Flood Frequency Hydrology

MCMC Bayesian analysis in R

Alberto Viglione[1]

[1]Department of Environment, Land and Infrastructure Engineering (DIATI)
Politecnico di Torino, Italy

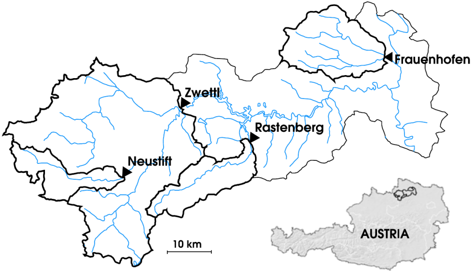
IUGG-2019, Montréal, July 2019

Presentation and R code are available at:

URL: https://diatibox.polito.it/s/4LJdpPtIuHRq7pE

PSW: rinhydrology

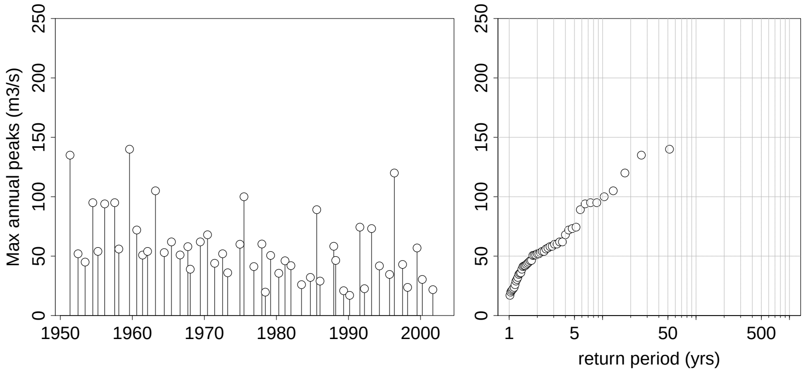
Example: Q_{100} for the Kamp at Zwettl



from Viglione et al.(2010)

Example: Q_{100} for the Kamp at Zwettl

Given the maximum annual peak discharges of the river Kamp at Zwettl (622 km^2) how much is the 100-year peak discharge?



Distribution functions

In hydrology many probability distributions have been adopted to describe flood peaks. Here we use the Generalised Extreme Value distribution (GEV) which is:

$$f_X(x|\boldsymbol{\theta}) = \frac{1}{\theta_2}\left[1 - \frac{\theta_3(x-\theta_1)}{\theta_2}\right]^{1/\theta_3 - 1} \exp\left\{-\left[1 - \frac{\theta_3(x-\theta_1)}{\theta_2}\right]^{1/\theta_3}\right\}$$

$$F_X(x|\boldsymbol{\theta}) = \exp\left\{-\left[1 - \frac{\theta_3(x-\theta_1)}{\theta_2}\right]^{1/\theta_3}\right\}$$

$$x(F|\boldsymbol{\theta}) = \theta_1 + \frac{\theta_2}{\theta_3}\left[1 - (-\ln F)^{\theta_3}\right]$$

therefore

$$Q_T = \theta_1 + \frac{\theta_2}{\theta_3}\left[1 - \left(-\ln\left(1 - \frac{1}{T}\right)\right)^{\theta_3}\right]$$

Parameter estimation

To estimate $\boldsymbol{\theta} = (\theta_1, \theta_2, \theta_3)$, many methods exist such as:

- Method of moments: after deriving equations that relate the population moments (mean, variance, skewness, ...) to its parameters, use the sample moments of the data in the equations
- Method of L-moments: same thing but with L-moments
- Maximum Likelihood method: after defining as likelihood the joint density function of the observations, find the parameters that maximise it
- Bayesian inference: estimate the probability density function of the parameters from the observations and prior knowledge about them

Bayesian inference

The Bayes's Theorem

$$p(\boldsymbol{\theta}|D) = \frac{\ell(D|\boldsymbol{\theta})\pi(\boldsymbol{\theta})}{\int_{\text{all}\boldsymbol{\theta}} \ell(D|\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}} \propto \ell(D|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$$

states that the posterior distribution of $\boldsymbol{\theta}$ given data D is equal to the product of the likelihood of observing D given $\boldsymbol{\theta}$ and the prior distribution of $\boldsymbol{\theta}$ divided by the integrated likelihood.

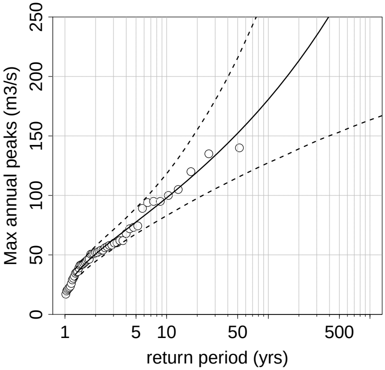
The second formulation gives the posterior distribution only up to a multiplicative constant, but often this is enough, and avoids the difficulty of evaluating the integrated likelihood, also called the normalizing constant in this context.

Example: Q_{100} for the Kamp at Zwettl

By writing

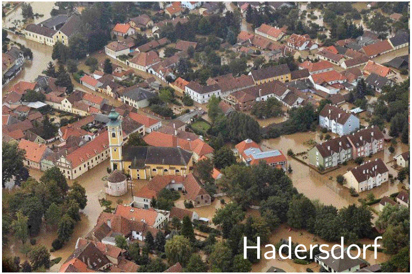
$$\ell(D|\boldsymbol{\theta}) = \prod_{i=1}^{s} f_X(x_i|\boldsymbol{\theta}) \qquad \pi(\boldsymbol{\theta}) \propto 1/\theta_2$$

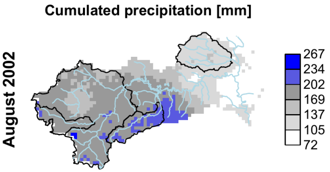
where the sample of annual discharge maxima systematically recorded is $x_1, x_2, \ldots, x_s$ (in our case, s=50 years), one gets, after applying the MCMC method, the posterior distribution mean and its (e.g., 90%) credible intervals



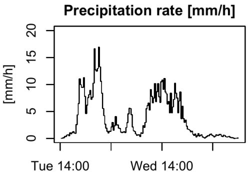
Ok! We can now read Q_{100} from the graph: it's say 175 m^3/s $\pm$50 m^3/s (well... not so symmetrically)

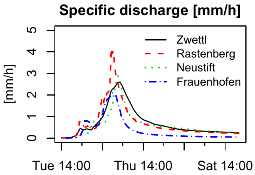
2002 Flood Event!





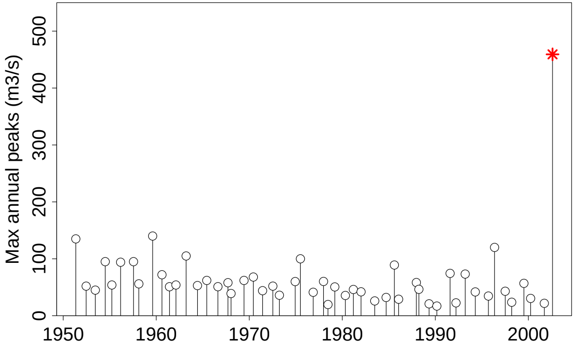
from Viglione et al.(2010)





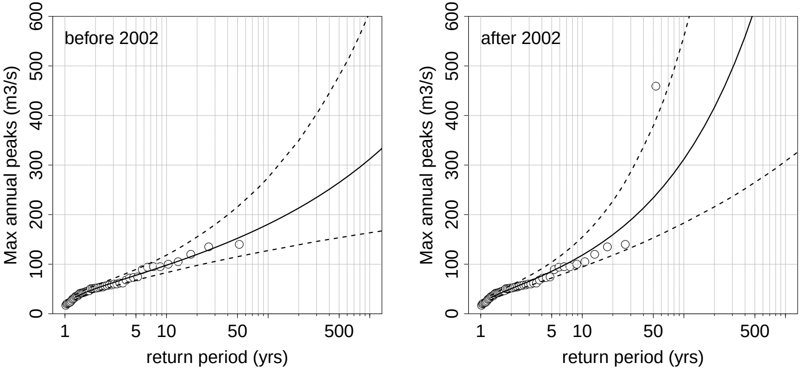
2002 Flood Event!

After having observed the huge flood, how much is the 100-year peak discharge? And what's the return period of the 2002 event?



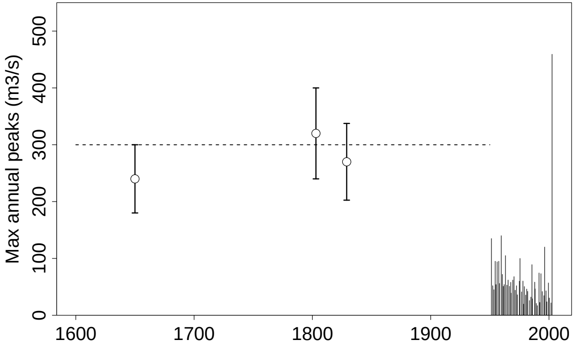
Example: Q_{100} for the Kamp at Zwettl

If I redo everything with the additional 2002 event, how much is the 100-year peak discharge? And what's the return period of the 2002 event?



Flood Frequency Hydrology: temporal expansion

Three major historical floods are documented in the region (Viglione et al., 2013; Wiesbauer, 2007)

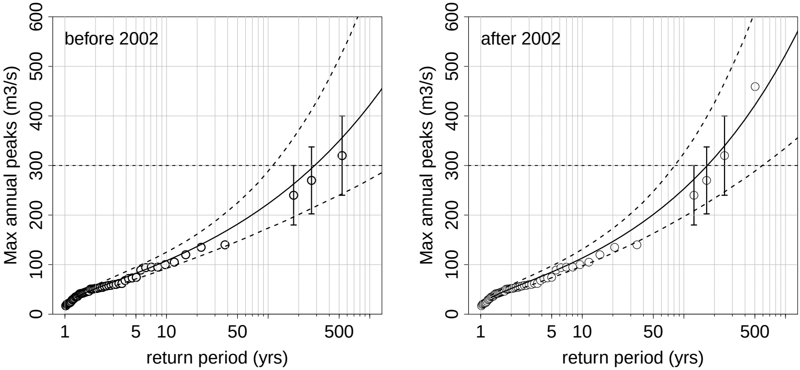


Flood Frequency Hydrology: temporal expansion

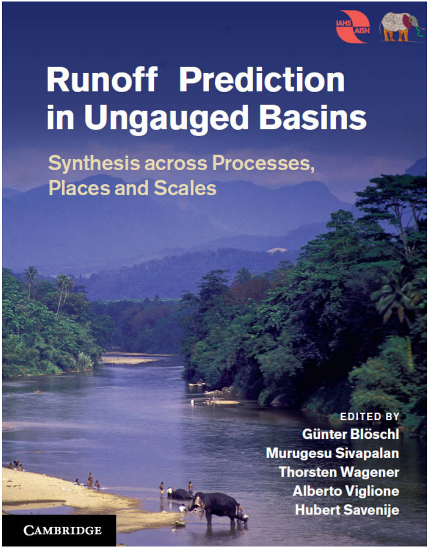
By writing (see Stedinger and Cohn, 1986)

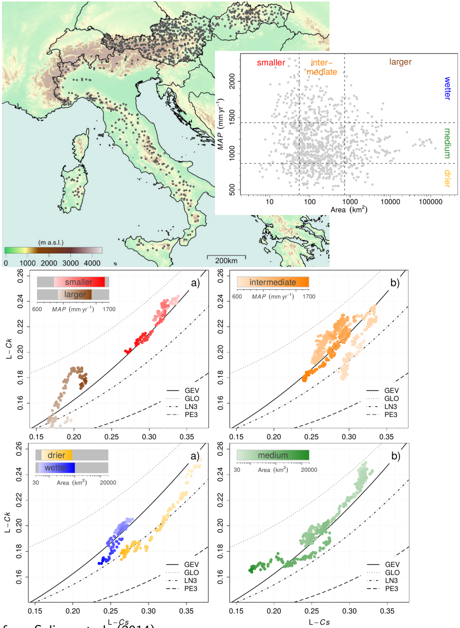
$$\ell(D|\boldsymbol{\theta}) = \prod_{i=1}^{s} f_X(x_i|\boldsymbol{\theta}) \binom{h}{k} F_X(X_0|\boldsymbol{\theta})^{(h-k)} \left\{ \prod_{j=1}^{k} [F_X(y_{Uj}|\boldsymbol{\theta}) - F_X(y_{Lj}|\boldsymbol{\theta})] \right\}$$

where in this case k=3, h=350 and X_0=300 m^3/s, one gets...



Flood Frequency Hydrology: spatial expansion





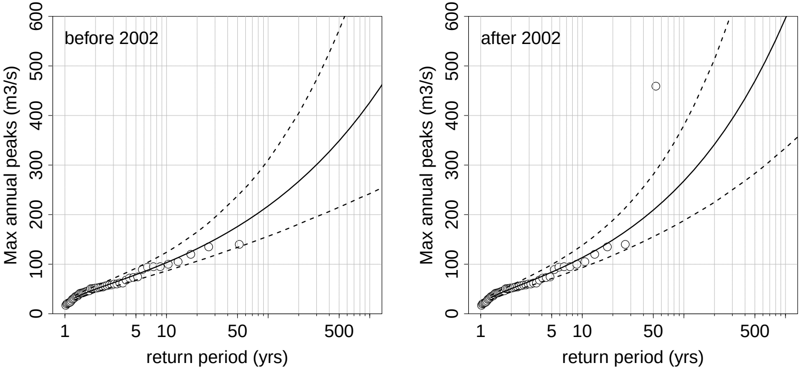
from Salinas et al. (2014)

Flood Frequency Hydrology: spatial expansion

By writing

$$\pi(\boldsymbol{\theta}) \propto \frac{1}{\theta_2}\mathcal{N}\left(\theta_3 | \mu_{\theta_3}, \sigma^2_{\theta_3}\right)$$

where regional data are used for the guessing reasonbable values for the GEV shape parameter θ_3, i.e., $\mu_{\theta_3} = -0.3$ and $\sigma_{\theta_3} = 0.1$, one gets...



Flood Frequency Hydrology: temporal + spatial expansion

I combine the two sources of information through the Bayes' theorem:

$$p(\boldsymbol{\theta}|D) \propto \ell(D|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$$

where the systematic data and historic information define the likelihood:

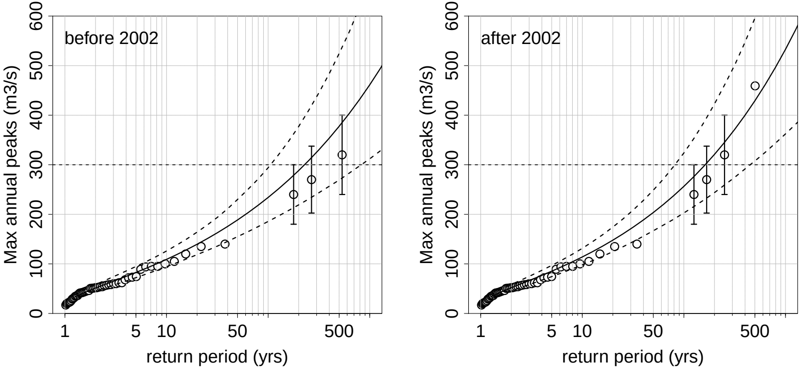
$$\ell(D|\boldsymbol{\theta}) = \prod_{i=1}^{s} f_X(x_i|\boldsymbol{\theta}) \binom{h}{k} F_X(X_0|\boldsymbol{\theta})^{(h-k)} \left\{ \prod_{j=1}^{k} \left[F_X(y_{Uj}|\boldsymbol{\theta}) - F_X(y_{Lj}|\boldsymbol{\theta}) \right] \right\}$$

and the regional information on the shape parameter of the GEV distribution goes into the prior distribution of the parameters:

$$\pi(\boldsymbol{\theta}) \propto \frac{1}{\theta_2} \mathcal{N}\left(\theta_3|\mu_{\theta_3}, \sigma^2_{\theta_3}\right)$$

Flood Frequency Hydrology: temporal + spatial expansion

If I combine the two sources of information, how much is the 100-year peak discharge? Well, say 250±50 m^3/s And what's the return period of the 2002 event? Large uncertainty remains but...



Viglione, A., R. Merz, J. S. Salinas, and G. Blöschl (2013), Flood frequency hydrology: 3. A Bayesian analysis. *Water Resources Research* **49**(2), 675-692, doi:10.1029/2011WR010782.

How do we do this in R?

URL: https://diatibox.polito.it/s/4LJdpPtIuHRq7pE
PSW: rinhydrology

code: MCMC_FFH_codes20190709.R

Bayesian inference

As already discussed, the Bayes's Theorem can be written as

$$p(\boldsymbol{\theta}|D) \propto \ell(D|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$$

that gives the posterior distribution only up to a multiplicative constant.
Among the advantages over other parameter estimation methods:

- $\ell(D|\boldsymbol{\theta})$ can be easily defined even for complex models
- $\pi(\boldsymbol{\theta})$ provides a way of incorporating external information (outside the current data set)

Sampling from $p(\boldsymbol{\theta}|D)$ can be performed through Markov chain Monte Carlo (MCMC) methods, which are based on constructing a Markov chain that has the desired distribution as its equilibrium distribution.

MCMC: Metropolis-Hastings algorithm

The Metropolis-Hastings algorithm is a Markov chain Monte Carlo (MCMC) method for obtaining a sequence of random samples from any probability distribution (a.k.a. the target distribution), provided you can compute the value of a function that is proportional to its density.

This sequence can be used to approximate the distribution (e.g., to generate the histogram of the target distribution), or to compute an integral (such as its expected value).

In our case, the target distribution is $p(\boldsymbol{\theta}|D)$, while $\ell(D|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$ is the function proportional to its density.

MCMC: Metropolis-Hastings algorithm

Let $f(x)$ be a function that is proportional to the desired target distribution $p(x)$.

- Choose an arbitrary point x_0 to be the first sample, and choose an arbitrary probability density $g(x'|x)$ (a.k.a. the proposal density) that suggests a candidate for the next sample value x', given the previous sample value x.
- For each iteration t:
  1. Generate a candidate x' for the next sample by picking from the distribution $g(x'|x_t)$.
  2. Calculate the acceptance ratio

     $$\alpha = \frac{f(x')}{f(x_t)} \frac{g(x_t|x')}{g(x'|x_t)}$$

     which will be used to decide whether to accept or reject the candidate.
  3. If $\alpha \geq 1$ accept the candidate by setting $x_{t+1} = x'$. Otherwise, accept the candidate with probability α. If the candidate is rejected, set $x_{t+1} = x_t$ instead.

MCMC: Metropolis-Hastings algorithm

The most common choice is for a symmetric proposal density g, i.e. $g(x|y) = g(y|x)$, in which case the algorithm is called Metropolis algorithm, and $\alpha = f(x')/f(x_t) = p(x')/p(x_t)$.

Why? in the end you want the condition $p(x_t) \cdot Pr[x_t \rightarrow x'] = p(x') \cdot Pr[x' \rightarrow x_t]$, to mantain equilibrium, so, if $p(x_t) > p(x')$ you may choose $Pr[x' \rightarrow x_t] = 1$ and $Pr[x_t \rightarrow x'] = p(x')/p(x_t) = \alpha$ and vice-versa

The variance of the proposal density has to be tuned because too small and too large variances would lead to a slow convergence of the chain.

Since the resulting samples are correlated, we have to throw away the majority of samples and only take every n-th sample, for some value of n (typically determined by examining the autocorrelation between adjacent samples) which defines the thinning period.

Since the initial samples may follow a very different distribution than $p(x)$, we have to throw them away by setting a burn-in period.

Noninformative Prior Distributions

There have been many efforts to find priors that carry no information, or noninformative priors. In general, this has turned out to be a modern version of the Philosopher's Stone. There are some very simple problems for which there are agreed *reference* priors. One example is the normal mean problem, for which a flat prior

$$p(\theta) \propto 1$$

is often used. This is an improper prior, i.e. it does not integrate up to 1, nevertheless the resulting posterior distribution is proper.

Improper noninformative priors can lead to paradoxes and strange behavior and should be used with extreme caution. The current trend in applied Bayesian statistical work is towards informative and, if necessary, spread out but proper prior distributions.

Noninformative Prior Distributions

The Jeffreys prior is a noninformative prior distribution for a parameter space, which is invariant under reparameterization.

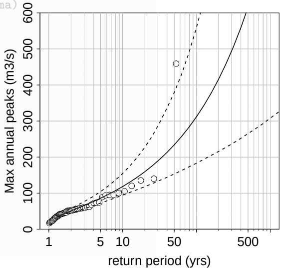
- For the Gaussian distribution with known variance, the Jeffreys prior for the mean is $p(\mu) \propto 1$, which is translation-invariant corresponding to no information about location.
- For the Gaussian distribution with known mean the Jeffreys prior for the standard deviation is $p(\sigma) \propto 1/\sigma$, or equivalently $p(\log \sigma) \propto 1$, which is scale-invariant corresponding to no information about scale.
- For the Gaussian distribution with unknown mean and variance, Jeffreys' advice is to assume that μ and σ are independent apriori and use $p(\mu, \sigma) \propto 1/\sigma$, which is translation-scale invariant.

Northrop and Attalides (2015) demonstrate that for the GEV the Jeffreys prior does not yield a proper posterior while independent uniform priors do: i.e., $\pi(\boldsymbol{\theta}) \propto 1/\theta_2$.

MCMC for the GEV distribution in R

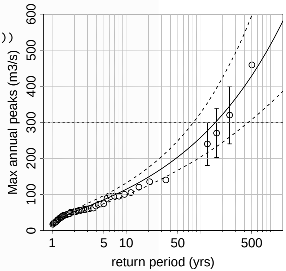
URL: https://diatibox.polito.it/s/4LJdpPtIuHRq7pE
PSW: rinhydrology

```
> MCMC01 <- function (x, N=1000, theta0=c(1,0,-.5), pseudo_var=c(1,1,1),
...                        burnin=100) {
...   # N = final sample size (i.e., excluding the burn-in length)
...   # theta0 = starting point of your Metropolis chain containing (mu0, log(sigma0), xi0)
...   # pseudo_var = variance for the normal that is used as the proposal distribution for random-walk
...   #               Metropolis (independent sampling)
...   # burnin = number specified will be the number of initial samples chucked
...   require(MASS)   #requires package MASS for normal sampling
...   thetas <- theta0
...   for (i in 2:(burnin+N)) {
...    loglikelihood0 <- sum(log(dGEV(x, mu=theta0[1],
...                                  sigma=exp(theta0[2]), xi=theta0[3])))
...    logprior0 <- 0  # because 1/sigma corresponds to uniform distr of the log(sigma)
...    logtarget0 <- loglikelihood0 + logprior0
...    if(is.nan(logtarget0)) logtarget0 <- -10000000
...    prop <- mvrnorm(n=1, mu=theta0, Sigma=diag(pseudo_var))
...    loglikelihood1 <- sum(log(dGEV(x, mu=prop[1],
...                                  sigma=exp(prop[2]), xi=prop[3])))
...    logprior1 <- 0
...    logtarget1 <- loglikelihood1 + logprior1
...    if(is.nan(logtarget1)) logtarget1 <- -10000000
...    if (runif(1) < min(1, exp(logtarget1 - logtarget0))) {
...     theta0 <- prop
...    }
...    thetas <- rbind(thetas, theta0)
...   }
...   thetas[(burnin+1):(N+burnin),]
... }
```



MCMC for GEV in R: temporal + spatial expansion

```
> MCMC04 <- function (x, prior_t3=c(0,10),
...                      inthist, suphist, thres, nbelow,
...                      N=1000, theta0=c(1,0,-.5), pseudo_var=c(1,1,1), burnin=100) {
...  # prior_t3 = parameters of normal distribution for theta3 (shape of GEV)
...  # infhist = lower limits for historic discharges
...  # suphist = upper limits for historic discharges
...  # thres = perception threshold for historic period
...  # nbelow = period (in years) over which the threshold has not been exceeded
...  #         except for the historical data
...  require(MASS)  #requires package MASS for normal sampling
...  thetas <- theta0
...  for (i in 2:(burnin+N)) {
...   loglikelihood0 <- sum(log(dGEV(x, mu=theta0[1], sigma=exp(theta0[2]), xi=theta0[3])))
...   loglikelihood0hist <- sum((nbelow - 1) * log(pGEV(thres, mu=theta0[1],
...                                       sigma=exp(theta0[2]), xi=theta0[3]))) +
...                     sum(log(pGEV(suphist, mu=theta0[1], sigma=exp(theta0[2]), xi=theta0[3]) -
...                             pGEV(infhist, mu=theta0[1], sigma=exp(theta0[2]), xi=theta0[3])))
...   logprior0 <- log(dnorm(theta0[3], mean=prior_t3[1], sd=prior_t3[2]))
...   logtarget0 <- loglikelihood0 + loglikelihood0hist + logprior0
...   if(is.nan(logtarget0)) logtarget0 <- -10000000
...   prop <- mvrnorm(n=1, mu=theta0, Sigma=diag(pseudo_var))
...   loglikelihood1 <- sum(log(dGEV(x, mu=prop[1], sigma=exp(prop[2]), xi=prop[3])))
...   loglikelihood1hist <- sum((nbelow - 1) *
...                         log(pGEV(thres, mu=prop[1],
...                                  sigma=exp(prop[2]), xi=prop[3]))) +
...                     sum(log(pGEV(suphist, mu=prop[1],
...                                  sigma=exp(prop[2]), xi=prop[3]) -
...                             pGEV(infhist, mu=prop[1],
...                                  sigma=exp(prop[2]), xi=prop[3])))
...   logprior1 <- log(dnorm(prop[3], mean=prior_t3[1], sd=prior_t3[2]))
...   logtarget1 <- loglikelihood1 + loglikelihood1hist + logprior1
...   if(is.nan(logtarget1)) logtarget1 <- -10000000
...   if (runif(1) < min(1, exp(logtarget1 - logtarget0))) {
...    theta0 <- prop
...   }
...   thetas <- rbind(thetas, theta0)
...  }
...  thetas[(burnin+1):(N+burnin),]
... }
```



MCMC in R

There are many (MANY!) R packages for Bayesian inference through MCMC algorithms. Search for "CRAN Task View: Bayesian Inference" on the web (131 packages are listed there!). Among the ones that I have tried, or heard of, are:

- **mcmc** Markov Chain Monte Carlo with the random-walk Metropolis algorithm
- **MCMCpack** Markov Chain Monte Carlo (MCMC) Package with algorithms for a wide range of models
- **MCMCglmm** MCMC Generalised Linear Mixed Models
- **R2WinBUGS** Running 'WinBUGS' (`http://www.mrc-bsu.cam.ac.uk/software/bugs/`) and 'OpenBUGS' (`http://www.openbugs.net/w/FrontPage`) from R or S-PLUS
- **R2jags** Using R to Run 'JAGS' (`http://mcmc-jags.sourceforge.net/`)
- **rstan** R Interface to 'Stan' (`https://mc-stan.org/`)
- **dream** DiffeRential Evolution Adaptive Metropolis: efficient global MCMC even in high-dimensional spaces
- **extRemes** Extreme Value Analysis, which includes Bayesian inference with MCMC
- **nsRFA** Non-Supervised Regional Frequency Analysis, which includes Bayesian inference with MCMC