MATH 335: Numerical Analysis

Problem Set 5, Solutions

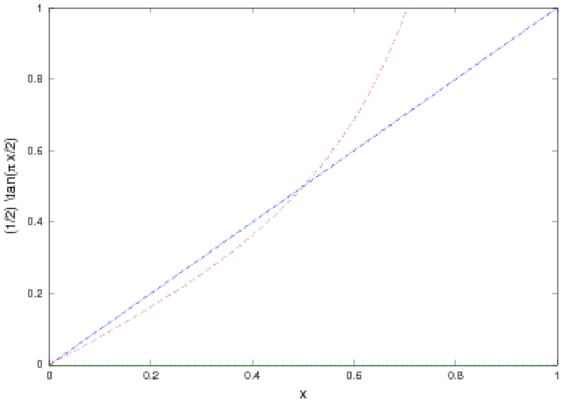
<u>3.3.5</u> (a) One possible interval would be $[0.8, 1.2]$, as clearly the function $(x \mapsto 1 + 0.2\sin x)\colon [0.8, 1.2] \to [0.8, 1.2]$, and the derivative of this function is $0.2\cos x$ which, in absolute value, never exceeds 0.2.

(b) We might choose an interval like $[0, 1]$, since $(x \mapsto 1 - x^2/4)\colon [0, 1] \to [0, 1]$ (in fact, it maps $[0, 1]$ into the even smaller interval $[0.75, 1]$), and its derivative $(-x/2)$ never exceeds $1/2$ in magnitude in this interval.

<u>3.3.12</u> As may be seen, for all x-values sufficiently close to $\alpha = 0.5$ (i.e., in some interval $[0.5 - \delta, 0.5 + \delta]$ with $\delta > 0$), the values of the derivative function

$$g'(x) = \frac{\pi}{4}\sec^2\left(\frac{\pi x}{2}\right)$$

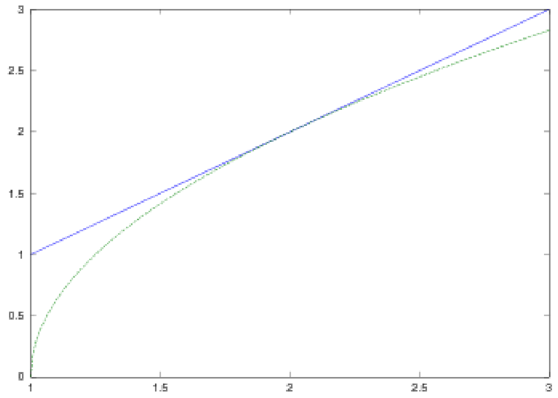
are greater than 1 (in fact, for all $x \in ((2/\pi)\arccos(\sqrt{\pi}/2), 1] \approx (0.30663, 1])$. If some iterate x_n fell inside $[0.5 - \delta, 0.5 + \delta]$ (but were still not equal to $\alpha = 0.5$), then the distance from the next iterate to α would satisfy



$$|x_{n+1} - \alpha| = |g(x_n) - g(\alpha)| = \left|\frac{g(x_n) - g(\alpha)}{x_n - \alpha}\right| |x_n - \alpha| = |g'(\xi)||x_n - \alpha| \geq |x_n - \alpha| .$$

That is, iterate x_{n+1} would be farther from α than iterate x_n.

<u>3.3.14</u> The plot gives the appearance that the graphs of $y = x$ and $y = g(x)$ intersect precisely once at a point of tangency. To see that this is, indeed, the case, consider the function



$$r(x) = x - 2\sqrt{x - 1} ,$$

which has derivative

$$r'(x) = 1 - \frac{1}{\sqrt{x - 1}} .$$

It can be easily seen that both $r'(2)$ and $r(2)$ are zero (i.e., the two functions are equal and have equal derivatives at $x = 2$, hence are tangent there). Here, side-by-side, are some results of the two iterations:

n	p_n	n	p_n
0	1.5	0	2.5
1	1.41421	1	2.4495
2	1.28719	2	2.4079
3	1.07180	3	2.3731
4	0.53591	4	2.3436
5	1.36249 i	5	2.3183
6	1.17480 + 2.31952 i	6	2.2963
	$\vdots$		$\vdots$
500	2.0082 + 0.0001 i	500	2.0079

We see that $p_0 = 2.5$ is a seed that lies in the region to the right of the fixed point $x = 2$, where the derivative satisfies $0 < g'(x) < 1$. Its progression is steadily towards $x = 2$, though it is quite slow (as n increases, $g'(p_n)$ gets closer to 1).

In contrast, for $x < 2$ the derivative $g'(x) > 1$, and iterates p_n move away from the fixed point $x = 2$. It is not surprising that, once $p_n < 1$, the next iterate is a nonreal number. What is surprising is that, ultimately, the sequence somehow makes it back to the fixed point $x = 2$.

3.2.7 Since f is continuous and

$f(-6.25) = -15.391$ and $f(-6) = 11.940$, there is a root in $[-6.25, -6]$;
$f(3.5) = 2.9150$ and $f(4) = -2.8600$, there is a root in $[3.5, 4]$;
$f(4) = -2.8600$ and $f(5) = 0.5000$, there is a root in $[4, 5]$.

Using the program `falsep.m` supplied with the text, we get the following results:

```
octave:463> function y = f (x)
> y = x .^ 3 - 2.56 * x .^ 2 - 34.6 * x + 112.5;
> end

octave:464> falsep(@f, -6.25, -6, .000001, 25)
_________________________________________________________________
 iter     a             b           c             f(c)       |b-a|
_________________________________________________________________

 0      -6.2500      -6.0000     -6.109218     0.321652    0.250000
```

```
1     -6.2500     -6.1092     -6.112100    0.008358   0.140782
2     -6.2500     -6.1121     -6.112175    0.000217   0.137900
3     -6.2500     -6.1122     -6.112177    0.000006   0.137825
4     -6.2500     -6.1122     -6.112177    0.000000   0.137823
```

We get an approximate root -6.112177, and see that the iteration ended not because the bracketing interval grew to be as small as our tolerance 10^{-6}, but rather because the movement from c_n to c_{n+1} dropped below this level.

A similar call to `falsep.m` with bracketing intervals $[3.5, 4]$ and $[4, 5]$ yields approximate roots 3.706976 and 4.965201 respectively.

3.5.5 There are three zeros apparent (as determined by the graph of f) in the interval $[-3, 3]$. Here are the results using the program `newton.m` (supplied with the text) when the initial seed $x_0 = 0$:

```
octave:475> function y = f(x)
> y = x.^5 - 3*x.^3 - 5*x + 4;
> end

octave:476> function y = fprime(x)
> y = 5*x.^4 - 9*x.^2 - 5;
> end

octave:477> newton(@f, @fprime, 0, .000001, 20)
-------------------------------------------------------------
 iter      x           f(x)         df(x)      |xn+1-xn|
-------------------------------------------------------------
 0      0.000000     4.000000    -5.000000
 1      0.800000    -1.208320    -8.712000     0.800000
 2      0.661304    -0.047655    -7.979650     0.138696
 3      0.655332    -0.000109    -7.942960     0.005972
 4      0.655318    -0.000000    -7.942875     0.000014
 5      0.655318     0.000000    -7.942875     0.000000
```

The zero it has found is approximately 0.655318. The other zeros are approximately -2.126445 and 1.943876.

3.5.17 (a) We see $x = 0$ is a root for all choices of λ. But, for $\lambda \leq 1$ there are no roots in the interval $(0, \pi/2)$ ($y = \tan x$ takes off from the origin with slope 1 and gets steeper as it goes; $y = \lambda x$ takes off from the origin with slope λ). When $\lambda > 1$ a root in that interval appears. Apparently (as I am told by Matt Walhout of the Physics Dept.), the value of λ is related to the "height" of the walls of the box; only once this root appears (i.e., the walls are "high" enough) does containment of the particle become possible.

(b) There should be approximately one root per period of $\tan x$, and these roots approach $(2n+1)\pi/2$ asymptotically for large n, x.

(c) Taking $f(x) = \lambda x - \tan x$, we get

$$g(x) \;:=\; x - \frac{f(x)}{f'(x)} \;=\; x - \frac{\lambda x - \tan x}{\lambda - \sec^2 x} \;=\; \frac{\tan x - x\sec^2 x}{\lambda - \sec^2 x}\;.$$

Using this g, $x_{n+1} = g(x_n)$.

(d) After trying it with starting values x_0 at π and near $\pi/2$—both yielding iterations that did not converge to anything useful—I tried $x_0 = 3\pi/2 - 0.1 \doteq 4.6124$ and got, in the case where $\lambda = 0.95$, the root $\alpha \doteq 4.481697$:

```
octave:484> newton(@f, @fprime, 3*pi/2-.1, .000001, 20)
___________________________________________________________
 iter      x            f(x)          df(x)       |xn+1-xn|
___________________________________________________________
 0      4.612389     -5.584875    -99.384001
 1      4.556194     -2.021723    -40.373865      0.056195
 2      4.506119     -0.498250    -22.889446      0.050075
 3      4.484351     -0.048830    -18.617167      0.021768
 4      4.481729     -0.000573    -18.182399      0.002623
 5      4.481697     -0.000000    -18.177262      0.000032
 6      4.481697     -0.000000    -18.177261      0.000000
```

When $\lambda = 1$ and $\lambda = 1.05$, I get approximate roots $\alpha = 4.493409$ and $\alpha = 4.504007$ respectively.

★4 (a) Let us start with $f(x) = x^n - A$, a simple function that has $\sqrt[n]{A}$ as a root. Newton's method is the fixed-point iteration that arises from

$$g(x) \;=\; x - \frac{f(x)}{f'(x)} \;=\; x - \frac{x^n - A}{nx^{n-1}} \;=\; \frac{nx^n - (x^n - A)}{nx^{n-1}} \;=\; \frac{(n-1)x^n + A}{nx^{n-1}}\,,$$

which is equivalent to the formula above.

(b) With code like the following

```
octave> function y = g(x, enn, ay)
> y = ((enn - 1)*x + ay/x^(enn - 1)) / enn;
> end

octave> xIters = 1.5;
octave> for i=1:20
> xIters = [xIters g(xIters(length(xIters)), 4, 7)];
> end
```

I get the approximate value $\sqrt[4]{7} \doteq 1.62657656169779$. This happens to be the same value that appears if I enter the command

```
> 7^(1/4)
```

and even though I calculated 20 iterates, upon further inspection, the 4th iterate was already accurate to this many places.