Fun with Taylor Series

Nicole Dong | Kyle Shi

Introduction

In this project, we will explore multivariable Taylor polynomials and their error bounds, and use these to approximate the area under a function.

We will use Taylor polynomials to experiment with the following functions:

$$f(x,y) = \frac{1}{1-x-y}$$

$$f(x,y) = e^{-x^2-y^2}$$

$$f(x,y) = e^{-x^2} sin(y)$$

$$f(x,y) = tan^{-1}\left(\frac{x-y}{1+xy}\right)$$

$$g(x,y) = \int_x^y e^{-t^2}\, dt$$

Section I

Consider the function $f(x,y) = \frac{1}{1-x-y}$

The partial derivatives of this function are as follows:

$f_x = \frac{1}{(1-x-y)^2}$ $\quad$ $f_y = \frac{1}{(1-x-y)^2}$

$f_{xx} = \frac{2}{(1-x-y)^3}$ $\quad$ $f_{yy} = \frac{2}{(1-x-y)^3}$ $\quad$ $f_{xy} = \frac{2}{(1-x-y)^3}$

$f_{xxx} = \frac{6}{(1-x-y)^4}$ $\quad$ $f_{yyy} = \frac{6}{(1-x-y)^4}$ $\quad$ $f_{xyy} = \frac{6}{(1-x-y)^4}$ $\quad$ $f_{xxy} = \frac{6}{(1-x-y)^4}$

The first degree (linear) Taylor polynomial of $f(x,y)$ centered at the origin is as follows:

$$f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b)$$

$$= f(0,0) + f_x(0,0)(x-0) + f_y(0,0)(y-0)$$

$$= 1 + x + y$$

The second degree (quadratic) Taylor polynomial of $f(x,y)$ centered at the origin is as follows:

$$f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b) + \tfrac{1}{2}[f_{xx}(a,b)(x-a)^2 + 2f_{xy}(a,b)(x-a)(y-b) + f_{yy}(a,b)(y-b)^2]$$
$$= 1 + x + y + \tfrac{1}{2}[\tfrac{2}{(1-x-y)^3}(0,0)(x-0)^2 + \tfrac{4}{(1-x-y)^3}(0,0)(x-0)(y-0) + \tfrac{2}{(1-x-y)^3}(0,0)(y-0)^2]$$

$$= 1 + x + y + x^2 + 2xy + y^2$$

$$= 1 + x + y + (x+y)^2$$

The third degree (cubic) Taylor polynomial of $f(x,y)$ centered at the origin is as follows:

$$f(a,b) + f_x(a,b)(x-a) + f_y(a,b)(y-b) + \tfrac{1}{2}[f_{xx}(a,b)(x-a)^2 + 2f_{xy}(a,b)(x-a)(y-b) + f_{yy}(a,b)(y-b)^2]$$
$$+ \tfrac{1}{6}[f_{xxx}(a,b)(x-a)^3 + 3f_{xxy}(a,b)(x-a)^2(y-b) + 3f_{xyy}(a,b)(x-a)(y-b)^2 + f_{yyy}(a,b)(y-b)^3]$$

$$= 1 + x + y + (x+y)^2 + x^3 + 3x^2y + 3xy^2 + y^3$$

$$= 1 + x + y + (x+y)^2 + (x+y)^3$$

Next, we will create three graphs. The first graph overlays $f(x,y)$ and the first order (linear) Taylor polynomial; the second one overlays $f(x,y)$ and the second order (quadratic) Taylor polynomial; and the third one overlays the function and the third order (cubic) Taylor polynomial

As the order of the Taylor polynomial increases, the graph of the Taylor polynomial becomes a closer approximation of $f(x,y)$. As can be seen in the following graphs, the Taylor polynomial gradually begins to look more and more similar to $f(x,y)$

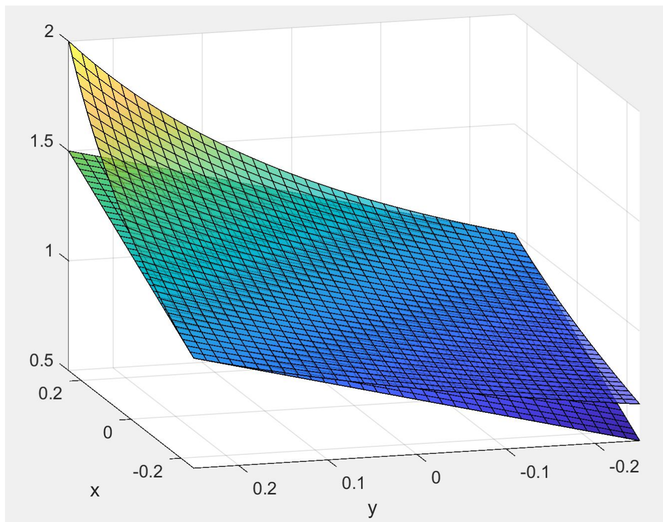


Figure 1.1 - A graph of $f(x,y)$ and the first degree (linear) Taylor polynomial $T_1(x,y)$ centered at the origin

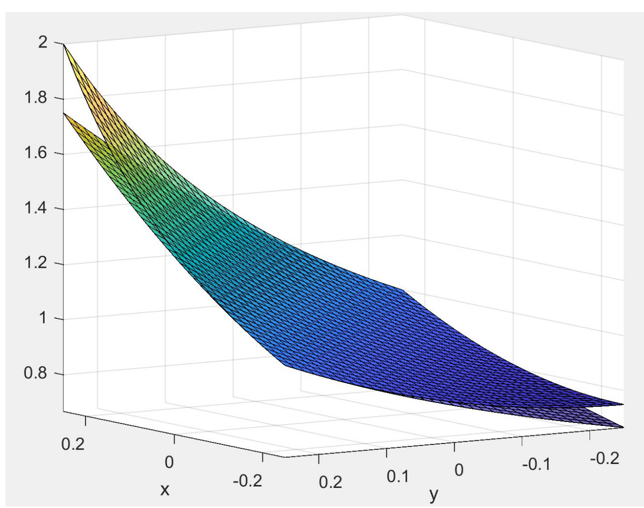


Figure 1.2 - A graph of $f(x,y)$ and the second degree (quadratic) Taylor polynomial $T_2(x,y)$ centered at the origin.

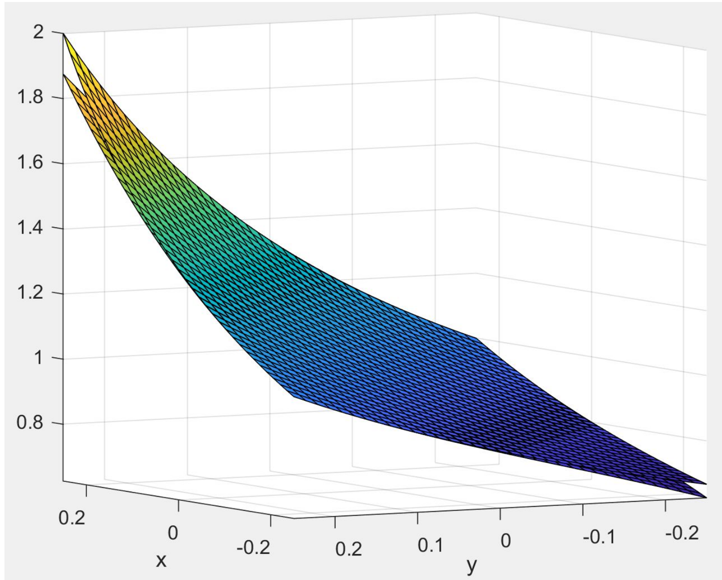


Figure 1.3 - A graph of $f(x,y)$ and the third degree (cubic) Taylor polynomial $T_3(x,y)$ centered at the origin

The next graph displays the absolute value of the actual error between the three Taylor polynomial and $f(x,y)$. As can be seen, the absolute error decreases as we approximate the function $f(x,y)$ using higher order Taylor polynomials. Thus, the cubic Taylor polynomial most faithfully represents the original function.

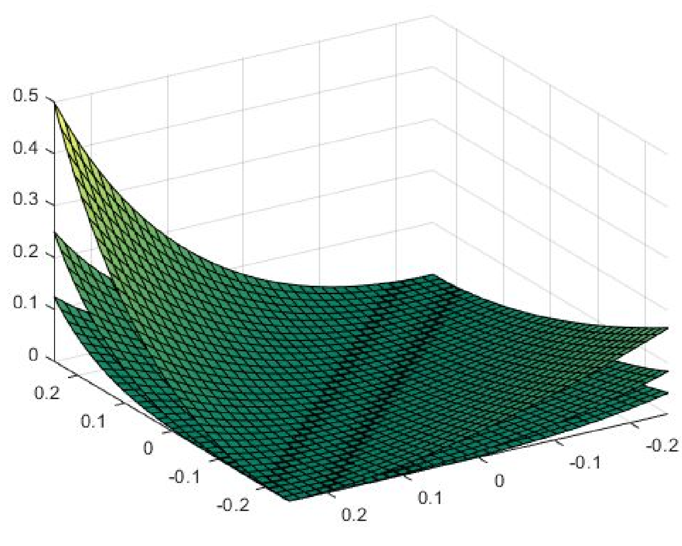


Figure 1.4 - A graph of $f(x,y)$ and the absolute value of the actual error between the three Taylor polynomials and the function ranked from top to bottom on the graph respectively: $\left|T_1(x,y)-f(x,y)\right|$, $\left|T_2(x,y)-f(x,y)\right|$, *and* $\left|T_3(x,y)-f(x,y)\right|$

Section II

Consider the function $f(x, y) = e^{-x^2-y^2}$.

First, we are going to calculate the first, second, and fourth degree Taylor polynomials of $f(x, y)$ centered at the origin $(0, 0)$.

The first degree (linear) Taylor polynomial of $f(x, y)$ centered at the origin $(0, 0)$ is as follows:

$$T_1(x, y) = 1$$

The second degree (quadratic) Taylor polynomial of $f(x, y)$ centered at the origin $(0, 0)$ is a cylinder, and is as follows:

$$T_2(x, y) = 1 - x^2 - y^2$$

The fourth degree (quartic) Taylor polynomial of $f(x, y)$ centered at the origin $(0, 0)$ is as follows:

$$T_4(x, y) = \frac{x^4}{2} + x^2y^2 - x^2 + \frac{y^4}{2} - y^2 + 1$$

The graphs of the above Taylor polynomials can be seen below:

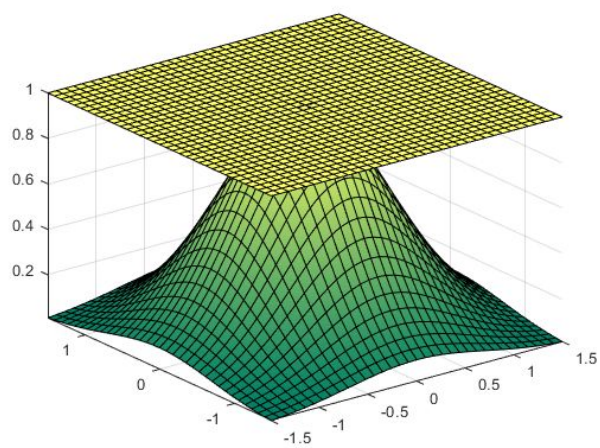


Figure 2.1 - A graph of $f(x, y)$ and the first degree (linear) Taylor polynomial $T_1(x, y)$ centered at the origin

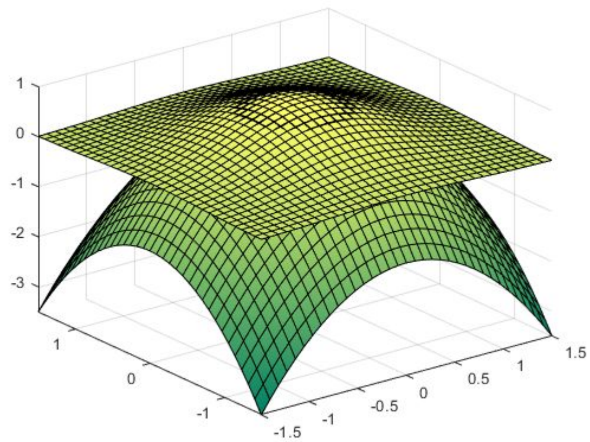


Figure 2.2 - A graph of $f(x,y)$ and the second degree (quadratic) Taylor polynomial $T_2(x,y)$ centered at the origin. This Taylor polynomial is a cylinder.

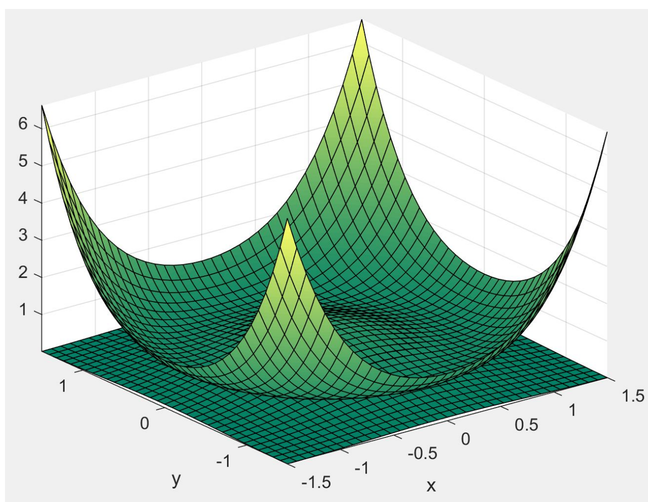


Figure 2.3 - A graph of $f(x,y)$ and the fourth degree (quartic) Taylor polynomial $T_4(x,y)$ centered at the origin

Next, we will calculate and graph the second degree (quadratic) Taylor polynomial of $f(x, y)$ centered at the point $(-0.3, -0.6)$. This surface is also a cylinder, and is as follows:

$$T_2(x, y) = 0.38x + 0.77y + \frac{1}{2}(y + 0.6)(0.92x + 0.28) - 0.52(x + 0.3)^2 - 0.18(y + 0.8)^2 + 1.21$$

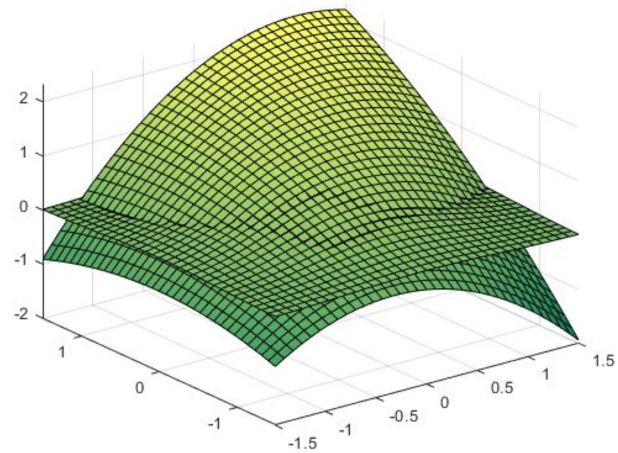


Figure 2.4 - A graph of $f(x, y)$ and the second degree (quadratic) Taylor polynomial $T_2(x, y)$ centered at the point $(-0.3, -0.6)$. The Taylor Polynomial has a quadric surface that resembles a parabolic cylinder.

Section III

Consider the function $f(x, y) = e^{-x^2} sin(y)$

The first degree (linear) Taylor polynomial of $f(x, y)$ centered at the origin $(0, 0)$ is as follows:

$$T_1(x, y) = y$$

The upper bound of the error, *M*, is determined by the maximum values for *fxx, fyy, and fxy*. The function that has the greatest max on the bound $-1 \leq x \leq 1, -1 \leq y \leq 1$ will be the value of *M.*

Thus, we will compute the maximum values of *fxx, fyy,* and *fxy* over the given interval.

$$|E(x, y)| = \frac{M}{2}[|x - a| + |y - b|]^2$$

$$M = max\{|f_{xx}|, |f_{xy}|, |f_{yy}|\}$$

$$f_{yy} = -e^{-x^2} siny$$

$$f_{xy} = -2xe^{-x^2} cosy$$

$$f_{xx} = -2e^{-x^2} siny + 4x^2 siny$$

The following graphs represent *fxx, fyy,* and *fxy:*

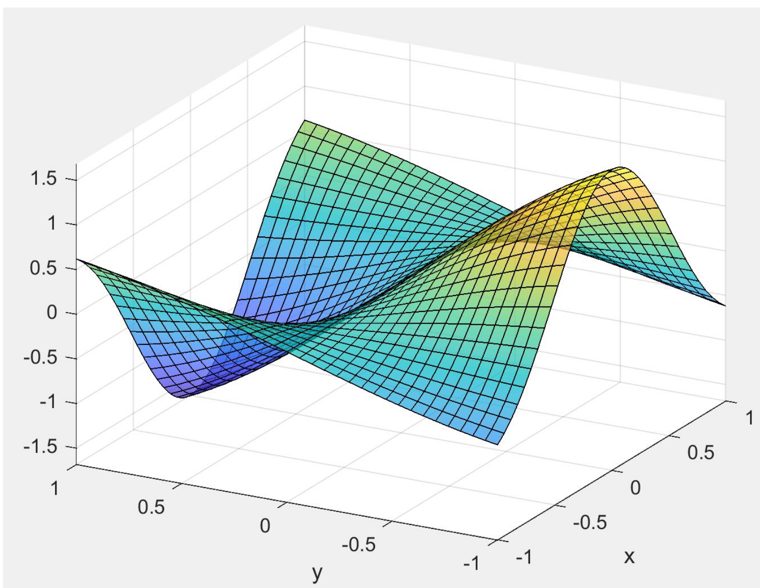


Figure 3.1 - Graph of f_{xx}

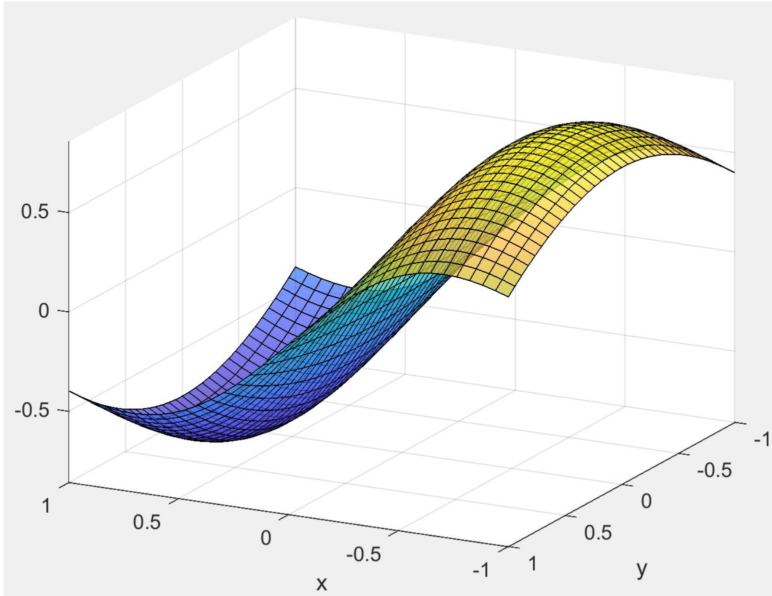


Figure 3.2 - Graph of f_{xy}

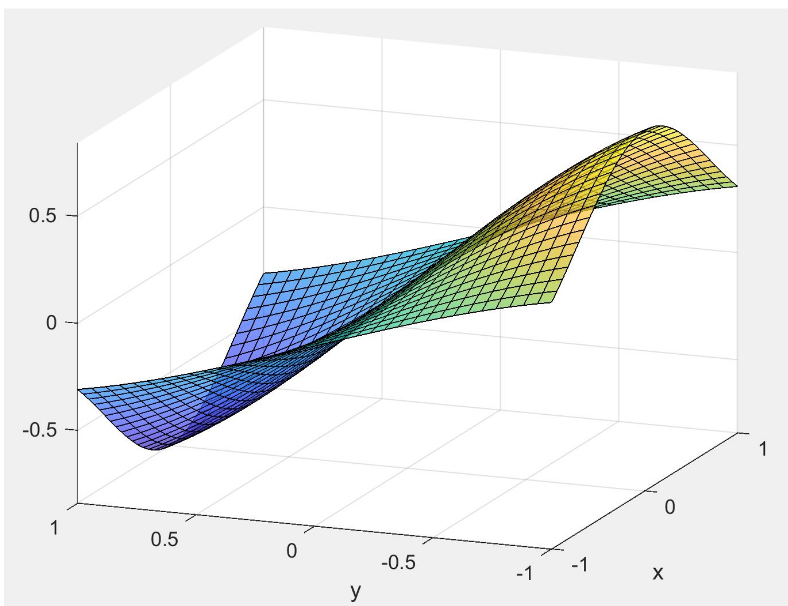


Figure 3.3 - Graph of f_{yy}

We see that *fxx* reaches the largest *z* value on the given interval. Thus, the error bound, *M*, is equal to the peak of *fxx* on its interval. We will first find the critical points of fxx on the interval.

Assume the following equation:

$$K = f_{xx} = -2e^{-x^2} siny + 4x^2 siny$$

We find the critical points of K by finding K_x and K_y :

$$K_x = 4xe^{-x^2} siny + 8xsiny = 0$$

$$K_y =- e^{-x^2} cosy + 4x^2 cosy =0$$

So when x = 0:

$$K_x = 0$$

$$K_y = cosy = 0$$

There's no value of y on this interval that will provide us with a critical point for our graph. However, if we take a look at the graph we can see that there is a point *(a,b)* where:

$$K(a,b) > K(x,y)$$

By the Extreme Value Theorem, this point will provide us with the maximum value of K on our interval. Furthermore, we know that the x coordinate of the critical point must be 0. There are no values of y on our interval that allow either K_x , K_y to be equal to 0.

When $x = 0$:

$$K(0,y) = -siny.$$

We observe that as $\lim_{y \to -1} - siny$, *siny* keeps increasing and reaches its peak on the interval.

We know that the function *siny* reaches its maximum and minimum at $\frac{\pi}{2}$ and $\frac{-\pi}{2}$.

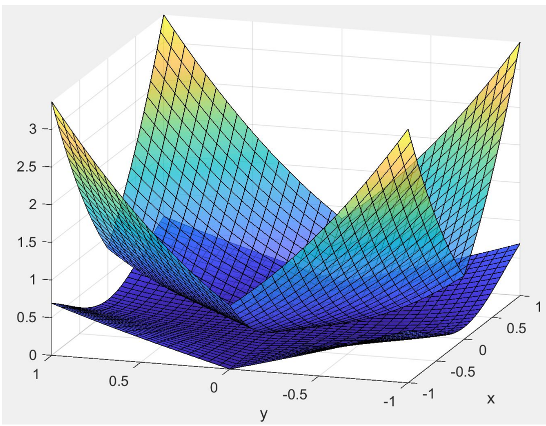
Our graph is bounded by $-1 \leq y \leq 1$, so the closest *y* values to $\frac{\pi}{2}$ and $\frac{-\pi}{2}$ we can get are 1 and -1.

At *K(0,-1)* we find that the maximum of our graph in this region is 1.683; this value is the value for our error bound, *M*.

In addition to the analytical techniques used, we can confirm the maximum value is indeed 1.683 through computational methods (as seen in section 4).

Given our newfound error value, we have plotted a graph of the actual error vs the error function. Where our error is $|E(x,y)| = \frac{1.683}{2}[|x| + |y|]^2$, and the actual error is $Error = |T_1 - f(x,y)|$

The graph of the actual error and the error function are related as they should be; when the actual error increases, the error function increases as well; when the actual error decreases, the error function also decreases. Most importantly, the actual error never surpasses the error function. Thus, the error function is a good approximation for the actual error of the Taylor series. The actual error is on the bottom of the error function.



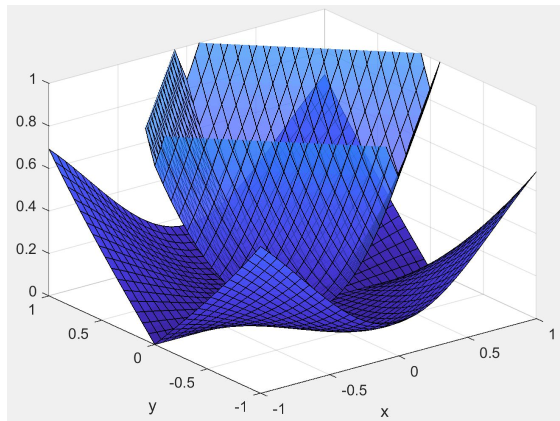


Figure 3.4 - Graph of actual error and the error function

Consider a situation where we want to find when the error of the function $T_1(x,y) = y$ is less than 10^{-4}, while it is bounded over a square region given by $|x| \leq s_0$, $|y| \leq s_0$.
$|E(x,y)| = 10^{-4}$, $M = 1.683$

To do this, we simply solve for s_0 by setting the error equal to 10^{-4} and by using our error bound.

$$|E(x,y)| = \frac{M}{2}[|x-a| + |y-b|]^2$$

The Taylor Polynomial is centered at the origin, so a and b are both 0.

$$10^{-4} = \frac{1.683}{2}[|x| + |y|]^2$$

$|x|$ and $|y|$ are both equal to s_0

$$10^{-4} = \frac{1.683}{2}[2s_0]^2$$

$$10^{-4} = \frac{1.683}{2}[2s_0]^2$$

$$s_0 = .00545$$

Section IV

Consider the function $f(x,y) = tan^{-1}(\frac{x-y}{1+xy})$

Next, we will find the first, second, and third order Taylor polynomials for $f(x,y)$ centered at the origin. We will plot these three surfaces using the domain $|x| \leq 0.9$, $|y| \leq 0.9$. A blue dot is plotted on the following surfaces at the point of tangency.

The first degree (linear) Taylor polynomial of $f(x,y)$ centered at the origin $(0,0)$ is as follows:

$$T_1(x,y) = x - y$$

The second degree (quadratic) Taylor polynomial of $f(x,y)$ centered at the origin $(0,0)$ is as follows:

$$T_2(x,y) = x - y$$

The third degree (cubic) Taylor polynomial of $f(x,y)$ centered at the origin $(0,0)$ is as follows:

$$T_3(x,y) = -\tfrac{1}{3}x^3 + x + \tfrac{1}{3}y^3 - y$$

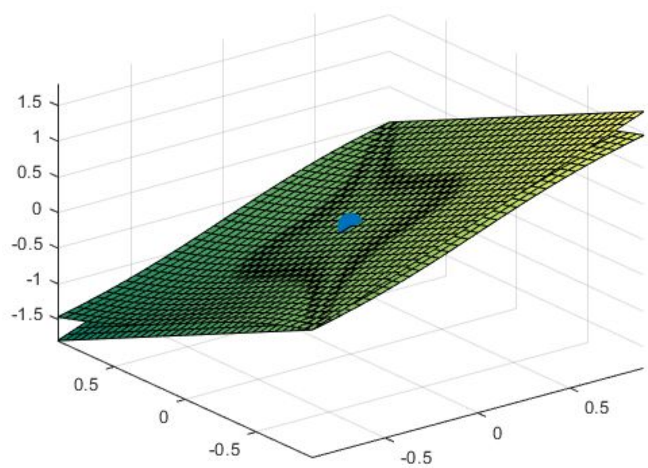


Figure 4.1 - A graph of $f(x,y)$ and the first degree (linear) Taylor polynomial $T_1(x,y)$ centered at the origin

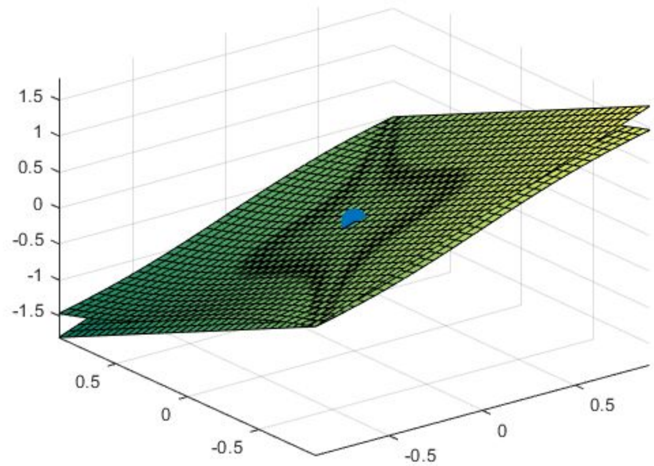


Figure 4.2 - A graph of $f(x,y)$ and the second degree (quadratic) Taylor polynomial $T_2(x,y)$ centered at the origin

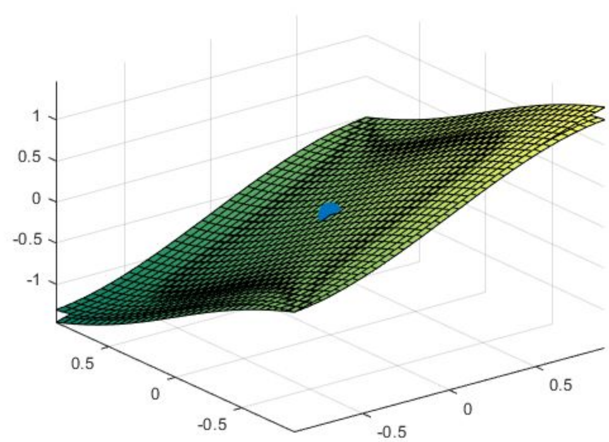


Figure 4.3 - A graph of $f(x,y)$ and the third degree (cubic) Taylor polynomial $T_3(x,y)$ centered at the origin

Next, we will calculate and graph the first, second, and third degree Taylor polynomials of $f(x,y)$ centered at the point $(3, \frac{1}{2})$. We will plot these three surfaces using the domain $|x-3| \leq 3$, $\left|y - \frac{1}{2}\right| \leq \frac{1}{2}$. A blue dot is plotted on the following surfaces at the point of tangency.

The first degree (linear) Taylor polynomial of $f(x,y)$ centered at the point $(3, \frac{1}{2})$ is as follows:

$$T_1(x,y) = 0.1x - 0.8y + 0.885$$

The second degree (quadratic) Taylor polynomial of $f(x,y)$ centered at the origin $(3, \frac{1}{2})$ is as follows:

$$T_2(x,y) = 0.1x - 0.8y - 0.03(x-3)^2 + 0.32(y-0.5)^2 + 0.885$$

The third degree (cubic) Taylor polynomial of $f(x,y)$ centered at the origin $(3, \frac{1}{2})$ is as follows:

$$T_3(x,y) = 0.1x - 0.8y - 0.03(x-3)^2 + 0.00866(x-3)^3 + 0.32(y-0.5)^2 + 0.0426(y-0.5)^3 + 0.885$$

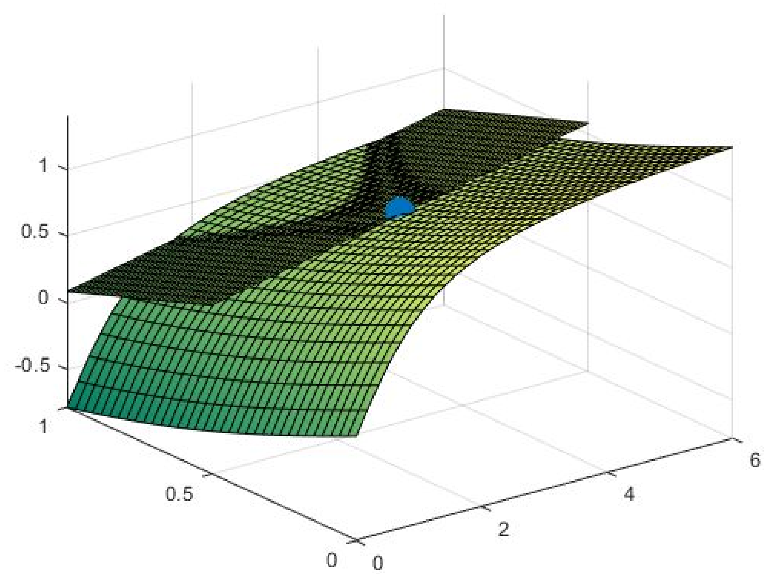


Figure 4.4 - A graph of $f(x,y)$ and the first degree (linear) Taylor polynomial $T_1(x,y)$ centered at the point (3, $\frac{1}{2}$)

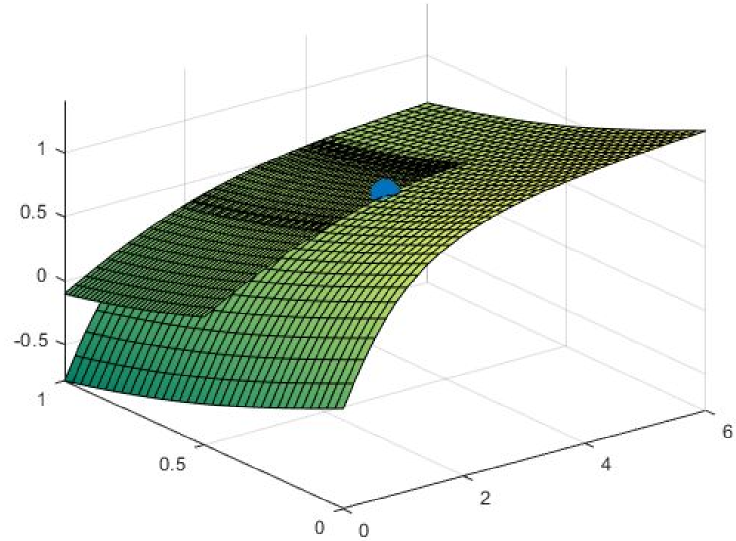


Figure 4.5 - A graph of $f(x,y)$ and the second degree (quadratic) Taylor polynomial $T_2(x,y)$ centered at the point (3, $\frac{1}{2}$)

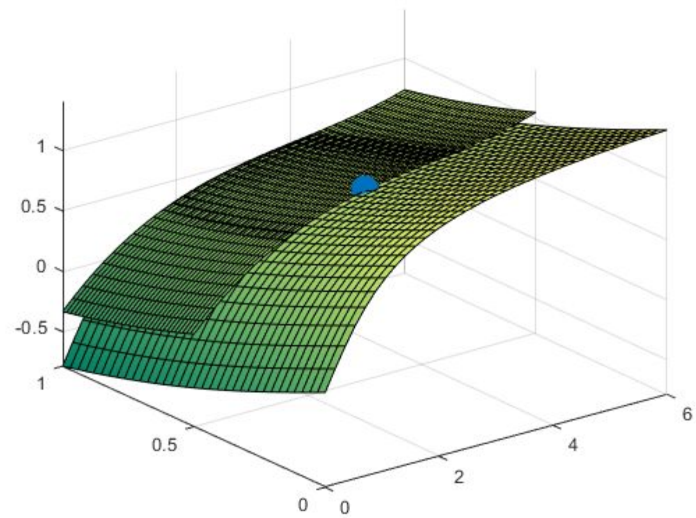


Figure 4.6 - A graph of $f(x,y)$ and the third degree (cubic) Taylor polynomial $T_3(x,y)$ centered at the point (3, $\frac{1}{2}$)

Through computational techniques we find that the upper error bound for the cubic taylor polynomial is equal to the maximum possible value for the fourth order derivatives on our interval. We find the upper error bound to be M = 4.6416. Given this value we are able to plot $|E(x,y)|$ and $|T_3 - f|$.

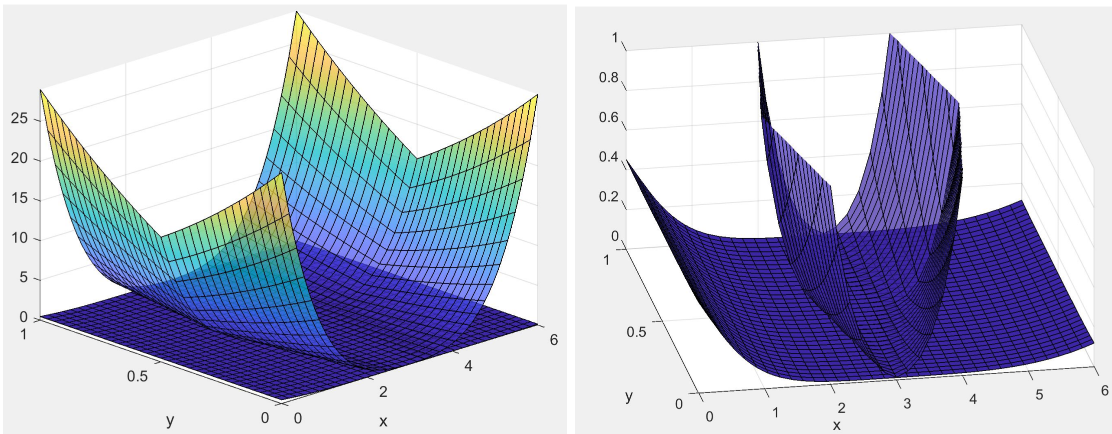


Figure 4.7 - Graph of the actual error $|T_3(x,y) - f(x,y)|$ *and the error approximation*
$|E(x,y)| = \frac{4.6416}{24}[|x-3| + |y-.5|]^2$

The graph of the actual error and the error function match with reality; as can be seen, the actual error never surpasses the error function. Thus, the error function is a good approximation for the actual error of the Taylor series. The error function is the parabola shaped surface, whereas the actual error is the flatter sheet on the bottom.

Section V

Consider the area A under the graph of e^{-x^2} from $-a$ to a, assuming that $a = 0.25$. The actual value of A is 0.489775774360512. That is, we want to evaluate:

$$A = \int_{-a}^{a} e^{-x^2} dx$$

We will use Taylor's formula in two variables, centered at (0,0), to approximate the function:

$$g(x,y) = \int_{x}^{y} e^{-t^2} dt$$

The following are the first, second, third, and fourth order Taylor polynomials (the fourth order Taylor polynomial is needed to calculate the Error Bound):

$$T_1(x,y) = \int_{x}^{y} dt = y - x$$

$$T_2(x,y) = \int_{x}^{y} (1 - t^2) dt = y - \tfrac{y^3}{3} - x + \tfrac{x^3}{3}$$

$$T_3(x,y) = \int_{x}^{y} (1 - t^2) dt = y - \tfrac{y^3}{3} - x + \tfrac{x^3}{3}$$

$$T_4(x,y) = \int_{x}^{y} (1 - t^2 + \tfrac{t^4}{2}) dt = \tfrac{y^5}{10} - \tfrac{y^3}{3} + y - \tfrac{x^5}{10} + \tfrac{x^3}{3} - x$$

We can get the Approximation by substituting *(-0.25,0.25)* for *(x,y)*, and we get the Actual Error by calculating the absolute value of the difference between the actual value of A and the calculated Approximation.

The Error Bound can be calculated by taking the absolute value of the size of the first neglected Taylor polynomial term. The succeeding terms in the Taylor polynomial when we calculate *g(-0.25,0.25)* gradually decrease in magnitude, so the error is limited by the size of the first neglected term.

For instance, to calculate the error bound for $T_1(x,y) = y - x$, we look at the neglected terms in $T_2(x,y)$, which is $-\tfrac{y^3}{3} + \tfrac{x^3}{3}$. Next, we substitute *(x,y)=(-0.25,0.25)*, and get 0.01041666666 as the error bound.

Order	Polynomial	Approximation	Error Bound	Actual Error
Linear	$y - x$	0.5	0.01041666666	0.01022422563
Quadratic	$y - \tfrac{y^3}{3} - x + \tfrac{x^3}{3}$	0.48958333333	0.0001953125	0.00019244103
Cubic	$y - \tfrac{y^3}{3} - x + \tfrac{x^3}{3}$	0.48958333333	0.0001953125	0.00019244103

As can be seen in the table, the actual errors do indeed compare favorably to the theoretical errors.

Other ways to compute *A* include graphing *g(x,y)* and using Midpoint Riemann Approximations, Trapezoidal Riemann Approximations, Left-Hand Riemann Approximations, Right-Hand Riemann Approximations, or Simpsons approximations to approximate the area under the curve.

Appendix

NOTE: THE FUNCTION F HAD TO BE REDEFINED FOR THE DIFFERENT SECTIONS OF THE PROJECT TO CREATE THE APPROPRIATE GRAPHS

Computing Taylor Approximations

```
% declare variables x and y, and center (a,b)
syms x y; syms a b;

% define the function here
f = exp(-x.^2 - y.^2);
a = 0; b = 0;
F = vpa(subs(f,[x y],[a b]));

% calculate fx and fy
fx = diff(f,x); fy = diff(f,y);
FX = vpa(subs(fx,[x y],[a b]));
FY = vpa(subs(fy,[x y],[a b]));

% calculate fxy, fxx, and fyy
fxy = diff(fx,y);
fxx = diff(fx,x);
fyy = diff(fy,y);
FXY = vpa(subs(fxy,[x y],[a b]));
FXX = vpa(subs(fxx,[x y],[a b]));
FYY = vpa(subs(fyy,[x y],[a b]));

% calculate fxyy, fxxy, fxxx, and fyyy
fxyy = diff(fxy,y);
fxxy = diff(fxx,y);
fxxx = diff(fxx,x);
fyyy = diff(fyy,y);
FXYY = vpa(subs(fxyy,[x y],[a b]));
FXXY = vpa(subs(fxxy,[x y],[a b]));
FXXX = vpa(subs(fxxx,[x y],[a b]));
FYYY = vpa(subs(fyyy,[x y],[a b]));

% calculate fxyyy, fxxyy, fxxxy, fxxxx, and fyyyy
fxyyy = diff(fxyy,y);
fxxyy = diff(fxxy,y);
fxxxy = diff(fxxx,y);
fxxxx = diff(fxxx,x);
fyyyy = diff(fyyy,y);

% compute the linear approximation
T1 = F + FX*(x-a) + FY*(y-b);
T2 = T1 + (1/2) * ( FXX*(x-a).^2 + 2*FXY*(x-a)*(y-b) + FYY*(y-b).^2 );
T3 = T1 + T2 + (1/6) * ( FXXX*(x-a).^3 + 3*FXXY*(x-a).^2*(y-b) + 3*FXYY*(x-a)*(y-b).^2 + FYYY*(y-b).^3 );
T4 = T1 + T2 + T3 + (1/24) * ( FXXXX*(x-a).^4 + 4*FXXXY*(x-a).^3*(y-b) + 6*FXXYY*(x-a).^2*(y-b).^2 +
4*FXYYY*(x-a)*(y-b).^3 + FYYYY*(y-b).^4 );
```

Computing maximum values of functions

This is just an example of finding the max of a function. We set the domain using meshgrid. Then we create a new function and sub in the meshgrid bounds to evaluate the function over the interval. We then use max to find the maximum value of our function over the interval. This can be done for any function (just replace fxxxx and change the interval).

```
%fxxxx
[X,Y] = meshgrid(0:0.1:6 , 0:0.1:1);
evalfxxxx = double(subs(fxxxx, {x y}, {X Y}));
maxfxxxx = max(max(abs(evalfxxxx)));
```

Figure 1.4

```
[X,Y,Z] = meshgrid(-0.25:0.01:0.25 , -0.25:0.01:0.25, -3:0.1:3);
figure(1)

%graph error 1
fsurf(abs(T1-f), [-0.25 0.25 -0.25 0.25])
colormap(winter);
hold on

%graph error 2
fsurf(abs(T2-f), [-0.25 0.25 -0.25 0.25])
colormap(summer);
hold on

%graph error 3
fsurf(abs(T3-f), [-0.25 0.25 -0.25 0.25])
hold off
```

Figure 2.1

```
f = exp(- x.^2 - y.^2);
a = 0;
b = 0;

% graph f(x,y)
[X,Y] = meshgrid(-1.5:0.01:1.5);
figure(1)
fsurf(f, [-1.5 1.5 -1.5 1.5])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T1, [-1.5 1.5 -1.5 1.5])
colormap(summer);
```

Figure 2.2

```
f = exp(- x.^2 - y.^2);
a = 0;
b = 0;

% graph f(x,y)
[X,Y] = meshgrid(-1.5:0.01:1.5);
figure(1)
fsurf(f, [-1.5 1.5 -1.5 1.5])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T2, [-1.5 1.5 -1.5 1.5])
colormap(summer);
```

Figure 2.3

```
f = exp(- x.^2 - y.^2);
a = 0;
b = 0;

% graph f(x,y)
[X,Y] = meshgrid(-1.5:0.01:1.5);
figure(1)
fsurf(f, [-1.5 1.5 -1.5 1.5])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T4, [-1.5 1.5 -1.5 1.5])
colormap(summer);
```

Figure 2.4

```
f = exp(- x.^2 - y.^2);
a = -0.3;
b = -0.6;

% graph f(x,y)
[X,Y] = meshgrid(-1.5:0.01:1.5);
figure(1)
fsurf(f, [-1.5 1.5 -1.5 1.5])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T2, [-1.5 1.5 -1.5 1.5])
colormap(summer);
```

Figure 3.1 - 3.3

```
figure(1) %fxx
bb = fsurf(fxx, [-1 1 -1 1])
```

```
alpha(.7)
xlabel('x')
ylabel('y')

figure(2) %fxy
cc = fsurf(fxy, [-1 1 -1 1])
alpha(.7)
xlabel('x')
ylabel('y')

figure(3) %fyy
dd = fsurf(fyy, [-1 1 -1 1])
alpha(.7)
xlabel('x')
ylabel('y')
```

Figure 3.4

```
figure(1)

gt = ((1.683)/(2))*(abs(x)+abs(y))^2
aa = fsurf(gt, [-1 1 -1 1])
xlabel('x')
ylabel('y')
alpha(.7)
hold on
ft = abs(T1 - f);
graphMeBro = fsurf(ft, [-1 1 -1 1])
hold off
```

Figure 4.1

```
f = atan( (x-y) / (1+x*y) );
a = 0;
b = 0;

% graph f(x,y)
[X,Y] = meshgrid(-1.5:0.01:1.5);
figure(1)
fsurf(f, [-1.5 1.5 -1.5 1.5])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T1, [-0.9 0.9 -0.9 0.9])
colormap(summer);
hold on

% plot the point of intersection
plot3(0,0,0,'.','MarkerSize',50);
```

Figure 4.2

```
f = atan( (x-y) / (1+x*y) );
a = 0;
```

```
b = 0;

% graph f(x,y)
[X,Y] = meshgrid(-1.5:0.01:1.5);
figure(1)
fsurf(f, [-1.5 1.5 -1.5 1.5])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T2, [-0.9 0.9 -0.9 0.9])
colormap(summer);
hold on

% plot the point of intersection
plot3(0,0,0,'.','MarkerSize',50);
```

Figure 4.3

```
f = atan( (x-y) / (1+x*y) );
a = 0;
b = 0;

% graph f(x,y)
[X,Y] = meshgrid(-.9:0.005:.9);
figure(1)
fsurf(f, [-.9 .9 -.9 .9])
hold on

% graph T(x,y)
fsurf(T3, [-0.9 0.9 -0.9 0.9])
colormap(summer);

% plot the point of intersection
plot3(0,0,0,'.','MarkerSize',50);
hold off
```

Figure 4.4

```
f = atan( (x-y) / (1+x*y) );
a = 3;
b = 0.5;

% graph f(x,y)
[X,Y,Z] = meshgrid(0:0.01:6 , 0.5:0.01:1 , -1:0.01:1);
figure(1)
fsurf(f, [0 6 0.5 1])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T1, [0 6 0.5 1])
colormap(summer);
hold on
```

```
% plot the point of intersection
plot3(3,0.5,0.8,'.','MarkerSize',50);
hold off
```

Figure 4.5

```
f = atan( (x-y) / (1+x*y) );
a = 3;
b = 0.5;

% graph f(x,y)
[X,Y,Z] = meshgrid(0:0.01:6 , 0.5:0.01:1 , -1:0.01:1);
figure(1)
fsurf(f, [0 6 0.5 1])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T2, [0 6 0.5 1])
colormap(summer);
hold on

% plot the point of intersection
plot3(3,0.5,0.8,'.','MarkerSize',50);
hold off
```

Figure 4.6

```
f = atan( (x-y) / (1+x*y) );
a = 3;
b = 0.5;

% graph f(x,y)
[X,Y,Z] = meshgrid(0:0.01:6 , 0.5:0.01:1 , -1:0.01:1);
figure(1)
fsurf(f, [0 6 0.5 1])
colormap(winter);
hold on

% graph T(x,y)
fsurf(T3, [0 6 0.5 1])
colormap(summer);
hold on

% plot the point of intersection
plot3(3,0.5,0.8,'.','MarkerSize',50);
hold off
```

Figure 4.7

```
figure(1)
gt = (4.6416/(24)).*(abs(x-3)+abs(y-.5)).^4;
aa = fsurf(gt, [0 6 0 1])
```

```
zlim([0 1]);
xlabel('x')
ylabel('y')
alpha(.7)
hold on
ft = abs(T3 - f);
graphMeBro = fsurf(ft, [0 6 0 1])
hold off
```

=