Zhao Ping HW2

Ping Zhao

2/10/2018

Problem 1

(a) Gradient Function

```
#x=(x1,x2,x3,x4), and t os the parameter (theta)
grad <- function (x, t) {
  dl = x[1]/(2+t)-(x[2]+x[3])/(1-t)+x[4]/t
}
```

(b)Stop your algorithm when either of the following criteria is satisfied:

```
secant <- function(maxit, x, t0, t1, tolerr, tolgrad){
  tstar = -1657/7680+sqrt(3728689)/7680
  digits = -log10(abs(t1 - tstar))
  it=0
  stop=0

  theta=c()
  converage.rate=c()
  digit=c()
  iteration=c()
  modified.relerr=c()
  Gradients=c()
  while (it < maxit & stop==0){
      it = it+1
      dl.1 = grad(x,t1)
      dl.0 = grad(x,t0)

      tnew = t1-dl.1*(t1-t0)/(dl.1-dl.0)


      rate = abs(tnew - tstar)/abs(t1 - tstar) #check convergent rate
      digits = -log10(abs(tnew - tstar)/abs(tstar))
      mod.relerr = abs(t1 - tnew)/max(1,abs(tnew))
      dl.new <-grad(x,tnew)

      if (mod.relerr < tolerr & abs(dl.new) < tolgrad) stop=1 # Stop iteration condition
      #print(c(it, t, rate,digits))
      #print(sprintf('it = %2.0f   teta = %12.12f    Rate = %4.2f     digits = %2.1f',it,tnew,rate,digi
      #print(sprintf('          relerr = %4.1e,        grad = %4.1e',relerr,dl))
      t0 = t1 # Update t0
      t1 = tnew # Update and return

      theta[it]<- tnew
      converage.rate[it]<- rate
```

```r
      digit[it]<-digits
      iteration[it]<- it
      modified.relerr[it] <-mod.relerr
      Gradients[it]<-dl.1
    }
  conver.info<-as.data.frame(cbind(iteration, theta, converage.rate, digit, modified.relerr, Gradients))
  return(conver.info)
}

a=c()
```

(c) &(d)

$\theta^{(0)} = .02$ and $\theta^{(1)} = .01$, and use tolerr = 1e-6 and tolgrad=1e-9

The iteration information is in matrix m. A larger matrix m.index contains information of starting points.

```r
d <- c(1997,907,904,32)

#secant(20,x=d,0.02,0.01,1e-6,1e-9 )


m<-secant(20,x=d,0.02,0.01,1e-6,1e-9)
m
```

```
##   iteration      theta converage.rate      digit modified.relerr
## 1         1 0.02456185   4.328318e-01  0.5065360    1.456185e-02
## 2         2 0.02782321   7.065220e-01  0.6574103    3.261365e-03
## 3         3 0.03335105   2.959467e-01  1.1861968    5.527834e-03
## 4         4 0.03519756   2.053261e-01  1.8737526    1.846511e-03
## 5         5 0.03564619   5.967468e-02  3.0979625    4.486269e-04
## 6         6 0.03567431   1.218047e-02  5.0122985    2.812386e-05
## 7         7 0.03567466   7.259626e-04  8.1513843    3.465340e-07
## 8         8 0.03567466   8.819921e-06 13.2059196    2.517513e-10
##      Gradients
## 1 2.364239e+03
## 2 4.326183e+02
## 3 2.720887e+02
## 4 6.813010e+01
## 5 1.331728e+01
## 6 7.855955e-01
## 7 9.562060e-03
## 8 6.941636e-06
```

The matrix above gives iteration information about the secant method.

However, I need to add information of the starting point θ_0 and θ_1 to the convergence matrix. As we are applying secant method, the θ_2 is from the first iteration. (I don't think this index system is good.) I added an column of θ index to the matrix.

```r
Index.Theta<-seq(0,nrow(m)+1)
tstar = -1657/7680+sqrt(3728689)/7680

#mod.relerr.0 = abs(0.02 - tnew)/max(1,abs(tnew))
mod.relerr.1 = abs(0.01 - 0.02)/max(1,abs(0.01))
```

```r
rate.t1 = abs(0.01 - tstar)/abs(0.02 - tstar)

digits.1 = -log10(abs(0.01 - tstar)/abs(tstar))
digits.0 = -log10(abs(0.02 - tstar)/abs(tstar))

t0.info<-c(NA,0.02,NA,digits.0, NA, grad(d,0.02))
t1.info<-c(NA,0.01,rate.t1,digits.1,mod.relerr.1,grad(d,0.01))

m.starting<-rbind(t0.info,t1.info,m)
m.index<-cbind(Index.Theta,m.starting)

m.index
```

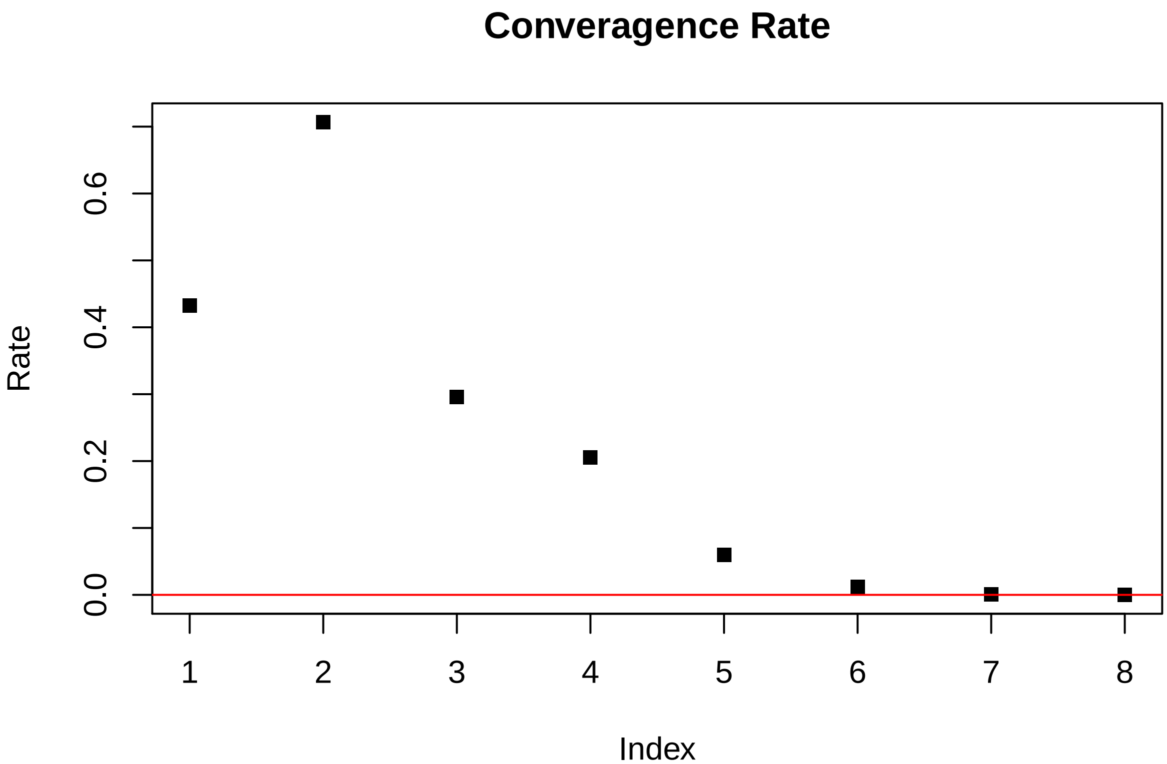
```
##    Index.Theta iteration      theta converage.rate      digit
## 1            0        NA 0.02000000             NA  0.3571618
## 2            1        NA 0.01000000   1.637973e+00  0.1428552
## 3            2         1 0.02456185   4.328318e-01  0.5065360
## 4            3         2 0.02782321   7.065220e-01  0.6574103
## 5            4         3 0.03335105   2.959467e-01  1.1861968
## 6            5         4 0.03519756   2.053261e-01  1.8737526
## 7            6         5 0.03564619   5.967468e-02  3.0979625
## 8            7         6 0.03567431   1.218047e-02  5.0122985
## 9            8         7 0.03567466   7.259626e-04  8.1513843
## 10           9         8 0.03567466   8.819921e-06 13.2059196
##    modified.relerr    Gradients
## 1               NA 7.406547e+02
## 2     1.000000e-02 2.364239e+03
## 3     1.456185e-02 2.364239e+03
## 4     3.261365e-03 4.326183e+02
## 5     5.527834e-03 2.720887e+02
## 6     1.846511e-03 6.813010e+01
## 7     4.486269e-04 1.331728e+01
## 8     2.812386e-05 7.855955e-01
## 9     3.465340e-07 9.562060e-03
## 10    2.517513e-10 6.941636e-06
```

(e)

```r
rate.sq<-m$converage.rate^2

plot(m$converage.rate, ylab = "Rate", main = "Converagence Rate", pch=15)

abline(h=0, col=2)
```

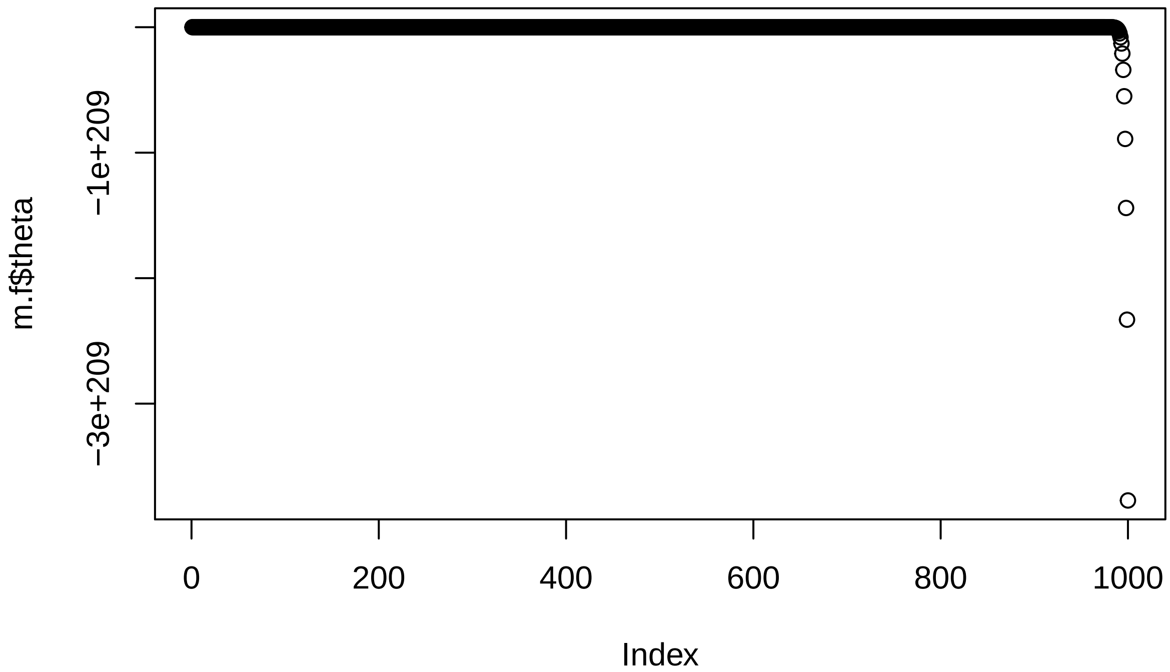


As we can see from the rate plot above, the rate is less than 1 and going to 0. Thus, it is supper-linear converagence.

(f)

$\theta^{(0)} = 0.5$ and $\theta^{(1)} = 0.01$

```
m.f<-secant(1000,x=d,0.5,0.01,1e-6,1e-6)
plot(m.f$theta)
```



From the result above, we can see that the secant method doesn't converage in this case.

Problem 2

(a) log likelihood function

$$l = -20 * log2\pi + \Sigma_1^{20} log(1 - cos(x_i - \theta))$$

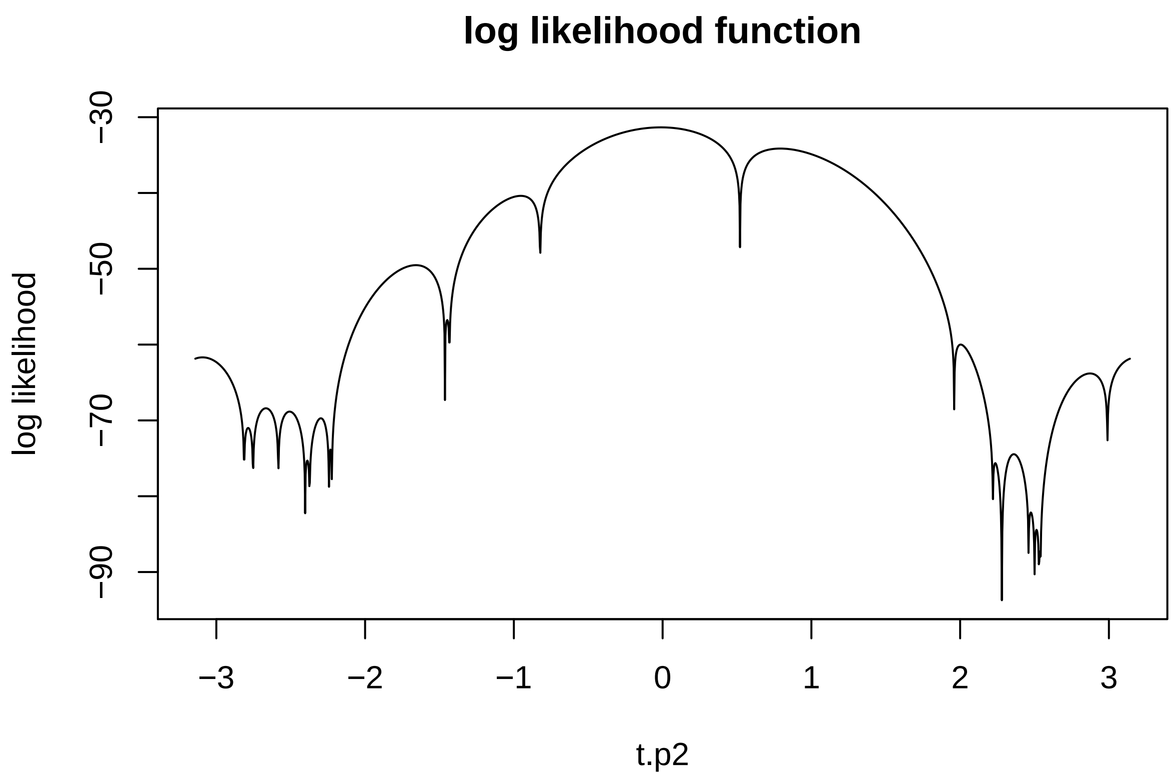
```
d.2<-c(3.91, 4.85, 2.28, 4.06, 3.70, 4.04, 5.46, 3.53, 2.28, 1.96, 2.53,
       3.88, 2.22, 3.47, 4.82, 2.46, 2.99, 2.54, 0.52, 2.50)

# Defind l as the log likelihood function
l<-function(t){sum(log(1-cos(d.2-t)))-20*log(2*pi)}

t.p2<-seq(-pi,pi,length.out = 2000)
y=c()

for(i in 1:2000){y[i]<-l(t.p2[i])}


plot(t.p2,y, type="l", main = "log likelihood function", ylab = "log likelihood")
```



(b) MOM Estimator

The first moment is

$$E(X) = \int_0^{2\pi} (1 - cos(x - \theta))/2\pi dx = 1$$

(c) Newton–Raphson method

The Newton-Raphson Method is updating θ via:

$$\theta^{(n+1)} = \theta^{(n)} - \frac{l'(\theta^{(n)})}{l''(\theta^{(n)})}$$

```
grad.l<-function(x,t){
  sum(sin(t-x)/(1-cos(t-x)))
}

hessian.2<-function(x,t){
  sum(cos(t-x)/(1-cos(t-x))-sin(t-x)^2/(1-cos(t-x))^2)
}
```