**Assignment 2**

*Module* : CF963 –Learning and Computational Intelligence in Economics and Finance (CF963-7-AU)

*Year*: 2011-2012

*Professor*: Edward Tsang

*Department*: CCFEA, University of Essex

*Course*: MSc Computational Finance

*Student*: Riccardo Valerio Lattanzi

*PRID*: LATTA96007

**Report**

 The aim of this work is finding the optimal portfolio that represents the best trade-off between risk and return. In order to achieve this result I chose to start my work by considering the general framework provided by the “Modern Portfolio Theory” (MPT) introduced by Harry Markovitz in 1952 (source: “en.wikipedia.org”). This model is a two-parameter portfolio analysis model ( “Investment mathematics”- 2003, p.229) because it assumes that a portfolio of “**n**” assets can be described by two parameters, the “ **Portfolio Expected return**” and the “**Portfolio Risk**”. Given a set of weights **wi**  with i=1,…,n that are assigned to each asset, we can define the Portfolio Expected Return **E[r]** as:

$E\left[r\right]= \sum\_{i}^{n}w\_{i}r\_{i}$ ,

where “**ri**” are the single returns, and the Portfolio Risk **σ(p)** as:

$σ\left(p\right)=\sqrt{\sum\_{i}^{n}\sum\_{j}^{n}w\_{i}w\_{j}σ\_{ij}}$ ,

where “**σij**” is the covariance between asset “i” and asset “j” (“Investment Science” 1998, p.150).

 Another very important concept provided by Markovitz is the idea of “***diversification***”, one of the pillars of the MPT. Briefly, it states that the variance of the return of a portfolio can be reduced by including additional assets in the portfolio (“Investment Science” 1998, p.151). This helps the investor to find the portfolio with the minimum risk and the maximum return, therefore the portfolio with the maximum **Sharpe Ratio** “**S**” where **S** can be defined as:

$S= \frac{E\left[r\right]-r\_{f}}{σ(p)}$ ,

With “**rf**”risk-free rate (“Investment mathematics”- 2003, p.295) .

 Starting from these assumptions I developed my model. The structure of the work is the following:

1. I computed all the possible portfolios that we can generate by combining five different stocks chosen from a group of ten;
2. I then computed the equally weighted risk and return for each portfolio and the correspondent Sharpe Ratio;
3. I picked up the portfolio with the maximum Sharpe Ratio and I found, using the Excel “Solver” tool, the optimal weights that gave me the highest Sharpe Ratio;
4. By taking into account that we are allowed to buy only integer number of stocks I computed the new weights and the correspondent new Sharpe Ratio. In this case the amount of money invested in stocks was less than the initial investment, therefore I decided to invest the rest of the money at the risk-free rate.
5. The final result is :

|  |  |
| --- | --- |
| *Optimal values* |  *Weights* |
| *VOD* | *BT* | *BATS* | *IMT* | *OML* |
| 0.302261 | 0 | 0.697654 | 0 | 0 |
| *Standard Deviation* | *Returns* | *OPTIMAL SHARPE RATIO* |
| 0.004207917 | 0.341853 | **76.48747984** |

 .

**THE MODEL**

 The model has been developed using Microsoft Excel and partially using Matlab. The spreadsheet named “ **cf963\_12nd\_Assignment\_Riccardo\_Valerio\_Lattanzi** “ contains four worksheets with all the data used for the computations and the correspondent results.

 The Worksheet named “**Introduction**” briefly explains the content of the following three worksheets.

 The Worksheet named “**Data(Step1)**” contains all the data used for the calculations, included all the different portfolios obtained through the combination of five different stocks. The combinations “**C(n,k) = n!/k!(n – k)!** “ have been computed in Matlab by using the function **C = combnk(v,k)** that “**returns all combinations of the n elements in v taken k at a time** “ (source : Matlab Help ). The returns are the ones that have been provided for the assignment.

 The next step was computing the Sharpe Ratio for all the portfolios using equally weighted returns and standard deviations. The computation of the Variance-Covariance matrices and the standard deviations is based on a VBA macro developed for this specific purpose. The macro is named “**Portfolio\_Std**” and the code can be found in the Appendix A.

 The last step was finding the optimal weights in order to maximize the Sharpe Ratio. For this purpose I created the worksheet “**Portfolio\_Weights(Step3)**”. The optimization problem has been solved by using the Excel Solver. The Solver allows you to set up the constraints of the problem ( in our case the weights must be >=0 and the sum of the weights must be equal to 1) and to choose between different solving methods ( in our case we have chosen Generalized Reduced Gradient solving method, see <http://www.solver.com/suppstdgrgstop.htm> and also <http://www.utexas.edu/courses/lasdon/design3.htm> ). After having found a local optimal solution I adjusted the value of the weights in order to obtain an integer number of stocks. The new adjusted weights and the new values of return, standard deviation and Sharpe Ratio have been reported into the green table in the worksheet “**Portfolio\_Weights(Step3)**” .

**STRENGHTS AND WEAKNESSES**

 The main strength of this work is that the analysis of the investment includes all the possible portfolios that can be generated by combining five different stocks while the main weakness is represented by the fact that, due to the difficulty of applying the Excel Solver to each of the 252 different portfolios, the Excel Solver has been used only after having chosen the portfolio with the maximum Sharpe Ratio (step2) and not during portfolio selection phase (step2).

**APPENDIX A**

Sub Portfolio\_Std()

' Portfolio\_Std Macro

' Keyboard Shortcut: Ctrl+p

Dim i As Integer

For i = 11 To 262 Step 1

' It creates a new portfolio everytime "i" changes

Cells(4, 16) = Sheets("Data(Step1)").Cells(i, 4)

Cells(4, 17) = Sheets("Data(Step1)").Cells(i, 5)

Cells(4, 18) = Sheets("Data(Step1)").Cells(i, 6)

Cells(4, 19) = Sheets("Data(Step1)").Cells(i, 7)

Cells(4, 20) = Sheets("Data(Step1)").Cells(i, 8)

' It copies the new portfolio standard deviation into a new cell

Cells(i + 4, 8) = Cells(13, 8)

' It copies the weights of "Sharpe\_Ratios(Step2)" into "Portfolio\_Weights(Step3)"

Sheets("Portfolio\_Weights(Step3)").Cells(11, 8) = Sheets("Sharpe\_Ratios(Step2)").Cells(11, 8)

Sheets("Portfolio\_Weights(Step3)").Cells(11, 9) = Sheets("Sharpe\_Ratios(Step2)").Cells(11, 9)

Sheets("Portfolio\_Weights(Step3)").Cells(11, 10) = Sheets("Sharpe\_Ratios(Step2)").Cells(11, 10)

Sheets("Portfolio\_Weights(Step3)").Cells(11, 11) = Sheets("Sharpe\_Ratios(Step2)").Cells(11, 11)

Sheets("Portfolio\_Weights(Step3)").Cells(11, 12) = Sheets("Sharpe\_Ratios(Step2)").Cells(11, 12)

Next

End Sub

**REFERENCES:**

1. Luenberger, David G. (1998): Investment Science, New York, Oxford University Press.
2. Adams A., Booth P., Bowie D., Freeth D. (2003): Investment Mathematics, Wley Finance Series;