Linear Approximation and Newton’s Method Worksheet

A very famous and powerful application of the tangent line approximation idea is Newton’s Method for finding approximations of roots of equations. Say we want to find a solution to an equation

\[ f(x) = 0. \]

So, we want a value, \( r \), such that \( f(r) = 0 \). If the function \( f \) is not of a rather particular type, such as linear or quadratic, we generally would have a hard time finding \( r \). In such cases, we often resort to finding an approximation of \( r \) using Newton’s Method, which is based on the following idea.

If we are looking for a root \( r \), we might start with a value \( x = a \) as an estimate of \( r \). We then improve the estimate by using the linear approximation of \( f(x) \) at \( a \), and finding the root of the linear approximation. This gives us a new approximation \( b \), which, in many cases will be a better estimate than \( a \).

The linear approximation of \( f(x) \) at \( x = a \) is

\[ L(x) = f(a) + f'(a)(x - a), \]

and if we set this equal to zero and solve for \( x \) we find

\[ b = a - \frac{f(a)}{f'(a)}. \]

The real power of the method, though, comes from the idea that if \( b \) is a better estimate for \( r \), we can repeat the method starting at \( b \), and get a new, even better estimate, and we can keep repeating this process as long as we want.

Here’s the formulation of the method.

1. Start with an estimate (i.e., a guess) of \( r \). Let’s call that guess \( r_1 \).

2. Create the recursive formula:

\[ r_{n+1} = r_n - \frac{f(r_n)}{f'(r_n)}. \]

3. Use the formula repeatedly, to generate \( r_2, r_3, r_4, \ldots \), until the values you get don’t change much (or you come to the conclusion that this method is not working).
For instance, suppose we want a root of the equation \( x^2 - 2 = 0 \). We could solve this algebraically, but for the sake of example, let’s see what Newton’s Method does with it.

Say we start with the guess \( r_1 = 1.5 \). Our recursive formula is

\[
    r_{n+1} = r_n - \frac{r_n^2 - 2}{2r_n}
\]

Plugging in \( r_1 = 1.5 \) gives us

\[
    r_2 = 1.5 - \frac{1.5^2 - 2}{2(1.5)} = 1.41666666666666666666666666666666.
\]

Plugging that into the formula, and repeating, gives us the sequence

\[
    r_3 = 1.414215686274509803921568627
\]
\[
    r_4 = 1.414213562374689910626295578
\]
\[
    r_5 = 1.414213562373095048801689623
\]
\[
    r_6 = 1.414213562373095048801688724
\]
\[
    r_7 = 1.414213562373095048801688724
\]

Since \( r_6 \) and \( r_7 \) are equal, every additional application of the formula will give the same result, so this is our best approximation of the root of the equation \( x^2 - 2 = 0 \) that we can get with this method. (This all depends as well on the accuracy of our calculating device: if your calculator presents fewer digits, you might have seen no change earlier in the sequence).

**Examples**

1. Use Newton’s method to find a solution to \( x^2 - 17 = 0 \).

2. (a) Show that when applying Newton’s method to equations of the form \( x^2 - B = 0 \), the result can be simplified to

\[
    r_{n+1} = \frac{1}{2} \left( r_n + \frac{B}{r_n} \right)
\]

(b) Use the simplified formula to find the square root of 23.

(c) What effect does using different starting guesses have?

3. Find the solution to \( \cos x = x \) (make a sketch to help you make a first guess).

4. Use Newton’s method to find \( \ln 2 \) (hint: start by finding an equation whose solution is \( \ln 2 \)). What’s a reasonable initial guess? What happens if you start with an initial guess of \(-4\)? What’s going on?

5. The equation \( x^2 = 2^x \) has two integer solutions: \( x = 2 \) and \( x = 4 \). Use Newton’s method to approximate the other solution.