Name: Yuen Chun Wing, Tyrus SID: 1155077960 Mark: 95 STAT/3
Project 1 Individual Part Computational Statistics Report

1 Introduction

In January, I have studied a note about computational statistics and its algorithm. I will introduce the methods and evaluate them in the following parts.

2 Function of the methods

2.1 Root finding (Bisection & Newton-Raphson)

Root finding algorithm finds the approximate value of x (i.e. roots) when a continuous function equation equals to 0.

2.2 Optimization (Genetic Algorithm)

Genetic algorithm mimics the biological evolutionary mechanism, it figures out the codes or blocks that determine the characteristics of a function. Conduct recursive tests with the all possible combinations such as mutation, crossover, and elimination of the blocks to select high-quality solutions to optimize the system.

2.3 Numerical Integration (Quadrature)

Numerical integration takes advantage of the concepts of integration. Quadrature rules are developed for one-dimension integral. Integral is evaluated by calculating the weighted sum of the selected points at a finite set of points. The integrand can be extended to be the numerical solutions of functions.

2.4 Monte Carlo Integration

Monte Carlo Integration consists of various kinds of sampling method base on probability theories. It takes advantages of the random number to solve computational problems and stimulate numeric results which can be compared to the theoretical results.

The general idea of Monte Carlo method is non-deterministic and base on the law of large numbers. Let X_1, …, X_n be a sequence of independent random variable X. It states that the sample mean of X converges towards the expected value of X which is population mean of X in probability. This law of large number has extended the use of Monte Carlo Methods including Monte Carlo integration.

Monte Carlo integration gives an approximation of a complex definite integral by the expected value of a function. It is often applied when the closed-form solution is too difficult to obtain.

2.5 Summary

The above-mentioned algorithms bring the users the solution of complicated functions by various techniques and theorems including derivatives, integration, intermediate value theorem and law of large numbers. Real-life models and phenomenon such as natural selection are taken as the reference to developing a new system-optimizing algorithm.

3 Procedures of the methods

3.1 Root finding

One essential element of root finding is setting up a precision to stop repeating the iteration.

3.1.1 Bisection

The bisection method takes advantage of the root located theorem which states that there must be a root between x_1 and x_2 if the product of x_1 and x_2 is smaller than 0.

First, we obtained $f(x_1)$ and $f(x_2)$. If the result is larger than the precision. Calculate x_3 which is bisection of x_1 and x_2. If $f(x_3)$ is still larger than 0. Calculate bisection of a and x_3. Repeat until $f(x_n)$ is smaller than the precision.

3.1.2 Newton-Raphson

Newton method takes advantage of Taylor's series expansion.

First, we obtained $f(x_1)$ which is close to the root of the function. By Taylor's approximation and definition of root, we have $f(x_1) + f'(x_1)*\Delta x=0$. $\Delta x=-[f(x_1) / f'(x_1)]$.

By expanding the tangent line at point x_1, we can update x_{i+1} that equals $x_i+\Delta x$. Recursively calculate $\Delta x=-[f(x_{i+1}) / f'(x_{i+1})]$ by the previous result until f $(x_n)<$ the precision value. x_n is the root we find.

3.2 Optimization (Genetic Algorithm)

Genetic algorithm stimulates the model of natural selection and evolution based on "survival of the fittest".

First, we represent the special properties of all the possible solutions (i.e. individuals) such as using binary digits 1 and 0. Next, initialize by randomly generating individuals and randomly distribute the individuals to populations. Prepare an objective function to judge the fitness of the individuals. Sort out all the individuals according to its value of the objective function to the optimization problem. More fitted individuals are selected stochastically and then reconstructed to establish a brand-new generation. The best selection is obtained by recursively mutating, eliminating and modifying of generations until a solution generated reaches a set satisfactory fitness level or it reaches the set maximum number of trials.

3.3 Numerical Integration (Quadrature)

A family of rules is constructed for different kinds of functions. For example, rectangle rule is applied to polynomials. Let the interpolation function be constant and it passes through the point $\left(\frac{a+b}{2}, f\left(\frac{a+b}{2}\right)\right)$. The approximated value of the integrand is: $(b-a) * f\left(\frac{a+b}{2}\right)$. The trapezoidal rule is applied for a polynomial with degree 1 (i.e. an affine function). Let the function passes through polynomial of degree 1) which passes through the points (a, f(a)) and (b, f(b)). The approximated value is: $(b-a) * \frac{[f(a)+f(b)]}{2}$.

3.4 Monte Carlo Integration

By the general Monte Carlo theory, the Monte Carlo can be calculated by the average of the function output of x in N trials. Random numbers are drawn uniformly but with a probability distribution as samples and the inversed CDF are computed for converting uniform random numbers to non-uniform.

To conduct the integration, a sufficient number of random samples is generated, usually 1000000 random numbers or above to ensure an error lower than $5*10^{-5}$. Choose appropriate random variable X according to the domain of X. Stimulate the random values in the function and compute their own results for all the random numbers. Obtain the average sum of the simulation, this is interpreted as the average value of y when x is drawn randomly from the selected probability distribution and the approximation of the integral.

Markov Chain Monte Carlo methods contain various sampling methods based on the idea of Markov chains to change the desired distribution to an equilibrium distribution.

4 Advantages of the methods

4.1 Root finding

4.1.1 Bisection

This method is always convergent once we confirm there is a root within the interval. The guess length keeps diminishing by halving the interval in each iteration. It is possible to find out all the roots when we divide the interval into small parts as much as the computer can.

The approximated value of root becomes more accurate as the time of division of interval size increases. The error we bound within the interval is halved for each iteration.

Also, it can control the error by changing the maximum error accepted (i.e. the precision).

4.1.2 Newton-Raphson

It is the most popular method of root finding. It has a fast rate of convergence because of the rapid decrease of the error term for each iteration. The number of iteration will be greatly reduced. It approximates better than the other methods because of its quadratic characteristic. The decimal places of the guess doubles in each iteration. The root produced by other method can be updated to higher accuracy by Newton method. Only an initial guess is needed only.

4.2 Optimization (Genetic Algorithm)

It is an efficient algorithm to easily search all the possible solutions in the sample space of the solution and compare the individuals. Also, it has high flexibility to decide the amount of randomly generating the initiation.

4.3 Numerical Integration (Quadrature)

It is simple to calculate the integral by using the rectangle and trapezoidal rules requiring a few points of the functions. So, a wide variety of numerical integration task can be achieved by the two rules. It performs the best especially on the linear functions such as piecewise linear functions.

Rectangle rules need one fewer function evaluations for the intervals than the trapezium rule.

4.4 Monte Carlo Integration

Monte Carlo methods can handle complicated problems even with a high dimension, complicated domain, and multiple integrals. Convergence rate of MCMC will not be affected by the dimensionality or complexity.

It is not costly with advanced computer technology, only required a set of random samples following a probability distribution, computers will finish the tasks of recursive sum and average of data in a short time compared to the time consumed by human served.

The data generated by this method is unbiased, also consistent. The findings can be easily described in a graph. It provides convenience to communicate the results to the readers. Also, as we produce as many as samples, tends to infinity, this method is capable of converging to the intrinsic value of the integral.

5 Disadvantages of the methods

5.1 Root finding

5.1.1 Bisection method

Convergence is very slow compared to other methods. The amount of error is halved during each iteration. Moreover, it is necessary to choose an appropriate interval to prevent the need of many iterations to converge. The initial lower guess is close to root, the number of iterations increases dramatically because of the long distance from the initial upper guess to the root. Also, the bisection method fails to find all the roots if there is more than one root. Two initial guesses are needed to use this method.

There are special cases that bisection method can’t figure out the root. When the root is also the peak value of the function. The function outputs of the two sides are identically positive or negative. Also, if the function having an asymptote, the signs of lower guess and upper guess are different but there is no root within the interval. For example, there is no roots between x=1 and x=-1 in $f(x)=x^{-1}$. Bisection method fails to find out the root in these situations. Other methods should be applied.

5.1.2 Newton-Raphson

It may not compute a root if the initial guess is too far from the real roots.

It is necessary to find out the function and the derivative of the continuous function for every iteration which is very computationally costly. Also, it has poor global convergence, when local maxima (i.e. $f'(x_{n-1}) =0$) are obtained in one iteration, the method results in negative infinity. In some special functions, the slopes of tangent lines between two consecutive iterations are identical. This method enters a loop in such case and a new initial guess is recommended to re-run the method.

5.2 Optimization (Genetic Algorithm)

Programming is a complicated and difficult process. The first step is coding the proprieties of solutions, then selecting the most optimized solution and finally decode the solution.

Three operations in this algorithm, mutation, crossover, and selection, consist of many parameters. And the variation of these parameters causes a serious effect on the quality of the solution.

5.3 Numerical Integration (Quadrature) & Monte Carlo Integration

Intrinsically, the solution calculated is an approximated value which implies that the approximation must over- or under-estimates the integral, especially the function is strictly decreasing or strictly increasing. Bias is produced in this method. Numerical integrals fail to compute when the range of upper limit or lower limit is indefinite. The results will be not trivial. It is computationally costly when the dimension of integral is large. The rate of convergence will decrease exponentially because N time more samples are required as the dimension increases.

Trapezoid over a large interval will cause a large error since the curvature is not considered in the calculation. The interval can be cut in half to reduce the amount of error. The trapezoidal rule may consist of singularity since it evaluates f at endpoint a or b.

When we apply Monte Carlo methods to calculate the integrals, many of them will diverge to infinity at the endpoint, one or both. The rate of convergence is relatively slow. And the variance of the function can't be derived easily. So, it is hard to evaluate the approximation error.

The accuracy of the estimate decreases as the sample variance increases. It is appropriate to choose a sample set with a lower variance to generate the result efficiently. A lot of sampling techniques are invented for variance reduction such as adaptive sampling and importance sampling. The latter method concentrated on the problem of the small value of PDF. Small PDF leads to a large value of function dividing PDF f(x)/p, the large sample will cause a sample mean skewed from the true mean. We can solve it by avoiding the small value samples by getting more appropriate sample in the interval that f(x) has a high value.

6 Evaluation of performance of the methods

To evaluate the performance of the methods, we can analyze the amount of error. Examine the error size and percentage relative to the approximate value generated by the methods.

6.1 Root finding

6.1.1 Bisection

Due to halving the width of the interval [a,b] for each iteration, the maximum error h= b-a is halved in each iteration. We can derive the error term ε after n iteration is $h*1/2^n$. It is possible to calculate the number of iterations needed if we fix the precision criteria. Minimum steps for reaching the root is $\log_2((b-a)/\varepsilon)$.

6.1.2 Newton-Raphson

By Taylor's approximation, the function f(x) = 0 is $f(k) = f(x_n) + (k - x_n)f'(x_n) + \frac{1}{2}(k - x_n)^2 f''(t_n)$. The error is then calculated as $k - x_{n+1} = (k - x_n)^2[-f''(t_n)/2f'(x_n)]$ where k is the root of the function and t_n is an arbitrary point between x_n and k. Estimation of error can be $k - x_n = -\frac{f(x_n)}{f'(t_n)} \approx -\frac{f(x_n)}{f'(x_n)} \approx x_{n+1} - x_n$ because of the quadratically increasing error.

6.2 Optimization (Genetic Algorithm)

The concern of genetic algorithm is not on the convergence speed which is the quality of the solution computed. We can evaluate the performance of the quality and the computational cost. The quality of the solution can be measured by the average fitness value which is average of the highest fitness values get within the number of generations in the number of runs (i.e. iterations). We can also calculate the likelihood of evolution leap Lel(k) which is the average number of solutions at the

Efficiency should also be considered in evaluation.

generation performing better than the best solution before the current generation divided by the number of runs.

6.3 Numerical Integration (Quadrature)

The Rectangle and trapezoid rule applies to always nonnegative or non-positive function $\omega(t)|_{[a,b]} = \prod_{i=0}^{n}(x - x_i)$. The general error term is $f^{(n+1)}(c)/(n+1)! \int_a^b \omega(t)\, dt$. And that for rectangle rule is $\frac{1}{2}f'(c)(b-a)^2$ where n=1. The error term for trapezoidal rule is $\frac{1}{12}\mathrm{f}''(\mathrm{c})(\mathrm{b}-\mathrm{a})^3$ where n=2.

6.4 Monte Carlo Integration

According to the law of large numbers, the result converges to the true value when the number of samples tends to infinity. We can examine the error term by variance Var[E(f(x))]. The sample variance of f(x) is the summation of the square of the difference between f (x_i)-E (f(x))$]^2$ dividing by sample size minus one. We can compute the variance of E [f(x)] with the use of the above formula which is as follows.

$$\mathrm{Var}[\mathrm{f(x)}] = \frac{1}{\mathrm{N}-1}\sum_{i=1}^{N}[f(x_i) - E(f(x))]^2$$

$$\mathrm{Var}\left[\sum_{i=1}^{N} Y_i\right] = \sum_{i=1}^{N} Var[Y_i] \qquad Var[E(f(x))] = \frac{1}{N}Var[f(x)]$$

7 Real-life application of the methods

7.1 Root finding

Root finding gives an approximate root efficiently, it is close to the real roots and it is adequate for most applications. In business, we can figure out the minimum amount of sales required for covering the cost in a selected period. First, we figure out the function of total sales and total costs. Then, set up a precision value which the tolerance. Finally, we can apply the root finding algorithms to calculate the approximate minimum amount of sales which is minimizing loss functions.

7.2 Optimization (Genetic Algorithm)

This genetic algorithm was used in machine learning. An interesting example which is called MarI/O is published recently. The author had written a program made of neural networks and genetic algorithms. Neutral networks stimulate all combinations of possible inputs which is the control pad and ABXY buttons in human brain. The program evolves by time with finite times of trials, it recursively tries to reach the goal with different inputs until one combination of input is derived to arrive the goal. The program finally controls the player ‘Mario’ finishing the stage in Super Mario World perfectly.

7.3 Numerical Integration (Quadrature)

In material science, area of different material is calculated by the numerical integration. The rules are appropriate to find area of irregular shapes as we have the function of the area of the material.

7.4 Monte Carlo Integration

It can be used to predict in finance. Monte Carlo integration generates the random samples to stimulate the degree of effect of various factors on price of financial instruments. Then we can determine the probabilistic distribution to approximate the price of an instrument. For example, valuing of an option will stimulate many possible price paths for shares. Compute and discount the payoff to today’s price to make a better decision of the selling price of the option.

8 Conclusion

These methods provide a general idea of evaluating solutions of functions. They have unique and corresponding advantages and disadvantages. Appropriate method or algorithm should be applied in the corresponding situation such as the degree of a polynomial to increase the accuracy of the result. Not only can this method be used in fields of Statistics and computer science, but also the business sector, science field and the social science research.