assignment02

September 27, 2018

1 Assignment 02 : Python Programming - Taylor Approximation

2 Name : Seung Yeop, Seon

3 Student ID : 20144753

```
In [33]: import numpy as np
         import matplotlib.pyplot as plt
         import random
         from scipy.misc import derivative
```

4 Define my function: $f(x) = \frac{1}{1+e^{-x}}$

```
In [34]: def my_func(x):
             result = 1/(1+np.exp(-x))
             return result
```

5 Define my deravative function: $f(x) = \frac{e^{-x}}{(1+e^{-x})^2}$

```
In [35]: def deravative_func(x):
             return derivative(my_func, x, dx = 1e-5)
```

6 Define taylor approxiamtion form.

```
In [36]: def taylor_approximation(domain, point):
             result = deravative_func(point) * (domain - point) + my_func(point)
             return result
```

7 Define domain and ouput of above functions.

```
In [40]: x=np.arange(-5,5,0.1)
         y=my_func(x)
         d_y=deravative_func(x)
```

8 Randomly choose 3 points.

```
In [41]: point = np.zeros(3)

         for i in range(3):
             point[i] = random.choice(x)
             print(point[i])
```

```
4.199999999999967
3.49999999999997
2.1999999999999744
```

9 Compute the graphs

```
In [42]: plt.figure(1)
         plt.plot(x, y, 'b', label="function")
         plt.plot(x,taylor_approximation(x,point[0]), 'g', label="taylor approximation on fist p
         plt.plot(x,taylor_approximation(x,point[1]), 'y', label="taylor approximation on second
         plt.plot(x,taylor_approximation(x,point[2]), 'c', label="taylor approximation on third
         plt.legend(bbox_to_anchor=(1.05, 1), loc=2, borderaxespad=0.)
         plt.show()
```