This code contains algorithm that illustrates the use of the Gibbs sampling algorithm to explore the model space. For illustrative purposes, the Bayes factor is taken as defined by the unit information prior but you are supposed to make use of your own Bayes factor. (The standard way of using BayesVarSel is much more efficient than this code.)

```r
library(MASS)
library(BayesVarSel)
data(UScrime)
#47 16
y <- UScrime$y
X <- UScrime[,1:15]
p <- dim(X)[2]

# Factor Bayes x Prior
B.PM <- function(model){
data <- data.frame(y=y, X=X[,model==1])
tt <- Btest(models=list(M1=~., M0=~1), prior.betas="g", priorprob="c", data=data)
return(tt$BFi0[1]/choose(length(model), sum(model)))
}

###
# B.PM(sample(c(1,0), rep=T, size=16))

# Null model
modelsB.PM <- rep(0, p+1) # the last column contains BF*Pr(M)
modelsB.PM[p+1] <- 1 # B.PM(rep(0,p)) Null vs Null
modelsB.PM[p+2] <- 0 # Label for each model

d.ini = 8  # This is the dimension of the initial model, we have choose an intermediate dimension, it can be modified or random generated.
current.model <- sample(c(rep(1, d.ini), rep(0, p-d.ini)))
B.PMcurrent <- B.PM(current.model)
proposal.model <- current.model
pow2 = 2^((p-1):0)
# matrix of unique models and BFxP(M)
modelsB.PM <- rbind(modelsB.PM, c(current.model, B.PMcurrent, ind=c(pow2%*%current.model)))

visitedmodels.PM <- modelsB.PM[, -(p+2)]

set.seed(12)
N = 300
```
for (i in 1:N){
    #cat("It:",i,"\n")
    for (j in 1:p){
        proposal.model <- current.model; proposal.model[j] <- 1-current.model[j]
        #check if it is in your list
        ind.prop <- c(pow2%%proposal.model)
        coincident <- which(modelsB.PM[,p+2]==ind.prop)
        #cat("coincident", coincident, "\n")
        if (length(coincident)==0) {
            B.PMproposal <- B.PM(proposal.model);
            modelsB.PM <- rbind(modelsB.PM, c(proposal.model, B.PMproposal, ind.prop))
        } else
            B.PMproposal <- modelsB.PM[coincident, p+1]

        #B.PMproposal <- B.PM(proposal.model)
        ratio <- B.PMproposal/(B.PMproposal+B.PMcurrent)
        #cat("ratio", ratio, "\n")

        if (runif(1)<ratio) {current.model[j] <- proposal.model[j]; B.PMcurrent <- B.PMproposal}
    }
}

visitedmodels.PM <- rbind(visitedmodels.PM, c(current.model, B.PMcurrent))

round(colMeans(visitedmodels.PM[, -(p+1)]), 2)

#0.59 0.13 0.82 0.87 0.24 0.12 0.29 0.15 0.09 0.15 0.33 0.12 0.29 0.16 0.12 0.17 0.34 0.22 0.97 0.54 0.11

#the real
crimeBvs <- Bvs(formula=y~., n.keep=10, prior.models="S", prior.betas="g", data=UScrime)
round(crimeBvs$inclprob, 2)

M  So  Ed  Po1  Po2  LF  M.F  Pop  NW  U1  U2  GDP  Ineq  Prob  Time
0.59 0.13 0.80 0.84 0.27 0.13 0.29 0.16 0.12 0.17 0.34 0.22 0.97 0.54 0.14