

Bayesianische Variablenselektion

Volker Schmid

10. Juli 2017

Reversible Jump MCMC (RJMCMC)

Reversible Jump MCMC (RJMCMC)

Modellwahl/Variablenwahl mit unterschiedlichen Dimensionen

Vergleichen wir zwei unterschiedliche Modelle

1. $y = \alpha + \beta x + \epsilon$

2. $y = \alpha + \epsilon$

so lässt sich die Modellwahl auch als Variablenselektion mit Indikatorvariablen interpretieren. Im zweiten Modell ist die Indikatorvariable für β gleich 0.

Im MCMC-Algorithmus sind also identisch:

- ▶ $I = 1 \rightarrow I = 0$
- ▶ $\beta \rightarrow 0$
- ▶ Modell 1 \rightarrow Modell 2
- ▶ $\theta = (\alpha, \beta) \rightarrow \theta^* = (\alpha)$

Reversible Jump MCMC I

- ▶ RJMCMC nach Green (1995)
- ▶ Allgemein sind bei Reversible Jump verschiedene Parameterräume erlaubt
- ▶ Zwischen den Parameterräumen müssen Abbildungen (reversible jumps) möglich sein
- ▶ Im obigen Beispiel:

death step

$$(\alpha, \beta) = (\alpha)$$

birth step

$$(\alpha) \rightarrow (\alpha, \beta) \text{ mit } \beta \sim \text{prior}$$

- ▶ Der Modellwechsel geht in die Akzeptanzwahrscheinlichkeit ein (auch *Metropolis-Hastings-Green-Wahrscheinlichkeit*).
- ▶ Mit θ alter Zustand und θ^* Vorschlag

Reversible Jump MCMC II

$$\alpha = \frac{f(y|\theta^*)}{f(y|\theta)} \frac{p(\theta^*)}{p(\theta)} \frac{q(\theta|\theta^*)}{q(\theta^*|\theta)} |J|$$

wobei J die Jacobi-Matrix für den deterministischen Übergang von $\theta \rightarrow \theta^*$ ist

RJMCMC Beispiel I

```
g.predict<-function(Mod, X)
{
  j      = Mod[2]
  beta0 = Mod[3]
  beta1 = Mod[4]
  beta2 = Mod[5]

  P = 0*X + beta0
  if (j >= 2) {P = P + X*beta1}
  if (j >= 3) {P = P + (X^2)*beta2}

  return(P)
}
```

RJMCMC Beispiel II

```
g.perterb<-function(M=c(-Inf, 3, 0, 0, 0), Qsd=c(0, 0, 0.1, 0.1, 0.1))
{
  # unpacking hte parameters
  LL      = M[1]
  j       = M[2]
  #beta0 = M[3]
  #beta1 = M[4]
  #beta2 = M[5]
  x       = data[,1]
  y       = data[,2]

  ORDER = sample(3:(3+j-1), j)

  for (i in ORDER)
  {
    M.prime = M
    M.prime[i] = M.prime[i] + rnorm(1, mean = 0, sd= Qsd[i]) # main
    P = g.predict(M.prime, x) # ge
```


RJMCMC Beispiel Ergebnisse

```
g.rjMCMC(Ndat = 20)
```

