VaR_Calculate

September 26, 2016

In [1]:
```
#####Problem 4
import scipy.stats as ss
import numpy as np
def VaRCalculate(V0, mu, sigma, T, p):
    ans = []
    Expect = V0 * np.exp(mu * T)
    ans.append(Expect)
    Var = pow(V0,2) * ( np.exp(pow(sigma,2)*T) - 1) * np.exp(2*mu*T)
    ans.append(Var**(0.5))
    VaR = V0 - V0 * np.exp( sigma * T**(0.5)* ss.norm.ppf(1-p) + (mu - pow(sigma,2)/2)*T )
    ans.append(VaR)
    return ans

V0 = 10000
mu = 0.05
sigma = 0.3
T = 1/252
p = 0.9
VaRCalculate(V0, mu, sigma, T, p)
```

Out[1]: [10001.98432383514, 189.03661471820388, 239.08753658932801]

In [2]:
```
V0 = 10000
mu = 0.05
sigma = 0.3
T = 5/252
p = 0.9
VaRCalculate(V0, mu, sigma, T, p)
```

Out[2]: [10009.925557498198, 423.18546674214952, 526.21167589945435]

In [3]:
```
V0 = 10000
mu = 0.05
sigma = 0.3
T = 1
p = 0.9
VaRCalculate(V0, mu, sigma, T, p)
```

Out[3]: [10512.710963760241, 3226.1227438149945, 3157.7294840360928]

In [ ]: