Many Local Maxima and Modeling Beetle Data Problems

5361 Homework 4

Qinxiao Shi *

29 September 2018

1 Many Local Maxima

1.1 Log-likelihood Function

The probability density function with parameter θ is:

$$f(x;\theta) = \frac{1-\cos(x-\theta)}{2\pi}, 0 \le x \le 2\pi, \theta \in (-\pi, \pi)$$

\end{equation*} The likelihood function of $f(x;\theta)$ is:

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

The log-likelihood function is:

$$\ell(\theta) = -n\log(2\pi) + \sum_{i=1}^{n}\log[1-\cos(X_i-\theta)]$$

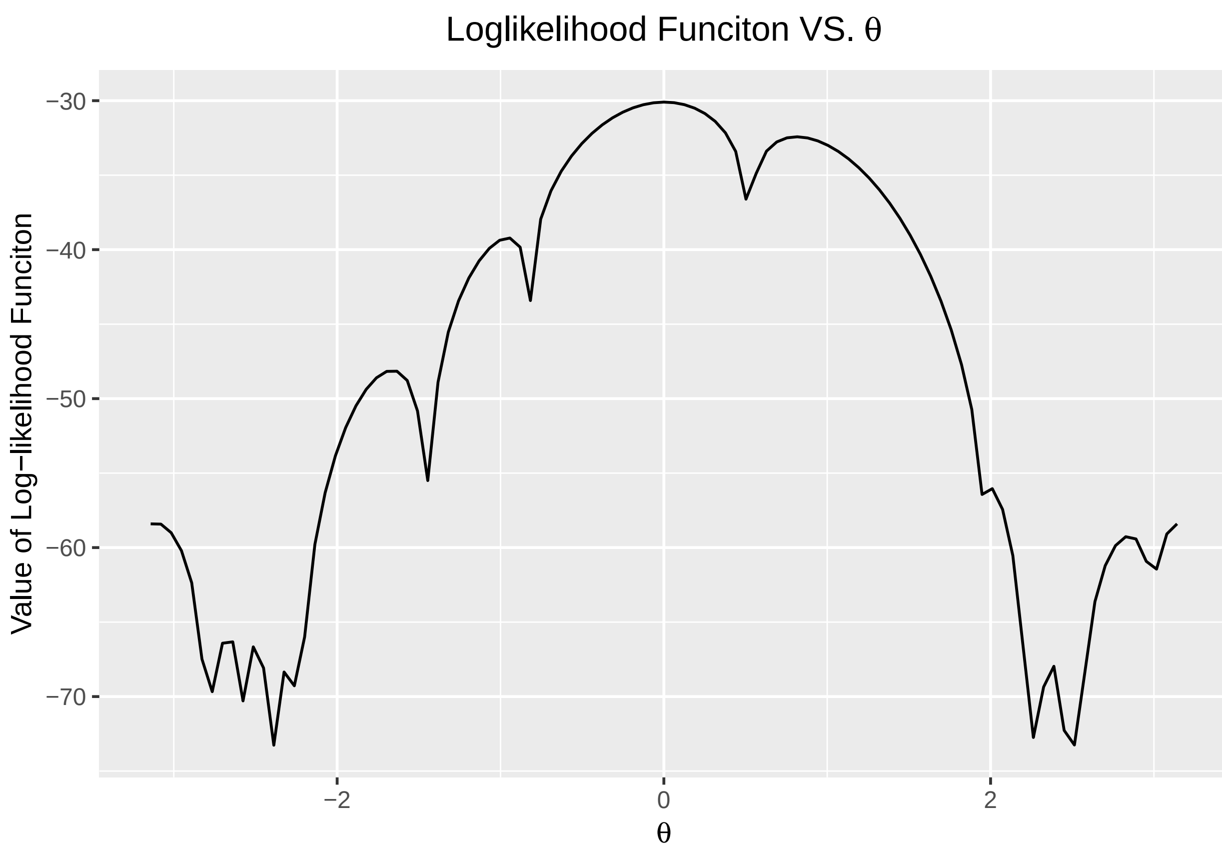
```r
library("ggplot2")
set.seed(20180909)

x <- c(3.91, 4.85, 2.28, 4.06, 3.70, 4.04, 5.46, 3.53, 2.28, 1.96,
       2.53, 3.88, 2.22, 3.47, 4.82, 2.46, 2.99, 2.54, 0.52)

y <- function(theta){
  y <- 0
  for (i in 1:length(x)){
    y <- y-log(2*pi)+log(1-cos(x[i]-theta))
  }
  return(y)
}

ggplot(data.frame(theta=c(-pi,pi)), aes(x=theta))+
  stat_function(fun = function(theta) y(theta))+
  ggtitle(expression("Loglikelihood Funciton VS."~theta))+
  theme(plot.title = element_text(hjust = 0.5))+
  labs(y="Value of Log-likelihood Funciton", x=expression(theta))
```

*qinxiao.shi@uconn.edu



1.2 Method-of-Moments Estimator

The expectation of $f(x;\theta)$ is:

$$
\begin{aligned}
\mathbb{E}(X \mid \theta) &= \int_0^{2\pi} x f(x;\theta)dx \\
&= \frac{1}{2\pi}(\int_0^{2\pi} x dx - \int_0^{2\pi} x\cos(x-\theta)dx) \\
&= \frac{1}{2\pi}[2\pi^2 - (x\sin(x-\theta)|_0^{2\pi} - \int_0^{2\pi} \sin(x-\theta)dx)] \\
&= \frac{1}{2\pi}[2\pi^2 - (x\sin(x-\theta) + \cos(x-\theta))\Big|_0^{2\pi}] \\
&= \frac{1}{2\pi}\{2\pi^2 - [x(\sin(x)\cos(\theta) - \cos(x)\sin(\theta)) + \cos(x)\cos(\theta) + \sin(x)\sin(\theta)\Big|_0^{2\pi})]\} \\
&= \frac{1}{2\pi}[2\pi^2 + 2\pi\sin\theta] \\
&= \pi + \sin\theta
\end{aligned}
$$

So when we do estimation: $\bar{X_n} = \pi + \sin\hat{\theta}$, $\tilde{\theta}_n = \arcsin(\bar{X_n} - \pi) = arcsin(3.2368 - \pi) = 0.0954$

1.3 Newton–Raphson Method For MLE

The gradient is:

$$\ell'(\theta) = \sum_{i=1}^{n} \frac{-\sin(X_i - \theta)}{1 - \cos(X_i - \theta)}$$

The hessian is:

$$\begin{aligned}
\ell''(\theta) &= \sum_{i=1}^{n} \frac{-\cos(X_i - \theta)[1 - \cos(X_i - \theta)] + [\sin(X_i - \theta)]^2}{[1 - \cos(X_i - \theta)]^2} \\
&= \sum_{i=1}^{n} \frac{-\cos(X_i - \theta) + 1}{[1 - \cos(X_i - \theta)]^2} \\
&= \sum_{i=1}^{n} \frac{1}{1 - \cos(X_i - \theta)}
\end{aligned}$$

```r
library("pracma")
library("pander")
library("gridExtra")
library("grid")
library("knitr")
library("kableExtra")

gradient <- function(theta){
    gradient <- sum(-sin(x-theta)/(1-cos(x-theta)))
  return(gradient)
}

hessian <- function(theta){
    hessian <- sum(1/(cos(x-theta)-1))
  return(hessian)
}

theta0 <- asin(mean(x)-pi)

newton1 <- newtonRaphson(fun=function(theta) gradient(theta), x0=theta0,
                         dfun=function(theta) hessian(theta))
  root1 <- newton1$root

table1 <- data.frame(Theta=theta0, Root=root1)

kable(table1, booktabs = TRUE, align = 'c', row.names = 1)
```

	Theta	Root
1	0.0953941	0.0031182

So the MLE for θ using the Newton-Raphson method with initial value $\theta_0 = \tilde{\theta}_n$ is 0.0031.

1.4 Find MLE at $\theta_0 = \pm 2.7$

```
theta2 <- c(-2.7, 2.7)
newton2 <- vector("list", length = length(theta2))
root2 <- array(NA, dim=length(theta2))
for (i in 1:length(theta2)){
  newton2[[i]] <- newtonRaphson(fun=function(theta) gradient(theta),
                               dfun=function(theta) hessian(theta), x0=theta2[i])
  root2[i] <- newton2[[i]]$root
}

table2 <- data.frame(Theta=theta2, Root=root2)
kable(table2, booktabs = TRUE, align = 'c', row.names = 1)
```

	Theta	Root
1	-2.7	-2.668857
2	2.7	2.848415

So the MLE for θ at $\theta_0 = -2.7$ is -2.6689, and the MLE at $\theta_0 = 2.7$ is 2.8484.

1.5 200 Values Optimization Between $-\pi$ and π

```
dff <- 2*pi/199
theta3 <- array(-pi, dim = 200)
for (i in 2:200)
  theta3[i] <- theta3[as.numeric(i-1)]+dff

newton3 <- vector("list", length = length(theta3))
root3 <- array(NA, dim=length(theta3))

for (i in 1:length(theta3)){
  newton3[[i]] <- newtonRaphson(fun=function(theta) gradient(theta),
                               dfun=function(theta) hessian(theta), x0=theta3[i])
  root3[i] <- newton3[[i]]$root
}
  table3 <- data.frame(Theta=theta3, Root=root3)

temp <- as.data.frame(table(table3$Root))
mult.group <- vector("list", length(temp$Freq))
pos <- 0

for (i in 1:length(temp$Freq)) {
  for (j in 1:temp$Freq[i]){
  mult.group [[i]][j] <- table3$Theta[pos + j]
  }
```

```r
  pos <- pos + temp$Freq[i]
}
```

So the group of initial value, whose local maximam are the same, should be:

```r
print(mult.group)
```

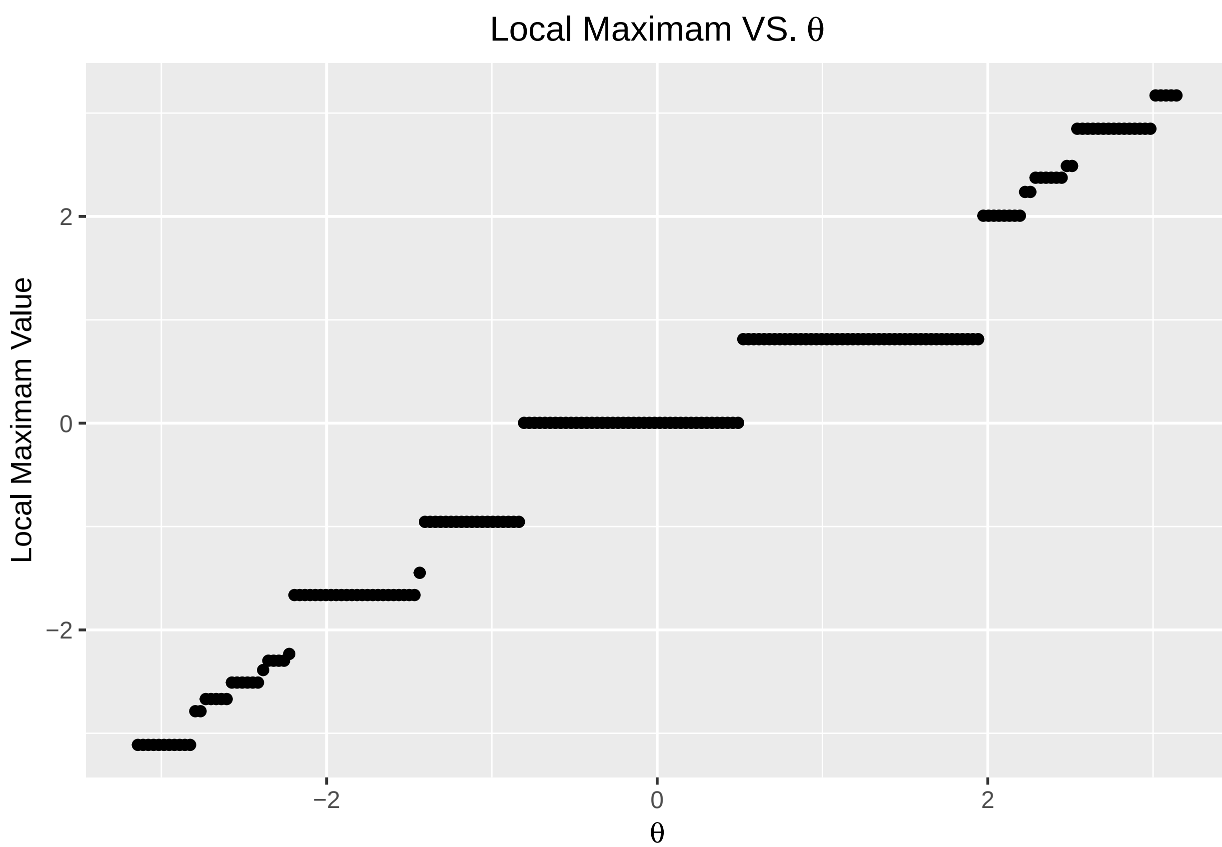
```
## [[1]]
##  [1] -3.141593 -3.110019 -3.078445 -3.046871 -3.015297 -2.983724 -2.952150
##  [8] -2.920576 -2.889002 -2.857428 -2.825855
##
## [[2]]
## [1] -2.794281
##
## [[3]]
## [1] -2.762707
##
## [[4]]
## [1] -2.731133 -2.699560 -2.667986 -2.636412 -2.604838
##
## [[5]]
## [1] -2.573264 -2.541691 -2.510117 -2.478543 -2.446969 -2.415395
##
## [[6]]
## [1] -2.383822
##
## [[7]]
## [1] -2.352248 -2.320674
##
## [[8]]
## [1] -2.289100 -2.257526
##
## [[9]]
## [1] -2.225953
##
## [[10]]
##  [1] -2.194379 -2.162805 -2.131231 -2.099657 -2.068084 -2.036510 -2.004936
##  [8] -1.973362 -1.941788 -1.910215 -1.878641 -1.847067 -1.815493 -1.783919
## [15] -1.752346 -1.720772 -1.689198 -1.657624 -1.626050 -1.594477 -1.562903
## [22] -1.531329 -1.499755 -1.468181
##
## [[11]]
## [1] -1.436608
##
## [[12]]
## [1] -1.405034 -1.373460 -1.341886 -1.310313 -1.278739 -1.247165 -1.215591
## [8] -1.184017
```

```
##
## [[13]]
##  [1] -1.1524435 -1.1208697 -1.0892959 -1.0577221 -1.0261484 -0.9945746
##  [7] -0.9630008 -0.9314270 -0.8998532 -0.8682794 -0.8367056
##
## [[14]]
## [1] -0.8051318
##
## [[15]]
## [1] -0.7735580 -0.7419842
##
## [[16]]
## [1] -0.7104104
##
## [[17]]
## [1] -0.6788366 -0.6472628
##
## [[18]]
## [1] -0.615689
##
## [[19]]
## [1] -0.5841152
##
## [[20]]
## [1] -0.5525414
##
## [[21]]
## [1] -0.5209676
##
## [[22]]
## [1] -0.4893938
##
## [[23]]
## [1] -0.4578200 -0.4262462
##
## [[24]]
## [1] -0.3946724 -0.3630986
##
## [[25]]
## [1] -0.3315249
##
## [[26]]
## [1] -0.2999511 -0.2683773 -0.2368035
##
## [[27]]
## [1] -0.2052297 -0.1736559 -0.1420821
##
## [[28]]
```

```
## [1] -0.11050828 -0.07893449
##
## [[29]]
## [1] -0.04736069 -0.01578690  0.01578690
##
## [[30]]
## [1] 0.04736069 0.07893449
##
## [[31]]
## [1] 0.1105083
##
## [[32]]
## [1] 0.1420821 0.1736559 0.2052297 0.2368035 0.2683773
##
## [[33]]
## [1] 0.2999511
##
## [[34]]
## [1] 0.3315249 0.3630986
##
## [[35]]
## [1] 0.3946724
##
## [[36]]
## [1] 0.4262462
##
## [[37]]
## [1] 0.45782
##
## [[38]]
## [1] 0.4893938
##
## [[39]]
## [1] 0.5209676
##
## [[40]]
##  [1] 0.5525414 0.5841152 0.6156890 0.6472628 0.6788366 0.7104104 0.7419842
##  [8] 0.7735580 0.8051318 0.8367056 0.8682794 0.8998532 0.9314270 0.9630008
## [15] 0.9945746 1.0261484 1.0577221 1.0892959 1.1208697 1.1524435 1.1840173
## [22] 1.2155911 1.2471649 1.2787387 1.3103125 1.3418863 1.3734601 1.4050339
## [29] 1.4366077 1.4681815 1.4997553 1.5313291 1.5629029 1.5944767 1.6260505
## [36] 1.6576243 1.6891981 1.7207719 1.7523457 1.7839194 1.8154932 1.8470670
## [43] 1.8786408 1.9102146 1.9417884
##
## [[41]]
## [1] 1.973362 2.004936 2.036510 2.068084 2.099657 2.131231 2.162805 2.194379
##
## [[42]]
```

```
## [1] 2.225953
##
## [[43]]
## [1] 2.257526
##
## [[44]]
## [1] 2.289100 2.320674 2.352248 2.383822 2.415395 2.446969
##
## [[45]]
## [1] 2.478543
##
## [[46]]
## [1] 2.510117
##
## [[47]]
##  [1] 2.541691 2.573264 2.604838 2.636412 2.667986 2.699560 2.731133
##  [8] 2.762707 2.794281 2.825855 2.857428 2.889002 2.920576 2.952150
## [15] 2.983724
##
## [[48]]
## [1] 3.015297 3.046871 3.078445 3.110019 3.141593
```

```r
ggplot(table3, aes(x = theta3, y = root3)) +
  geom_point()+
  ggtitle("Local Maximam VS."~theta)+
  theme(plot.title = element_text(hjust = 0.5))+
  labs(y="Local Maximam Value", x=expression(theta))
```



2 Modeling Beetle Data

2.1 Population Growth Model - Gauss-Newton Approach

```
beetles <- data.frame(
    days    = c(0,  8,  28,  41,  63,  69,   97, 117,  135,  154),
    beetles = c(2, 47, 192, 256, 768, 896, 1120, 896, 1184, 1024))

bt <- list(
t <- beetles$days,
y <- beetles$beetles
)

fo <- y~2*K/(2+(K-2)*exp(-r*t))
nls(fo, data=bt, start = list(K=1000, r=1))
```

```
## Nonlinear regression model
##   model: y ~ 2 * K/(2 + (K - 2) * exp(-r * t))
##    data: bt
##         K         r
```

```
## 1049.4068    0.1183
##  residual sum-of-squares: 73420
##
## Number of iterations to convergence: 9
## Achieved convergence tolerance: 5.892e-06
```

The fitted model should be:

$$f(t) = \frac{2098.8136}{2 + 1047.4068\exp(-0.1183t)}$$

where the minimized sum of squared error should be 73420

```r
K <- seq(500, 1500, by=5)
r <- seq(0, 1, by=0.005)


z <- matrix(NA, nrow = 201, ncol = 201)
for (i in 1:201){
  for (j in 1:201){
    z[i, j] <- sum((y-2*K[i]/(2+(K[i]-2)*exp(-r[j]*t)))^2)
  }
}
contour(K, r, z, xlab = "r", ylab = "K")
```