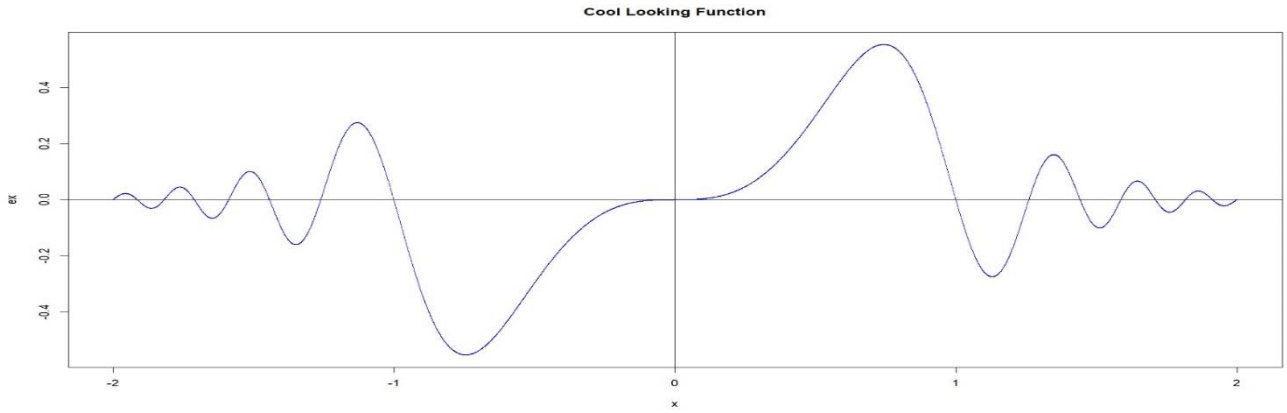
Optimization Methods

For optimization, by using R, we can have various tools: Newton Raphson method, basic Monte-Carlo optimization, Monte Carlo optimization, and simulated annealing. None of them are flawless, and each has negative and positive aspects. In this study, I will highlight each method's differences, advantages and disadvantages. However, first, let me introduce the equation to be optimized.

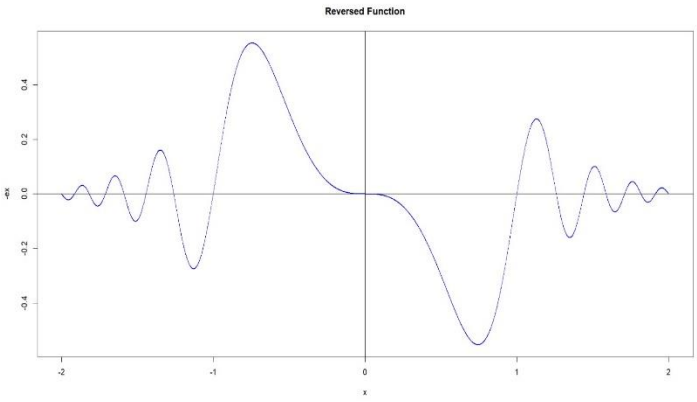
$$\varphi(x) = \sin(\pi x^3)\, e^{(-x^2)}; x \in [-2, 2\,]$$

The reason why this function was chosen is merely due to its interesting plot and challenging optimization procedure. To illustrate my reasons, one can observe the below plot with its multiple local maximum points, which makes it hard to distinguish the global maximum for each method. With its complex shape, it is hoped that each method's weakness will be highlighted more clearly. Also, it resembles a heart signal for people like me who have no idea what an actual heart signal looks like.



1.1) Newton Raphson Method

Newton Raphson method was implemented through the build-in R, nlm package. Since the nlm package only calculates the minimum, we have first reversed the function as follows and calculated the minimum:

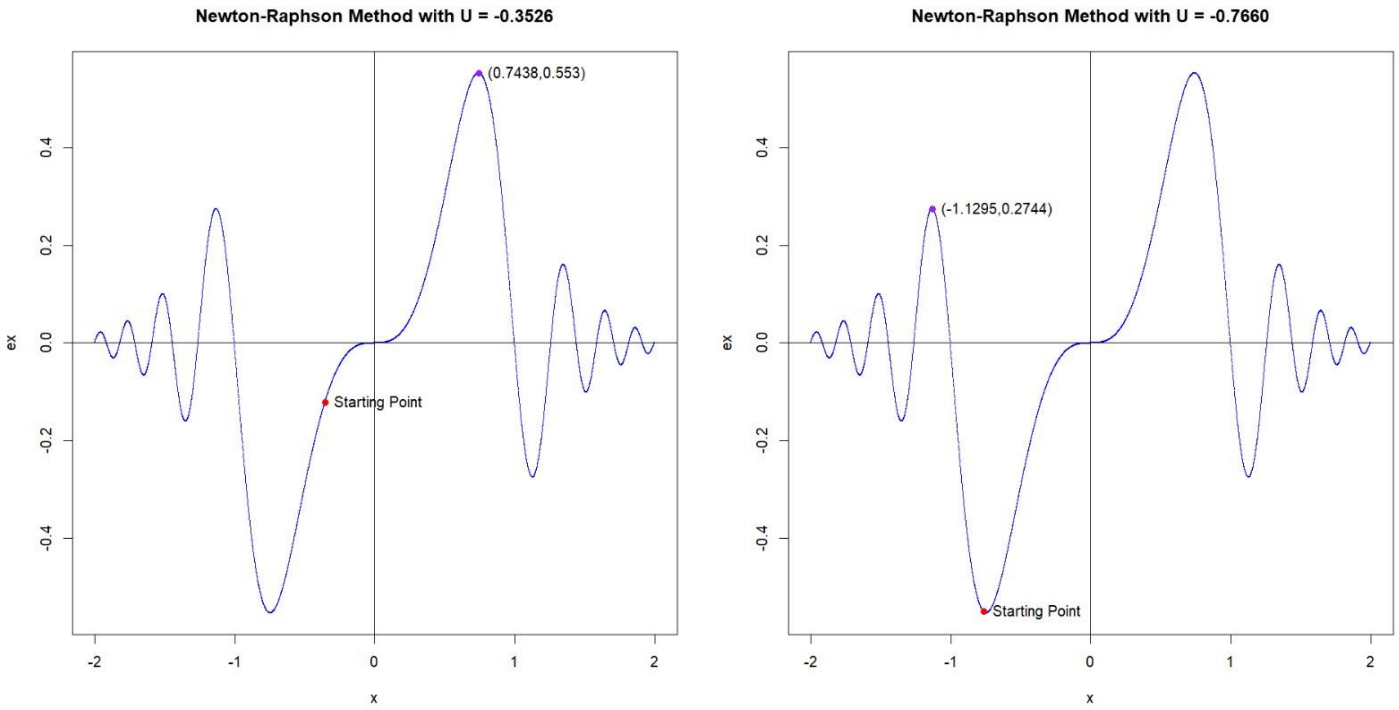


Then, as one can observe, the result multiplied by minus one would give the maximum point where the x remains unchanged.

Also, although Newton Raphson can sometimes provide precise results, it is highly sensitive to the starting point. To put light on this problem, starting point u was chosen randomly for each time where:

$$U \sim Uniform[-1, 1\,]$$

To illustrate the method, the following plots were plotted using two random u's, as mentioned.



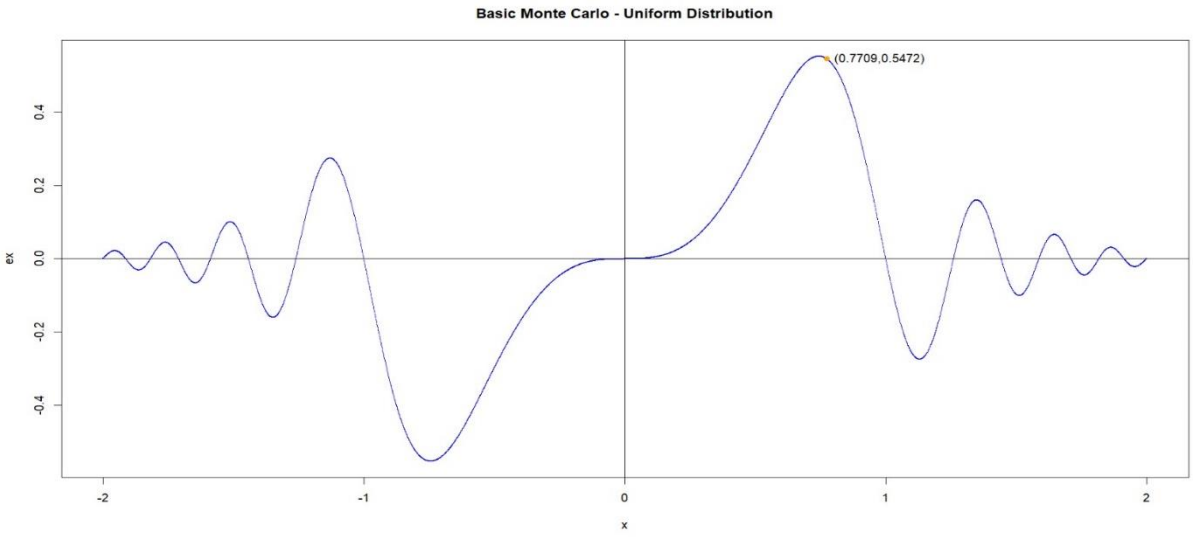
As it can seen, the starting point can highly affect the maximum point. The right-hand side plot is stuck under the local maximum instead of reaching the global point. This is one of the apparent weaknesses of Newton Raphson's method.

1.2) Basic Monte Carlo Optimization

Basic Monte Carlo optimization offers a simplistic approach for functions that are not so complicated and have finite intervals. A major problem often occurs when the functions are defined under infinity. However, in our equation, we do not carry such a problem.

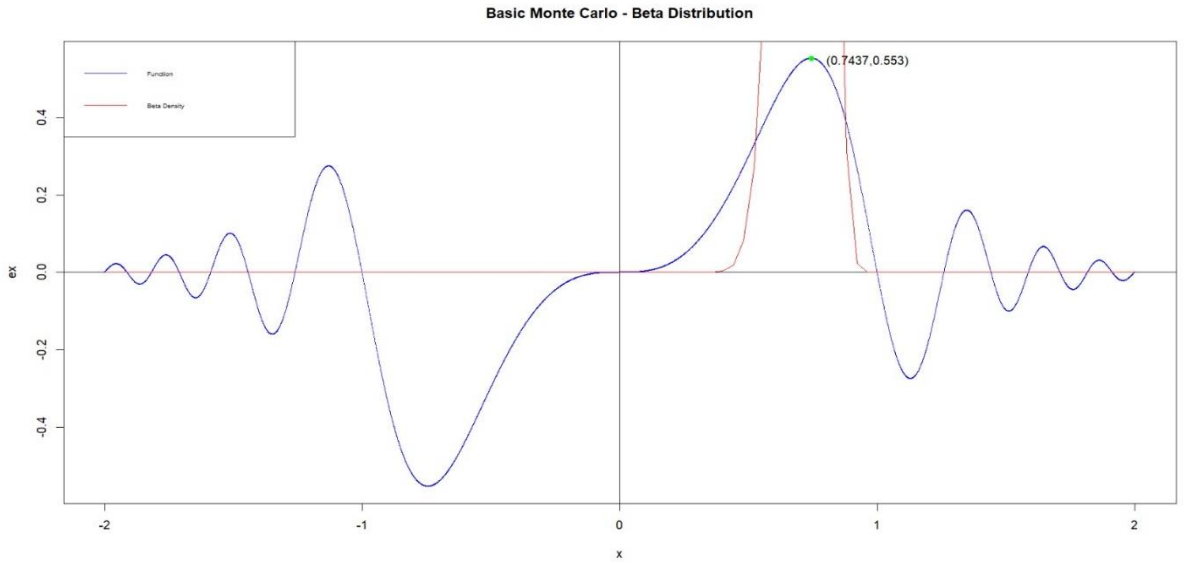
Another problematic part of this technique is that it gives the same weight to each possible point. For instance, in our equation from the plot, it can be observed that the maximum point will most likely be around 0 to 1. Hence, one could claim we are wasting points by weighing each point equally.

The illustrated results of basic Monte Carlo optimization with n = 100 are as follows:



1.3) Monte Carlo Method with Beta Distribution

To avoid the aforementioned weakness of uniform distribution, we can make a wiser choice. By



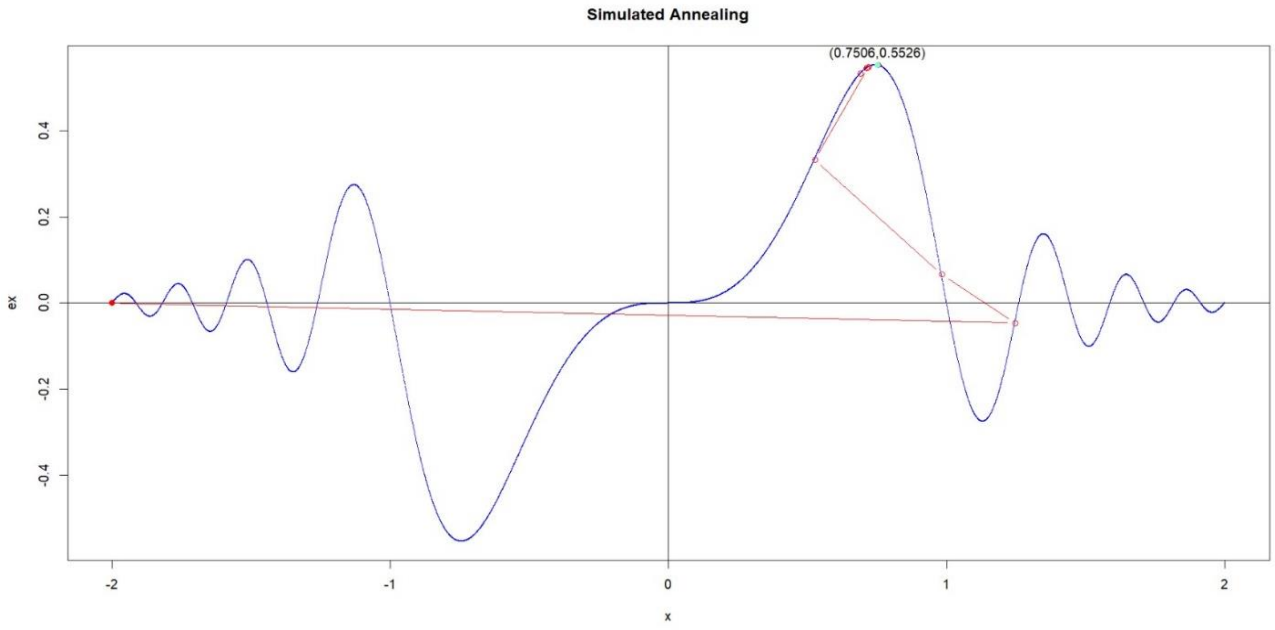
manipulating the beta distribution parameters, one can put more weight under the 0 to 1 interval with more precision. With alpha = 25 & beta = 10, we can center around the curve and prevent wasting observations. The following plot illustrates the beta distribution density and the calculated maximum point with n = 100.

1.4) Simulated Annealing

Simulated annealing can have a hard time overcoming the local points if it is not designed carefully. In this study, in order to avoid getting stuck in the local maximum, firstly, the minimum iteration has held moderately high, and secondly, since we require big jumps, the symmetric interval was chosen widely, and the scale multiplication was removed. As a result, a great success rate has been accomplished. To illustrate my point, in the below simulated annealing run with the following manipulations:

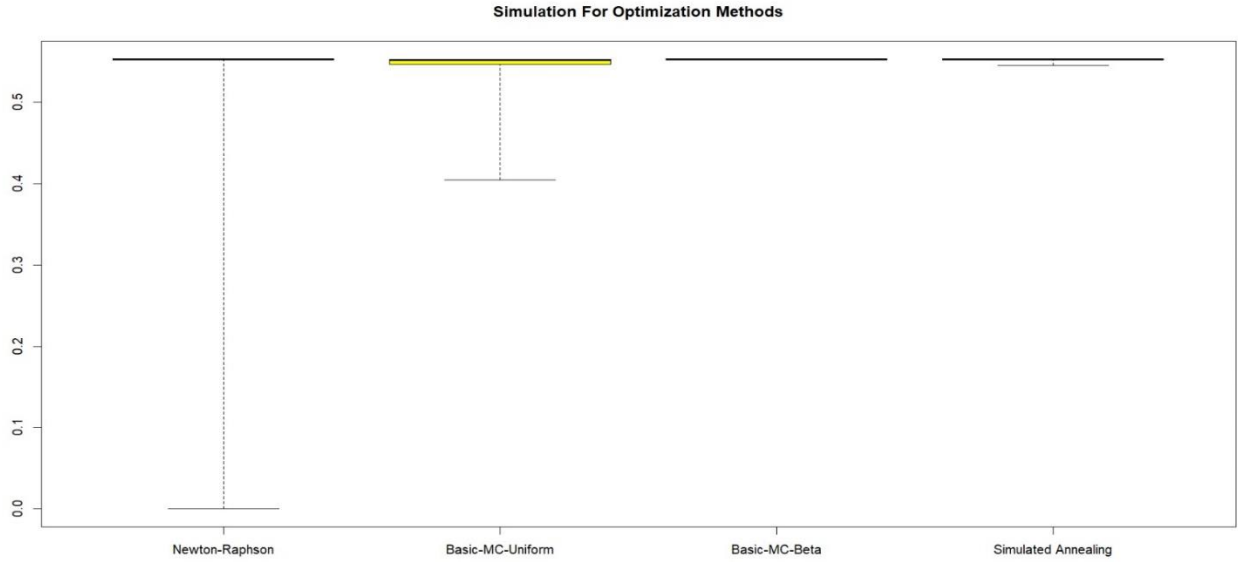
- Minimum iteration = 1000 & Maximum iteration = 3000
- Proposition = $x_i + u \ ; U \sim Uniform[-5, 5]$

The following result was obtained.



2) Simulation Study

A simulation study was carried out to observe the differences and variations for each trial. With B=1000, meaning one thousand trials, a simulation was conducted. The following plot highlights the difference between each method.



The plot visualizes the possible problems we have mentioned. As can be observed, although **Newton Raphson** can be extremely precise for most of the results, it can also deviate significantly from the actual maximum point due to its sensitivity to the starting point. **Basic Monte Carlo optimization with uniform** distribution signifies a large sample size requirement. Although it provides efficient results, most of the time, it lacks precision. Since we had used tricks with **Monce Carlo optimization with beta distribution** by foreseeing the plot, it provided the most significant and reliable result. However, under the circumstances where the plot can not be reached, it might fail dramatically. Lastly, despite the multiple deceptive local maximum points, **simulated annealing** appears to provide efficient results.

In conclusion, under the assumption that we do not have access to a plot or any major information about the function to be optimized, **simulated annealing** would be the best option.