One-Dimensional Optimization

Optimization is the process of finding the value of a variable that minimizes or maximizes a real function. It is very important since we are more often than not interested in properties at the extrema. For example, we may develop a model to interpret our seismic observations and want to find the parameters that minimize the difference between our model and the observations.

In the lecture we looked at the golden-section search method. It makes use of the golden ratio so that we only need to calculate one of x_1 or x_2 at each iteration.

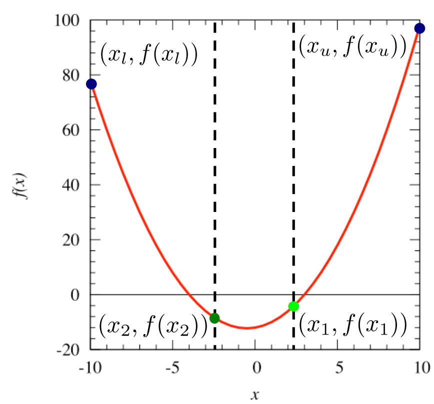
$$\text{Golden Ratio} = \frac{1+\sqrt{5}}{2}$$

$$\boxed{\begin{aligned} d &= \frac{x_u - x_l}{\phi} \\ x_1 &= x_l + d \\ x_2 &= x_u - d \end{aligned}} \qquad \begin{aligned} &f(x_1) \\ &f(x_2) \end{aligned}$$

$$x_o = \text{predicted location of optimum}$$

$$\boxed{\begin{gathered} \text{if } f(x_2) < f(x_1) \text{ then } x_o = x_2 \\ x_u \Leftarrow x_1 \quad x_1 \Leftarrow x_2 \end{gathered}}$$

$$\begin{gathered} \text{if } f(x_1) < f(x_2) \text{ then } x_o = x_1 \\ x_l \Leftarrow x_2 \quad x_2 \Leftarrow x_1 \end{gathered}$$



next iteration

$$\varepsilon_a = \left| \frac{x_2 - x_l}{x_o} \right| \times 100\%$$

Note that you cannot use the relative error for the golden-section search method, and must use the alternative estimate.

Your Task

Your task is to write a MATLAB function that uses the Golden-Section search method to find the minimum of a predefined function, when provided with two bracketing values. You can presume that the function is real and continuous.

Your MATLAB function should stop either when the approximate error has fallen below a certain limit or a pre-defined maximum number of iterations is reached. In addition, your function should write out the iteration number, the estimated value for the optimum and approximate error at each iteration. Upon completion, it should plot the estimated values and associated error as a function of iteration. Plot the error as confidence bands.

The function, bracketing values, error limit and maximum number of iterations should be passed to the function at the command line.

For example

```
>> f=@(x) x^2
>> xl = -1; xu = 1, miner = 0.000001; maxit = 20;
>> gss(f,xl,xu,miner,maxit)
```

or in a more efficient manner

```
>> f=@(x) x^2
>> gss(f,-1,1,0.000001,20)
```

Testing Your MATLAB Script

Use your program to find the minimum of the function

$$f(x) = e^x - 5x$$

Use 1.0 and 2.0 as your two bracketing values or, even better, use Matlab to plot the function to estimate two values close to the minimum. Set the minimum error to 0.000001 % and maximum number of iterations to 40.

The true value is: 1.609 (to 3 d.p.).

Further Work

If you have time, write a MATLAB script to perform parabolic interpolation and use it to solve the same problem.