Lab10

April 24, 2019

In [ ]:
```
'''
                                        MA374 | Lab 08
                                Deepak Kumar Gouda | 160123054
        '''
```

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
plt.rcParams['figure.figsize'] = [10, 5]
```

In [2]:
```
%matplotlib inline
```

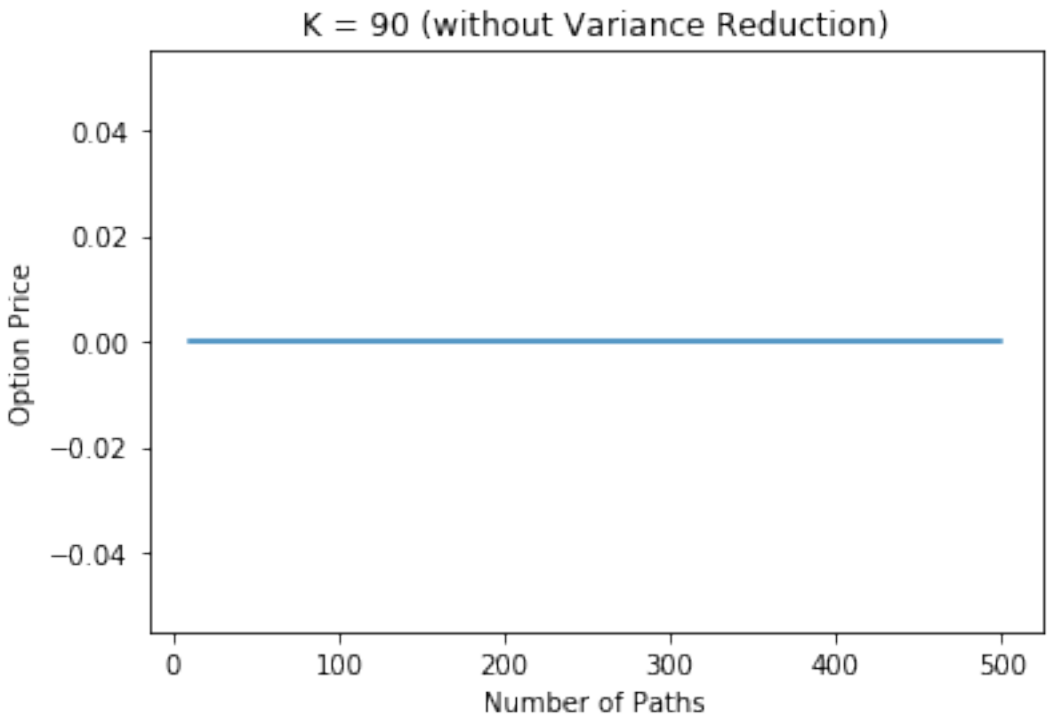
In [3]:
```
def getStockPath(S0, r, mu, sig, n, varReduction=False):
    S_plus = np.zeros(n)
    S_plus[0] = S0
    if varReduction:
        S_minus = np.zeros(n)
        S_minus[0] = S0
    dt = 1.0/365
    W = np.random.randn(n)*dt
    for i in range(1, n):
        W = dt*np.random.randn()
        S_plus[i] = S_plus[i-1]*np.exp((r-0.5*(sig**2))*dt + sig*W)
        if varReduction:
            S_minus[i] = S_minus[i-1]*np.exp((r-0.5*(sig**2))*dt - sig*W)
    if varReduction:
        return S_plus, S_minus
    else:
        return S_plus
```

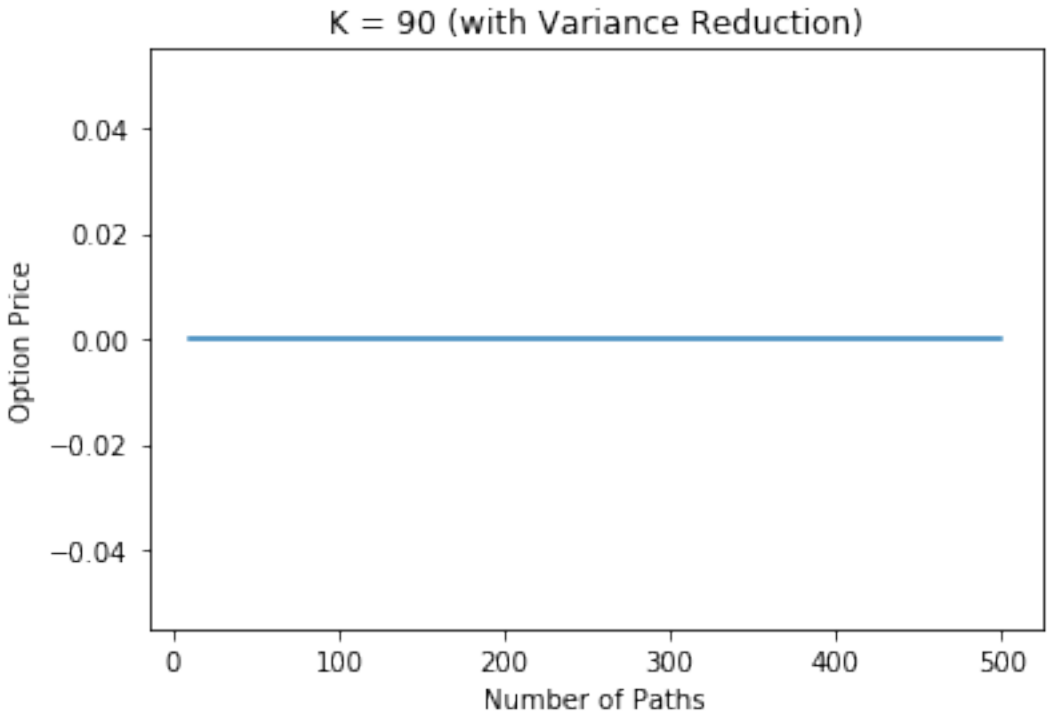
In [4]:
```
def getAsianOptionPrice(S0, r, mu, sig, n, K, varReduction=False):
    dt = 1.0/365

    maxPaths = 500
    sumPayOff = 0
    step = 10
    X = range(10, maxPaths+1, step)
    price = np.zeros(len(X))
```

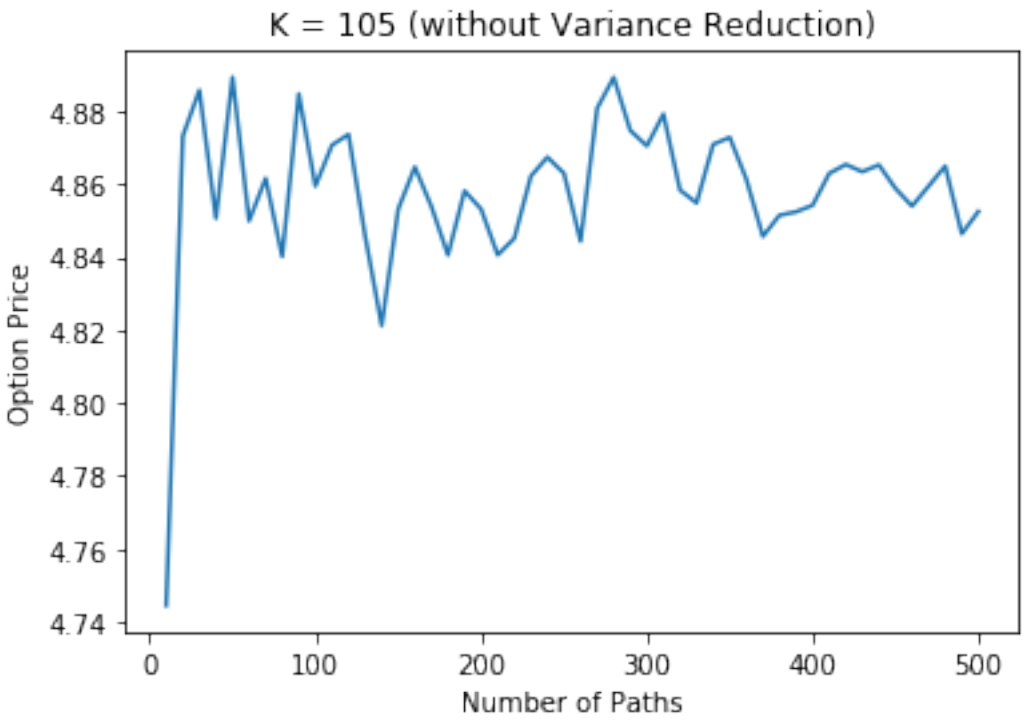
```
            for t, numPaths in enumerate(range(10, maxPaths+1, 10)):
                sumPayOff = 0
                for i in range(numPaths):
                    if varReduction:
                        S_plus, S_minus = getStockPath(S0, r, mu, sig, n, varReduction)
                        V_plus = max(K - np.mean(S_plus), 0)
                        V_minus = max(K - np.mean(S_minus), 0)
                        sumPayOff = sumPayOff+(V_plus+V_minus)*0.5
                    else:
                        S = getStockPath(S0, r, mu, sig, n, varReduction)
                        V = max(K - np.mean(S), 0)
                        sumPayOff = sumPayOff+V
                price[t] = (np.exp(-r*(n)*dt)*sumPayOff/numPaths)
            return price
```

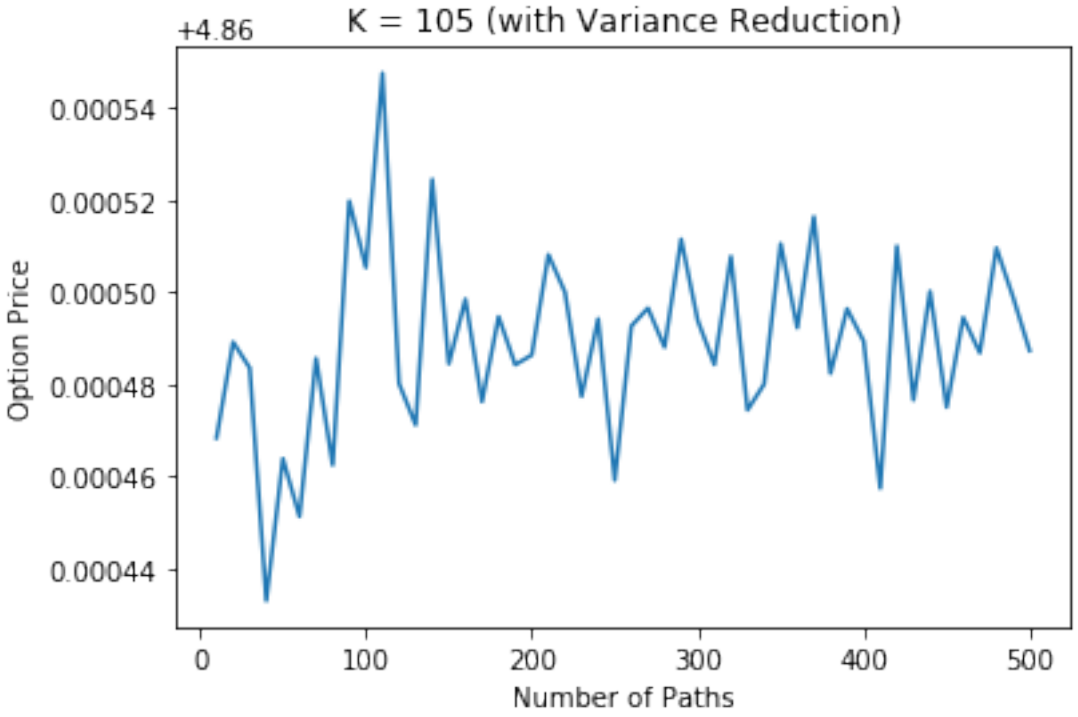
In [5]:
```
S0 = 100.0
r = 0.05
mu = 0.1
sig = 0.2
n = 30
K = 110
maxPaths=500
```

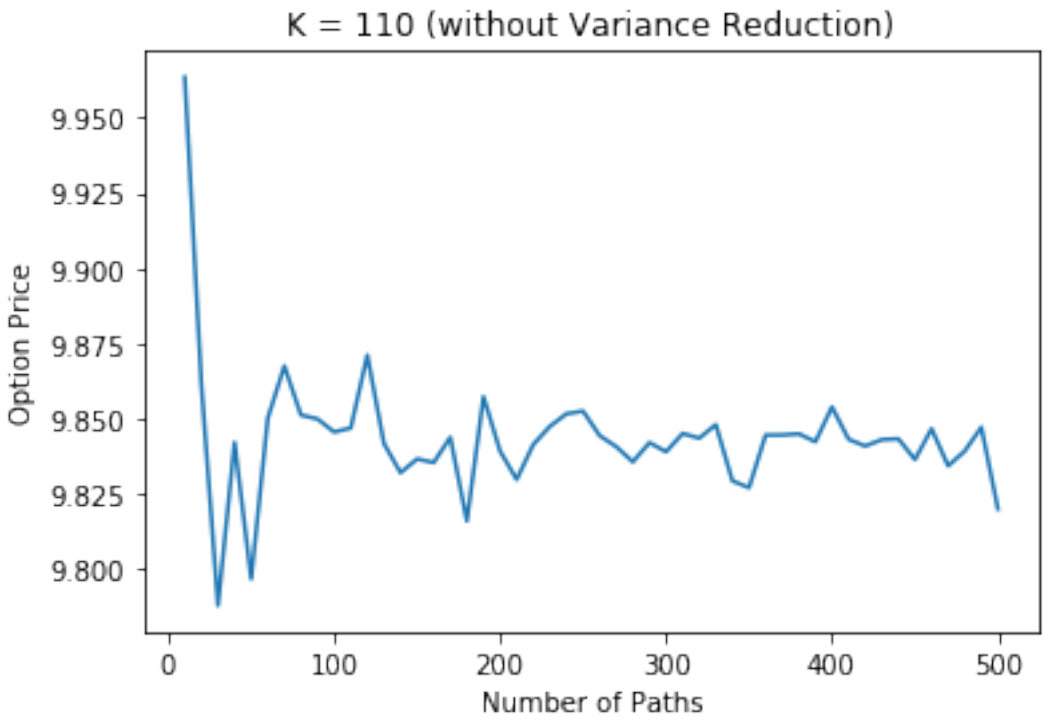
In [6]:
```
for K in [90, 105, 110]:
    # Without Variance reduction
    price = getAsianOptionPrice(S0, r, mu, sig, n, K, varReduction=False)

    X = np.arange(10, maxPaths+1, 10)
    Y = price
    plt.xlabel("Number of Paths")
    plt.ylabel("Option Price")
    plt.title("K = %d (without Variance Reduction)"%(K))
    plt.plot(X, Y)
    plt.show()

    # With Variance reduction
    price = getAsianOptionPrice(S0, r, mu, sig, n, K, varReduction=True)

    X = np.arange(10, maxPaths+1, 10)
    Y = price
    axes = plt.gca()
    if K is 110:
        axes.set_ylim([9.8375, 9.8425])
    plt.xlabel("Number of Paths")
    plt.ylabel("Option Price")
    plt.title("K = %d (with Variance Reduction)"%(K))
    plt.plot(X, Y)
    plt.show()
```

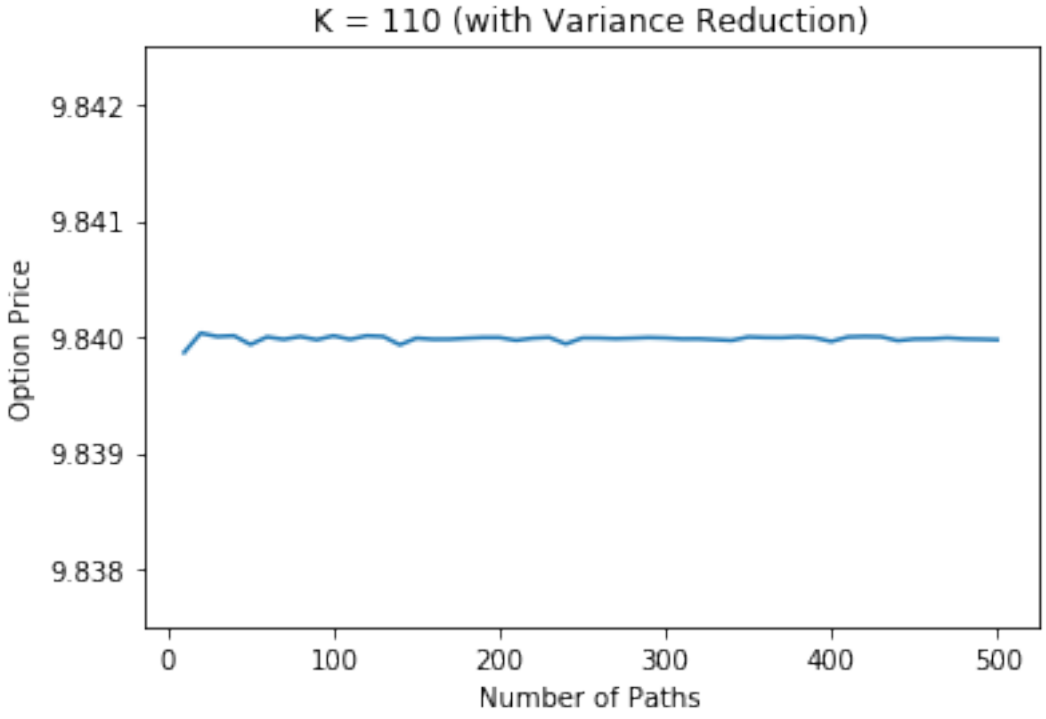

K = 90 (without Variance Reduction)
0.04
0.02
0.00
−0.02
−0.04
Option Price
0
100
200
300
400
500
Number of Paths


K = 90 (with Variance Reduction)
0.04
0.02
0.00
−0.02
−0.04
Option Price
0
100
200
300
400
500
Number of Paths


K = 105 (without Variance Reduction)
Option Price
4.88
4.86
4.84
4.82
4.80
4.78
4.76
4.74
0
100
200
300
400
500
Number of Paths

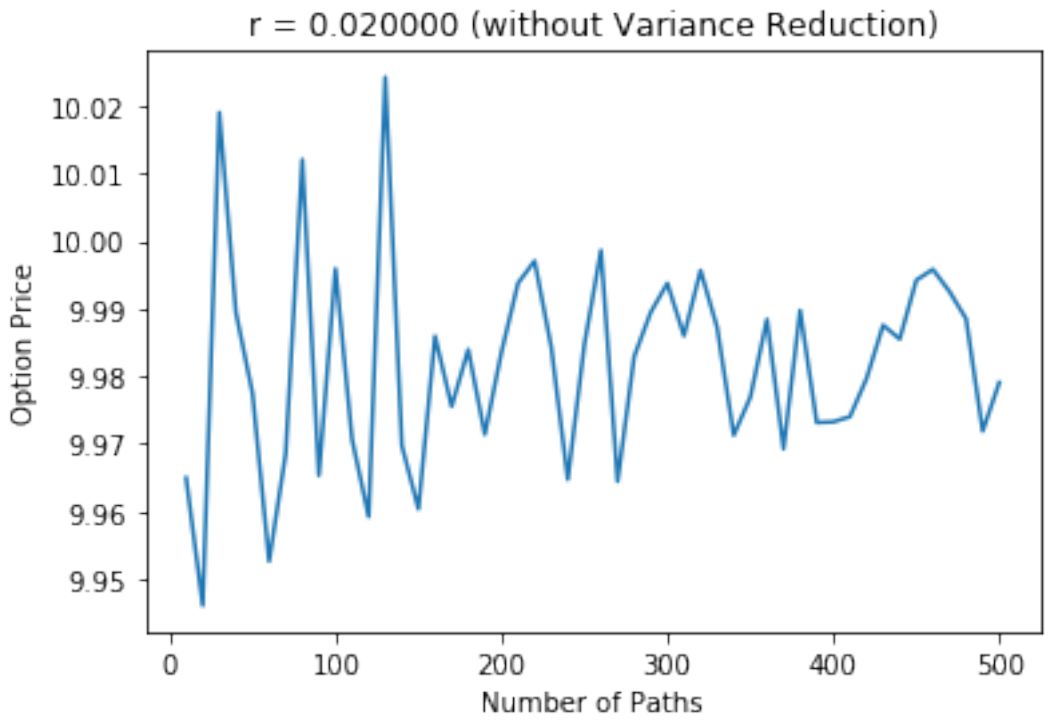

+4.86
K = 105 (with Variance Reduction)
Option Price
0.00054
0.00052
0.00050
0.00048
0.00046
0.00044
0
100
200
300
400
500
Number of Paths

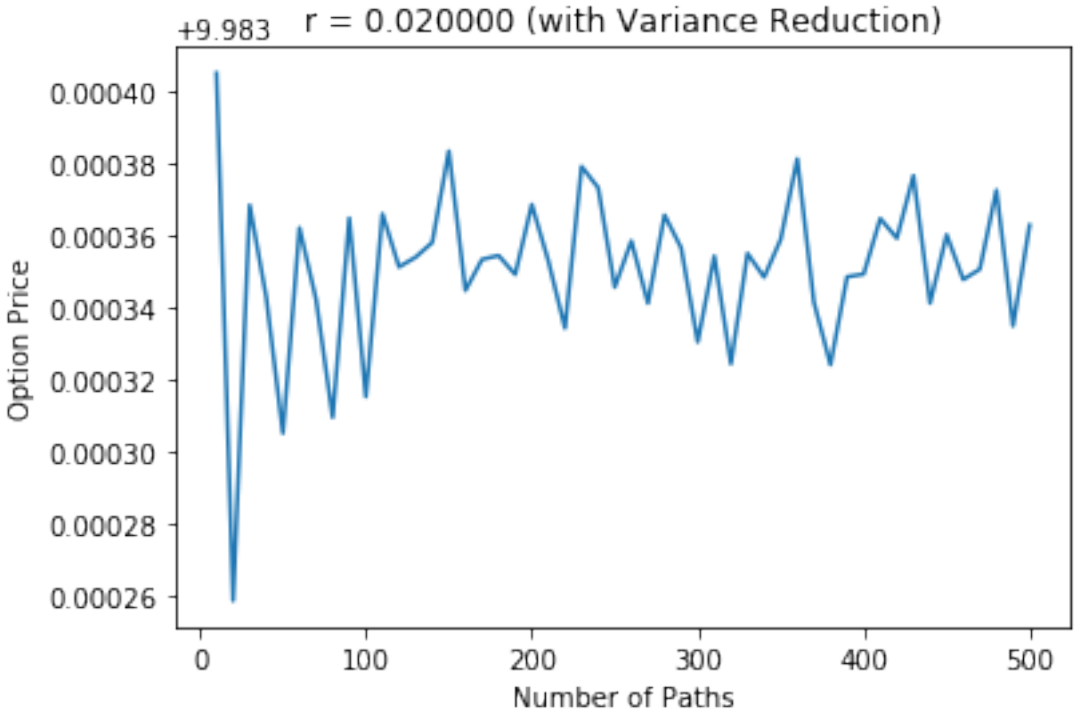

K = 110 (without Variance Reduction)
Option Price
9.950
9.925
9.900
9.875
9.850
9.825
9.800
0
100
200
300
400
500
Number of Paths

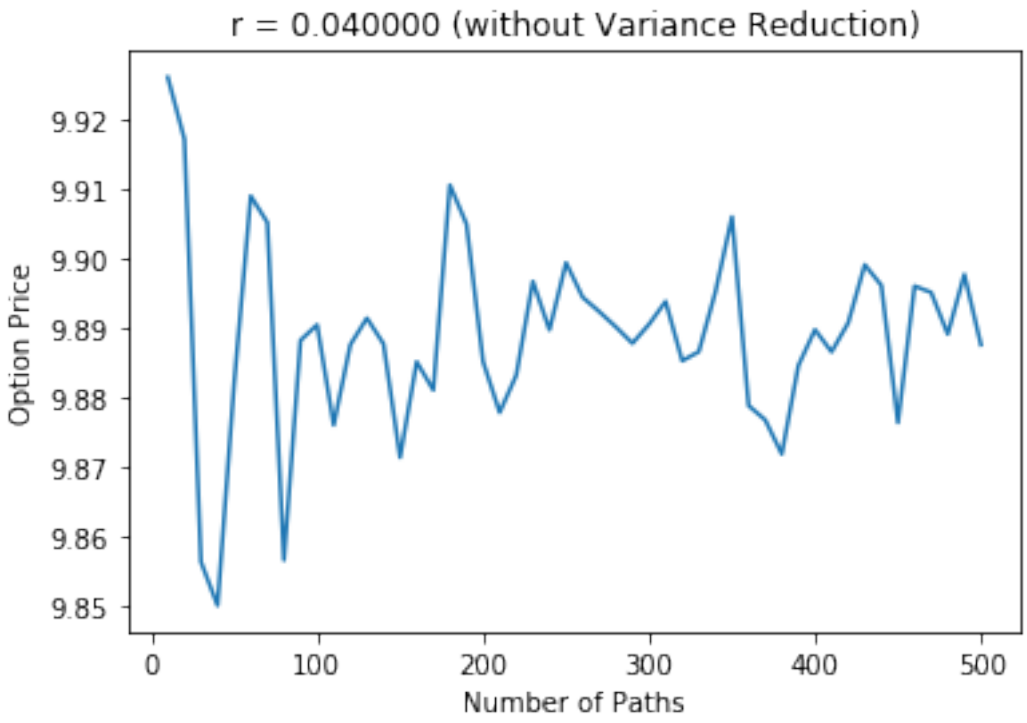

K = 110 (with Variance Reduction)
Option Price
9.842
9.841
9.840
9.839
9.838
0
100
200
300
400
500
Number of Paths

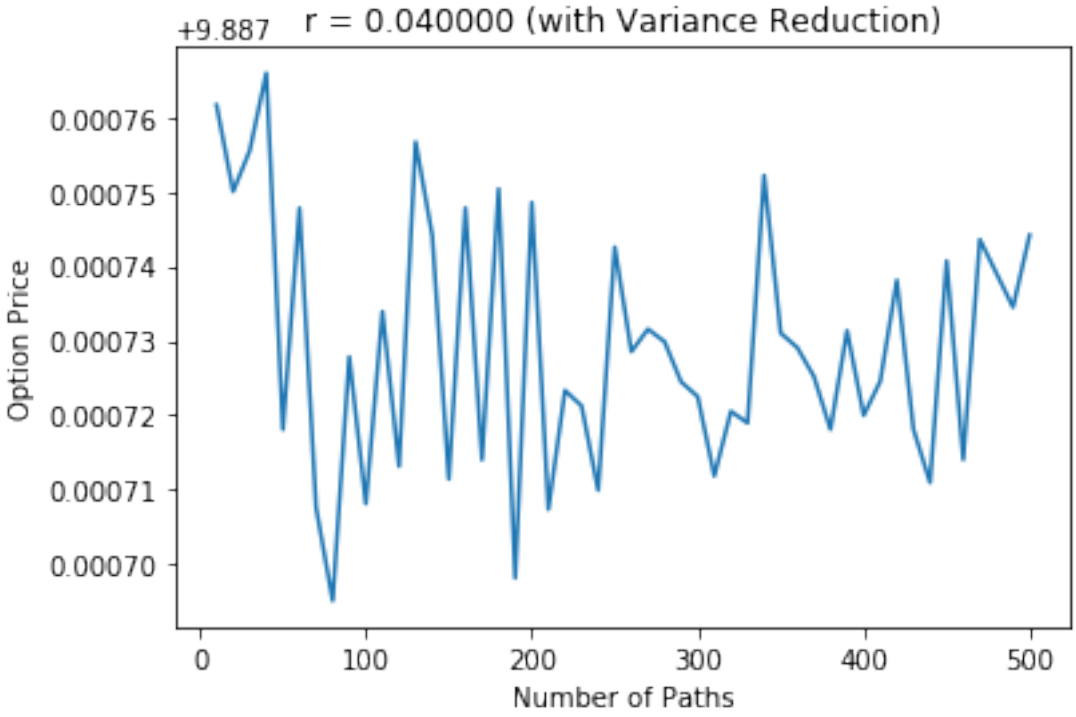
In [7]:
```
S0 = 100.0
r = 0.05
mu = 0.1
sig = 0.2
n = 30
K = 110
maxPaths=500
```

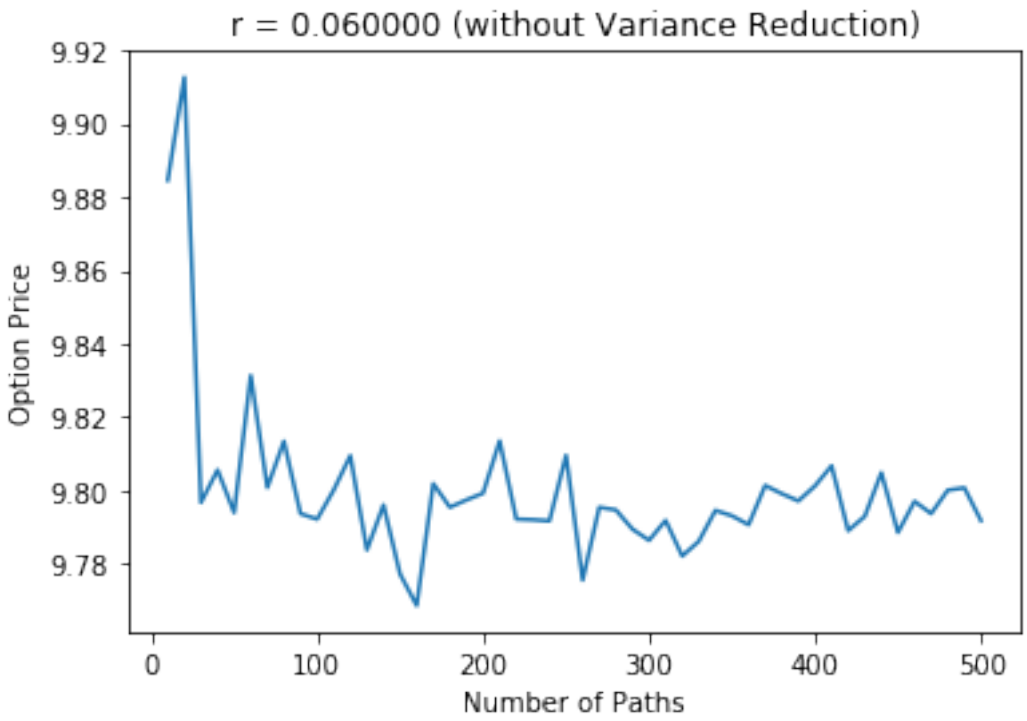
In [8]:
```
for r in np.arange(0.02, 0.11, 0.02):
    # Without Variance reduction
    price = getAsianOptionPrice(S0, r, mu, sig, n, K, varReduction=False)

    X = np.arange(10, maxPaths+1, 10)
    Y = price
    plt.xlabel("Number of Paths")
    plt.ylabel("Option Price")
    plt.title("r = %f (without Variance Reduction)"%(r))
    plt.plot(X, Y)
    plt.show()

    # With Variance reduction
    price = getAsianOptionPrice(S0, r, mu, sig, n, K, varReduction=True)

    X = np.arange(10, maxPaths+1, 10)
    Y = price
    plt.xlabel("Number of Paths")
    plt.ylabel("Option Price")
    plt.title("r = %f (with Variance Reduction)"%(r))
    plt.plot(X, Y)
    plt.show()
```

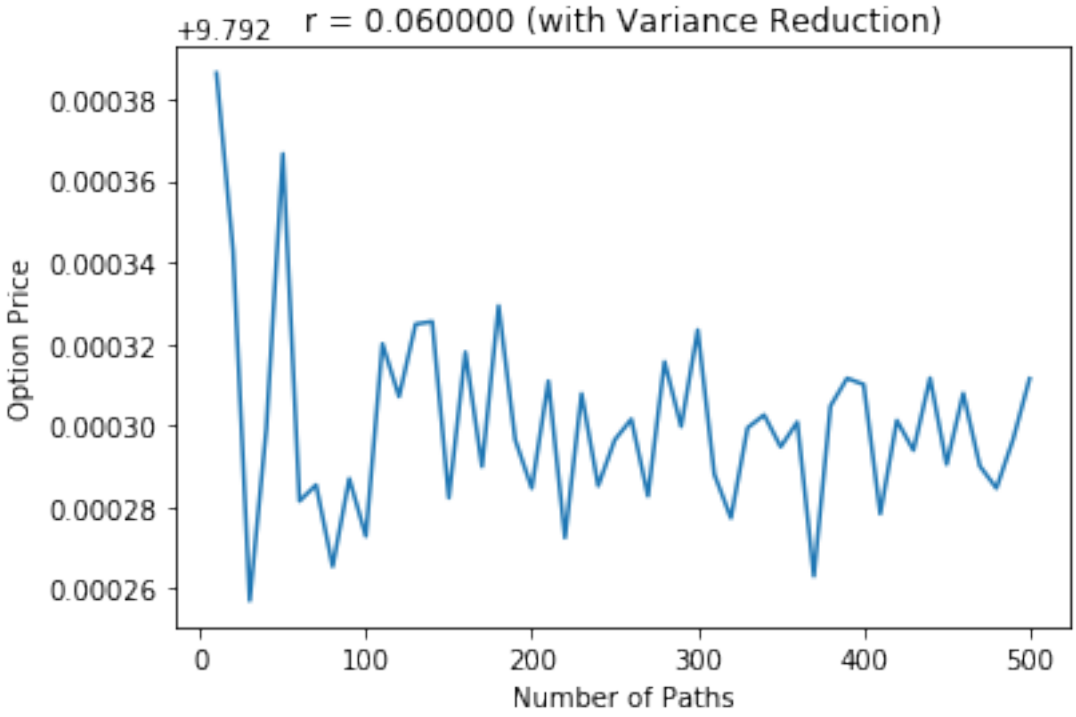

r = 0.020000 (without Variance Reduction)
Option Price
10.02
10.01
10.00
9.99
9.98
9.97
9.96
9.95
0
100
200
300
400
500
Number of Paths

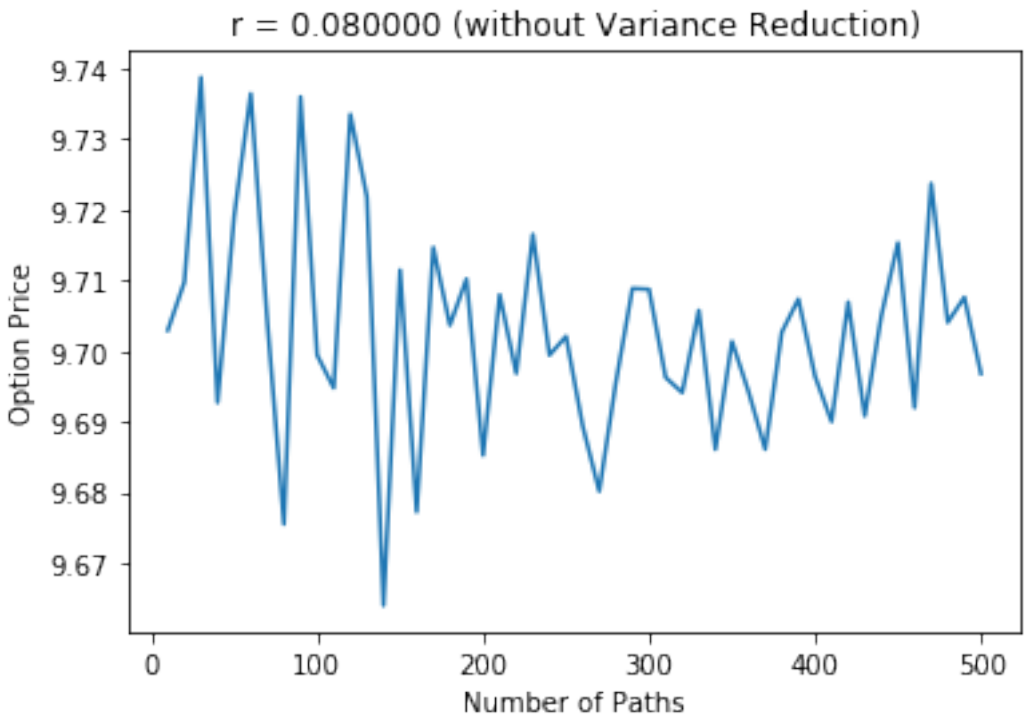

+9.983
r = 0.020000 (with Variance Reduction)
Option Price
0.00040
0.00038
0.00036
0.00034
0.00032
0.00030
0.00028
0.00026
0
100
200
300
400
500
Number of Paths

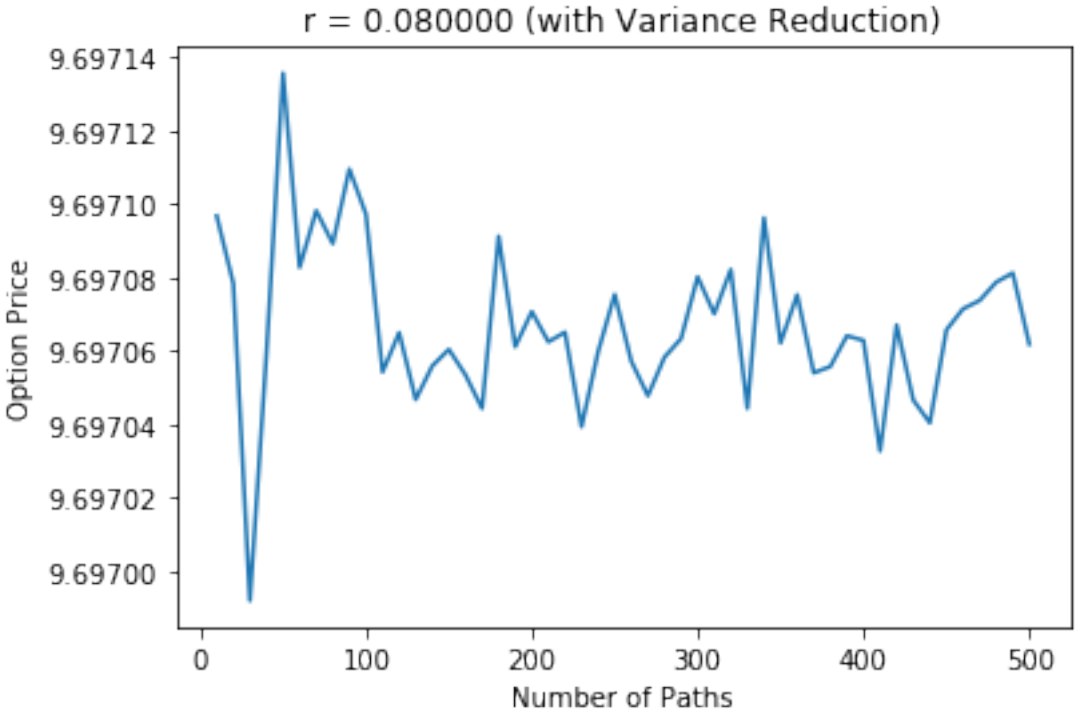

r = 0.040000 (without Variance Reduction)
Option Price
9.92
9.91
9.90
9.89
9.88
9.87
9.86
9.85
0
100
200
300
400
500
Number of Paths

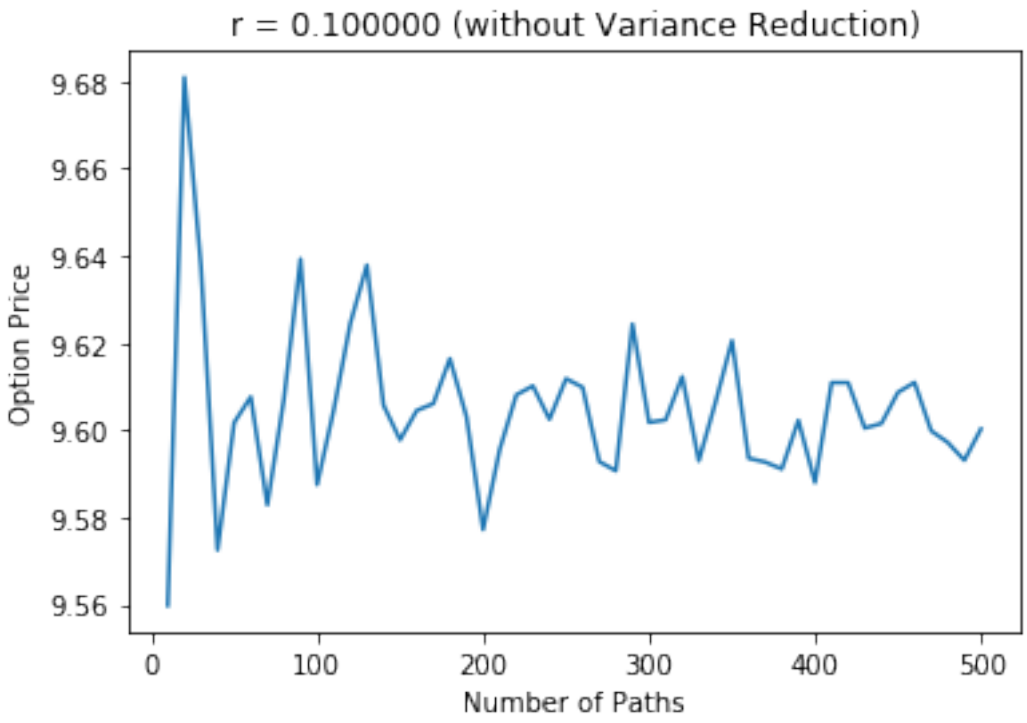

+9.887
r = 0.040000 (with Variance Reduction)
Option Price
0.00076
0.00075
0.00074
0.00073
0.00072
0.00071
0.00070
0
100
200
300
400
500
Number of Paths

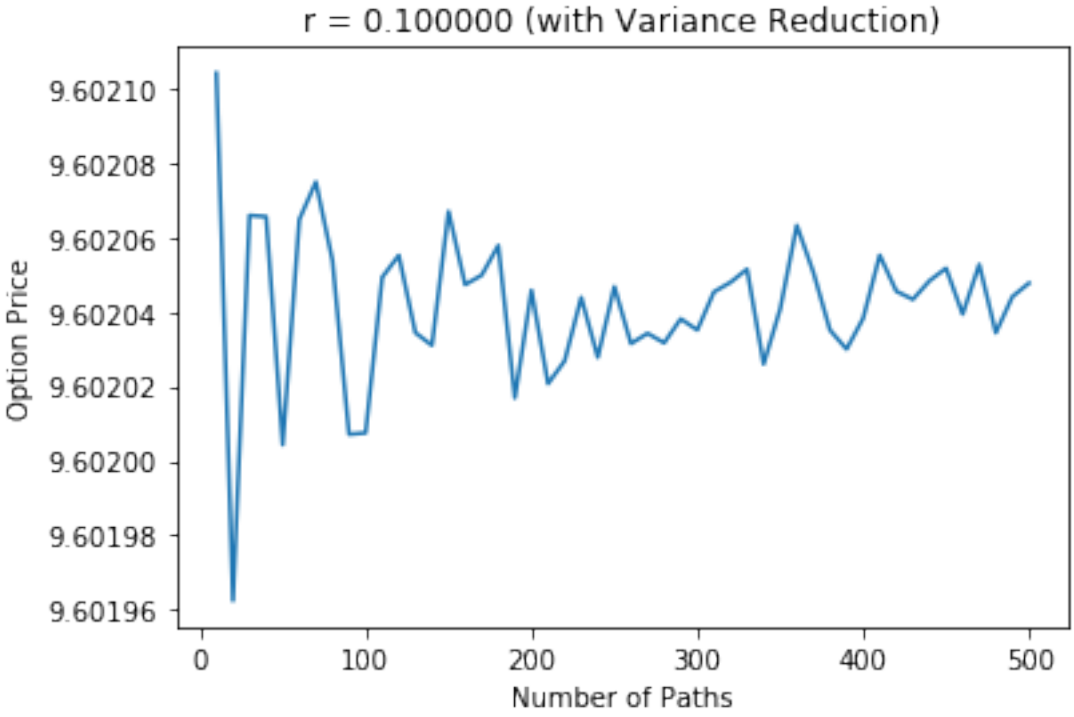

r = 0.060000 (without Variance Reduction)
Option Price
9.92
9.90
9.88
9.86
9.84
9.82
9.80
9.78
0
100
200
300
400
500
Number of Paths


+9.792
r = 0.060000 (with Variance Reduction)
Option Price
0.00038
0.00036
0.00034
0.00032
0.00030
0.00028
0.00026
0
100
200
300
400
500
Number of Paths

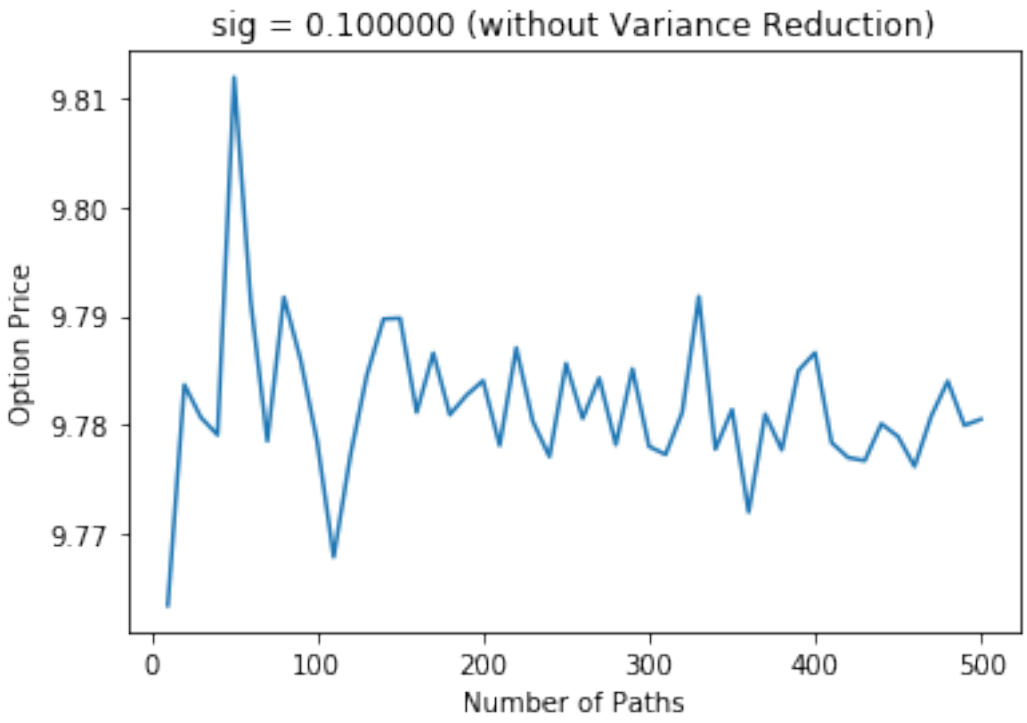

r = 0.080000 (without Variance Reduction)
Option Price
9.74
9.73
9.72
9.71
9.70
9.69
9.68
9.67
0
100
200
300
400
500
Number of Paths

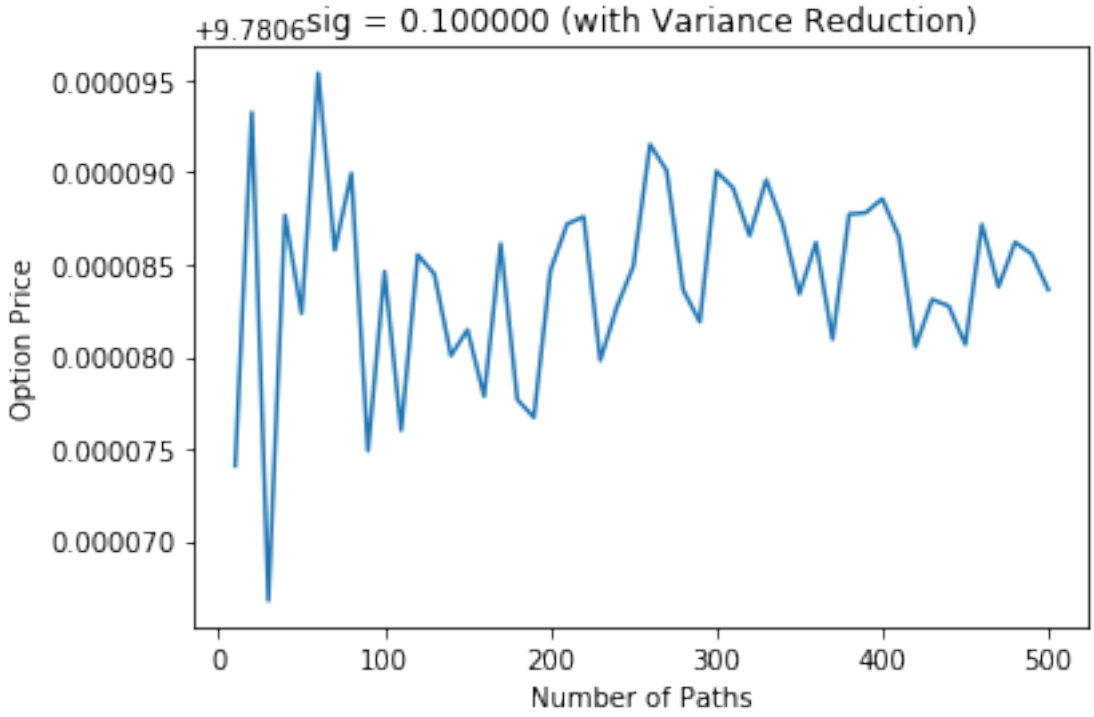

r = 0.080000 (with Variance Reduction)
Option Price
9.69714
9.69712
9.69710
9.69708
9.69706
9.69704
9.69702
9.69700
0
100
200
300
400
500
Number of Paths

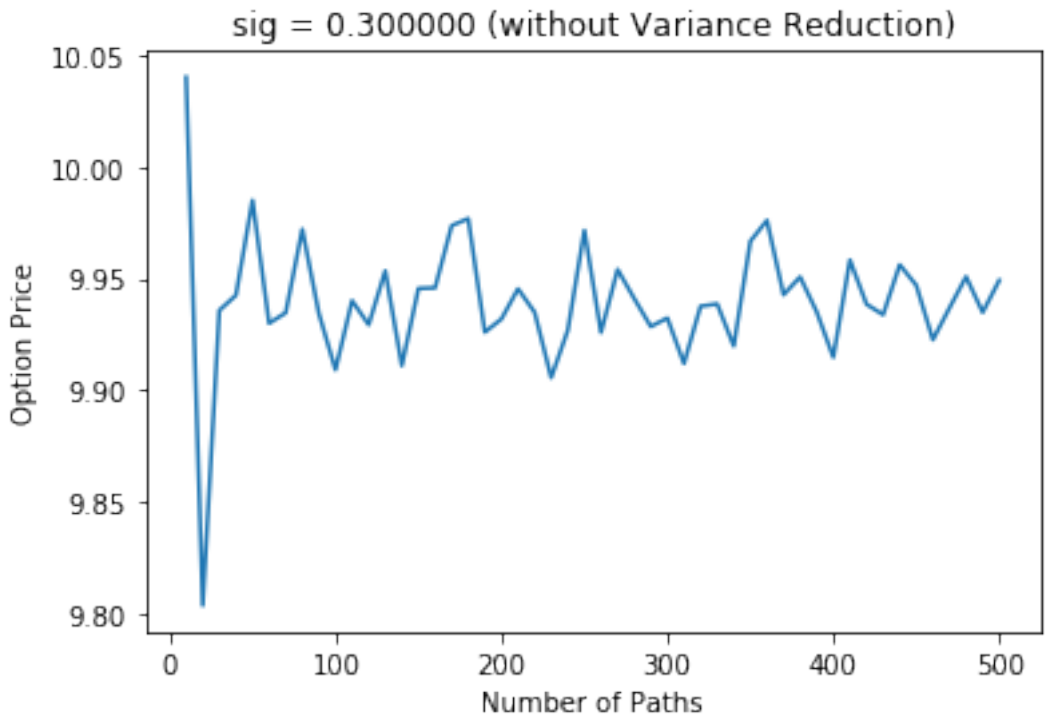

r = 0.100000 (without Variance Reduction)
Option Price
9.68
9.66
9.64
9.62
9.60
9.58
9.56
0
100
200
300
400
500
Number of Paths

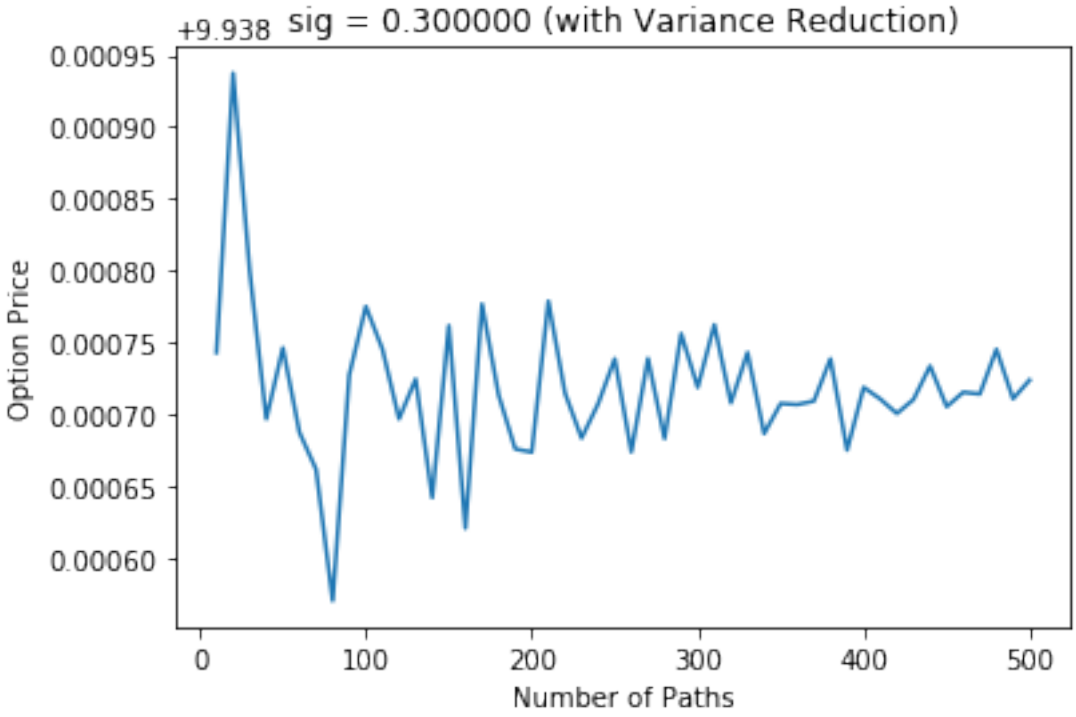

r = 0.100000 (with Variance Reduction)
Option Price
9.60210
9.60208
9.60206
9.60204
9.60202
9.60200
9.60198
9.60196
0
100
200
300
400
500
Number of Paths

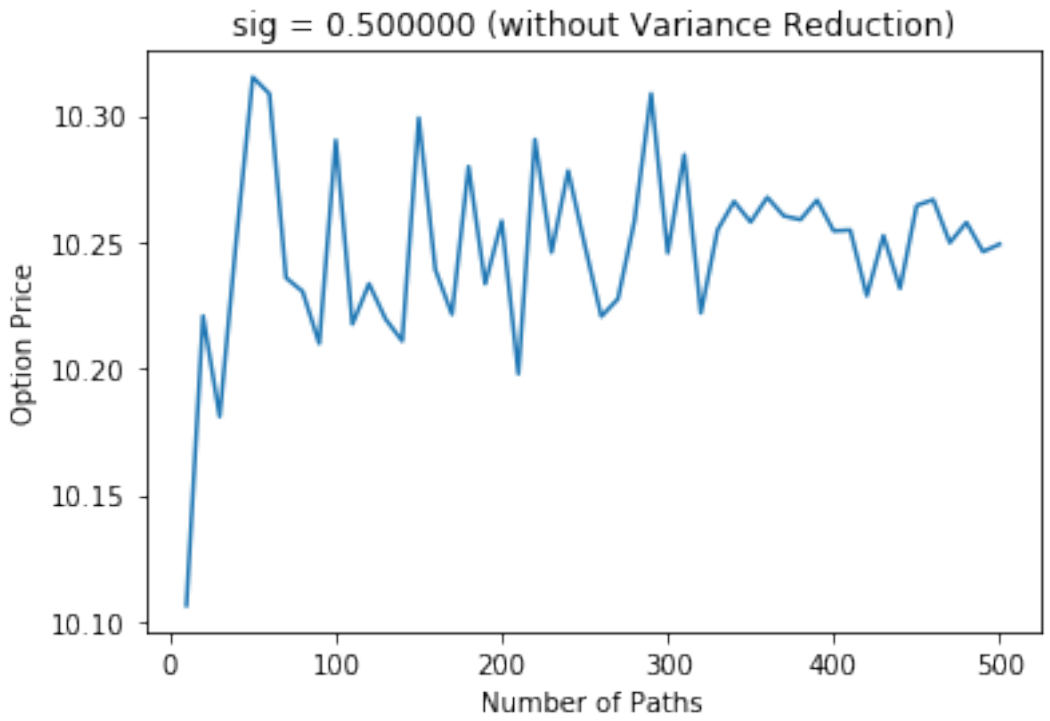
```
In [9]: S0 = 100.0
        r = 0.05
        mu = 0.1
        sig = 0.2
        n = 30
        K = 110
        maxPaths=500

In [11]: for sig in np.arange(0.1, 1.1, 0.2):
             # Without Variance reduction
             price = getAsianOptionPrice(S0, r, mu, sig, n, K, varReduction=False)

             X = np.arange(10, maxPaths+1, 10)
             Y = price
             plt.xlabel("Number of Paths")
             plt.ylabel("Option Price")
             plt.title("sig = %f (without Variance Reduction)"%(sig))
             plt.plot(X, Y)
             plt.show()

             # With Variance reduction
             price = getAsianOptionPrice(S0, r, mu, sig, n, K, varReduction=True)

             X = np.arange(10, maxPaths+1, 10)
             Y = price
             plt.xlabel("Number of Paths")
             plt.ylabel("Option Price")
             plt.title("sig = %f (with Variance Reduction)"%(sig))
             plt.plot(X, Y)
             plt.show()
```

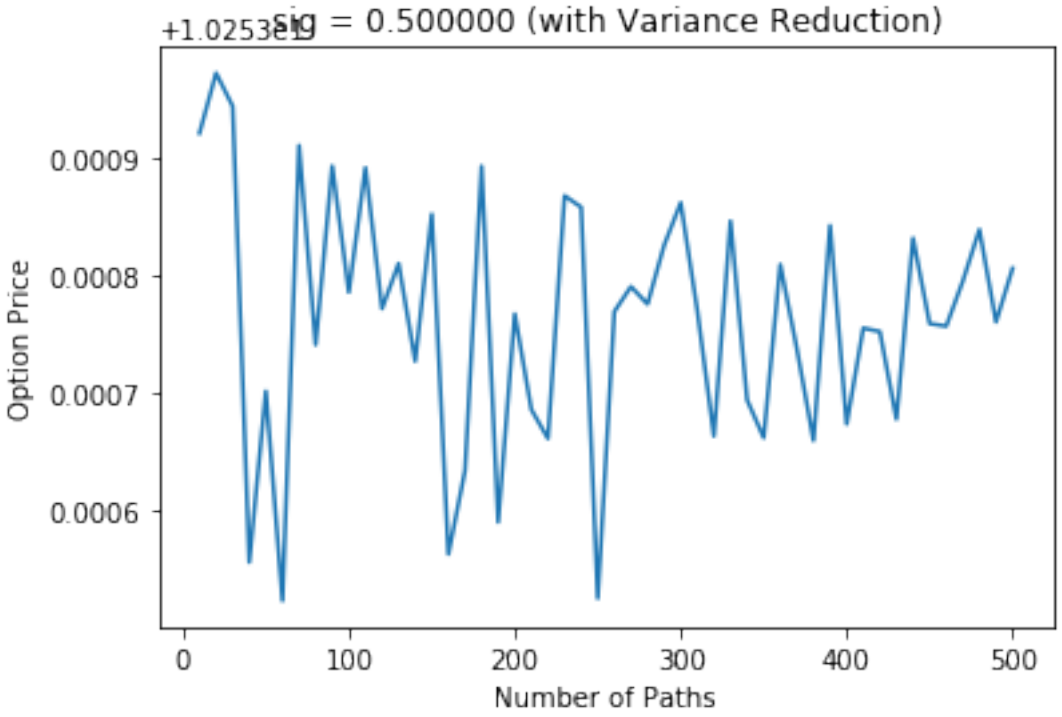

sig = 0.100000 (without Variance Reduction)
Option Price
9.81
9.80
9.79
9.78
9.77
0
100
200
300
400
500
Number of Paths

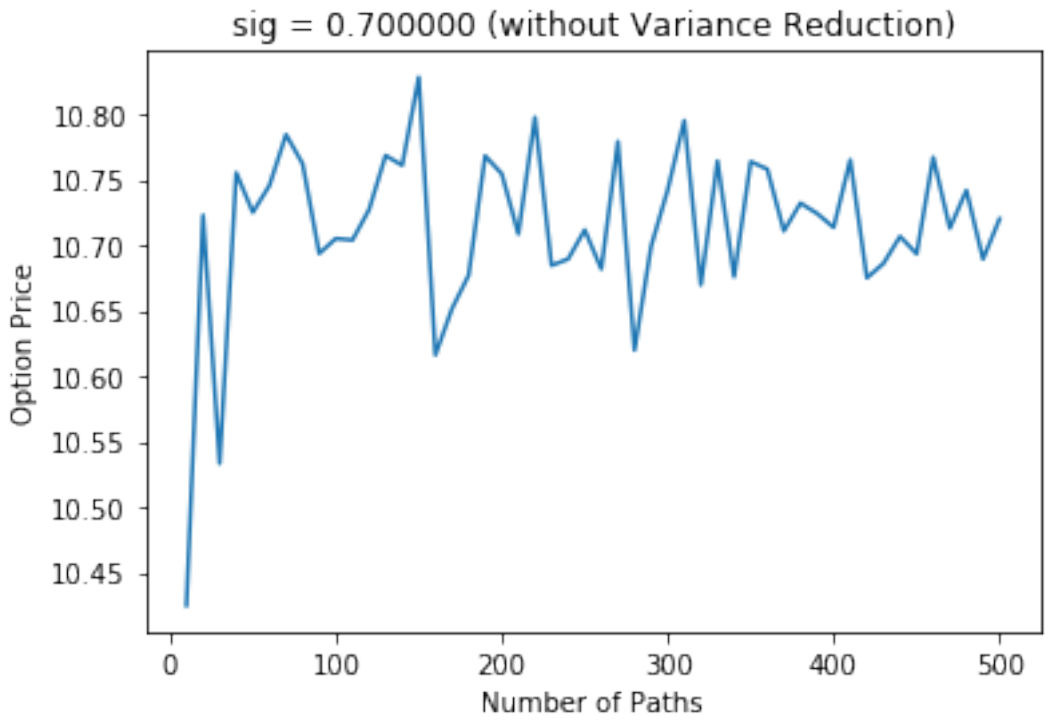

+9.7806sig = 0.100000 (with Variance Reduction)
Option Price
0.000095
0.000090
0.000085
0.000080
0.000075
0.000070
0
100
200
300
400
500
Number of Paths

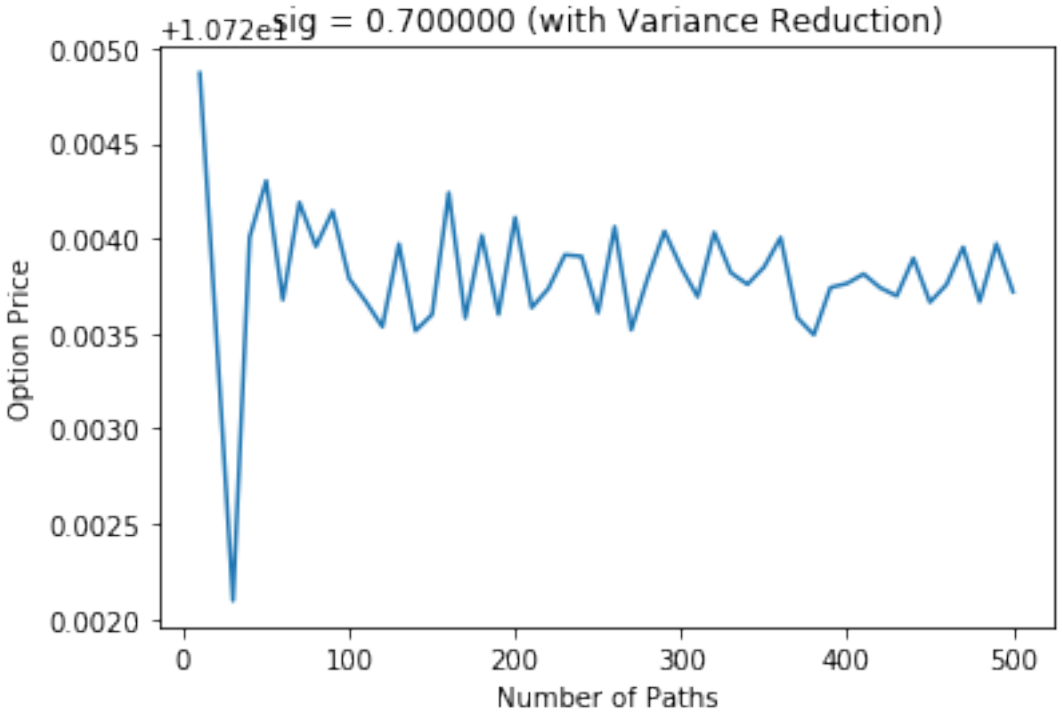

sig = 0.300000 (without Variance Reduction)
10.05
10.00
9.95
9.90
9.85
9.80
Option Price
0
100
200
300
400
500
Number of Paths

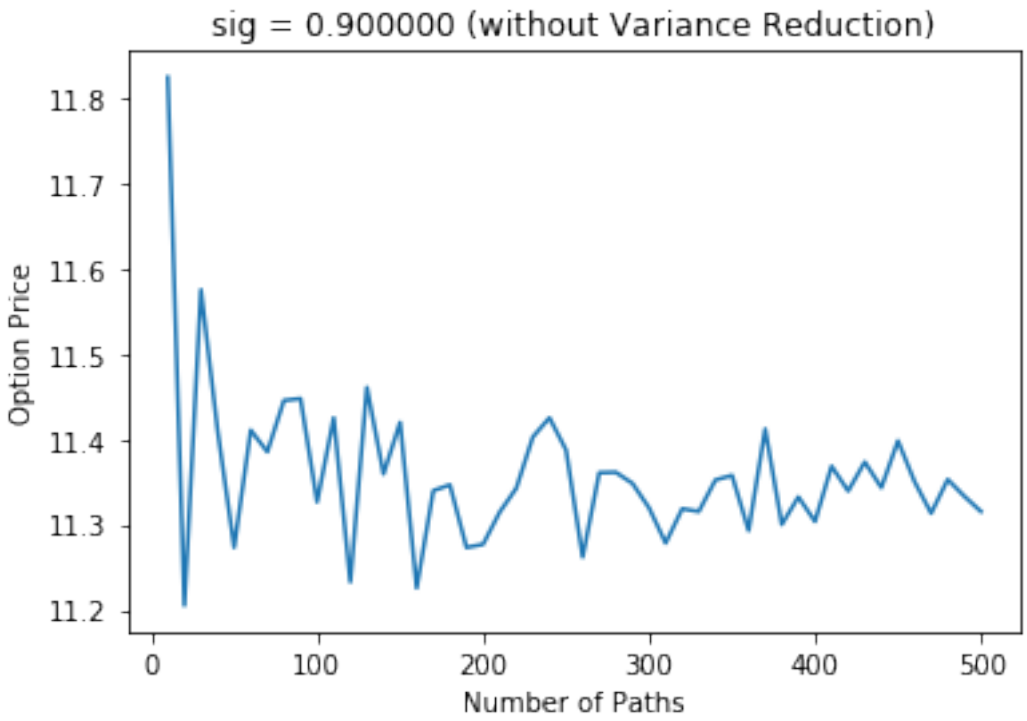

+9.938
sig = 0.300000 (with Variance Reduction)
0.00095
0.00090
0.00085
0.00080
0.00075
0.00070
0.00065
0.00060
Option Price
0
100
200
300
400
500
Number of Paths

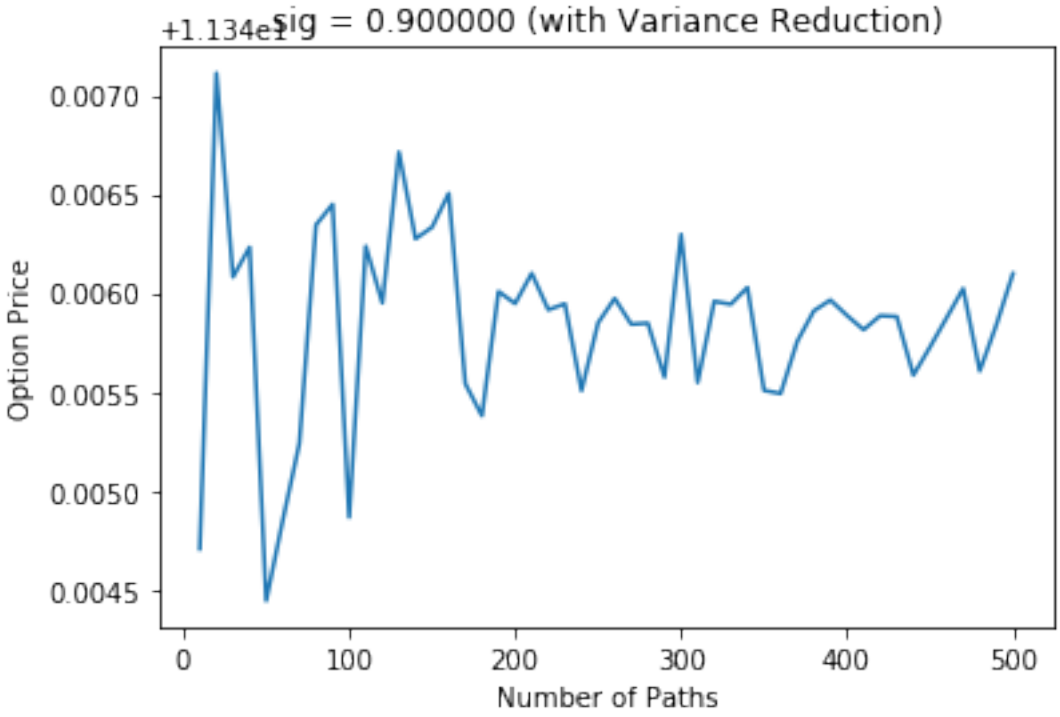

sig = 0.500000 (without Variance Reduction)
Option Price
10.30
10.25
10.20
10.15
10.10
0
100
200
300
400
500
Number of Paths


sig = 0.500000 (with Variance Reduction)
+1.0253e+0
Option Price
0.0009
0.0008
0.0007
0.0006
0
100
200
300
400
500
Number of Paths


sig = 0.700000 (without Variance Reduction)
Option Price
10.80
10.75
10.70
10.65
10.60
10.55
10.50
10.45
0
100
200
300
400
500
Number of Paths


sig = 0.700000 (with Variance Reduction)
+1.072e1
Option Price
0.0050
0.0045
0.0040
0.0035
0.0030
0.0025
0.0020
0
100
200
300
400
500
Number of Paths


sig = 0.900000 (without Variance Reduction)
Option Price
11.8
11.7
11.6
11.5
11.4
11.3
11.2
0
100
200
300
400
500
Number of Paths


sig = 0.900000 (with Variance Reduction)
+1.134e1
Option Price
0.0070
0.0065
0.0060
0.0055
0.0050
0.0045
0
100
200
300
400
500
Number of Paths

In [ ]: