Chapter III

Unconstrained Univariate Optimization

- Introduction
- Interval Elimination Methods
- Polynomial Approximation Methods
- Newton’s Method
- Quasi-Newton Methods
1 Introduction

Univariate optimization means optimization of a scalar function of a single variable:

\[ y = P(x) \]

These optimization methods are important for a variety of reasons:

1. there are many instances in engineering when we want to find the optimum of functions such as these (e.g. optimum reactor temperature, etc.),
2. almost all multivariable optimization methods in commercial use today contain a line search step in their algorithm,
3. they are easy to illustrate and many of the fundamental ideas are directly carried over to multivariable optimization.

Of course these discussions will be limited to nonlinear functions, but before discussing optimization methods we need some mathematical background.
Continuity:

In this course we will limit our discussions to continuous functions.

Functions can be continuous but their derivatives may not be.
A function $P(x)$ is continuous at a point $x_0$ iff:

$$P(x_0) \text{ exists and } \lim_{x \to x_0^-} P(x) = \lim_{x \to x_0^+} P(x) = P(x_0)$$

Some discontinuous functions include:

→ price of processing equipment,

→ manpower costs,

→ measurements, ... 

Many of the functions we deal with in engineering are either continuous or can be approximated as continuous functions. However, there is a growing interest in solving optimization problems with discontinuities.
Convexity of Functions

In most optimization problems we require functions which are either convex or concave over some region (preferably globally):

i) for a convex function and \( \forall \alpha \in [0, 1] \),
\[
P(\alpha x + (1 - \alpha)y) \leq \alpha P(x) + (1 - \alpha)P(y)
\]

ii) for a concave function and \( \forall \alpha \in [0, 1] \),
\[
P(\alpha x + (1 - \alpha)y) \geq \alpha P(x) + (1 - \alpha)P(y)
\]
Convexity is a general idea used in optimization theory which when applied to a function, describes the property of having an optimum. Convexity combines both stationarity and curvature into a single concept. Often both convex and concave functions are described as exhibiting the property of convexity.

These conditions are very inconvenient for testing a specific function, so we tend to test the derivatives of the function at selected points. First derivatives give us a measure of the rate of change of the function. Second derivatives give us a measure of curvature or the rate of change of the second derivatives.
Notice for a convex function:

The first derivatives increase (become more positive) as $x$ increases. Thus, in the region we are discussing, the second derivative is positive. Since the first derivative starts out negative and becomes positive in the region of interest, there is some point where the function no longer decreases in value as $x$ increases and this point is the function minimum.
This is another way of stating that the point \( x^* \) is a (local) minimum of the function \( P(x) \) iff:

\[
P(x^*) \leq P(x), \quad \forall x \in [a, b]
\]

If this can be stated as a strict inequality \( x^* \) is said to be a unique (local) minimum.

A similar line of argument can be followed for concave functions, with the exception that: the first derivatives decrease (become more negative) with increasing \( x \), the second derivative is negative and as result, the optimum is a maximum.

The problem with these ideas is that we can only calculate the derivatives at a point. So looking for an optimum using these fundamental ideas would be infeasible since a very large number points (theoretically all of the points) in the region of interest must be checked.
Necessary and Sufficient Conditions for an Optimum

Recall that for a twice continuously differentiable function $P(x)$, the point $x^*$ is an optimum iff:

\[
\frac{dP}{dx} \bigg|_{x^*} = 0 \quad \text{stationarity}
\]

and

\[
\frac{d^2 P}{dx^2} \bigg|_{x^*} > 0 \quad \text{a minimum}
\]

\[
\frac{d^2 P}{dx^2} \bigg|_{x^*} < 0 \quad \text{a maximum}
\]

What happens when the second derivative is zero?
Consider the function:

\[ y = x^4 \]

which we know to have a minimum at \( x = 0 \).

at the point \( x = 0 \):

\[
\frac{dy}{dx}\bigg|_0 = 4x^3 = 0, \quad \frac{d^2y}{dx^2}\bigg|_0 = 12x^2 = 0, \quad \frac{d^3y}{dx^3}\bigg|_0 = 24x = 0, \quad \frac{d^4y}{dx^4}\bigg|_0 = 24 > 0
\]
We need to add something to our optimality conditions. If at the stationary point $x^*$ the second derivative is zero we must check the higher-order derivatives. Thus, for the first non-zero derivative is odd, that is:

$$\frac{d^n P}{dx^n} \bigg|_{x^*} \neq 0 \quad \text{where} \quad n \in \{3, 5, 7, \ldots\}$$

Then $x^*$ is an inflection point. Otherwise, if the first higher-order, non-zero derivative is even:

$$\frac{d^n P}{dx^n} \bigg|_{x^*} > 0 \quad \text{a minimum}$$

$$\frac{d^n P}{dx^n} \bigg|_{x^*} < 0 \quad \text{a maximum}$$
Some Examples

1. \[ \min_x ax^2 + bx + c = 0 \]

    - stationarity:
      \[ \frac{dP}{dx} = 2ax + b \rightarrow x^* = -\frac{b}{2a} \]
    - curvature:
      \[ \frac{d^2P}{dx^2} = 2a \rightarrow \begin{cases} a > 0 & \text{maximum} \\ a < 0 & \text{minimum} \end{cases} \]
2. \[
\min_x x^3 - 2x^2 - 5x + 6
\]

stationarity:
\[
\frac{dP}{dx} = 3x^2 - 4x - 5 = 0 \rightarrow x^* = \frac{2 \pm \sqrt{19}}{3}
\]

curvature:
\[
\left. \frac{d^2P}{dx^2} \right|_{x^*} = 6x - 4 = \begin{cases} 
2\sqrt{19} > 0 & \text{minimum} \\
-2\sqrt{19} < 0 & \text{maximum}
\end{cases}
\]

3. \[
\min(x - 1)^n
\]
4. A more realistic example. Consider the situation where we have two identical settling ponds in a waste water treatment plant:

These are normally operated at a constant throughput flow of $F$, with an incoming concentration of contaminants $c_i$. The system is designed so that at nominal operating conditions the outgoing concentration $c_o$ is well below the required levels. However, we know that twice a day (7 am and 7 pm) the incoming concentration of contaminants to this section of the plant dramatically increase above these nominal levels. What we would like to determine is whether the peak outgoing levels will remain within acceptable limits.
What assumptions can we make?

1) well mixed,
2) constant density,
3) constant flowrate.

contaminant balance:

\[
\frac{d\rho V c_o}{dt} = F c_1 - F c_o \quad \text{(second pond)}
\]

\[
\frac{d\rho V c_1}{dt} = F c_i - F c_1 \quad \text{(first pond)}
\]

Let the residence time be \( \tau = \frac{\rho V}{F} \). Then the balances become

\[
\tau \frac{dc_o}{dt} = c_1 - c_o \quad \text{(second pond)}
\]

\[
\tau \frac{dc_1}{dt} = c_i - c_1 \quad \text{(first pond)}
\]
Since we wish to know how changes in $c_i$ will affect the outlet concentration $c_o$. Define perturbation variables around the nominal operating point and take Laplace Transforms:

\[ \tau s C_o(s) = C_1(s) - C_o(s) \quad \text{(second pond)} \]
\[ \tau s C_1(s) = C_i(s) - C_1(s) \quad \text{(first pond)} \]

Combining and simplifying into a single transfer function:

\[ C_o(s) = \frac{1}{(\tau s + 1)^2} C_i(s) \]

If we represent the concentration disturbances as ideal impulses:

\[ C_i(s) = k \]

Then, transfer function of the outlet concentration becomes:

\[ C_o(s) = \frac{k}{(\tau s + 1)^2} \]
In the time-domain this is:

\[ c_o(t) = kte^{-\frac{t}{\tau}} \]

Since we want to determine:

\[ \max_t c_o(t) = kte^{-\frac{t}{\tau}} \]
stationarity:

\[
\frac{dc_o}{dt} = ke^{-\frac{t}{\tau}} \left( 1 - \frac{t}{\tau} \right) = 0 \implies 1 - \frac{t}{\tau} = 0 \implies t = \tau
\]

curvature:

\[
\frac{d^2c_o}{dt^2} = \frac{k}{\tau} \left( \frac{t}{\tau} - 2 \right) e^{-\frac{t}{\tau}}|_{t=\tau} = -\frac{k}{\tau e} < 0 \implies \text{maximum}(k > 0)
\]

Finally, the maximum outlet concentration is:

\[
c_o(\tau) = k\tau e^{-1} \simeq 0.368 \frac{k\rho V}{F}
\]
This example serves to highlight some difficulties with the analytical approach. These include:

→ determining when the first derivative goes to zero. This required the solution of a nonlinear equation which is often as difficult as the solution of the original optimization problem.

→ computation of the appropriate higher-order derivatives.

→ evaluation of the higher-order derivatives at the appropriate points.

We will need more efficient methods for optimization of complex nonlinear functions. Existing numerical methods can be classified into two broad types:

1. those that rely solely on the evaluation of the objective function,
2. those that use derivatives (or approximations of the derivatives) of the objective function.
2 Interval Elimination Methods

These methods work by systematically reducing the size of the interval within which the optimum is believed to exist. Since we are only ever working with an interval, we can never find the optimum exactly. There are many interval elimination algorithms; however, they all consist of the same steps:

1. determine an interval in which the optimum is believed to exist,
2. reduce the size of the interval,
3. check for convergence,
4. repeat steps 2 & 3 until converged.

We will examine only one interval elimination method, called the "Golden Section Search".
Scanning & bracketing

As an example, consider that our objective is to minimize a function $P(x)$. In order to start the optimization procedure we need to determine the upper and lower bounds for an interval in which we believe an optimum will exist. These could be established by:

- physical limits,
- scanning,
  - choosing a starting point,
  - checking the function initially decreases,
  - finding another point of higher value.
Golden Section Search

Consider that we want to break up the interval, in which we believe the optimum exists, into two regions:

\[
\begin{align*}
L &= l_1 + l_2 \\
&= rL + r^2L
\end{align*}
\]

or

\[
r^2 + r - 1 = 0
\]
There are two solutions for this quadratic equation. We are only interested in the positive root:

\[ r = \frac{-1 + \sqrt{5}}{2} \approx 0.61803398 \ldots \quad \text{Golden ratio} \]

Which has the interesting property:

\[ r = \frac{1}{r + 1} \]

We will use this Golden ratio to break up our interval as follows:
By breaking up the interval in this way we will be able to limit ourselves to one simple calculation of an $x_i$ and one objective function evaluation per iteration.

The optimization algorithm is:

1. using the interval limits $a_k$ and $b_k$, determine $x_{1}^{k}$ and $x_{2}^{k}$,
2. evaluate $P(x_{1}^{k})$ and $P(x_{2}^{k})$,
3. eliminate the part of the interval in which the optimum is not located,
4. repeat steps 1 through 3 until the desired accuracy is obtained.

How can we decide which part of the interval to eliminate?
There are only three possible situations. For the minimization case, consider:

- $P(x^i) > P(x^k)$: eliminate the interval to the left of $x^k$. For a convex function, all values to the right of $x^k$ are less than those on the left.

- $P(x^i) < P(x^k)$: eliminate the interval to the right of $x^k$. For a convex function, all values to the left of $x^k$ are less than those on the right.

- $P(x^i) = P(x^k)$: we know that the optimum is between $x^k$ and $x^k$, so we could eliminate the intervals to the left of $x^k$ and to the right of $x^k$. In practice, we eliminate only one region for computational ease.
Graphically the algorithm is:

Three things worth noting:

1. only one new point needs to be evaluated during each iteration,
2. the length of the remaining interval after $k$ iterations (or accuracy to which you know the optimum value of $x$) is:

$$L^k = r^k L^0 \simeq (0.618034)^k L^0$$

3. the new point for the current iteration is:

$$x_1^k = a^k + b^k - x_2^k$$

or:

$$x_2^k = a^k + b^k - x_1^k$$
Example

Find the solution to within 5% for

$$\min_{x} x^2 - 2x + 1$$

Guess an initial interval for the optimum as $0 \leq x^* \leq 2$.

<table>
<thead>
<tr>
<th>$k$</th>
<th>$a^k$</th>
<th>$b^k$</th>
<th>$x_1^k$</th>
<th>$x_2^k$</th>
<th>$P(x_1^k)$</th>
<th>$P(x_2^k)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0000</td>
<td>2.0000</td>
<td>0.7640</td>
<td>1.2360</td>
<td>5.5696E-02</td>
<td>5.5696E-02</td>
</tr>
<tr>
<td>1</td>
<td>0.7640</td>
<td>2.0000</td>
<td>1.2360</td>
<td>1.5280</td>
<td>5.5696E-02</td>
<td>2.7878E-01</td>
</tr>
<tr>
<td>2</td>
<td>0.7640</td>
<td>1.5280</td>
<td>1.0558</td>
<td>1.2360</td>
<td>3.1158E-03</td>
<td>5.5696E-02</td>
</tr>
<tr>
<td>3</td>
<td>0.7640</td>
<td>1.2360</td>
<td>0.9443</td>
<td>1.0558</td>
<td>3.1025E-03</td>
<td>3.1158E-03</td>
</tr>
<tr>
<td>4</td>
<td>0.7640</td>
<td>1.0558</td>
<td>0.8755</td>
<td>0.9443</td>
<td>1.5507E-02</td>
<td>3.1025E-03</td>
</tr>
<tr>
<td>5</td>
<td>0.8755</td>
<td>1.0558</td>
<td>0.9443</td>
<td>0.9869</td>
<td>3.1025E-03</td>
<td>1.7130E-04</td>
</tr>
</tbody>
</table>
Note that the number of calculation we have performed is:

- \((k + 2) = 8\) evaluations of \(P(x_i)\),
- \((k + 2) = 8\) determinations of a new point \(x_i\).

There are a variety of other interval elimination methods including Interval Bisection, Fibonacci Searches, and so forth. They are all simple to implement and not computationally intensive. They tend to be fairly robust and somewhat slow to converge.
3 Polynomial Approximation Methods

There are a number of polynomial approximation methods, which differ only in the choice of polynomial used to locally approximate the function to be optimized. Polynomials are attractive since they are easily fit to data and the optima of low-order polynomials are simple to calculate. We will only discuss Successive Quadratic Approximation as it illustrates the basic ideas and will carry over into multivariable optimization techniques.

Successive Quadratic Approximation

Suppose we have evaluated some test points of a function as follows:
We could approximate the objective function by fitting a quadratic through the test points.

The main advantage of using a quadratic approximation is that we know that optimum is:

\[ x^* = -\frac{b}{2a} \]
The first step is to calculate the approximation constants \((a, b, c)\) in the quadratic from the available points \((x_1, x_2, x_3)\). Using these three evaluation points we can write:

\[
P(x_1) = ax_1^2 + bx_1 + c
\]
\[
P(x_2) = ax_2^2 + bx_2 + c
\]
\[
P(x_3) = ax_3^2 + bx_3 + c
\]

We have three equations and three unknowns, so in general the constants \((a, b, c)\) can be uniquely determined. In matrix form the set of equations are written:

\[
\begin{bmatrix}
x_1^2 & x_1 & 1 \\
x_2^2 & x_2 & 1 \\
x_3^2 & x_3 & 1
\end{bmatrix}
\begin{bmatrix}
a \\
b \\
c
\end{bmatrix}
= 
\begin{bmatrix}
P(x_1) \\
P(x_2) \\
P(x_3)
\end{bmatrix}
\]
Which has the solution:

\[
\begin{bmatrix}
a \\
b \\
c
\end{bmatrix}
= \begin{bmatrix}
x_1^2 & x_1 & 1 \\
x_2^2 & x_2 & 1 \\
x_3^2 & x_3 & 1
\end{bmatrix}^{-1}
\begin{bmatrix}
P(x_1) \\
P(x_2) \\
P(x_3)
\end{bmatrix}
\]

Some algebra yields the solution for the minimum as:

\[
\hat{x}^* = \frac{1}{2} \frac{\left( x_2^2 - x_3^2 \right) P(x_1) + \left( x_3^2 - x_1^2 \right) P(x_2) + \left( x_1^2 - x_2^2 \right) P(x_3)}{\left( x_2 - x_3 \right) P(x_1) + \left( x_3 - x_1 \right) P(x_2) + \left( x_1 - x_2 \right) P(x_3)}
\]
Graphically, the situation is:

To perform the next iteration, we need to choose which three points to use next. The textbook gives you some elaborate rules for coding into a computer. If you’re doing hand calculations choose the point $(x_i)$ with the smallest value of the objective function $P(x)$ and the points immediately on either side. Note that $\hat{x}^*$ will not necessarily have the smallest objective function value.
Finally, the accuracy to which the optimum is known at any iteration is given by the two points which bracket the point with the smallest objective function value.

The procedure is:

1. choose three points that bracket the optimum,
2. determine a quadratic approximation for the objective function based on these three points.
3. calculate the optimum of the quadratic approximation (the predicted optimum \( \hat{x}^* \)),
4. repeat steps 1 to 3 until the desired accuracy is reached.
Example: revisit the wastewater treatment problem with $\tau = 1$.

$$\max_t te^{-t} = \min_t -te^{-t}$$

<table>
<thead>
<tr>
<th>it</th>
<th>$t_1$</th>
<th>$t_2$</th>
<th>$t_3$</th>
<th>$P(t_1)$</th>
<th>$P(t_2)$</th>
<th>$P(t_3)$</th>
<th>$t^*$</th>
<th>$P(t^*)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.5</td>
<td>1.5</td>
<td>2.5</td>
<td>-0.3303</td>
<td>-0.3347</td>
<td>-0.2052</td>
<td>1.1953</td>
<td>-0.3617</td>
</tr>
<tr>
<td>1</td>
<td>0.5</td>
<td>1.1953</td>
<td>1.5</td>
<td>-0.3303</td>
<td>-0.3617</td>
<td>-0.3347</td>
<td>1.0910</td>
<td>-0.3664</td>
</tr>
<tr>
<td>2</td>
<td>0.5</td>
<td>1.0910</td>
<td>1.1953</td>
<td>-0.3303</td>
<td>-0.3664</td>
<td>-0.3347</td>
<td>1.0396</td>
<td>-0.3676</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>1.0396</td>
<td>1.0910</td>
<td>-0.3303</td>
<td>-0.3676</td>
<td>-0.3664</td>
<td>1.0185</td>
<td>-0.3678</td>
</tr>
<tr>
<td>4</td>
<td>0.5</td>
<td>1.0185</td>
<td>1.0396</td>
<td>-0.3303</td>
<td>-0.3678</td>
<td>-0.3676</td>
<td>1.0083</td>
<td>-0.3679</td>
</tr>
</tbody>
</table>
4 Newton’s Method

This is a derivative-based method that uses first and second derivatives to approximate the objective function. The optimum of the objective function is then estimated using this approximation.

Consider the 2nd-order Taylor Series approximations of our objective function:

\[ P(x) \simeq P(x_k) + \frac{dP}{dx}\bigg|_{x_k} (x - x_k) + \frac{1}{2} \frac{d^2 P}{dx^2}\bigg|_{x_k} (x - x_k)^2 \]

Then we could approximate the first derivative of the objective function as:

\[ \frac{dP}{dx} \simeq \frac{dP}{dx}\bigg|_{x_k} + \frac{d^2 P}{dx^2}\bigg|_{x_k} (x - x_k) \]
Since we require that the first derivative of the objective function to be zero at the optimum, then:

\[
\frac{dP}{dx} \simeq \frac{dP}{dx} \bigg|_{x_k} + \frac{d^2P}{dx^2} \bigg|_{x_k} (x - x_k) = 0
\]

A little algebra will allow us to estimate what the optimum value of \( x \) is:

\[
\hat{x}^* = x_{k+1} = x_k - \left[ \frac{dP}{dx} \bigg|_{x_k} \right] \left[ \frac{d^2P}{dx^2} \bigg|_{x_k} \right]
\]

This is called a Newton step and we can iterate on this equation to find the optimum value of \( x \).
Graphically, this is:

Newton's method attempts to find the stationary point of a function by successive approximation using the first and second derivatives of the original objective function.
Example: revisit the wastewater treatment problem with $\tau = 1$.

$$\max_t te^{-t} = \min_t -te^{-t}$$

<table>
<thead>
<tr>
<th>it</th>
<th>$t^k$</th>
<th>$P'(t^k)$</th>
<th>$P''(t^k)$</th>
<th>$P(t^k)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.000</td>
<td>1.000</td>
<td>-2.000</td>
<td>-0.000</td>
</tr>
<tr>
<td>1</td>
<td>0.500</td>
<td>0.3303</td>
<td>-0.9098</td>
<td>-0.3303</td>
</tr>
<tr>
<td>2</td>
<td>0.8333</td>
<td>0.0724</td>
<td>-0.5070</td>
<td>-0.3622</td>
</tr>
<tr>
<td>3</td>
<td>0.9762</td>
<td>0.0090</td>
<td>-0.3857</td>
<td>-0.3678</td>
</tr>
<tr>
<td>4</td>
<td>0.9994</td>
<td>0.0002</td>
<td>-0.3683</td>
<td>-0.3679</td>
</tr>
<tr>
<td>5</td>
<td>1.000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
In summary, Newton’s Method:

1. shows relatively fast convergence (finds the optimum of a quadratic in a single step),
2. requires first and second derivatives of the function,
3. has a computational load of first and second derivative evaluations, and a Newton step calculation per iteration,
4. may oscillate or not converge when there many local optima or stationary points.
5 Quasi-Newton Methods

A major drawback of Newton’s method is that it requires us to have analytically determined both the first and second derivatives of our objective function. Often this is considered onerous, particularly in the case of the second derivative. The large family of optimization algorithms that use finite difference approximations of the derivatives are called ”Quasi-Newton” methods.

There are a variety of ways first derivatives can be approximated using finite differences:

\[
P'(x) \approx \frac{P(x + \Delta x) - P(x - \Delta x)}{2\Delta x}
\]

\[
P'(x) \approx \frac{P(x + \Delta x) - P(x)}{\Delta x}
\]

\[
P'(x) \approx \frac{P(x) - P(x - \Delta x)}{\Delta x}
\]
Approximations for the second derivatives can be determined in terms of either the first derivatives or the original objective function:

\[ P''(x) \approx \frac{P'(x + \Delta x) - P'(x - \Delta x)}{2\Delta x} \]

\[ P''(x) \approx \frac{P'(x + \Delta x) - P'(x)}{\Delta x} \]

\[ P''(x) \approx \frac{P'(x) - P'(x - \Delta x)}{\Delta x} \]

\[ P''(x) \approx \frac{P(x + \Delta x) - 2P(x) + P(x - \Delta x)}{\Delta x^2} \]

\[ P''(x) \approx \frac{P(x + 2\Delta x) - 2P(x + \Delta x) + P(x)}{\Delta x^2} \]

\[ P''(x) \approx \frac{P(x) - 2P(x - \Delta x) + P(x - 2\Delta x)}{\Delta x^2} \]
The major differences within the Quasi-Newton family of algorithms arises from: the number of function and/or derivatives that must be evaluated, the speed of converge and the stability of the algorithm.
Regula Falsi

This method approximates the second derivative as:

\[
\frac{d^2 P}{dx^2} \approx \frac{p'(x_q) - P'(x_p)}{x_q - x_p}
\]

where \(x_p\) and \(x_q\) are chosen so that \(P'(x_q)\) and \(P'(x_p)\) have different signs (i.e. the two points bracket the stationary point).

Substituting this approximation into the formula for a Newton step yields:

\[
\hat{x}^* = x_{k+1} = x_q - \left[ \frac{\frac{dP}{dx}|_{x_q} - \frac{dP}{dx}\bigg|_{x_p}}{\left(\frac{dP}{dx}|_{x_q} - \frac{dP}{dx}\bigg|_{x_p}\right)} \right]
\]
The Regula Falsi method iterates on this formula taking care on each iteration to retain two points:

\[ \hat{x}^* \quad \text{and} \quad x_p \text{ or } x_q \]

depending upon:

\[ \text{sign}[P'(\hat{x}^*_k)] = \text{sign}[P'(x_p)] \implies \text{keep } \hat{x}^*_k \text{ and } x_q \]

or:

\[ \text{sign}[P'(\hat{x}^*_k)] = \text{sign}[P'(x_q)] \implies \text{keep } \hat{x}^*_k \text{ and } x_p \]
Graphically:
Example: revisit the wastewater treatment problem with $\tau = 1$.

$$\max_t t e^{-t} = \min_t -te^{-t}$$

<table>
<thead>
<tr>
<th>it</th>
<th>$x_p$</th>
<th>$x_q$</th>
<th>$x_k^*$</th>
<th>$P'(x_p)$</th>
<th>$P'(x_q)$</th>
<th>$P'(x_k^*)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>2.000</td>
<td>1.7616</td>
<td>-1</td>
<td>-0.1353</td>
<td>0.1308</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1.7616</td>
<td>1.5578</td>
<td>-1</td>
<td>0.1308</td>
<td>0.1175</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1.5578</td>
<td>1.3940</td>
<td>-1</td>
<td>0.1175</td>
<td>0.0978</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1.3940</td>
<td>1.2699</td>
<td>-1</td>
<td>0.0978</td>
<td>0.0758</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1.2699</td>
<td>1.1804</td>
<td>-1</td>
<td>0.0758</td>
<td>0.00554</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1.1804</td>
<td>1.1184</td>
<td>-1</td>
<td>0.00554</td>
<td>0.0387</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>1.1184</td>
<td>1.0768</td>
<td>-1</td>
<td>0.0387</td>
<td>0.0262</td>
</tr>
</tbody>
</table>

Note: computational load is one derivative evaluation, a sign check and a step calculation per iteration.
In summary, Quasi-Newton methods:

1. show slower convergence rates the Newton’s Method,
2. use some finite difference approximations for derivatives of the objective function,
3. may oscillate or not converge when there many local optima or stationary points.
6 Summary

There are only two basic ideas in unconstrained univariate optimization:

1. direct search (scanning, bracketing, interval elimination, approximation),

2. derivative based methods (optimality conditions, Newton and Quasi-Newton methods).

The direct search methods work directly on the objective function to successively determine better values of \( x \). The derivative based methods try to find the stationary point of the objective function, usually by some iterative procedure.
Choosing which method to use is always a trade-off between:

1. ease of implementation,
2. convergence speed,
3. robustness,
4. computational load (number of function evaluations, derivative evaluations / approximations, etc.).

There are no clear answers as to which univariate method is better. The best choice is problem dependent:

· for nearly quadratic functions, the Newton / Quasi-Newton methods are generally a good choice,
· for very flat functions, interval elimination methods can be very effective.
When the univariate search is buried in a multivariate code, the best choice of univariate search will depend upon the properties and requirements of the multivariate search method.

Finally remember that to solve an unconstrained univariate optimization problem you must:

1. have a function (and derivatives) you can evaluate,
2. choose an optimization method appropriate for the problem,
3. choose a starting point (or interval),
4. select a termination criteria (or accuracy).