# Visualization

May 16, 2018

```
In [1]: from __future__ import division
        from numpy import *
        from sympy import *
        import matplotlib.pyplot as plt
        %matplotlib notebook
        from mpl_toolkits import mplot3d
        x, y, z, t = symbols('x y z t')
        k, m, n = symbols('k m n', integer=True)
        f, g, h = symbols('f g h', cls=Function)

In [3]: def f(x, y):
             return x + y
        def c(x):
            return 0

        def d(x):
            return 1

In [4]: def Simpson_Double(a, b, m, n):

                h = (b - a)/n
                J1, J2, J3 = 0, 0, 0


                for i in range(0, n + 1):
                             x = a + i*h
                             HX = (d(x) - c(x))/m
                             k1 = f(x, c(x)) + f(x, d(x))
                             k2 = 0
                             k3 = 0

                             for j in range (1, m):
                                   y = c(x) + j*HX
                                   Q = f(x, y)

                                   if j%2 == 0:
                                       k2 = k2 + Q
```

```
                                    else:
                                        k3 = k3 + Q

                              L = (k1 + 2*k2 + 4*k3)*HX/3

                              if i == 0 or i == n:
                                      J1 = J1  + L

                              elif i%2 == 0:
                                      J2 = J2 +  L

                              else:
                                 J3 = J3 + L

                 J = h*(J1 + 2*J2 + 4*J3)/3
                 return J
```

```
In [5]: xlist = []
        ylist = []
        zlist = []
        real_z = []
        error = []
        total_error = 0
```

```
In [6]: for i in arange(0, 1, 0.05):
                for k in arange(0, 1, 0.05):

                    #redefine the y-functions.
                     def c(x):
                        return 0

                     def d(x):
                        return  k


                     z = Simpson_Double(0, i, 30, 30)
                     r_z = (i**2)*(k/2) + (k**2)*(i/2)
                     zlist = zlist + [z]
                     xlist = xlist + [i]
                     ylist = ylist + [k]
                     real_z = real_z + [r_z]
                     error = error + [abs(z - r_z)]
                     total_error = total_error + abs(z - r_z)
                    # print('xlist = %3f, ylist = %3f, actual = %.17f, approximate = %.17f'%(i
```

```python
print(total_error)
ax = plt.axes(projection='3d')
ax.set_xlabel('X')
ax.set_ylabel('Y')
ax.set_zlabel('Z')
ax.scatter3D(xlist, ylist, zlist, cmap='Greens');


plt.draw()
```

```
6.739346494061271e-15
```