

20598 – Finance with Big Data

PC Lab #1: Applied Portfolio Theory (Week 2)

Clément Mazet-Sonilhac

`clement.mazetsonilhac@unibocconi.it`

Finance Department, Bocconi University

PC Lab Groups



PC Labs Groups

Surname	Name	Attending	Group
D'ANGELO	JACOPO MATTIA	Y	1
THOMOPOULOS	IOANNIS	Y	
RIENTH	MAXIMILIAN PAUL	Y	
MILANI	LUCA	Y	2
LASKOWSKA	MARTA MARIA	Y	
KACZOROWSKA	MONIKA	Y	
SEGATORI	LIVIA	Y	3
CARETTI	GIORGIO FILIPPO	Y	
CECCAGNOLI	CHIARA	Y	
VILLA	SOFIA	Y	4
LADAA	MICHAEL	Y	
MUKHTAR	OMAR NABIL MOHAMMED SAMI	Y	
PATEL	YASH ESHWAR SANJAY	Y	
ARSHAD	MUHAMMAD BIN	Y	5
IORI	GAIA	Y	
BORISOV	VLADISLAV STANISLAVOV	Y	
BRAZZOLI	FRANCESCA	Y	6
GARCIA LORENZANA	LUCA	Y	
MURARO	MARGHERITA	Y	

PC Labs Groups

IACCARINO	FRANCESCO	Y	7
BARICEVIC	ERIK	Y	
GIANOGLIO	GIACOMO	Y	
POLONI	GIADA	Y	8
BENGALLI	DESIREE	Y	
BORDIGNON	LICIA	Y	
CANNAROZZO	GIUSEPPE	Y	
KOLDYSHEV	ILIA	Y	9
VOYNOVA	ADELINA	Y	
BAKHTIARI	PARSA	Y	
CAGNACCI	JACOPO	?	10?
VAN DER SANGEN	ERIK JOHAN	?	
GUYOT DE LA POMMERAYE	CLEMENT ALBERTO MARIE ERIC	?	
BELLODI	GIACOMO	?	

PC Labs Groups

Non-attending:

LEGOTKIN	GLEB	N	Non Attending
CASCIO	AGOSTINO	N	
BRUNI	LORENZO	N	
KARAHAN	ARZUM	N	
ISSABAYEV	ALI	N	
CIESLAR	MARIE	N	
ZANDRINO	ANDREA BEATRICE	N	
VAN DEN BERGHE	TOMMASO	N	
CAVALLERA	LORENZO	N	
KALAK	UTKU ATAKAN	N	

Still waiting for an answer:

FILIPPI EDOARDO

RODA MATTEO


BAO YISHUN

PC Labs Grading

- PC Labs solutions are submitted as Jupyter Notebooks, via email to me and Andrea (TA): andrea.andolfatto@phd.unibocconi.it
 - Email title : PCLab#1 - Group X - Name1 Name2 Name3
 - Your Jupyter Notebook starts in the same way

PC Labs Grading

- PC Labs solutions are submitted as Jupyter Notebooks, via email to me and Andrea (TA): andrea.andolfatto@phd.unibocconi.it
 - Email title : PCLab#1 - Group X - Name1 Name2 Name3
 - Your Jupyter Notebook starts in the same way
- PC Labs grade will depend on :
 - Your ability to submit it **before the deadline (Friday, 12pm midnight)**
 - The **quality** of your code (comments, readability, use of functions, etc.)
 - The **structure** of the Jupyter Notebook : well organized, explain what you are doing and why
 - Your ability to **complete the tasks**



Applied Portfolio Theory

Goals

- Manipulate and visualize stock market data (S&P 500)
- Create portfolios and compute basics statistics
- Draw the efficient frontier and find the *best* (i.e., tangency) portfolio
- Test the Markowitz mean-variance theory in practice: can you beat the market?

Big picture context



- You've just been hired by a large *old-school* Asset Management Company
- Your team manages a *regulated mutual fund*

Big picture context



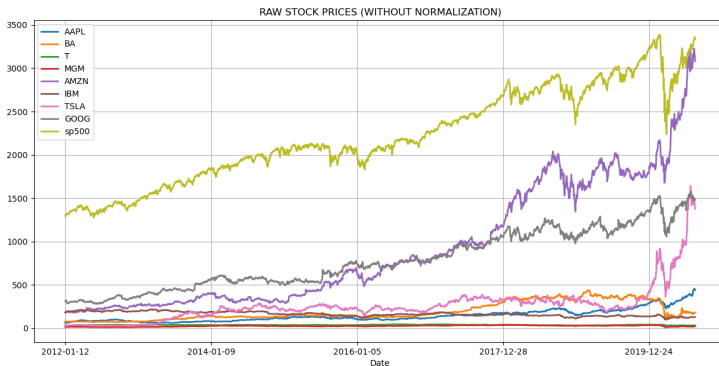
- You've just been hired by a large *old-school* Asset Management Company
- Your team manages a *regulated mutual fund*
- The fund manager asks you to figure out the optimal weights to create a tangency portfolio with some stocks, for which your broker offers no *transaction fees*.
- The data is on Bboard or on *my website*

Task #1 :Import the data and describe the sample

- Sort the stock data by date and print the number of stocks
- Check if data contains any null values
- What is the average market value of the S&P500 ?
- Which stock or index has the minimum dispersion in dollar value ?
- What is the maximum price for Amazon stock over the specified time period ?

Task #2 : Plot the data

- Define a function to plot the entire dataframe
 - The function takes in a dataframe as an input argument and does not return anything back

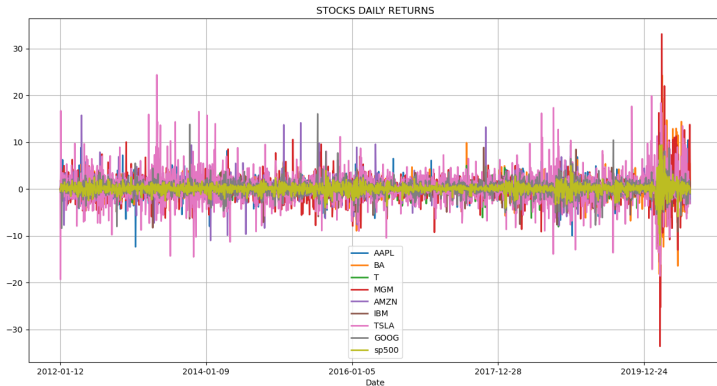


Task #3 : Print out normalized (scaled) stock prices

- Define a function to normalize the prices based on the initial price
 - The function simply divides every stock by it's price at the start date (i.e. : Date = 2012-01-12)
- Plot normalized data
- **Optional** : Define a function to perform an **interactive** data plotting using **plotly express**

Task #5 : Calculate stock returns (1)

- Define a function to calculate stocks daily returns (for all stocks)
 - Loop through each stock
 - Loop through each row belonging to the stock
- Plot it !



Task #5 : Calculate stock returns (2)

- Calculate the correlations between daily returns
- Plot the correlation table as in heat-map
- What are the top 2 stocks that are positively correlated with the S&P500 ?
- What is the correlation between Amazon and Boeing ? Comment on your answer
- What is the correlation between MGM and Boeing ? Comment on your answer

Task #5 : Calculate stock returns (3)

- Plot the histograms of daily returns and comment
- **Optional** : Define a function to perform an **interactive and fancy** histograms plots using **Plotly**
 - Plotly's Python API contains a super powerful module known as figure factory module

Task #6 : Portfolio weights

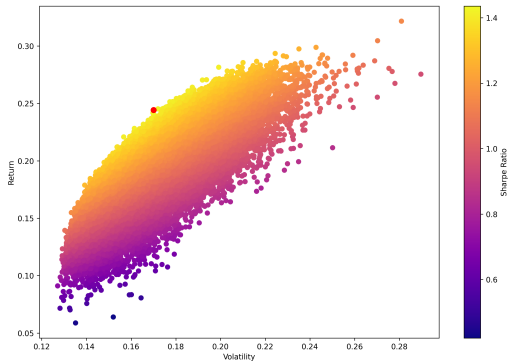
Simulation exercise

- Write a function that will :
 - Simulate 1000 portfolios with random weights
 - Compute the return, the variance and the Sharpe ratio of each portfolio
 - Return the maximum Sharpe ratio and its weights
- What are the weights of your tangency portfolio ? What would have been the performance of this portfolio over the sample period ? Plot it !

Task #6 : Portfolio weights

Simulation exercise

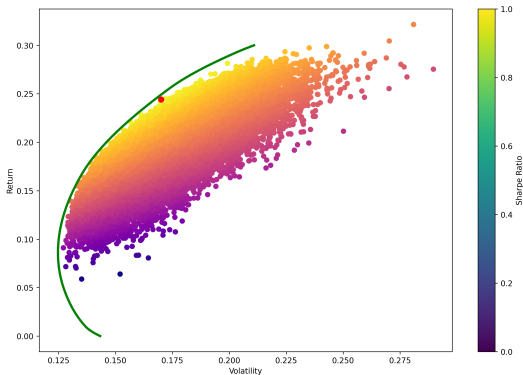
- What are the weights of your tangency portfolio? What would have been the performance of this portfolio over the sample period? Plot it!



Task #6 : Portfolio weights

Simulation exercise

- **Optional** : Define a function that finds the weights that minimize the variance for all possible level or return (i.e., that draw the efficient frontier). Plot it against the previous scatter plot



Task #7 : Testing Portfolio Theory - **Optional**

- Re-run Task #6 but only on the first part of the sample (2012-2016)
- Find out what are the optimal weights for this period
- Simulate your optimal portfolio on the second part of the sample (2016-end)
 - How does it perform ?
 - Plot the expected (measured over the 1st period) vs. the realized return and volatility (over the second).
- **Super optional** : Let's go a step further. Every year, you re-balance your portfolio based on last year data. What is your performance overall ?