

Automating Markowitz Optimizations Using VBA

David Porter and Robert Stretcher

In the interest of reducing grading workload and managing cross-student sharing (cheating), we provide an example of automating the construction of Markowitz (1959) efficient frontiers. The program automates the entire procedure from downloading data for a specified number of securities (using either randomly generated securities from a provided list or specified tickers) to generating graphs of the efficient frontier. Many commercially available optimization programs are available, but this program is free to use and works within Microsoft Excel (2007-2010), the software package most professors use when creating optimization assignments. The program is written in Visual Basic for Applications (VBA) but does not require any knowledge of VBA to use. The program allows easy creation of random portfolios for student assignments so that cross-student sharing can be minimized without adding an additional grading burden on the professor or TA. The professor can also be assured that the stocks selected produce the desired type of optimization (actually work) before using each set of securities. Where students are allowed to select their own portfolios, the program allows professors to easily check computations and optimizations without having to manually optimize each portfolio. The program can also be used to demonstrate the similarity of efficient frontiers across different stocks, portfolio sizes, and dates.

INTRODUCTION

Cutbacks in government funding to public universities has resulted in reductions across many areas including but certainly not limited to the hiring of new faculty to keep class sizes small. As class sizes increase, grading of assignments becomes more problematic both in the amount of time required and in the possibility of cross-student sharing (also referred to as free-riding or cheating). In the case of investments courses, many professors require students to complete some type of optimization assignment. With class sizes exceeding 100 students, however, offering individual students the opportunity to select their own stocks makes the grading process difficult or prohibitive. As a result, students are commonly placed in groups, and entire classes may be assigned the same set of stocks to optimize. Although this technique significantly reduces the length of time needed to evaluate the assignments, it also increases the probability of cross-student or cross-group sharing making it difficult to determine which students actually understand the optimization process. One solution to these issues is to use an automated system designed to generate Markowitz (1959) optimizations so that every group or student could be either assigned a separate set of stocks to optimize, or be allowed to select their own securities. In this paper we describe a VBA program that can be used to automate the optimization of portfolios using Solver in Microsoft Excel. The object is not to teach VBA but to offer some typical code that can be used to expedite a professor's use of this type of automation.

DESCRIPTION OF THE "FRONT-END" OF THE PROGRAM

The program is designed for a personal computer (PC) running a recent version of Excel (2007 or 2010) with the

Solver add-in installed and does not require any knowledge of VBA, unless the individual professor wishes to adjust the code for his/her own purposes. The program will not run on an Apple operating system but could work if the MAC was dual booted into Windows and was running the appropriate Excel version. Since the VBA code is essentially a large macro, when the file containing the program is opened, two warnings may occur: (1) This file originated from an Internet location and might be unsafe and, (2) Some active content has been disabled. Click for more details. In both cases, clicking "Enable Content" allows the VBA to run. The file will then default to the "Explanation" sheet that contains a brief explanation of the program.

As noted in the red box, the program downloads monthly data from Yahoo for either a randomly selected portfolio of securities or a specific set of ticker symbols. If the user wants random portfolios, the number of securities is entered in cell F1. If the user wants a specific set of securities, the tickers should be entered in column A, starting with cell A1, and cell F1 should be set to 0. Cell F1 is checked first, so if it contains a number between 2 and 20, a random portfolio will be generated even if there are specific tickers in column A. The restriction of 2 to 20 tickers is usually sufficient for most assignments but can easily be altered by changing the VBA code. The program will compute and graph the Markowitz optimization assuming no short selling. Stocks with less than 30 months of data are not used since the mean return is used as the expected return and 30 observations is typically assumed to be a "reasonable" statistical sample. The user can also change the minimum observation assumption by altering the VBA code.

Figure 1. Front End Explanation Box

The screenshot shows a spreadsheet with columns A through L. In column A, rows 1-5 contain the tickers DVN, ETN, LTD, VMC, and ADM. Cell C1 contains the text "Number of Random Tickers:" and cell F1 contains the value "0". Cell G1 contains the instruction "Enter 0 if you are specifying the tickers in Column A1:A...". A red text box with white text explains the program's functionality: "This program computes and charts a Markowitz optimization for random stocks selected from the stock data on the worksheet labeled 'Stocks' or from the tickers listed in column 'A' (minimum 2 tickers, maximum 20 tickers). The program requests the begin and end dates of the data and always uses monthly data. Data is downloaded from Yahoo Finance. If Yahoo cannot find the ticker or if any of the tickers do not have 30 months of data over the dates selected (typically considered the minimum number of observations for a 'good' statistical sample), the program displays a warning and terminates. This version does not allow for short selling." Below the text box is a red button labeled "Run Program".

Before clicking “Run Program”, the user should note the status bar at the bottom of the spreadsheet:

Figure 2. Status Bar.

The screenshot shows the status bar at the bottom of the spreadsheet. A blue box with white text says "↓↓ Program status is displayed on the status bar below ↓↓". The spreadsheet tabs at the bottom include Explanation, Stocks, Prices, Returns, BasicStats, Model, ChartData, and Graph. A small text box below the tabs says "Click the 'Run Program' button when you are ready to start..."

Figure 3. Prices Sheet.

	A	B	C	D	E	F
1		PCG	ETR	IFF	CTL	APC
2	Date	Adj Close	Adj Close	Adj Close	Adj Close	Adj Close
3	Aug-12	43.41	68.08	60.18	41.55	69.18
4	Jul-12	46.16	71.83	55.43	40.84	69.35
5	Jun-12	45.27	67.11	54.49	38.82	66.12
6	May-12	43.26	63.79	55.75	37.85	60.84
7	Apr-12	43.74	63.98	59.54	37.21	73.02
8	Mar-12	42.97	65.58	57.95	37.30	78.13
9	Feb-12	40.83	65.02	56.09	38.13	83.81
10	Jan-12	39.83	66.89	54.89	35.08	80.42
11	Dec-11	40.38	70.43	51.55	35.24	76.05
12	Nov-11	37.63	67.83	53.05	34.85	80.88
13	Oct-11	41.57	65.89	59.21	32.75	78.12
14	Sep-11	40.99	63.15	54.96	30.76	62.75
15	Aug-11	40.59	62.12	56.43	32.90	73.30
16	Jul-11	39.71	62.75	59.49	33.78	82.06
17	Jun-11	40.79	64.14	62.48	36.80	76.29

The status bar reports where the program is at any point in time. If it appears that nothing is happening, the user should check the status bar. Also note that the program generates six additional spreadsheets. Prices, Returns, BasicStats, Model, ChartData and Graph. The seventh

additional sheet is the list of stocks used for randomly generating portfolios. This sheet can be populated by the user but the original file contains about 500 stocks as examples. The "Prices" sheet contains the adjusted closing prices downloaded from Yahoo. Since the program uses monthly prices, these prices are the last adjusted closing price of each month.

The "Returns" sheet contains the returns computations based on the downloaded prices. Where the number of prices differs across the securities, the returns are truncated to the shortest set of data. This is necessary for the computation of correlations and covariances. If the number of returns is less than 30 for any of the securities the sample is considered too small and the program will terminate.

The Basic Stats sheet contains summary measures for the stocks: means, standard deviations, minimum return, maximum return, and range of returns, as well as the correlation matrix.

Figure 4. Returns Sheet.

	A	B	C	D	E
1	PCG	ETR	IFF	CTL	APC
2	-5.9575	-5.2207	8.5694	1.7385	-0.2451
3	1.9660	7.0332	1.7251	5.2035	4.8851
4	4.6463	5.2046	-2.2601	2.5627	8.6785
5	-1.0974	-0.2970	-6.3655	1.7200	-16.6804
6	1.7919	-2.4398	2.7437	-0.2413	-6.5404
7	5.2412	0.8613	3.3161	-2.1768	-6.7772
8	2.5107	-2.7956	2.1862	8.6944	4.2154
9	-1.3621	-5.0263	6.4791	-0.4540	5.7462
10	7.3080	3.8331	-2.8275	1.1191	-5.9718
11	-9.4780	2.9443	-10.4036	6.4122	3.5330
12	1.4150	4.3389	7.7329	6.4694	24.4940

Figure 5. Basic Stats Sheet.

	A	B	C	D	E	F
1	Summary Measures for Stock Returns					
2						
3	Stocks	PCG	ETR	IFF	CTL	APC
4	Means	0.3894	-0.1912	0.7039	0.5886	1.4230
5	Std. Dev.	4.3698	5.8038	6.3021	6.8484	12.5459
6	Min	-9.4780	-13.9126	-19.2308	-31.4738	-30.8929
7	Max	8.6084	16.3804	16.7496	13.6035	36.2108
8	Range	18.0864	30.2930	35.9804	45.0773	67.1037
9						
10	Correlations	PCG	ETR	IFF	CTL	APC
11	PCG	1.0000	0.4871	0.3017	0.1946	0.2045
12	ETR	0.4871	1.0000	0.2949	0.4240	0.5072
13	IFF	0.3017	0.2949	1.0000	0.5141	0.4569
14	CTL	0.1946	0.4240	0.5141	1.0000	0.5042
15	APC	0.2045	0.5072	0.4569	0.5042	1.0000

The "Model" sheet contains the data needed for the optimization: the weights for each stock for the last optimization run, the expected returns (the means), the variance-covariance matrix, the expected return, variance and standard deviation for the optimized portfolio and the target cell. The target cell changes as the program generates the chart data from the minimum variance portfolio to the maximum return portfolio.

Figure 6. Model Sheet.

	A	B	C	D	E	F	G
1	Portfolio Selection Model						
2							
3	Stock	PCG	ETR	IFF	CTL	APC	Sum
4	Weights	0.0134	0.0000	0.4568	0.0000	0.5299	1.0000
5	Means	0.3894	-0.1912	0.7039	0.5886	1.4230	
6							
7	Covariances	PCG	ETR	IFF	CTL	APC	
8	PCG	18.7773	12.1475	8.1690	5.7267	11.0263	
9	ETR	12.1475	33.1226	10.6063	16.5712	36.3190	
10	IFF	8.1690	10.6063	39.0544	21.8195	35.5250	
11	CTL	5.7267	16.5712	21.8195	46.1188	42.5989	
12	APC	11.0263	36.3190	35.5250	42.5989	154.7774	
13							
14	E(Rp)	1.0807		Target	8.3102		
15							
16	Var(Rp)	69.0594		Sigma(Rp)	8.3102		

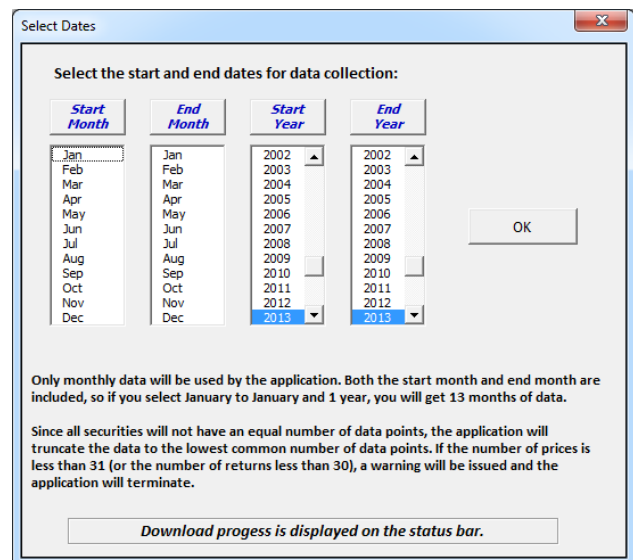
The "ChartData" sheet contains the Solver optimization output for each of the 16 runs from the minimum variance portfolio to the maximum return – the weights for each security for each run, as well as the portfolio expected return and standard deviation. The number of runs is arbitrary and 16 (15 Solver runs plus the maximum return portfolio) was chosen as a number that would produce a nice looking graph the majority of the time. The "Graph" sheet contains a scatter plot of the chart data. Since most graphs of the efficient frontier look better with more points near the minimum variance point, the chart data is generated using that assumption. The graph will also auto-scale itself based on the data. An example of the graph output is in Figure 10.

Figure 7. ChartData Sheet.

	A	B	C	D	E	F	G	H	I
1	Efficient Frontier					Optimal Portfolio Weights			
2		Port. Std.Dev.	Port. Return		PCG	ETR	IFF	CTL	APC
3		12.5459	1.4230		0.0000	0.0000	0.0000	0.0000	1.0000
4		8.3102	1.0807		0.0134	0.0000	0.4568	0.0000	0.5299
5		6.1355	0.8569		0.2784	0.0000	0.3871	0.0000	0.3345
6		5.0190	0.7197		0.4213	0.0000	0.3268	0.0393	0.2125
7		4.4457	0.6318		0.5010	0.0000	0.2774	0.0887	0.1330
8		4.1514	0.5729		0.5542	0.0000	0.2443	0.1217	0.0797
9		4.0003	0.5323		0.5911	0.0000	0.2215	0.1445	0.0429
10		3.9227	0.5036		0.6169	0.0000	0.2055	0.1605	0.0170
11		3.8829	0.4831		0.6394	0.0000	0.1894	0.1712	0.0000
12		3.8624	0.4607		0.6465	0.0235	0.1661	0.1638	0.0000
13		3.8519	0.4436		0.6369	0.0469	0.1598	0.1564	0.0000
14		3.8465	0.4315		0.6300	0.0636	0.1553	0.1511	0.0000
15		3.8438	0.4228		0.6251	0.0755	0.1521	0.1473	0.0000
16		3.8424	0.4165		0.6215	0.0841	0.1498	0.1446	0.0000
17		3.8416	0.4121		0.6190	0.0902	0.1481	0.1427	0.0000
18	M-V Portfolio	3.8409	0.4008		0.6126	0.1057	0.1439	0.1377	0.0000

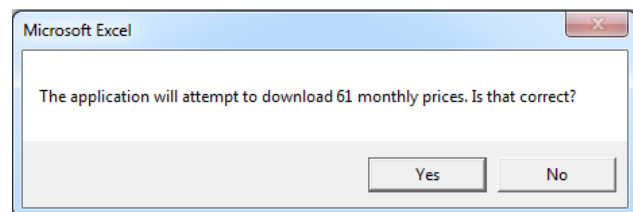
When you click "Run Program" the date selection dialog box pops up:

Figure 8. Date Selection Dialogue Box.



Use the mouse to select the desired start and end dates. There are several checks built into the dialog box such as the date cannot be in the future and the start date must be before the end date. Once again, the user is reminded that a minimum of 30 observations must be used. Once the dates are selected and the OK is clicked, a number of observations check dialog box is displayed:

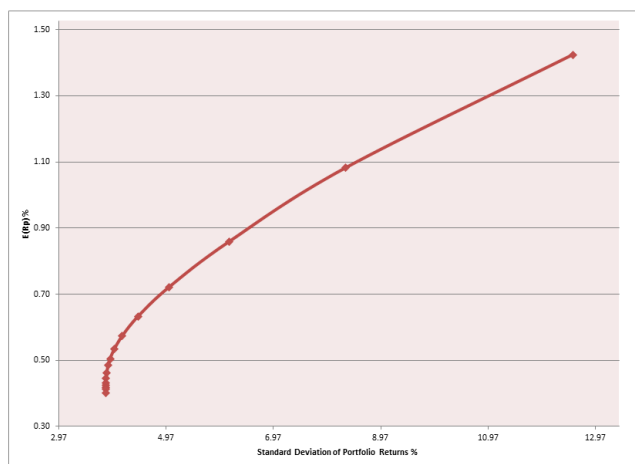
Figure 9. Download Initiation Selection Box.



If the user clicks "No" they are returned to the date dialog box. After clicking "Yes", the user can watch the

status bar as the program attempts to download the data from Yahoo and complete the optimization. Once completed, the "Graph" sheet is shown:

Figure 10. Markowitz Graph Output.



SUGGESTED CLASSROOM USE

UW-Whitewater runs two sections of undergraduate Investments each term with section sizes between 55 and 60 students. Lectures and exams cover the optimization process for two risky assets when r_f exists, but the undergraduate students rarely show a thorough understanding of the method. One way to increase their understanding is use real world data with more than 2 risky assets. With undergraduates, 5 risky assets is sufficient to determine if they can complete the optimization process and graph an efficient frontier. Instructions on how to download data from finance.yahoo.com are provided to students along with instructions on how to optimize a portfolio using Solver in Excel. Some textbooks contain this information but the undergraduate textbook currently in use at UW-Whitewater does not. The yahoo instructions are also available from the authors and instructions for Markowitz style optimization are available in many textbooks including Bodie, Kane and Marcus, *Investments* (McGraw-Hill Irwin), Chapter 7, Appendix A (pages 234-239 in the 9th edition) and Craig Holden's *Excel Modeling and Estimation in Investments* (Pearson Prentice Hall), Chapters 7 and 8 in the 3rd edition.

When class sizes were smaller (in the early 1990's Investment class sizes at Whitewater were in the 15-20 range), students were put into small groups and allowed to select their own securities. Grading of 6-10 assignments was not prohibitive and assignments could be returned in a few days. As class sizes grew, group sizes increased but class sizes reached the point where it was necessary to assign a single group of specific stocks for the entire class. These adjustments allowed the grading of 30-40 assignments in a reasonable timeframe but dramatically increased the amount

of cross-group sharing. Requesting students behave ethically and increasing the penalty for cross-group sharing had little effect on the issue. It was clear that each group or student would need to be assigned a separate set of securities and that a more efficient method of grading had to be used; thus the development of the automated program.

We use the program for several purposes. For the undergraduates, we randomly generate the 5 stock portfolios they use in their optimizations. If there are 40 groups, we randomly generate 40 5-stock portfolios. Since the program generates the graph for the random securities, it is possible to determine if the optimization produces a "good" graph before including that set of securities in the assignment. Students may select different points for their optimization than the automated program but the program significantly reduces the time needed to determine if the optimization was done correctly. For the graduates, they select their own 10 stock portfolios, but those assignments can still be easily graded since the program can download and optimize a 10 stock portfolio in about 5 seconds (assuming a reasonable Internet connection). We also use the program in-class to demonstrate to students that most efficient frontiers have a similar shape regardless of the time period, length of time period or number of securities in the portfolio. We find that simply discussing correlations and their relationship to the shape of the efficient frontier is never as effective as using real data. The students also get to see the entire process from hand calculating a two stock optimization, to using Excel to optimize a multi-stock portfolio, to a fully automated computation of an optimization.

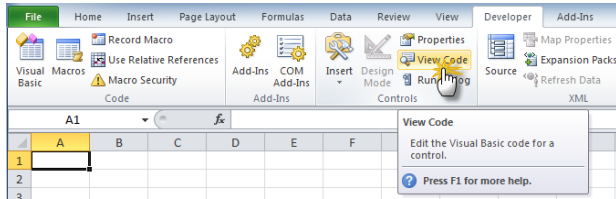
Assigning every group their own set of stocks has significantly reduced the amount of cross-group sharing but free-riding within groups still exists. Use of student self-evaluations within the group helps with this issue but often students are more concerned they will have to work with the same student again in another class than they are with the ethics of giving another student a good evaluation when they were actually a free-rider.

CONCLUSION

This paper described the "front-end" of an automated program for generating efficient frontiers using Microsoft Excel. The program can be used to reduce grading time while significantly curtailing cross-group sharing (cheating) when professors use assignments to improve student understanding of Markowitz (1959) optimizations. The program is freely available by contacting David Porter in the UW-Whitewater Department of Finance and Business Law. To minimize the distribution of the program to students, professors should use either their office phone or university email as evidence they are not a student. For those familiar with VBA, a PDF file is also available which explains the code thereby reducing the learning curve if changes are desirable to meet the individual needs of a specific course. To view the VBA code in the Excel file, use Alt+F11 or use

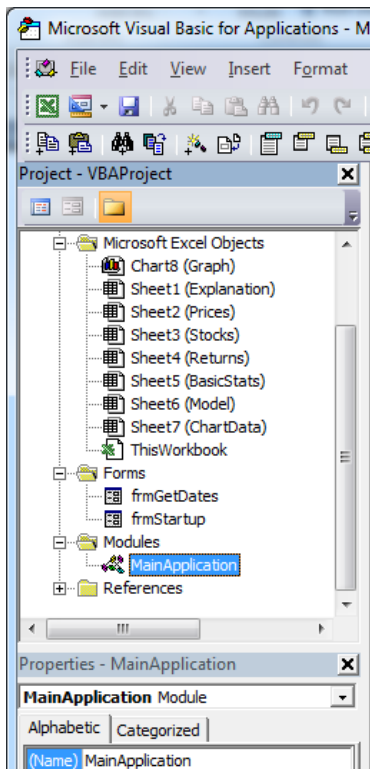
the Developer tab. The Developer tab is not shown on the Excel toolbar by default but can be turned on by going to File | Options | Customize Ribbon and checking "Developer" under the "Main Tabs". The code can then be viewed by clicking on "View Code":

Figure 11. Developer Tab.



The main code can be found by double clicking on the Module "MainApplication":

Figure 12. Main Application Location.



REFERENCE

Markowitz, H. (1959). *Portfolio Selection: Efficient Diversification of Investments*, John Wiley & Sons, New York City, New York.

David Porter is a Professor of Finance at the University of Wisconsin-Whitewater.

Robert Stretcher is a Professor of Finance at Sam Houston State University.

Demonstrating Retrieval of Financial Information in XBRL

Ameeta Jaiswal-Dale and Jianing (Jade) Fang

The Securities and Exchange Commission (SEC) has upgraded the existing system of Electronic Data Gathering, Analysis, and Retrieval (EDGAR), to Interactive Data Electronic Applications (IDEA) platform, using eXtensible Business Reporting Language (XBRL). In January 2009, the SEC issued its final mandate for XBRL adoption and the conversion target dates for all firms. This conversion enables users to retrieve listed companies' financial statement information at two levels, document and the data element, compared to the document level alone under the existing system of EDGAR. Beneficial to all users, this change is particularly important to resource-strapped entities such as small businesses, and universities. The result is user friendly for students researching information. With simply the industry standard tool of Microsoft Excel, without expensive proprietary XBRL modules, detailed information is now available, quickly, efficiently and at minimum cost. This paper explains the basic concept of XBRL and demonstrates the ease of the retrieval process in Microsoft Excel.

INTRODUCTION

On January 30, 2009, the Securities and Exchange Commission (SEC) chairperson, Mary Schapiro, announced the agency's final mandate for eXtensible Business Reporting Language (XBRL) adoption and the firm conversion target dates (SEC, January 2009). The mandate stated the largest domestic and foreign public companies that use U.S. Generally Accepted Accounting Principles (GAAP) to file their financial statements in XBRL format by June 15, 2009¹: medium-sized filers by June 15, 2010, and the rest of the filers, either using U.S. GAAP or International Financial Reporting Standards (IFRS), by June 15, 2011 (Fang 2010).

The new rules are intended to make financial information easier for investors to analyze and to assist in automating regulatory filings and business information processing. The XBRL system lists information at both the document level, such as the entire set of financial statements for a given firm, and the data element level, such as individual accounts like inventory. In XBRL interactive data, or data tagged at the data element level, can function across multiple and/or different platforms or application programs. Thus, XBRL has the potential to increase the speed, accuracy and usability of financial disclosure, and