Project 2

Computer Intensive Statistical Methods

Erling Fause Steen og Christian Oppegård Moen

08 03 2022

Contents

Introduction

The Tokyo rainfall dataset contain the amount of rainfall for each of the 366 days (including February 29.) for several years. We will consider a portion of this dataset, specifically from $1951 - 1989$, such that for each day $t \neq 60$ we have $n_t = 39$ observations and for $t = 60$, February 39, we have $n_t = 10$ observation. For each day we have the response $y_t = 0, 1, 2, ..., n_t$ being the amount of times the rainfall exceeded 1mm over the given period, given by

$$y_t|\tau_t \sim \text{Bin}(n_t, \pi(\tau_t)), \quad \pi(\tau_t) = \frac{\exp(\tau_t)}{1+\exp(\tau_t)} = \frac{1}{1+\exp(-\tau_t)}. \tag{1}$$

Here, $\pi(\tau_t)$ is the probability of rainfall exceeding 1mm and τ_t is the logit probability of exceedence. For this project we assume conditional independence among the $y_t|\tau_t$ $\forall t = 1, 2, ..., 366$.

We will apply a Bayesian hierarchical model to the dataset, using a random walk of order 1 (RW1) to model the trend. For the model we will implement a Markov chain Monte Carlo (MCMC) sampler for the posterior using Metropolis-Hastings (MH) and Gibbs steps for specific parameters. Then we will investigate the accuracy and computational speed of the implementation compared to the built in method `INLA` in R.

Problem 1

a) Display

In Figure 1 we see the number of times the rainfall has exceeded 1mm. There seem to be fewer days in the start of the year and in the end of the year with an amount of rainfall exceeding 1 mm. This is in January and December. The number of days steadily increases until the begining of the summer which seems to be the period with the most days with an amount of rainfall over 1 mm. Then, the amount of days decreases during july and august before increasing during the autumn. There also seem to be fluctuations on a daily basis from the just mentiod trend in the data. The red dot is the observation for February 29.

```
load("./rain.rda")
## Plotting the data
ggplot(data = rain, mapping = aes(x = day, y = n.rain)) + geom_line() + xlab("Day") +
    ylab("Number of days with more than 1mm rain") + geom_point(aes(x = day[60],
    y = n.rain[60]), colour = "red")
```

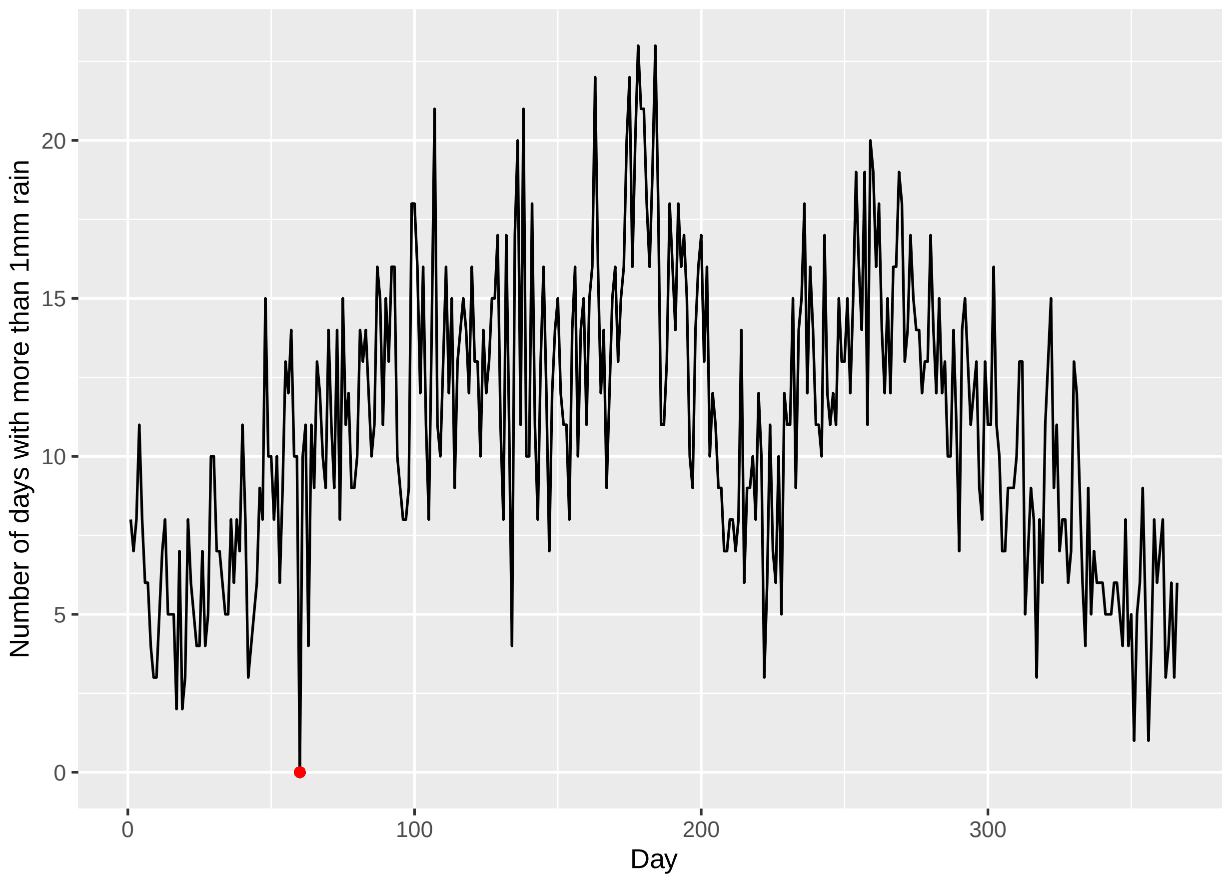


Figure 1: The Tokyo Rainfall dataset

b) Likelihood

The likelihood of Equation (1) is given by

$$
\begin{aligned}
L(\pi(\boldsymbol{\tau})) &= \prod_{i=1}^{T} \begin{pmatrix} n_t \\ y_t \end{pmatrix} \pi(\tau_t)^{y_t}(1-\pi(\tau_t)^{n_t-y_t} \\
&\propto \prod_{i=1}^{T} \pi(\tau_t)^{y_t}(1-\pi(\tau_t)^{n_t-y_t} \\
&= \prod_{t=1}^{T} \left(\frac{\exp(\tau_t)}{1+\exp(\tau_t)}\right)^{y_t} \left(1-\frac{\exp(\tau_t)}{1+\exp(\tau_t)}\right)^{n_t-y_t},
\end{aligned}
$$

where $\boldsymbol{\tau} = (\tau_1, ..., \tau_T)$, $y_t = 1, 2, ..., 39$ and $n_t = 39$ for $t \neq 60$, and $y_t = 1, 2, ..., 10$ and $n_t = 10$ for $t \neq 60$.

c) Posterior

As briefly mentioned in the introduction we need the posterior $P(\sigma^2|\tau, y)$ for the Gibbs step in our implementation, given by

$$\begin{aligned} P(\sigma^2|\boldsymbol{\tau}, \boldsymbol{y}) &= \frac{P(\sigma_u^2, \boldsymbol{\tau}, \boldsymbol{y})}{P(\boldsymbol{\tau}, \boldsymbol{y})} \\ &\propto P(\boldsymbol{y}|\sigma_u^2, \boldsymbol{\tau})P(\sigma_u^2, \boldsymbol{\tau}) \\ &= P(\boldsymbol{y}|\sigma_u^2, \boldsymbol{\tau})P(\tau|\sigma_u^2)P(\sigma_u^2), \end{aligned}$$

where $\boldsymbol{y} = (y_1, ..., y_T)^T$, $\boldsymbol{\tau} = (\tau_1, ..., \tau_T)$ and $P(\boldsymbol{y}|\sigma_u^2, \boldsymbol{\tau}) = L(\pi(\boldsymbol{\tau}))$. Based on model assumptions mentioned in the introduction, we have $\tau_t \sim \tau_{t-1}$ for $u_t \overset{iid}{\sim} \mathcal{N}(0, \sigma_u^2)$ so that

$$p(\boldsymbol{\tau}|\sigma_u^2) = \prod_{t=2}^{T} \frac{1}{\sigma_u} \exp\left\{-\frac{1}{2\sigma_u^2}(\tau_t - \tau_{t-1})^2\right\}.$$

We place an inverse gamma prior (IG) on σ_u^2 given by

$$p(\sigma_u^2) = \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma_u^2}\right)^{\alpha+1} exp\left\{-\frac{\beta}{\sigma_u^2}\right\}.$$

Then, the posterior is

$$\begin{aligned} P(\sigma^2|\boldsymbol{\tau}, \boldsymbol{y}) = & \underbrace{\prod_{t=1}^{T} \left(\frac{\exp(\tau_t)}{1+\exp(\tau_t)}\right)^{y_t} \left(1 - \frac{\exp(\tau_t)}{1+\exp(\tau_t)}\right)^{n_t - y_t}}_{\text{Constant w.r.t. } \sigma^2} \cdot \\ & \prod_{t=1}^{T} \frac{1}{\sigma_u} \exp\left\{-\frac{1}{2\sigma_u^2}(\tau_t - \tau_{t-1})^2\right\} \cdot \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma_u^2}\right)^{\alpha+1} exp\left\{-\frac{\beta}{\sigma_u^2}\right\} \\ \propto & \prod_{t=1}^{T} \frac{1}{\sigma_u} \exp\left\{-\frac{1}{2\sigma_u^2}(\tau_t - \tau_{t-1})^2\right\} \cdot \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma_u^2}\right)^{\alpha+1} exp\left\{-\frac{\beta}{\sigma_u^2}\right\} \\ = & \frac{1}{\sigma_u^{T-1}} exp\left\{-\frac{1}{2\sigma_u^2}\boldsymbol{\tau Q \tau}\right\} \cdot \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma_u^2}\right)^{\alpha+1} exp\left\{-\frac{\beta}{\sigma_u^2}\right\} \\ \propto & \left(\frac{1}{\sigma_u^2}\right)^{\alpha + \frac{T-1}{2} + 1} exp\left\{-\frac{1}{\sigma_u^2}\left(\frac{1}{2}\boldsymbol{\tau Q \tau} + \beta\right)\right\} \end{aligned}$$

for a tridiagonal matrix $\boldsymbol{Q}$ with diagonal elements equal to 2 except first and last element which are 1, and the offdiagonal elements equal to -1. We recognize the posterior as the core of an inverse gamma IG(α^*, β^*) with shape $\alpha^* = \alpha + \frac{1}{2}(T-1)$ and scale $\beta^* = \beta + \frac{1}{2}\boldsymbol{\tau Q \tau}$.

d) Acceptance probability

Let $\mathcal{I} \subseteq \{1, 2, ..., 366\}$ be a set of time indices, and let $-\mathcal{I} = \{1, 2, ..., 366\} \setminus \mathcal{I}$. Furthermore, let $\boldsymbol{\tau}'$ denote the proposed values for $\boldsymbol{\tau}$. The MH step needs an acceptance probability denoted α for the proposed values $\boldsymbol{\tau}'_{\mathcal{I}}$. By using iterative conditioning we can write the acceptance probability as

$$\alpha(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y}) = \min\left(1, \frac{P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y})}{P(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y})} \frac{Q(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y})}{Q(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y})}\right),$$

where our prior proposal distribution is $Q(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y}) = P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)$. By considering

$$
\begin{aligned}
P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y}) &= \frac{P(\boldsymbol{\tau}'_{\mathcal{I}}, \boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y})}{P(\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y})} \\
&= \frac{P(\boldsymbol{y}|\boldsymbol{\tau}'_{\mathcal{I}}, \boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2) P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2) P(\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}{P(\boldsymbol{y}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2) P(\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)} \\
&= \frac{\overbrace{P(\boldsymbol{y}|\boldsymbol{\tau}'_{\mathcal{I}}, \boldsymbol{\tau}_{-\mathcal{I}})}^{\text{Conditionally independent}} \quad P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}{P(\boldsymbol{y}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)} \\
&= \frac{P(\boldsymbol{y}_{\mathcal{I}}|\boldsymbol{\tau}'_{\mathcal{I}}) P(\boldsymbol{y}_{-\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}) P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}{P(\boldsymbol{y}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}
\end{aligned}
$$

and equally

$$
P(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y}) = \frac{P(\boldsymbol{y}_{\mathcal{I}}|\boldsymbol{\tau}_{\mathcal{I}}) P(\boldsymbol{y}_{-\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}) P(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}{P(\boldsymbol{y}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}
$$

we can rewrite the acceptance probability as

$$
\begin{aligned}
\alpha(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2, \boldsymbol{y}) &= \min\left(1, \frac{P(\boldsymbol{y}_{\mathcal{I}}|\boldsymbol{\tau}'_{\mathcal{I}}) P(\boldsymbol{y}_{-\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}) P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)/P(\boldsymbol{y}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}{P(\boldsymbol{y}_{\mathcal{I}}|\boldsymbol{\tau}_{\mathcal{I}}) P(\boldsymbol{y}_{-\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}) P(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)/P(\boldsymbol{y}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)} \frac{P(\boldsymbol{\tau}_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}{P(\boldsymbol{\tau}'_{\mathcal{I}}|\boldsymbol{\tau}_{-\mathcal{I}}, \sigma_u^2)}\right) \\
&= \min\left(1, \frac{P(\boldsymbol{y}_{\mathcal{I}}|\boldsymbol{\tau}'_{\mathcal{I}})}{P(\boldsymbol{y}_{\mathcal{I}}|\boldsymbol{\tau}_{\mathcal{I}})}\right),
\end{aligned}
$$

which is the minimum of 1 and the ratio of likelihoods conditioned on the proposed values and the old values.

e) Implementation

In this section we implement an MCMC sampler for the posterior $P(\boldsymbol{\pi}, \sigma_u^2|\boldsymbol{y})$. For the conditional prior, $P(\tau_t|\boldsymbol{\tau}_{-t}, \sigma_u)$, we use an MH steps, and for σ_u we use Gibbs steps. We assume $\alpha = 2$ and $\beta = 0.05$ for the response given in Equation (1), and the initial value for $\sigma_u^2 = 0.1$. Initial values for τ are drawn from a standard normal distribution and the number of iterations is $N = 50'000$.

```
link = function(tau) {
    # Expit link
    return(exp(tau)/(1 + exp(tau)))
}

logbin = function(n, y, tau) {
    # Remake and use this
    return(y * log(1 + exp(-tau)) - (n - y) * log(1 + exp(tau)))
}

acceptRatio = function(n, y, tauProp, tau) {
    # Confirmed faster. N=1000: 4.7 vs 3.81
    return(exp(y * (tauProp - tau) + n * log((1 + exp(tau))/(1 + exp(tauProp)))))
}


mhRW = function(tau, sigma, yt, t, normVec = NA) {
    if (t == 1) {
        mu_ab = tau[2]
        sigma_aa = sigma
    } else if (t == 366) {
```

```r
        mu_ab = tau[365]
        sigma_aa = sigma
    } else {
        mu_ab = 1/2 * (tau[t - 1] + tau[t + 1])
        sigma_aa = sigma/2
    }
    # prop_tau = rnorm(1,mean=mu_ab, sd=sqrt(sigma_aa)) prop_tau =
    # normVec[t]*sigma + mu_ab
    prop_tau = normVec * sigma + mu_ab
    n = ifelse(t == 60, 10, 39)
    ratio = acceptRatio(n, yt, prop_tau, tau[t])
    if (runif(1) < min(c(1, ratio))) {
        return(list(tau = prop_tau, accepted = 1))
    } else {
        return(list(tau = tau[t], accepted = 0))
    }
}


mcmcIndivid = function(N, dt, sigma0 = 0.1) {
    # Allocate memory
    Ttot = 366
    tau = matrix(NA, nrow = N, ncol = Ttot)
    sigma = numeric(length = N)
    tau_i = numeric(length = Ttot)
    normMat = matrix(rep(rnorm(Ttot), N), nrow = N, ncol = Ttot)
    normVec = normMat[1, ]

    # Find init vals
    tau[1, ] = rnorm(Ttot)  # init tau drawn from normal distr.
    # tau[1,] = runif(366, -100, 100) # init tau drawn from uniform distr.
    sigma[1] = sigma0

    # Make Q matrix
    Q = matrix(0, nrow = Ttot, ncol = Ttot)
    diag(Q) = 2
    Q[c(1, length(Q))] = 1
    Q[abs(row(Q) - col(Q)) == 1] <- -1

    # Run mcmc for N iterations
    accepted = 0
    for (i in 2:N) {
        tau_i = tau[i - 1, ]
        sigma_i = sigma[i - 1]
        for (t in 1:Ttot) {
            # rtemp= mhRW(tau_i, sigma_i, dt$n.rain[t], t)
            rtemp = mhRW(tau_i, sqrt(sigma_i), dt$n.rain[t], t, normVec[t])
            tau[i, t] = rtemp$tau
            accepted = accepted + rtemp$accepted
        }
        normVec = normMat[i, ]
        # Squared diff. of tau vec.
        tQt = sum((tau[i, -Ttot] - tau[i, -1])^2)  # this sim tau vals.
```

```
        # tQt = sum((tau[i-1,-366] - tau[i-2,-1])^2) # prev sim tau vals.

        # Gibbs step (Draw from IG)
        sigma[i] = 1/rgamma(1, 2 + (Ttot - 1)/2, 0.05 + 0.5 * tQt)  # Gibbs inline

        # if (i%%(N/10)==0){ print(i/N*100) print(accepted/(i*366)) }
    }
    return(list(tau = tau, sigma = sigma, accProb = accepted/(N * Ttot)))
}
```

```
set.seed(321)
N = 50000
ptm = proc.time()
results = mcmcIndivid(N, rain)
time = proc.time() - ptm
```

```
CImean = function(p, col = "cyan3") {
    c(mean = mean(p), quantile(p, probs = c(0.025, 0.975)))
}

plotTAH = function(p, hcol = "cyan3", xlab = "", ylab = "") {
    # Plots trace, autocorr and hist of probs
    plot(p, type = "l", xlab = "Iterations", ylab = "Probability")
    abline(h = mean(p), col = "red")
    acf(p, main = "")
    hist(p, nclass = 40, prob = T, main = "")
    abline(v = quantile(p, probs = 0.025), col = hcol)
    abline(v = quantile(p, probs = 0.975), col = hcol)
}

p1 = link(results$tau[, 1])
p201 = link(results$tau[, 201])
p366 = link(results$tau[, 366])

par(mfrow = c(3, 3))
plotTAH(p1)
plotTAH(p201)
plotTAH(p366)
```

The vertical red line in the trace plots to the left in Figure 2 show the mean of $\pi(\tau_t)$ over all $50'000$ iterations, for $t = 1, 201, 366$. The horizontal blue lines in the histogram plot The traceplot to the left in Figure bares resemblance to that of a random walk for all values of τ_t, $t = 1, 201, 366$.

```
par(mfrow = c(3, 1))
plot(results$sigma, type = "l")
acf(results$sigma)
hist(results$sigma, nclass = 100, prob = T)
```

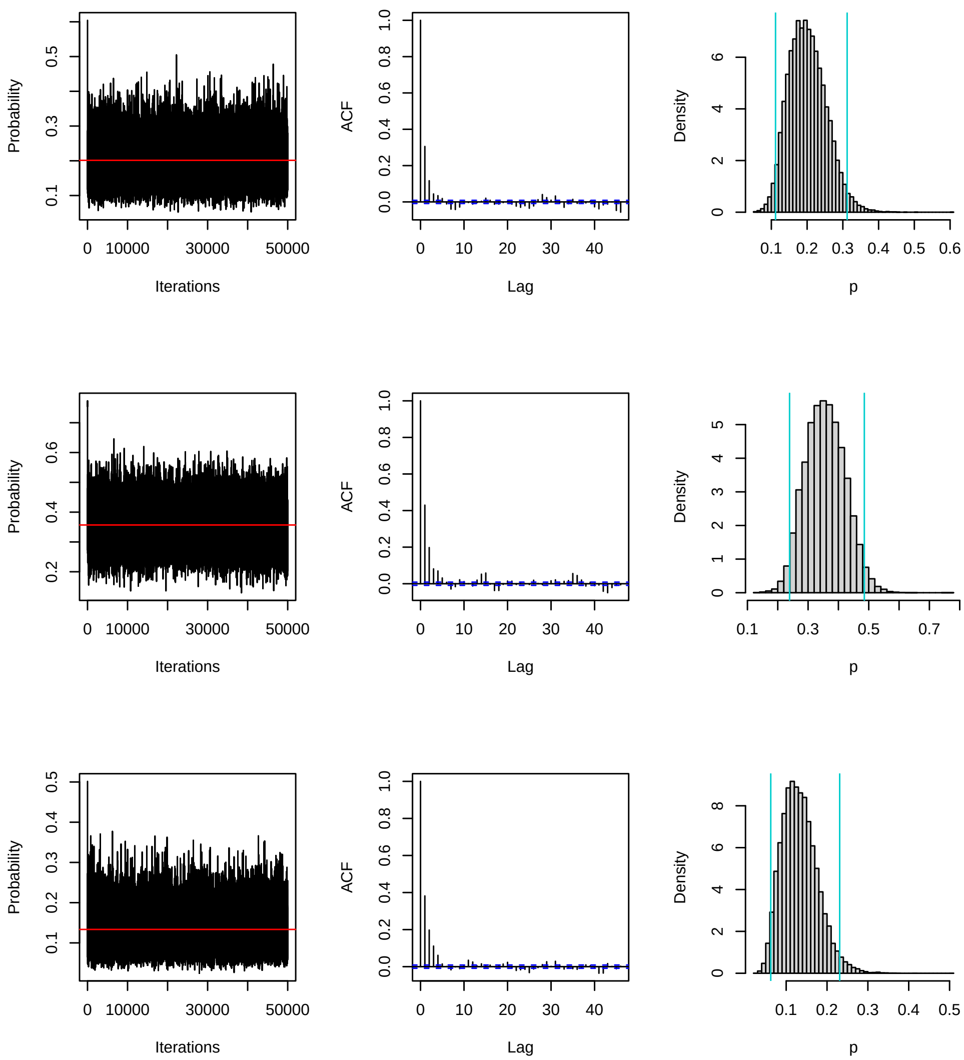
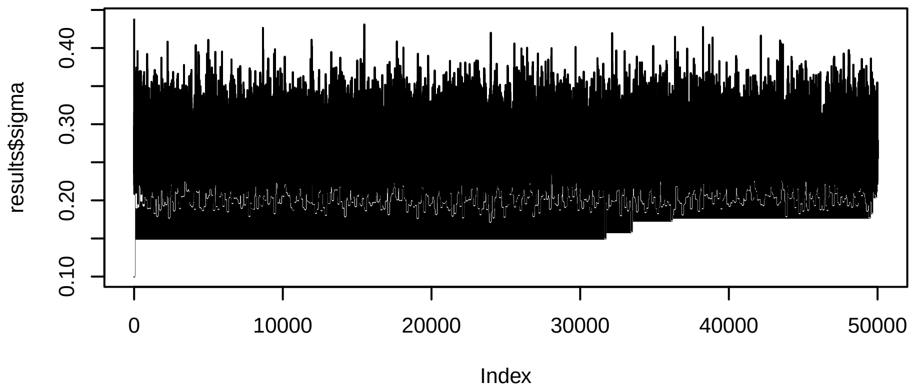
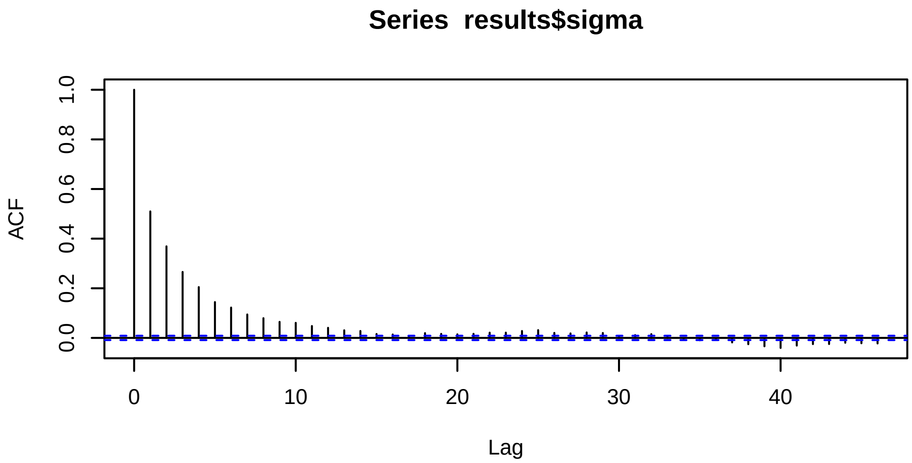
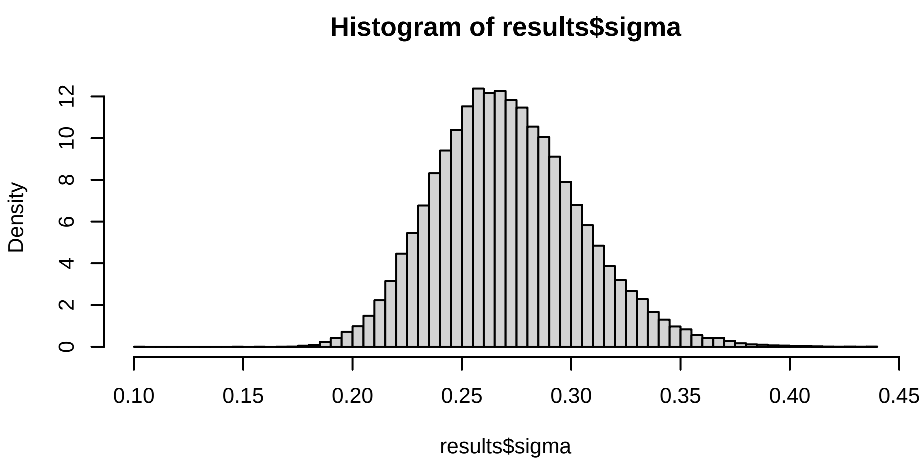


Figure 2: Traceplot, autocorrelation and histogram plots for $\pi(\tau_1), \pi(\tau_{201})$ and $\pi(\tau_{366})$ from top to bottom.





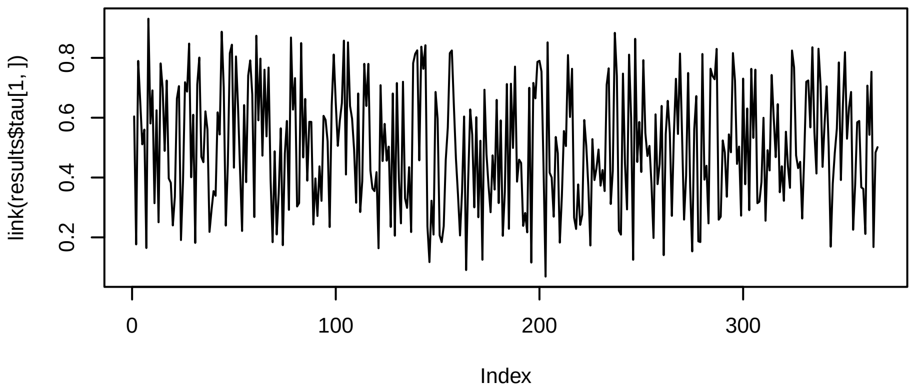


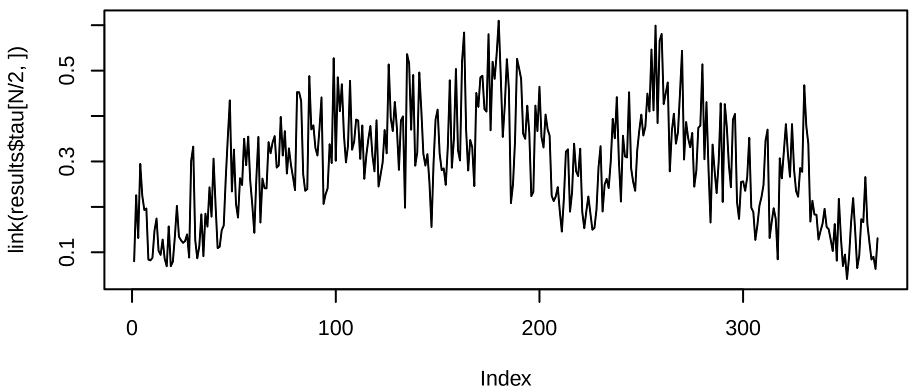

```
ciAndMean = rbind(CImean(p1[1:N]), CImean(p201[1:N]), CImean(p366[1:N]))
rownames(ciAndMean) = c(1, 201, 366)
ciAndMean
```

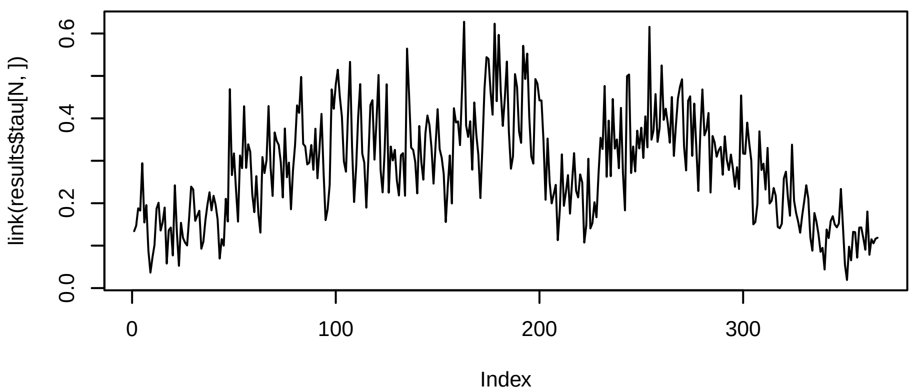
```
##           mean       2.5%      97.5%
```

```
## 1   0.2013289 0.11166385 0.3120152
## 201 0.3569094 0.23896194 0.4854839
## 366 0.1334187 0.06226288 0.2310067
```

```r
par(mfrow = c(3, 1))
plot(link(results$tau[1, ]), type = "l")
plot(link(results$tau[N/2, ]), type = "l")
plot(link(results$tau[N, ]), type = "l")
```


link(results$tau[1,])
0.2
0.4
0.6
0.8
0
100
200
300
Index


link(results$tau[N/2,])
0.1
0.3
0.5
0
100
200
300
Index


link(results$tau[N,])
0.0
0.2
0.4
0.6
0
100
200
300
Index