EFM Week 5 Assignment

Jason Phang, Allan Zhang

November 1, 2014

1 Instructions

Only one question on this week's assignment, which will hopefully integrate what you have been doing on the Python assignments until now with the basic Portfolio Theory stuff that you saw this week. Try to do as many parts of this question as you can, though again don't spend more than one hour or so on this.

Some of the coding here might be a little tedious, particularly with handling both dates and price/returns series. This will be the last time you have to do this manually. In the coming weeks, we will introduce the **NumPy** and **pandas** libraries to make your life a lot simpler.

Email your solutions to efm.uchicago@gmail.com in a single python file, named `LASTNAME_FIRSTNAME.py` Also feel free to email us if you have any problems.

2 The Efficient Frontier with a Sample Portfolio

Before we move to trying to pick an optimal portfolio for the future, let's first begin by verifying what an optimal portfolio would have looked like in the past:

2.1 Setting Up

For this problem we will use data from 2003-2013 for the following equities: Microsoft (MSFT), Apple (AAPL), and Corning (GLW). Go to Quandl and get the data sets for each security with monthly returns.

Note that we want to keep track of the dates as well as the prices. For each asset, make a dictionary consisting of the list of **dates** and **prices**. Then, generate the returns series from the prices, so we have 3 lists in each asset's dictionary.

2.2 Basic Functions

Write a function that returns the mean of an asset, a function that returns the standard deviation of an asset, and a function that returns the covariance of any two assets.

2.3 More General Functions

Extend the above functions so that they also take in two datetime objects from matplotlib's dates library, and returns the mean, standard deviation, and covariance of an asset within any given time frame.

2.4 Crafting A Portfolio

Write a function that takes in: 1) a dictionary of assets and weights for each asset and 2) two datetime objects, and returns the expected return and standard deviation of that portfolio over the given time period.

2.5 Plotting the Efficient Frontier

Now choose 1 out of the 3 possible 2 asset portfolio, and using 0.05 intervals for asset weights, plot each portfolio in the standard deviation-expected return space using matplotlib.

2.6 Efficient Frontier with More Assets

Now do the same as the above, but do it for a portfolio with all 3 assets.

2.7 Risk Free Rate and Sharpe Ratio

Now use this 3 month US Treasury series: `https://www.quandl.com/WREN/W9-US-3mth-Treasury-Bills` as a risk free rate of return for each month, and calculate the Sharpe ratio for each asset as well as for each of your two portfolios above.

2.8 Plot the Capital Allocation Line

Now for each of the two portfolios you crafted above, plot the capital allocation line using the above risk free rate.

2.9 Variations over Time Frames

Plot the Efficient Frontier graph using matplotlib for the 3 asset portfolio over the time frames 2003-2008 and from 2008-2013.

2.10 Comparing Over Time Frames

Find the portfolio with the highest Sharpe ratio from the time period 2003-2008. Now see how well that portfolio did from 2008-2013.