sum(data) + a, sum(lambda[i,]*time) + b)

  #getting the new lambda
  lambda[i+1, ] = rgamma(n=10, shape = colSums(data) + phi[i+1],
      rate = theta[i+1]*colSums(time) + phi[i+1])
}

lambda.vector = colMeans(lambda)

par(mfrow=c(1,2))
plot(lambda[,1], type="l") #plotting the lambda one rates
plot(theta, type="l")
par(mfrow=c(1,1))
```

```
plot(lambda[,1]*theta, type="l")

MSE = 0
for (k in (1:length(lambda.1))){
  MSE = MSE + ((lambda.1[k]-lambda.vector[k])^2)/length(lambda.1)
}
plot(lambda.vector, main="Lambda Estimates and True Lambda Values", ylim=c(0, 17),
     ylab="Lambda", col="red", pch =16)
points(lambda.1, col="blue", pch=16)
legend("topright", c("Estimate", "True"), pch=c(16,16), col=c("red", "blue"))
```

I got a lot of help from classmates on this assignment in terms of figuring out a model and determining the code necessary for the different methods. I also consulted with online sources.

Prior to executing the method above, I was working with a different model in order to approach this problem. However, it is pretty challenging to code as it involves using the griddy Gibbs method. I worked on this model with a peer for a while, and I wanted to include the derivations in case the method above was incorrect. I also wanted to include it because of the amount of time and energy that was devoted to it. The last pages are relevant to this model.

Megan Robertson
Statistics 601
November 3, 2015

Define the model as follows:

Y_{ij} = count of candy $\quad Y_{ij} \sim Poisson(t_{ij}\lambda_{ij})$

t_{ij} = time spent trick or treating

λ_{ij} = rate of candy $\quad \lambda_{ij} \sim \log Normal(\alpha_i + \mu_j, \tau^{-1}) \quad \tau \sim Gamma(c,d)$

α_i = costume effect $\quad \alpha_i \sim N(0, \psi^{-1}) \quad \psi \sim Gamma(a,b)$

μ_j = person effect $\quad \mu_j \sim N(0, \gamma^{-1}) \quad \gamma \sim Gamma(e,f)$

C = overall mean effect for everyone in each costume $\quad C \sim N(\mu_0, \sigma_0^2)$ where μ_0 is mean of data and σ_0^2 is variance of data

Joint Distribution

$$\pi(Y_{ij}, t_{ij}, \lambda_{ij}, \alpha_i, \mu_j, C, \psi, \tau, \gamma) \propto \frac{(t_{ij}\lambda_{ij})^{Y_{ij}} \exp(-t_{ij}\lambda_{ij})}{Y_{ij}!}$$

$$\cdot \lambda_{ij}^{-1} \tau^{1/2} \exp\left(-\tfrac{1}{2}\tau(\ln\lambda_{ij} - C - \alpha_i - \mu_j)^2\right)$$

$$\cdot (\psi 2\pi)^{-1/2} \exp\left(\frac{-\alpha_i^2}{2}\psi\right)$$

$$\cdot (\gamma 2\pi)^{-1/2} \exp\left(\frac{-\mu_j^2}{2}\gamma\right)$$

$$\cdot \frac{b^a}{\Gamma(a)} \psi^{a-1} \exp(-b\psi)$$

$$\cdot \frac{d^c}{\Gamma(c)} \tau^{c-1} \exp(-d\tau)$$

$$\cdot \frac{f^e}{\Gamma(e)} \gamma^{e-1} \exp(-f\gamma)$$

$$\cdot (\sigma_0^{-2})(2\pi)^{-1/2} \exp\left(\frac{-(C^2 - \mu_0)}{2\sigma_0^2}\right)$$

Deriving Full Conditionals

$$\pi(\alpha_i \mid Y_{ij}, t_{ij}, \lambda_j, \mu_j, c, \psi, \tau, \gamma) \propto \prod_j exp(-\tfrac{1}{2}\tau(\ln\lambda_j - c - \alpha_i - \mu_j)^2) exp(\tfrac{-\alpha_i^2}{2}\psi)$$

$$= exp(-\tfrac{1}{2}\tau \cdot \sum_j(\cancel{(\ln\lambda_{ij})^2} - \cancel{2\ln\lambda_j c} - 2\ln\lambda_j\alpha_i - \cancel{2\ln\lambda_j\mu_j} + \cancel{c^2} + \cancel{2c\mu_j} + \alpha_i^2 + 2\alpha_i\mu_j + \cancel{\mu_j^2})) exp(\tfrac{-\alpha_i^2}{2}\psi)$$

$$= exp(-\tfrac{1}{2}\tau(-2\alpha_i\sum_j \ln\lambda_{ij} + \sum_j \alpha_i^2 + 2\alpha_i\sum_j \mu_j)) exp(\tfrac{-\alpha_i^2}{2}\psi)$$

$$= exp(\alpha_i^2(\underbrace{-\tfrac{1}{2}\tau j - \tfrac{1}{2}\psi}_{a}) + \alpha_i(\underbrace{\tau\sum_j \ln\lambda_{ij} + 2\sum_j \mu_j}_{b}))$$

$$\frac{b}{2a} = \frac{\tau\sum_j \ln\lambda_{ij} + 2\sum_j \mu_j}{-(\tau j + \psi)} \qquad \text{Define } A = \frac{\tau\sum_j \ln\lambda_{ij} + 2\sum_j \mu_j}{\tau_j + \psi}$$

By completing the square

$$= exp(-\tfrac{1}{2}(\tau_j + \psi)(\alpha_i - \frac{\tau\sum_j \ln\lambda_{ij} + 2\sum_j \mu_j}{\tau_j + \psi})) \Rightarrow \alpha_i \sim Normal(A, \frac{1}{\tau_j + \psi})$$

$$\pi(\mu_j \mid Y_{ij}, t_{ij}, \lambda_{ij}, c, \psi, \tau, \gamma, \alpha_i) \propto \prod_i exp(-\tfrac{1}{2}\tau(\ln\lambda_{ij} - c - \alpha_i - \mu_j)^2) exp(-\tfrac{\mu_j^2}{2}\gamma)$$

$$= exp(-\tfrac{1}{2}\tau \cdot \sum_i(-2\ln\lambda_{ij}\mu_j + 2c\mu_j + \mu_j^2)) exp(\tfrac{-\mu_j^2}{2}\gamma)$$

$$= exp(\mu_j^2(\underbrace{-\tfrac{1}{2}\tau i - \tfrac{1}{2}\gamma}_{a}) + \mu_j(\underbrace{\tau\sum_i \ln\lambda_{ij} - \tau c i}_{b}))$$

$$\frac{b}{2a} = \frac{\tau\sum_i \ln\lambda_{ij} - \tau c i}{-(\tau i + \gamma)} \qquad \text{Define } F = \frac{\tau\sum_i \ln\lambda_{ij} - \tau c i}{(\tau i + \gamma)}$$

By completing the square,

$$= exp(-\tfrac{1}{2}(\tau i + \gamma)(\mu_j - F)^2) \Rightarrow \mu_j \sim Normal(F, \frac{1}{\tau i + \gamma})$$

$$\pi(\psi \mid Y_{ij}, t_{ij}, \lambda_{ij}, c, \tau, \gamma, \alpha_i, \mu_j) \propto \prod_i [\psi^{1/2}(2\pi)^{-1/2} exp(\tfrac{-\alpha_i^2}{2}\psi)] \psi^{a-1} exp(-b\psi)$$

$$\propto (\psi)^{1/2} exp(-\tfrac{\psi}{2}\sum_i \alpha_i^2)\psi^{a-1} exp(-b\psi)$$

$$= (\psi)^{a+1/2-1} exp(-\psi(\tfrac{1}{2}\sum_i \alpha_i^2 + b))$$

$$\Rightarrow Gamma(a + 1/2, \tfrac{1}{2}\sum_i \alpha_i^2 + b))$$

$$\pi(\tau \mid \gamma, \psi, \mu_j, \alpha_i, \tau, c, \lambda_{ij}, Y_{ij}) \propto \prod_i \prod_j \tau^{1/2} exp(-\tfrac{1}{2}\tau(\ln\lambda_{ij} - c - \alpha_i - \mu_j)^2)\, \tau^{c-1} exp(-d\tau)$$

$$= \tau^{np/2} exp(-\tfrac{1}{2}\tau \sum_i \sum_j (\ln\lambda_{ij} - c - \alpha_i - \mu_j)^2)\, \tau^{c-1} exp(-d\tau)$$

$$= \tau^{np/2+c-1} exp(-\tau(\tfrac{1}{2}\sum_i \sum_j (\ln\lambda_{ij} - c - \alpha_i - \mu_j)^2 + d))$$

$$\Rightarrow \tau \sim Gamma(\tfrac{np}{2}+c,\ \tfrac{1}{2}\sum_i \sum_j (\ln\lambda_{ij} - c - \alpha_i + \mu_j)^2 + d)$$

$$\pi(\gamma \mid \tau, \psi, \mu_j, \tau, \alpha_i, c, \lambda_{ij}, Y_{ij}) \propto \prod_{j=1}^{p} \left[(\gamma)^{1/2} exp(\tfrac{-\mu_j^2}{2}\gamma)\right] \gamma^{e-1} exp(-f\gamma)$$

$$\propto \gamma^{p/2} exp(\sum_j -\tfrac{\mu_j^2}{2}\gamma)\, \gamma^{e-1} exp(-f\gamma)$$

$$= \gamma^{p/2+e-1} exp(-\gamma(\sum_j \tfrac{\mu_j^2}{2} + f))$$

$$\gamma \sim Gamma(p/2 + e,\ \sum_j \tfrac{\mu_j^2}{2} + f)$$

$$\pi(\lambda_{ij} \mid \tau, \gamma, \psi, \mu_j, \alpha_i, c, Y_{ij}) \propto (t_{ij}\lambda_{ij})^{Y_{ij}} exp(-t_{ij}\lambda_{ij})\, \lambda_{ij}^{-1} exp(-\tfrac{1}{2}\tau(\ln\lambda_{ij} - c - \alpha_i - \mu_j)^2)$$

↳ This is not conjugate, so griddy Gibbs (or another method) is necessary.

$$\pi(c \mid \lambda_{ij}, \tau, \gamma, \psi, \mu_j, \alpha_i, c, Y_{ij}) \propto exp(-\tfrac{1}{2}\tau(\ln\lambda_{ij} - c - \alpha_i - \mu_j)^2) \tfrac{1}{\sigma_0} exp(\tfrac{-(c-\mu_0)^2}{2\sigma_0^2})$$

$$= exp(-\tfrac{1}{2}\tau(-2\ln\lambda_{ij} c + c^2 + 2c\mu_j))\, exp(\tfrac{-c^2 - 2\mu_0 c}{2\sigma_0^2})$$

⇒ By completing the square, this will be a normal distribution for the full conditional for c.