Approximating Polynomials

Robert A. Beezer
University of Puget Sound

February 20, 2010

Introduction

This is a short introduction to the notion of using polynomials to approximate more complicated functions. It is entirely informal, with the intent of motivating a careful study of infinite series prior to learning about Taylor polynomials and Taylor series.

1 A Very Basic Approximating Polynomial

Consider the following algebra centering on polynomial multiplication,

$$\begin{aligned}(1-x)(1+x+x^2+x^3+\cdots+x^n) &= 1+x+x^2+x^3+\cdots+x^n \\ &\quad -(x+x^2+x^3+\cdots+x^n+x^{n+1}) \\ &= 1+(x-x)+(x^2-x^2)+\cdots+(x^n-x^n)-x^{n+1} \\ &= 1-x^{n+1} \\ &\approx 1\end{aligned}$$

The approximation in the last step is valid if x^{n+1} is small, which will be the case if $-1 < x < 1$ and n is large. Keep those conditions in mind as we continue.

If we assume $x \neq 1$ and divide both sides of the above by $1-x$ we obtain

$$\frac{1}{1-x} \approx 1+x+x^2+x^3+\cdots+x^n \tag{1}$$

This will be the basis of all but one of our approximations. In the demonstration below notice the following:

- The approximation gets better as the degree, n, increases.
- No matter how large the degree is, the approximation appears limited to $-1 < x < 1$.
- For even versus odd degrees, the left end of the approximating polynomial approaches $\pm\infty$.
- The degree 1 approximation is just the tangent line at the point $(0,1)$.

```
%hide
%auto
a=-1.25
b= 0.95
original_color='blue'
approx_color='red'
@interact
def _( n = slider(0, 20, 1, 2 , label = "Degree") ):
    var('x')
    f(x)=1/(1-x)
    approx(x)=0
    for i in srange(n+1):
        approx(x)=approx(x)+x^i
    original_plot = plot( f(x), a, b ,color=original_color)
    approx_plot = plot( approx(x),  a, b, color=approx_color)
    html("Function: <font color='%s'>$%s$</font>$" % (original_color, latex(f(x))) )
    html("Approximation: <font color='%s'>$%s$</font>$" % (approx_color, latex(approx(x
    show(original_plot+approx_plot, xmin=a, xmax=b, ymin=0, ymax=10)
```

2 Approximating a Rational Function

We can use the approximation above to create an approximation of a rational function (a fraction of two polynomials). With a systematic use of partial fractions, this method can be extended to more complicated examples. Consider the following:

$$\begin{aligned}\frac{1+x^2}{1-x^2} &= \frac{1-x^2}{1-x^2}+\frac{2x^2}{1-x^2}\\ &= 1+2x^2\frac{1}{1-x^2}\end{aligned}$$

Employ equation (1) where we replace x by x^2,

$$\begin{aligned}&\approx 1+2x^2\left(1+x^2+(x^2)^2+(x^2)^3+\cdots+(x^2)^n\right)\\ &= 1+2x^2\left(1+x^2+x^4+x^6+\cdots+x^{2n}\right)\\ &= 1+2x^2+2x^4+2x^6+2x^8+\cdots+2x^{2n+2}\end{aligned}$$

Notice that our approximation should again be best when n is large and now we would require $-1 < x^2 < 1$ which simply translates back to $-1 < x < 1$. In the demonstration below we only plot the function for $-1 < x < 1$. The full graph would have vertical asymptotes at $x = -1$ and $x = 1$ and has two branches below the x-axis — one for $x < -1$ and another for $x > 1$.

```
%hide
%auto
a=-0.9999
```

```
b= 0.9999
original_color='blue'
approx_color='red'
@interact
def _( n = slider(0, 20, 2, 2 , label = "Degree") ):
    var('x')
    f(x)=(1+x^2)/(1-x^2)
    approx(x)=1
    for i in srange(2,n+1,2):
        approx(x)=approx(x)+2*x^i
    original_plot = plot( f(x), a, b ,color=original_color)
    approx_plot = plot( approx(x),  a, b, color=approx_color)
    html("Function: <font color='%s'>$%s$</font>$" % (original_color, latex(f(x))) )
    html("Approximation: <font color='%s'>$%s$</font>$" % (approx_color, latex(approx(x
    show(original_plot+approx_plot, xmin=a, xmax=b, ymin=0, ymax=10)
```

3 Approximating a Transcendental Function

We begin with equation (1), replacing x by $-t^2$, then form a definite integral that equals the inverse tangent. Again, we would expect larger values of n, with $-1 < x < 1$, to yield better approximations. First,

$$\begin{aligned}\frac{1}{1+t^2} &= \frac{1}{1-(-t^2)} \\ &\approx 1 + (-t^2) + (-t^2)^2 + (-t^2)^3 + \cdots + (-t^2)^n \\ &= 1 - t^2 + t^4 - t^6 + \cdots + (-1)^n t^{2n}\end{aligned}$$

We will now use a definite integral and the derivative of the inverse tangent in a novel way,

$$\begin{aligned}\arctan(x) &= \arctan(x) - \arctan(0) \\ &= \int_0^x \frac{1}{1+t^2} dt \\ &\approx \int_0^x 1 - t^2 + t^4 - t^6 + \cdots + (-1)^n t^{2n}\, dt \\ &= t - \frac{t^3}{3} + \frac{t^5}{5} - \frac{t^7}{7} + \cdots + \frac{(-1)^n t^{2n+1}}{2n+1}\Bigg|_0^x \\ &= x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \cdots + \frac{(-1)^n x^{2n+1}}{2n+1}\end{aligned}$$

In the demonstration below we have drawn attention to the value of the function and the approximating polynomial at $x = 1$. It is of course debatable if the approximation is even valid at $x = 1$, but we will examine this question carefully later. We know that in a 45-45-90 right triangle, the two non-hypotenuse sides are equal. Expressing angles in radians we formulate this fact as $\tan\left(\frac{\pi}{4}\right) = 1$, or equivalently, $\arctan(1) = \frac{\pi}{4}$. So if we evaluate our approximating polynomial at $x = 1$ we should get a reasonable approximation of $\frac{\pi}{4}$, and by extension, multiplying by 4 we could obtain an estimate of π. Consider that π is defined as the ratio of a circle's circumfrence to its diameter. Could we derive the necessary trigonometric facts, limits, derivatives and integrals used above without ever computing an actual value for π? I think so. Hmmmmmmmmm. The "real" value of $\frac{\pi}{4}$, and the approximation, are given below, in addition to being pinpointed by specific points on the plot.

```
%hide
%auto
a=-1.75
b= 1.75
original_color='blue'
approx_color='red'
pi_true = point((1, float(pi/4)), rgbcolor='black', pointsize=20)
@interact
def _( n = slider(1, 21, 2, 1 , label = "Degree") ):
    var('x')
    f(x)=arctan(x)
    approx(x)=0
    sign=1
    for i in srange(1,n+1,2):
        approx(x)=approx(x)+sign*x^i/i
        sign=-1*sign
    pi_approx = point( (1, float(approx(1))), rgbcolor='green', pointsize=20)
    original_plot = plot( f(x), a, b ,color=original_color)
    approx_plot = plot( approx(x),  a, b, color=approx_color)
    html("Function: <font color='%s'>$%s$</font>$" % (original_color, latex(f(x))) )
    html("Approximation: <font color='%s'>$%s$</font>$" % (approx_color, latex(approx(x
    print
    html("<font color='%s'>$\\frac{\\pi}{4}=\\arctan(1)=%s$</font>" % (original_color,
    html("<font color='%s'>$P_{%s}(1)=%s$</font>" % (approx_color, latex(n), latex(floa
    show(original_plot+approx_plot+pi_true+pi_approx, xmin=a, xmax=b, ymin=-1.5, ymax=1
```

4 An Approximation Valid Everywhere

Our previous approximating polynomials were each valid, at best, on the interval $-1 < x < 1$. We will change our approach for this final example by simply producing a very interesting polynomial and examining its properties. Recall that "n-factorial" is defined by $n! = n(n-1)(n-2)\cdots 3\cdot 2\cdot 1$, and by convention $0! = 1$. Consider the polynomials, indexed by their degree n,

$$P_n(x) = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots + \frac{x^n}{n!}$$

Each of these polynomials has a derivative, which we will compute. (Notice how fractions with factorials simplify nicely.)

$$
\begin{aligned}
P_n'(x) &= \frac{d}{dx}\left(1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \cdots + \frac{x^n}{n!}\right) \\
&= 0 + 1 + \frac{2x}{2!} + \frac{3x^2}{3!} + \frac{4x^3}{4!} + \cdots + \frac{nx^{n-1}}{n!} \\
&= 1 + \frac{2x}{2(1!)} + \frac{3x^2}{3(2!)} + \frac{4x^3}{4(3!)} + \cdots + \frac{nx^{n-1}}{n((n-1)!)} \\
&= 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots + \frac{x^{n-1}}{(n-1)!}
\end{aligned}
$$

So $P_n(x)$ and its derivative, $P_n'(x)$, are very similar, differing only in the term $\frac{x^n}{n!}$.

$$P_n(x) - P_n'(x) = \frac{x^n}{n!}$$

Even for fixed values of x greater than 1, if we let n get large enough, the denominator of this fraction will overwhelm the numerator, and this difference will be small and tend to zero. So $P_n(x)$ is very nearly equal to its derivative. Do we know any functions like this? Ah, yes, e^x! The demonstration below examines the possibility that these polynomials might be good approximations to the exponential function.

```
%hide
%auto
a=-2
b= 5
original_color='blue'
approx_color='red'
@interact
def _( n = slider(0, 10, 1, 2 , label = "Degree") ):
    var('x')
    f(x)=e^x
    approx(x)=0
    for i in srange(n+1):
        approx(x)=approx(x)+x^i/factorial(i)
    original_plot = plot( f(x), a, b ,color=original_color)
    approx_plot = plot( approx(x),  a, b, color=approx_color)
    html("Function: <font color='%s'>$%s$</font>$" % (original_color, latex(f(x))) )
    html("Approximation: <font color='%s'>$%s$</font>$" % (approx_color, latex(approx(x
    show(original_plot+approx_plot, xmin=a, xmax=b, ymin=0, ymax=100)
```