R and WinBUGS

Peng Ding
Department of Probability and Statistics
School of Mathematical Sciences
Peking Univeristy
Email: dingyunyiqiu@163.com

Outline

- Bayesian Statistics
- WinBUGS
- R and WinBUGS: some examples

Bayesian Statistics

- 在频率派看来，参数是客观存在的固定常数，统计的任务之一是估计这些参数，包括点估计和区间估计。
- 贝叶斯学派认为，参数是随机变量，有一个概率分布，贝叶斯统计主要任务就是推断参数在给定数据下的条件分布。
- 如果参数 $\theta \in \Theta$ 有一个先验分布 $\pi(\theta)$，我们观测到了从条件分布 $p(y|\theta)$ 产生的样本 y ，由贝叶斯公式可以得到 θ 的后验分布是
$$\pi(\theta|y) = \frac{p(y|\theta)\pi(\theta)}{\int_\Theta p(y|\theta)\pi(\theta)d\theta}.$$

贝叶斯统计中的一些问题

- 先验分布的选取
- 后验分布的计算问题

R中做贝叶斯推断的包

- MCMCpack 中单参数、多参数以及广义线性模型的贝叶斯实现;
- 包 “arm” 中的函数 bayesglm 也可以用贝叶斯方法实现广义线性模型;
- R2WinBUGS 和 BRugs 中的函数 bugs() 和 BRugsFit()，后面专门介绍。

WinBUGS 简介

- WinBUGS 的前身是 BUGS(Bayesian inference Using Gibbs Sampling)
- 剑桥大学开发的贝叶斯统计推断的专门软件；并且它是免费的
- 简单说来，WinBUGS 中的模型，都是用一个有向无环图(DAG)来描述的。
- 在每个有向无环图中，我们可以将变量之间的联合分布写成：

  $$P(V) = \prod_{v \in V} P(v|pa(v)),$$

  其中，V 表示所有变量的集合，v 是其中的一个元素，$pa(v)$ 是 v 的父亲节点。
- 在 WinBUGS 的模型中，我们需要描述变量系统的所有条件分布。

一个简单的例子: WinBUGS 中的例子 Surgical

- 数据来源于 12(N) 个医院中某种手术的病人总数 (n) 和死亡数 (r)，如下存放在一个 list 中:

```
list(n = c(47, 148, 119, 810, 211, 196,
           148, 215, 207, 97, 256, 360),
 r  = c(0, 18, 8, 46, 8,
           13, 9, 31, 14, 8, 29, 24),
 N = 12)
```

- 目的在于估计每个医院此手术的死亡率 $p_i, i = 1, ..., N$.
- 模型$r_i \sim Binomial(n_i, p_i)$，取先验分布为 $p_i \sim Unif(0, 1)$.

WinBUGS中的描述

- WinBUGS 中的模型为:

```
model
{
   for( i in 1 : N ) {
              p[i] ~ dbeta(1.0, 1.0)
              r[i] ~ dbin(p[i], n[i])
   }
}
```

- WinBUGS 的语言和 R 语言基本上是一致的，不同之处在于某些分布的名称以及参数的顺序不同，这点可以查阅 WinBUGS 帮助文件中的“distribution”。
- 初始值也存放在一个 list 中:

```
list(p = c(0.1, 0.1, 0.1, 0.1,0.1, 0.1,
           0.1, 0.1, 0.1, 0.1, 0.1, 0.1))
```

WinBUGS中实现的详细步骤(I)

1. WinBUGS → Model → Specification → Specification Tool;
2. 将光标移动到上面所写的 model 之前并将 model 选中→ 点击 Specification Tool 中的 model check ，如果在 WinBUGS 的左下方出现了“model is syntactically correct”，说明你的模型没有语法的错误;
3. 将光标移动到上面存放 data 的 list 之前并将 list 选中→ 点击 Specification Tool 中的 load data ，如果在 WinBUGS 的左下方出现了“data loaded”，说明已经正常读入数据;
4. 点击 Specification Tool 中的 compile ，如果在 WinBUGS 的左下方出现了“model compiled”，说明已经编译通过;

WinBUGS中实现的详细步骤(II)

5. 将光标移动到上面存放初始值的 list 之前并将 list 选中→ 点击 Specification Tool 中的 load inits ，如果在 WinBUGS 的左下方出现了“model is initialized”，说明初始值已经正常产生;
6. WinBUGS → Inference → Samples → Sample Monitor Tool;
7. 在 node 出输入你关心的参数(比如上面我们关心参数p，注意这是一个向量)，并点“set”，如此重复输入所有你关心的参数;
8. WinBUGS → Model → Update → Update Tool;
9. 在 updates 处输入你需要的迭代数目，然后点 updata ，你将看到 WinBUGS 正在迭代的次数;
10. 在 Sample Monitor Tool 的 node 出输入关心的参数，若输入“*”表示全部的参数，点 density 或者 stats 等等，得到后验分布的密度和统计量。

R 调用 WinBUGS

在使用 R 调用 WinBUGS 之前，请做好如下的准备:

1. 安装 WinBUGS 到: c:/Program Files/ WINBUGS14/，并注册，否则你讲不能使用 WinBUGS 全部的功能;
2. 安装 OpenBUGS 到: c:/Program Files/OpenBUGS/;
3. 安装最近版本的 R;
4. 通常情况下，每次用 R 调用 WinBUGS 之前都请调用如下的 R package: “arm”，“Brugs”，“R2WinBUGS”；如果你使用 Windows Vista，可能会遇到麻烦，可到 Andrew Gelman 的个人主页查看相关信息: http://www.stat.columbia.edu/ gelman/bugsR/。

例1: Meta-analysis

所谓 meta-analysis，简单的想就是综合利用不同的数据来源，整合他们的结果。下面的数据来自 WinBUGS 的例子 Blocker，存放在一个 list 中:

```
data=list(rt = c(3,  7,  5,  102,  28, 4,  98,  60, 25, 138
                45,  9, 57, 25, 33, 28, 8, 6, 32, 27, 22 ),
     nt = c(38, 114, 69, 1533, 355, 59, 945, 632, 278,1916,
           263, 291, 858, 154, 207, 251, 151, 174, 209, 391
     rc = c(3, 14, 11, 127, 27, 6, 152, 48, 37, 188, 52,
           47, 16, 45, 31, 38, 12, 6, 3, 40, 43, 39),
     nc = c(39, 116, 93, 1520, 365, 52, 939, 471, 282, 192
           266, 293, 883, 147, 213, 122, 154, 134, 218, 364
     Num = 22)
```

Model: meta-analysis

假如如下的模型保存在文件 Meta RR with weight.bug 中:

```
model
{
   for( i in 1 : Num ) {
        r[i] ~ dbin(pc[i], nc[i])      ##control group
        rt[i] ~ dbin(pt[i], nt[i])     ##treatment group
        log(pc[i]) <- mu[i]
        log(pt[i]) <- mu[i] + delta[i]##here delta[i] is in
        mu[i] ~ dnorm(0.0,1.0E-5)
        delta[i] ~ dnorm(d, tau)       ##here d is overall l
                             }
       ##prior
       d ~ dnorm(0.0,1.0E-6)
       tau ~ dgamma(0.001,0.001)
}
```

R: meta-analysis

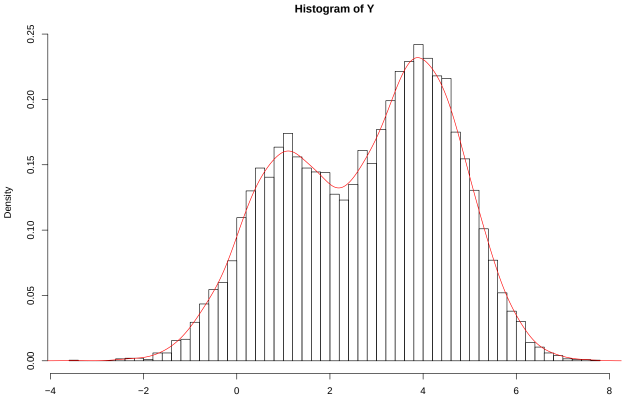
在 R 中用如下的方式拟合分层贝叶斯模型:

```
inits=list(list(d = 0, tau=1,
           mu = c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
                  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),
   delta = c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
               0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0)))
parameters=c("d","delta")   ##parameters
meta=bugs(data,inits,parameters,n.chains=1,n.sims=10000,
          "Meta RR with weight.bug")
meta$summary
attach.bugs(meta)
plot(density(d))
plot(density(delta))
```

其中的“meta$summary”语句可以输出所有后验分布的统计量，而两个plot函数，则可以输出相应参数的后验密度。

例2: 混合分布

- 模型: $Y = PY_1 + (1 - P)Y_0$;
- Y_1和Y_0是两个正态分布, P是一个二值变量, 指示样本来自于哪个总体;
- 例子



例2: 混合分布(I)

```
##Gaussian Mixture model in Winbugs: Eyes
##note: R can not read "I()", so there are some differences
##"%_%" is used before I().
gmmodel=function(){
       for( i in 1 : N ){
            y[i] ~ dnorm(mu[i], tau)
            mu[i] <- lambda[T[i]]
            T[i] ~ dcat(P[])
                }
        P[1:2] ~ ddirch(alpha[])
        theta ~ dnorm(0.0, 1.0E-6)%_%I(0.0, )
        lambda[2] <- lambda[1] + theta
        lambda[1] ~ dnorm(0.0, 1.0E-6)
        tau ~ dgamma(0.001, 0.001)
        sigma <- 1 / sqrt(tau)  }
```

例2: 混合分布(II)

```
##write model into the dictionary
if (is.R()){ # for R
    ## some temporary filename:
    filename <- file.path(tempdir(), "gmm.bug")
} else{ # for S-PLUS
     ## put the file in the working directory:
     filename <- "gmm.bug"
}
## write model file:
write.model(gmmodel, filename)
```

例2: 混合分布(III)

```
data=list(y = c(529.0, ...), N = 48, alpha = c(1, 1),
      T = c(1, NA, ..., NA, NA, 2))
inits=function()
{  list(lambda = c(535, NA), theta = 5, tau = 0.1)  }
parameters=c("P","lambda","tau")
GaussianMixture=bugs(data,inits,parameters,
                     n.chains=1,model.file=filename)
print(GaussianMixture)
attach.bugs(GaussianMixture)
par(mfrow=c(2,2))
plot(density(P[,1]),main="P1")
plot(density(1/tau),main="1/tau")
plot(density(lambda[,1]),main="lambda1")
plot(density(lambda[,2]),main="lambda2")
```

例3：非随机缺失数据的参数估计

- 考虑如下的图模型：

$$Z \to Y \to R$$

  例如，Z 表示随机化试验，Y 表示试验的结果，R 是一个示性变量，当 Y 缺失时取 0，完全观测时取 1。
- R 的分布依赖于结果变量 Y，这是完全非随机的缺失。
- Ma, Geng and Hu (2003) 证明了此模型的可识别性(identifiability).
- 频率学派对于这种缺失数据问题一般用 EM 算法，这里给出的贝叶斯方式显得更加的直观。
- 贝叶斯方法对于缺失数据有天然的优势，WinBUGS 中采用 Gibbs 采样，实际上是把缺失的数据也看成了参数，使用数据扩充算法更新后验分布。

Model: missing completely nonignorable

```
model = function()
              {
               for(j in 1:N){
                    Z[j]~ dbern(p)
                    Y[j] ~ dbern(pzy[Z[j]+1])
                     ##this intermediate is a must!!
                    index[j]<-Y[j]+1
                    R[j] ~ dbern(pyr[index[j]])
                                          }
               ##prior
               p ~ dunif(0,1)
               pzy[1] ~ dunif(0,1)
               pzy[2] ~ dunif(0,1)
               pyr[1] ~ dunif(0,1)
               pyr[2] ~ dunif(0,1)
               }
```

例4: Bayesian LASSO

- 在经典的线性回归

$$y = X\beta + \varepsilon$$

中,如果解释变量X的个数比较多，则会出现如下问题:

1. 解释变量的共线性增强，使得OLS估计非常不稳定，在统计上表现为估计系数的方差过大;
2. 回归模型过于冗长，不利于解释，并且预测的方差很大。

- 针对 1，传统的方法是岭回归(ridge regression)，即

$$\widehat{\beta}_{ridge} = (X'X + \lambda I)^{-1} X'Y.$$

对于 2，在实际中，我们希望得到简洁的、更加易于解释且预测方差较小的模型。这是模型选择的领域，通常的方法有向前回归、向后回归和逐步回归，模型的选择标准也多种多样(如 AIC 和 BIC)。

LASSO

- Tibshirani (1996)提出了

$$\widehat{\beta}_{LASSO} = \text{argmin}_{\beta} \sum_{i=1}^{N} (y_i - X_i\beta)^2, \text{s.t.} ||\beta||_1 \leq t$$

- 岭回归有着和LASSO类似的形式，即

$$\widehat{\beta}_{ridge} = \text{argmin}_{\beta} \sum_{i=1}^{N} (y_i - X_i\beta)^2, \text{s.t.} ||\beta||_2^2 \leq t,$$

- 岭回归的贝叶斯观点(Weisberg,1985):在线性模型

$$Y = X\beta + \varepsilon, \varepsilon \sim N(0, \sigma^2 I_n)$$

中，若取β的先验分布是$\beta_j \ iid \sim p(\beta_j) \sim N(0, \tau^2)$，如果用后验的众数作为$\beta$的估计，导出岭回归。
- LASSO对应Laplace分布。

Bayesian LASSO

- Laplace分布可以看成正态分布关于尺度参数的混合分布:

$$\frac{a}{2}e^{-a|z|} = \int_0^\infty \frac{1}{\sqrt{2\pi s}} e^{-z^2/2s} \frac{a^2}{2} e^{-a^2 s/2} ds, a > 0.$$

- 建立如下的与$LASSO$本质等价的贝叶斯分层模型:

$$\begin{aligned}
Y|\mu, X, \beta, \sigma^2 &\sim N_n(\mu 1_n + X\beta + \sigma^2 I_n), \\
\beta|\sigma^2, \tau_1^2, ..., \tau_p^2 &\sim N_p(0_p, \sigma^2 D_\tau), D_\tau = diag\{\tau_1^2, ..., \tau_p^2\}, \\
\sigma^2, \tau_1^2, ..., \tau_p^2 &\sim \pi(\sigma^2) d\sigma^2 \prod_{j=1}^p \frac{\lambda^2}{2} e^{-\lambda^2 \tau_j^2/2} d\tau_j^2,
\end{aligned}$$

另外取μ为平坦的先验分布，$\pi(\sigma^2) = \frac{1}{\sigma^2}$(或者其他Gamma分布)。

Model: Bayesian LASSO

```
model  = function()
{    for(i in 1:N)
      {
           y[i] ~ dnorm(b[i], tau)
           b[i] <- mu + inprod(beta[],x[i,])
      }
     mu ~ dnorm(0.0, 1.0E-6)
     for(j in 1:p)
     {    beta[j] ~ dnorm(0.0, tau.beta[j])
          tau.beta[j] <- 1/tau.beta.inv[j]
          tau.beta.inv[j] ~ dgamma(1,lamda)
     }
     lamda ~ dgamma(0.1, 0.1)
     tau <- pow(sigma, -2)
     sigma ~ dunif(0, 1000)
}
```

致谢

Thank you very much.