Tutorial 4 - Likelihood (Solutions).

Christopher A Oldnall

Welcome to the fourth tutorial of the Regression and Simulation methods module. This is the next script in developing your skills in R, whilst learning about the use of likelihood. Throughout this notebook we will be looking at using functions that are built in for solving likelihood estimation problems both analytically and numerically.

Exercise 0

Throughout remember we will need tidyverse, go ahead and do this as your first task. In addition install the package 'maxLik' and load this in as well.

Your Answer:

```r
library(tidyverse)
#install.packages("maxLik")
library(maxLik)
```

Before we explore some exercises this week, we are going to see an example of constructing a function and also a while loop; let us consider a very basic example. Here is an example function in R:

This function takes in a values of a and b from a right angled triangle and returns the value of the hypotenuse.

```r
Hypotenuse <- function(a, b){sqrt(a**2 + b**2)}
Hypotenuse(3, 4)
```

```
## [1] 5
```

Perhaps I want to find an equation where the hypotenuse is bigger than a certain value given a set b but variable a.

```r
eps = 1000
a = 9
b = 842.390
while(Hypotenuse(a, b) < eps){a <- a + 0.001}
```

Exercise 1

Now lets do some stuff related to the MLE proceudure. A random sample is available:

```r
y = c(1.09, -0.23, 0.79, 2.31, -0.81)
```

Determine the maximum likelihood estimate of theta using the score function derived using the Fisher's method of scoring, which says:

$$\theta_{r+1} = \theta_r + U(\theta_r)/I(\theta_r).$$

Here set $I(\theta_r) = n/2$ - this will be explored next week. Use the stopping criterion $|U(\theta_r) - \theta| \leq \epsilon$ for some small epsilon.

```r
U <- function(theta){2*sum((y-theta)/(1+(y-theta)^2))}
theta <- 0
eps <- 0.0000001
while(abs(U(theta)) > eps) {theta <- theta + U(theta)*2/5}

theta
```

```
## [1] 0.6393402
```

Exercise 2

Try to do the same thing again but instead this time use 'maxLik' function with the Newton-Raphson method. Hint: once installed use '?maxLik' to find out about the function.

Your Answer:

```r
loglik <- function(theta){-sum(log(1+(y-theta)^2))}
NRmax <- maxLik(loglik, grad = NULL, hess = NULL, -10, method="NR", constraints=NULL)
summary(NRmax)
```

```
## --------------------------------------------
## Maximum Likelihood estimation
## Newton-Raphson maximisation, 6 iterations
## Return code 1: gradient close to zero (gradtol)
## Log-Likelihood: -3.234488
## 1  free parameters
## Estimates:
##      Estimate Std. error t value Pr(> t)
## [1,]   0.6393     0.6143   1.041   0.298
## --------------------------------------------
```

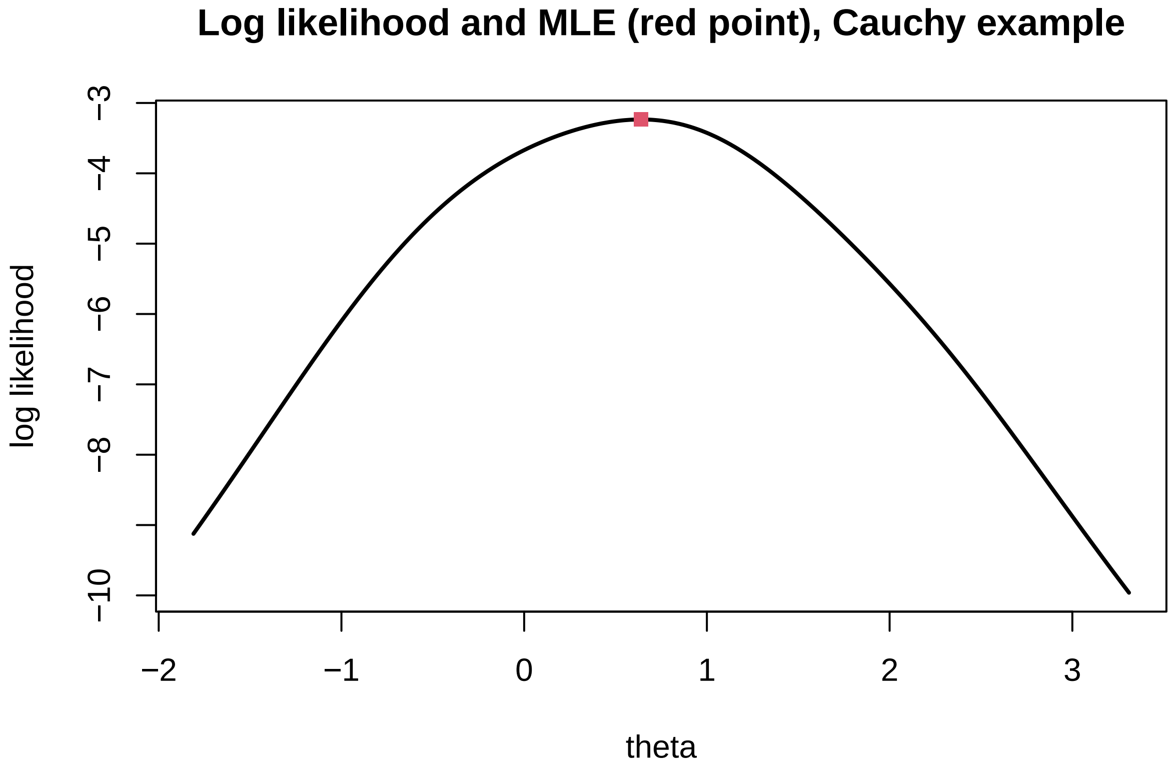
Exercise 3

Plot the log-likelihood function as a function of theta to check that the identified MLE corresponds to the maximum.

Your Answer:

```r
# Set some entry points from below the smallest y and above the largest.
# Have these points increase in increments of 0.01
x = seq(min(y)-1, max(y)+1, 0.01)
# Initilise a vector of length x with NAs (ready for the likelihoods).
loglik.x = rep(NA, length(x))
# For each value in the vector x, calculate the likelihood and
# then store it in the correct position in the loglik.x vector.
for (i in 1:length(x))
  loglik.x[i]= loglik(x[i])
# Set the optimal value as known from the NR MLE.
theta.hat =  0.6393
# Plot the values from the log-likelihood against the x values.
```

```r
plot(x,  loglik.x, type = 'l', lwd=2, xlab = "theta", ylab="log likelihood")
# Add the optimal value point.
points(theta.hat, loglik(theta.hat), col=2, pch=15)
# Add a title to the plot.
title("Log likelihood and MLE (red point), Cauchy example")
```



This is the end of this tutorial. Firstly next you should attempt the question at the end of the slides for this week. This can then be following by 4-1, 4-2 and 4-3 from the SMSTC exercises this week.