Computational Statistics Lab3 Group 29

Jin Yan (jinya425), Yaning Wang (yanwa579)

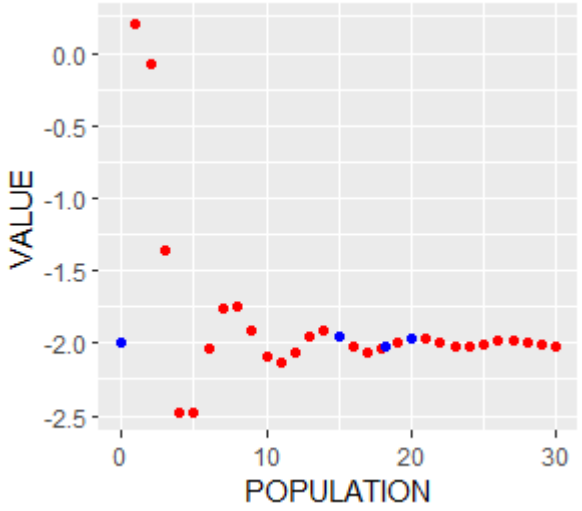
Question 1: Genetic algorithm

TASK ONE;TASK TWO;TASK THREE; TASK FOUR The code relevant to these tasks is given in APPENDIX.

TASK FIVE

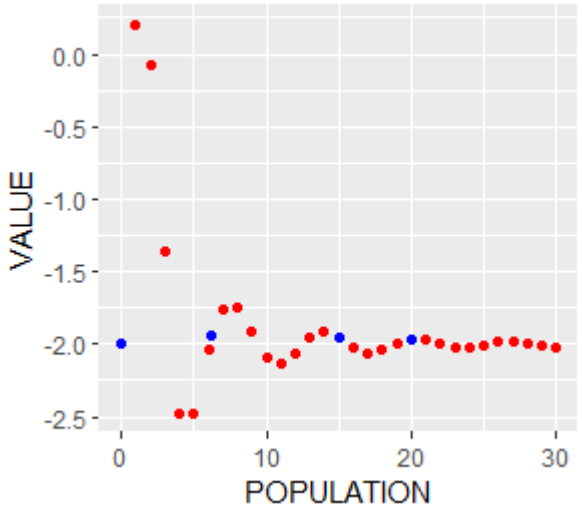
maxiter =10, mutprob = 0.1

The final population is 0 15 15 15 20 18.28125 20



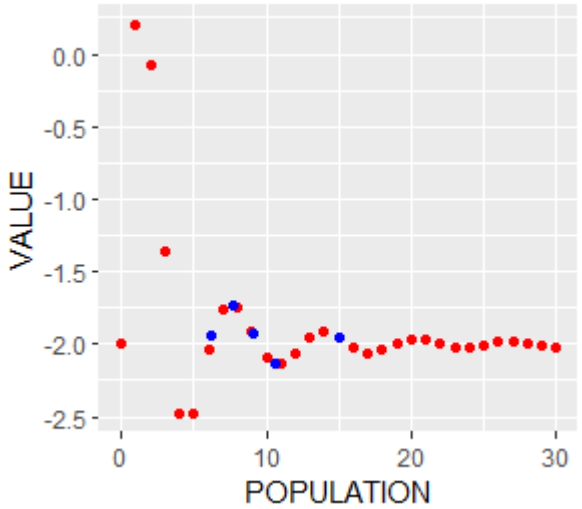
maxiter =10, mutprob = 0.5

The final population is 0 15 15 15 20 20 6.25



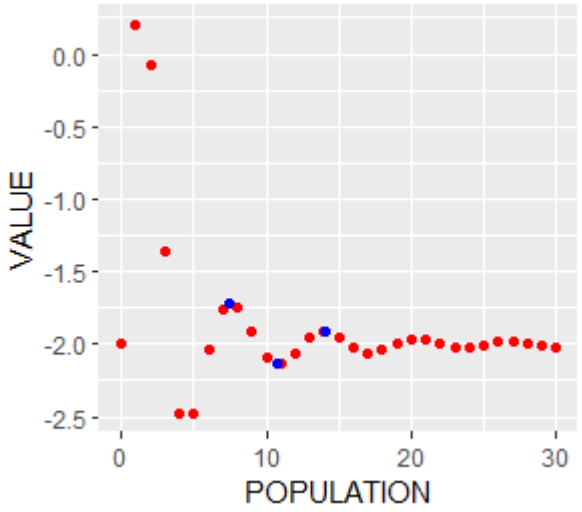
maxiter =10, mutprob = 0.9

The final population is 9.0625 15 6.25 15 10.625 7.65625 6.25



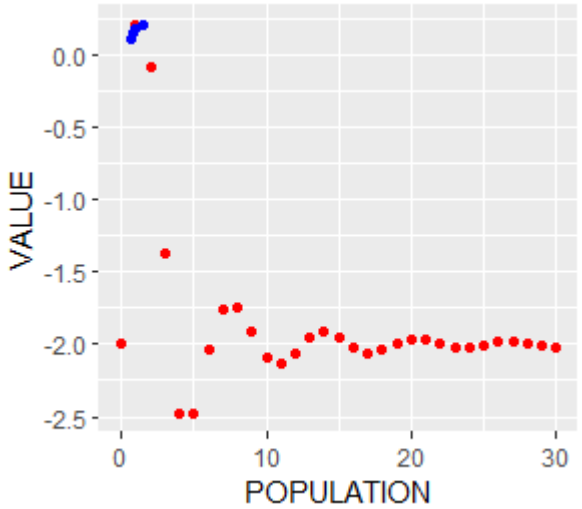
maxiter =100, mutprob = 0.1

The final population is 10.78125 7.5 14.0625 14.0625 14.0625 14.0625 14.0625



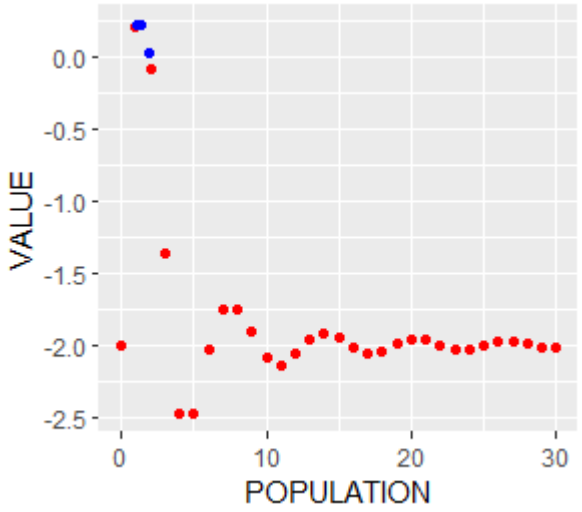
maxiter =100, mutprob = 0.5

The final population is 1.456551 0.8018984 1.456551 0.8954878 0.72028 0.8018984 0.8896793



maxiter =100, mutprob = 0.9

The final population is 1.155513 1.293591 1.861039 1.240155 1.335211 1.155513 1.245362

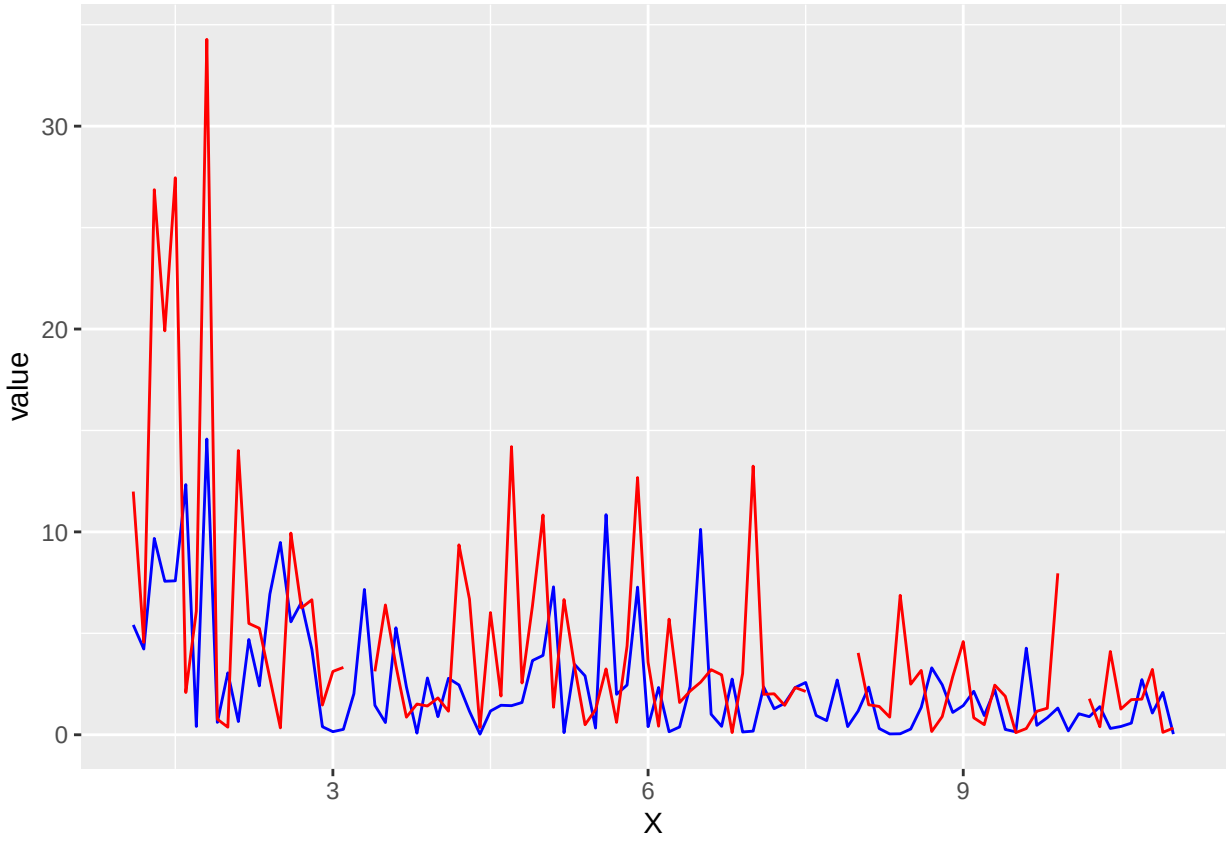


CONLCUTION: When the number of iteration becomes larger, it is more likely for the algorithm to find the general optimum. It should also be noticed that when the number of iteration is large, with the mutation probability increasing, population points will converge more obviously.

Question 2: EM algorithm

Task 1

Plot Z and Y versus X



Q: Does it seem that two processes are related to each other? What can you say about the variation of the response values with respect to X?

From the figure we can find that both plots have the same trend. When X is at 1.8, the values of Y and Z are both maximal. So we can say the two processes are related to each other.

Task 2

When we implement EM algorithm, based on E-step we need to derive $Q(\theta, \theta^k)$

From the question,we know:

$Y_i \sim exp(\frac{X_i}{\lambda})$

$Z_i \sim exp(\frac{X_i}{2\lambda})$

Now, we get the density function for Y and Z:

$f(Y_i) = \frac{X_i}{\lambda} \cdot \exp(-\frac{X_i}{\lambda} Y_i)$

$f(Z_i) = \frac{X_i}{2\lambda} \cdot \exp(-\frac{X_i}{2\lambda} Z_i)$

The likelihood function is:

$L(\lambda|Y, Z)$

$=\prod_{i=1}^{n} P(Y_i, Z_i|\lambda)$

$=\prod_{i=1}^{n} P(Y_i|\lambda) * \prod_{i=1}^{n} P(Z_i|\lambda)$

$=\prod_{i=1}^{n} \frac{X_i}{\lambda} \cdot \exp(-\frac{X_i}{\lambda} Y_i) * \prod_{i=1}^{n} \frac{X_i}{2\lambda} \cdot \exp(-\frac{X_i}{2\lambda} Z_i)$

The log-likelihood function is:

$lnL(\lambda|Y,Z) = \sum_{i=1}^{n} \ln(X_i) - n\ln(\lambda) - \sum_{i=1}^{n} \frac{X_i}{\lambda} Y_i + \sum_{i=1}^{n} \ln(X_i) - n\ln(2\lambda) - \sum_{i=1}^{n} \frac{X_i}{2\lambda} Z_i$

And we know the formula:$Q(\theta, \theta^k) = E[loglik(\theta|Y,Z)|\theta^k, Y]$

For this question, we can write this with λ, λ^k and separate Z into two parts, one containing the missing values and the other with no missing values,and the number of missing values is m. And the expected value of the exponential distribution $\lambda \exp(-\lambda x)$ is $\frac{1}{\lambda}$

$Q(\lambda, \lambda^k) = E[\sum_{i=1}^{n} \ln(X_i) - n\ln(\lambda) - \sum_{i=1}^{n} \frac{X_i}{\lambda} Y_i + \sum_{i=1}^{n} \ln(X_i) - n\ln(2\lambda) - \sum_{i=1}^{n} \frac{X_i}{2\lambda} Z_i]$

$= \sum_{i=1}^{n} \ln(X_i) - n\ln(\lambda) - \sum_{i=1}^{n} \frac{X_i}{\lambda} Y_i + \sum_{i=1}^{n} \ln(X_i) - n\ln(2\lambda) - \sum_{i=1}^{n-k} \frac{X_i}{2\lambda} Z_i - \sum_{i=n-k+1}^{n} \frac{\lambda^k}{\lambda}$

Next we need to follow M-step:$\lambda^{k+1} = argmax_\lambda Q(\lambda, \lambda^k)$

We take the derivative with respect to λ and the formula should equal to 0

$-\frac{n}{\lambda} + \sum_{i=1}^{n} \frac{X_i}{\lambda^2} Y_i - \frac{n}{\lambda} + \sum_{i=1}^{n-m} \frac{X_i}{2\lambda^2} Z_i + \frac{m\lambda^k}{\lambda^2} = 0$

We get the λ

$\lambda = \frac{1}{2n}(\sum_{i=1}^{n} X_i Y_i + \sum_{i=1}^{n-m} \frac{1}{2} X_i Z_i + m\lambda^k)$

n-m is the number of value except the missing value.m corresponds to the number of missing value.

The last step use this formula and the code to estimate λ

Task 3

Implement this algorithm in R and find the optimal λ

```
## The iterative data are
```

```
##      sum_lamprev sum_lamcurr
## [1,]   100.00000    14.26782
## [2,]    14.26782    10.83853
## [3,]    10.83853    10.70136
## [4,]    10.70136    10.69587
## [5,]    10.69587    10.69566
```

```
## The optimal lambda is 10.69566
```

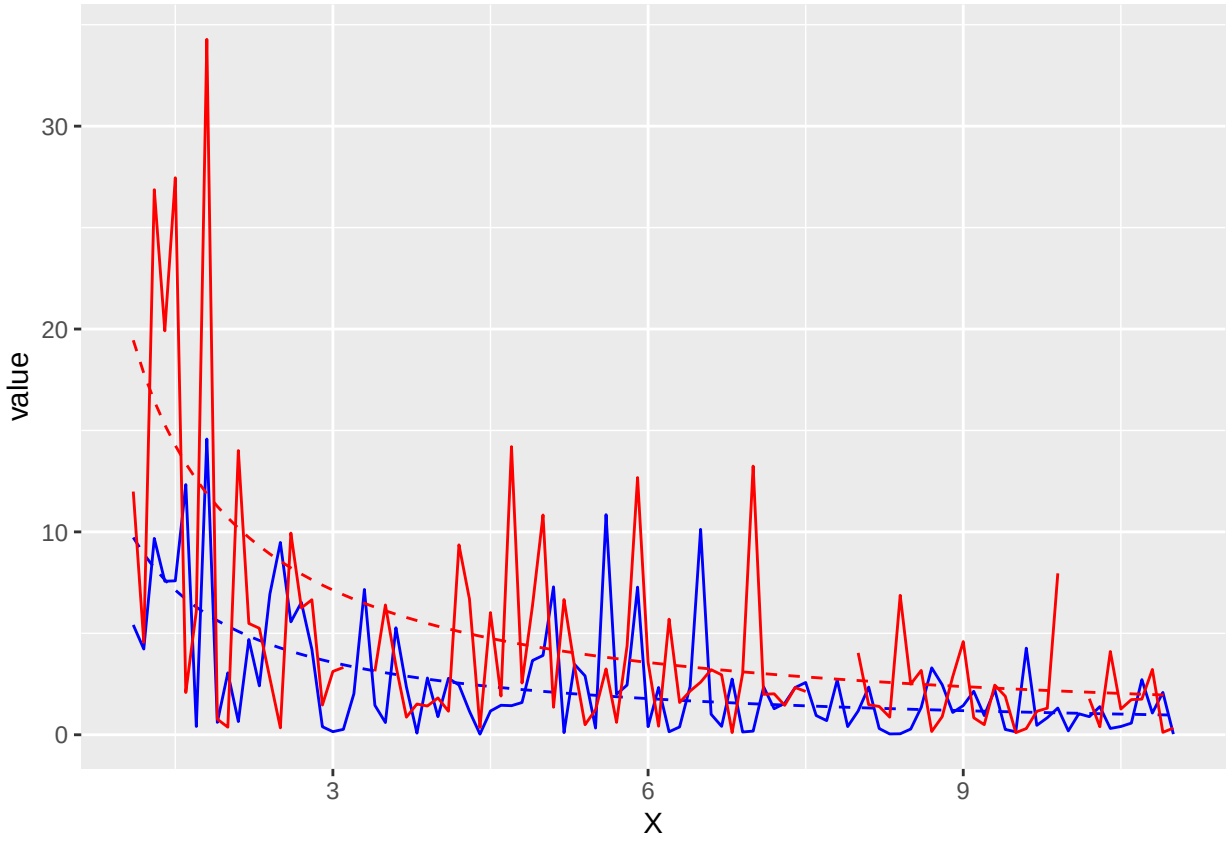
```
## The number of interation is 5
```

Task 4

Plot E[Y] and E[Z] versus X in the same plot as Y and Z versus X

Since y is an exponential distribution, from the properties of the exponential distribution we know the expected value for Y distribution is λ/X. The expected value for Z distribution is $2 * \lambda/X$.

We use the optimal lambda to calculate the expected value.



From the figure (dashed line) we can find that it can capture the trend of data.

APPENDIX

Question 1: Genetic algorithm

```
library(ggplot2)

f_x <- function(x){
  value <- x^2 / exp(x) - 2 * exp(-(9 * sin(x)) / (x^2 + x + 1))
  return(value)
}

crossover <- function(x,y){
  value <- (x + y) / 2
  return(value)
}

mutate <- function(x){
  value <- x^2 %% 30
  return(value)
}

target_function <- function(maxiter,mutprob){
```

```
  first_population <- c(0:30)
  first_value <- f_x(first_population)
  dataf1 <- data.frame(first_population,first_value)

  # the main part of the function
  value_max <- c()
  X <- 5 * c(0:6)
  Value <- f_x(X)

  for(i in 1:maxiter){
    parents <- sample(X,2)
    kid <- crossover(parents[1],parents[2])
    index_victim <- order(Value)[1]
    if(runif(1) <= mutprob){
      kid <- mutate(kid)

    }
    X[index_victim] <- kid
    Value[index_victim] <- f_x(X[index_victim])

    value_max <- c(value_max,max(Value))
  }

  dataf2 <- data.frame(X,Value)
  plot1 <- ggplot(data = dataf1,aes(first_population,first_value)) +
    geom_point(color = "red") +
    geom_point(data = dataf2, aes(X,Value), color = "blue") +
    xlab("POPULATION") + ylab("VALUE")

  print(plot1)
  cat("The final population is\n",X)

}

target_function(10,0.1)

target_function(10,0.5)

target_function(10,0.9)

target_function(100,0.1)

target_function(100,0.5)

target_function(100,0.9)
```

Question 2: EM algorithm

```
#task1

physical<-read.csv("C:/Users/wyn19/OneDrive/Desktop/Study in LIU/Semester 1/Computational Statistics/la
#physical<-read.csv("physical1.csv")
```

```r
ggplot(data=physical)+
  geom_line(aes(x=X,y=Y),colour="blue")+
  geom_line(aes(x=X,y=Z),colour="red")+
  ylab("value")

#task2
##Please see above about the formula

#task3
Zobs <- physical$Z[!is.na(physical$Z)]
Zmiss <- physical$Z[is.na(physical$Z)]
X_zobs<-physical$X[which(!is.na(physical$Z))]
m<-length(Zmiss)
n<-length(physical$X)
lambda_prev<-0
lambda_curr<-100
k<-0
min<-0.001
sum_lamprev<-c()
sum_lamcurr<-c()
while(abs(lambda_curr-lambda_prev)>min){
  lambda_prev<-lambda_curr
  lambda_curr<-1/(2*n)*(sum(physical$X*physical$Y)+
                        0.5*sum(X_zobs*Zobs)+
                        m*lambda_prev)
# k count the number of iteration
  sum_lamprev<-c(sum_lamprev,lambda_prev)
  sum_lamcurr<-c(sum_lamcurr,lambda_curr)
  k<-k+1
}
lambda<-cbind(sum_lamprev,sum_lamcurr)
cat("The iterative data are")
print(lambda)
cat("The optimal lambda is",lambda_curr)
cat("The number of interation is",k)

#task4
Y_expect<-lambda_curr/physical$X
Z_expect<-2*lambda_curr/physical$X
df<-data.frame(Y_expect,Z_expect)
ggplot(data=physical)+
  geom_line(aes(x=X,y=Y),colour="blue")+
  geom_line(aes(x=X,y=Y_expect),colour="blue",linetype="dashed")+
  geom_line(aes(x=X,y=Z),colour="red")+
  geom_line(aes(x=X,y=Z_expect),colour="red",linetype="dashed")+
  ylab("value")
```