Solution 7: Simulations

Siyue Yang

06/03/2022

Simulations on Cauchy

Suppose $\boldsymbol{X} = (X_1, \ldots, X_n)$ is an i.i.d. sample from the shifted Cauchy distribution with density

$$f(x \mid \theta) = \frac{1}{\pi\left(1 + (x-\theta)^2\right)}, \quad x \in \mathbb{R}$$

Our goal is to compare the following 4 estimators of the parameter θ.

- Sample mean
$$\hat{\theta}_n^{(1)} = \bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$$
- Sample median
$$\hat{\theta}_{\mathrm{n}}^{(2)} = M_{\mathrm{n}} = \frac{1}{2}\left(X_{(k)} + X_{(k+1)}\right)$$
- Modified sample mean
$$\hat{\theta}_n^{(3)} = M_n + \frac{2}{n} \cdot \left.\frac{\partial \ell}{\partial \theta}\right|_{\theta = M_n}$$
where ℓ is the log-likelihood function.
- Maximum likelihood estimator (MLE) $\hat{\theta}_{\mathrm{n}}^{(4)}$ defined by
$$\ell\left(\hat{\theta}_n^{(4)} \mid X\right) = \max_{\theta \in \mathbb{R}} \ell(\theta \mid \boldsymbol{X})$$
where ℓ is the log-likelihood function.

1. Derive the likelihood function and log-likelihood function.
2. Simulate data from Cauchy distribution with location 5, and scale 1.
3. Choose your number of simulations.
4. Verify consistency of the estimators. There are different approaches. You can samples the data sequentially and plot the sequence of the results as a function of n. What do you observe if it is a consistent estimator? The second approach is to use only the representative increasing values of the sample size. e.g. use $n = 10, 50, 100, 200, \ldots, 1000$ and what do you observe?
5. Calculate the mean square error of the estimators.
6. Calculate the coverage probability of the estimators. Calculate $\mathbf{P}_\theta\left(\left|\hat{\theta}_n - \theta\right| \leq \varepsilon\right)$, for $\varepsilon = 0.1$, and $\theta = 5$.

Solution

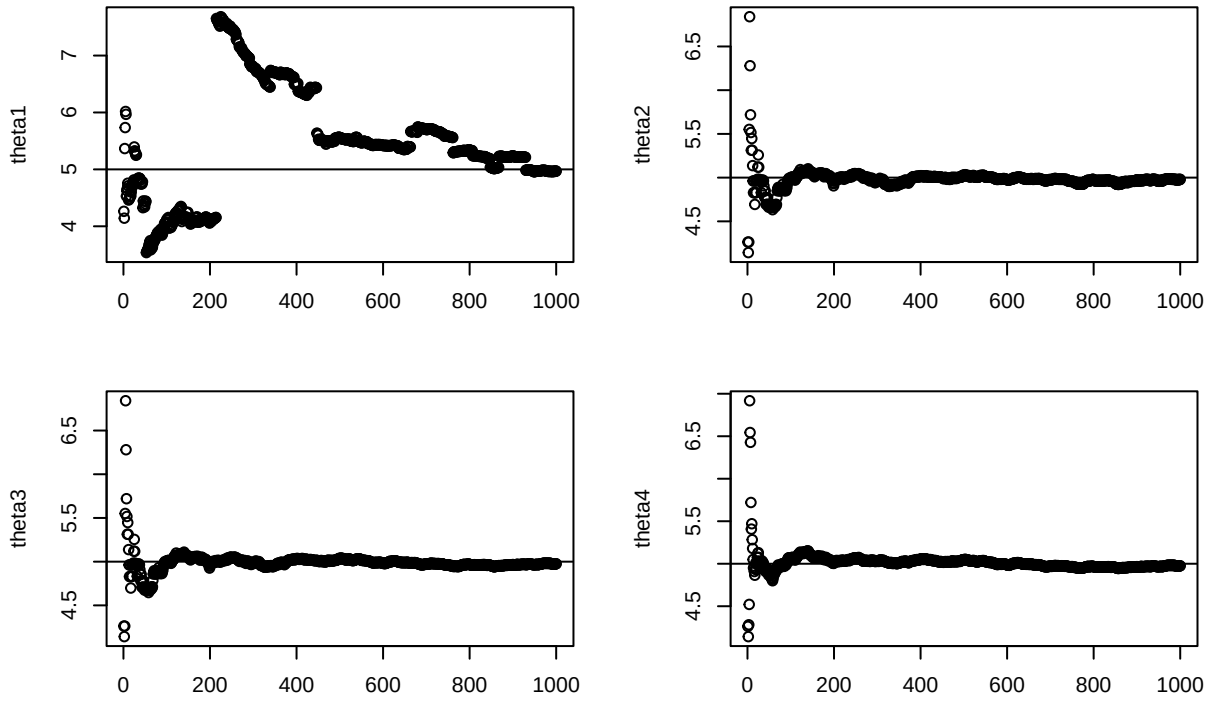
4. Goal: to verify $\hat{\theta}_n \to \theta$ as $n \to \infty$.

Two different approaches are used to verify the consistency. The first approach samples the values sequentially and plot the sequence of the resulting values as a function of n (shown below). For an consistent estimator, the dots will tends to get close to the true θ as n increases. From the figures, all the estimators are consistent except for $\hat{\theta}_n^{(1)}$.

```
# Simulate Cauchy.
n <- 1000
set.seed(123456)
cauchy_samples <- rcauchy(n, location = 5, scale = 1)

# Function to calculate the derivative of the log-likelihood.
dloglik <- function(x, theta) {
  d1 <- x - theta
  d2 <- 1 + (x - theta)^2
  return(2*sum(d1/d2))
}

# Function to calculate the log-likelihood.
loglik <- function(theta) {
   x <- cauchy_samples
   n <- length(x)
   l1 <- -n*log(pi)
   l2 <- -sum(log(1 + (x - theta)^2))
   return(l1 + l2)
}

nloglik<- function(theta) {
  return(-loglik(theta))
}
```

```
# Consistency.
theta1 <- vector(length = n)
theta2 <- vector(length = n)
theta3 <- vector(length = n)
theta4 <- vector(length = n)

full_samples <- cauchy_samples
for (k in 1:n) {
  cauchy_samples <- NULL
  cauchy_samples <- full_samples[1:k]
  #print(cauchy_samples)
  theta1[k] <- mean(cauchy_samples)
  #print(theta1[i])
  theta2[k] <- median(cauchy_samples)
  theta3[k] <- theta2[k] + 2/n * dloglik(cauchy_samples, theta2[k])
  theta4[k] <- optimize(nloglik, interval = c(-10,10))$minimum
}

par(mfrow = c(4, 2), mar = c(1, 4.1, 4.1, 2.1))
plot(theta1)
abline(h = 5)
plot(theta2)
abline(h = 5)
plot(theta3)
abline(h = 5)
```

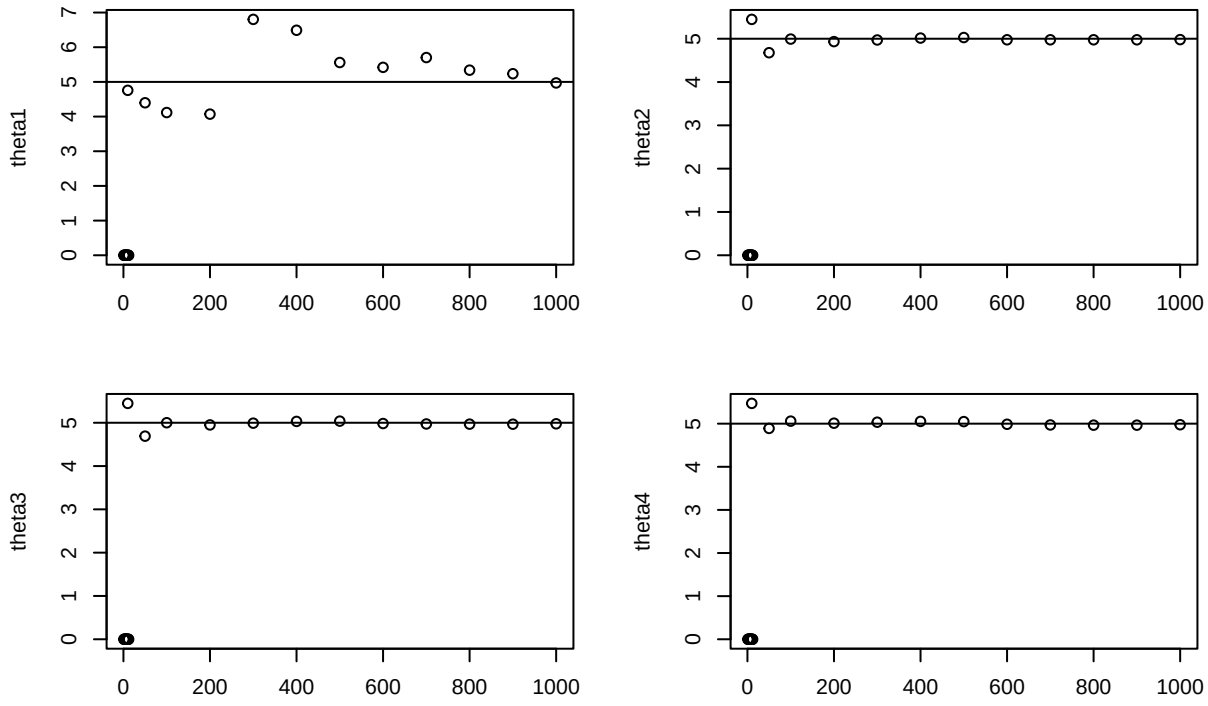
```
plot(theta4)
abline(h = 5)
```



The second approach to verify the consistency using only the representative increasing values of the sample size. Here, $n = 10, 50, 100, 200, \ldots, 1000$ are used. The results are consistent with those observed from the first approach.

```
# Consistency approach 2.
full_samples <- cauchy_samples
theta1 <- vector(length = 12)
theta2 <- vector(length = 12)
theta3 <- vector(length = 12)
theta4 <- vector(length = 12)
for (k in c(10, 50, 100, 200, 300, 400, 500,
            600, 700, 800, 900, 1000)) {
  cauchy_samples <- NULL
  cauchy_samples <- full_samples[1:k]
  #print(cauchy_samples)
  theta1[k] <- mean(cauchy_samples)
  #print(theta1[i])
  theta2[k] <- median(cauchy_samples)
  theta3[k] <- theta2[k] + 2/n * dloglik(cauchy_samples, theta2[k])
  theta4[k] <- optimize(nloglik, interval = c(-10,10))$minimum
}

par(mfrow = c(4, 2), mar = c(1, 4.1, 4.1, 2.1))
plot(theta1)
abline(h = 5)
plot(theta2)
abline(h = 5)
plot(theta3)
abline(h = 5)
```

```
plot(theta4)
abline(h = 5)
```



5. Goal: to calculate $\text{MSE}_\theta\left(\hat{\theta}_n\right) = \text{E}_\theta\left(\left(\hat{\theta}_n - \theta\right)^2\right)$, and $\theta = 5$.

Monte Carlo method based on the law of large number are used for simulation. Specifically, 1000 different datasets are replicated, i.e. for each replication, I set different seeds to generate the data.

```
nsim <- 1000
theta1 <- vector(length = nsim)
theta2 <- vector(length = nsim)
theta3 <- vector(length = nsim)
theta4 <- vector(length = nsim)

for (i in 1:nsim) {
  set.seed(1234 + i)

  n <- 1000
  cauchy_samples <- rcauchy(n, location = 5, scale = 1)

  theta1[i] <- mean(cauchy_samples)
  theta2[i] <- median(cauchy_samples)
  theta3[i] <- theta2[i] + 2/n * dloglik(cauchy_samples, theta2[i])
  theta4[i] <- optimize(nloglik, interval = c(-10,10))$minimum
}
```

```
# hist(theta1)
# hist(theta2)
# hist(theta3)
# hist(theta4)

mse <- function(hat_theta, theta) {
```

```
  return(mean(hat_theta - theta)^2)
}

cat("mse1 =", mse(theta1, 5), "\n")
```

```
## mse1 = 1.530928
```

```
cat("mse2 =", mse(theta2, 5), "\n")
```

```
## mse2 = 6.172925e-06
```

```
cat("mse3 =", mse(theta3, 5), "\n")
```

```
## mse3 = 1.968817e-06
```

```
cat("mse4 =", mse(theta4, 5), "\n")
```

```
## mse4 = 1.835778e-06
```

6. Coverage probability

Goal: To calculate $\mathbf{P}_\theta\left(\left|\hat{\theta}_n - \theta\right| \leq \varepsilon\right)$, for $\varepsilon = 0.1$, and $\theta = 5$.

```
# Coverage probability
cov_prob <- function(hat_theta, theta, eps = 0.1) {
  return(mean(abs(hat_theta - theta) <= eps))
}

cat("cov_prob1 =", cov_prob(theta1, 5), "\n")
```

```
## cov_prob1 = 0.064
```

```
cat("cov_prob2 =", cov_prob(theta2, 5), "\n")
```

```
## cov_prob2 = 0.957
```

```
cat("cov_prob3 =", cov_prob(theta3, 5), "\n")
```

```
## cov_prob3 = 0.972
```

```
cat("cov_prob4 =", cov_prob(theta4, 5), "\n")
```

```
## cov_prob4 = 0.973
```