Newton's Method

This is a tutorial on Newton's Method. It only contains the material that I needed.

1 Introduction

Newton's method is a root-finding algorithm which produces approximation to the root of real-value function. It has a lot of variations solving the problems from the most basic version - single variable function to more complicated version such as multiple functions with multiple variables.

2 Newton Method

We start with the simple case of single variable function f defined for a real variable x, the function's derivative f' and a initial guess of the root x_0. Let r be the actual root of the function. Then, $r = x_0 + h$ where the number h evaluate the difference between the guess and the actual root. If we assume that h is small, we can use the linear approximation[1] to conclude that

$$0 = f(r) = f(x_0 + h) \approx f(x_0) + hf'(x_0) \tag{1}$$

Therefore, when $f'(x_0)$ is not equal to zero, we have

$$h = -\frac{f(x_0)}{f'(x_0)} \tag{2}$$

The "improved" estimate of the root x_1 is given by

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)} \tag{3}$$

Continuing the update step, we end up with a estimate that can be very close to the actual root. Thus, when x_n represent the current estimate, the next estimate x_{n+1} is given by

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{4}$$

2.1 Limitations

Newton's method is only guaranteed to converge if certain conditions are satisfied.

- **Stationary Point**: When the derivative of function at estimate x_n is zero, the next approximation will be infinite. Even if the derivative is small, the next estimate will still be very "bad".

[1]linear approximation is an approximation of a general function using a linear function. The value of function at point x is approximated as the value of the function at point a plus the difference between x and s times the derivative of function at point a that is $f(x) \approx f(a) + f'(a)(x - a)$.

- **cycle**: The estimate can alternate between the several values without converging to a root.
- **Derivative does not exist at root**: The algorithm overshoots the solution and lands on the other side of the root.
- **Discontinuous derivative**: If the derivative is not continuous at the root, then convergence may fail to occur in any neighborhood of the root.
- **Slow convergence**: It converges slow when the derivative at the root is equal to zero or the second derivative at the root does not exist.

3 Secant Method

Secant method is one of the most popular variation of Newton method. It can be thought as the finite difference[2] approximation of Newton method. The Secant method is defined by a recurrence relation

$$x_{n+1} = x_n - f(x_n)\frac{x_{n-1} - x_n}{f(x_{n-1}) - f(x_n)} \tag{5}$$

If x_{n-1} is close enough to x_n, $\frac{x_{n-1}-x_n}{f(x_{n-1})-f(x_n)}$ is close to $f'(x_n)$. It is more stable than Newton method and does not require the calculation of derivation which can sometimes be very problematic.

Comparing with Newton method, it converges slower. However Newton method requires the calculation of f and f' at each iteration, while Secant method requires only f. Thus, in some cases, Secant method may runs faster than Newton method.

4 Broyden's Method

Broyden's method is a generalization of the Secant method to more than one dimension. It is the most successful Secant method (also known as quasi-Newton method[3]) for solving systems of nonlinear equations. Broyden method considered the following problem

$$f : \mathbb{R}^n \to \mathbb{R}^m \tag{6}$$

In vanilla Newton method for solving the problem in equation (6) involved the Jacobian matrix[4] at each

[2]A finite difference is a mathematical expression of the form $f(x+b) - f(x+a)$. If a finite difference is divided by $b-a$, one gets a difference quotient. It is often used as an approximation of the derivative

$$f'(x) = \lim_{h\to 0}\frac{f(x+h) - f(x)}{h}$$

[3]Quasi-Newton methods are methods used to either find zeroes or local maxima and minima of functions, as an alternative to Newton's method. They can be used if the Jacobian or Hessian is unavailable or is too expensive to compute at every iteration

[4]the Jacobian matrix of a vector-valued function in several variables is the matrix of all its first-order partial derivatives.

$$(Df)_{ij} = \frac{\partial f_i}{\partial f_j}$$

.

iteration. The estimate is updated as

$$\vec{x}_{n+1} = \vec{x}_n - [Df(\vec{x}_n)]^{-1} f(\vec{x}_n) \tag{7}$$

However, the partial derivative matrix changes at each iteration and calculate the matrix can be expensive. In Broyden method, they construct an approximation to the matrix, which is updated at each iteration, so that it behaves similarly to the true Jacobian along the step. By replacing the partial derivative with the notion J and using the approximation in Secant method, we have

$$J_n \cdot (x_{n-1} - x_n) \approx f(x_{n-1}) - f(x_n) \tag{8}$$

Thus, the estimate in Broyden method, the estimate is updated as

$$\vec{x}_{n+1} = \vec{x}_n - J_n^{-1} \cdot f(\vec{x}_n) \tag{9}$$

Broyden suggests using the current estimate of the Jacobian matrix J_{n-1} and improving upon it by taking the solution to the secant equation that is a minimal modification to J_{n-1}. That is

$$\begin{aligned} &minimize_{J_n} ||J_n - J_{n-1}||^2 \\ &subject\ to\ J_n \cdot (x_{n-1} - x_n) = f(x_{n-1}) - f(x_n) \end{aligned} \tag{10}$$

The update is given by

$$J_n = J_{n-1} + \frac{\Delta f_n - J_{n-1}\Delta x_n}{||\Delta x_n||^2} \Delta x_n^T \tag{11}$$

where $\Delta f_n = f(x_{n-1}) - f(x_n)$ and $\Delta x_n = x_{n-1} - x_n$.

The algorithm is as follows:

1. Initial guess $\vec{x}_0$ and calculate f_0. Initial value of J_0^{-1}(Usually equals to 1).
2. Assuming the current estimate is $\vec{x}_n$ and value of function is $f(\vec{x}_n)$. The Jacobian matrix is J_n.
3. Update the estimate using equation (9).
4. Update the Jacobian matrix using equation (11).

In step 3, we need to calculate the inverse of J_n. However, we can instead update J_n^{-1} directly. The update is given by:

$$J_n^{-1} = J_{n-1}^{-1} + \frac{\Delta x_n - J_{n-1}^{-1}\Delta f_n}{||\Delta f_n||^2} \Delta f_n^T \tag{12}$$

When Δf_n is a scalar, the update of Jacobian matrix is given by

$$J_n^{-1} = \frac{\Delta x_n}{\Delta f_n} \tag{13}$$

5 Summary

- Newton Method

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{14}$$

- Secant Method

$$x_{n+1} = x_n - f(x_n)\frac{x_{n-1} - x_n}{f(x_{n-1}) - f(x_n)} \tag{15}$$

where $f'(x_n) \approx \frac{x_{n-1}-x_n}{f(x_{n-1})-f(x_n)}$

- Broyden Method

$$\vec{x}_{n+1} = \vec{x}_n - f(\vec{x}_n) \cdot J_n^{-1} \tag{16}$$

where $J_n^{-1} \approx \frac{x_{n-1}-x_n}{f(x_{n-1})-f(x_n)}$

$$J_n = J_{n-1} + \frac{\Delta f_n - J_{n-1}\Delta x_n}{||\Delta x_n||^2}\Delta x_n^T \tag{17}$$

or

$$J_n^{-1} = J_{n-1}^{-1} + \frac{\Delta x_n - J_{n-1}^{-1}\Delta f_n}{||\Delta f_n||^2}\Delta f_n^T \tag{18}$$

when Δf_n is a scalar

$$J_n^{-1} = \frac{\Delta x_n}{\Delta f_n} \tag{19}$$

where $\Delta f_n = f(x_{n-1}) - f(x_n)$ and $\Delta x_n = x_{n-1} - x_n$

6 Reference

[1] https://en.wikipedia.org/wiki/Newton%27s_method

[2] https://www.math.ubc.ca/~anstee/math104/newtonmethod.pdf

[3] https://en.wikipedia.org/wiki/Broyden%27s_method

[4] https://graphics.stanford.edu/courses/cs205a-13-fall/assets/lecture_slides/nonlinear_systems_ii.pdf