**Problem Set 6**

Directions:

For each of the following problems, use Python to find your answers.

Show me the Python code you used in your answer.

1. For this use the usual SP500 data set that we have used in class. In all cases use logarithmic returns.
2. Estimate the 40 day VaR(p=0.025) using the method of sampling the prices every 40 days.
3. Now estimate the 40 day VaR by assuming that 1 day returns are log normal, and use the delta-normal method and the **square root of h** rule to expand the return distribution to 40 days.
4. Now repeat this VaR estimation using the independent bootstrap with 10,000 replications.
5. Repeat the last part with a blocked bootstrap, drawing a 40 day block each time. Again, use 10,000 replications.
6. Your bootstrap procedures are used to estimate VaR, but not confidence bands. How might you try to estimate confidence bands?

Part 5:

Part 5 is a thought question. We are now using the bootstrap to get an actual

VaR estimate itself, and not the confidence band. This is new. What you would

have to do is to build another resampling bootstrap around your inner bootstrap. Each time construct a new bootstrapped version of retVecb=retVec then go into the inner loop where you draw retb from this new vector, retVecb. This gives a form of a double bootstrap. This will greatly slow down your program.

1. You have a portfolio whose final value is based on the future price of a stock. However, because of a complex set of derivative transactions, the valuation is nonlinear with **V = 101 - (P-101)^2**. The initial price, P0 = 100, and the initial value of the portfolio is V0 = 100. You know that the future price, P1, is distributed as a normal distribution with mean 101, and standard deviation 1.
   1. Write Python code to find the 0.01 VaR and ES in this case. In both cases use a monte-carlo simulation with the assumed distribution for prices and 10,000 monte-carlo simulations.
   2. Now do this with a delta normal approach. First, write down a linearization of the valuation, V(P), evaluated at P = P0 = 100. (This is the first order Taylor approximation.) Use this and the appropriate quantile for the price to find the 0.01 VaR (no ES this time). You can write down Python code for this, but you shouldn’t do a monte-carlo approximation.
   3. If the price really does follow that normal distribution which of these answers is closest to the correct measure of risk? Why?
2. I am an investor concerned about the ratio of my long run wealth, W(T), divided by my current wealth, W(1). I’m planning on investing wealth in a single risky asset with a given expected return, and variance, holding and reinvesting all returns from period 1 to period T. Returns are independent over time.
   1. Do I know anything about the distribution of W(T)/W(1)?
   2. I am expecting that this ratio would be (1+*u*) *T*  (1+u)T where u = the expected annual return for the asset. Will I underperform this target more or less than 50 percent of the time? Why?