Project 2: Newton's Method

Due: October 18, 1999

In this session, we will learn about a common technique for iteratively approximating the solutions to an equation of the form $f(x) = 0$. The method we will use is called ''Newton's Method'', or ''The Newton-Raphson Method''. Please read pages 345-349 from your textbook, so that you will understand what you are doing.

We Need to Differentiate!

To use Newton's Method we need to differentiate. Maple uses the command D to differentiate functions, and *diff* to differentiate expressions.

```
> F:=x -> x^2;
> F_prime:=D(F);
> E:=F(x);
> E_p:=diff(E,x);
> E_prime:=unapply(E_p,x);
```

Exercises

To practice the differentiation commands, solve the following: Find the line tangent to $y = f1(x)$ at $x = a$ in the next problem. Plot the function and the tangent line on the same coordinate system.

1. $f1(x) = x\sin(x)$, $a = \frac{\pi}{2}$.

Newton's Method

As our first example of using Newton's method, we will compute approximations to $\sqrt{1000}$. Finding this number requires solving the equation $x^2 - 1000 = 0$. Therefore we will define a function $f(x) = x^2 - 1000$ and then use Newton's Method to solve the equation $f(x) = 0$.

```
> f := x -> x^2 - 1000;
> N := x -> x - f(x)/D(f)(x);
```

N is a function which accepts an approximate solution to $f(x) = 0$ and (hopefully) computes a more accurate approximation. We will begin by guessing that $a_1 = 20$ is a reasonable first approximation to $\sqrt{1000}$.

```
> a[1] := 20.0;
> a[2] := N(a[1]);
> a[3] := N(a[2]);
> a[4] := N(a[3]);
> a[5] := N(a[4]);
> a[6] := N(a[5]);
```

Notice that $a_5 = a_6$. This means that if we continue our approximations we will get the same number for all future approximations. This does NOT mean we have the answer, but only that (possibly because of roundoff error) Maple cannot compute a better answer by this method. We should check our answer, and also increase the values of *Digits* if we are not satisfied.

```
> a[6]^2;
```

I'm satisfied. (Are you??)

Exercises

2. Find the cube root of 5000 using Newton's Method. Start with an initial approximation of $a_1 = 30$ and continue until successive approximations are equal.
3. Find a root of the polynomial $f(x) = x^5 - 3x^3 + 2x + 1$. Start with $a_1 = -2$.
4. Find a value of x for which $\tan(x) = 5$ by solving $f(x) = 0$ for $f(x) = \tan(x) - 5$. Pick your own starting value, and continue until successive approximations are equal.

Loops Make Life Easier

It is tedious to calculate the successive approximations $a_1, a_2, a_3,$ ... one by one. We can write a loop to do all the calculations at once.

```
> restart;
> f := x -> x^2 - 1000;
> N := x -> x - f(x)/D(f)(x);
> a[0]:=0.0;
> a[1]:=20.0;
```

Warning: in the loop that follows, it is essential that we use floating point (decimal) arithmetic rather than exact arithmetic. The loop depends on roundoff error to know when to stop! If we do not use decimals, then the loop will run forever (or until your machine crashes, whichever is first). Therefore, decimal expressions have been used for the initial values a_0 and a_1.

```
> for n from 1 while evalb(a[n]=a[n-1])=false do
>     a[n+1]:=N(a[n]):
> od;
```

If we desire more or less accuracy in our answer, we can adjust the number of significant digits used in floating point arithmetic.

```
> Digits:= 15;
> for n from 1 while evalb(a[n]=a[n-1])=false do
>     a[n+1]:=N(a[n]):
> od;
```

Solving Equations Iteratively

Equations of the form $\mathrm{F}(x) = \mathrm{G}(x)$ can be solved with Newton's Method by setting $\mathrm{f}(x) = \mathrm{F}(x) - \mathrm{G}(x)$ and solving $\mathrm{f}(x) = 0$.

Exercises

Solve the following equations to the indicated degree of accuracy.

5. $\mathbf{e}^x = \sin(x)$. Solve for 10 digits of accuracy.
6. $x^{\left(\frac{1}{x}\right)} = \ln(x)$. Solve for 7 digits of accuracy.
7. $x^x = \sin\left(\frac{1}{x}\right)$. Solve for 15 digits of accuracy. Find a second root to the same degree of accuracy.

A *fixed point* of a function $y = \mathrm{f}(x)$ is a value of x for which $\mathrm{f}(x) = x$. Use Newton's Method to find the smallest positive fixed point of each of the following functions. (To check that your fixed point is the smallest, look at the graph of the function.)

8. $\mathrm{f}(x) = \cos(x)$.
9. $\mathrm{f}(x) = \mathbf{e}^{(-x)}$.
10. $\mathrm{f}(x) = \dfrac{1}{(x^2+1)^2}$.

It Doesn't Always Work

If you think about the book's explanation of why Newton's Method works, you will be able to imagine examples of functions where the method fails to locate a root of $\mathrm{f}(x) = 0$.

```
> restart;
> f := x -> abs(x)^(1/3);
> N := x -> x - f(x)/D(f)(x);
> a[0] := 5;
> a[1] := 1;
> for n from 1 to 25 do
>     a[n+1]:=N(a[n]):
> od;
```

You can see from the following plot (or from a hand calculation) that the only root of this function is $x = 0$. But

Newton's Method does not converge to zero if you start with a nonzero initial value. In fact, for this particular function, whichever number you take as your initial value, Newton's Method produces a value twice as far from the root as the value you started with!

```
> plot(f,-2..2,scaling=constrained);
>
```

Exercises

For each of the following examples, try to locate a value $a_1 = a$ for which Newton's Method seems to converge to a root of $\mathrm{f}(x) = 0$. Then find a value $a_1 = b$ for which Newton's Method seems not to converge.

11. $\mathrm{f}(x) = 27\,x^6 - 120\,x^4 + 160\,x^2 - 63$.

12. $\mathrm{g}(x) = |\sin(x)|^{\left|\frac{1}{2} - x\right|}$.