Linear and Quadratic Approximation

Author	Aaron Tresham
Date	2017-06-09T19:26:24
Project	a8975d68-235e-4f21-8635-2051d699f504
Location	06 - Linear and Quadratic Approximation Assignment/Linear and Quadratic Approximation Notes.sagews
Original file	Linear and Quadratic Approximation Notes.sagews

Linear and Quadratic Approximation

It is often convenient to approximate a complicated function using a simpler function. The "simpler" function is often a polynomial. The simplest polynomial is a straight line (degree 1). Approximating a function with a linear function is called linearization (or linear approximation). Approximating a function with a degree 2 polynomial (a parabola) is called quadratic approximation. We will discuss these two approximation methods today, and we'll save approximation with higher degree polynomials for Calculus 2.

Of course, it's kind of silly to do linear or quadratic approximation with the computer, since the computer can do much more accurate approximations of these functions. The point of this lab is to see just how accurate these simple approximations can be.

Linearization

As we have seen several times this semester, differentiable functions are "close to" their tangent lines near the point of tangency. This means that we can approximate any differentiable function with a tangent line (at least over a small interval).

Let's start with a relatively simple example.

Example 1

Suppose we want to approximate $\sqrt[3]{28}$ without a computer or calculator. The cube root function is complicated (at least by hand), so we'll approximate the cube root with a simple linear function, a tangent line.

What point of tangency do we choose? There are two requirements:

1. The x-coordinate needs to be close to 28, since our approximation only works close to the point of tangency.
2. It needs to be simple to compute the function value at this x-coordinate (this is the y-coordinate of the point of tangency, and we don't want to approximate this).

For this example, we will choose the tangent line at $x = 27$. This value is fairly close to 28, and it is easy to compute $\sqrt[3]{27} = 3$ (27 is a perfect cube).

Now let's compute the tangent line and use this line to approximate $\sqrt[3]{28}$.

First, define the function we're trying to approximate:

```
f(x)=x^(1/3)  #cube root
```

Find the derivative:

```
df(x)=derivative(f,x);show(df(x))
```

$$\frac{1}{3\,x^{\frac{2}{3}}}$$

Now we need to find $f'(27)$ to get the slope of the tangent line. If $f(27)$ is simple to compute, then $f'(27)$ is probably simple to compute as well.

In this case, $f'(x) = \dfrac{1}{3x^{2/3}}$, so $f'(27) = \dfrac{1}{3\cdot 27^{2/3}} = \dfrac{1}{3\cdot 3^2} = \dfrac{1}{27}$.

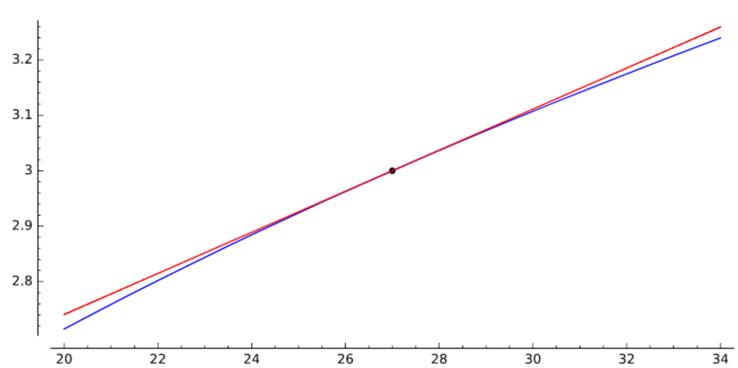
Find an equation for the tangent line: $TL(x) = f(x_0) + f'(x_0)(x - x_0)$

```
TL(x)=3+1/27*(x-27); show(TL(x))
```

$$\frac{1}{27}\,x + 2$$

Just to check our work, let's graph f and TL near $x = 27$. [Of course, if we were actually doing approximation by hand, we wouldn't do this step.]

```
f(x)=x^(1/3);TL(x)=3+1/27*(x-27)
plot(f,xmin=20,xmax=34)+plot(TL,xmin=20,xmax=34,color='red')+point((27,3),size=25,color='black')
```



As we expected, the tangent line is close to the graph of f near $x = 27$.

Now let's use the tangent line to approximate $\sqrt[3]{28}$.

The tangent line has equation $TL(x) = 3 + \frac{1}{27}(x - 27)$. Since $\sqrt[3]{x} \approx TL(x)$ (near $x = 27$), we have $\sqrt[3]{28} \approx TL(28) = 3 + \frac{1}{27}(28 - 27) = 3 + \frac{1}{27} = \frac{82}{27} \approx 3.0370$.

```
TL(28); N(_)
```

```
82/27
3.03703703703704
```

Compare this to Sage's approximation of $\sqrt[3]{28}$:

```
N(28^(1/3))
```

```
3.03658897187566
```

Our approximation is off by about $3.03703703703704 - 3.03658897187566 = 0.000448065161379851$, for a percent error of around 0.0148%.

Note: percent error $= \left| \dfrac{approximation - actual}{actual} \right| \cdot 100$

```
N(abs((TL(28)-f(28))/f(28))*100)
```

```
0.0147555420086370
```

Suppose we would like to find $\sqrt[3]{29}$ as well?

Since 29 is still fairly close to 27, we can use the same tangent line to get a linear approximation:

$\sqrt[3]{29} \approx TL(29) = 3 + \frac{1}{27}(29 - 27) = 3 + \frac{2}{27} = \frac{83}{27} \approx 3.0741$.

```
TL(29); N(_)
```

```
83/27
3.07407407407407
```

Sage gives us $\sqrt[3]{29} \approx 3.0723$.

```
N(29^(1/3))
```

```
3.07231682568585
```

This gives a percent error of about 0.0572%

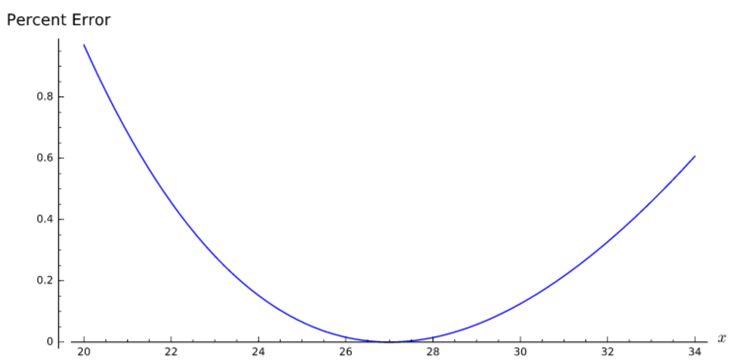
```
N(abs((TL(29)-f(29))/f(29))*100)
```

```
0.0571961971348601
```

This linear approximation is only good when x is close to 27.

Here is a graph of the percent error versus x. You can see that the error increases as you move away from 27.

```
f(x)=x^(1/3);TL(x)=3+1/27*(x-27)
plot(abs((TL(x)-f(x))/f(x))*100,xmin=20,xmax=34,axes_labels=['$x$','Percent Error'],fontsize=8)
```



Even 7 units away from 27 (in either direction), the error is less than 1%. Not bad for an approximation that you could easily do on a napkin over lunch!

Example 2 (copy and paste for linear approximation)

Now let's use linear approximation to estimate $\sin\left(\frac{\pi}{5}\right)$. This is not one of the exact values we learned in precalc, but $\frac{\pi}{5}$ is not too far from $\frac{\pi}{6}$, and we recall from precalc that $\sin\left(\frac{\pi}{6}\right) = \frac{1}{2}$. So we'll use $\left(\frac{\pi}{6}, \frac{1}{2}\right)$ as our point of tangency.

For this example, we'll let Sage do all the work, but it could be done by hand without too much trouble.

```
f(x)=sin(x)     #define the function we want to approximate
x_tangent=pi/6  #define the x-coordinate of the point of tangency
x_approx=pi/5   #define the x-value where we are trying to approximate the function

```

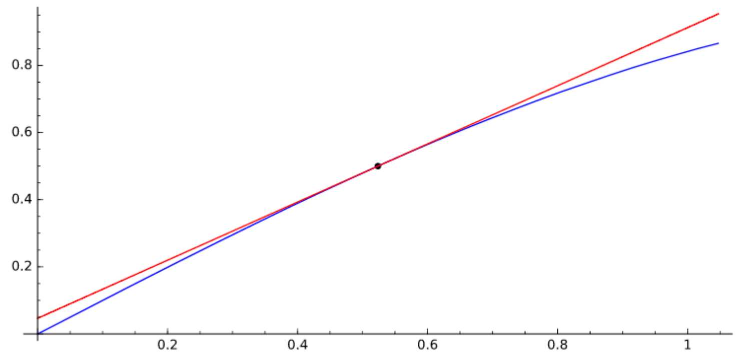
```
df(x)=derivative(f,x)
TL(x)=f(x_tangent)+df(x_tangent)*(x-x_tangent)
print 'Our approximation is:',N(TL(x_approx))
print ''
print 'Sage gives:',N(f(x_approx))
print ''
print 'The percent error is:',N(abs((TL(x_approx)-f(x_approx))/f(x_approx))*100),'%'
```

```
Our approximation is: 0.590689968211711

Sage gives: 0.587785252292473

The percent error is: 0.494179788946536 %
```

Our approximation is off by about 0.5%.

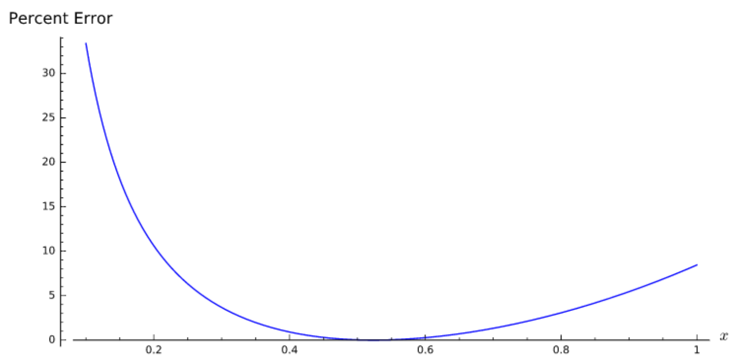
To check our work, here's a graph of f and TL near $x = \frac{\pi}{6}$.

```
f(x)=sin(x);a=pi/6;df(x)=derivative(f,x);TL(x)=f(a)+df(a)*(x-a)
plot(f,xmin=0,xmax=pi/3)+plot(TL,xmin=0,xmax=pi/3,color='red')+point((pi/6,f(pi/6)),color='black',size=25)
```



Here's a graph of the percent error. You can see that it's much worse than our cube root example above.

```
f(x)=sin(x);a=pi/6;df(x)=derivative(f,x);TL(x)=f(a)+df(a)*(x-a)
plot(abs((TL(x)-f(x))/f(x))*100,xmin=0.1,xmax=1,axes_labels=['$x$','Percent Error'],fontsize=8)
```



Just out of curiosity, would would happen if we chose the tangent line at $x = \frac{\pi}{4}$? This is close to $\frac{\pi}{5}$ and we know the exact value of $\sin\left(\frac{\pi}{4}\right)$ as well.

I'll just cut and paste the input from above. The only thing I need to change is x_tangent.

```
f(x)=sin(x)     #define the function we want to approximate
x_tangent=pi/4  #define the x-coordinate of the point of tangency
x_approx=pi/5   #define the x-value where we are trying to approximate the function

df(x)=derivative(f,x)
TL(x)=f(x_tangent)+df(x_tangent)*(x-x_tangent)
print 'Our approximation is:',N(TL(x_approx))
print ''
print 'Sage gives:',N(f(x_approx))
print ''
print 'The percent error is:',N(abs((TL(x_approx)-f(x_approx))/f(x_approx))*100),'%'
```

```
Our approximation is: 0.596034707732588

Sage gives: 0.587785252292473

The percent error is: 1.40348118771963 %
```

Now I'm off by about 1.4%. This is not surprising, since $\frac{\pi}{4} \approx 0.785$ is further away from $\frac{\pi}{5} \approx 0.628$ than $\frac{\pi}{6} \approx 0.524$.

Quadratic Approximation

Suppose we have a function $f(x)$ and its tangent line $TL(x)$ at the point $(x_0, f(x_0))$.

Recall, an equation of the tangent line is $TL(x) = f(x_0) + f'(x_0)(x - x_0)$. The derivative of TL is the constant $f'(x_0)$ (the derivative of a linear function is a constant, equal to its slope). In other words, at the point $(x_0, f(x_0))$, the functions f and TL have the same derivative (or slope). [The point of the tangent line in the first place was to find the slope of f at the point of tangency.] A linear function is completely determined by two pieces of information, in this case (1) the point of tangency, and (2) the slope.

To do quadratic approximation, we want to find a "tangent parabola" for our function f at the point $(x_0, f(x_0))$. A quadratic function is determined by *three* pieces of information. Once again we will use the point of tangency and the slope, but we need one more. The key idea is to use the second derivative. Just like f and TL have the same *first* derivative at $x = x_0$, we will choose our tangent parabola so that it has the same first *and* second derivatives as f at $x = x_0$.

Let's call our quadratic function $P(x) = A \cdot (x - x_0)^2 + B \cdot (x - x_0) + C$, for some constants A, B, and C. Our goal is to find A, B, and C so that P is "close to" f near the point $(x_0, f(x_0))$.

[Using $(x - x_0)$ instead of just x will make our calculations easier. Note the presence of $(x - x_0)$ in the tangent line formula as well.]

Note the following facts:

1. $P(x_0) = A \cdot (x_0 - x_0)^2 + B \cdot (x_0 - x_0) + C = C$
2. $P'(x) = 2A \cdot (x - x_0) + B$, so $P'(x_0) = 2A \cdot (x_0 - x_0) + B = B$
3. $P''(x) = 2A$, so $P''(x_0) = 2A$.

So, if P and f are to have the same (1) function value, (2) first derivative, and (3) second derivative at $x = x_0$, we need $C = f(x_0)$, $B = f'(x_0)$, and $A = \frac{f''(x_0)}{2}$.

Therefore, an equation for the "tangent parabola" is

$$P(x) = \frac{f''(x_0)}{2} \cdot (x - x_0)^2 + f'(x_0) \cdot (x - x_0) + f(x_0)$$

[Notice the equation for the tangent line within this formula. The tangent parabola extends the tangent line with a degree-2 term.]

Example 3

Use quadratic approximation to estimate $\sqrt[3]{28}$.

Solution: The function f we are trying to approximate is the cube root function, so $f(x) = \sqrt[3]{x}$. Just like in Example 1 above, we will use $(27, 3)$ as our point of tangency (because $\sqrt[3]{27}$ is easy to compute).

We want to find the "tangent parabola" $P(x) = A \cdot (x - x_0)^2 + B \cdot (x - x_0) + C$.

We need $C = f(27) = 3$, $B = f'(27) = \frac{1}{27}$, and $A = \frac{f''(27)}{2} = -\frac{1}{2187}$.

Thus, $P(x) = -\frac{1}{2187} \cdot (x - 27)^2 + \frac{1}{27} \cdot (x - 27) + 3$.

Now we can approximate $\sqrt[3]{28} = f(28) \approx P(28) = -\frac{1}{2187} \cdot (28 - 27)^2 + \frac{1}{27} \cdot (28 - 27) + 3 = -\frac{1}{2187} + \frac{1}{27} + 3 = \frac{6641}{2187}$

≈ 3.03657978966621

Compare this to the tangent line approximation from Example 1:

3.03703703703704

and the approximation from Sage:

3.03658897187566

This quadratic approximation gives us a percent error of about 0.0003% (compared with 0.0148% from linear approximation).

```
#Finding the coefficients:
f(x)=x^(1/3)
df(x)=derivative(f,x)
d2f(x)=derivative(f,x,2)
f(27)      #C
df(27)     #B
d2f(27)/2 #A
```

```
3
1/27
-1/2187
```

```
#Quadratic approximation
P(x)=-1/2187*(x-27)^2+1/27*(x-27)+3
P(28)
n(_)
```

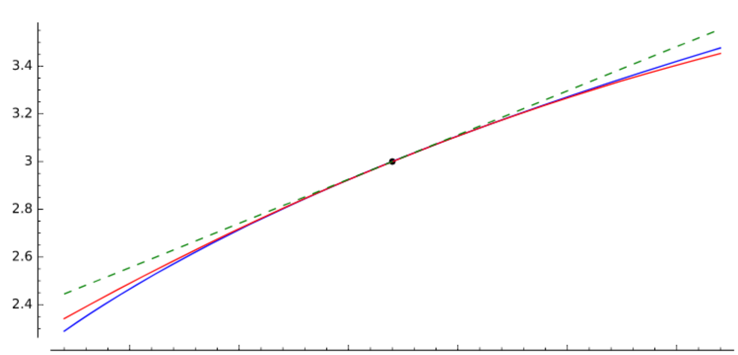
```
6641/2187
3.03657978966621
```

```
#Percent error
N(abs((P(28)-f(28))/f(28))*100)
```

```
0.000302385655022364
```

Let's look at a graph. The cube root function is in blue, the tangent parabola is in red, and the tangent line is in green.

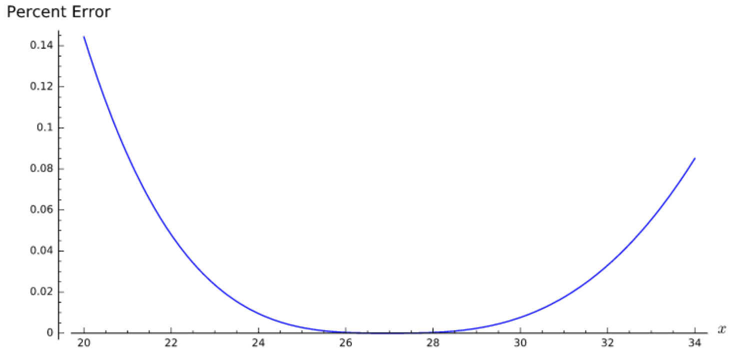
```
plot(x^(1/3),xmin=12,xmax=42)+plot(-1/2187*(x-27)^2+1/27*(x-27)+3,xmin=12,xmax=42,color='red')+point((27,3),color='black',size=25)+plot(3+1/27*(x-27),xmin=12,xmax=42,color='green',linestyle='dashed')
```



Notice how much farther we can get from 27 before the approximation breaks down (compared with the tangent line).

Below is the graph of the percent error for quadratic approximation. Remember that the tangent line had percent errors less than 1% for the same range of x-values - now we have percent errors less than 0.15%.

```
f(x)=x^(1/3);P(x)=-1/2187*(x-27)^2+1/27*(x-27)+3
plot(abs((P(x)-f(x))/f(x))*100,xmin=20,xmax=34,axes_labels=['$x$','Percent Error'],fontsize=8)
```



Example 4 (copy and paste for quadratic approximation)

Use quadratic approximation to estimate $\sin\left(\frac{\pi}{5}\right)$. Use $\left(\frac{\pi}{6}, \frac{1}{2}\right)$ as the point of tangency, as in Example 2 above.

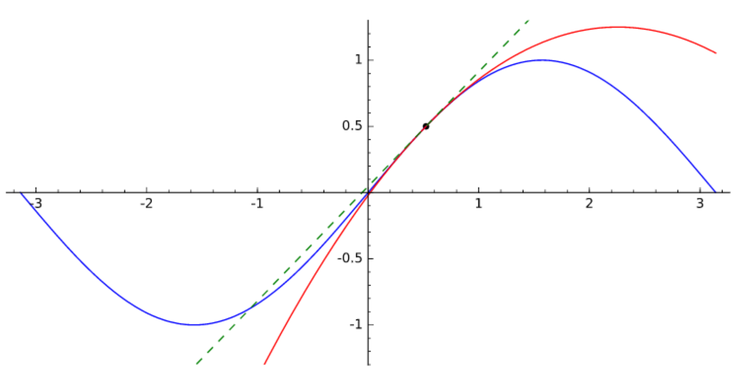
```
f(x)=sin(x)     #define the function we want to approximate
x_tangent=pi/6  #define the x-coordinate of the point of tangency
x_approx=pi/5   #define the x-value where we are trying to approximate the function

df(x)=derivative(f,x)
d2f(x)=derivative(f,x,2)
P(x)=d2f(x_tangent)/2*(x-x_tangent)^2+df(x_tangent)*(x-x_tangent)+f(x_tangent)
print 'Our approximation is:',N(P(x_approx))
print ''
print 'Sage gives:',N(f(x_approx))
print ''
print 'The percent error is:',N(abs((P(x_approx)-f(x_approx))/f(x_approx))*100),'%'
```

```
Our approximation is: 0.587948411433631

Sage gives: 0.587785252292473

The percent error is: 0.0277582910630994 %
```

Compare this percent error with the 0.494179788946525% we got from linear approximation.

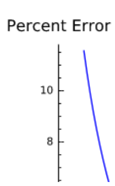
Here's a graph.

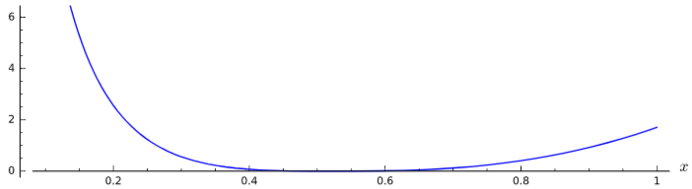
```
f(x)=sin(x);a=pi/6;df(x)=derivative(f,x);TL(x)=f(a)+df(a)*(x-a)
plot(f,xmin=-pi,xmax=pi)+plot(P,xmin=-pi,xmax=pi,ymin=-1.5,color='red')+point((pi/6,1/2),color='black',size=25)+plot(TL,xmin=-pi,xmax=pi,color='green',linestyle='dashed',ymax=1.25,ymin=-1.25)
```



Here's the percent error graph. Recall, we had errors over 30% with the tangent line.

```
plot(abs((P(x)-f(x))/f(x))*100,xmin=0.1,xmax=1,axes_labels=['$x$','Percent Error'],fontsize=8)
```





The next step would be cubic approximation. To get a cubic, we need one more piece of information. We'll make the *third* derivative of our "tangent cubic" equal to the third derivative of f at $x = x_0$. You can continue this process as many times as you like to get a polynomial of any degree. We'll come back to this in Calculus 2.