Homework 2 (Part 1)

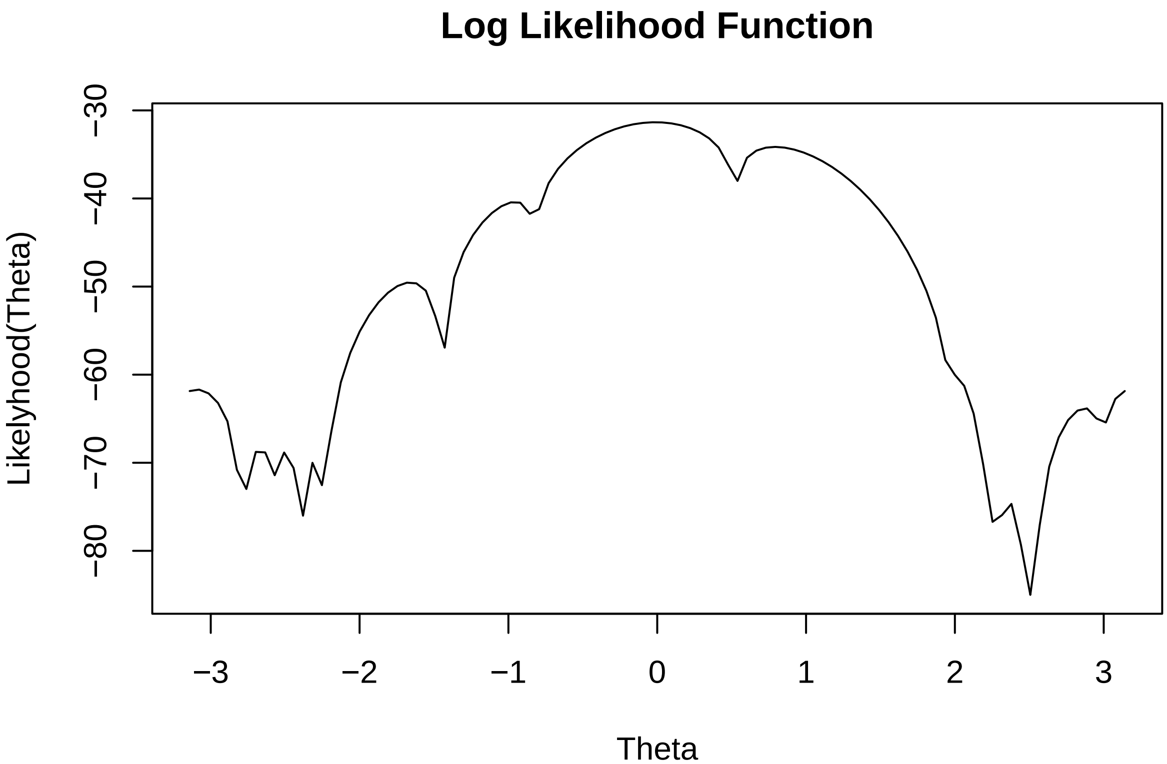
Michael Pena

2024-02-04

(a)

```r
# placing in the information


# building the functions
f <- function(theta){
  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, 2.50)
  llh <- 0
  for (i in x){
    llh <- llh + log(
      (1 - cos(i - theta))/(2*pi)
      )
  }
  return(llh)
}

# setup sequences
theta_vec <- seq(-pi,pi,length=100)

# setup y-values
y <-  f(theta_vec)
```

```r
# render plot
plot(theta_vec, y, type = 'l' ,main = "Log Likelihood Function", xlab = "Theta", ylab = "Likelyhood(The
```



(b).

$$\bar{x} = \frac{1}{2\pi}\int_0^{2\pi} x(1 - cos(x - \tilde{\theta}))dx = sin(\tilde{\theta}) + \pi$$

thus $\tilde{\theta} = \arcsin[\bar{x} - \pi]$

(c).

$$\ell(\theta) = \sum_{i=1}^{n} \ln[\frac{1}{2\pi}(1 - \cos(x_i - \theta))]$$

$$\ell'(\theta) = \sum_{i=1}^{n} \frac{2\pi\sin(x_i - \theta)}{\cos(x_i - \theta) - 1}$$

$$\ell''(\theta) = \sum_{i=1}^{n} \frac{2\pi}{\cos(x_i - \theta) - 1}$$

```
# setup newtons method
ell <- function(theta){
  S <- 0
   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, 2.50)
  for (i in x){
    S <- S + log((1/2*pi)*(1-cos(i-theta)))
  }
  return(S)
}
```

```r
dell <- function(theta){
  S <- 0
   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, 2.50)
  for (i in x){
    S <- S + (2*pi*sin(i-theta))/(cos(i-theta)-1)
  }
  return(S)
}

ddell <- function(theta){
  S <- 0
   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, 2.50)
    for (i in x){
    S <- S + (2*pi)/(cos(i-theta)-1)
  }
  return(S)
}
```

```r
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, 2.50)

# newton's method
newtM <- function(theta0,ipo){
# initialization
init_theta <- theta0
it <- 0
stop <- 0
df <- data.frame()
# begin the while loop
while (it < 1000 & stop == 0){
  theta1 <- theta0 - (dell(theta0))/(ddell(theta0))
  it = it + 1
  absg <- abs(dell(theta0))
  mre <- abs(theta1 - theta0)/max(1,abs(theta1))
  row <- c(it,theta1,mre,absg,init_theta)
  df <- rbind(df, row)
  if (mre < 1*exp(-6) &  absg <  1*exp(-9)){
    stop <- 1
    theta0 <- theta1
  }
  else {
    stop  = 0
    theta0 <- theta1
  }
}
df <- data.frame(df) %>% set_names("Iteration","Theta","Relative Error","Gradient at Theta","Initial The
  Theta = round(Theta, digits = 12),
  `Relative Error` = sprintf("%.1e", `Relative Error`),
  `Gradient at Theta` = sprintf("%.1e", `Gradient at Theta`)
```

```
  )

df$final <- theta1
if(ipo == FALSE){
  df <- select(df, c("Iteration","Theta","Relative Error","Gradient at Theta"))
}

return(df)
}

# starting at MME
mme <- asin(mean(x) - pi)
newtM(mme, F)
```

```
##   Iteration        Theta Relative Error Gradient at Theta
## 1         1 -0.009098574        6.8e-02           1.0e+01
## 2         2 -0.011968738        2.9e-03           4.0e-01
## 3         3 -0.011972002        3.3e-06           4.5e-04
## 4         4 -0.011972002        4.1e-12           5.7e-10
```

```
# starting at 2.7
newtM(2.7,F)
```

```
##   Iteration    Theta Relative Error Gradient at Theta
## 1         1 2.825724        4.4e-02           2.4e+02
## 2         2 2.877549        1.8e-02           6.5e+01
## 3         3 2.873184        1.5e-03           7.2e+00
## 4         4 2.873095        3.1e-05           1.4e-01
## 5         5 2.873095        1.2e-08           5.5e-05
```

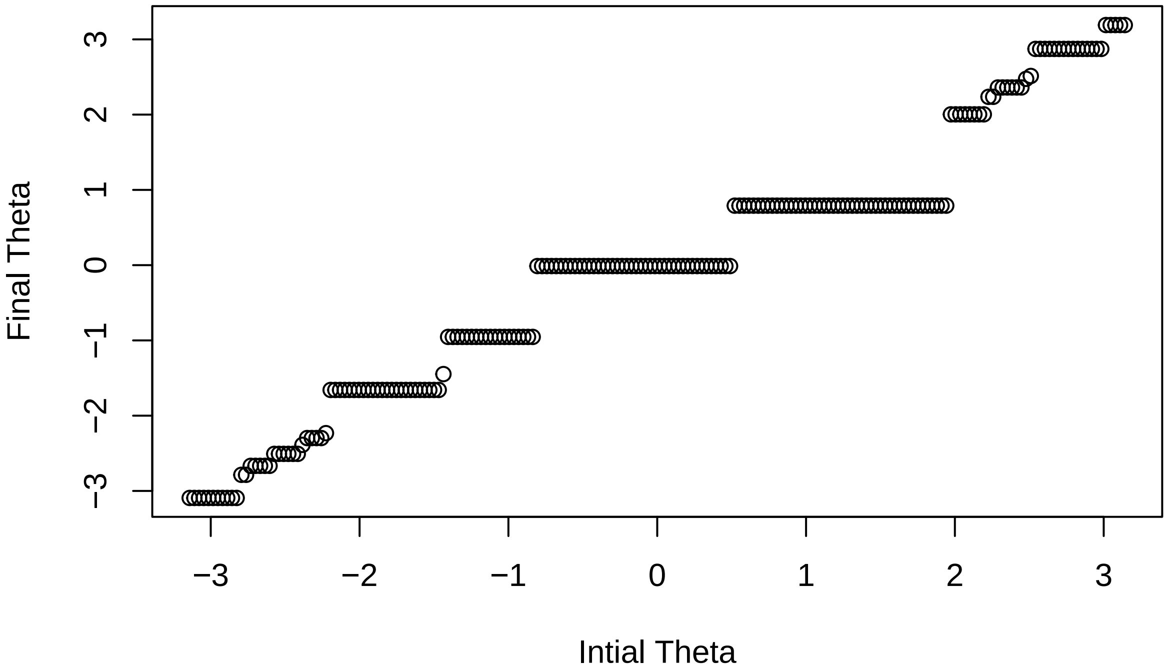
```
#starting at -2.7
newtM(-2.7,F)
```

```
##   Iteration     Theta Relative Error Gradient at Theta
## 1         1 -2.674114        9.7e-03           1.8e+02
## 2         2 -2.666794        2.7e-03           3.5e+01
## 3         3 -2.666700        3.5e-05           4.4e-01
## 4         4 -2.666700        3.9e-10           4.8e-06
```

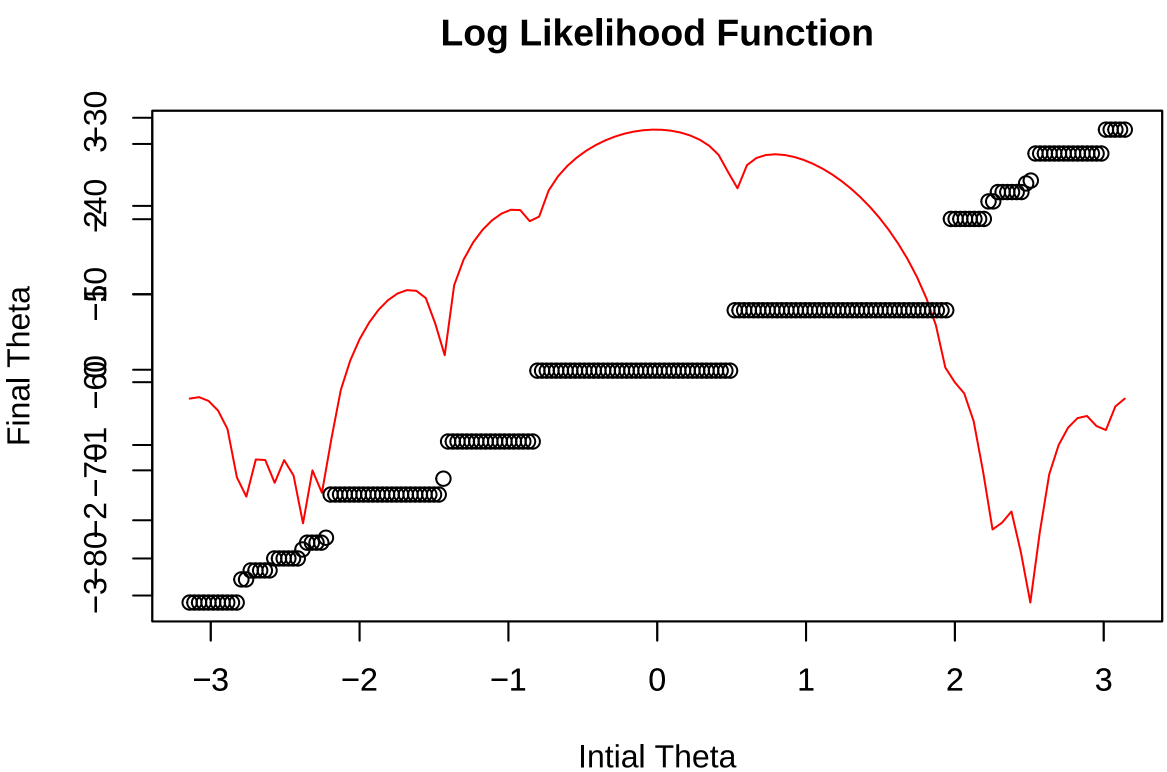
(d).

```
# setting up theta
init_thetas <- seq(-pi,pi,length = 200)

#intialize dataframe
df1 <- data.frame()

# make dataset to graph later
for (i in 1:200){
  df1 <- bind_rows(df1,newtM(init_thetas[i],T))
}

# clean data
df1 <- select(df1,c("Initial Theta",final))
df1 <- distinct(df1)
```

```r
# render graph
plot(df1$`Initial Theta`,df1$final, xlab = "Intial Theta", ylab = "Final Theta")
```




```r
plot(theta_vec, y, type = 'l' ,main = "Log Likelihood Function", xlab = "", ylab = "",col = 'red')
par(new = TRUE)
plot(df1$`Initial Theta`,df1$final, xlab = "Intial Theta", ylab = "Final Theta")
```



```r
par(new = FALSE)
```

when we superimpose the Log-Likelyhood Function over the last graph we made, it appears that a single line segment of Initial Theta's will be as long as a concave down curve on the Log-Likelyhood function.