1) Monte Carlo Integration:

Goal:

The aim of this study is to evaluate the integral of function f(x) = 3x^4 + 2x exp(x) + 1 over the interval [0,1]. This function is shown in black in Figure 1. The true value of this integral is 3.60. It will be used as a benchmark when comparing the methods.

Methods:

The first step is to convert the integral into the expectation of some random quantity. Unlike in numerical methods, we can specify the region of importance by using the information provided by a given density function in the setting of simulations. We can achieve this goal in three different ways: simple (basic) Monte Carlo, importance sampling and stratified sampling.

1) Regular Monte Carlo: in the simplest form of MC integration we sample from a uniform distribution on interval [0, 1] and use these RVs as input to the function. Evaluating the mean of the output values is equivalent to one estimate of our original integral over [0, 1]. In other words, we convert the integral into the combination of a uniform density and a function. Next we sample from the uniform distribution, evaluate the function and take the expectation of that to be our estimate of the integral.

We also implement a simulation study to show the convergence of this regular MC as the number of simulations increases (Figure 2). Note that the yellow confidence bands are not true confidence intervals because we are only assessing the results of one specific iteration (i.e. if we were to stop at a given iteration).

2) Importance Sampling: we can always improve our estimates by sampling from a different proposal density instead of sampling from a uniform as in above section. Hypothetically, we have an infinite number of ways to come up with a function/density pair to convert the integral into an expectation. However, there are certain criteria for the proposal that could help improve the performance of the method: 1) proposal density should be as similar as possible to the function and have compatible support, 2) keep abs(h(x))*f(x)/g(x) as constant as possible, 3) density should be approximately proportional to the function.

We propose two different importance densities that look similar to the function (at least on interval [0,1], see Figure 1 red and blue curves). We continue to assess how they perform and whether they improve the method at all: 1) g(x) = 3*x^2 ; 2) g(x) = 4*x^3 .

In addition, we analyze the convergence of a given iteration towards the real value for each of two cases in this section.

3) Stratified Sampling: we also perform stratification as a variance reduction technique and assess its performance for: A) two different allocation rules for number of RVs taken from each stratum, B) two different importance densities that were first introduced in section 2) and C) two different number of strata. Therefore, we have six different variations of the stratified sampling method to compare with other methods.

Comparison:

Figure 5 shows the results of using all the methods that have been discussed so far.

By comparing the Basic boxplot with Importance 1 and Importance 2, it becomes evident that the simple MC performs much better than the proposed importance sampling algorithms in this example. Moreover, we can conclude that stratification improves the accuracy of estimates in every single method by comparing Basic with any one of the Strat series boxplots. Therefore, simple MC is inferior to all stratification methods.

It is worth mentioning that none of the importance sampling proposals seem to work well in this example. Also note that the second proposal (Importance 2 boxplot) performs worse than the first proposal (Importance 1 boxplot). This outcome is not very surprising since the shape of this function is not similar to the function f(x)(see Figure 1, red curve).

This poor result can also be observed in Figures 3 and 4 where the first proposal converges closer to the true value of the integral, whereas the second proposal remains further away at even very large iterations.

Strat A 1 and Strat A 2 are the results of implementing stratification with 10 strata and equal number of RVs from each stratum. Strat B 1 and Strat B 2 are the results of stratification with 10 strata but proportional allocation of RVs to each stratum. Finally, Strat C 1 and Strat C 2 are stratification results for 30 strata and proportional allocation.

Taking Strat A 2 and Strat B 2 (controlling for the number of strata), we can see that proportional allocation improves the accuracy of the estimates as expected.

Taking Strat B 2 and Strat C 2 (controlling for allocation method), we can conclude that increasing the number of strata has reduced the error of estimates even further.

Conclusion:

In this example the stratification with the highest number of strata and proportional allocation for RVs from each stratum is the most accurate method. In addition, we conclude that unless the proposal densities are truly suitable for a given function, they can be subpar, compared to the simple MC from a uniform distribution. We also observed that even the least accurate methods can become more efficient as we increase the number of iterations. Therefore, it is worth attempting to increase the number of simulations where possible.

Figure 1.

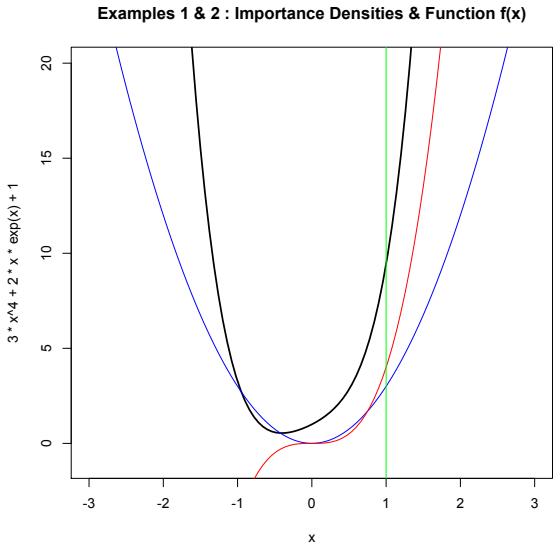


Figure 2.

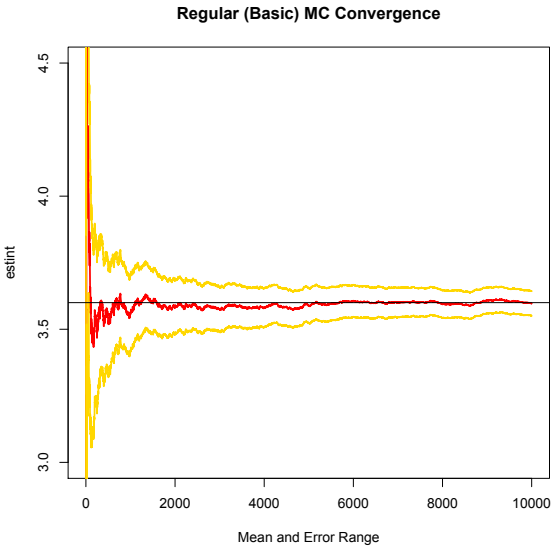


Figure 3.

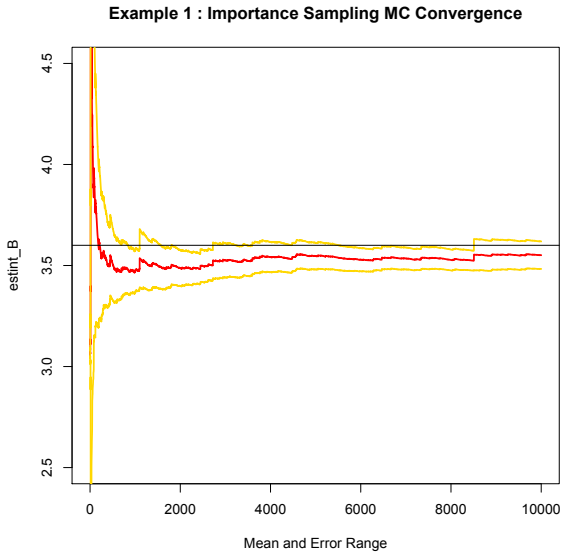


Figure 4.

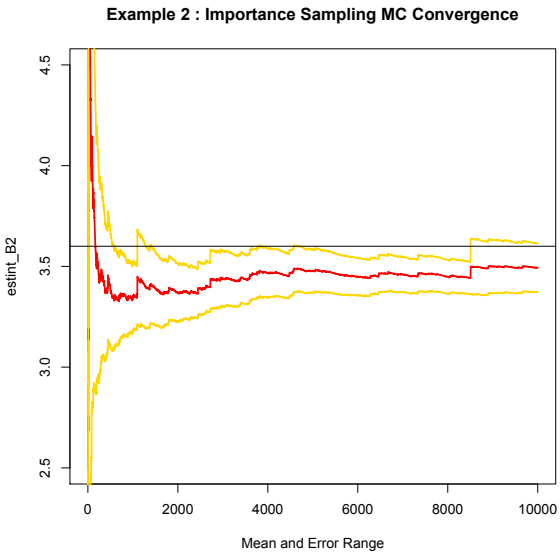
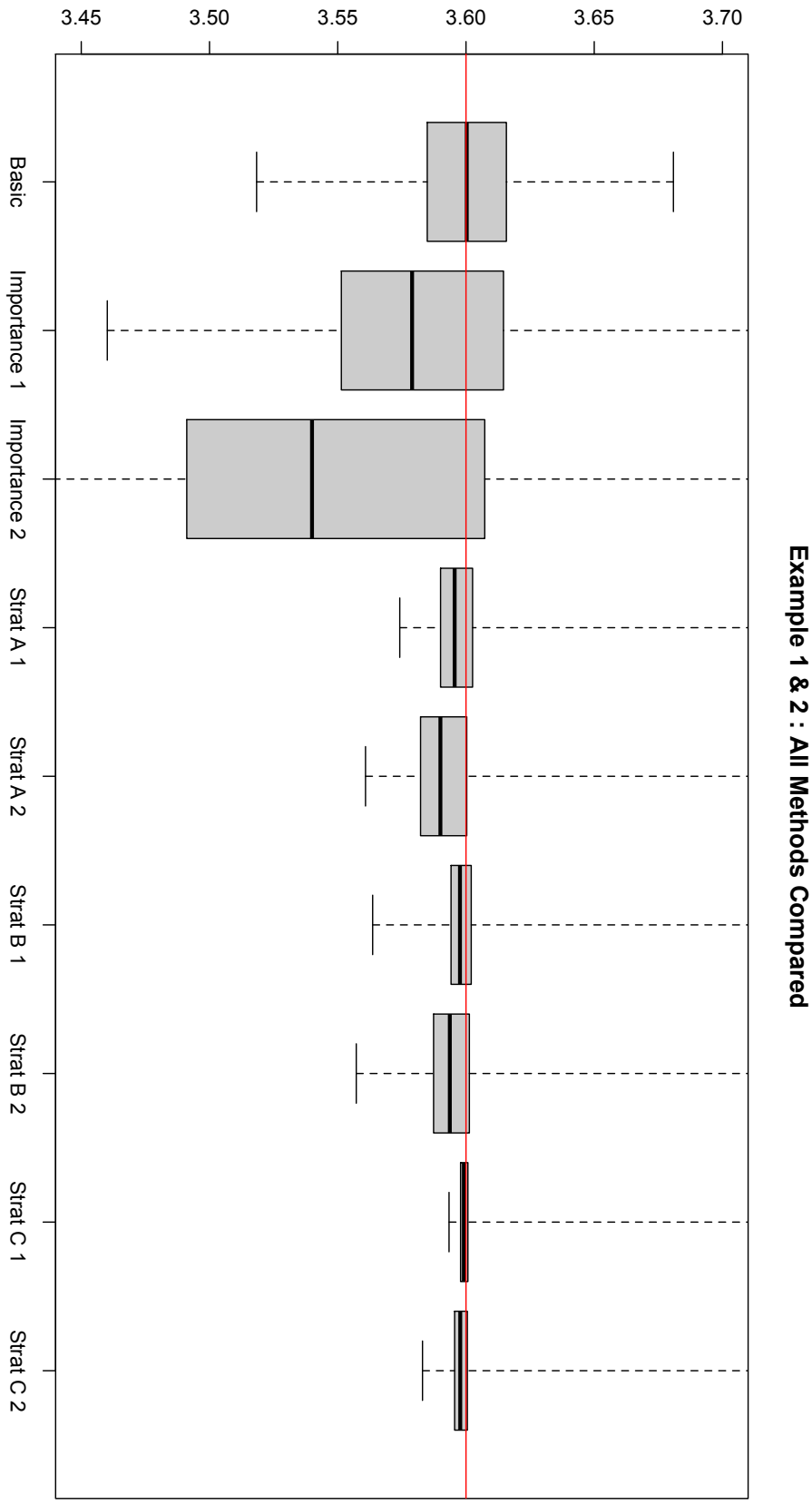


Figure 5.


Example 1 & 2 : All Methods Compared
3.45
3.50
3.55
3.60
3.65
3.70
Basic
Importance 1
Importance 2
Strat A 1
Strat A 2
Strat B 1
Strat B 2
Strat C 1
Strat C 2

Part 2) Optimization Methods:

Goal:

The aim of this study is to find the global minimum / argmin of function f(x) = abs(x*exp(sin(x)^2)) over the interval [-15,15]. This function is shown in black in Figure 1.

It is clear that this function is rather complex because it has many local extrema as well as one global minimum over [-15,15]. The true value of this global minimum is x=0 where f(x=0) = 0. This will be used as a benchmark for simulation studies later in this report.

Methods:

In general we have two main choices for finding the extrema of any given function f(x): 1) numerical optimization methods such as: the Newton–Raphson, and 2) stochastic optimization methods. Stochastic methods are further divided into two main categories of: stochastic search/exploration and stochastic approximation of the function. The main advantage of stochastic methods over numerical ones is that the specific features of the function are much less important in such settings. This is especially advantageous if the function is too complex (which is the case in this study) or its domain is too irregular.

In this report we consider Newton–Raphson as an example of numerical methods and we explore the domain according to four different densities for stochastic methods. In addition, we implement the simulated annealing algorithm in three variations using different temperature schedules.

1) Newton–Raphson: we implement this algorithm by using the built-in nlm() function. We then consider several different starting points to demonstrate whether it performs well or not. Checking whether nlm() gets stuck in any of the local minima would be one way to assess the performance. This method can also break down if the function is highly non-linear.

2) Basic MC: this is the simplest stochastic search method. To implement this, we generate a certain number of uniform random variables and simulate points over domain [-15,15] according to this uniform density. We then input these RVs into the function f(x) and choose the smallest value as an estimate for the minimum. This method is most relevant when the domain has a regular shape. Note that we refer to this method as ‘Basic_1’ in the results.

3) MC Other Than Uniform: in order to ensure that we fully utilize the stochastic search method and not simply use the uniform distribution for exploration, we choose other distributions and evaluate the performance of the method in those cases. The optimal choice for the sampling distribution is problem-dependent but the general rule is to find a density that is as similar as possible to the function. This similarity can be common extrema between f(x) and g(x). After experimenting with several functions, we propose the following three densities that look similar to f(x): 1) g(x) = x^2 ; 2) g(x) = x^2/6 ; g(x) = x^4/200. Figures 1, 2 and 3 respectively show how similar these densities are to the function. Next, we use the Inverse Transform method to sample from the densities and explore the domain accordingly. These methods are labeled MC_1, MC_2 and MC_3 respectively in the results.

4) Simulated Annealing: we take three different pairs of temperature schedules and scales in order to analyze the effect of temperature and scale on the quality of estimates. The optimal choice for schedules is problem-dependent. However, it is generally advised to choose a schedule that is both decreasing and logarithmic (rather than geometric). The advantage of this method is that it does not

get stuck in local extrema unlike numerical methods. Figure 4 shows one evaluation of this method for function f(x). The blue point is the end of the sequence and hollow red points are the remainder values of this sequence. The solid red point is the starting point.

Comparison:

Figure 5 shows the results of the simulation study for all the methods considered in this report. Note that the Newton-Raphson values are not true boxplots since the method is deterministic. Hence we get the same result for every simulation. Nevertheless, they add comparative value and therefore are included in the figure.

By comparing Newt_1, Newt_2 and Newt_3 we can see that the further away the starting point is from the true global minimum, the more likely it is to end up in a local minimum. Newt_2 clearly displays such behavior. As mentioned before, this issue is one of the downsides of numerical methods.

It seems that stochastic search methods perform just as well as Newton-Raphson when considering Basic_1, MC_1, MC_2, MC_3. We zoom in on their boxplots to get a better picture. Figures 6 and 7 demonstrate the results. Figure 6 shows that sampling from other proposed distributions is more accurate and hence superior to sampling from a uniform. It is also worth mentioning that all three proposed densities are effective in estimating the minimum. The differences in accuracy are also very small. This indicates that similarities between g(x) and f(x) were sufficient enough to improve the estimates (i.e. compared to Basic_1).

We also consider the evolution of errors of estimate as the sample size increases for these MC methods. Figure 9 shows the results of this simulation. It appears that MC_1, MC_2 and MC_3 converge very close to the true value of the minimum, whereas Basic_1 remains fairly divergent even at very large iterations. Therefore, Basic_1 is inferior to all other MC methods in terms of reliability.

Next we focus on SA_1, SA_2 and SA_3. Figure 8 shows that the first set of temperature schedule and scale performs better in finding the minimum compared to other two variations. It has to be noted that SA_3 is especially erroneous compared to all other methods considered so far. This suggests that we would need to change our choice of schedule to improve the results.

Conclusion:

The study shows that MC stochastic search methods do best in this specific example. This is especially true for the densities that we proposed (different from uniform). The Newton-Raphson method comes next in terms of accuracy, but only if the starting point of the sequence is chosen to be very close to the true global minimum of function f(x). Otherwise it is likely to get stuck in local minima and hence provide suboptimal results.

The simulated annealing methods vary in their performance. In this case none of the chosen annealing variations performed as well as other methods. However, their main advantage is that they can explore the domain without ending up in local extrema.

All in all, the optimal method is usually problem-dependent and also varies according to the user-defined variables/proposals. Therefore, one needs to be cautious about interpreting the results of a given method and should also experiment empirically to find better choices to improve estimates.

Figure 1.

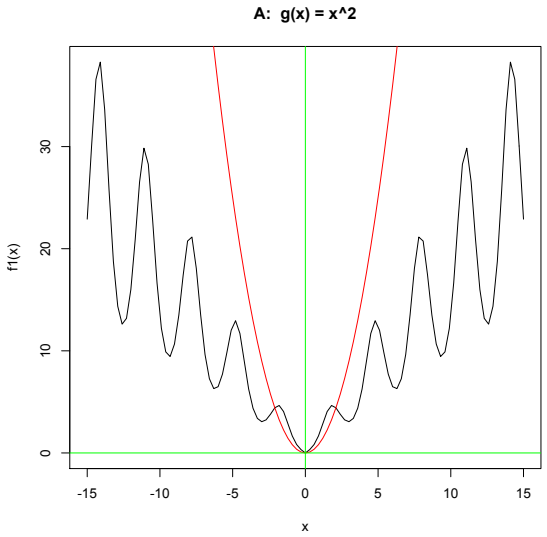


Figure 2.

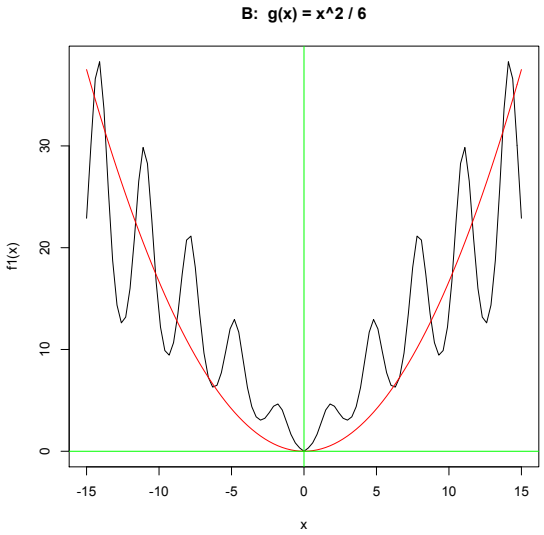


Figure 3.

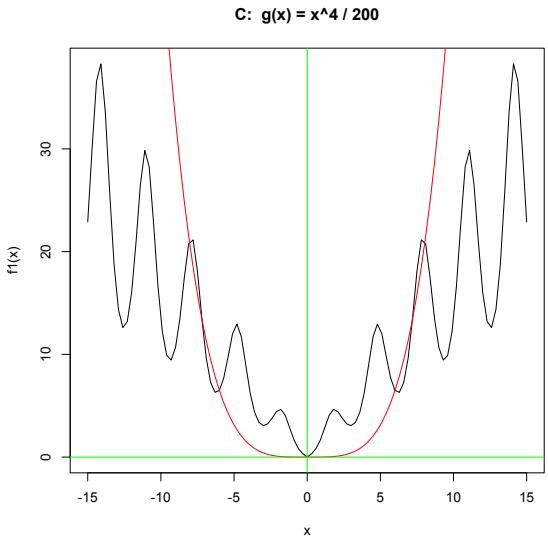


Figure 4.

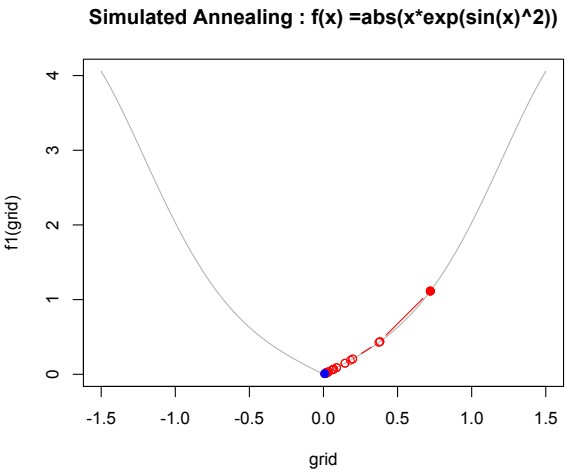


Figure 5.

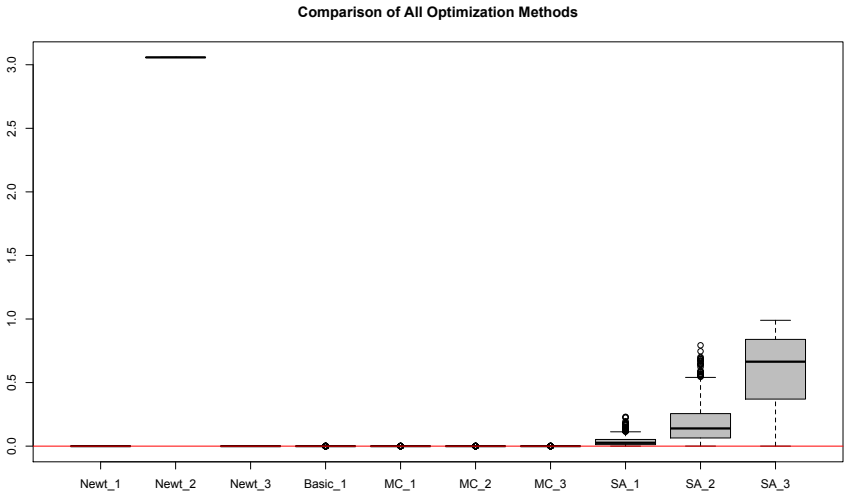


Figure 6.

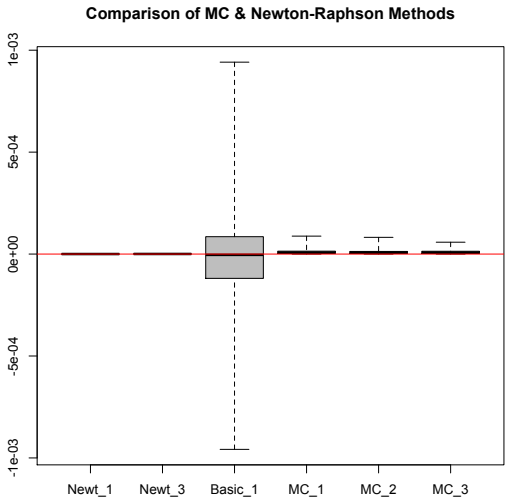


Figure 7.

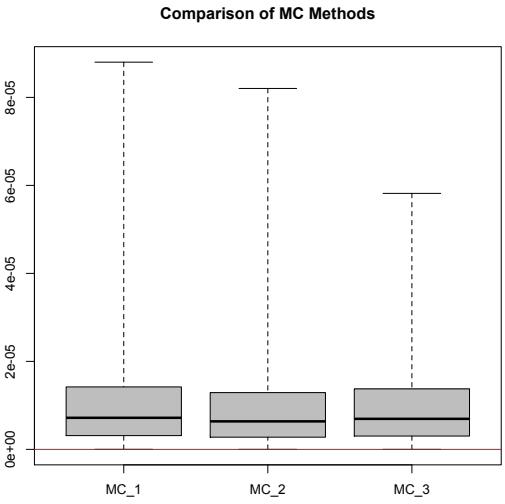


Figure 8.

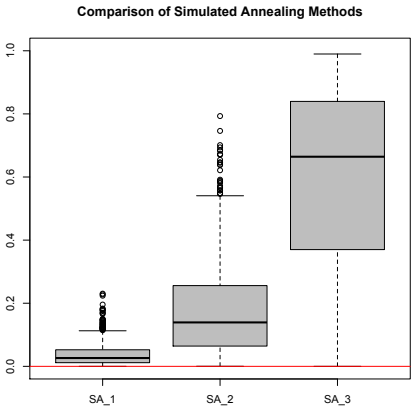


Figure 9.