# 1170:Lab7

October 22nd, 2013

#### Goals For This Week

In this seventh lab we will explore Polynomial Approximation

- Motivation
- Approximation error
- Quadratic Approximation
- Taylor Polynomial Approximation

#### 1 Motivation

In applied mathematics we often encounter complicated functions describing our system of interest. Large and complicated functions not only make it difficult to explain the system to someone less mathematically inclined, but they also can be more challenging to work with both analytically and numerically. Polynomials on the other hand are easily accessible and are very well behaved (the are easy to differentiate, integrate, and transform). The goal then is to be able to approximate complex functions with polynomials.

## 2 Approximation Error

There are many different ways to characterize the error between a function and its approximation. For this lab however we will consider the error to be the absolute difference between the function and its polynomial approximation. For example if the function is defined as f(x) and we define the polynomial approximation to be p(x), then the error is

$$E(x) = |f(x) - p(x)|$$
 (1)

Where |.| denotes the absolute value. Defined this way the error is a function bound below by zero, and as such we could find the local maximum or minimum error by checking the critical points of our function E(x).

To see this idea in action lets consider a topic from lab 2, tangent lines and secant lines. Suppose we wish to approximate the function  $f(x) = e^x$  with a first degree polynomial (a line) near the point x=1. To do this let us pick a point near to 1, a=0.9.

Then:

$$p(x) = f'(a)(x-a) + f(a)$$
(2)

$$p(x) = f'(0.9)(x - 0.9) + f(0.9)$$
(3)

In gedit enter the following, and then copy and paste it line by line into R.

```
f<-function(x){exp(x)}
tangent<-function(x){exp(0.9)*(x-0.9)+exp(0.9)}
x<-seq(0,2,.01)
y<-seq(0.8,1.1,.001)
plot(x,f(x),type='1')
lines(x,tangent(x))</pre>
```

Notice that this is the equation of a line that passes through the point  $(0.9, e^{(0.9)})$  and is tangent to the curve.

Now let us define the error of our approximation.

```
tanerror<-function(x) {abs(f(x)-tangent(x))}
plot(x,tanerror(x),type='l')
plot(y,tanerror(y),type='l')</pre>
```

Notice that the tangent error is very good close to the point a=0.9, but not very good farther away. Lets also consider using a secant line as an approximation. The equation of a secant line S(x) is:

$$S(x) = \frac{(y^2 - y^1)}{(x^2 - x^1)} * (x - x^1) + y^1$$
(4)

where (x1,y1), (x2,y2) are the points we choose for the secant line. For example let us use  $(0.9,e^{0.9})$ ,  $(1.1,e^{1.1})$ . Then,

```
Secant<-function(x){(exp(1.1)-exp(0.9))/(1.1-0.9)*x+exp(0.9)}
Errorsecant<-function(x){abs(Secant(x)-f(x))}
plot(x,f(x),type='1')
lines(x,Secant(x))
plot(x,Errorsecant(x),type='1')
lines(x,tanerror(x))</pre>
```

Which approximation is better, the tangent line or the secant line?

## 3 Quadratic Approximation

A linear approximation matches the slope of the function at a nearby point and then extrapolates to the point of interest. We however are attempting to approximate a curve with a line, so there will be an inherent large error. By using higher derivatives of the function we can get a better approximation of the curve. The quadratic approximation to the curve is:

$$q(x) = f(a) + f'(a) * (x - a) + \frac{f''(a)}{2} * (x - a)^2$$
(5)

Now let us compare our quadratic approximation with our tangent line approximation.

q<-function(x){exp(0.9)+(x-0.9)\*exp(0.9)+1/2\*exp(0.9)\*(x-0.9)^2}
plot(x,q(x),type='1')
lines(x,f(x))
lines(x,tangent(x))</pre>

The error of our quadratic approximation is:

```
eq<-function(x){abs(q(x)-f(x)}
plot(x,eq(x),type='l')</pre>
```

Notice how the quadratic approximation is more accurate over a large range of values than the linear approximation.

#### 3.1 Taylor Polynomial Approximation

If all of the derivatives of the function f(x) are known then the infinite sum

$$T_f(x) = f(a) + f'(a) * (x - a) + \frac{(f''(a))}{2!} * (x - a)^2 + \dots + \frac{f^n(x)}{n!} * (x - a)^n + \dots$$
(6)

converges to the function f(x).

Thus any partial sum will give an approximation of the function f(x). The partial sum of the first n terms of the Taylor series is called the Taylor polynomial of degree n.

Notice that the linear approximation using the tangent line is the Taylor Polynomial of degree 1, and that the quadratic approximation is the Taylor Polynomial of degree 2.

Now let us see how well a 4th degree Taylor polynomial approximates  $e^x$ 

```
T4 <-function(x) \{ exp(0.9) + exp(0.9) * (x-0.9) + exp(0.9)/2 * (x-0.9)^2 + exp(0.9)/6 * (x-0.9)^3 + exp(0.9)/24 * (x-0.9)^4 \}
```

```
T4error<-function(x){f(x)-T4(x)}
plot(x,T4(x),type='l')
lines(x,f(x))
plot(x,T4error(x))</pre>
```

## 4 Assignment for this week

So far we have been approximating  $e^x$  with all of our polynomial approximations. For this weeks assignment use the function  $g(x) = e^{\sin(x)}$ . We would like to find an approximation of this function near the point x=2. Hint: use a point near x=2 for the approximation base point, like a=1.9, or a=2.1

- 1. Plot g(x), the linear approximation, the quadratic approximation and the 3rd degree Taylor polynomial all on the same plot from x=[1,3]
- 2. Plot the errors of the approximations over the same domain x=(1,3). Which approximation gave the least error at x=2. Which approximation did the best over the whole domain? Will this always be the case?