4. MCMC Posterior Sampling

Dimitrios Ligas

Metropolis-Hastings Algorithm to recover the Posterior Distribution:

The Likelihood of the data is: $X|\theta \sim Multinomial(n, 4, g(\theta))$. The Prior is an uninformative uniform $\pi(\theta) \sim Beta(1, 1)$. The posterior is: $p(\theta|X) \propto f(X|\theta)\pi(\theta)$.

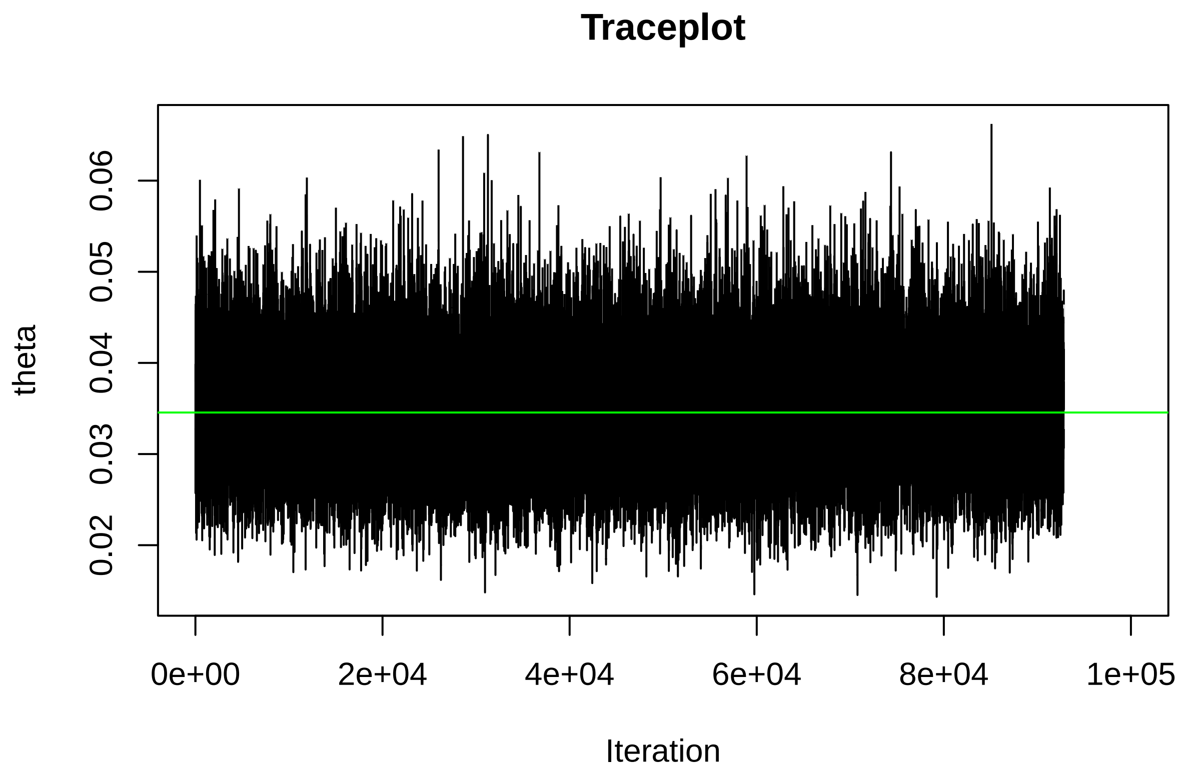
The kernel of this density does not match the kernel of a standard density that we can directly sample from. Therefore, we approximate the posterior using MCMC Sampling. The choice of the jumping distribution q for θ is $\theta^* \sim q(\theta^*|\theta_{j-1}) = Beta(a_{jump}, b_{jump})$ to match the support $\theta \in [0, 1]$. To center the jumping distribution in the previous value θ_{j-1} we set the hyperparameters to $a_{jump} = a \ \ and \ \ b_{jump} = \frac{a_{jump}(1-\theta_{j-1})}{\theta_{j-1}}$ so that $\mathbb{E}[\theta^*|\theta_{j-1}] = \theta_{j-1} and \ scale \ a_{jump}$ to adjust the jumping standard deviation and tune the algorithm in order to accept between 30-40% of the proposed samples. Furthermore, since the jumping distribution is not symmetric, we use the Metropolis-Hastings algorithm instead of plain Metropolis. The acceptance ratio at each iteration will be: $r = \frac{f(X|\theta^*)\pi(\theta^*)q(\theta_{j-1}|\theta^*)}{f(X|\theta_{j-1})\pi(\theta_{j-1})q(\theta^*|\theta_{j-1})}$. To avoid ill-conditioned situations in which the value of the posterior is not well-defined for some values of θ, we compute the log-posteriors, that is the acceptance ratio is evaluated as: $r = \ln[p(\theta^*|X)] - \ln[p(\theta_{j-1}|X)]$.

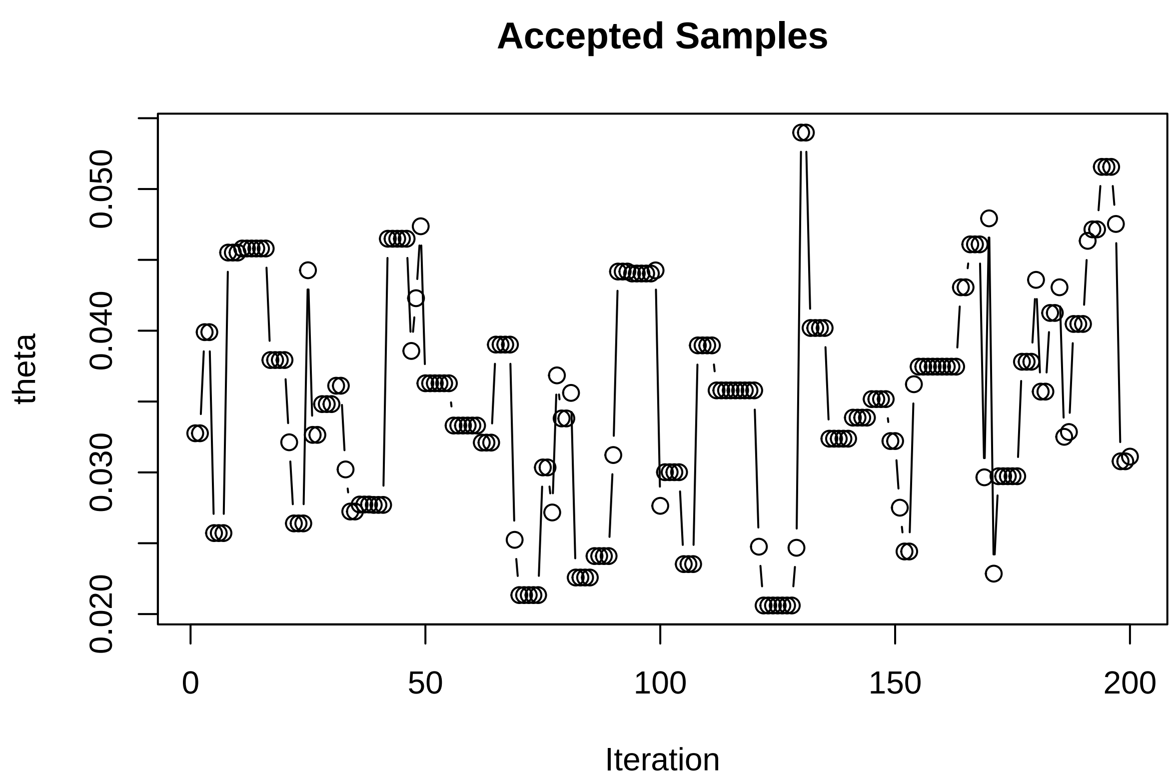
Initiate Metropolis-Hastings sampling:

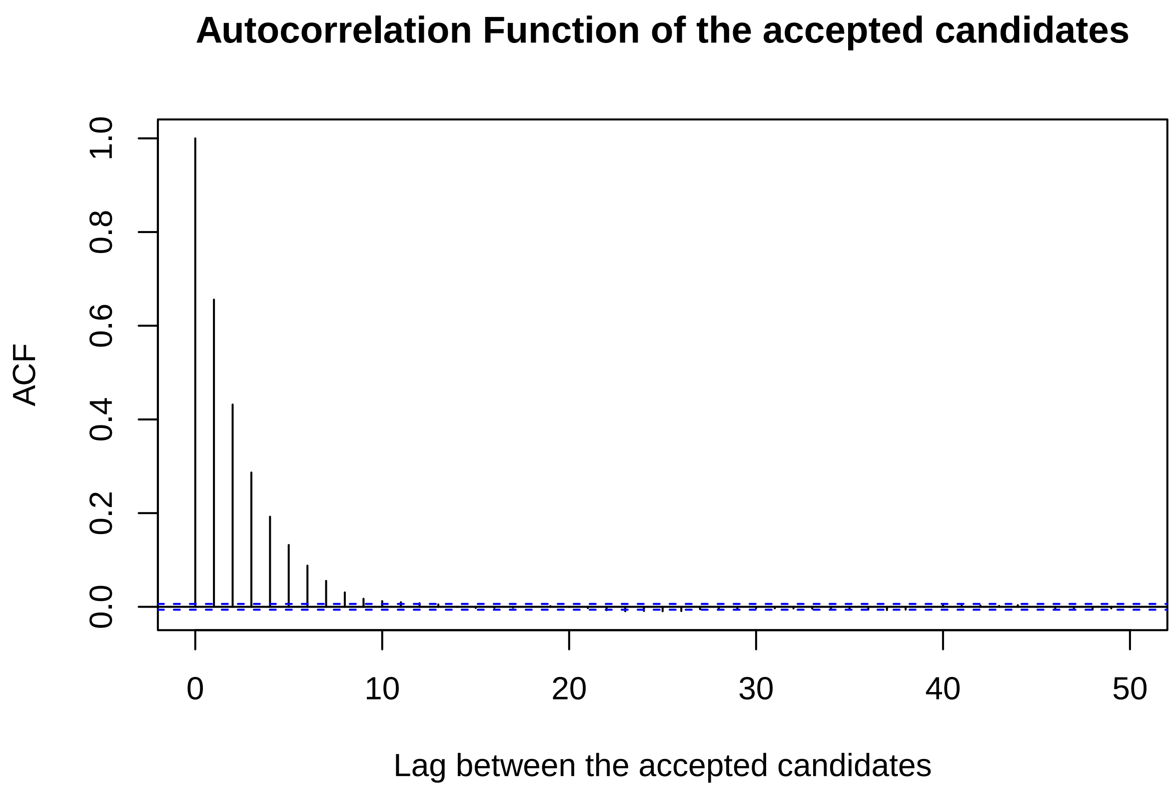
```r
#MCMC Sampling for Posterior recovery
#Load the data:
set.seed(774)
Y=c(1495,750,729,26)
#Preparing for Metropolis Sampling:
S=100000
samples=matrix(NA,S,1)
#Initial Values:
a=b=1#Uniform Prior
#We choose to initialize theta from its maximum likelihood estimation
inits_mle=0.03276042
theta_inits=inits_mle
theta=theta_inits
#Define a routine to update the multinomial probabilities:
mult_prob=function(theta){
  p1=0.25*(2+theta)
  p2=p3=0.25*(1-theta)
  p4=0.25*theta
  p=c(p1,p2,p3,p4)
  return(p)
}
p=mult_prob(theta)
#Tuning the Jumping Distribution:
#The choice of the jumping distribution is Beta(a,b).
#We center it initially to the MLE estimation of theta:
a_jump=3
b_jump=(1-theta)*a_jump/theta
```

```r
jumping_sd=round(sqrt((a_jump*b_jump)/(((a_jump+b_jump)^2)*(a_jump+b_jump+1))),5)
#Initialize Metropolis-Hastings:
acc=0
for (i in 1:S) {
  #Draw the Candidate:
  candidate_theta=rbeta(1,a_jump,b_jump)
  #Evaluate the posteriors:
  p=mult_prob(theta)
  p_new=mult_prob(candidate_theta)
  #Since the Jumping Distribution is not symmetric, we have to include it in the ratio r:
  q_prior=dbeta(theta,((candidate_theta)/(1-candidate_theta)),1,log=TRUE)
  q_candidate=dbeta(candidate_theta,(theta/(1-theta)),1,log=TRUE)
  posterior2=dmultinom(Y,prob=p_new,log=TRUE)+dbeta(candidate_theta,a,b,log=TRUE)+q_prior
  posterior1=dmultinom(Y,prob=p,log=TRUE)+dbeta(theta,a,b,log=TRUE)+q_candidate
  r=posterior2-posterior1
  u=runif(1)
  logu=log(u)
  if(logu<r){
    theta=candidate_theta
    acc=acc+1
  }
  samples[i]=theta
  b_jump=a_jump*(1-theta)/theta
}
prop_accepted=acc/S
#Discard the first 5000 samples:
burn_in=5000
post_samples=samples[(burn_in+1):S]
#Summarizing the univariate posterior:
post_mean=mean(post_samples)
post_sd=sd(post_samples)
q=c(0.025,0.975)
temp=as.matrix(post_samples)
Q=apply(temp,2,quantile,q)
credible_range=Q[2]-Q[1]
```

Inspecting the convergence of the chain visually:





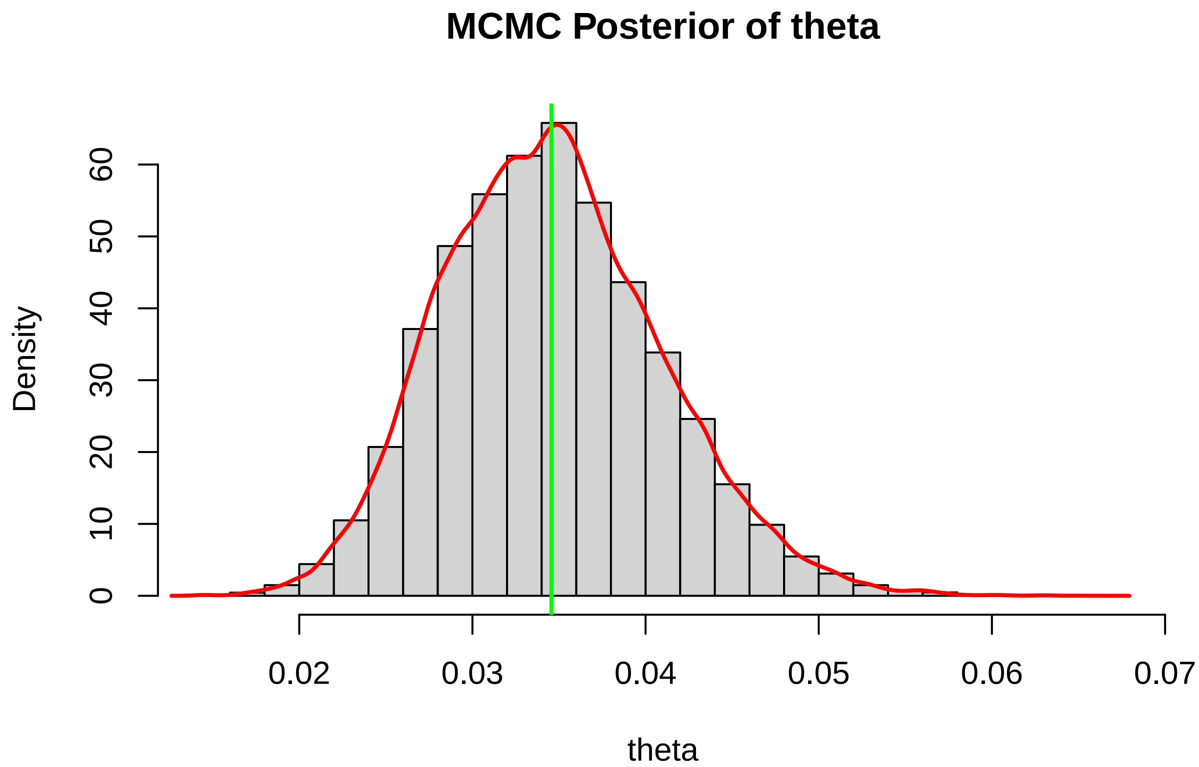



```
## [1] "The proportion of the samples accepted is: 35.045 %"
```

```
## [1] "For jumping sd: 0.0185 and initial value for theta: 0.03276042"
```

After 100,000 iterations, the markov chain seems to be stationary and has converged to the true posterior distribution. The ACF dies out exponentially, which confirms the stationarity of the chain and cuts-off completely after lag 10. The MCMC Algorithm accepted about 35% of the samples and has explored adequately the whole posterior distribution. The approximate posterior is summarized by approximating the posterior mean by the mean of the MCMC Samples, the posterior standard deviation by the standrad deviation of the MCMC samples, and the 95% credible sets by the empirical percentiles.

Summarizing the univariate posterior:




```
##                                  MCMC Posterior theta|X
## Posterior Mean                                 0.03456
## Posterior SD                                   0.00628
## 2.5% Credible %-ile                            0.02335
## 97.5% Credible %-ile                           0.04773
## 95% Credible Interval Range                    0.02438
```

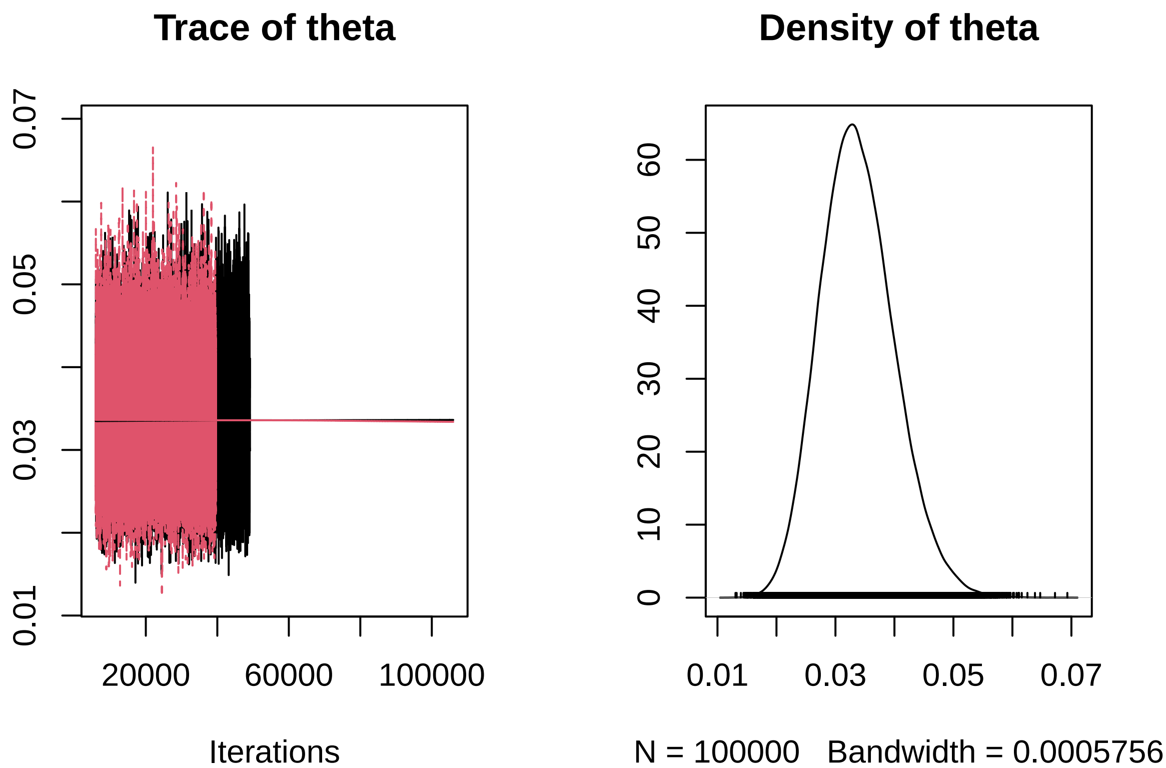
Given the data and the prior, our best estimate about the parameter θ is the posterior mean, $\mathbb{E}[\theta|X] = 0.03456$. The 95% credible interval is: $0.02335 \leq \theta|X \leq 0.04773$

MCMC Sampling in JAGS

Below we compare the results of the customized MCMC sampling algorithm to the ones obtained using the Bayesian Inference software JAGS:

```
## Linked to JAGS 4.3.1
```

```
## Loaded modules: basemod,bugs
```




```
##
## Iterations = 6001:106000
## Thinning interval = 1
## Number of chains = 2
## Sample size per chain = 100000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean             SD       Naive SE Time-series SE
##     0.03382190     0.00623736     0.00001395     0.00001807
##
## 2. Quantiles for each variable:
##
##    2.5%     25%     50%     75%   97.5%
## 0.02264 0.02944 0.03346 0.03781 0.04703
```

The results are very close but not the same.

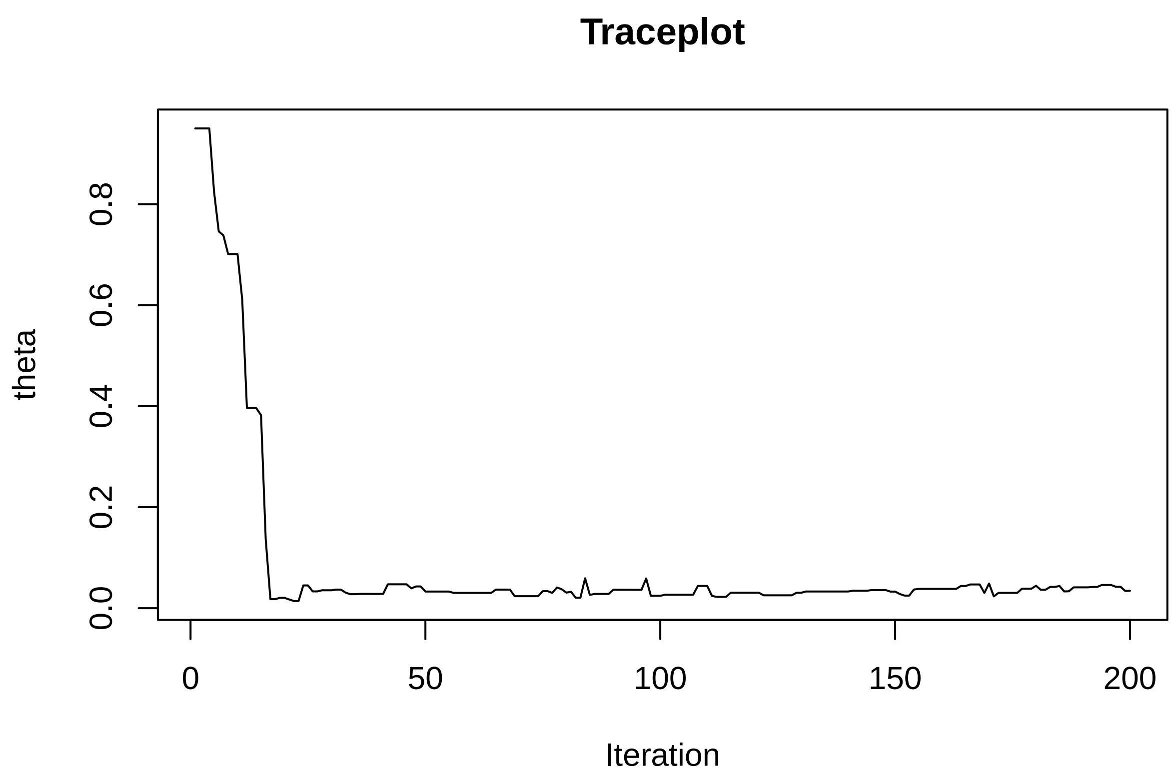
Effect of the jumping standard deviation to the convergence of the algorithm:

For a jumping distribution $\theta^* \sim q(\theta^*|\theta_{j-1}) = Beta(a_{jump}, b_{jump})$, the standard deviation is $SD_{jump} = \frac{a_{jump}b_{jump}}{(a_{jump}+b_{jump})^2(a_{jump}+b_{jump}+1)}$. Decreasing the parameter a, increases the jumping standard deviation. For high standard deviation, the percentage of the samples proposed that are accepted decreases. For instance, if we set $a_{jump} = 0.5$, then the jumping standard deviation increases form 0.0185 (the corresponing value for hyperparameter a=3) to 0.04414, and the percentage of the samples that are accepted drops from 35 % to 14.2 %. As a result, the convergence of the MCMC sampling algorithm is delayed and we might need more iterations to explore the whole posterior. The opposite is true if the jumping standard deviation is very low. In this case, the MCMC algorithm will accept all the proposed samples and will get stuck only to one territory of the posterior distribution. Thus, very low jumping standard deviations will impede convergence to the true posterior.

Effect of the initial value of theta to the convergence of the algorithm:

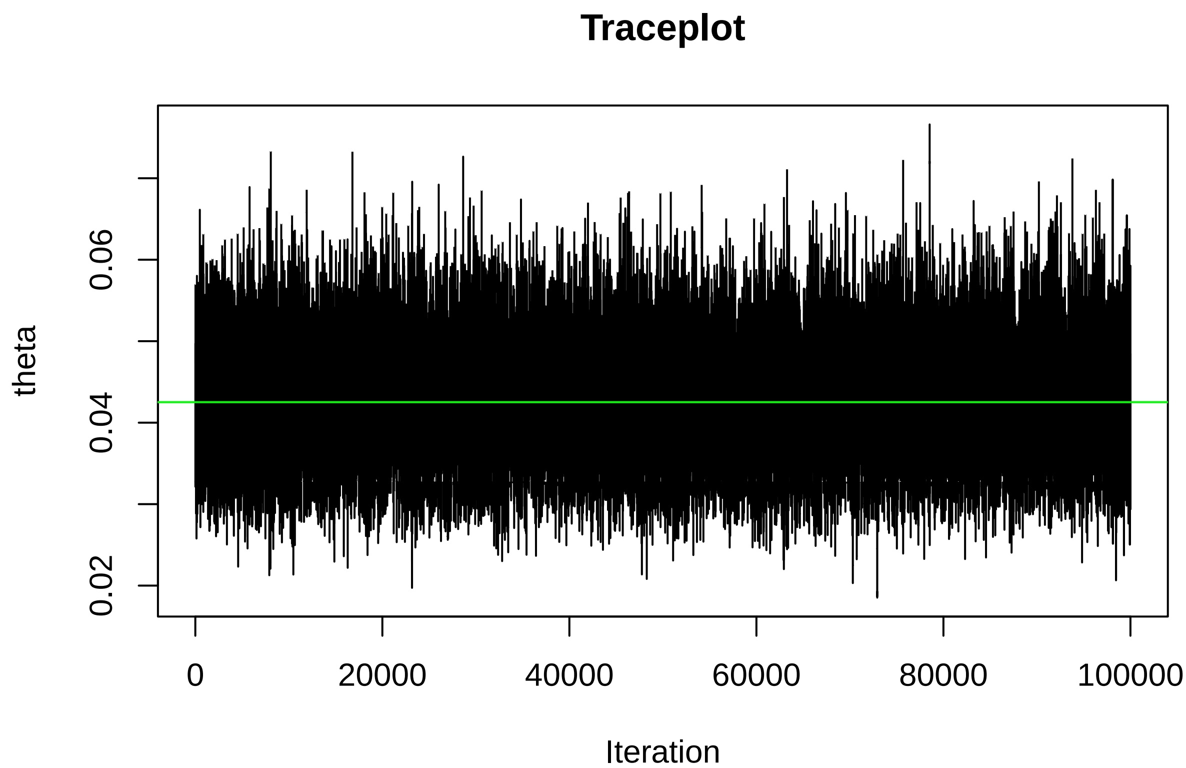
The Metropolis-Hastings sampling algorithm does not display any sensitivity to the initial value of θ. We choose to initiate θ from its maximum likelihood estimate to let the data inform the agent about the starting point. Regardless of the initial value, the agent will converge to the true posterior in a matter of iterations, as long as the posterior is well defined for the initial value of theta. In this case, for values of $\theta \geq 0.554$ the value of the multinomial density for the corresponding cell probabilities will be very close to 0 and the posterior will be ill-defined at those points. For this reason, we work with the log-posteriors when evaluating the acceptance ratio to avoid ill-conditioned circumstances. Using the log-posteriors the agent will converge to the true posterior distribution in a matter of iterations irrespective of the initial value. Below is a traceplot of the MCMC samples for initial value $\theta o = 0.95$

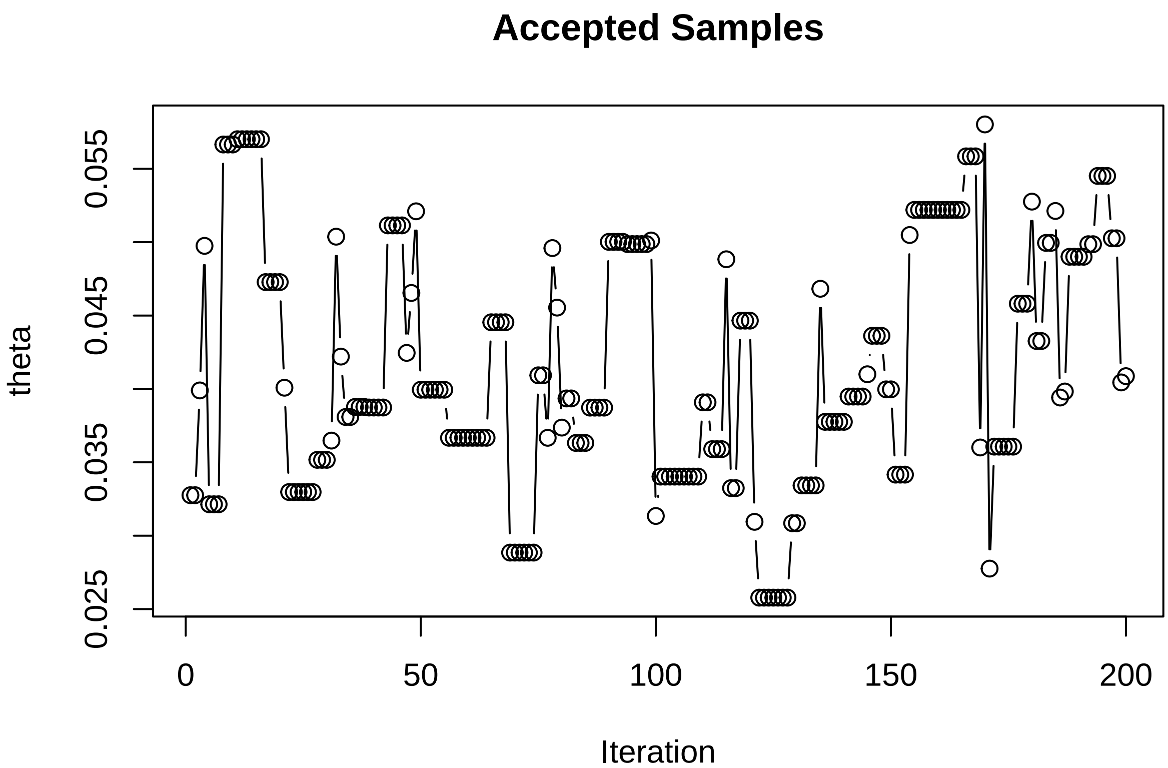
```
plot(samples[1:200],xlab="Iteration",ylab="theta",type="l",main="Traceplot")
```



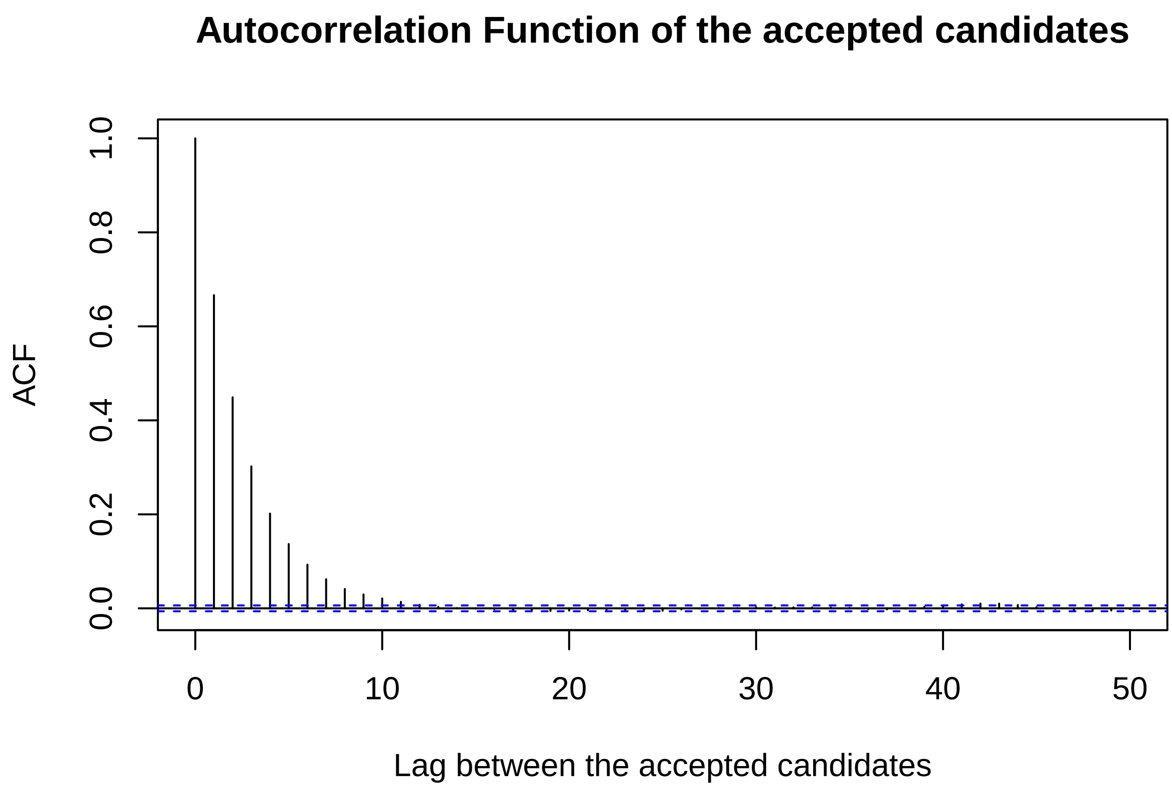
Prior sensitivity analysis:

To test whether the MCMC Sampling algorithm is sensitive to the choice of the prior, we perform the same analysis by placing an informative prior over θ. Assume that prior literature or research suggests that the parameter θ is most likely close to 0.8. Wishing to incorporate this piece of knowledge into our analysis, we place a Beta prior over θ centered at 0.8, that is $\theta \sim Beta(8, 2)$. The results are:

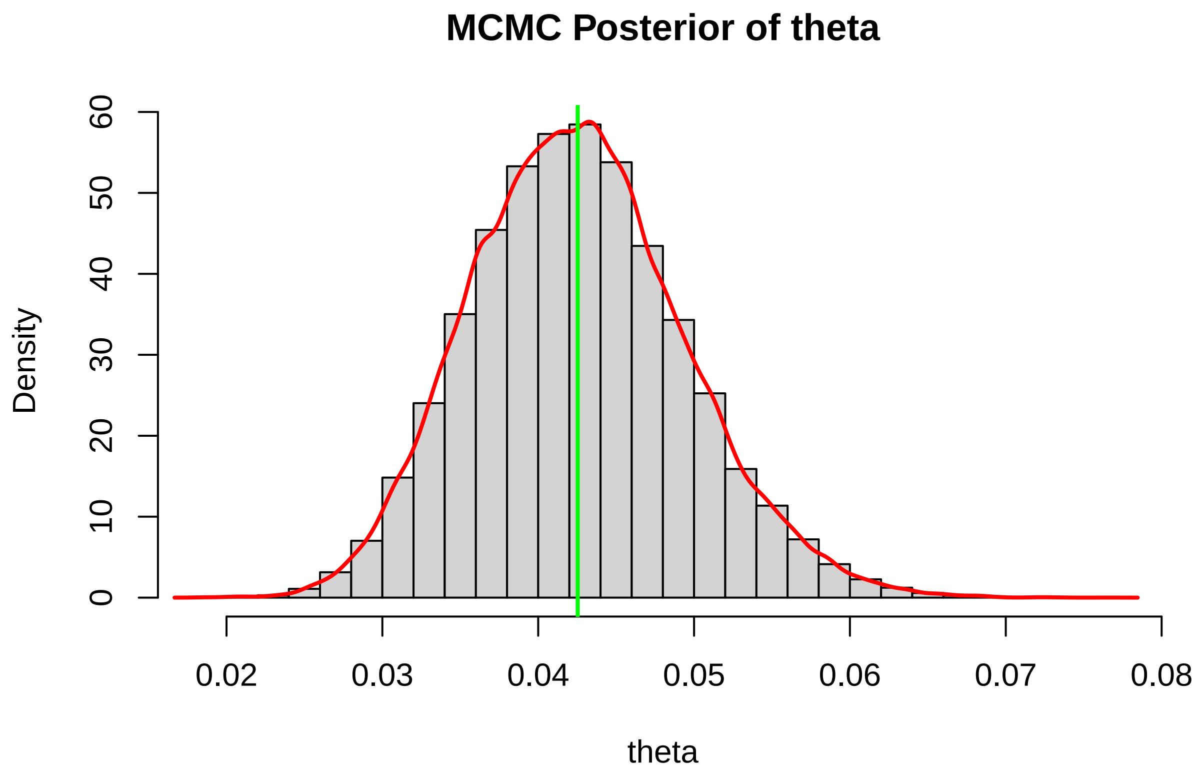

Traceplot
theta
0.02
0.04
0.06
0
20000
40000
60000
80000
100000
Iteration



Accepted Samples
theta
0.025
0.035
0.045
0.055
0
50
100
150
200
Iteration




```
## [1] "The proportion of the samples accepted is: 31.843 %"
## [1] "For jumping sd: 0.0185 and initial value for theta: 0.03276042"
```




```
##                                MCMC Posterior theta|X
## Posterior Mean                                0.04253
## Posterior SD                                  0.00682
## 2.5% Credible %-ile                           0.03013
## 97.5% Credible %-ile                          0.05676
## 95% Credible Interval Range                   0.02663
```

The results are somewhat sensitive to the choice of the prior. Given the data and the new prior, our best estimate about the parameter θ now changes to $\mathbb{E}[\theta|X] = 0.04253$. We see that the posterior mean is revised upwards. In addition, the posterior uncertainty over θ increases by 0.00054 and the 95% Credible Set widens to: $0.03013 \leq \theta|X \leq 0.05676$ with 95% probability. For an informative prior, the posterior mean tends to shrink towards the prior mean.