Lab6_comp_stat

Gowtham(gowku593) & Biswas (bisku859)

06/03/2020

Question 1: Genetic Algorithm

In this assignment, you will try to perform one-dimensional maximization with the help of a genetic algorithm.

1.1. Defne the function :

$f(x) = \frac{x^2}{e^x} - 2exp(-(9sin(x)/(x^2 + x + 1)))$

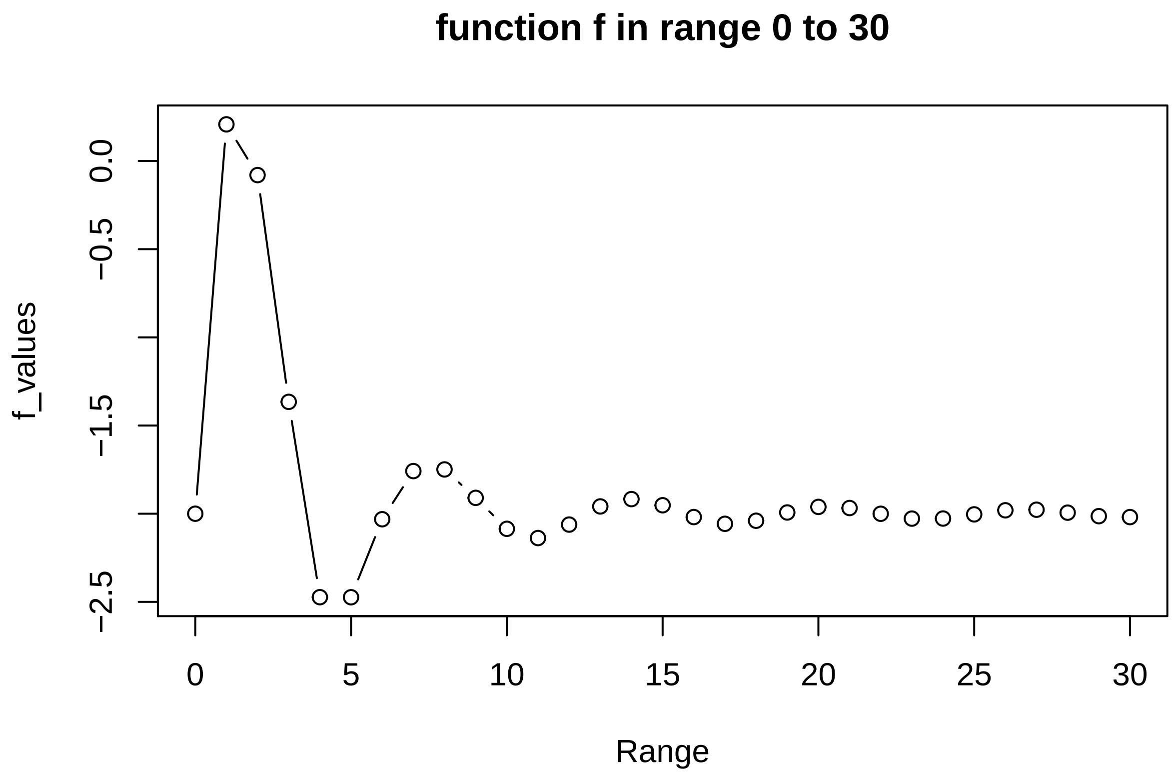
1.2. Define the function crossover(): for two scalars x and y it returns their kid as

$(x + y)/2$.

1.3. Define the function mutate that for a scalar x returns the result of the integer division x square mod 30.

1.4. Write a function that depends on the parameters maxiter and mutprob and:

(a) Plots function f in the range from 0 to 30. Do you see any maximum value?




```
## the maximum value is 0.207668792245968
```

(b) Defines an initial population for the genetic algorithm as X = (0, 5,10, 15,… 30).

```
## [1]  0  5 10 15 20 25 30
```

(c) Computes vector Values that contains the function values for each population point.

1.4.(d) Performs maxiter iterations where at each iteration

i. Two indexes are randomly sampled from the current population, they are further used as parents (use sample()).

ii. One index with the smallest objective function is selected from the current population,the point is referred to as victim (use order()).

iii. Parents are used to produce a new kid by crossover. Mutate this kid with probability mutprob (use crossover(), mutate()).

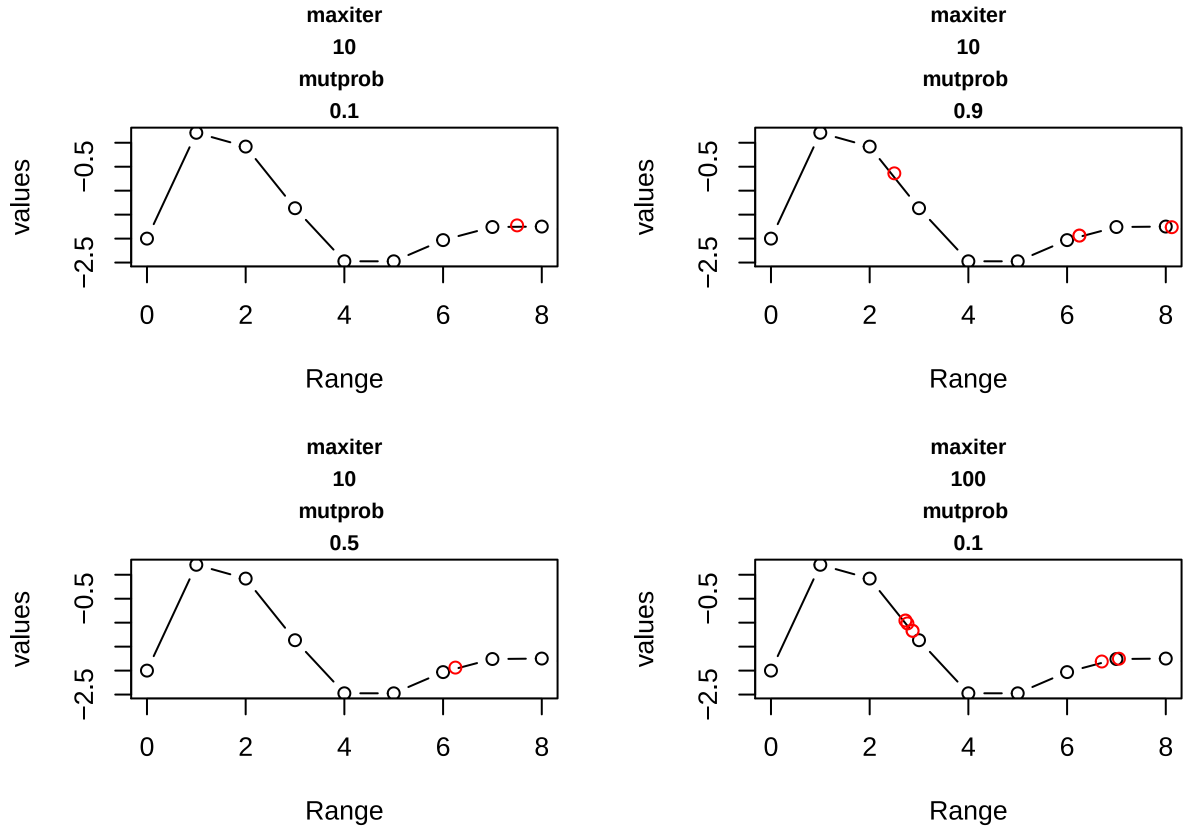
iv. The victim is replaced by the kid in the population and the vector Values is updated.

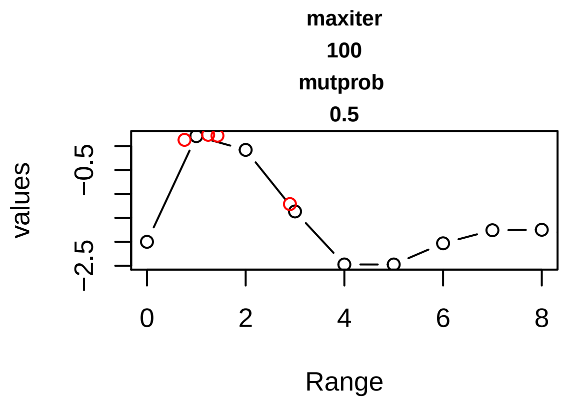
v. The current maximal value of the objective function is saved

1.4 (e) Add the final observations to the current plot in another colour.

1.5. Run your code with different combinations of maxiter= 10,100 and mutprob= 0.1,0.5, 0.9.Observe the initial population and final population. Conclusions?

Figure 1-3: Function output with maxiter=10 and from top to bottom, mutprob=0.1,0.5, 0.9 respectively





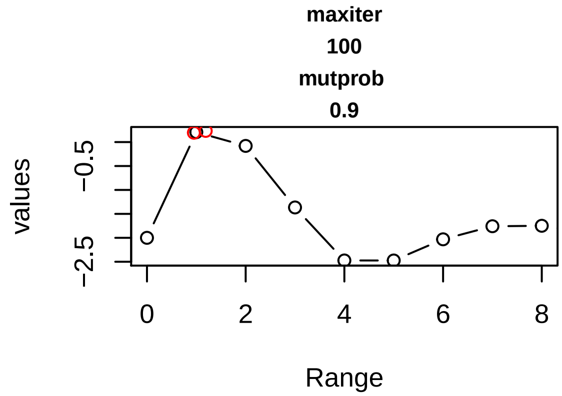


Figure 4-6 (above): Function output with maxiter=100, mutprob=0.1,0.5, 0.9 respectively.

Looking at the plotted graphs, we have two sets of maxiter (10 & 100) and corresponding three sets of mutprob iteration(0.1,0.5 and 0.9) conducted over mutprob.

The black dots represents initial population while red dots represents final population.

The two sets of maxiter with iterations over mutprob, conveys us that when value of maxiter is : a. 10 (low): The sample for final population stays lower (goes maximum between -2 to -1.5) b 100 (high) : The sample for final population aims higher with increase mutation probability (goes maximum around 0.207)

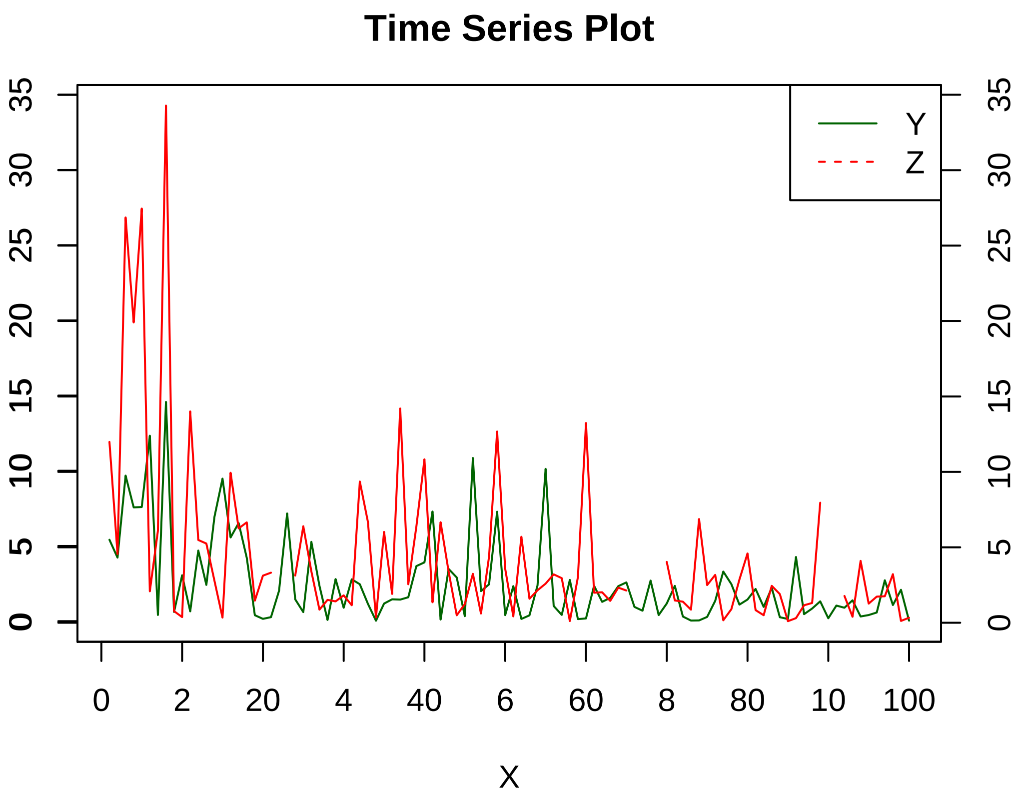
On a closer look at the graph within a short range of x, we see that the difference between initial and final population is high when maxiter is 100 alongwith an intention to catch higher high with advancing mutation probability. This behaviour suggests that the mutated child (or final population) is strong and gets even more stronger with increase in probability, thereby pitching for a stronger population as outcome.

On the other hand,the lower values for maxitor=10 reflects that the mutated child is not stronger. It further highlights the decreased level of variations in the parents. They are not heterogenous enough.

Question 2: EM algorithm

The data file physical.csv describes a behavior of two related physical processes Y = Y (X) and Z = Z(X).

1. Make a time series plot describing dependence of Z and Y versus X. 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?



Looking at the above Time series plot, we observe that both Y and Z decreses over a larger time frame. The data sets is likely from a similar source as both are following same scale and suggest similar time series trend. However, it may be noted that Z tends to spike more and higher when compared with Y, which indicates that it may be more sensitive towards any parameter.

2. Note that there are some missing values of Z in the data which implies problems in estimating models by maximum likelihood. Use the following model $Y_i \sim exp(X_i/\lambda)$, $Z_i \sim exp(X_i/2\lambda)$ where λ is some unknown parameter. The goal is to derive an EM algorithm that estimates λ.

Step 1 : Likelihood function (Estimating lambda , given Y and Z PDF) We Know that : $P(Y_i \mid \lambda) = \frac{\prod_{i=1}^{n} X_i}{\lambda^n} e(-\frac{1}{\lambda}\sum_{i=1}^{n} Y_i X_i)\ldots$equation (1) $P(Z_i \mid \lambda) = \frac{\prod_{i=1}^{n} X_i}{(2\lambda)^n} e(-\frac{1}{2\lambda}\sum_{i=1}^{n} Z_i X_i)\ \ldots$ equation(2)

Now, multiplying equation(1) and equation(2) , we get :

$P(\lambda \mid Y_i, Z_i) = \frac{\prod_{i=1}^{n} X_i^2}{(2\lambda)^{2n}} e(-\sum_{i=1}^{n} -\frac{X_i}{\lambda} Y_i - \frac{X_i}{2\lambda} Z_i)$

$P(\lambda \mid Y_i, Z_i) = \frac{\prod_{i=1}^{n} X_i^2}{(2\lambda)^{2n}} e(-\sum_{i=1}^{n} \frac{X_i}{\lambda} Y_i - \sum_{i=1}^{n} \frac{X_i}{2\lambda} Z_i)\ldots$equation(3)

step 3: Bifurcating Z values into observed and missing or unobserved. We can say that Z(total)= Z(Observed)+ Z(Missing) $\sum_{i=1}^{n} \frac{X_i}{2\lambda} Z_i \Rightarrow \sum_{total} \frac{X_i}{2\lambda} Z_i = \sum_{observed} \frac{X_i}{2\lambda} Z_i + \sum_{missing} \frac{X_i}{2\lambda} Z_i\ \ldots$equation(4)

Step 2 : Log- Likelihood implementation $log(\lambda \mid Y_i, Z_i) = -2n \times \log(\lambda) - \sum_{i=1}^{n} -\frac{X_i}{\lambda} Y_i - \sum_{observed} \frac{X_i}{2\lambda} Z_i - \sum_{missing} \frac{X_i}{2\lambda} Z_i + constant\ldots$ equation(5)

Step 3:Expectation

Now, using equation(5) and calculating its Expected value of Z(missing values) when Y is given at timeframe t of lambda is given by:

$E_{Z_{missing}|Y,\lambda_t} = -2n \times \log(\lambda) - \sum_{i=1}^{n} -\frac{X_i}{\lambda} Y_i - \sum_{observed} \frac{X_i}{2\lambda} Z_i - \frac{\lambda_t}{\lambda}|m| + constant$

where |m| = magnitude of missing values

Step 4 Maximization : Taking partial derivative w.r.t λ and equate it to zero

or, $\frac{\partial E}{\partial \lambda} = \frac{-2n}{\lambda} - \sum_{i=1}^{n} + \frac{X_i}{\lambda^2} Y_i + \sum_{observed} \frac{X_i}{2\lambda^2} Z_i - \frac{\lambda_t}{\lambda^2}|m|$

or, $\frac{2n}{\lambda} = \sum_{i=1}^{n} \frac{X_i}{\lambda^2} Y_i + \sum_{observed} \frac{X_i}{2\lambda^2} Z_i - \frac{\lambda_t}{\lambda^2}|m|$

Finally, Solving for λ in above equation we get :

$\lambda = \frac{1}{2n}(\sum_{i=1}^{n} X_i Y_i + \frac{1}{2}\sum_{observed} X_i Y_i + \lambda_t |m|)$

3.

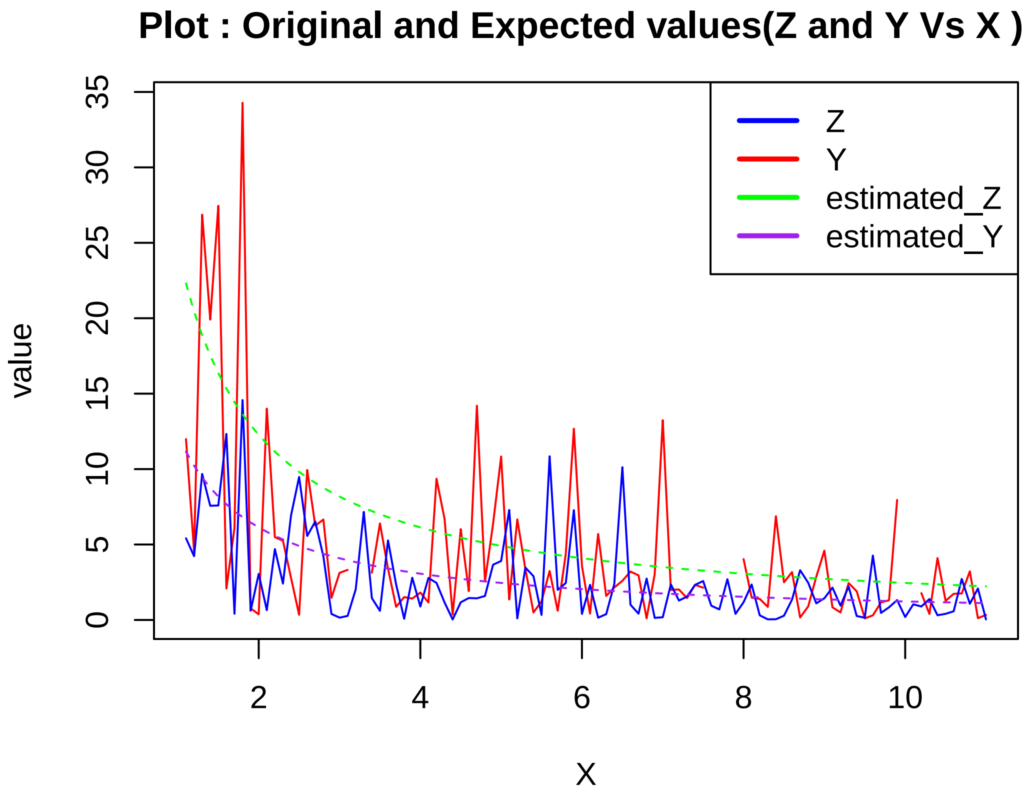
Implement this algorithm in R, use $\lambda_0 = 100$ and convergence criterion "stop" if the change in λ is less than 0.001". What is the optimal λ and how many iterations were required to compute it?

```
## [1] 100.00000  12.26782  10.51318  10.47808  10.47738
```

From the output, we can see that the optimal $\lambda = 10.47738$ which we reach after 5 iterations.

4.

Plot E[Y] and E[Z] versus X in the same plot as Y and Z versus X. Comment whether the computed $/lambda$ seems to be reasonable.



Looking at the graph, we can see that estimated Z (green)and estimated Y (purple)follows the trend of distribution nicely and smoothly through the plotted curve. Also,the estimated Z curve is almost twice the estimated Y curve, which gives us the confirmation of twice the lambda value used for Z with respect to Y.

It is therefore,looking at both curve trend and the difference aspect between curve, we can say that our computed optimal lambda seems to be good and reasonable.