2. UNIVARIATE OPTIMIZATION

Problem: $\min_{x \in \mathbb{R}} f(x)$ or $\min_{a \le x \le b} f(x)$

Univariate optimization methods or optimization methods for a one variable function are also called line search methods because in multivariate optimization, e.g.

$$\min f(\mathbf{x}) , \mathbf{x} \in \mathbb{R}^n$$

a typical algorithm is of the following form:

1. Choose starting point $\mathbf{x_0}$, k:=0.
2. Compute search direction $\mathbf{p_k}$.
3. Compute step length α_k by minimizing (or approximating the minimum of)
   $f_1(\alpha) = f(\mathbf{x_k} + \alpha \mathbf{p_k})$.
4. Set $\mathbf{x_{k+1}} := \mathbf{x_k} + \alpha_k \mathbf{p_k}$
5. Terminate if convergence conditions are satisfied. Else, set k:=k+1 and go to step 2.

UNIMODAL FUNCTION

Function f(x) is *unimodal* in [a, b] if there exists a unique $x^* \in [a, b]$ such that, given any $x_1, x_2 \in [a, b]$ for which $x_1 < x_2$:

if $x_2 < x^*$ then $f(x_1) > f(x_2)$
if $x_1 > x^*$ then $f(x_1) < f(x_2)$.

A unimodal function has a unique local minimum point x^* which is also global. Function f is strictly monotonic on both sides of x^*. (There are other definitions that do not require uniqueness.)

The univariate minimization methods can be divided into two main categories:
1) Backeting or elimination methods, e.g. methods of bisection, Fibonacci, golden section.
2) Interpolation methods, e.g. Newton's method, polynomial interpolation.

2.1. BRACKETING METHODS

Problem: $\min_{x \in [a,b]} f(x)$

Let us assume f is unimodal in the interval [a,b]. The bracketing methods are based on comparing function values, no derivatives required (except for the bisection method for derivative function f '(x)). No continuity or smoothness properties assumed.

The solution given by these methods is an interval containing the minimum point. When needed, the midpoint of the final interval serves as an approximation for the minimum point.

2.1.1. METHOD OF BISECTION

This method is used for finding a zero of a function f(x) on an interval [a, b]. Assume f(a)f(b) < 0. The interval of uncertainty is reduced by repeating the following steps:

1. Calculate f at the midpoint $x = (a+b)/2$.
2. If $f(x)=0$ or $b-a < \delta$, terminate.
   If $f(x)f(a) < 0$, set $b:=x$. Else, set $a:=x$.

The interval is halved at each iteration. The required tolerance is achieved with $\log_2 ((b-a)/\delta)$ function evaluations.

Bisection and other zero finding algorithms (e.g. the secant method) can be used for finding zero of f '(x), a candidate for a minimum point.

2.1.2. FIBONACCI SEARCH

When f is unimodal, it is possible to reduce the interval by comparing two function values.

Interval elimination:

Evaluate f(x) in two points x_1, x_2, for which $a \leq x_1 < x_2 \leq b$.

1. If $f(x_1) < f(x_2)$, the minimum lies on interval $[a, x_2]$ and $[x_2, b]$ can be eliminated.
2. If $f(x_1) > f(x_2)$, the minimum lies on interval $[x_1, b]$ and $[a, x_1]$ can be eliminated.
3. If $f(x_1) = f(x_2)$, the minimum lies on interval $[x_1, x_2]$.

Fibonacci numbers: $F_0 = F_1 = 1$,
$F_k = F_{k-1} + F_{k-2}$, k=2,3,...

Fibonacci sequence: 1, 1, 2, 3, 5, 8, 13, ...

Let
L_k = interval of uncertainty after k function evaluations
L_k^* = distance of the last (k^{th}) point from the other end point

N is fixed $\Rightarrow$ Final interval of uncertainty after N function evaluations is approximately
$L_N = L_0 / F_N$.

Fibonacci search

1. $L_0 = b-a$,
   $L_2^* = (F_{N-2}/F_N)\, L_0$

Evaluate f at points $x_1 = a + L_2^*$
$x_2 = b - L_2^*$

Elimination step:
If $f(x_1) < f(x_2)$, the minimum lies on interval $[a, x_2]$. Set $b:=x_2$.
If $f(x_1) > f(x_2)$, the minimum lies on interval $[x_1, b]$. Set $a:=x_1$.

Length of the interval after elimination: $L_2 = L_0 - L_2^* = (1 - F_{N-2}/F_N) / L_0 = (F_{N-1}/F_N)\, L_0$.
The current interval contains one point where f has been evaluated, at distance L_2^* of the other end.

2. For k = 3,...,N-1:

Distance from the end points to the evaluation points at the k^{th} stage is $L_k^* = (F_{N-k} / F_N)\, L_0$.

Evaluate f at points $x_1 = a + L_k^*$
$x_2 = b - L_k^*$ (Only one new point!)

Carry out the elimination step.

After elimination i.e. after k function evaluations:
Length of the interval is $L_k = (F_{N-(k-1)} / F_N) L_0$
Distance of the last point from the other end point is $L_k^* = (F_{N-k} / F_N) L_0 = (F_{N-k} / F_{N-(k-2)}) L_{k-1}$

3. After k = N-1 function eveluations:
Length of the interval is $L_{N-1} = (F_2/F_N)L_0 = 2L_0 / F_N$
Distance of the last point from the other end point is $L_{N-1}^* = (F_1/F_N) L_0 = L_0 / F_N$,
so the last point is the midpoint.

Finally, evaluate f at a negligible distance δ from the midpoint and eliminate.
Final length of the uncertainty interval approximately

$L_N = L_0/F_N$.

Fibonacci search yields the maximum reduction in the interval of uncertainty for a given number of function evaluatons.

Example 2.1. Minimize f(x) = |x - 0.7| in the interval [0, 2] using the Fibonacci search with N=5 function evaluations.

2.1.3. GOLDEN SECTION SEARCH

Like Fibonacci search, based on interval elimination where only one new function evaluation is needed at every step.

Definition of golden section: Divide a line segment into two parts c and d such that the ratio between the longer part d and the whole segment c+d is the same as the ratio between the shorter part c and the longer part d.

This is the ratio of the golden section:

$$\tau = \frac{d}{c + d} = \frac{c}{d}$$

$$\Rightarrow \quad \tau^2 + \tau - 1 = 0 \quad \Rightarrow$$

$$\tau = \frac{-1 + \sqrt{5}}{2} \approx 0.6180$$

It can be shown that

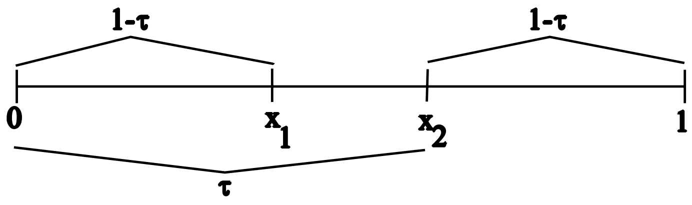
$$\lim_{k \to \infty} \frac{F_{k-1}}{F_k} = \tau \approx 0.6180$$

Golden section search:
Evaluate f at a fraction τ of the end points:

$$x_1 = b - \tau(b-a) = a + (1-\tau)(b-a)$$
$$x_2 = a + \tau(b-a) = b - (1-\tau)(b-a)$$

Eliminate. Only one new function evaluation is needed at subsequent iterations.
The search is terminated when the the interval has reduced below a given tolerance.

Let a=0, b=1, $L_0 = 1$



If $f(x_1) < f(x_2)$, the remaining interval after the elimination is $[0,x_2]$, of length τ. The other point will be placed symmetrically with x_1, at distance $1-\tau$ from the right end point. This should correspond to the fraction τ of the current interval of length τ, giving the condition
$\tau \cdot \tau = 1 - \tau \Rightarrow \tau \approx 0.6180$.

The interval of uncertainty after n function evaluations is $L_n = \tau^{n-1} L_0 = \tau^{n-1}(b - a)$.

Example 2.2. Minimize $f(x) = |x - 0.7|$ in the interval [0, 2] using the golden section search where the final interval of uncertainty $L_N < 0.25$.

2.1.4. BOUNDING THE MINIMUM

How to find the initial bracketing interval [a, b] for a unimodal function f(x)?
Here is one version of an accelerated search:

1. Set initial point x_0 and step size $d_0 > 0$.

Evaluate f at x_0-d_0, x_0, x_0+d_0.

If $f(x_0-d_0) \geq f(x_0) \geq f(x_0+d_0)$, the minimum lies to the rigth of x_0. Set $d := d_0$. $(x_0 \leq x^*)$
If $f(x_0-d_0) \leq f(x_0) \leq f(x_0+d_0)$, the minimum lies to the left of x_0. Set $d:=-d_0$. $(x^* \leq x_0)$
If $f(x_0-d_0) \geq f(x_0) \leq f(x_0+d_0)$, the minimum is bracketed. Set $a:=x_0-d_0$, $b:=x_0+d_0$. Stop.

2. For k=0,1...

$x_{k+1} := x_k + 2^k d$

If $f(x_{k+1}) \geq f(x_k)$, set $a:= x_{k-1}$, $b:= x_{k+1}$ or $a:= x_{k+1}$, $b:= x_{k-1}$ and stop.

Alternatively, a fixed step size could be used in the above recursion.