**Iterative Methods in Optimization: Newton and Secant Method**

**Yann-bor Wen**

**Dr. Kojouharov**

**Math 5300**

**3 May 2018**

**I. Statement of Problem**

Optimization is a branch of mathematics that is concerned with development of algorithms that guarantee convergence to an optimal solution. It is used in many fields including engineering, finance, topology, and statistics. It is a technique for finding a maximum or minimum value of a function of one or more variables subject to a set of constraints. This equates to finding the root of the derivative of a function. The use of iterative root finding methods such as Newton’s method, secant method, and bisection method is a popular way to find these roots. In this project we will focus on a specific example in the field of economics in which the goal is to maximize the profit function of a competitive firm using the Cobb-Douglas production function. We will use the newton method and secant method with two parameters (in three dimensions) to perform iterations to reach optimization and compare the results to that obtained by the actual solution equations obtained by using calculus. More precisely to the problem, we will find the values of labor and capital for which profit is maximized. Error analysis for each variable will be performed on each method.

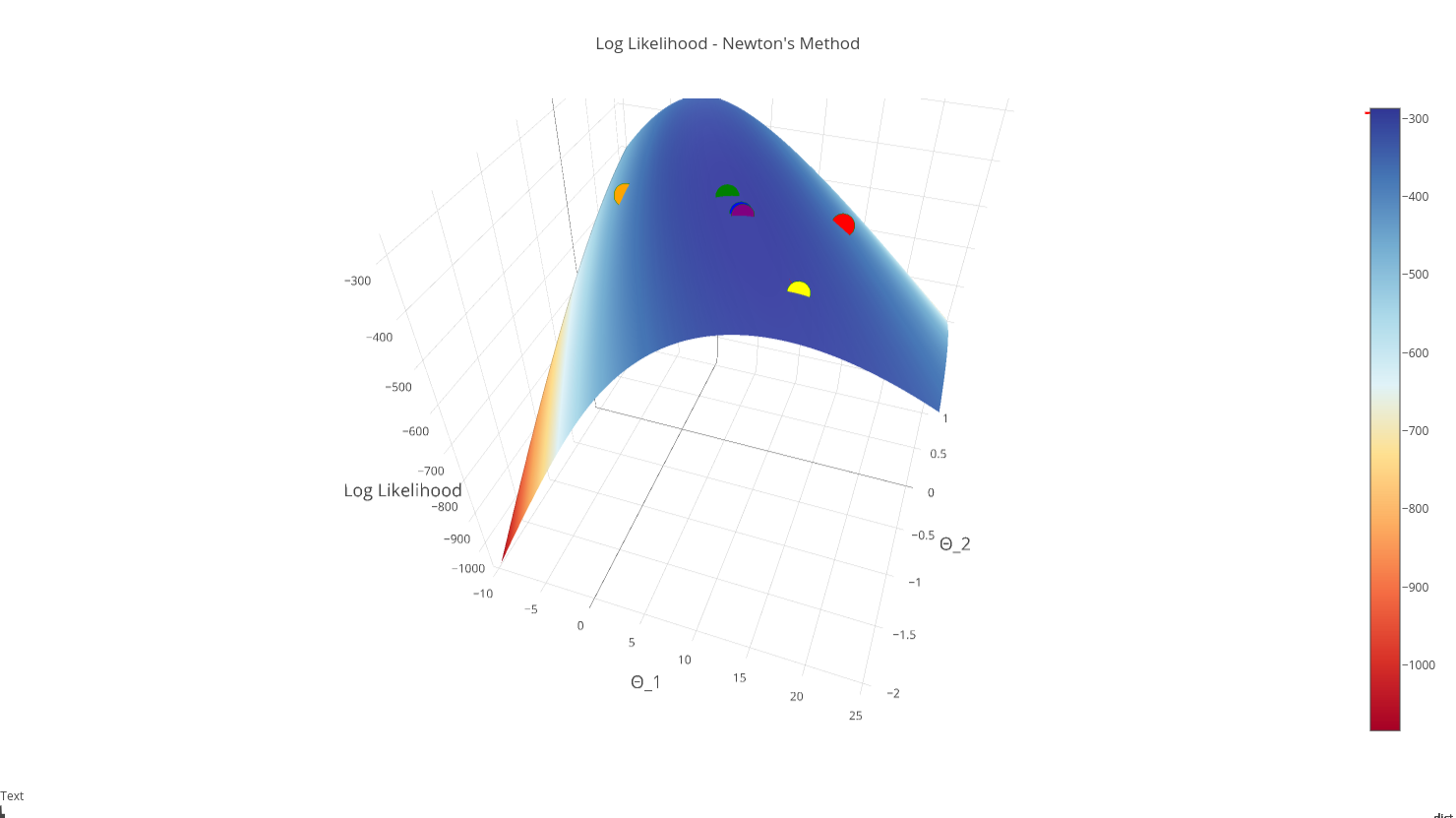


Figure 1. Newton’s iterative method on the log likelihood function of two parameters

**II. Description of The Mathematics**

Consider a competitive firm with the following profit function

*π = P Q − wL – rK*

where P is market price, Q is output, L is labor, K is capital, w is the wage rate, and r is the rental rate. The variables P, w and r are dependent on outside factors, as the firms is operating in a competitive market. The Cobb-Douglas production function relates Q to K and L

*Q = LaK­­b*

where a and b are positive parameters. Assuming decreasing returns, a+b<1. For simplicity in this example we will consider the symmetric case where a=b=1/4. The profit function then becomes

*π = P L1/4K­­1/4 − wL – rK*

Newton and Secant Method in Optimization

One common problem that newton and secant method is used for is finding the roots of a nonlinear function. In optimization problems, we want to find the maximum or minimum of a function subject to some constraint(s). Finding the root of the first derivative of that function will give us the points of global and local maximum(s) and/or minimum(s). Therefore, these iterative methods will be a perfect choice to use in optimizing the function. There are some cases in which each method will fail. In newton’s method, the derivative may be zero at the root, the function may not be twice continuously differentiable on the interval of interest, and some functions cause the iterations to bounce around, never converging. One must also choose the initial point wisely lest it be outside the range of convergence. In secant method

Newton and Secant Method in More than One Dimension

When performing newton and secant method in more than one dimension, we will be finding the roots of the gradient function. The gradient of f is a vector of its partial derivatives [∂f /∂x, ∂f /∂y, ∂f /∂z…]. In the Newton method, the Hessian, which is a matrix of the second partial derivatives of f, is used [fxx fxy : fyx fyy ]. For multiple parameters, the newton method takes on the form

xn+1 = xn − H−1 (x­­n)∇f( xn)

where x represents the vector of parameters. We take the inverse of the Hessian of f, H(x­­n), since it is of matrix form and it wouldn’t make sense to divide by it. The gradient of f, ∇f( xn), is its vector of partial derivatives. We note that the dimensions of the inverse of the Hessian and gradient must match to perform matrix multiplication. The formula for secant method for multiple parameters is

xin+1 = xin − [(x­­n-xn-1)/{∇f(xn) - ∇f(xn-1)} x ∇f( xn-1)]

Error Prediction

For this problem, we predict the errors will be less than the actual values since we are maximizing. Newton’s method is predicted to be faster because the convergence is quadratic. Also, the function calculations per loop would be more in secant method since we are calling each value of the gradient matrix. In the newton’s method, the inverse Hessian can be multiplied directly with the gradient vector. This is only true in the multi parameter problem as the gradient would be two formula computations per iteration as opposed to one in the one variable problem.

**III. Description of the Algorithm**

For this model, the profit formula *π = P L1/4K­­1/4 − wL – rK* was used where L and K are the parameters in question. This equation was entered as f in Matlab. The code finds the partial derivatives and second partials with respect to each parameter variable. P = 1000, w = 20, and r = 10 were used arbitrarily (they are known variables subject to the market). The initial guess of P=1 and K=1 was taken arbitrarily as well. We can do this because from earlier examination of the Hessian, it is seen that a maximum exists (further Theorem below).The following segments of code were implemented in MatLab.

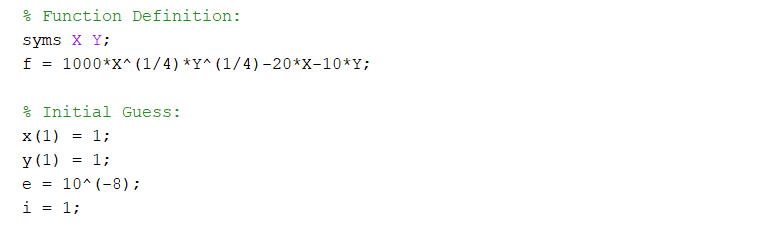
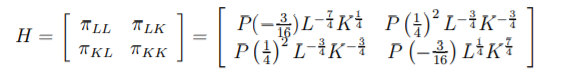


Figure 2. Function definition, Initial values, and Tolerance.



Figure 3. Secant Method uses an extra second starting point

We took the initial values for X and Y to be 1 arbitrarily. We are able to do this because we have verified the second order conditions needed to have a maximum. To do this, take the Hessian of the original equation to get a matrix:



Since H1 ­< 0 and H > 0, the sufficient conditions are met. In general, the following theorem provides a sufficient condition for finding local maxima or minima:

Let U → R be twice continuously differentiable, where U is an open subset of Rn, and the first order condition ∂f /∂xi(x\*) = 0 holds for some x∗ ∈ U:

1. If the Hessian matrix ∂2f /∂xi2(x\*) is a negative definite matrix, then x ∗ is a strict local maximum of F.

2. If the Hessian matrix ∂2f /∂xi2(x\*) is a positive definite matrix, then x ∗ is a strict local minimum of F.

3. If the Hessian matrix ∂2f /∂xi2(x\*) is an indefinite matrix, then x ∗ is neither a local maximum nor a local minimum of F In this case x ∗ is called a saddle point.

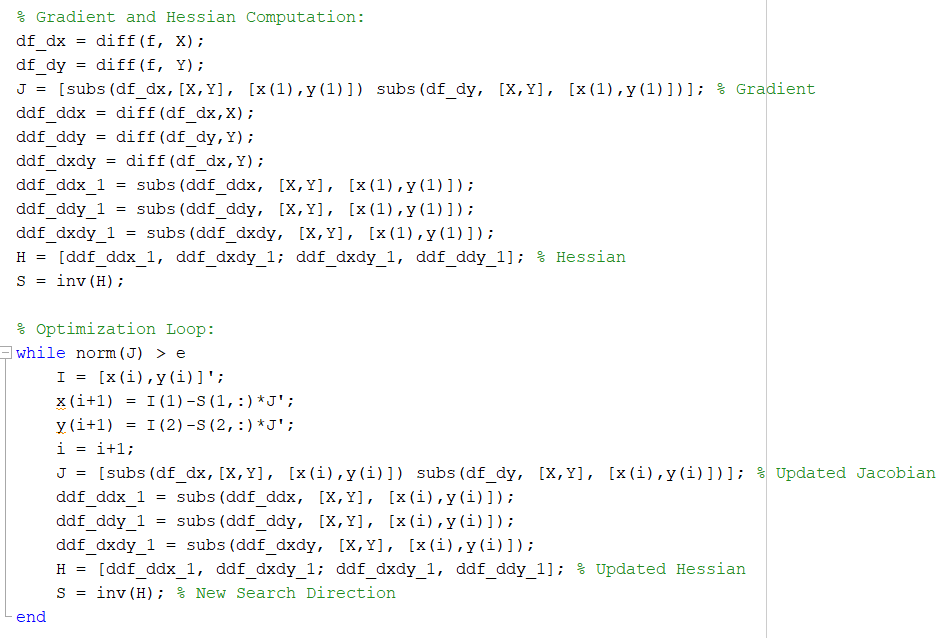


Figure 4. Code for Newton’s Method Gradient and Hessian Computation and Optimization Loop

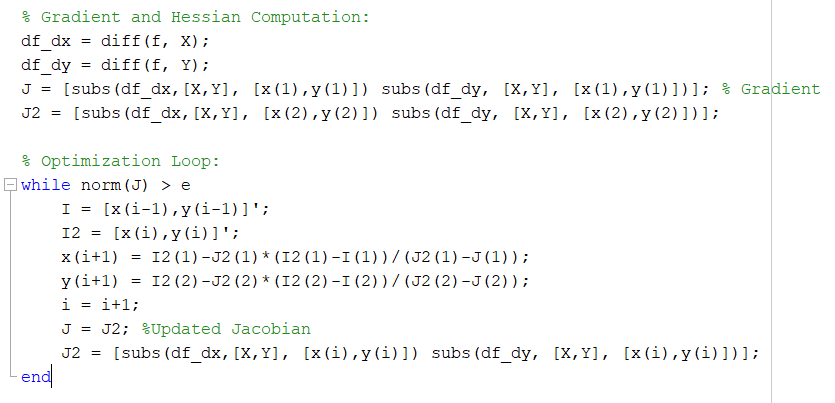


Figure 5. Code for Secant Method Gradient Computation and Optimization Loop

**IV. Results**

The results obtained in Matlab output are included at the end of Results for reference. The figures below show the contour plots connecting each pair of iterations. The yellow “level” is the highest and blue the lowest. The x axis represents L labor and y axis represents K capital.

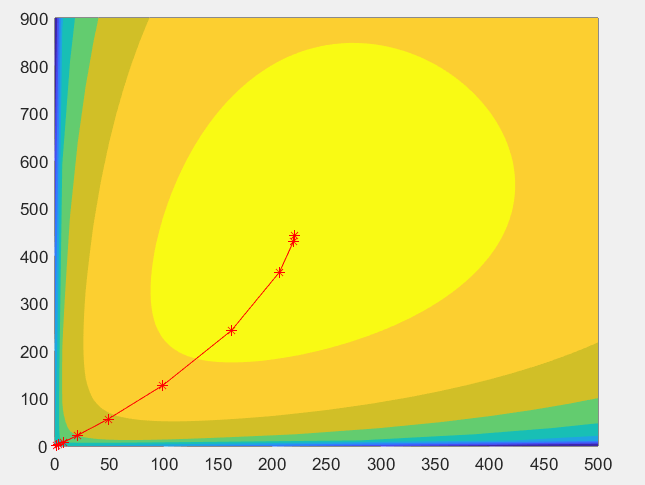


Figure 6. The Newton’s Method Contour Plot with Iterations

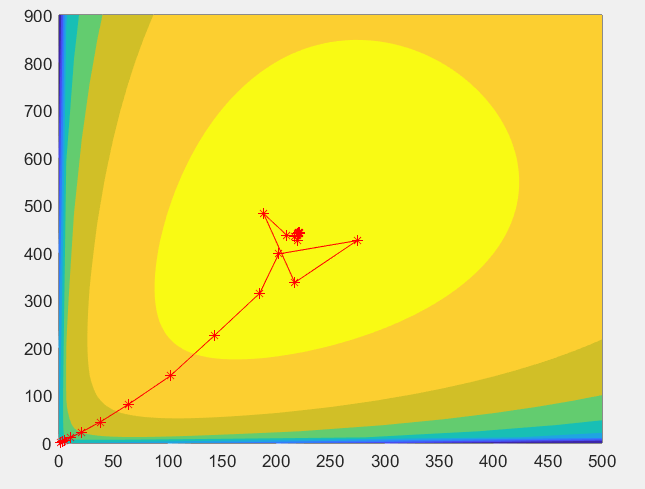


Figure 7. The Secant Method Contour Plot with Iterations

The tables below show the result of approximately 220 units of labor and 442 units of capital are required to reach a maximum profit of appx 8838.84, with the newton method as the winner. The number of iterations was less and X value and Y value closer to the actual than those of the secant method.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Iterations | X Maximum (L) | Y Maximum (K) | Function Maximum (*π*) |
| Newton | 12 | 220.970869120794 | 441.941738241565 | 8838.834765 |
| Secant | 57 | 220.970869042187 | 441.941737771931 | 8838.834765 |
| Actual |  | 220.970869120796 | 441.941738241592 | 8838.834765 |

Table 1. X and Y values and Function Maximum of Each Method

|  |  |  |
| --- | --- | --- |
|  | % Error in X | % Error in Y |
| Newton | -1.041838e-14 | -6.083819e-14 |
| Secant | -3.557464e-10 | -1.062721e-09 |

Table 2. Percent Error of Each Method

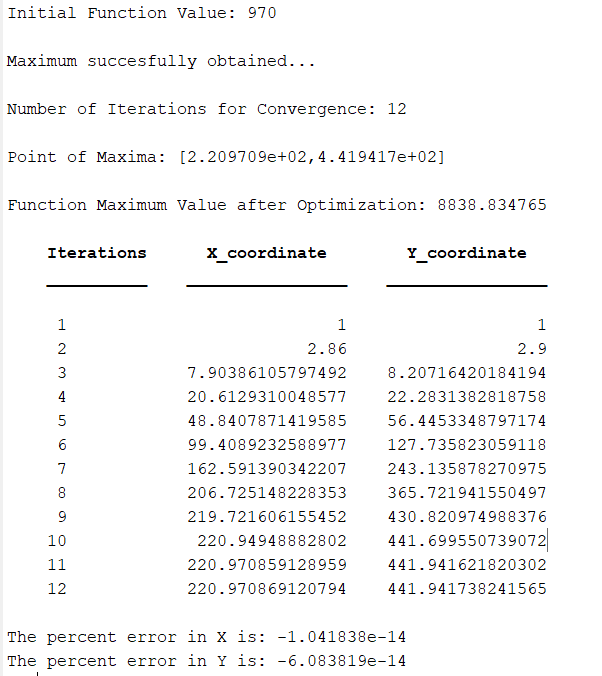
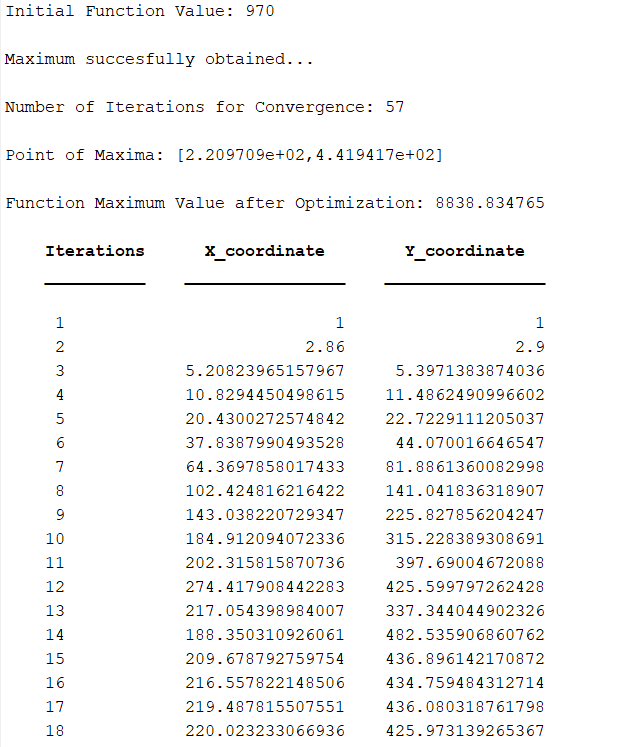
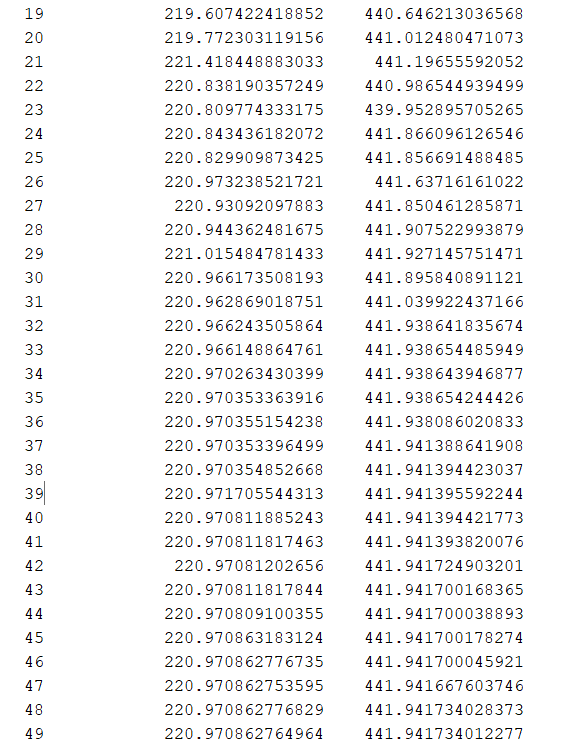
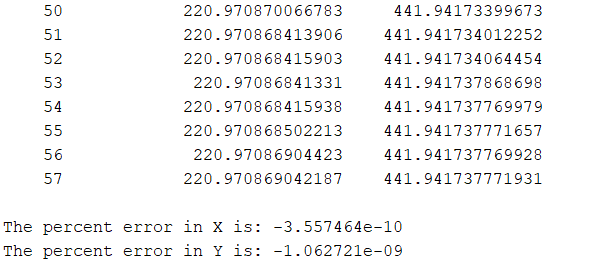


Figure 8. Newton’s Method Output

  
 Figure 9. Secant Method Output

**V. Conclusions**

As predicted, the number of iterations for convergence is less in Newton’s method. This is due to Newton’s method converging quadratically while secant method converges superlinearly. The error in Newton’s method is also less than that of Secant method. This may be due to the extra function calculations for the gradient taken in each loop, adding error in each calculation. This may vary by problem depending on the curve of the graph, the error in secant could be less than in newtons. Since we are calling each value of the gradient matrix per loop in secant method and the inverse Hessian can be multiplied directly with the gradient vector in the newton method, newton’s method actually comes out to less function calculations per iteration. This is only true in the multi parameter problem as in the one in the one variable problem f values in the secant can be stored thus resulting in less function calculations per iteration.The errors obtained are all negative because in this particular problem, we are predicting a maximum, so the predictions will naturally be less than the actual maxima. If we were predicting a minimum, the percent error would be positive. This is something that varies depending on the type of optimization done.

This method can be used for a multitude of optimization problems, including more variable parameters. One setback of newton’s method is that one must be able to take the second partial derivative of the model function, and this is often not plausible in real world scenarios. In secant method, the first order partial derivative would have to exist. One difficulty is that MatLab is unable to perform more involved symbolic formula calculations and thus would not be able to solve many model formulas of interest (including many tried before using the simpler example given). However by using this example, we can see what newton and secant methods are capable of in the field of optimization, and what might be possible using higher level programs.

**VI. Program Listing**

MatLab R2017b was used

**VII. References**

Wainwrig. 2017. *Using Calculus For Maximization Problems.* <<http://www.sfu.ca/~wainwrig/Econ331/notes-unconstrained-max.pdf>.> [Accessed 3 May 2018].

Cheney, Ward. (2013). *Numerical Mathematics and Computing.* India: Brooks/Cole, a part of Cengage Learning.