STA Homework 6

Chen Zihao 915490404

1.

(a)

$$\rho_{Y,Z} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}$$

Use Pearson correlation coefficient as the estimate of the correlation coefficient.

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

(b)

Estimate the standard error of the Pearson correlation coefficient.

using jackknife

$$\hat{se}_{jack} = \{\frac{n-1}{n}\sum[\hat{r}_{(i)} - \hat{r}_{(.)}]^2\}^{1/2}$$

we can get $\hat{se}_{jack}$ is

```
## [1] 0.2547034
```

using bootstrap

1. select B independent bootstrap samples
2. calculate the bootstrap replication corresponding to each bootstrap samples.
3. estimate the S.E. by the sample standard deviation of the B replicates.

we can get $\hat{se}_B$ is

```
## [1] 0.1978107
```

(c)

1. using "normal theory"

$\hat{\theta}_{(.)} \pm \Phi(0.975)\hat{se}_B$

the C.I is

```
## [1] 0.2097442 0.9851479
```

2. using bootstrap t-interval approaches.
3. generate B bootstrap samples
4. for each sample, calculate the Pearson correlation coefficient and estimated standard error (using bootstrap in the bootstrap, as in (b) to get the estimated standard error).
5. calculate $z^*(b) = \frac{\hat{\theta}(b) - \hat{\theta}}{\hat{se}(b)}$ for each b.
6. get the α percentile of $z^*(b)$ is estimated by the value $\hat{t}^{(\alpha)}$ such that $\#\{z^*(b) < \hat{t}^{(\alpha)}\}/B = \alpha$
7. the bootstrap-t confidence interval is $(\hat{\theta} - \hat{t}^{(1-\alpha)}\hat{se}, (\hat{\theta} - \hat{t}^{(\alpha)}\hat{se})$,where $\hat{se}$ is the standard deviation of $\hat{\theta}^*(b)$'s.

```
## [1] -1.3782126  0.9002225
```

2.

(a) Find the distribution.

$$\begin{aligned} P(X_{\max} < x) &= \prod_{i=1}^{n} P(X_i < x) \\ &= (x/\theta)^n \\ f_{X_{\max}}(x) &= \frac{nx^{n-1}}{\theta^n} \end{aligned}$$

so that we get the pdf of $X_{\max}$

(b) Derive the analytic expression for the variance.

$$\begin{aligned} E(\hat{\theta}) &= \int_0^\theta x \frac{nx^{n-1}}{\theta^n} dx \\ &= \frac{n}{\theta^n} \times \frac{1}{n+1} x^{n+1}|_0^\theta \\ &= \frac{n}{n+1}\theta \\ E(\hat{\theta}^2) &= \int_0^\theta x^2 \frac{nx^{n-1}}{\theta^n} dx \\ &= \frac{n}{\theta^n} \times \frac{1}{n+2} x^{n+2}|_0^\theta \\ &= \frac{n}{n+2}\theta^2 \\ Var(\hat{\theta}) &= E(\hat{\theta}^2) - E(\hat{\theta})^2 \\ &= \frac{n}{n+2}\theta^2 - (\frac{n}{n+1}\theta)^2 \\ &= \frac{n\theta^2}{(n+1)^2(n+2)} \end{aligned}$$

(c) Generate a data set of size $n = 50$ and $\theta = 3$. Then generate $B = 5000$ bootstrap samples using parametric bootstrap. Use the bootstrap samples to approximate $Var_{F_\theta}(\hat{\theta})$. Compare your answer to (b).

Take the maximum of the sample as the $\hat{\theta}$, simulate bootstrap samples from unif(0,$\hat{\theta}$)

the parametric bootstrap result is

```
## [1] 0.003174573
```

the answer to (b) is

```
## [1] 0.003327123
```

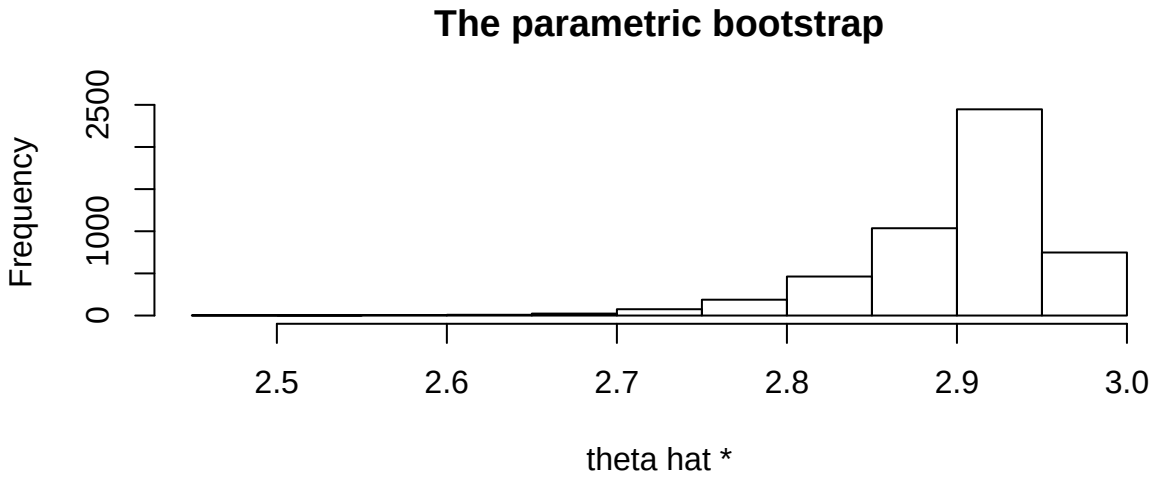
They are close.

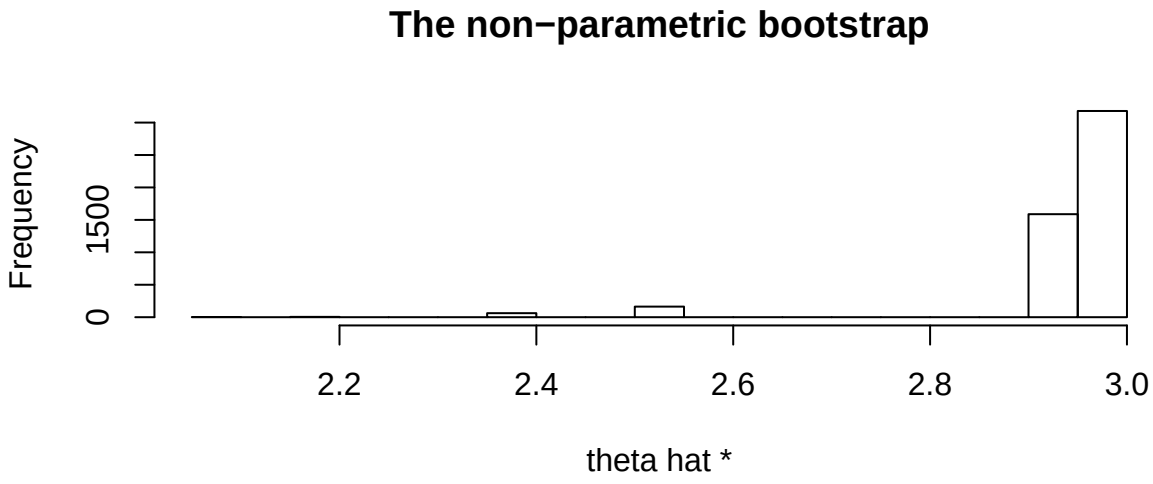
(d)

the nonparametric bootstrap samples results is

```
## [1] 0.01081648
```

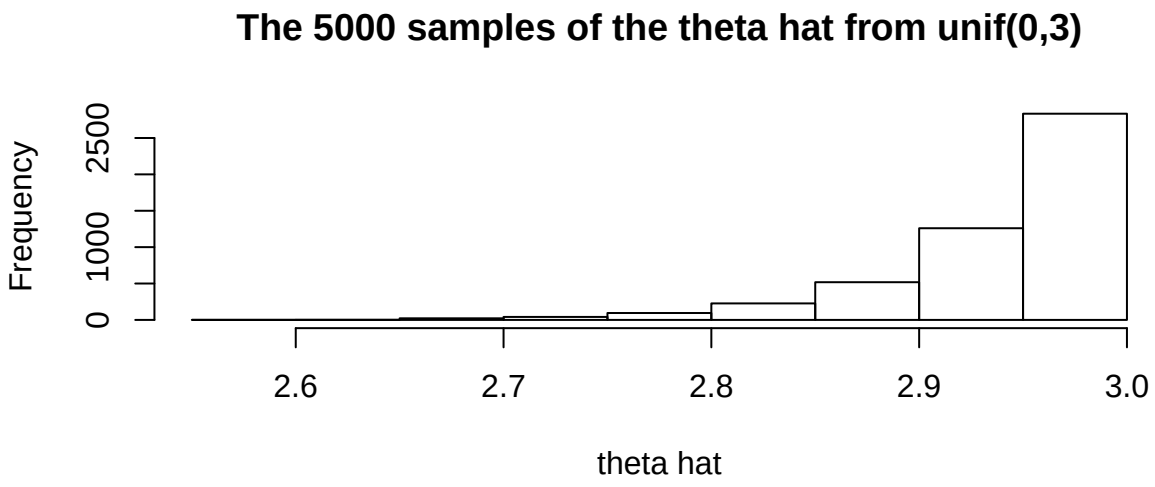
(e)





(f)

the true distribution of $\hat{\theta}$



The non-parametric plot did not look like the true one.

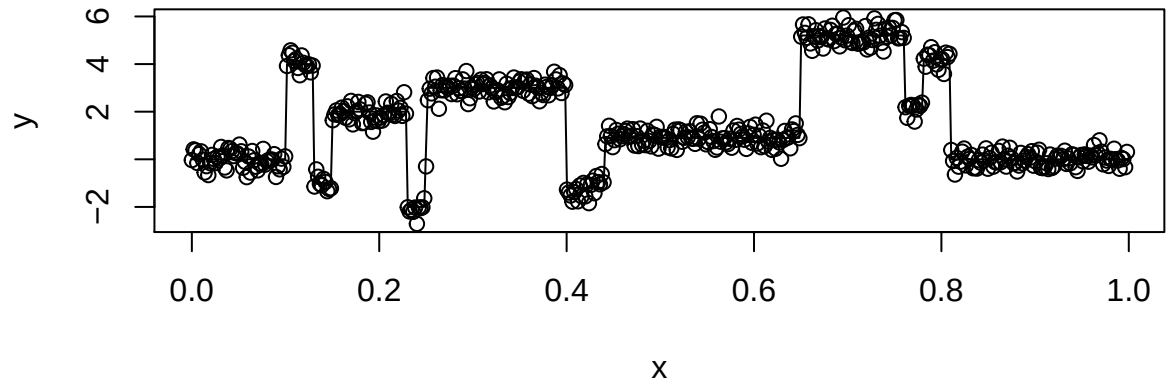
The reason is that for each non-parametric bootstrap, the chance of not choosing the largest value of the sample as the $\hat{\theta}$ is (49/50)^50 = 0.3641697 which is very low.

3.

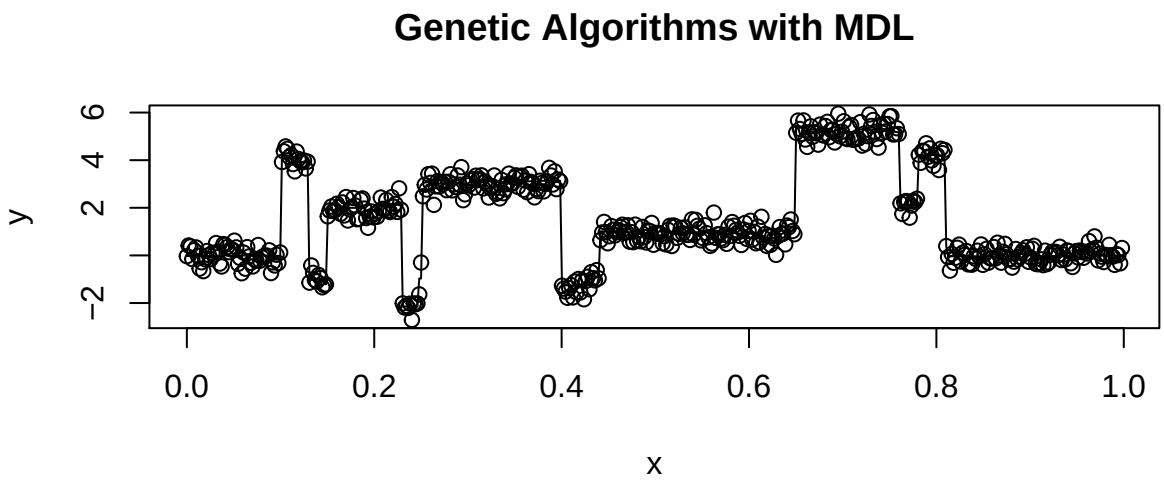
For the genetic algorithm, I just copy the code from my homework assignment 2 here.

Here is the result for the test function in Assignment 2.

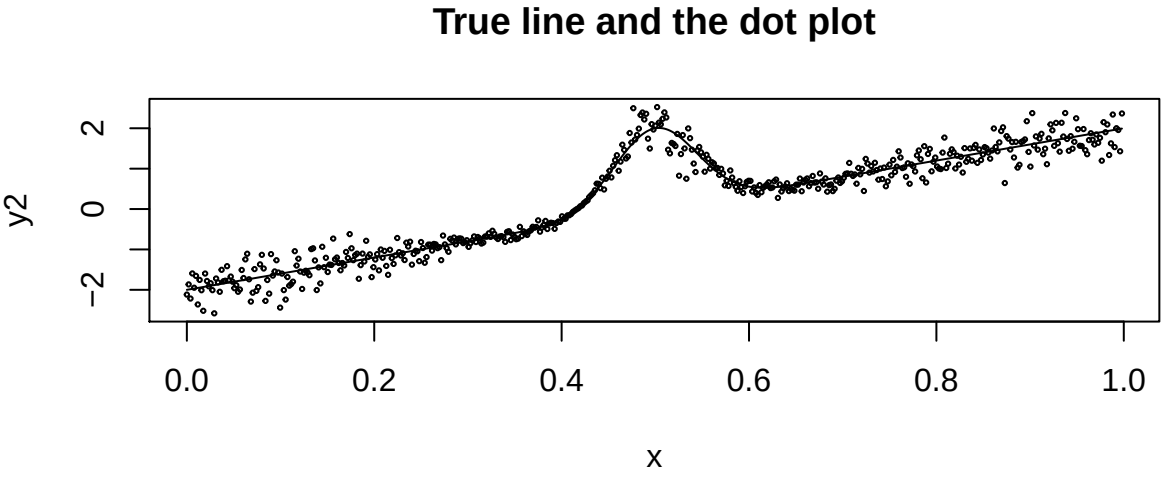
The true plot.

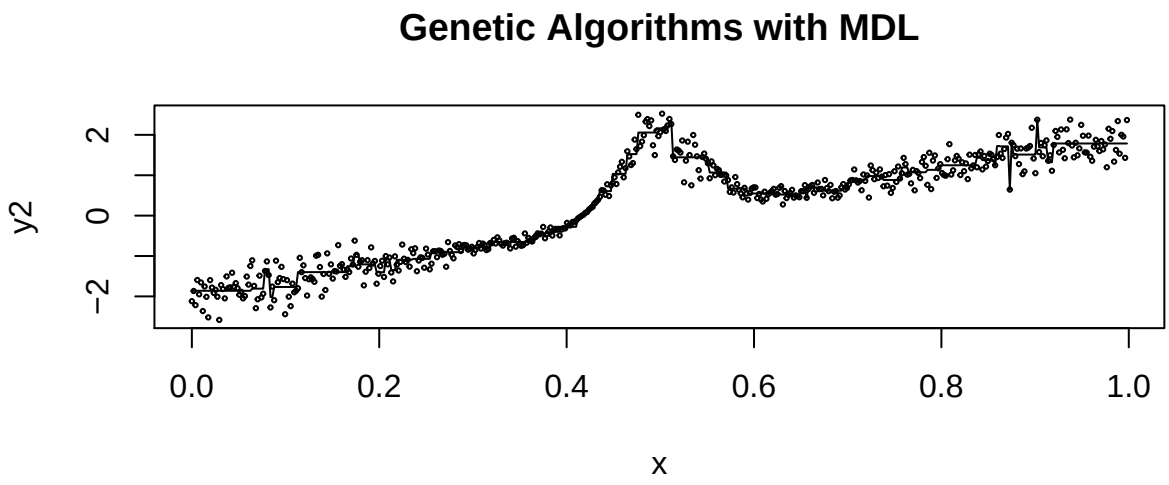


I just use the MDL method to generate the y and the plot.



for the sencond model



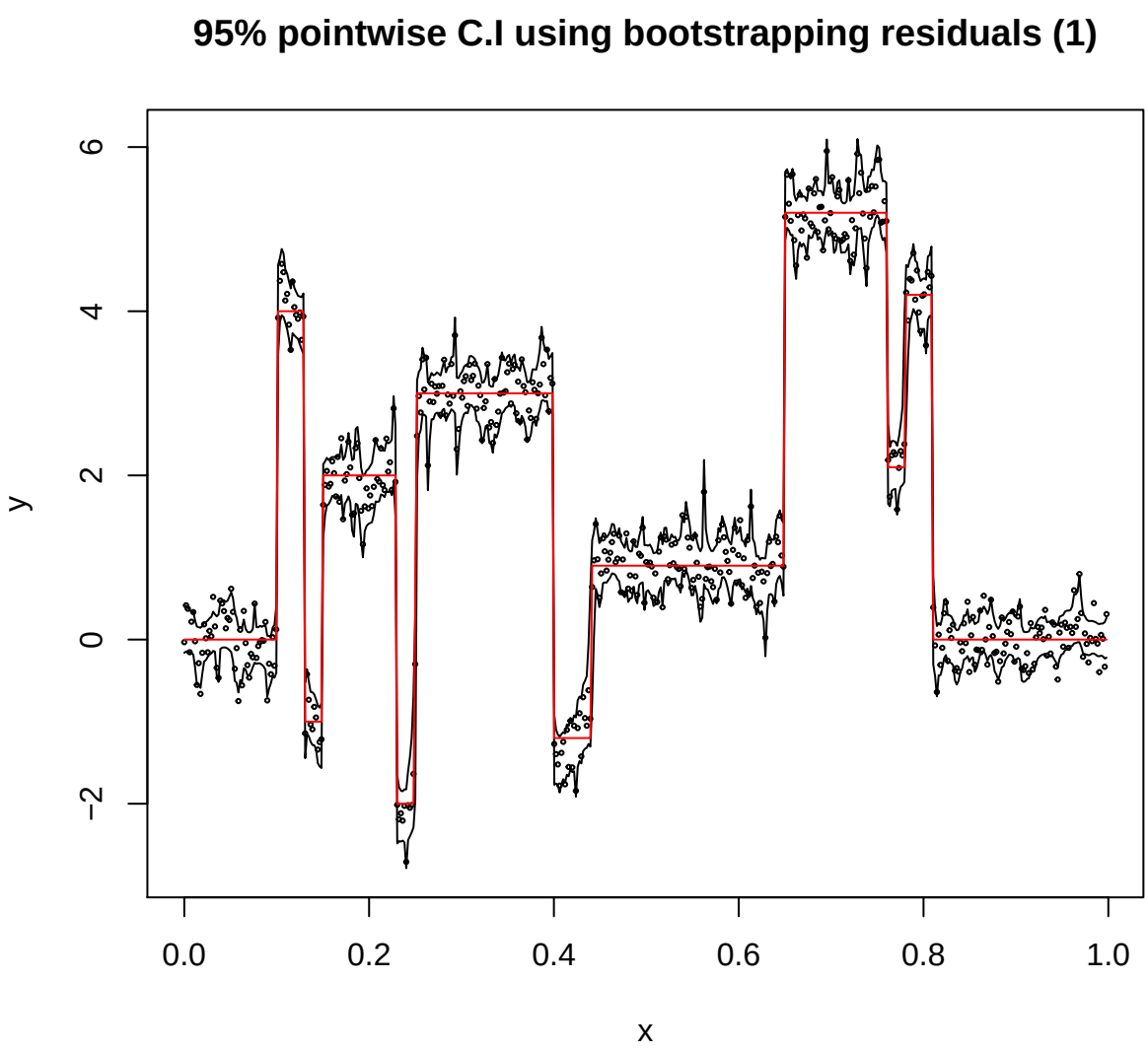


(b) The genetic algorithm i used did not control the maximum breakpoint number which may case a overfitting issue, although I am using MDL to penalize the number of breakpoit.

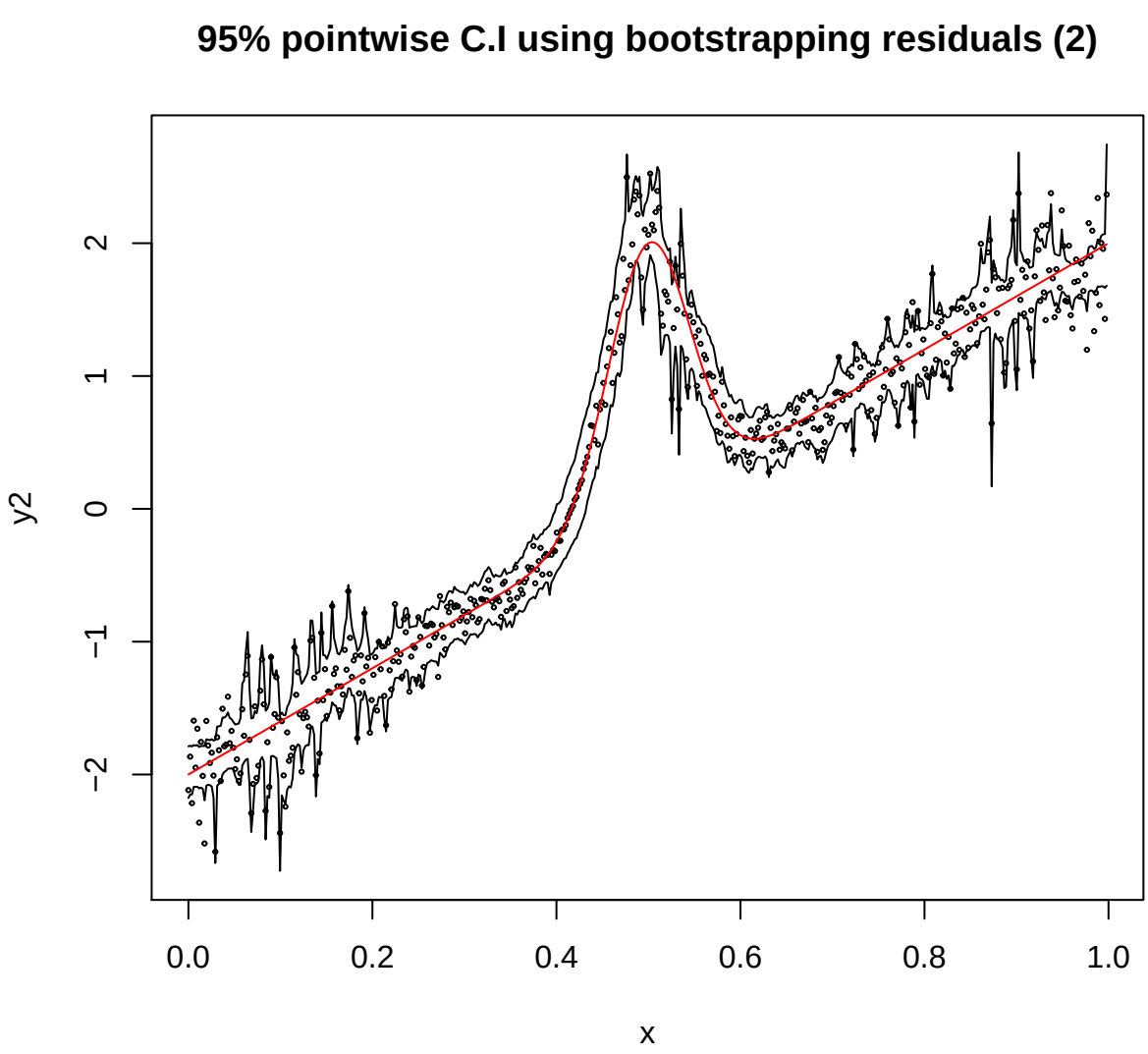
part 1

Get the residuals of the model, resample the residuals and then get the bootstarps samples to get 1000 times bootstraps predicted value. Consturct pointwise C.I. and plot it like below.

Although I already using the snow package to run it parallely, it cost me over a day to run the 1000 times boorstraps to get the four plots as below:



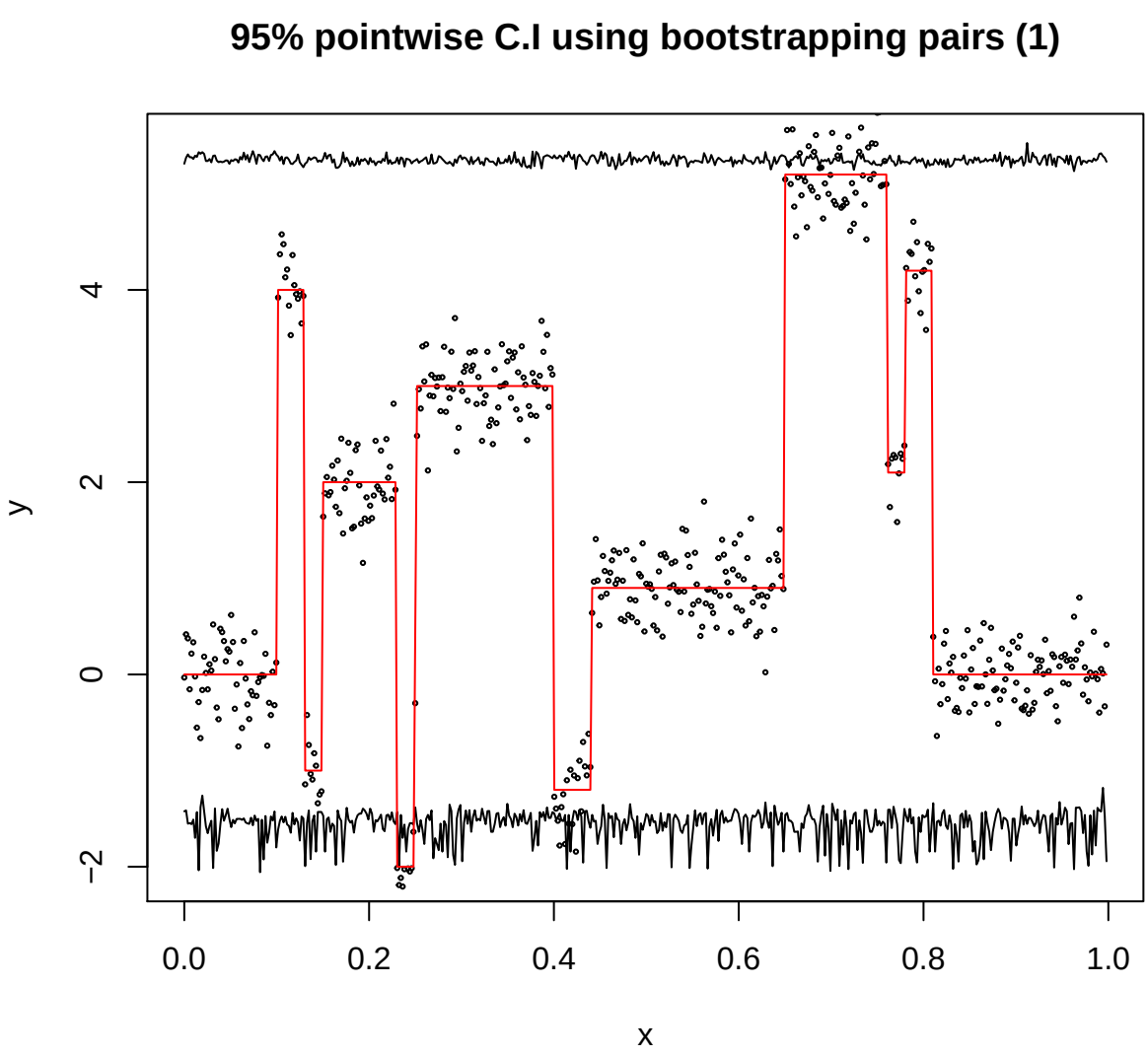
Comment on the shape of the confidence bands near jump points: The confidence bands are very wide in these jump points. It is because these break point may consider in the nearest two line.

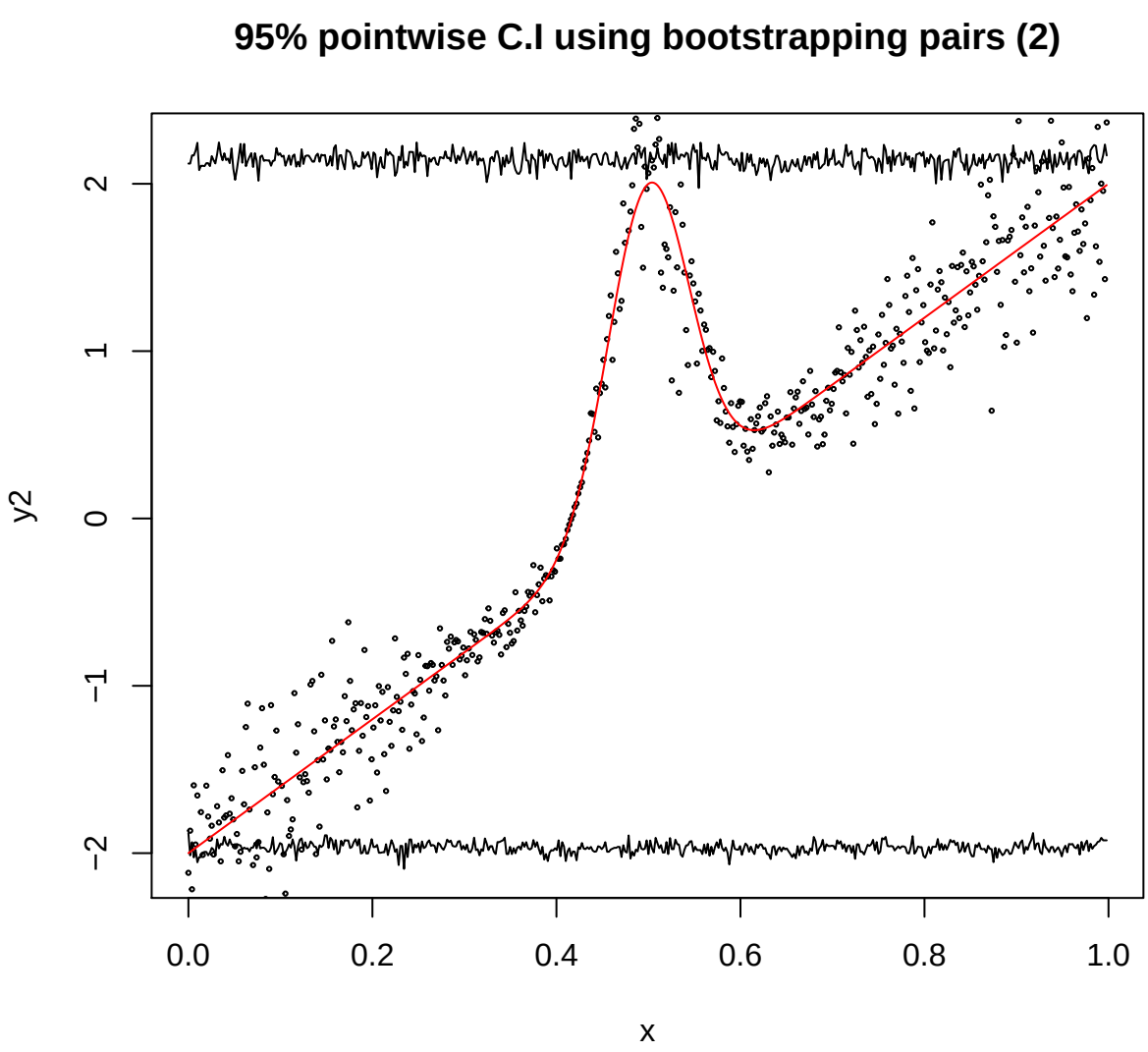


part 2

bootstrapping pairs

resample pairs from the original data and then get the predicted values and consturcted the pointwise C.I. plot it like below:


95% pointwise C.I using bootstrapping pairs (1)
y
x
−2
0
2
4
0.0
0.2
0.4
0.6
0.8
1.0



It seems the C.I did not work well in pairwise bootstraps. The reason I thought is that resampling pairs casuing a single point repeatly several times in my dataset, my algorithm take all the points equally i did not get the unique one.

To test this, I add a unique() before the resampling and to try it one more time.

I am running out of time, I did not have the time to rerun it in the scale of 1000 times to get a plot. But i did run it in 392 times(using 7 cores), the situation did not improve, the plot is very similar to the one above.

The reason may be the genetic algorithm i uesed in this question is so sensitive to the sample data. As i mentioned before, i am running out of time to debug them or to make sure it is the truth.

The following can construct confidence set for a break point:

1. Generate bootstrap sample from the data.
2. Estimate the location of the break points for each the bootstrap sample using genetic method(chromosome c_i is the best chromosome in the ith bootstrap.
3. get $C = \sum c_i$ as the count for all the points considering to be the break point.
4. get $\alpha/2$ lowest value $C_(\alpha/2)$and the $1-\alpha/2$ highest value $C_{(1-\alpha/2)}$of C.

5. the upper set of the confidence set is that $\{X_i|i : C_i > C_(1-\alpha/2)\}$,the lower set of the confidence set is that $\{X_i|i : C_i > C_(\alpha/2)\}$ where C_i is the ith element in C

4.

(a)

$$\begin{aligned} f(x) &= \lambda e^{-\lambda x} \\ l(\lambda) &= logL(\lambda|X) = n\log\lambda - \lambda\sum x_i \\ l'(\lambda) &= \frac{n}{\lambda} - \sum x_i \\ \hat{\lambda}_{MLE} &= \frac{n}{\sum x_i} \end{aligned}$$

(b)

$$\sqrt{n}(\hat{\lambda} - \lambda) = \sqrt{n}(\frac{1}{\bar{X}} - \frac{1}{\mu})$$

using the delta method,

$$\sqrt{n}(f(x) - f(\mu)) \to N(0, f'(x)^2\sigma^2)$$

in distribution,

$(\frac{1}{x})' = -\frac{1}{x^2}$ and $Var(x) = \lambda^{-2}$, we can easily get

$$\sqrt{n}(\hat{\lambda} - \lambda) = \sqrt{n}(\frac{1}{\bar{X}} - \frac{1}{\mu}) \to N(0, \lambda^2)$$

in distribution.

the same situation using delta method, $(\log\frac{1}{x})' = -\frac{1}{x}$

$$\sqrt{n}(\log\hat{\lambda} - \log\lambda) = \sqrt{n}(\log\frac{1}{\bar{X}} - \log\frac{1}{\mu}) \to N(0, \lambda^2 \times \frac{1}{\lambda^2}) = N(0,1)$$

in distribution

(c)

using the results from (b),when n is large

$$\begin{aligned} \sqrt{n}(\log\hat{\lambda} - \log\lambda) &\to N(0,1) \\ P(z(\alpha) < \sqrt{n}(\log\hat{\lambda} - \log\lambda) < z(1-\alpha)) &\approx 2\alpha \\ P(-\frac{z(1-\alpha)}{\sqrt{n}} < \log\lambda - \log\hat{\lambda} < \frac{-z(\alpha)}{\sqrt{n}}) &\approx 2\alpha \\ P(\frac{-z(1-\alpha)}{\sqrt{n}} + \log\hat{\lambda} < \log\lambda < \frac{-z(\alpha)}{\sqrt{n}} + \log\hat{\lambda}) &\approx 2\alpha \\ P(e^{\frac{-z(1-\alpha)}{\sqrt{n}}}\hat{\lambda} < \lambda < e^{\frac{-z(\alpha)}{\sqrt{n}}}\hat{\lambda}) &\approx 2\alpha \\ P(e^{\frac{-z(1-\alpha/2)}{\sqrt{n}}}\hat{\lambda} < \lambda < e^{\frac{-z(\alpha/2)}{\sqrt{n}}}\hat{\lambda}) &\approx \alpha \end{aligned}$$

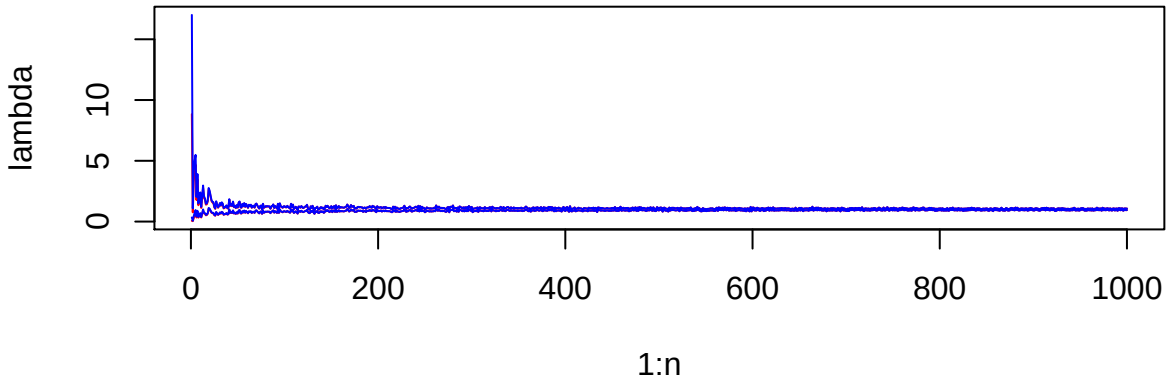
then we get the result.

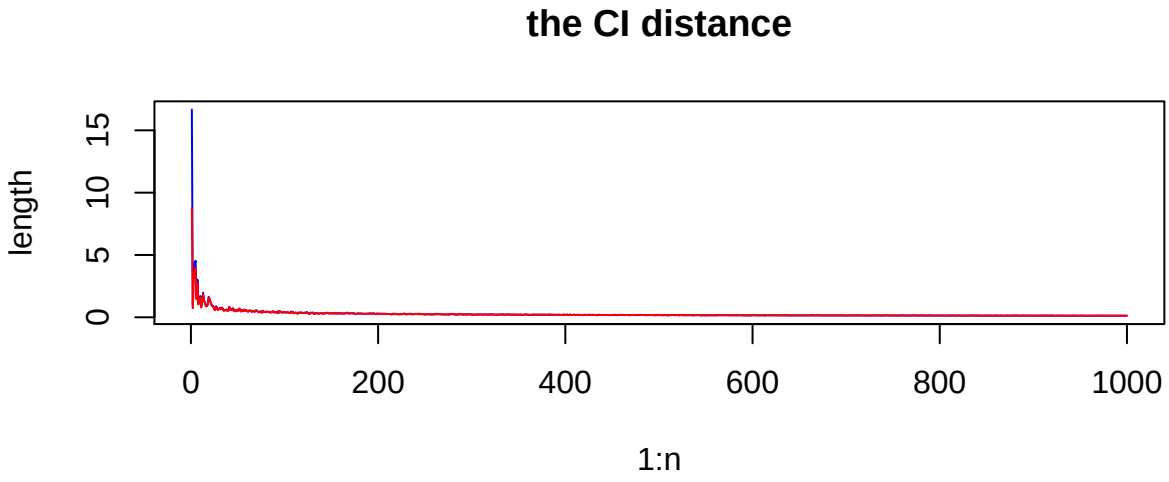
(d)

we have $\lambda(\sum x_i) \sim G(n,1)$, then

$$\begin{aligned}\alpha &= P(G^{-1}(\alpha/2) < \lambda \sum x_i < G^{-1}(1-\alpha/2)) \\ &= P(G^{-1}(\alpha/2) < n\frac{\lambda}{\hat{\lambda}_n} < G^{-1}(1-\alpha/2)) \\ &= P(\hat{\lambda}_n G^{-1}(\alpha/2)/n < \lambda < \hat{\lambda}_n G^{-1}(1-\alpha/2)/n)\end{aligned}$$

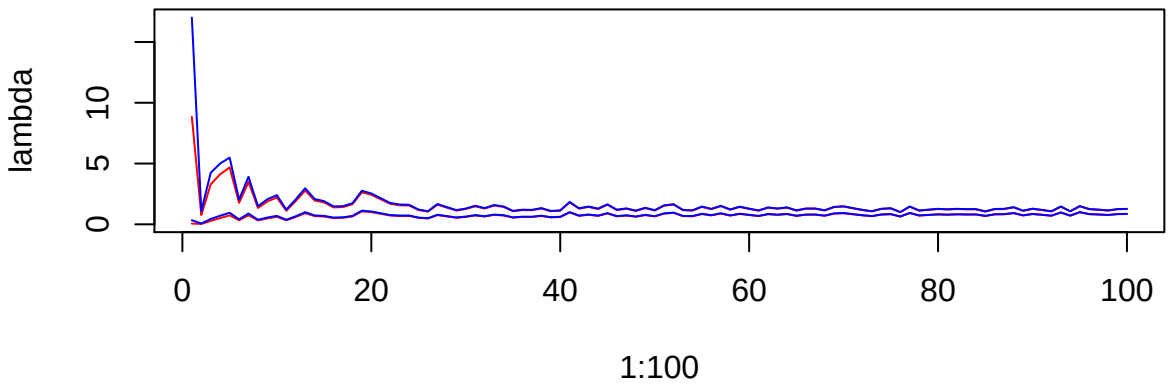
(e)

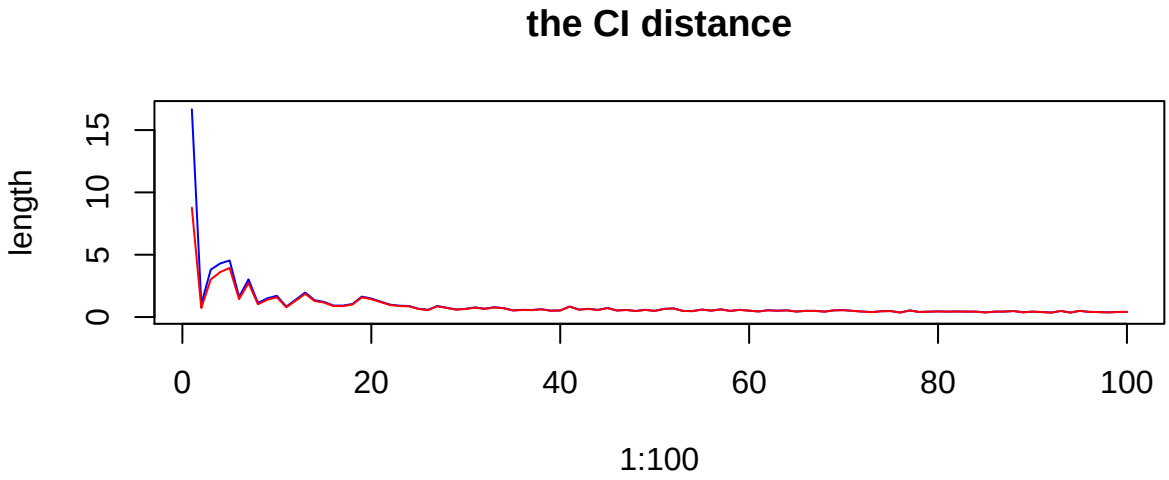




the blue line above is the asymptotic line, the red line is the line using the true distribution.

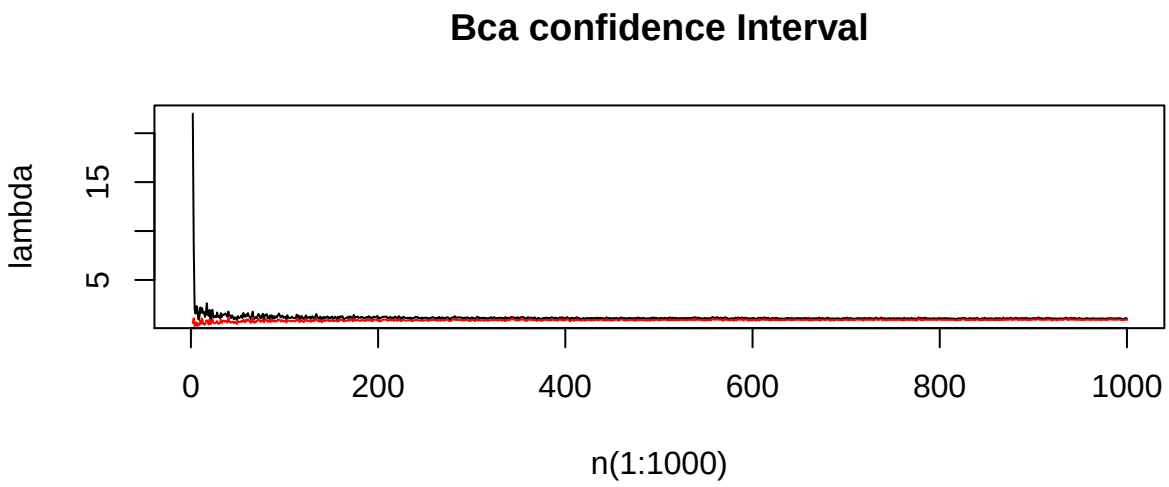
I can not tell much difference from them. they match with each other after 100 iterations so that i decide to plot the 1 to 100 iteration plot.

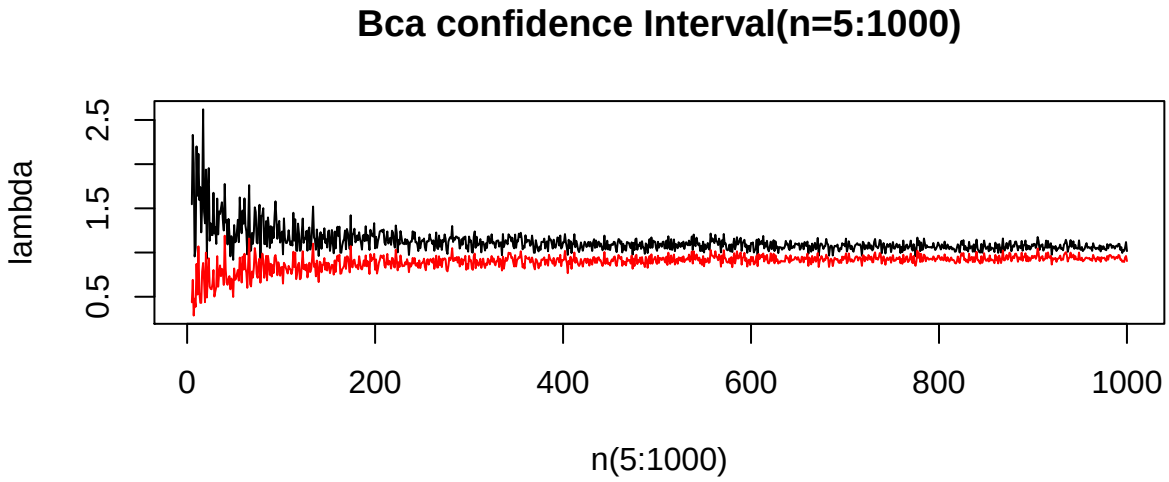


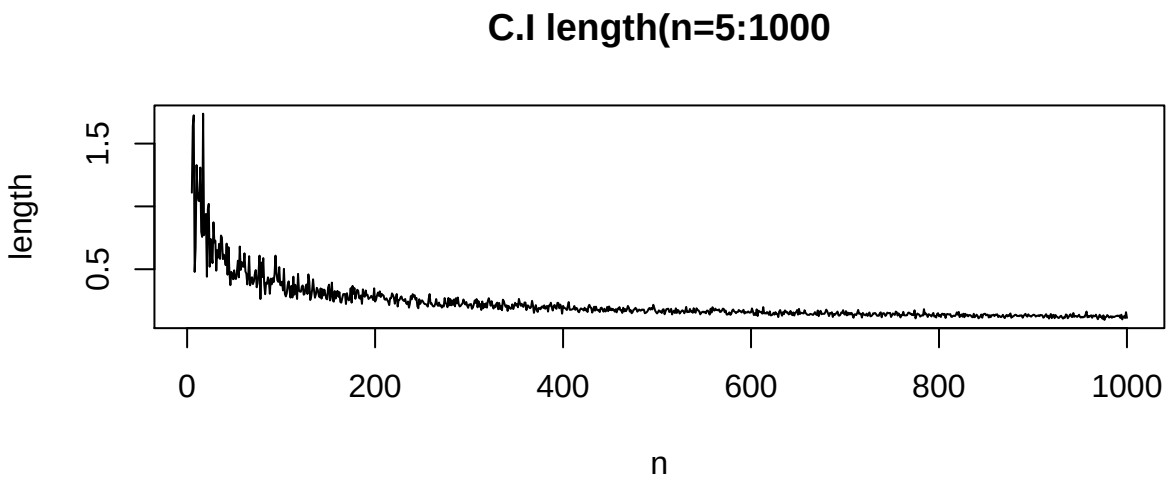


as we can see, the blue line from 1 to 10 is above the red line which means the asymptotic one is wider at the beginning and then we are about the same.

the BCa confidence Interval(95%)


Bca confidence Interval
lambda
5
15
0
200
400
600
800
1000
n(1:1000)


Bca confidence Interval(n=5:1000)
lambda
0.5
1.5
2.5
0
200
400
600
800
1000
n(5:1000)



Here are the plots of BCa C.I. the red(black) line is the L(U) in C.I [L,U].

As we can see when n is below 5, the bootstrap is meaningless since the 97.5% and 2.5% percentile point of the mean of bootstrap sample is almost surely the min and max.

It look similar to the theoretical value (the bootstrap iteration is only 200.)