# ENSAE - Computational Statistics

Professor Christian P. Robert

Zakarya Ali

December 15, 2017

**Problem 9.3.** The following are genotype on blood type.

| Genotype | Probability | Observed | Probability | Frequency |
|---|---|---|---|---|
| AA | $p_A^2$ | A | $p_A^2 + 2p_Ap_O$ | $n_A = 186$ |
| AO | $2p_Ap_O$ | | | |
| BB | $p_B^2$ | B | $p_B^2 + 2p_Bp_O$ | $n_B = 38$ |
| BO | $2p_Bp_O$ | | | |
| AB | $2p_Ap_B$ | AB | $p_Ap_B$ | $n_{AB} = 13$ |
| OO | $p_O^2$ | O | $p_O^2$ | $n_O = 284$ |

Estimate $p_A$, $p_B$ and $p_O$ using a Gibbs sampler. Make a histogram of the samples.

We know $P_A + P_B + P_O = 1$, thus we only need to deduce $P_A$ and $P_B$.
From Problem 5.18, we get

$$(P_A, P_B, 1-P_A-P_B)|(n, Z_A, Z_B) \sim D(Z_A+n_A+1, Z_B+n_B+n_{AB}+1, n_A-Z_A+n_B-Z_B+2n_O+1),$$

where $n = (n_A, n_B, n_{AB}, n_O)$ and $Z_A, Z_B$ are the missing values such as,

$$Z_A|n, P_A, P_B \sim Bin(n_A, \frac{P_A^2}{P_A^2 + 2P_AP_O})$$

$$Z_B|n, P_A, P_B \sim Bin(n_B, \frac{P_B^2}{P_B^2 + 2P_BP_O})$$

From these laws, we can produce the following Gibbs Sampler :
Figure 1 show the generation of $10^5$ observations from the Gibbs Sampler. And Table 1 present the values estimated.

**Algorithm 1** Gibbs Sampler

$\forall t$,

- Draw $(P_A^{t+1}, P_B^{t+1}) \sim f(Z_A^t, Z_B^t)$ with f density of

$$D(Z_A + n_A + 1, Z_B + n_B + n_{AB} + 1, n_A - Z_A + n_B - Z_B + 2n_O + 1)$$

- $P_O^{t+1} = 1 - P_A^{t+1} - P_B^{t+1}$
- Draw $Z_A^{t+1} \sim g(P_A^t, P_B^t)$ where g is the density of $Bin(n_A, \frac{P_A^2}{P_A^2+2P_AP_O})$
- Draw $Z_B^{t+1} \sim h(P_A^t, P_B^t)$ where h is the density of $Bin(n_B, \frac{P_B^2}{P_B^2+2P_BP_O})$

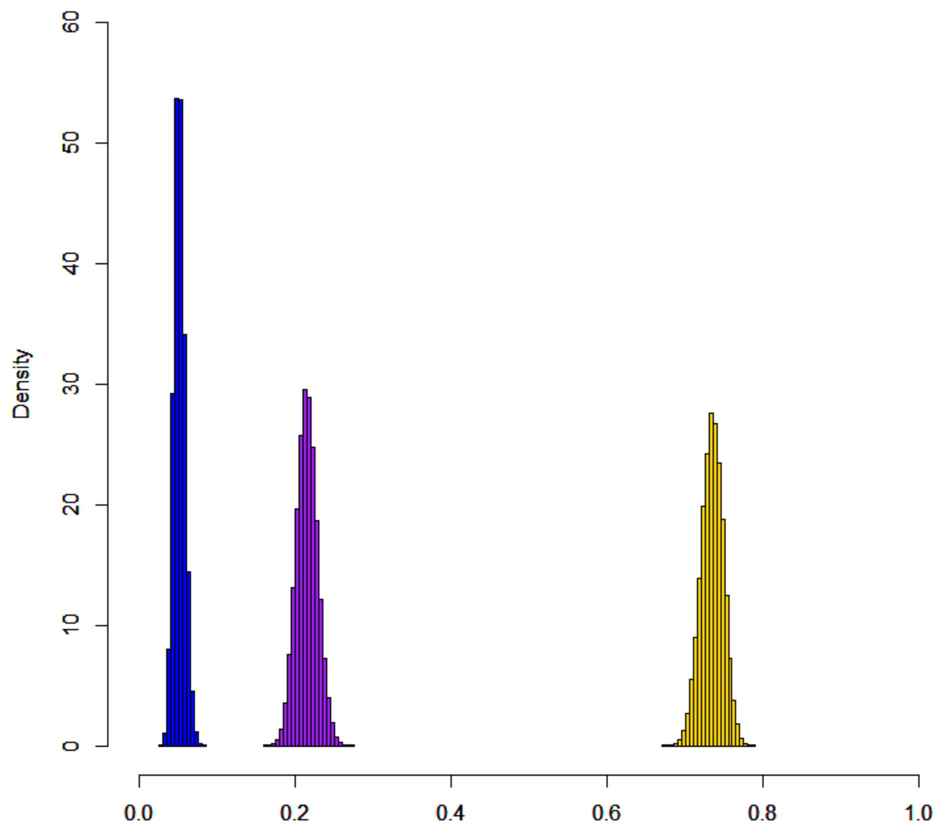


Figure 1: Historgrams of $p_A$ (purple), $p_B$ (blue) and $p_O$ (gold), with Gibbs Sampler

| $p_A$ | $p_B$ | $p_O$ |
|---|---|---|
| 0.2150003 | 0.05104553 | 0.7339541 |

Table 1: Estimations of $p_A, p_B, p_O$

```r
#Initialization
library(gtools)
iterations=10**5
nA=186
nB=38
nAB=13
nO=284

#gibbs sampler
gibbs_sampler = function(iterations, nA, nB, nAB, nO){
  pA=c(.25)
  pB=c(.05)
  pO = 1 - pA - pB
  for (i in 1:iterations-1){
    ZA=rbinom(1,nA,pA[i]^2/(pA[i]^2+2*pA[i]*pO))
    ZB=rbinom(1,nB,pB[i]^2/(pB[i]^2+2*pB[i]*pO))
    temp=rdirichlet(1,c(nA+nAB+ZA+1,nB+nAB+ZB+1,nA-ZA+nB-ZB+2*nO+1))
    pA=c(pA,temp[1])
    pB=c(pB,temp[2])
    pO = c(pO,1-temp[1]-temp[2])
  }
  return (data.frame(pA,pB,pO))
}

#Sample generation
sample = gibbs_sampler(iterations, nA, nB, nAB, nO)

#Histograms
hist(sample[,1],main="",freq=F,col="purple", xlim=c(0,1), ylim=c
    (0,60), xlab="")
par(new=TRUE)
hist(sample[,2],main="",freq=F,col="blue", xlim=c(0,1), ylim=c(0,60),
     xlab="")
par(new=TRUE)
hist(sample[,3], main="",freq=F,col="gold", xlim=c(0,1), ylim=c(0,60)
    , xlab="")

#Estimations
pA_estimate = mean(na.omit(sample[,1]))
pB_estimate = mean(na.omit(sample[,2]))
pO_estimate = 1 - pA_estimate - pB_estimate
```

```
cat("pA_estimate: ", pA_estimate, "; pB_estimate: ", pB_estimate, ";
   pO_estimate: ", pO_estimate, sep = "")
```

Listing 1: Code for Slice Sampler generation and plotting