

HW2

Jiang Bian

08 February 2018

Q1

a

According to the question, we know that

$$p(x; \theta) = \frac{1}{\pi[1 + (x - \theta)^2]}$$

Because x_1, \dots, x_n is an i.i.d. sample and $l(\theta)$ the log-likelihood function of θ based on the sample. Therefore,

$$\begin{aligned} l(\theta) &= \ln\left(\prod_{i=1}^n p(x_i; \theta)\right) \\ &= \ln\left(\prod_{i=1}^n \frac{1}{\pi[1 + (x_i - \theta)^2]}\right) \\ &= \sum_{i=1}^n \ln\left(\frac{1}{\pi[1 + (x_i - \theta)^2]}\right) \\ &= -n \ln \pi - \sum_{i=1}^n \ln[1 + (\theta - x_i)^2] \end{aligned}$$

And,

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

As for $I(\theta)$,

$$\begin{aligned} I(\theta) &= n \int \frac{\{p'(x)\}^2}{p(x)} dx \\ p'(x) &= \frac{-2(x - \theta)}{\pi[1 + (x - \theta)^2]^2} \end{aligned}$$

Thus,

$$I(\theta) = \frac{4n}{\pi} \int_{-\infty}^{\infty} \frac{x^2 dx}{(1 + x^2)^3}$$

Suppose $x = \tan(t)$ where $t \in (-\frac{\pi}{2}, \frac{\pi}{2})$,

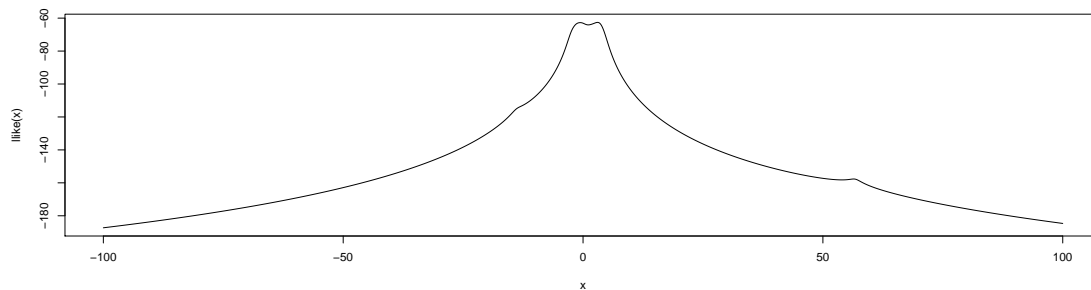
$$\begin{aligned}
I(\theta) &= \frac{4n}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{\tan^2(t) d(\tan(t))}{(1 + \tan^2(t))^3} \\
&= \frac{4n}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{\tan^2(t)}{\left(\frac{1}{\sec^2(t)}\right)^3} \left(\frac{1}{\sec^2(t)}\right) dt \\
&= \frac{4n}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{\sin^2(t)}{\cos^2(t)} \cos^4(t) dt \\
&= \frac{4n}{\pi} \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \sin^2(t) \cos^2(t) dt \\
&= \frac{4n}{\pi} * \frac{\pi}{8} = \frac{n}{2}
\end{aligned}$$

So, $I(\theta) = \frac{n}{2}$.

b

The graph of log-likelihood function.

```
###q1b
x <- c(1.77,-0.23,2.76,3.80,3.47,56.75,-1.34,4.24,-2.44,
       3.29,3.71,-2.40,4.53,-0.07,-1.05,-13.87,-2.53,-1.75)
#l(theta)
llikeli <- function(theta,x. = x){
  sum(-log(pi)-log(1+(x. - theta)^2))
}
llike <- Vectorize(llikeli)
#graph the log-likelihood function, limit from -100 to 100
curve(llike, from = -100, to = 100, n = 10000)
```



The results of using Newton's method is shown below.

```
counts_1b
```

```
## [1] 24  4  5  5  4  5 25 26 22
```

```
lvals_1b
```

```
## [1] -682.61211 -62.73186 -62.73186 -64.10186 -62.59032 -62.73186
```

```
## [7] -676.10197 -678.15143 -673.45284
```

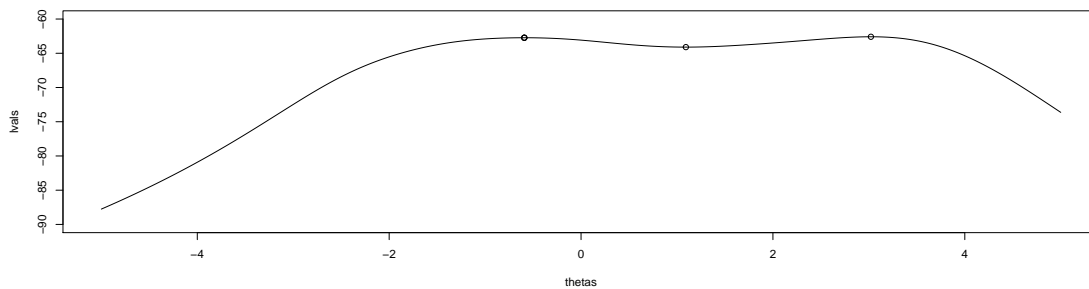
```
#determine the range of the plot  
llikeli(-5, x)
```

```
## [1] -87.75915
```

```
max(lvals_1b)
```

```
## [1] -62.59032
```

```
#plotting  
plot(thetas_1b, lvals_1b, "p", xlim = c(-5, 5),  
      ylim = c(-90, -60), xlab = c("thetas"), ylab = c("lvals"))  
curve(llike, from = -5, to = 5, n = 1000, add = TRUE)
```



When take sample mean as the start point.

```
counts_1bb
```

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

```
lvals_1bb
```

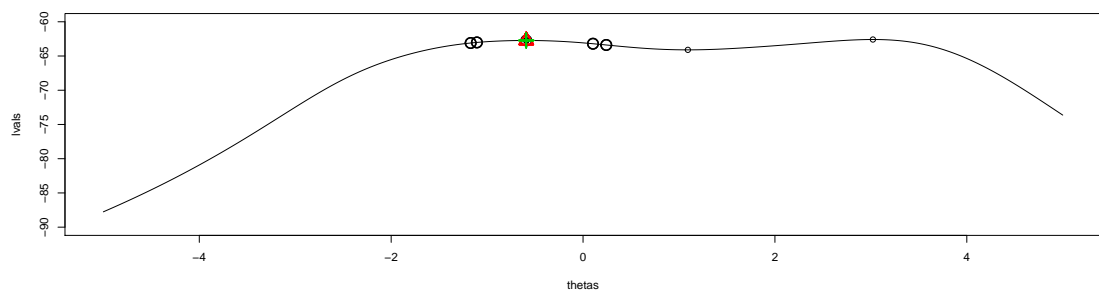
```
## [1] -62.59032
```

When using sample mean as a start point, the count_1bb is 5 and the l value is closer to maximum value, so it is a good start point.

c

The added points of fixed-point method is shown in the graph.

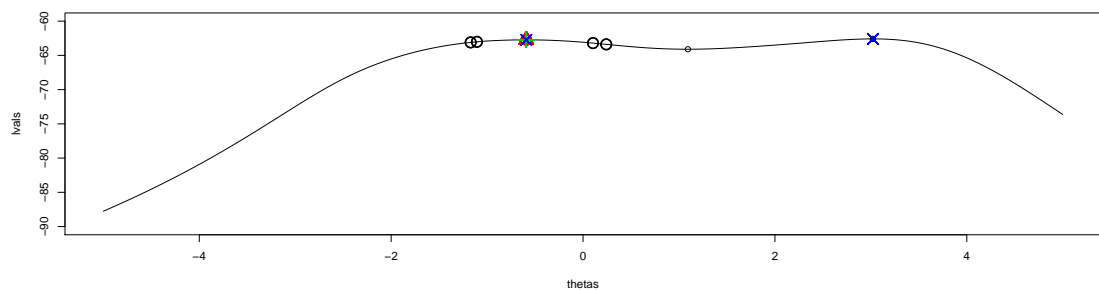
```
plot(thetas_1b, lvals_1b, "p", xlim = c(-5, 5),  
      ylim = c(-90, -60), xlab = c("thetas"), ylab = c("lvals"))  
curve(llike, from = -5, to = 5, n = 1000, add = TRUE)  
typ <- seq(1, length(alphas), 1)  
for (m in 1:length(typ)){  
  points(thetas_1c[,m], lvals_1c[,m], pch=typ[m], col=typ[m], cex=2, lwd=2)  
}
```



d

The added points of Fisher Scoring is shown in the graph.

```
plot(thetas_1b,lvals_1b, "p", xlim = c(-5, 5),
      ylim = c(-90, -60), xlab = c("thetas"), ylab = c("lvals"))
curve(llike, from = -5, to = 5, n = 1000, add = TRUE)
typ <- seq(1,length(alphas),1)
for (m in 1:length(typ)){
  points(thetas_1c[,m], lvals_1c[,m], pch=typ[m], col=typ[m], cex=2, lwd=2)
}
points(thetas_1dn,lvals_1dn, pch=length(typ)+1, col=length(typ)+1, cex=2, lwd=2)
```



e

Comparing the counts number of three different methods.

```
counts_1b
```

```
## [1] 24  4  5  5  4  5 25 26 22
```

```
knitr::kable(counts_1c)
```

	1	0.64	0.25
-11	218	1	1
-1	10000	12	1

	1	0.64	0.25
0	10000	11	1
1.5	10000	12	1
4	10000	11	1
4.7	10000	10	1
7	10000	10	1
8	10000	12	1
38	10000	12	1

```
counts_1d
```

```
## [1] 80 57 59 39 27 28 30 32 220
```

And comparing the thetas of three different methods.

```
thetas_1b
```

```
## [1] -9.688959e+07 -5.914735e-01 -5.914735e-01 1.092730e+00 3.021345e+00
## [6] -5.914735e-01 8.086125e+07 8.559820e+07 7.512458e+07
```

```
knitr::kable(thetas_1c)
```

	1	0.64	0.25
-11	-0.5914740	-0.5914734	-0.5914735
-1	-1.1713919	-0.5914736	-0.5914736
0	0.1035079	-0.5914736	-0.5914736
1.5	-1.1713919	-0.5914736	-0.5914736
4	0.1035079	-0.5914736	-0.5914736
4.7	0.2417269	-0.5914734	-0.5914735
7	0.2417269	-0.5914734	-0.5914735
8	-1.1063091	-0.5914736	-0.5914735
38	-1.1063091	-0.5914736	-0.5914735

```
thetas_1d
```

```
## [1] -0.5914739 -0.5914739 -0.5914732 3.0213453 3.0213456 3.0213455
## [7] 3.0213456 3.0213456 3.0213456
```

By comparing the results we achieved from above methods, we found that Newton-Raphson method has more than half of the results faster than other methods except the results when alpha 0.25 in fixed-point iterations. But when using other start points, the counts will be more than 20 and the theta is divergent to infinite, so the stability of Newton-Raphson is not as good as other methods.

In fixed-point iterations, it depends strongly on the value of alpha, when $\alpha = 1$, the counts reach to the maximum counts limit.

In Fisher Scoring, it can make rapid improvement but using more times than Newton's method.

Q2

a

Known the probability density with parameter is

$$p(x; \theta) = \frac{1 - \cos(x - \theta)}{2\pi}$$

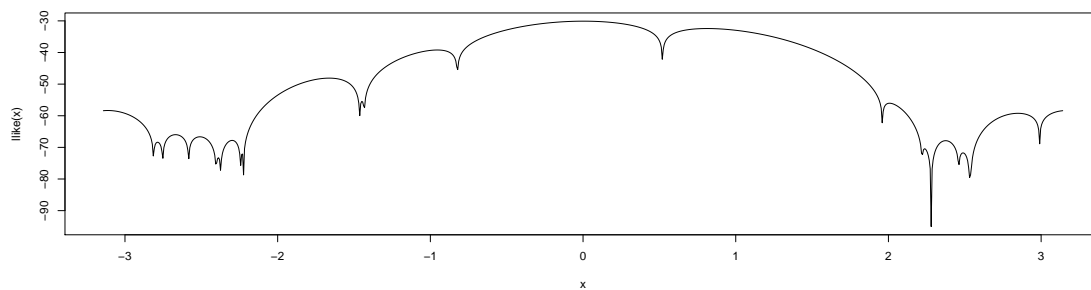
The log-likelihood function of θ is

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

Therefore, the graph of the function between $-\pi$ and π is,

```
rm(list=ls())
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)

#l(theta)
llikeli <- function(theta,x. = x){
  sum(-log(2*pi)+log(1-cos(x. - theta)))
}
llike <- Vectorize(llikeli)
#graph the log-likelihood function, limit from -pi to pi
curve(llike, from = -pi, to = pi, n = 1000)
```



b

Known

$$\begin{aligned}
 E[X|\theta] &= \int_0^{2\pi} x * \frac{1 - \cos(x - \theta)}{2\pi} dx \\
 &= \int_0^{2\pi} \frac{x}{2\pi} dx - \int_0^{2\pi} \frac{x * \cos(x - \theta)}{2\pi} dx \\
 &= \frac{1}{2\pi} * \frac{x^2}{2} \Big|_0^{2\pi} - \int_0^{2\pi} \frac{x}{2\pi} d[\sin(x - \theta)] \\
 &= \pi - \frac{x}{2\pi} * \sin(x - \theta) \Big|_0^{2\pi} - \int_0^{2\pi} \sin(x - \theta) d\left(\frac{x}{2\pi}\right) \\
 &= \pi - \sin(2\pi - \theta) - \frac{1}{2\pi} \int_{-\theta}^{2\pi-\theta} \sin u du \\
 &= \pi + \sin \theta + \frac{1}{2\pi} \cos u \Big|_{-\theta}^{2\pi-\theta} \\
 &= \pi + \sin \theta
 \end{aligned}$$

the method-of-moments estimator of θ is,

```
###q2b
estimator <-function(theta, xbar){
  pi+sin(theta)-xbar
}
print(root1<-uniroot(estimator,c(0,pi/2),xbar=mean(x))$root)
```

```
## [1] 0.09539388
```

```
print(root2<-uniroot(estimator,c(pi/2,pi),xbar=mean(x))$root)
```

```
## [1] 3.046199
```

c & d

Set the required thetas as starting points from former question and requirements.

```
#starting points
startp <- c(root1, root2, -2.7, 2.7)
```

We received,

```
knitr::kable(thetas_2cd)
```

0.1	3.05	-2.7	2.7
0.0031182	3.170715	-2.668858	2.848415

```
knitr::kable(counts_2cd)
```

0.1	3.05	-2.7	2.7
4	6	5	5

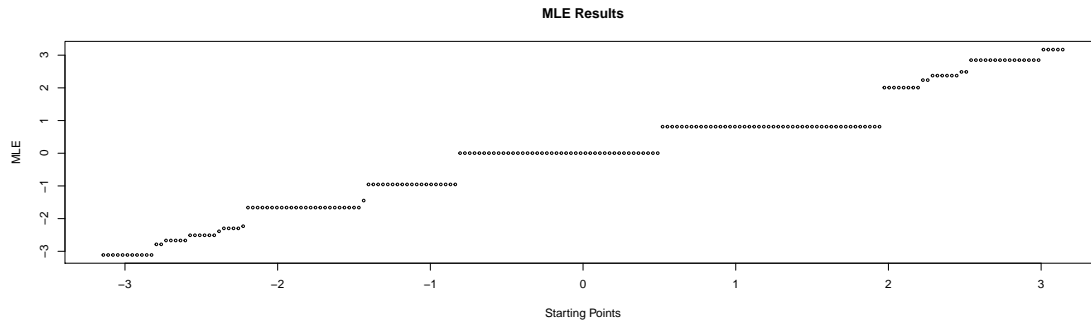
```
knitr::kable(lvals_2cd)
```

0.1	3.05	-2.7	2.7
-30.09493	-58.34181	-65.98183	-59.22775

e

The graph of the MLE results is shown below.

```
plot(starttp, thetas_2e, main='MLE Results', cex = 0.5,
     ylab='MLE', xlab='Starting Points', pch = 1)
```



To divide the set of starting values into separate groups, with each group corresponding to a separate unique outcome of the optimization. We use the “unique” function to receive the index (the start value) of each group.

```
thetas_2ee <- as.data.frame(round(t(thetas_2e), digits = 2))
colnames(thetas_2ee) <- c("MLE")
uniq <- unique(thetas_2ee)
knitr::kable(uniq)
```

MLE	
-3.14	-3.11
-2.79	-2.79
-2.73	-2.67
-2.57	-2.51
-2.38	-2.39
-2.35	-2.30
-2.23	-2.23
-2.19	-1.66
-1.44	-1.45
-1.41	-0.95

MLE	
-0.81	0.00
0.52	0.81
1.97	2.01
2.23	2.24
2.29	2.37
2.48	2.49
2.54	2.85
3.02	3.17

use `split()` instead of using `unique()` to group up.

```
grp <- split(startp, round(thetas_2e, digits = 4))
minl <- sapply(grp, min)
maxl <- sapply(grp, max)
fingrp <- round(cbind(minl, maxl), digits = 4)
colnames(fingrp) <- c("Lower Limit", "Upper Limit")
knitr::kable(fingrp)
```

	Lower Limit	Upper Limit
-3.1125	-3.1416	-2.8259
-2.7866	-2.7943	-2.7627
-2.6689	-2.7311	-2.6048
-2.5094	-2.5733	-2.4154
-2.3883	-2.3838	-2.3838
-2.2979	-2.3522	-2.2575
-2.2322	-2.2260	-2.2260
-1.6627	-2.1944	-1.4682
-1.4475	-1.4366	-1.4366
-0.9544	-1.4050	-0.8367
0.0031	-0.8051	0.4894
0.8126	0.5210	1.9418
2.0072	1.9734	2.1944
2.237	2.2260	2.2575
2.3747	2.2891	2.4470
2.4884	2.4785	2.5101
2.8484	2.5417	2.9837
3.1707	3.0153	3.1416

Q3

a

```
rm(list=ls())
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))

errors <- rep(NA, nrow(beetles))

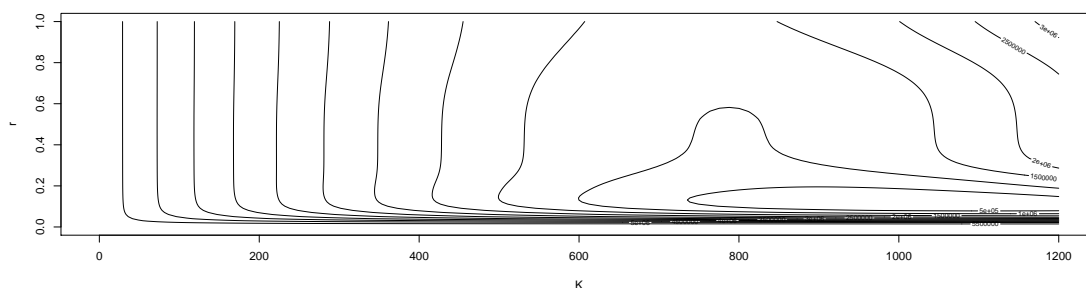
nls(beetles ~ (K*beetles[1])/(beetles[1]+(K-beetles[1])*exp(-r*days)),
    start = list(K = 1000, r = 1), data = beetles, trace = TRUE))

## 1986152 : 1000      1
## 951504.1 : 792.000022 0.3313653
## 316776.6 : 812.6160939 0.1314344
## 142639.1 : 920.0366145 0.1255537
## 73781.4 : 1039.9824403 0.1191475
## 73424.36 : 1048.7520644 0.1184275
## 73419.83 : 1049.3061406 0.1182953
## 73419.7 : 1049.390377 0.118273
## 73419.7 : 1049.4043999 0.1182692
## 73419.7 : 1049.4067631 0.1182685

## Nonlinear regression model
## model: beetles ~ (K * beetles[1])/(beetles[1] + (K - beetles[1]) * exp(-r * days))
## data: beetles
##      K      r
## 1049.4068 0.1183
## residual sum-of-squares: 73420
##
## Number of iterations to convergence: 9
## Achieved convergence tolerance: 5.892e-06
```

We received $K = 1049.4068$, $r = 0.1183$, residual sum-of-squares = 73420.

b



c

We first write the codes for the partial derivatives about K, r, sigma.

Then determine the initial value of $K = 1050$, $r = 0.12$, and $\sigma = 0.5$.

```
krs <- matrix(c(1050, 0.12, 0.5))
row.names(krs) <- c("K", "r", "sigma")
knitr::kable(krs)
```

K	1050.00
r	0.12
sigma	0.50

By using the Gauss-Newton method, after 8 times, we received the maximum likelihood estimators of $\{K, r, \sigma\}$ and $\{K, r, \sigma^2\}$

```
## [1] 8
```

	K	r	sigma
	820.38	0.1926401	0.6440842

	K	r	sigma2
	820.3801	0.19264	0.4148444

From the previous knowledge, we know that the information matrix of parameter estimates is the negative inverse matrix of fisher information, and the values on the diagonal of the information matrix are the variance of parameter estimates.

```
vari <- solve(-M)
colnames(vari) <- row.names(vari) <- c("K", "r", "sigma")
knitr::kable(vari)
```

	K	r	sigma
K	62464.5546683	-9.0597875	0.0000049
r	-9.0597875	0.0039784	0.0000000
sigma	0.0000049	0.0000000	0.0207422