Chapter 8: Performance Surfaces and Optimum Points

Brandon Morgan

1/14/2021

E8.2

We are given the following vector function

$$F(x) = e^{2x_1^2+2x_2^2+x_1-5x_2+10}$$

The second-order Taylor series approximation for a vector function, $F(x)$, is given by:

$$F(x) = F(x^*) + \nabla F(x)|_{x=x^*}(x - x^*) + \frac{1}{2}\nabla^2 F(x)|_{x=x^*}(x - x^*)^2$$

Where $\nabla F(x)$ is known as the gradient, by $[\frac{\partial}{\partial x_1}F(x)|\frac{\partial}{\partial x_2}F(x)|\dots|\frac{\partial}{\partial x_n}F(x)]$, and where $\nabla^2 F(x)$ is known as the Hessian given by equation 8.11 in book.

The gradient of our function is calcualted to be:

$$\nabla F(x) = [\ (4x_1+1)e^{2x_1^2+2x_2^2+x_1-5x_2+10} \ \ | \ \ (4x_2-5)e^{2x_1^2+2x_2^2+x_1-5x_2+10}\ ]$$

A Hessian matrix is given by:

$$\nabla^2 F(x) = \begin{bmatrix} \frac{\partial^2}{\partial x_1^2}F(x) & \frac{\partial^2}{\partial x_1 \partial x_2}F(x) \\ \frac{\partial^2}{\partial x_2 \partial x_1}F(x) & \frac{\partial^2}{\partial x_2^2}F(x) \end{bmatrix}$$

Where it can be shown from our example that

$$\frac{\partial^2}{\partial x_1^2}F(x) = 16e^{2x_1^2+2x_2^2+x_1-5y+10}x_2^2 - 40e^{2x_1^2+2x_2^2+x_1-5x_2+10}x_2 + 29e^{2x_1^2+2x_2^2+x_1-5x_2+10}$$
$$\frac{\partial^2}{\partial x_2^2}F(x) = 16e^{2x_1^2+2x_2^2+x_1-5x_2+10}x_1^2 + 8e^{2x_1^2+2x_2^2+x_1-5x_2+10}x_1 + 5e^{2x_1^2+2x_2^2+x_1-5x_2+10}$$
$$\frac{\partial^2}{\partial x_1 \partial x_2}F(x) = (4x+1)\,e^{2x^2+2y^2+x-5y+10}\,(4y-5)$$
$$\frac{\partial^2}{\partial x_2 \partial x_1}F(x) = (4y-5)\,e^{2x^2+2y^2+x-5y+10}\,(4x+1)$$

1

From the above working, we can now find our second order Taylor series approximation for $F(x)$ about the point $x = [0, 0]^t$.

We get the following working:

Vector Function

$$F[0, 0] = 22026.4658$$

Gradient

$$\frac{\partial}{\partial x_1}F([0,0]) = 22026.46579$$
$$\frac{\partial}{\partial x_2}F([0,0]) = -110132.329$$

$$\begin{bmatrix} 22026.46579 & -110132.329 \end{bmatrix}$$

Hessian

$$\frac{\partial^2}{\partial x_1^2}F([0,0]) = 638767.508$$
$$\frac{\partial^2}{\partial x_2^2}F([0,0]) = 110132.329$$
$$\frac{\partial^2}{\partial x_1 \partial x_2}F([0,0]) = -110132.329$$
$$\frac{\partial^2}{\partial x_2 \partial x_1}F([0,0]) = -110132.329$$

$$\begin{bmatrix} 638767.508 & -110132.329 \\ -110132.329 & 110132.329 \end{bmatrix}$$

Combined

Together, we get the following second order Taylor approximation from the form $F(x) = F(x^*) + \nabla F(x^*)(x - x^*) + \frac{1}{2}(x - x^*)^T \nabla^2 F(x^*)(x - x^*)^T$, where $x^* = [0,0]$:

$$F(x) = 22026.458 + \begin{bmatrix} 22026.46579 & -110132.329 \end{bmatrix}(X) + \frac{1}{2}(X)^T \begin{bmatrix} 638767.508 & -110132.329 \\ -110132.329 & 110132.329 \end{bmatrix}(X)$$

Which reduces down to:

$$F(x) = 22026.458 + 22026.46579x - 110132.329y + 0.5(638767.508x^2 - 220264.658xy + 110132.329y^2)$$

2

The stationary point of a function is found by any set of points that satisfy Eq. (8.27), $\nabla F(x)|_{x=x^*} = 0$. Thus any set of points that makes the gradient equal to zero is considered stationary.

For our second order taylor series approximation, we need to compute and find the gradient. It's gradient can be found by finding its partial derivates, as described earlier; however, this time we have our function in matrix form. Our Taylor series can be represented by $F(x) = c + d^T x + \frac{1}{2}x^T Ax$, where $c = 22026.458$, $d^T = \begin{bmatrix} 22026.46579 & -110132.329 \end{bmatrix}$, and $A = \begin{bmatrix} 638767.508 & -110132.329 \\ -110132.329 & 110132.329 \end{bmatrix}$. From Eq. 8.38, $\nabla F(x) = Ax + d$.

Therefore, we will get the following gradient:

$$\nabla F(x) = \begin{bmatrix} 638767.508 & -110132.329 \\ -110132.329 & 110132.329 \end{bmatrix} X + \begin{bmatrix} 22026.46579 \\ -110132.329 \end{bmatrix}$$

$$\nabla F(x) = \begin{bmatrix} 638767.508x_1 - 110132.329x_2 + 22026.46579 \\ -110132.329x_1 + 110132.329x_2 - 110132.329 \end{bmatrix}$$

The reduced form was given earlier as well, one could have taken the partial derivatives with respect to each value of X and computed the same answer; however, matrix differentiation is a little bit more fun!

Solving for when these two equations are set to zero will result in $x = [1/6, 7/6]$. The process can be done algebraically because the first entry contains $-110132.329x_2$ and the bottom entry contains $110132.329x_2$; thus, adding these two equations together will yield $528635.18x_1 - 88108.86 = 0$. Then one can solve for x_1 and plug it back into either equation to solve for x_2.

3

For our original function, the gradient is $\nabla F(x) = [(4x_1+1)e^{2x_1^2+2x_2^2+x_1-5x_2+10} \mid (4x_2-5)e^{2x_1^2+2x_2^2+x_1-5x_2+10}]$.

We can set these two partial derivates equal to each other and solve for x_1 and x_2. The exponential term on both sides is the same so it reduces down to $4x_1 + 1 = 4x_2 - 5 = 0$, which can be solved algebraically to find $x_1 = -0.25$ and $x_2 = 1.25$.

The stationary point for our function is $x = [-0.25, 1.25]$ as $\nabla F([-0.25, 1.25]) = [0, 0]$.

4

In a scalar function, the points that make the first derivative equal to zero were known as extrema, where furthur tests were necessary to determine if these were maxima or minima. Whenever the first derivative is equal to zero, that set of points indicate where the function has no slope.

In our multidimensional case, the first derivative is known as the gradient, where points that make it equal to zero are known as stationary points. In part 3, we found the actual stationary points of our function, the points where possible minima or maxima are observed; whereas in part 2 we found the stationary points of our second order Taylor series centered about the position $x = [0, 0]$.

Figure 1 shows our original function from bounds $x_1 = [-0.249, -0.251]$ and $x_2 = [1.249, 1.259]$. We can see that there appears to be a minima at the bottom of the sub plot, which is shown by the point. In Figure 2 we heave the same plot again but this time the view is inverted to give the reader another perspective of the graph.

Figure 3 shows our second order Taylor series function from bounds $x_1 = [0.165\bar{6}, 0.167\bar{6}]$ and $x_2 = [1.165\bar{6}, 1.167\bar{6}]$. We can see that there appears to be a minima at the floor of the sub plot, which is shown by the point. In Figure 2 we heave the same plot again but this time the view is inverted to give the reader another perspective of the graph.

The code to create these plots is down below:

Use the following libraries:

```
library(purrr) # used for map2_dbl function
library(plotly) # used for 3D plotting
```

```
## Warning: package 'plotly' was built under R version 3.6.3
```

```
## Loading required package: ggplot2
```

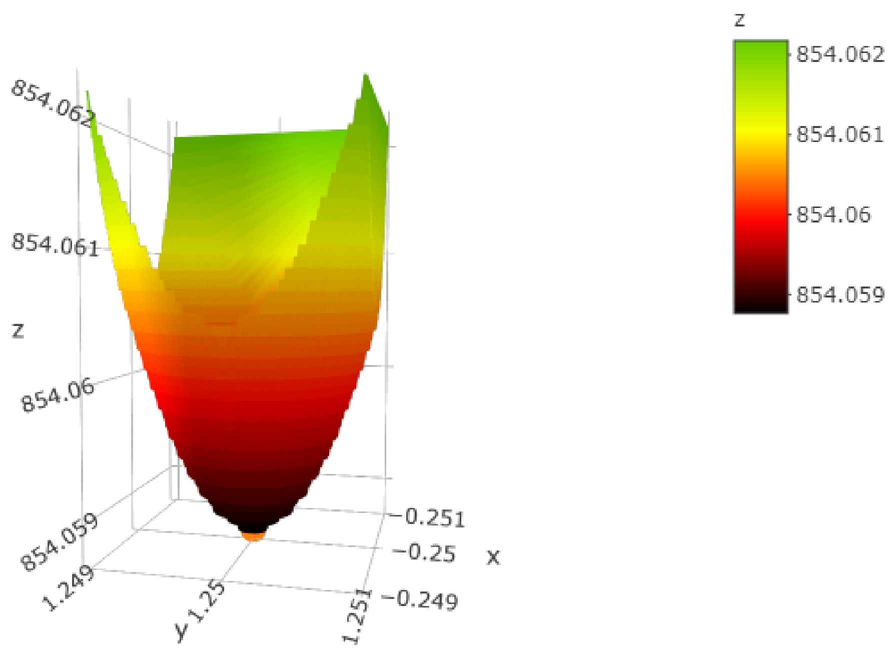

Figure 1: Original Function with stationary point

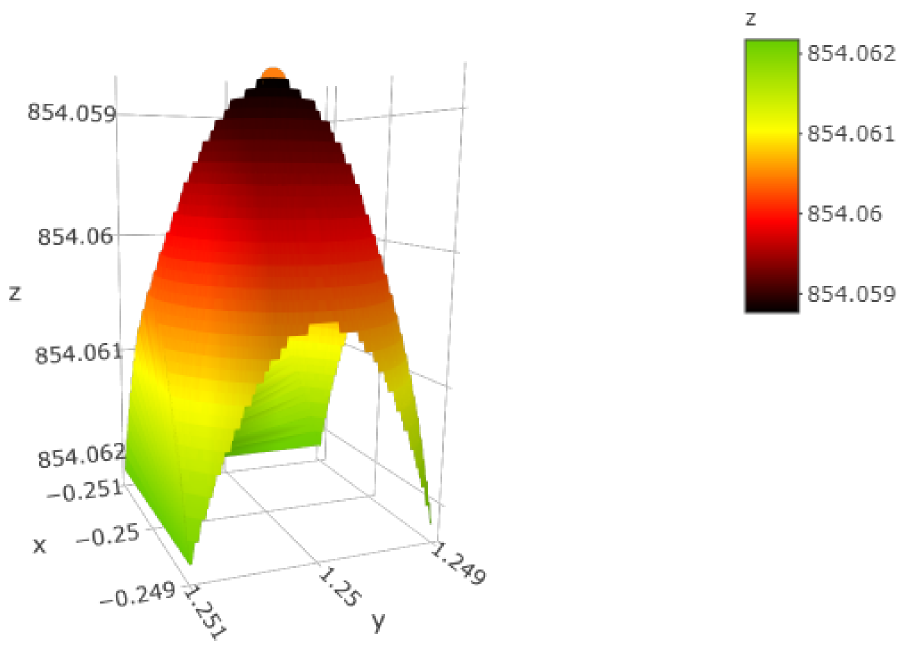

Figure 2: Original Function with stationary point (inverted view)

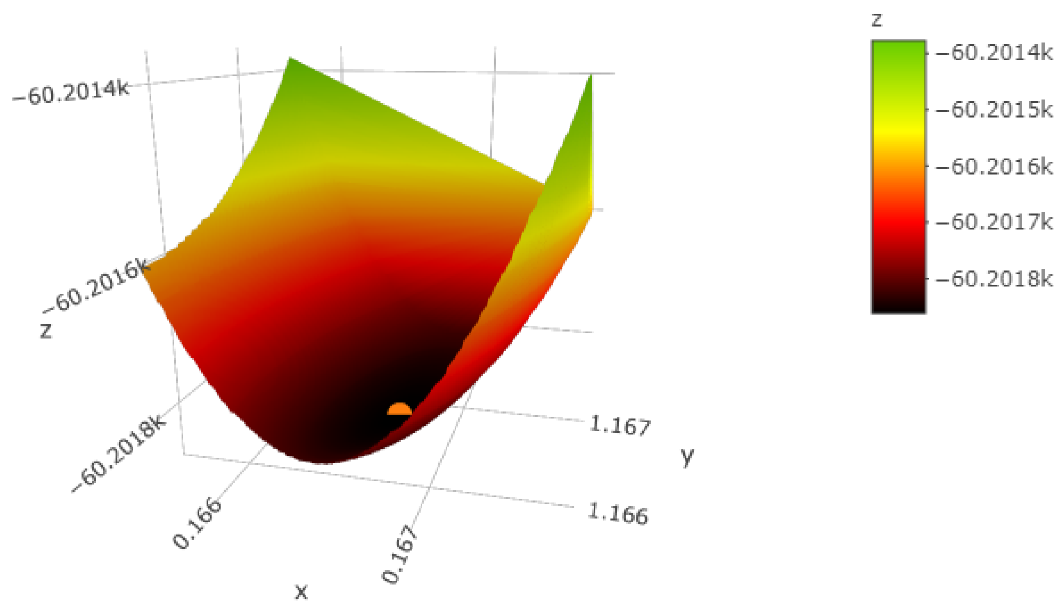

Figure 3: Taylor Series Expansion with stationary point

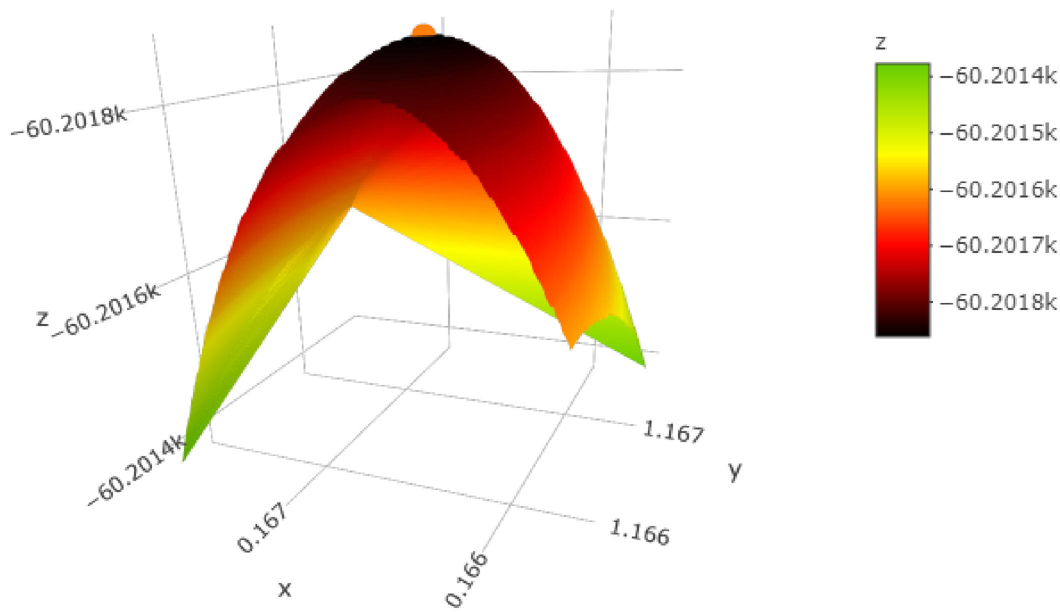

Figure 4: Taylor Series Expansion with stationary point (inverted view)

```
##
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':
##
##     last_plot

## The following object is masked from 'package:stats':
##
##     filter

## The following object is masked from 'package:graphics':
##
##     layout
```

Our functions

```r
fun = function(x, y) { # our original function
  exp(2*x^2+2*y^2+x-5*y+10)
}
```

```r
taylor = function(x, y) { # our taylor expansion
  X = matrix(c(x, y), nrow=2)
  A = matrix(c(638767.508, -110132.329, -110132.329, 110132.329), ncol=2, byrow=TRUE)
  d = matrix(c(22026.46579, -110132.329), ncol=2)
  2206.458+d%*%X+0.5*t(X)%*%A%*%X
  }
```

Code used to create the 3D plots

Original Function

```r
pointsX = seq(-0.249, -.251, length=200)# create a 200x1 vector of values for x axis
pointsY = seq(1.249, 1.251, length=200) # create a 200x1 vector of values for y axis
myGrid = expand.grid(pointsX, pointsY) # create 200x200 grid of points
z = map2_dbl(myGrid$Var1, myGrid$Var2, ~fun(.x, .y)) # maps all possible combinations
# of the grid through the function
x = myGrid$Var1
y =  myGrid$Var2
myPlot = plot_ly(x=~x, y=~y, z=~z, type="mesh3d", intensity=~z,
        colors = colorRamp(c("black","red","yellow","chartreuse3")))

myPlot = add_trace(p=myPlot, z = c(fun(-0.25, 1.25)), x=c(-0.25), y=c(1.25), type="scatter3d")
# add another layer onto the 3D rendering, in this case a scatterplot
# of a single point, the stationary point, AKA minima
#myPlot
```

Taylor Series Expansoin

```r
pointsX = seq(1/6-0.001, 1/6+0.001, length=200)# create a 200x1 vector of values for x axis
pointsY = seq(7/6-0.001, 7/6+0.001, length=200) # create a 200x1 vector of values for y axis
myGrid = expand.grid(pointsX, pointsY) # create 200x200 grid of points
z = map2_dbl(myGrid$Var1, myGrid$Var2, ~taylor(.x, .y)) # maps all possible combinations
# of the grid through the function
x = myGrid$Var1
y =  myGrid$Var2
myPlot = plot_ly(x=~x, y=~y, z=~z, type="mesh3d", intensity=~z,
        colors = colorRamp(c("black","red","yellow","chartreuse3")))

# add another layer onto the 3D rendering, in this case a scatterplot
# of a single point, the stationary point, AKA minima
myPlot = add_trace(p=myPlot, z = c(taylor(1/6, 7/6)), x=c(1/6), y=c(7/6), type="scatter3d")
# myPlot
```