Thais Minet

2/10/18

Lab 4 Newton's & Secant Methods

5. Use Newton's method to find solutions accurate to within 10^{-4} for the following problems

a) $x^3 - 2x^2 - 5 = 0,\ [1, 4]$

```
>> f
f =

@(x) x .^ 3 - 2 * x .^ 2 - 5

>> fprime
fprime =

@(x) 3 * x .^ 2 - 4 * x

>> newton(2, f, fprime, 10.^(-4), 100)
ITERATION 0
p = 3.250000
err = 1.250000
p0 = 3.250000

ITERATION 1
p = 2.811037
err = 0.438963
p0 = 2.811037

ITERATION 2
p = 2.697990
err = 0.113047
p0 = 2.697990

ITERATION 3
p = 2.690677
err = 0.007312
p0 = 2.690677

ITERATION 4
```

```
p = 2.690647
err = 0.000030

ans = 2.690647
```

The answer is 2.690647

Verification by direct calculation x = 2.690647

So the result agrees with the direct calculation to the specified accuracy.

c) $x - cos(x) = 0, [0, pi/2]$

```
>> f
f =

@(x) x - cos (x)

>> fprime
fprime =

@(x) 1 + sin (x)

>> newton(2, f, fprime, 10.^(-4), 100)
ITERATION 0
p = 0.734536
err = 1.265464
p0 = 0.734536

ITERATION 1
p = 0.739090
err = 0.004554
p0 = 0.739090

ITERATION 2
p = 0.739085
err = 0.000005

ans = 0.739085
```

The answer is 0.739085

Verification by direct calculation x = 0.739085

So the result agrees with the direct calculation to the specified accuracy.

6. Use Newton's method to find solutions accurate to within 10^{-5} for the following problems.

a) $e^x + 2^{-x} + 2cos(x) - 6 = 0, \; for \; 1 \leq x \leq 2$

```
>> f
f =

@(x) e .^ x + 2 .^ (-x) + 2 * cos (x) - 6

>> fprime
fprime =

@(x) e .^ x - log (2) * 2 .^ (-x) - 2 * sin (x)

>> newton(1.5, f, fprime, 10.^(-5), 100)
ITERATION 0
p = 1.956490
err = 0.456490
p0 = 1.956490

ITERATION 1
p = 1.841533
err = 0.114957
p0 = 1.841533

ITERATION 2
p = 1.829506
err = 0.012027
p0 = 1.829506

ITERATION 3
p = 1.829384
err = 0.000122
p0 = 1.829384
```

```
ITERATION 4
p = 1.829384
err = 0.000000

ans = 1.829384
```

The answer is 1.829384

Verification by direct calculation x = 1.8293836

So the result agrees with the direct calculation to the specified accuracy.

c) $2x\cos(2x) - (x-2)^2 = 0,\ for\ 2 \le x \le 3\ and\ 3 \le x \le 4$

$for\ 2 \le x \le 3$

```
>> f
f =

@(x) 2 * x * cos (2 * x) - (x - 2) .^ 2

>> fprime
fprime =

@(x) -2 * x - 4 * x * sin (2 * x) + 2 * cos (2 * x) + 4

>> newton(2.5, f, fprime, 10.^(-5), 100)
ITERATION 0
p = 2.372407
err = 0.127593
p0 = 2.372407

ITERATION 1
p = 2.370688
err = 0.001719
p0 = 2.370688

ITERATION 2
p = 2.370687
```

```
err = 0.000001

ans = 2.370687
```

The answer is 2.370687

Verification by direct calculation x = 2.3706869

So the result agrees with the direct calculation to the specified accuracy.

$for\ 3 \le x \le 4$

```
>> f
f =

@(x) 2 * x * cos (2 * x) - (x - 2) .^ 2

>> fprime
fprime =

@(x) -2 * x - 4 * x * sin (2 * x) + 2 * cos (2 * x) + 4

>> newton(3.5, f, fprime, 10.^(-5), 100)
ITERATION 0
p = 3.783191
err = 0.283191
p0 = 3.783191

ITERATION 1
p = 3.724165
err = 0.059026
p0 = 3.724165

ITERATION 2
p = 3.722115
err = 0.002050
p0 = 3.722115

ITERATION 3
p = 3.722113
```

```
err = 0.000003

ans = 3.722113
```

The answer is 3.722113

Verification by direct calculation x = 3.7221127

So the result agrees with the direct calculation to the specified accuracy.

7. Use the secant method to find solutions accurate to within 10^{-4} for the following problems

a) $x^3 - 2x^2 - 5 = 0,\ [1, 4]$

```
>> f
f =

@(x) x .^ 3 - 2 * x .^ 2 - 5

>> secant(2, 3, f, 10.^(-4), 100)
ITERATION 1
p0 = 2.000000; p1 = 3.000000
q0 = -5.000000; q1 = 4.000000
p = 2.555556
err = 0.444444

ITERATION 2
p0 = 3.000000; p1 = 2.555556
q0 = 4.000000; q1 = -1.371742
p = 2.669050
err = 0.113494

ITERATION 3
p0 = 2.555556; p1 = 2.669050
q0 = -1.371742; q1 = -0.233802
p = 2.692369
err = 0.023319

ITERATION 4
```

```
p0 = 2.669050; p1 = 2.692369
q0 = -0.233802; q1 = 0.018877
p = 2.690627
err = 0.001742

ITERATION 5
p0 = 2.692369; p1 = 2.690627
q0 = 0.018877; q1 = -0.000227
p = 2.690647
err = 0.000021

ans = 2.690647
```

The answer is 2.690647

Verification by direct calculation x = 2.690647

So the result agrees with the direct calculation to the specified accuracy.

c) $x - cos(x) = 0, [0, pi/2]$

```
>> f = @(x) x - cos(x)
f =

@(x) x - cos (x)

>> secant(0, pi/2, f, 10.^(-4), 100)
ITERATION 1
p0 = 0.000000; p1 = 1.570796
q0 = -1.000000; q1 = 1.570796
p = 0.611015
err = 0.959781

ITERATION 2
p0 = 1.570796; p1 = 0.611015
q0 = 1.570796; q1 = -0.208050
p = 0.723270
err = 0.112254

ITERATION 3
p0 = 0.611015; p1 = 0.723270
```

```
q0 = -0.208050; q1 = -0.026376
p = 0.739567
err = 0.016298

ITERATION 4
p0 = 0.723270; p1 = 0.739567
q0 = -0.026376; q1 = 0.000807
p = 0.739083
err = 0.000484

ITERATION 5
p0 = 0.739567; p1 = 0.739083
q0 = 0.000807; q1 = -0.000003
p = 0.739085
err = 0.000002

ans = 0.739085
```

The answer is 0.739085

Verification by direct calculation x = 0.739085

So the result agrees with the direct calculation to the specified accuracy.

8. Use the secant method to find solutions accurate to within 10^{-5} for the following problems.

a) $e^x + 2^{-x} + 2cos(x) - 6 = 0, \; for \; 1 \leq x \leq 2$

```
>> f
f =

@(x) e .^ x + 2 .^ (-x) + 2 * cos (x) - 6

>> secant(1, 2, f, 10.^(-5), 100)
ITERATION 1
p0 = 1.000000; p1 = 2.000000
q0 = -1.701114; q1 = 0.806762
p = 1.678308
```

```
err = 0.321692

ITERATION 2
p0 = 2.000000; p1 = 1.678308
q0 = 0.806762; q1 = -0.545674
p = 1.808103
err = 0.129794

ITERATION 3
p0 = 1.678308; p1 = 1.808103
q0 = -0.545674; q1 = -0.085739
p = 1.832298
err = 0.024196

ITERATION 4
p0 = 1.808103; p1 = 1.832298
q0 = -0.085739; q1 = 0.011985
p = 1.829331
err = 0.002967

ITERATION 5
p0 = 1.832298; p1 = 1.829331
q0 = 0.011985; q1 = -0.000215
p = 1.829383
err = 0.000052

ITERATION 6
p0 = 1.829331; p1 = 1.829383
q0 = -0.000215; q1 = -0.000001
p = 1.829384
err = 0.000000

ans = 1.829384
```

The answer is 1.829384

Verification by direct calculation x = 1.8293836

So the result agrees with the direct calculation to the specified accuracy.

c) $2x\cos(2x) - (x-2)^2 = 0, \; for\ 2 \le x \le 3\ and\ 3 \le x \le 4$

$for\ 2 \le x \le 3$

```
>> f
f =

@(x) 2 * x * cos (2 * x) - (x - 2) .^ 2

>> secant(2, 3, f, 10.^(-5), 100)
ITERATION 1
p0 = 2.000000; p1 = 3.000000
q0 = -2.614574; q1 = 4.761022
p = 2.354490
err = 0.645510

ITERATION 2
p0 = 3.000000; p1 = 2.354490
q0 = 4.761022; q1 = -0.141717
p = 2.373149
err = 0.018659

ITERATION 3
p0 = 2.354490; p1 = 2.373149
q0 = -0.141717; q1 = 0.021669
p = 2.370674
err = 0.002475

ITERATION 4
p0 = 2.373149; p1 = 2.370674
q0 = 0.021669; q1 = -0.000113
p = 2.370687
err = 0.000013

ITERATION 5
p0 = 2.370674; p1 = 2.370687
q0 = -0.000113; q1 = -0.000000
p = 2.370687
err = 0.000000

ans = 2.370687
```

The answer is 2.370687

Verification by direct calculation x = 2.3706869

So the result agrees with the direct calculation to the specified accuracy.

$for\ 3 \le x \le 4$

```
>> f
f =

@(x) 2 * x * cos (2 * x) - (x - 2) .^ 2

>> secant(3, 4, f, 10.^(-5), 100)
ITERATION 1
p0 = 3.000000; p1 = 4.000000
q0 = 4.761022; q1 = -5.164000
p = 3.479699
err = 0.520301

ITERATION 2
p0 = 4.000000; p1 = 3.479699
q0 = -5.164000; q1 = 3.238465
p = 3.680233
err = 0.200534

ITERATION 3
p0 = 3.479699; p1 = 3.680233
q0 = 3.238465; q1 = 0.663661
p = 3.731920
err = 0.051688

ITERATION 4
p0 = 3.680233; p1 = 3.731920
q0 = 0.663661; q1 = -0.160911
p = 3.721834
err = 0.010087

ITERATION 5
p0 = 3.731920; p1 = 3.721834
q0 = -0.160911; q1 = 0.004547
p = 3.722111
err = 0.000277
```

```
ITERATION 6
p0 = 3.721834; p1 = 3.722111
q0 = 0.004547; q1 = 0.000029
p = 3.722113
err = 0.000002

ans = 3.722113
```

The answer is 3.722113

Verification by direct calculation x = 3.7221127

So the result agrees with the direct calculation to the specified accuracy.