Numerical Methods – HW7

b04902053 鄭淵仁

June 22, 2018

Environment

- **OS:** Windows
- **CPU:** Intel Core i5-4200H
- **Programming Language:** Matlab R2018a

Problem 7-1: Write bisection and Newton methods.

Codes

Bisection

```
function [p, iter] = bisec(f, a, b, tol)
    if f(a) * f(b) >= 0
        disp('f(a) * f(b) >= 0');
    else
        iter = 0;
        p = (a + b) / 2;
        while abs(f(p)) > tol
            iter = iter + 1;
            if f(a) * f(p) < 0
                b = p;
            else
                a = p;
            end
            p = (a + b) / 2;
        end
    end
end
```

Newton's Method

```
function [p, iter] = newton(f, df, x0, tol, max_iter)
    p = x0;
    iter = 0;
    while (abs(f(p) > tol)) && (iter < max_iter)
        iter = iter + 1;
        p = p - f(p) / df(p);
    end
end
```

Check Linear Convergence Rates

```
conv_rates = [];
while % ...
    % ... (Methods)
    % Record Convergence Rates
    if iter > 1
        conv_rates(end + 1) = abs((p - p_star) / last_err);
    end
    last_err = p - p_star;
```

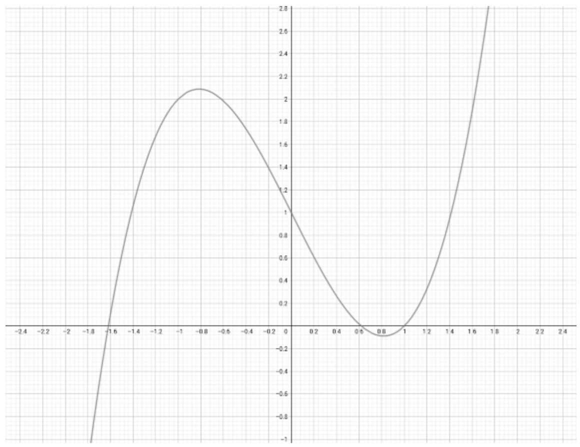
Check Quadratic Convergence Rates

```
conv_rates = [];
while % ...
    % ... (Methods)
    % Record Convergence Rates
    if iter > 1
        conv_rates(end + 1) = abs((p - p_star) / last_err ^ 2);
    end
    last_err = p - p_star;
```

Test with Functions and Results

Find Roots of Functions

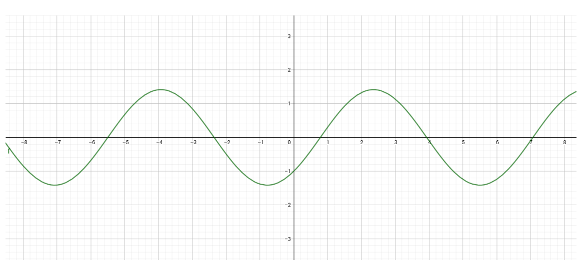
1. $f_1(x) = x^3 - 2x + 1$
   - Graph:

   - Analytical Solutions:
     - $x = 1$
     - $x = -\frac{1}{2} + \frac{\sqrt{5}}{2} \approx 0.6180$
     - $x = -\frac{1}{2} - \frac{\sqrt{5}}{2} \approx -1.6180$
   - Find roots by Bisection / Newton's Method (set tolerance = 1e-14):

	Bisection	**Newton's**
Parameters	$(beg, end) = (0.65, 8)$	$x_0 = 8$
Result	0.999999999999999	1
Iteration	47	11

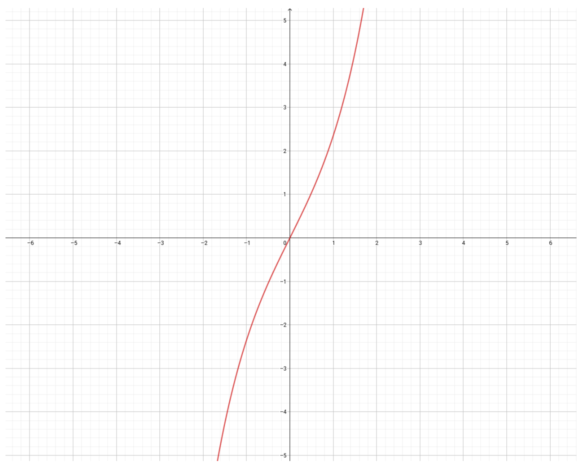
2. $f_2(x) = sin(x) - cos(x)$
   - Graph:

- Analytical Solutions:
  - $x = \frac{1}{4}(4\pi n + \pi), n \in \mathbb{Z}$
- Find roots by Bisection / Newton's Method (set tolerance = 1e-14):

	Bisection	**Newton's**
Parameters	$(beg, end) = (-2, 2)$	$x_0 = 2$
Result	0.785398163397451	0.785398163397448
Iteration	46	6

3. $f_3(x) = e^x - e^{-x}$
   - Graph:



   - Analytical Solutions:
     - $x = 0$
   - Find roots by Bisection / Newton's Method (set tolerance = 1e-14):

	Bisection	**Newton's**
Parameters	$(beg, end) = (-3.7, 4.7)$	$x_0 = 4.7$
Result	1.7552×10^{-15}	-6.3967×10^{-18}
Iteration	49	8

Try Different Initial Solutions

I try with the last function: $f_3(x) = e^x - e^{-x}$.

- Bisection

Parameters (beg, end)	$(-3.7, 4.7)$	$(-3.7, 8.7)$	$(-1.3, 5.5)$
Iteration	49	48	47

- Newton

Parameters (x_0)	4.7	8.7	5.5
Iteration	8	12	9

Check Convergence Rates

I check with the last function: $f_3(x) = e^x - e^{-x}$.

- Bisection
  - Check with **linear convergence**:

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|}$$

  - Rates: approximately repeat the following six values again and again:

$$1.1003, 0.0456, 9.4610, 0.4471, 0.3818, 0.1904$$

- Newton's Method
  - Check with **quadratic convergence**:

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^2}$$

  - Rates: except the last one, all the values are smaller than 1. The whole values are as following:

$$0.1973, 0.2343, 0.2644, 0.2081, 0.0412, 2.1365 \times 10^{-4}, 8.3020 \times 10^{+2}$$

Problem 7-2: Write Newton methods with Multiple Variables.

Codes

Newton's Method

```
function [p, iter] = newton(f, df, x0, tol, max_iter)
    iter = 0;
    p = x0;
    while (sum(f(p) .^ 2) > tol) && (iter < max_iter)
        iter = iter + 1;
        p = p - df(p) \ f(p);
    end
end
```

Jacobian of given f

```
function y = df(x)
    x1 = x(1);
    x2 = x(2);
    x3 = x(3);
    y = [2 * x1, 2 * x2, 2 * x3;
         2 * x1,      0, 2 * x3;
         2 * x1, 2 * x2,     -4];
end
```

Test with Functions and Results

- Analytical Solutions:

$$x = \sqrt{\frac{35}{4} + 4\sqrt{5}} \approx 0.440762872754908$$

$$y = \frac{\sqrt{3}}{2} \approx 0.866025403784439$$

$$z = -2 + \sqrt{5} \approx 0.236067977499790$$

- Test with Different Initial Solutions

Initial Solutions	Iterations
$\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$	4
$\begin{bmatrix} 1 & 10 & 100 \end{bmatrix}$	12
$\begin{bmatrix} 44 & 86 & 23 \end{bmatrix}$	9