Golden Section Search

To begin a description of the golden section search, we will define "unimodal."

> A function *f(x)* is *unimodal* if on the closed interval *[a,b]* if there exists a *unique* number $c \in [a,b]$ such that *f(x)* is strictly decreasing on the interval *[a,c]* and strictly increasing on the interval *[c,b]*.

This is a derivative-free way of saying that *f(x)* has a unique minimum on the interval *[a,b]*.

The Golden Section Search is a way of calculating this minimum.

> Given a function *f(x),* which is unimodal on a closed interval *[a,b],* calculate the value *c* such that $f(c) \leq f(x)$ for all values of *x* on the interval *[a,b]*. In short, calculate the minimum of the function *f(x)* on the closed interval *[a,b]*.

The Golden Section Search is a direct method of calculating the minimum within a user-defined tolerance, which is input. Given a unimodal function *f*, and endpoints *a* and *b,* calculate two intermediate points,

$$x_1 = a+r(b-a)$$

and

$$x_2 = b-r(b-a),$$

where

$$r = \frac{3-\sqrt{5}}{2}.$$

If $f(x_1) < f(x_2)$, then, because *f* is unimodal, we can rule out the possibility that the minimum is between x_2 and *b*. So, we set *b* equal to x_2. Then, we set x_2 equal to x_1. Finally, we calculate a new x_1 using the formula above. On the other hand, if $f(x_1) \geq f(x_2)$, then we can eliminate the possibility that the minimum is between *a* and x_1. So, we set *a* equal to x_1. Then, we set x_1 equal to x_2. Finally, we calculate a new x_2 using the above formula.

Note that we first calculate values of x_1 and x_2 **before** we enter the "while" loop. Then, we also have to calculate values of x_1 and x_2 **inside** the "while" loop.

How long do we continue? We continue while the absolute error is greater than the user-defined tolerance t,

$$|b-a| > t,$$

and the relative error is greater than the user-defined tolerance t,

$$\frac{2|b-a|}{|a|+|b|} > t.$$

Then, we return the vector consisting of the x-coordinate $(a+b)/2$ and the y-coordinate $f((a+b)/2)$.

Write the user-defined, **R** function `golden.R` that implements the golden section search. The inputs are the function f, the left endpoint a, the right endpoint b, and the tolerance t. The output is vector consisting of the x-coordinate $(a+b)/2$ and the y-coordinate $f((a+b)/2)$.

Here are some examples to try.

```
> golden(h1,-1.9,1.9,0.001)
[1] -0.9998269  0.5000000

> golden(h2,1,3,0.001)
[1]  2.355935 -2.885618

> golden(h3,0.5,3,0.001)
[1] 2.000550 1.847264
```