Newton's Method

Introduction

This paper will explain how to use Newton's method which is one of the tools used in engineering. The application of Newton's method is wide range, many technical books do not explain enough. The author thinks that there are roughly 2 scenes for the modeling and according to the model, it is necessary to select one or the other. This paper explains them as priority issues.

Newton's Method for 1 Variable

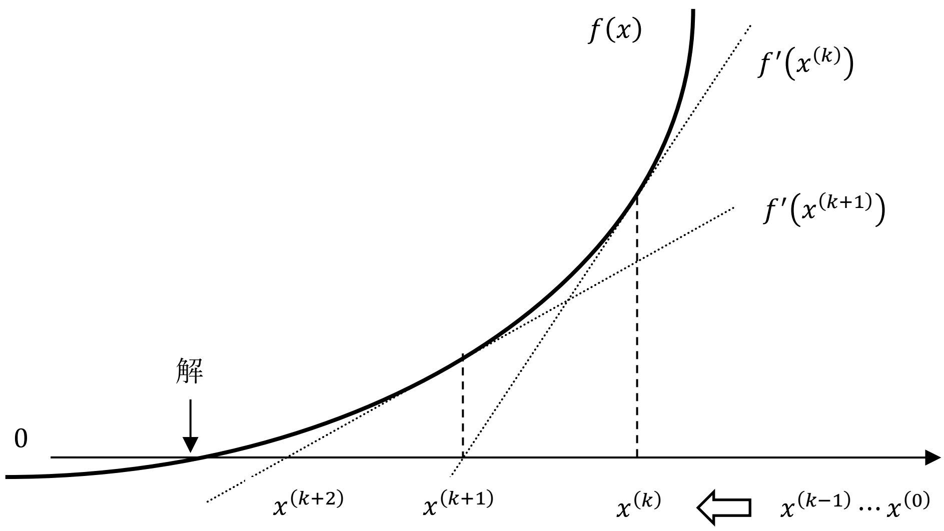
Newton's method for 1 variable is such as a method solving x with (1) .

$$f(x) = 0 \quad (1)$$

In detail, the method is an iteration method by recurrence formula (2), starting from the moderate initial value $x^{(0)}$.

$$x^{(k+1)} = x^{(k)} - \frac{f\left(x^{(k)}\right)}{f'\left(x^{(k)}\right)} \quad (2)$$

This seems to be **Fig.1**.



Figl.1 Newton's Method for 1 Variable

For the recurrence formula(2), This can be derived from tangent equation(3) at $x^{(k)}$,

$$y - f\left(x^{(k)}\right) = f'\left(x^{(k)}\right)\left(x - x^{(k)}\right) \quad (3)$$

Substitute $y = 0, x = x^{(k+1)}$ to (3), it becomes (4).

$$-f\left(x^{(k)}\right) = f'\left(x^{(k)}\right)\left(x^{(k+1)} - x^{(k)}\right) \quad (4)$$

Tranform (4), it becomes (2) definitely. Some technical books stay in above context. Actually, only this context will be almost useless. Of course, there are scenes handling (1) directly as the first scene. However, there are more useful scene, thar is, minimization (or maximization) for residual sum of squares. As the minimization and maximization are equivalent by inverting the sign, this paper describes “minimization” all together as below.

Now, the turn of second scene. First off, assume that function f is the derivative of some function(5).

$$f(x) = r'(x) \quad (5)$$

Then, (1) means the problem for solving an extreme value of function r. Besides, if r was a quadratic function, the extreme value’s existence, moreover its uniqueness could become certain. For the problem solving the coefficients of quadratic function y(6), generally, least squares method will be used for the problem for solving the approximate curve. Here, the residual sum of squares is defined as (7).

$$y = ax^2 + bx + c \quad (6)$$

$$r = \frac{1}{2}\sum_{n=1}^{k}\left(y_n - y(x_n)\right)^2 \quad (7)$$

k : Number of measurement data

(7)’s $\frac{1}{2}$ is generally a given coefficient for eliminating a coefficient after differential.

For coefficient a, differentiate (7), it becomes (8).

$$\frac{d}{da}r = 2\sum_{n=1}^{k}\left(y_n - y(x_n)\right)\frac{d}{da}\left(y_n - y(x_n)\right) \quad (8)$$

Therefore, it becomes (9) as Newton’s method.

$$f(a) = \frac{d}{da}r = r'(a) = 0 \quad (9)$$

For this example (6), and the other variable b, c are 3 variables. Then, it becomes the system of equations within the same way, of (8). Although Newton’s method for multivariable will be described later, the paper describes Newton’s method for 1 variable. If (6) was (10), Newton’s method for 1 variable could be written as (12).

$$y = c \quad (10)$$

$$\frac{d}{dc}r = 2\sum_{n=1}^{k}(y_n - c)\frac{d}{dc}(y_n - c) = 2\sum_{n=1}^{k}(c - y_n) \quad (11)$$

$$f(c) = \frac{d}{dc}r = 2\sum_{n=1}^{k}(c - y_n) = 0 \quad (12)$$

(12)'s solution definitely becomes an average(13).

$$c = \frac{1}{k}\sum_{n=1}^{k} y_n \quad (13)$$

The above example is the second scene.

Next, to explain Newton's method for multivariable, let me add somewhat. For now, put (14) , a recurrence formula of Newton's method becomes (15).

$$\varDelta x = x^{(k+1)} - x^{(k)} \quad (14)$$

$$f'\left(x^{(k)}\right)\varDelta x = -f\left(x^{(k)}\right) \quad (15)$$

Therefore, to handle as the residual error, substitute (5) to (15), it becomes (16).

$$r''\left(x^{(k)}\right)\varDelta x = -r'\left(x^{(k)}\right) \quad (16)$$

Newton's method for multivariable is (16)'s format. This comes from that the numerical calculation woks easily. See **Fig.2**, the second scene's image.

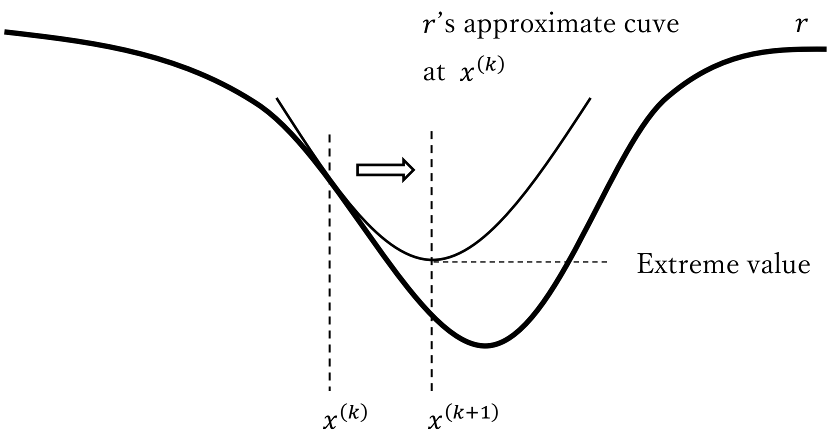


Fig.2 Newton's Method's Image for Minimizing Residual Error

Fig.2 can be told by Taylor expansion. First off, put (17), apply Taylor expansion to function r at $x^{(k)}$, it becomes (18). This is equivalent to the approximate curve shown in **Fig.2**.

$$\Delta x = x^{(k+1)} - x^{(k)} \quad (17)$$

$$r\left(x^{(k)} + \Delta x\right) = r\left(x^{(k)}\right) + r'\left(x^{(k)}\right)\Delta x + \frac{1}{2}r''\left(x^{(k)}\right)(\Delta x)^2 + \frac{1}{3!}r'''\left(x^{(k)}\right)(\Delta x)^3 \cdots \quad (18)$$

If $\Delta x \to 0$, the terms more than 3 order can be ignored. Then, differentiate by Δx, put (19) to get the extreme value.

$$r'\left(x^{(k)}\right) + r''\left(x^{(k)}\right)\Delta x = 0 \quad (19)$$

Transform (16), it will be recognized same as (16).

$$r''\left(x^{(k)}\right)\Delta x = -r'\left(x^{(k)}\right) \quad (16)$$

Up to now, the minimization for the residual error in the engineering is often shown, and is a useful technique. Newton's method will work well by binding this to Newton's method. As mentioned before, 2 order format has a certain existence of extreme value, and it is uniqueness. These are very useful characteristics to handle. In addition, in the case of overviewing the tiny range (domain range), any curve, whatever is, will be able to be absolutely approximated by a quadratic function. Then, in the engineering, this feeling is important. As a result, this paper says "Think the modeling to apply the tool."

On the other hand, it is not necessary for Newton's method that function(5) must be linear.

Handling non-linear function is possible, too. In addition, about the residual error, it is not absolutely necessary to be 2 order format. To tell the another, on the stage of differential(8), there are cases that the analytical solution can not be derived(non-differentiable function). But this can be processed by numerical differential. Actually, it seems to be forced to depend on the numerical differential for non-linear problems.

Newton's Method for Multivariable

Handling only 1 variable for Newton's method has poor worth to use. Here, this paper will explain Newton's method for 2 variables. By understanding Newton's method for 2 variables, it will reach to the recognition of Newton's method for multivariable as it goes on. First of all, before that, this paper explains total differential (tangent plane).

Total Differential (Tangent Plane)

Total differential is (20). In addition, generalized n dimentsion is (21).

$$df = \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy \quad (20)$$

$$df = \frac{\partial f}{\partial x_1} dx_1 + \frac{\partial f}{\partial x_2} dx_2 + \cdots + \frac{\partial f}{\partial x_n} dx_n \quad (21)$$

(20) is shown in **Fig.3**.

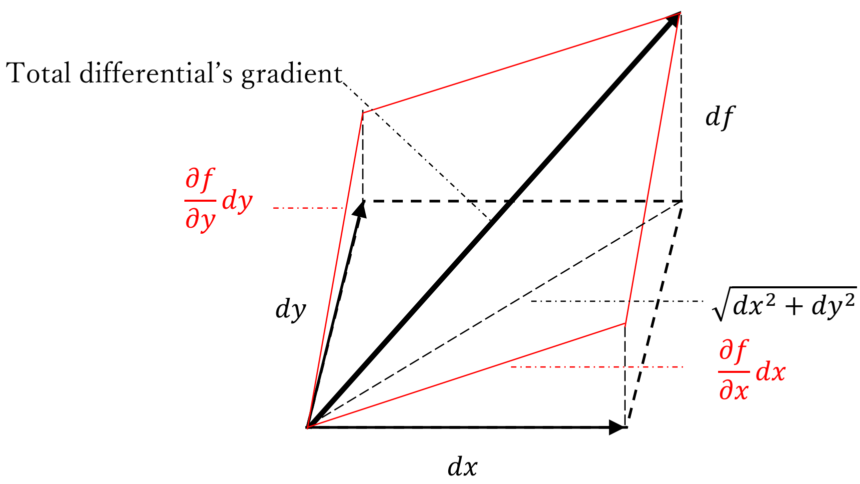


Fig.3 Total Differential Image

In the figure, the red line's parts show the tangent plane. At first, for the differential of total differential(20), originally, 2 variables' minute difference is expressed by (22). Then, the limit $\sqrt{dx^2 + dy^2} \to 0$ is total differential's definition.

$$\frac{\Delta f}{\sqrt{(\Delta x)^2 + (\Delta y)^2}} = \frac{f(x + \Delta x, y + \Delta y) - f(x, y)}{\sqrt{(\Delta x)^2 + (\Delta y)^2}} \quad (22)$$

Expand f like Taylor expansion for 1 variable, it becomes (23).

$$f(x + \Delta x, y + \Delta y) = f(x, y) + \frac{\partial f}{\partial x}\Delta x + \frac{\partial f}{\partial y}\Delta y +$$
$$\frac{1}{2}\frac{\partial^2 f}{\partial x^2}(\Delta x)^2 + \frac{1}{2}\frac{\partial^2 f}{\partial y^2}(\Delta y)^2 + \frac{\partial^2 f}{\partial x \partial y}\Delta x \Delta y +$$
$$\frac{1}{3!}\frac{\partial^3 f}{\partial x^3}(\Delta x)^3 + \frac{1}{3!}\frac{\partial^3 f}{\partial y^3}(\Delta y)^3 + \frac{1}{2}\frac{\partial^3 f}{\partial x^2 \partial y}(\Delta x)^2 \Delta y + \frac{1}{2}\frac{\partial^3 f}{\partial x \partial y^2}\Delta x(\Delta y)^2 + \cdots \quad (23)$$

Here, it is roughly categorized the terms in proportion to Δx and Δy, and high order terms in proportion to $(\Delta x)^2$, $(\Delta y)^2$, $\Delta x \Delta y$, $(\Delta x)^3$, $(\Delta y)^3 \cdots$. If $\Delta x \to 0, \Delta y \to 0$, the high order terms will converge rapidly than one order terms, and will be able to be ignored(24).

$$f(x + \Delta x, y + \Delta y) \fallingdotseq f(x, y) + \frac{\partial f}{\partial x}\Delta x + \frac{\partial f}{\partial y}\Delta y \quad (24)$$

Substitute this to (22), it becomes (25).

$$\frac{\Delta f}{\sqrt{(\Delta x)^2 + (\Delta y)^2}} = \frac{\frac{\partial f}{\partial x}\Delta x + \frac{\partial f}{\partial y}\Delta y}{\sqrt{(\Delta x)^2 + (\Delta y)^2}} \quad (25)$$

From the above, (20) has derived. (25) means that if only $\sqrt{dx^2 + dy^2}$ has been moved from any point, it will change only Δf. This must be an expression of the plane itself. Actually, tangent plane 's equation at the point(p, q) is (26), it will match total differential's format.

$$z - f(p, q) = \frac{\partial f(p, q)}{\partial x}(x - p) + \frac{\partial f(p, q)}{\partial y}(y - q) \quad (26)$$

Newton's Method for 2 Variables

Let me go ahead the discussion same as Newton's method for 1 variable. At first, Newton's method for 2 variables is a method for solving x_1, x_2 which implement (27) and (28).

$$f_1(x_1, x_2) = 0 \quad (27)$$

$$f_2(x_1, x_2) = 0 \quad (28)$$

At k times of the iteration, the tangent plane can be (29), (30) by applying to (26).

$$z - f_1\left(x_1^{(k)}, x_2^{(k)}\right) = \frac{\partial f_1\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_1}\left(x_1 - x_1^{(k)}\right) + \frac{\partial f_1\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_2}\left(x_2 - x_2^{(k)}\right) \quad (29)$$

$$z - f_2\left(x_1^{(k)}, x_2^{(k)}\right) = \frac{\partial f_2\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_1}\left(x_1 - x_1^{(k)}\right) + \frac{\partial f_2\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_2}\left(x_2 - x_2^{(k)}\right) \quad (30)$$

Like when 1 variable, put $z = 0$, $x_1 = x_1^{(k+1)}$, $x_2 = x_2^{(k+1)}$, it becomes (31), (32).

$$j_{11}\left(x_1^{(k+1)} - x_1^{(k)}\right) + j_{12}\left(x_2^{(k+1)} - x_2^{(k)}\right) = -f_1\left(x_1^{(k)}, x_2^{(k)}\right) \quad (31)$$

$$j_{21}\left(x_1^{(k+1)} - x_1^{(k)}\right) + j_{22}\left(x_2^{(k+1)} - x_2^{(k)}\right) = -f_2\left(x_1^{(k)}, x_2^{(k)}\right) \quad (32)$$

$$j_{11} = \frac{\partial f_1\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_1}, \quad j_{12} = \frac{\partial f_1\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_2}$$

$$j_{21} = \frac{\partial f_2\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_1}, \quad j_{22} = \frac{\partial f_2\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_2}$$

Put (33), (34), (35), express by matrix, it becomes (36).

$$\Delta x_1 = x_1^{(k+1)} - x_1^{(k)} \quad (33)$$

$$\Delta x_2 = x_2^{(k+1)} - x_2^{(k)} \quad (34)$$

$$X^{(k)} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \end{bmatrix} \quad (35)$$

$$\nabla F^{(k)} \Delta X = -F^{(k)} \quad (36)$$

$$\nabla F^{(k)} = \begin{bmatrix} j_{11} & j_{12} \\ j_{21} & j_{22} \end{bmatrix}, \quad \Delta X = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \end{bmatrix}, \quad F^{(k)} = \begin{bmatrix} f_1\left({X^{(k)}}^{\top}\right) \\ f_2\left({X^{(k)}}^{\top}\right) \end{bmatrix} = \begin{bmatrix} f_1\left(x_1^{(k)}, x_2^{(k)}\right) \\ f_2\left(x_1^{(k)}, x_2^{(k)}\right) \end{bmatrix}$$

(36) matches (4), (15) as 1 variable. $\nabla F^{(k)}$ means Jacobian matrix, this can be expanded to the multivariable as it is.

Now, assume f as r's first differential which means the minimization problem for residual sum of squares(37), (38).

$$\frac{\partial r(x_1, x_2)}{\partial x_1} = f_1(x_1, x_2) = 0 \quad (37)$$

$$\frac{\partial r(x_1, x_2)}{\partial x_2} = f_2(x_1, x_2) = 0 \quad (38)$$

Substitute this to (31), (32), it becomes (39), (40).

$$h_{11}\left(x_1^{(k+1)} - x_1^{(k)}\right) + h_{12}\left(x_2^{(k+1)} - x_2^{(k)}\right) = -\frac{\partial}{\partial x_1} r\left(x_1^{(k)}, x_2^{(k)}\right) \quad (39)$$

$$h_{21}\left(x_1^{(k+1)} - x_1^{(k)}\right) + h_{22}\left(x_2^{(k+1)} - x_2^{(k)}\right) = -\frac{\partial}{\partial x_2} r\left(x_1^{(k)}, x_2^{(k)}\right) \quad (40)$$

$$h_{11} = \frac{\partial^2 r\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial {x_1}^2}, \quad h_{12} = \frac{\partial^2 r\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_1 \partial x_2}$$

$$h_{21} = \frac{\partial^2 r\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_2 \partial x_1}, \quad h_{22} = \frac{\partial^2 r\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial {x_2}^2}$$

This is expressed by matrix(41).

$$H^{(k)} \Delta X = -\nabla r^{(k)} \quad (41)$$

$$H^{(k)} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}, \quad F^{(k)} = \begin{bmatrix} f_1\left(x_1^{(k)}, x_2^{(k)}\right) \\ f_2\left(x_1^{(k)}, x_2^{(k)}\right) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial r\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_1} \\ \dfrac{\partial r\left(x_1^{(k)}, x_2^{(k)}\right)}{\partial x_2} \end{bmatrix} = \nabla r^{(k)}$$

(41) matches (16) as 1 variable. $H^{(k)}$ means Hessian matrix(42), this can be expanded to the multivariable as it is.

$$H = \begin{bmatrix} \dfrac{\partial^2 r}{\partial {x_1}^2} & \dfrac{\partial^2 r}{\partial x_1 \partial x_2} & \cdots & \dfrac{\partial^2 r}{\partial x_1 \partial x_n} \\ \dfrac{\partial^2 r}{\partial x_2 \partial x_1} & \dfrac{\partial^2 r}{\partial {x_2}^2} & \cdots & \dfrac{\partial^2 r}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \dfrac{\partial^2 r}{\partial x_n \partial x_1} & \dfrac{\partial^2 r}{\partial x_n \partial x_2} & \cdots & \dfrac{\partial^2 r}{\partial {x_n}^2} \end{bmatrix} \quad (42)$$

As mathematics, if the function was continuous, the differential sequence could be exchanged, so Hessian matrix becomes symmetric matrix. For the numerical calculation, calculating only one side of off-diagonals is OK.

Deriving (41) from r can be possible. Apply Taylor expansion to r as n variables, it becomes (43)

$$r(x_1 + \Delta x_1, x_2 + \Delta x_2, \cdots, x_n + \Delta x_n) \\ = r(x_1, x_2, \cdots, x_n) + \sum_{i=1}^{n} \frac{\partial r}{\partial x_i} \Delta x_i + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\partial^2 r}{\partial x_i \partial x_j} \Delta x_i \Delta x_j + \cdots \quad (43)$$

This is limited to second order terms. Discard 3 order or more like 1 variable, differentiate partially by Δx_i, extreme values can be given (44).

$$\frac{\partial r}{\partial x_i} + \sum_{j=1}^{n} \frac{\partial^2 r}{\partial x_i \partial x_j} \Delta x_j = 0$$

$$\sum_{j=1}^{n} \frac{\partial^2 r}{\partial x_i \partial x_j} \Delta x_j = -\frac{\partial r}{\partial x_i} \quad (44)$$

$i = 1,2,\cdots,n$: System of equations by n formulas

(44) shows each row of (41), it becomes clear thar they are equivalent.

Up to now, the author thinks the above content must be the general format, the practical and the most useful application for Newton's method. In addition, Gaussian elimination is general for ΔX solution of (41). But in the case solving a huge matrix, the iteration method like SDM(steepest descent method) or CGM(conjugate gradient method) will fit. But in the case like the problem solving the coefficients of approximate formula, the variables are few and it will never be the huge matrix such as hundreds, thousands, or more. In addition, using an iteration method in the iteration method is not practical(for calculating cost and precision). In such cases, it is better to consider the model to apply SDM or CGM directly.

Gradient Method for Multivariable

For the minimization problem like the residual sum of squares, this is a method which reaches the solution by following the gradient such as **Fig.2** (see **Fig.4**).

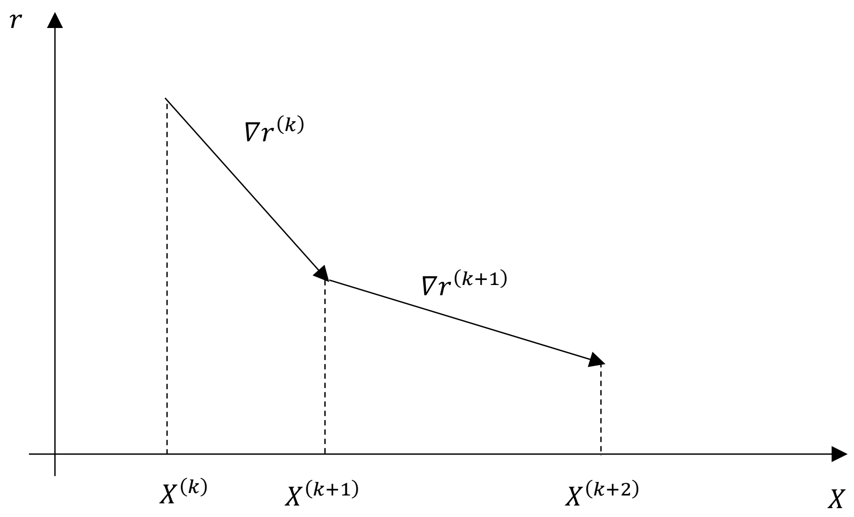


Fig.4 Gradient Method's Search

Now, there is a method to search roughly with multivariable by using 1 variable gradient method[1]. Assume that Newton's method(2) is assigned to 1 variable gradient method, install a coefficient γ(45).

$$x^{(k+1)} = x^{(k)} - \frac{f\left(x^{(k)}\right)}{\gamma f'(x^{(k)})} \quad (45)$$

$$\gamma > 1$$

The idea of the coefficient is that following the gradient does not run over. This is expressed as r(46).

$$x^{(k+1)} = x^{(k)} - \frac{r'\left(x^{(k)}\right)}{\gamma r''(x^{(k)})} \quad (46)$$

Expand to the multivariable as it is, it becomes (47).

$$\begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ \vdots \\ x_n^{(k+1)} \end{bmatrix} = \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix} - \frac{1}{\gamma} \begin{bmatrix} \frac{1}{\frac{\partial^2 r^{(k)}}{\partial {x_1}^2}} & 0 & \square & \cdots & 0 \\ 0 & \frac{1}{\frac{\partial^2 r^{(k)}}{\partial {x_2}^2}} & 0 & \cdots & 0 \\ \square & 0 & \square & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & \frac{1}{\frac{\partial^2 r^{(k)}}{\partial {x_n}^2}} \end{bmatrix} \begin{bmatrix} \frac{\partial r^{(k)}}{\partial x_1} \\ \frac{\partial r^{(k)}}{\partial x_2} \\ \vdots \\ \frac{\partial r^{(k)}}{\partial x_n} \end{bmatrix}$$

$$= \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ \vdots \\ x_n^{(k)} \end{bmatrix} - \frac{1}{\gamma} \begin{bmatrix} \frac{\partial^2 r^{(k)}}{\partial {x_1}^2} & 0 & \square & \cdots & 0 \\ 0 & \frac{\partial^2 r^{(k)}}{\partial {x_2}^2} & 0 & \cdots & 0 \\ \square & 0 & \square & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & \frac{\partial^2 r^{(k)}}{\partial {x_n}^2} \end{bmatrix}^{-1} \begin{bmatrix} \frac{\partial r^{(k)}}{\partial x_1} \\ \frac{\partial r^{(k)}}{\partial x_2} \\ \vdots \\ \frac{\partial r^{(k)}}{\partial x_n} \end{bmatrix} \quad (47)$$

This method can sometimes search the gradient. (47) by the matrix expression is (48).

$$X^{(k+1)} = X^{(k)} - \frac{1}{\gamma} D[H]^{-1} \nabla r \quad (48)$$

$D[\cdot]$：Operator which extracts the diagonal components

Transform (48), it becomes (49).

$$\gamma\left(X^{(k+1)} - X^{(k)}\right) = -D[H]^{-1} \nabla r$$

$$\gamma D[H] \Delta X = -\nabla r \quad (49)$$

(49) is a same format as Newton's method for multivariable(41). It is explained that the method ignores off-diagonal components, and follows the gradient by only diagonal components. Generally, this method is not used, it is the preparation for LM method as the following.

LM (Levenberg-Marquardt) Method

For the method without solving an inverse matrix, the stage when the energy is focused on the diagonal components, that is "When all off-diagonal components becomes 0(zero), it means a solution as it is ." The matrix like a upper triangular or a lower triangular follows this. And there is the method which adjusts the convergence speed by adjusting the diagonal components' power for the iteration method. In addition, because the off-diagonal components mean the correlations between the variables, to increase the power of diagonal components means trying to search the gradient except their correlations. And the other is right, too.

Now, LM method[1], what is, it is the combination method(50) by Newton's method for multivariable(41) and (49).

$$(H + \gamma D[H])\Delta X = -\nabla r \quad (50)$$

One more point, at (50), take care that γ's specified range has no limit. About γ 's concern, γ which is directly the multiplier to the diagonal components may be good for programming. Generally, By adjusting the diagonal components' power, the convergence and its speed will be able to be adjusted.

How to Specify Initial Value

There is not a definite pattern to specify the initial value. Basically, it depends on putting the initial value how nearly to the solution. No matter how Newton's method converges with quadratic convergence, where the location far from the solution is(expressed in another way, it can not be approximate curve by second order around the search point), stagnation makes sense. According to the target, it is necessary to improve. In the other addition, it is important to know the possible range(domain range) for the variable in advance. If the domain range was known, it could be thought that by specifying the resolution according to the precision of the solution's goal, the limit can be applied to the search range. Moreover, in the case for out of the domain range, it might be handled as search fail(program's halt), or the application which puts another initial value is possible, too.

Numerical Differential's Step Size

Newton's method needs the first and second differential. As of the modeling, although the differentiable function is OK, many applications will use the numerical differential. If the case of 1 variable's function, the function r must be sampled, as shown in **Fig.5**. In **Fig.5**, it shows sampling points for 1 variable differential.

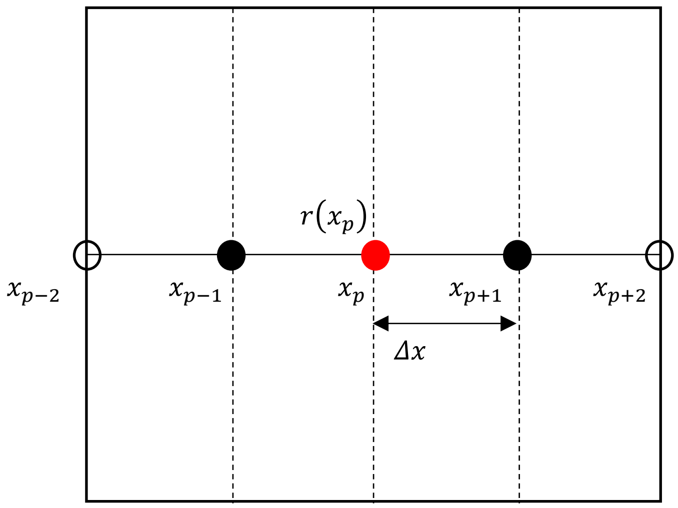


Fig.5 Differential Sampling Points(1Variable)

The first differential, second differential becomes (51), (52) respectively, the direction y is same, too.

$$\frac{\partial}{\partial x} r(x_p) = \frac{r(x_{p+1}) - r(x_{p-1})}{2\Delta x} \quad (51)$$

$$\frac{\partial^2}{\partial x^2} r(x_p) = \frac{r'(x_{p+1}) - r'(x_{p-1})}{2\Delta x} = \frac{\frac{r(x_{p+2}) - r(x_p)}{2\Delta x} - \frac{r(x_p) - r(x_{p-2})}{2\Delta x}}{2\Delta x} \quad (52)$$

Next, for the second differential for 2 variables, to differentiate the direction x first is **Fig.6**.

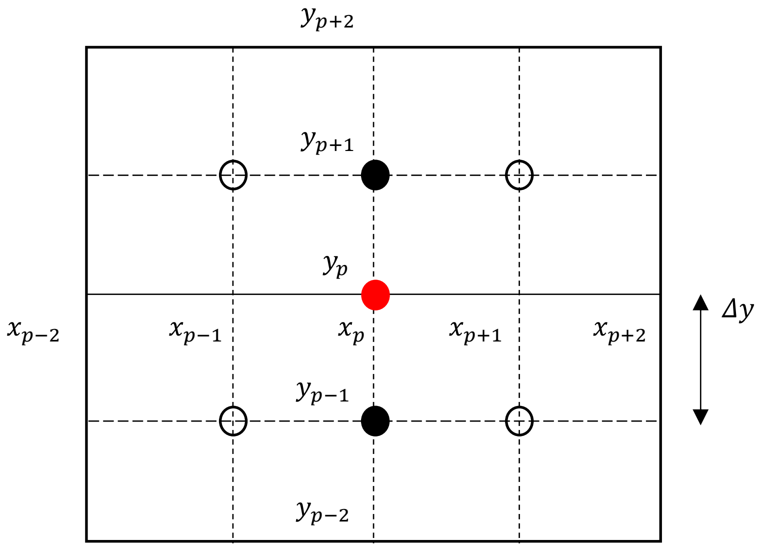


Fig.6 Second Differential(2 Values)

2 values' formula is (53).

$$\frac{\partial}{\partial y}\frac{\partial}{\partial x}r(x_{p,}y_p) = \frac{\frac{r(x_{p+1}, y_{p+1}) - r(x_{p-1}, y_{p+1})}{2\Delta x} - \frac{r(x_{p+1}, y_{p-1}) - r(x_{p-1}, y_{p-1})}{2\Delta x}}{2\Delta y} \quad (53)$$

As above, this is the most intelligent way to take the points for the numerical differential. On the other hand, there is a way called upwind difference shown in the simulation such as CIP method, but this paper will omit. The case which differentiates the direction y, the sampling points will be changed. If Hessian matrix was handled as symmetric matrix, one side of off-diagonal components are enough, but, both sides can be calculated. Although the author's understanding, it seems that they have a goodness respectively.

＊In the case as symmetric matrix, the iteration method will converge more easily. In addition, the calculation cost will be reduced.

＊In the case as off-diagonal, the gradient can be evaluated strictly. Especially, the correlation between variables can be reflected strictly. However, in the case that there is too large difference between off-diagonal components, it seems not to be a good condition problem, originally. Because this has a relation to the convergence, originally the method sometimes does not work well.

Now, the problem is “How to specify the amount of the numerical differential’s step size?” Originally, differential means the gradient by the limit for the tiny difference, →0(zero). Therefore, the value is desired as small as possible. As it is, however, it does not become “Machine epsilon which expresses the minimum value in the computer may be good. The machine epsilon may not be able to cause a significant amount of change. On the other, the precision loss at the numerical calculation becomes a problem. For the numerical differential’s sampling by the previous description. There is a guideline as below.

＊It is good to identify the place around 1/3 of calculation digits{2}

Calculation digits means significant digits. IEEE754 double precision’s mantissa has 52bit, so, the significant digits in decimal is about 15 digits(54).

$$52 \times \log 2 \fallingdotseq 15.65 \quad (54)$$

Of course, the calculation will be executed more than this digits by FPU buit-in CPU, they have a question for the evaluation base on the calculation error although, apply 1/3 according to the guideline, the numerical differential’s step size becomes 5 digits(55).

$$15.7 \times \frac{1}{3} \fallingdotseq 5 \quad (55)$$

For example, the residual sum of squares r is supposed to be calculated based on the measurement data’s intensity as below(56).

$$r \approx 500 = 5 \times 10^2 \quad (56)$$

Then, it becomes (57) as the numerical differential’s step size.

$$\mathit{\Delta} = 5 \times 10^{2-5} = 5 \times 10^{-3} \quad (57)$$

According to the author’s experience, this is only a rough indication. Actually, there are various behaviors, the problems as below has been faced.

① Is same step size OK between the first differential and the second differential?

② While processing to repeat the iteration, in the case that r ‘s intensity will change, should the step size be constant?

③ About the multivariable, how to handle the difference between variables for there domain? Actually, the iteration method may diverge because of the intensity difference between the variables.

④ As the step size is bigger, it sometimes converges without problems. In addition, sometime converges rapidly, too.

For the leading①, the first and second differential should be evaluated by same difference. More than, it is a paradox, aren't there any reasons to do it? At ③, a detail example(refer to "Newton's Method Example") given is, the curve fitting for the function s(58)(see **Fig.7**).

$$s(x) = ax^2e^{-bx} \quad (58)$$

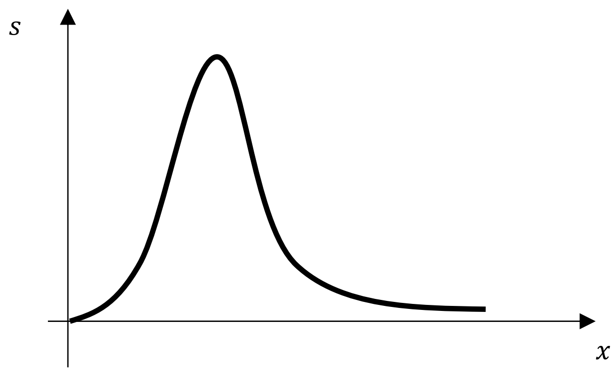


Fig.7 Approximate Function

Here, treat r as the minimization problem of the residual sum of squares(59).

$$r(a,b) = \frac{1}{2}\sum_{n=1}^{k}(s(x_n) - s_n)^2 \quad (59)$$

k: Measurement data's number

Because r is differentiable, calculating formula will be derived by getting the first differential, second differential. Be pardoned as the example to explain, here. However the theoretical formula is differentiable, if the function is complicated, it is convenient to use the formulated numerical differential, and it leads to the avoidance for program's bugs

For (58), as compared to a, it is clear from the formula(57) that exponent b impacts r by a tiny difference. In this case, it is often the case that the solution oscillates between the variables, and the convergence will be impossible. This comes from the fact that the one side's variable change will cause the another's huge variable change. As a result, the author always adjusts each variable individually. In addition, for the above example, to evaluate the residual error taken by the logarithm is possible. In detail, change s to (60), it becomes (61).

$$s(x) = \log a + 2\log x - bx \quad (60)$$

$$r(c,b) = \frac{1}{2}\sum_{n=1}^{k}(s(x_n) - g_n)^2 \quad (61)$$

$$c = \log a$$

After getting the solution of c, a can be derived. If this is so, the change caused by the step size can be improved. However, the method which takes the logarithm has a demerit. In the case for the measurement data with low SN, the signal component's weight becomes small, in the other hand noise's weight becomes bigger. As a result, the fitting's precision itself will become poor. As the way about this is not nothing, this is the point that the numerical calculation does not go easily. About ④, if the target has a fine characteristic(The approximate precision by 2 order curve is good → see **Fig.2**), it is sometimes no problem that the step size is so big. However, be careful about the less improvement for the solution's precision(significant digits) itself around the solution's neighborhood.

As above, it is only one case, including ②, the author by oneself does not have on hand to any case solution. As a result, to do trial and error, and to improve will be necessary.

References

[1] Kenichi Kanatani. [Agree to Understand, Optimization Mathematics]
Korenarawakaru saitekika sugaku
Kyoritsu Shuppan. 2005.

[2] Masao Iri, Yoritake Fujino. [Numerical Calculation's Common Sense]
Suchikeisan no jyosiki.
Kyoritsu Shuppan. 1985.