MLE of Location Parameter of a Cauchy Distribution by Using Different Algorithms

HW 3 of STAT 5361 Statistical Computing

Biju Wang[1]

09/17/2018

[1]bijuwang@uconn.edu

Contents

Chapter 1

Proofs and Loglikelihood Function Plot against θ

1.1 Proofs

The likelihood function is

$$L(\theta) = \prod_{i=1}^{n} \frac{1}{\pi[1 + (X_i - \theta)^2]}$$

Hence, the log likelihood fucntion is

$$l(\theta) = \log L(\theta) = \log \prod_{i=1}^{n} \frac{1}{\pi[1 + (X_i - \theta)^2]} = -n \log \pi - \sum_{i=1}^{n} \log[1 + (\theta - X_i)^2] \tag{1.1}$$

Further

$$l'(\theta) = -\sum_{i=1}^{n} \frac{d}{d\theta} \log[1 + (\theta - X_i)^2] = -2 \sum_{i=1}^{n} \frac{\theta - X_i}{1 + (\theta - X_i)^2} \tag{1.2}$$

Compute the second derivative according to $l'(\theta)$

$$l''(\theta) = -2 \sum_{i=1}^{n} \frac{d}{d\theta} \frac{\theta - X_i}{1 + (\theta - X_i)^2} = -2 \sum_{i=1}^{n} \frac{1 - (\theta - X_i)^2}{[1 + (\theta - X_i)^2]^2} \tag{1.3}$$

Therefore, the Fisher information is

$$\begin{aligned}
I_n(\theta) &= -E_X[l''(\theta)] \\
&= 2nE_X \left[\frac{1 - (\theta - X)^2}{[1 + (\theta - X)^2]^2} \right] \\
&= 2n \int_{-\infty}^{\infty} \frac{1 - (x - \theta)^2}{[1 + (x - \theta)^2]^2} \frac{1}{\pi[1 + (x - \theta)^2]} dx \\
&= \frac{2n}{\pi} \int_{-\infty}^{\infty} \frac{1 - x^2}{(1 + x^2)^2} \frac{1}{1 + x^2} dx = \frac{2n}{\pi} \int_{-\infty}^{\infty} \left(\frac{x}{1 + x^2} \right)' \frac{1}{1 + x^2} dx = \frac{2n}{\pi} \left[\frac{x}{(1 + x^2)^2} \Big|_{-\infty}^{\infty} + \int_{-\infty}^{\infty} \frac{2x^2}{(1 + x^2)^3} \right] \\
&= \frac{4n}{\pi} \int_{-\infty}^{\infty} \frac{x^2}{(1 + x^2)^3} dx = \frac{4n}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{\tan^2 \alpha}{(1 + \tan^2 \alpha)^3} \sec^2 \alpha d\alpha = \frac{4n}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{1 - \cos 4\alpha}{8} d\alpha = \frac{4n}{\pi} \cdot \frac{\pi}{8} \\
&= \frac{n}{2}
\end{aligned} \tag{1.4}$$

1.2 Loglikelihood Function Plot against θ

The following plot is the curve of log likelihood function

```
set.seed(20180909)
sample <- rcauchy(10, 5)

log_sum <- function(x, sample){
  log_sum <- 0
  for (i in 1:length(sample)) {
    log_sum <- log_sum -log(pi) - log(1 + (x - sample[i])^2)
  }
  log_sum
}

library("ggplot2")
ggplot(data.frame(x = c(0, 10)), aes(x = x)) +
stat_function(fun = function(x) log_sum(x, sample)) +
labs(x = expression("Values of"~theta), y = expression("Log Likelihood Function"~l(theta))) +
theme(plot.title = element_text(hjust = 0.5)) +
ggtitle(expression("Log Likelihood Function vs."~theta))
```

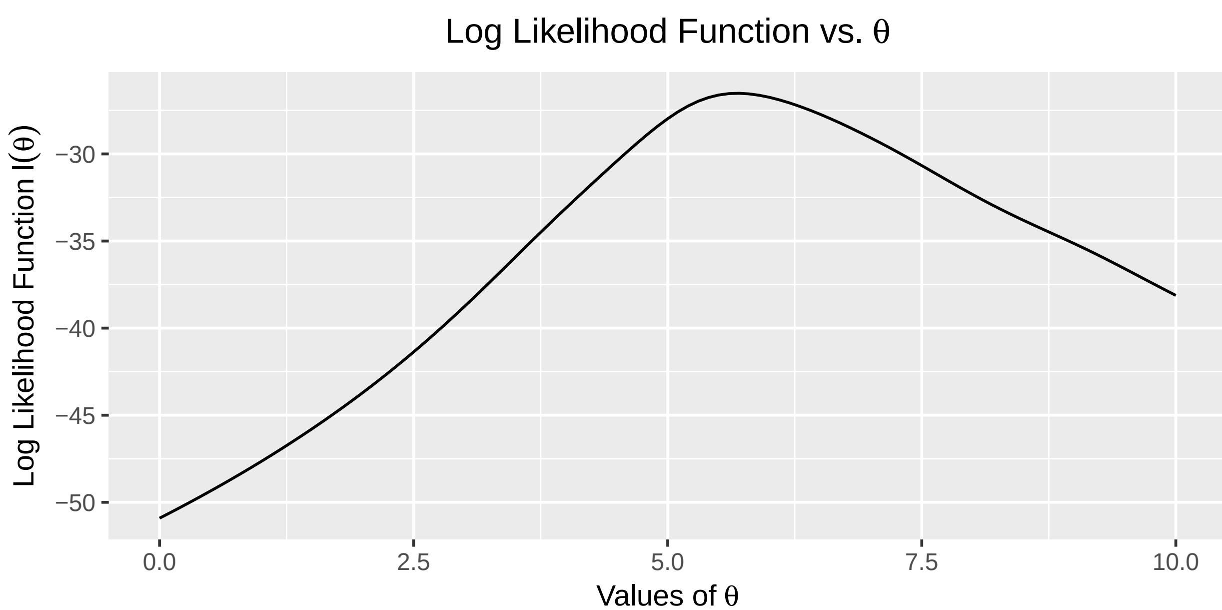


Figure 1.1: Log Likelihood Function vs. θ

Chapter 2

Newton-Raphson Method

```
set.seed(20180909)
sample <- rcauchy(10, 5)

log_sum <- function(x){
  log_sum <- 0
  for (i in 1:length(sample)) {
    log_sum <- log_sum -log(pi) - log(1 + (x - sample[i])^2)
  }
  log_sum
}

dev1_log_sum <- function(x){
  dev1_log_sum <- 0
  for (i in 1:length(sample)) {
    dev1_log_sum <- dev1_log_sum - 2 * (x - sample[i])/(1 + (x - sample[i])^2)
  }
  dev1_log_sum
}

dev2_log_sum <- function(x){
  dev2_log_sum <- 0
  for (i in 1:length(sample)) {
    dev2_log_sum <- dev2_log_sum - 2 * (1 - (x - sample[i])^2)/(1 + (x - sample[i])^2)^2
  }
  dev2_log_sum
}

newton.raphson <- function(init, fun, fun.dev, maxiter = 100, tol = .Machine$double.eps^0.2){
  x <- init
  for (i in 1:maxiter) {
    x1 <- x -fun(x)/fun.dev(x)
    if(abs(x1 - x) < tol) break
    x <- x1
  }
  if(i == maxiter)
    message("Reached the maximum iteration!")
```

```
  return(data.frame(root = x1, iter = i))
}

init <- seq(-10, 20, by = 0.5)
res <- data.frame(init = init, root = rep(NA, length(init)))
for (i in 1:length(init)) {
  res$root[i] <- newton.raphson(init[i], dev1_log_sum, dev2_log_sum)$root
}

res_trans <- t(as.matrix(round(res, 2)))
rownames(res_trans) <- c("Initial Values", "Roots")

library("pander")
library("ggplot2")
pander(res_trans, split.table = 100, style = 'rmarkdown')
```

Table 2.1: Table continues below

Initial Values	-10	-9.5	-9	-8.5	-8
Roots	-2.162e+31	-2.097e+31	-2.031e+31	-1.965e+31	-1.9e+31

Table 2.2: Table continues below

Initial Values	-7.5	-7	-6.5	-6	-5.5
Roots	-1.834e+31	-1.768e+31	-1.701e+31	-1.635e+31	-1.569e+31

Table 2.3: Table continues below

Initial Values	-5	-4.5	-4	-3.5	-3
Roots	-1.502e+31	-1.436e+31	-1.369e+31	-1.302e+31	-1.235e+31

Table 2.4: Table continues below

Initial Values	-2.5	-2	-1.5	-1	-0.5
Roots	-1.167e+31	-1.1e+31	-1.032e+31	-9.648e+30	-8.97e+30

Table 2.5: Table continues below

Initial Values	0	0.5	1	1.5	2
Roots	-8.295e+30	-7.623e+30	-6.962e+30	-6.327e+30	-5.76e+30

Table 2.6: Table continues below

Initial Values	2.5	3	3.5	4	4.5	5	5.5
Roots	-5.397e+30	-5.999e+30	7.515e+31	21.08	19.38	5.69	5.69

Table 2.7: Table continues below

Initial Values	6	6.5	7	7.5	8	8.5	9
Roots	5.69	5.69	1.608e+31	-4.779e+30	20.56	19.38	5.69

Table 2.8: Table continues below

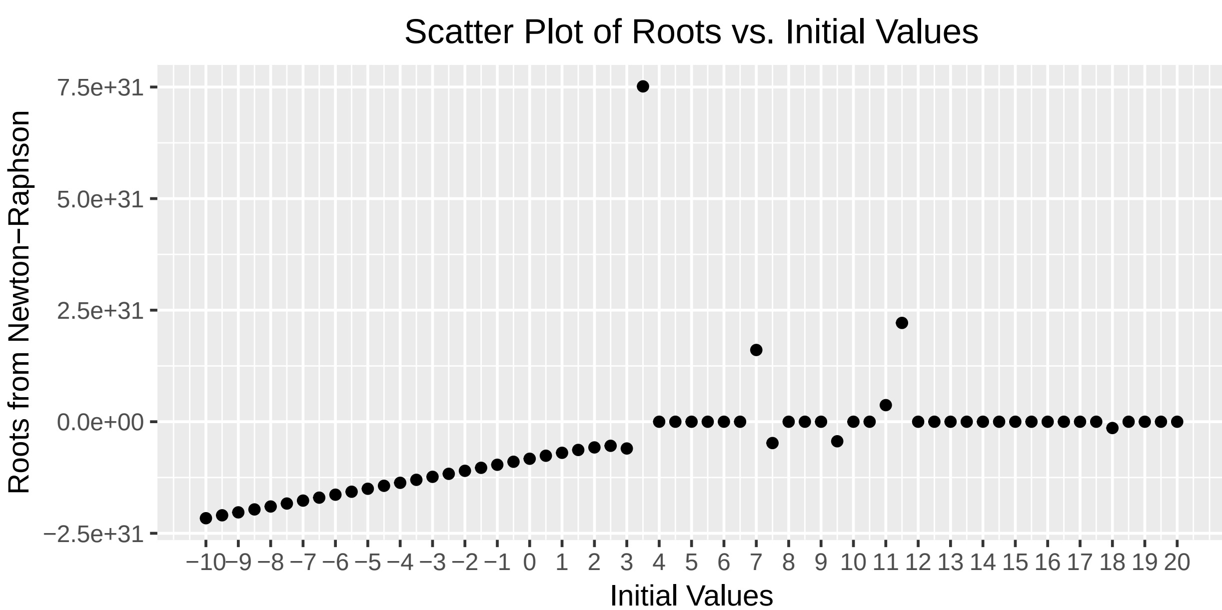
Initial Values	9.5	10	10.5	11	11.5	12	12.5
Roots	-4.38e+30	19.38	5.69	3.72e+30	2.215e+31	21.08	21.08

Table 2.9: Table continues below

Initial Values	13	13.5	14	14.5	15	15.5	16	16.5
Roots	21.08	19.38	19.38	19.38	20.56	21.08	21.08	20.56

Initial Values	17	17.5	18	18.5	19	19.5	20
Roots	20.56	20.56	-1.413e+30	21.08	21.08	21.08	21.08

```
ggplot(res, aes(x = init, y = root)) + geom_point() +
scale_x_continuous(breaks = round(seq(min(res$init), max(res$init), by = 1),1)) +
labs(x = "Initial Values", y = "Roots from Newton-Raphson") +
theme(plot.title = element_text(hjust = 0.5)) +
ggtitle("Scatter Plot of Roots vs. Initial Values")
```



From the above table and figure we can see, when the initial values is not less than 5 and around 5, the outcomes are quite close to the true value 5. For instance, from the table when the initial values are $5, 5, 5, 6, 6.5$, the roots are $5.69, 5.69, 5.69, 5.69$. While for other initial values, the roots are quite unstable, for some the roots are very large and for others the roots only have magnitude of 10.

Chapter 3

Fixed-Point Iterations

Chapter 4

Fisher Scoring Method and Newton-Raphson Method

Chapter 5

Comments