MCMC techniques

Ignacio Cascos

2019

Outline

5.1 **Markov chains (R12.1)**

5.2 **Metropolis-Hastings (R12.2)**

5.3 **Gibbs sampling (R12.3)**

5.1 Markov chains

A **discrete time Markov chain** $\{X_k\}_k$ is a sequence of rvs such that the distribution of X_k depens only on X_{k-1} and not on the previous rvs.

The Markov chain is **homogeneous** if the conditional distribution does not depend on time point. In such a case, the probabilities $P_{i,j} = P(X_k = j | X_{k-1} = i)$ are called **transition probabilities**.

A pmf π is **stationary distribution** of the Markov chain if for every state $\pi(j) = \sum P_{i,j}\pi(i)$.

Ergodic Markov chains have a unique stationary distribution and it will be reached as a limit distribution from any initial distribution.

MCMC methods consist in generating ergodic Markov chains whose limit distribution is a goal distribution. They do NOT produce sequences of independent observations!

5.2 Metropolis-Hastings

Our goal is to simulate from dmf f and we use an instrumental conditional dmf $g(\cdot|\cdot)$ from which it is simple to simulate. The support of g must contain that of f.

1. Set X_0 such that $f(X_0) > 0$ and $k = 0$.
2. Set $k = k + 1$. Generate $Y_k \sim g(\cdot|X_{k-1})$ and U random number in $(0, 1)$ independent.
3. If $U \leq \alpha(X_{k-1}, Y_k)$, set $X_k = Y_k$, otherwise $X_k = X_{k-1}$.
4. If $k < n$ go to Step 2., otherwise return $X_0, \ldots, X_n$.

$$\alpha(x, y) = \min\left\{\frac{f(y)g(x|y)}{f(x)g(y|x)}, 1\right\}$$

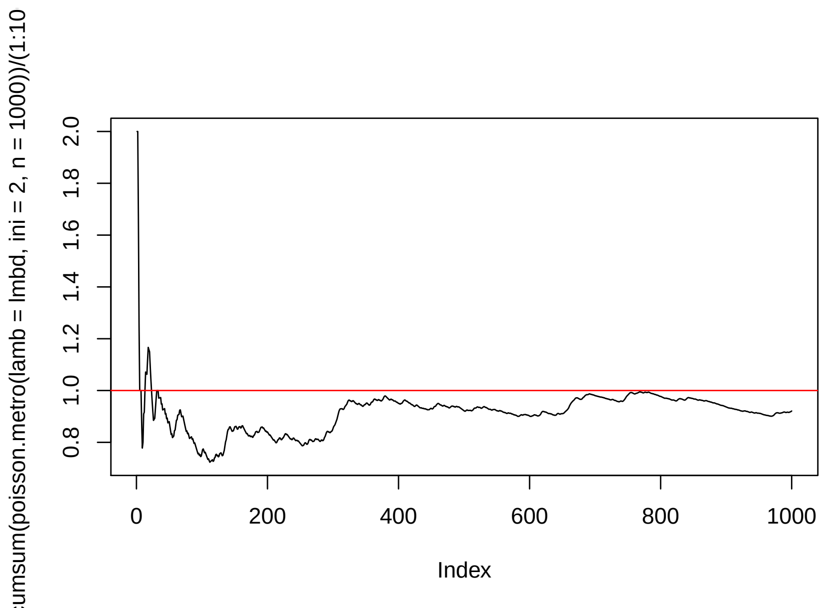
Metropolis-Hastings (Poisson)

In order to simulate a Poisson rv, we take as instrumental probability mass function (given $x \neq 0$), the one that assesses probability 1/2 to $x - 1$ and probability 1/2 to $x + 1$. If $x = 0$, then the probability of 0 is 1/2 and the probability of 1 is 1/2

```
rinst <- function(x) {
  if(x==0) return(sample(1,x=c(0,1)))
  else return(sample(1,x=c(x-1,x+1)))
}
```

```r
poisson.metro=function(lamb,ini,n){
 x <- c(ini)
 y <- rinst(ini)
 for(k in 2:n) {
  u <- runif(1)
  alpha<-lamb^y/factorial(y)/(lamb^x[k-1]/factorial(x[k-1]))
  if(u<=alpha) x <- c(x,y)
  else x <- c(x,x[k-1])
  y <- rinst(x[k])
 }
return(x)
}
```

```r
lmbd <- 1
plot(cumsum(poisson.metro(lamb=lmbd,ini=2,n=1000))/(1:1000),
     type="l")
abline(h=lmbd,col="red")
```



5.3 Gibbs sampling

Consider a random vector $\mathbf{X} = (X^{(1)}, \ldots, X^{(d)})$ with joint dmf f difficult to simulate from, but whose conditional densities of each component given the remaining componets $f_i(\cdot|x_k^{(1)}, \ldots, x_k^{(i-1)}, x_k^{(i+1)}, \ldots, x_k^{(d)})$ are easy to simulate from.

The **Gibbs sampler** is Metropolis-Hastings algorithm with $g(\cdot|\mathbf{X})$ having marginal densities as above. In such a case, $f(\mathbf{y})g(\mathbf{x}|\mathbf{y}) = f(\mathbf{x})f(\mathbf{y})$, so all candidates will be accepted.

1. Set $\mathbf{X}_0$ such that $f(\mathbf{X}_0) > 0$ and $k = 0$.
2. Set $k = k + 1$. For $i = 1, \ldots, d$, generate $X_k^{(i)}$ from $f_i(\cdot|X_k^{(1)}, \ldots, X_k^{(i-1)}, X_{k-1}^{(i+1)}, \ldots, X_{k-1}^{(d)})$.
3. If $k < n$ go to Step 2., otherwise return $\mathbf{X}_0, \ldots, \mathbf{X}_n$.

Example

If the random vector $(X, Y)^t$ follows a bivariate normal distribution with mean vector $(0, 0)^t$ and covariance matrix having 1s on the maing diagonal and ρs elsewhere (X and Y are standard normal random variables with correlation ρ)

$X \sim \mathrm{N}(\rho Y, \sqrt{1 - rho^2})$ and $Y \sim \mathrm{N}(\rho X, \sqrt{1 - \rho^2})$

```
set.seed(1) ; n <- 10^4 ; x <- c(0) ; y <- c(0)
rho <- 0.8 ; sigma <- sqrt(1 - rho^2)
for (i in 2:n) {
  x <- c(x,rnorm(1,mean=rho*y[i-1],sd=sigma))
  y <- c(y,rnorm(1,mean=rho*x[i],sd=sigma))
}
c(mean(x),mean(y))
```

```
## [1] -0.01986475 -0.01208011
```

```
c(sd(x),sd(y))
```

```
## [1] 0.9947670 0.9899634
```

```
cor(x,y)
```

```
## [1] 0.7984191
```

```
par(mfrow=c(2,2))
plot(x[1:100],type="l") ; acf(x)
s.norm <- rnorm(100) ; plot(s.norm,type="l") ; acf(s.norm)
```