Diversification and Portfolio Risk

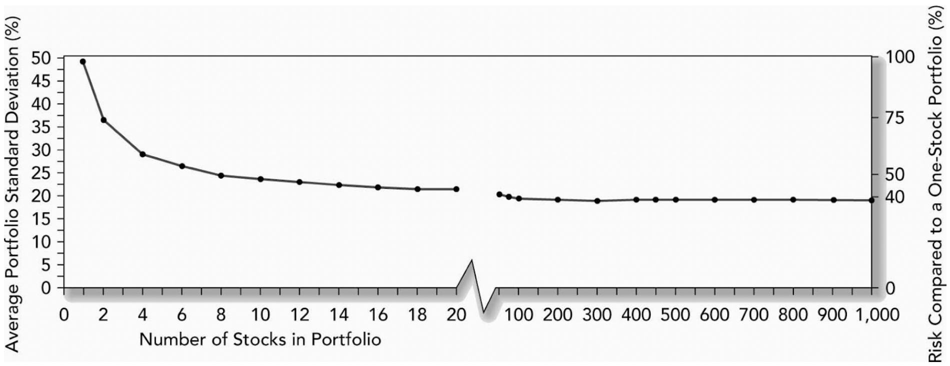
Project Summary

Tristan Lalor

May 3rd, 2019

Overview

The purpose of this program is to analyze the relationship between the number of assets in a portfolio, and the average standard deviation of that portfolio. This relationship for every stock in the US market is exhibited below. My program runs a simulation to analyze this relationship for different asset classes.



Abstract

1. Obtain daily and monthly return data for a selection of different US asset classes.
2. Each value of n shall represent a different number of assets that a portfolio can be made up of. For each value of n, there shall be a given number of trials, t.
3. For each iteration of t, assets will be selected at random until they reach the number, n. Then, the portfolio standard deviation will be calculated based on this random sample of n assets. These standard deviations will be recorded, and the average standard deviation for each value of n, over all trials t, will become the data points that are represented on the y-axis. Each of these data points represents the average standard deviation for an n asset portfolio, given t trials for each value of n.
4. After obtaining the y-axis data points, we can generate a list of numbers from 1 to n to use for the x axis, and generate a graph with matplotlib.

Methods

The most critical theory behind setting up this model is that of portfolio management. Obtaining the standard deviation of a portfolio is much more complex than taking a weighted average of the individual standard deviations of all its assets. The standard deviation of a portfolio is the square root of its variance, and the variance is a function of the covariance between each of its assets. So for a portfolio made up of 200 assets, its variance is determined by 40,000 (200 x 200) covariance relationships. In the context of this model, the most accurate covariance is based on the correlation between assets' daily returns, and the standard deviations from their monthly returns. (Standard deviations over multi-year periods are more accurately annualized from monthly trading data as opposed to daily because while the trading days per year is variable, there are always 12 months per year. This is an important consideration for calculations in the model.) An outline of portfolio variance is given on the following page.

Portfolio Variance

$$\sigma_{r_p}^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} w_j w_i \sigma_{ij} \qquad \sigma_{r_p}^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} w_j w_i \sigma_i \sigma_j \rho_{ij}$$

Where:
- w_i is the proportion of the portfolio invested in asset i at the beginning of the period
- $\sigma_{i,j}$ is the covariance of returns between i and j
- $\sigma_{i,j} = \sigma_i^2$ if and only if i = j
- σ_i is the standard deviation of asset i
- $\rho_{i,j}$ is the correlation of returns between assets i and j

Matrix view with the covariances between assets' returns and the portfolio weights:

weight	weight STOCK	w_a A	w_b B	w_c C
w_a	A	$Cov(r_a,r_a) = \sigma_a^2$ Covariance of *A* with *A* is just the variance of *A*	$cov(r_b,r_a)$	$cov(r_c,r_a)$
w_b	B	$cov(r_a,r_b)$	$cov(r_b,r_b) = \sigma_b^2$	$cov(r_c,r_b)$
w_c	C	$cov(r_a,r_c)$	$cov(r_b,r_c)$	$cov(r_c,r_c) = \sigma_c^2$

- Each element gives the covariance of returns for the intersection of the column's stock and the row's stock. (The diagonal from the NW to the SE is the covariance between each stock and itself. This is, by definition, the stock's variance.)

- Note as well that the matrix is symmetric: for each weight and covariance above the diagonal there is an equal covariance and weight below the diagonal

- Calculating the variance of the portfolio is done by multiplying each of the covariances in the matrix by the weight at the top of its column and the weight at the left side of its row

- You then obtain the variance of the portfolio by summing these products

Note: $\text{Covariance}_{A,B} = \text{Correlation}_{A,B} \times \text{Stdev}_A \times \text{Stdev}_B$

Package Installations

I had to install two packages to complete this project. The first is numpy. Numpy was critical for obtaining the correlations between all of the assets. I use the Anaconda extension for PyCharm to manage package installations in virtual environments. Instructions to configure a Conda virtual environment are given here, and instructions for how to install the numpy package are given here.

The second package I had to install was matplotlib. The instructions for this installation are given here. (*note: the installation is exactly the same as numpy.*)

Data

All data is extracted from the Bloomberg Terminal. For custom datasets, enter "Custom" upon the first input prompt, and then enter the filenames of the daily and monthly data sets. All data must be formatted correctly; refer to the file "Format Example.xlsx" to see exactly how to copy data into your custom text files.

Testing Instructions

It is vital to make sure that all of the corresponding text files are accessible by the program, and that the program is run with Pycharm in a Conda virtual environment with the two above packages installed. In-depth instructions are given when the program is run. The user will be prompted to specify a dataset, as well as how many trials they would like to run in the simulation. Using larger datasets with a high number of trials could take an extremely long time to process.