SEMINAR PAPER

Investment Analytics in the dawn of Artificial Intelligence

Institute of Financial and Actuarial Mathematics at Vienna University of Technology
TU Vienna

Lecturer

Dipl.-Ing. Dr.techn. Stefan Gerhold

Student

Vyacheslav Gorkunov
01527892

Vienna,
November 29, 2020

Preface

The following seminar paper is based on the book "Investment Analytics in the dawn of Artificial Intelligence", written by Dr. Bernard Lee. The main topic of the book and thus of this seminar paper is the focus on solving problems for professional and institutional asset managers, using methods, that solve the full chain of problems commonly found in institutional multi-asset investing, beginning from asset selection to portfolio rebalancing and implement the methodological gold standard for AI-driven analytics for institutional investing. The methods discovered in the aforementioned topic have their roots in a class of advanced mathematical algorithms which, in turn, have their basics on probability distributions with time subscripts and jumps. Time subscripts and jumps are combined with three-dimensional (3D) data mostly presented as "graphs". On the practical side, that can be applied to solve optimization problems and are involved into machine learning areas. Definitions and data for examples, which will be discussed in this topic, were taken from the book.

Contents

1 Introduction

First of all, we need to find a difference between artificial intelligence(ai) and investment analytics(ia) to answer for meaning of the symbolism behind title ia≠ai?:

> **Investment analytics (ia)**[1] means the process of judging an investment for income, risk, and resale value. It is important to anyone who is considering an investment, regardless of type. Investment analysis methods generally evaluate 3 factors: risk, cash flows, and resale value.
>
> **Artificial intelligence (ai)**[2] refers to the simulation of human intelligence in machines that are programmed to think like humans and mimic their actions. The term may also be applied to any machine that exhibits traits associated with a human mind such as learning and problem-solving. The ideal characteristic of artificial intelligence is its ability to rationalize and take actions that have the best chance of achieving a specific goal.

The remarkable title ia≠ai depictures the issue of the investment analytics considered to be a childish representation of ai techniques well-analyzed and checked in the sphere of engineering, which the press commonly misrepresents. The introduced topic contains the best-sophisticated and the most appropriate techniques, which are used to find the solution for high dimensional problems, which require to deepen into the process, that differs from the nowadays requirements to resolute the ordinary customer problems in modern engineering.

2 Understanding risk

There exists a lot of methods to construct and analyze a portfolio, in this topic I will concentrate only on constructing portfolio with risk calculating and spend attention only on things, which ai cannot accomplish or its role will be just like automatic computations. A lot of experienced investors know from experience that there doesn't exist a thing like perfect foresight and it doesn't matter how successful your asset selection skills are. Logically, risk-free profits can be problematic. If a group of investors could identify such "sure win" investments, then they would either invest so much that prices of the investments will be driven up, thus negating any profits. Considering the alternative way, the leverage providers will formulate the next steps in order to demand higher interest rates and possibly a share of their profits. In any cases described, risk-free profits will transform into risky profits. This will result in a portfolio accompanied with a risk profile and this, as a

[1] https://strategiccfo.com/investment-analysis/
[2] https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp

conclusion, should be supervised by the investors.

The process of the art of handling the risk of a portfolio while the fresh investment ideas are presented to the portfolio and the felicitous investment ideas or stop-loss positions are sold off is called the **portfolio rebalancing**. Therefore, in order to understand portfolio rebalancing, we must start with a better understanding of the techniques to analyse the risk of a portfolio, starting with a review of risk management as a discipline.
According to the analysis made by the experts in the financial sphere, one of the most significant essentials to cause the Global Financial Crisis is the failure for risk models to appropriately fixate extreme events. Hence the correlation for the fundamental problems in risk management can be observed: the estimation of extreme quantiles or determining how the value of a non-normal distributed variable can exceed low probability confidence. The practical application for this observation is the assessment of losses in insurance and Value-at-Risk computations in a financial sphere. Other related measures are:

1) **Value-at-Risk (VaR)**

2) **Conditional Value at Risk (CVaR)**

3) **Return Level**

4) **Maximum Drawdown**

2.1 Value-at-Risk (VaR)

Definition 1. *Given some confidence level $\alpha \in (0,1)$. The VaR of our portfolio at the confidence level α is given by the smallest number l such that the probability that the loss L exceeds l is no larger than $(1-\alpha)$. Formally.*

$$VaR := \inf\{l \in \mathbb{R} : P(L > l) \leq 1 - \alpha\} = \inf\{l \in \mathbb{R} : F(l) \geq \alpha\}$$

In probabilistic terms, VaR is thus simply a *quantile* of the loss distribution. Typical values for α are $\alpha = 0.95$ or $\alpha = 0.99$. Note that by its very definition the VaR at confidence level α does not give any information about the severity of losses which occur with a probability less than 1α. This is clearly a drawback of VaR as a risk measure. The VaR on a trading portfolio can be compared to the capital set aside by the institutions to ensure that any losses.

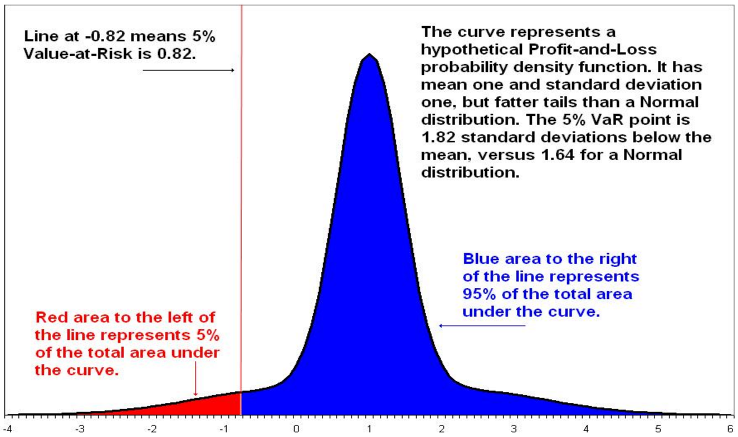

Figure 1 : The illustration of the notion of VaR[1]

2.2 Conditional Value-at-Risk (CVaR)

Definition 2. *Let X be a continuous random variable representing loss. Given a parameter $\alpha \in (0,1)$, the $\alpha - CVaR$ of X is:*

$$CVaR(X) := E[X|X > VaR_\alpha] = E[X|X > F_X^{-1}(\alpha)]$$

It is not usually understandable to choose between VaR and $CVaR$, but some kind of volatile investments can profit from $CVaR$ as a check to the assumptions inflicted by VaR.

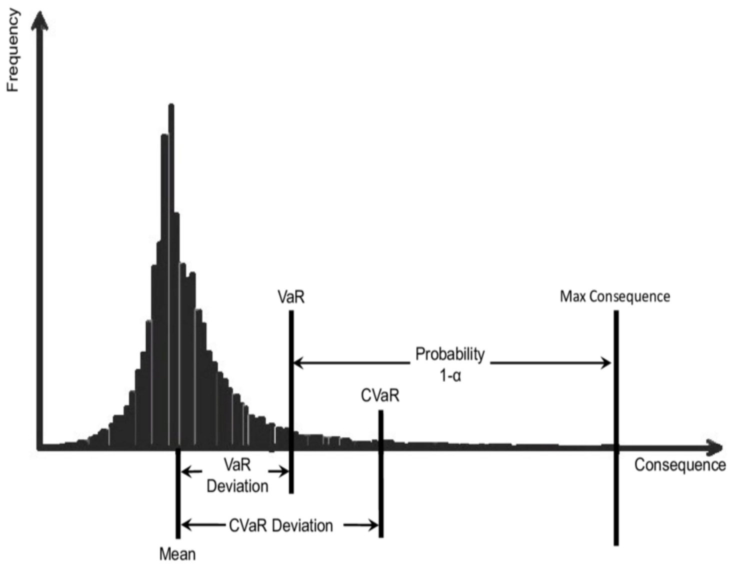

Figure 2 : The illustration of VaR and CVaR representing loss

[1]https://en.wikipedia.org/wiki/Value$_a t_r isk/media/File : VaR_d iagram.JPG$

2.3 Return level

Definition 3. *Return level can be simply described as the level which is foreseen to exceed, but with the condition that, around the average, only once in a sequence of k periods of length n. Hence,* R_k^n *is a quantile-based measure of the distribution function H. We have the aim of finding the event that will occur only once in the case if every k periods and which therefore has prob* $1/k$*. Thus we have* $p = 1/k$

$$R_k^n := H^{-1}(1 - \frac{1}{k})$$

2.4 Maximum Drawdown

Definition 4. *Drawdown can be explained as the measure of a variable's decline from its most recent historical peak. On the practice, the comparison of a drawdown and a cumulative profit is evaluated. The evaluation guarantees the existence of satisfactory equity available to finance a portfolio strategy. The translation of the maximum drawdown(MDD) up to time* T *is, hence, the maximum of the peak-to-trough loss in the variable's value over a time interval* $(0, T)$*.*

$$MDD(0,T) = max_{t \in (0,T)}\{max_{\tau \in (0,T)}[X(t) - X(\tau)]\}$$

3 Related Risk Modeling Techniques

In this subsection, we shortly consider the main causation for the 2008 Global Financial Crisis. The crisis resulted after a series of events, such as subprime mortgages, credit default swaps, and high leverage ratios. One of the important causations is also the lack of concerns for the financial institutions for the large potential downside risks, moreover, the shortcomings of normally distributed models remained unanalysed. The real financial system appeared to be under higher risk than the predicted financial system. The real financial system was created by the concept of visible profits and invisible losses, which, in turn, had the analytical background with 99.7% of asset variations fall within 3 standard deviations of their mean under the assumption of normality.

3.1 Fat tails

Defining a **fat-tailed distribution**, it is a probability distribution that exhibits a large skewness or kurtosis, relative to that of either a normal distribution or an exponential distribution. The fundamental for the protection of portfolio return underlies in construction of the portfolio in three Related Risk Modeling Techniques. These ways, as a priority, minimizes the tail risk exposure from unpredictable market events. The distribution of a random variable X is defined to have a fat tail in the case if:

$$Pr[X > x] \sim x^{-\alpha}, \alpha > 0$$

i.t., if X has probability density function $f_X(x)$ described as:

$$f_X(x) \sim x^{-(\alpha+1)}, \alpha > 0$$

Here " $\sim$ " refers to the asymptotic equivalence of functions.

3.2 Uniform Margins

For the future extreme events, the construction of the prediction or estimation for them is the complicated process, due to the wide range of variables used in order to construct the model. Considering a more detailed overview, the accuracy for the corresponding predictions is less than for the prediction time horizon lengthens, this can be found through a probabilistic approach. Within the framework of the practice, we will run the estimation, where assuming every risk variable is distributed in the uniform margins, moreover, decreasing the accuracy with the predictions for the future. In addition, the fact that losses are less frequent but more extreme leads to the estimation for credit and operational risk with the skewed distributions. In case the extreme event is not taken place, the other event has the observed tendency to move to another event. Describing more practically, the marginal distribution of the time series is approximated with ignoring the dependence structure in the data, if it is present there. Nevertheless, the assumption of independence described previously has no grounds to believe in, since it may not be sufficient enough.

Definition 5. *The marginal distribution function of a time series $X_t, t = 1, ..., T$ can be described as follows*[1]*:*

$$F_t(x) = P(X_t \leq x) = F(\infty, ..., \infty, x, \infty, ..., \infty)$$

where $F(x_1, ..., x_t, .., x_T)$ is the joint pdf of the time series. In practice, the marginal distribution of a time series is often approximated by ignoring the dependence structure that may exist in the data.

3.3 Arrival Times

Poisson distribution can be used to model operational risk.

[1] Definition from B.Lee, Investment analytics in the dawn of artificial intelligence, p.34

Definition 6. *A discrete random variable X is said to have a Poisson distribution with parameter $\lambda > 0$ if for $k = 0, 1, 2, ...,$ the probability mass function of X is given by*[2]*:*

$$f(k; \lambda) = P(X = k) = \frac{\lambda^k e^{-\lambda}}{k!}$$

3.4 Empirical Observations

Consistently, they are depictured as the distribution function connected with the empirical measure of a sample. The following cumulative distribution function presents oneself as a step function which increases be $1/n$ at every data point. The value of this step function at any specified value of the measured variable represents the fraction of observations for the measured variables that are less than or equal to the already specified value. Commonly, some period of a date is taken and the graphs our assets are analyzed in order to compare their losses.

Definition 7. *Let $(X_1, ..., X_n)$ be independent, identically distributed real random variables with the common cumulative distribution function $F(t)$. Then the empirical distribution function is defined as*[3]*:*

$$\hat{F}_n(t) = \frac{\text{number of elements in the sample} \leq t}{n} = \frac{1}{n} \sum_{i=1}^{n} \mathbb{1}_{X_i \leq t}$$

4 Objective Functions in Portfolio Construction

Now we are acquainted with all necessary techniques and risk measures, so we can look at some examples with using them. As a professional investment managers, we have special long-term targets on the portfolio‘s risk and returns, and we want to better understand how we can deliver a potential improvement on the portfolio‘s return comparative to downside risk. We understand that we need to diversify our portfolio by allocating our investments among multiple asset classes and assets. A bundle of stocks has to be taken, such as it will be predicted to rise in value in terms of the long-term duration and, moreover, relatively uncorrelated. Our question is: How better to provide a solid relationship between ex-ante risk on a portfolio and its realized returns?
We will discuss the 6 most commonly used statistics to demonstrate what can be achieve using these statistics and how to model proper objective functions (MDD, VaR, CVaR, Maximum Sharpe ratio, Maximum Alternative Sharpe ration, minimum Variance) based

[2]https://en.wikipedia.org/wiki/Poisson$_{d}istribution$
[3]https://en.wikipedia.org/wiki/Empiricaldistributionfunctio

on them.
First of all, we define some necessary values, which we will use for our examples:
Let[1]

1) $A(t)$ be the day-t mark-to-market value of the existing portfolio. In our case this is our portfolio

2) $X_{cumP\&L}(t)$ be the cumulative P&L of the target assets on day t, assuming $X_{cumP\&L}(0) = 0$. In our case, this will be the profit and loss of the portfolio combined with buy and sell recommendations.

3) h be the recommended position of a specific asset in the rebalancing basket

A historical data of stocks for the following examples was taken for the period 05.07.2013-08.07.2016 .

4.1 Minimum Peak-to-Trough MDD

Minimum Peak-to-Trough MDD minimizes the peak-to-trough MDD of the market value of the portfolio[2].

$$min_h MaxDD(T; \{A(t) + hx_{cumP\&L}(t)\}_{t=0}^{T})$$

Where $MTM(t) = A(t) + hx_{cumP\&L}(t)$ is the dollar mark-to-market value of the brokerage account at the end of the period $[0,t]$, $MTM^{peak}_{cum,0\leq\tau<t} = max_{\{}0 \leq t \leq t\}[MTM(\tau)]$ is the maximum dollar mark-to-markte value in the period $[0,t]$, $DD(t) = MTM^{peak}_{0\leq t\leq t} - MTM(t)$ and $MaxDD(T; \{MTM(t)\}_{t=0}^{T}) = max_{0\leq t\leq T}\{DD(t)\}$ is the MaxDD in the period [0,T].

Example 1. We want to build a portfolio, which is consisted from Starbucks Corporation (SBUX), Apple Inc.(AAPL), Alphabet Inc.(GOOG) stocks. Over a historical period we will use MDD as a measure of risk. We can calculate MDD in percentage measure, using the formula:

$$MDD\% = \frac{TroughValue}{RespectivePeakValue} - 1$$

Our portfolio has 18 levels of returns. We draw the efficient frontier, which will connect points, that are based on $minDD$. After, we keep only half of the observations above the horizontal line where the minimally required portfolio return is achieved comparing with all $MDDs$.

[1]Definitions from B.Lee, Investment analytics in the dawn of artificial intelligence, p.41
[2]Definition from B.Lee, Investment analytics in the dawn of artificial intelligence, p.42

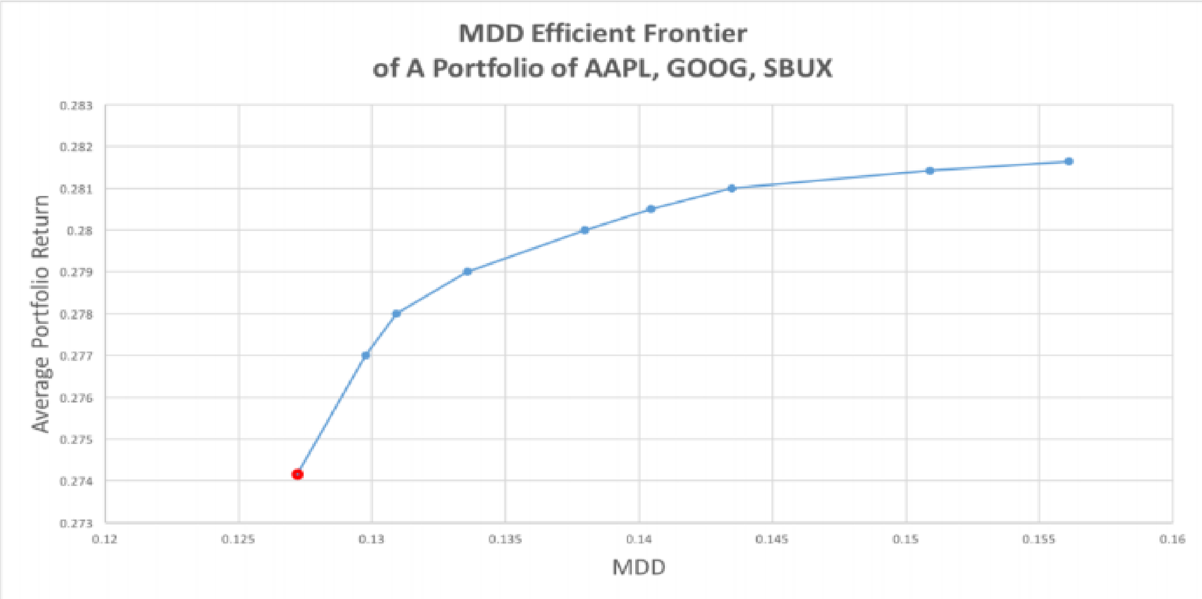

Figure 3 : Minimizing Maximum Drawdown[3]

4.2 Minimum 95% Value-at-Risk

Minimum 95% Value-at-Risk (in other words "Min Potential Loss") can be computed from the daily P&L of the portfolio[4]:

$$min_h VaR(A_{P\&L}(t) + hx_{P\&L}(t))$$

where $A_{P\&L}(t) = A(t) - A(t-1)$ and $x_{P\&L}(t) = x(t) - x(t-1)$.

Example 2. We use the same portfolio from the **Example 1**. First of all, we have to calculate the daily P&L of the portfolio. After getting historical returns we sort them in the ascending order and calculate the treshold, that is value of historical $VaR_{95\%}$, with $index = integer[5\% * count(observations)]$. We plot the efficient frontier of $VaR_{95\%}$ relatively levels of portfolio returns.

[3]The graph was replicated from B.Lee, Investment analytics in the dawn of artificial intelligence, p.44
[4]Definition from B.Lee, Investment analytics in the dawn of artificial intelligence, p.44

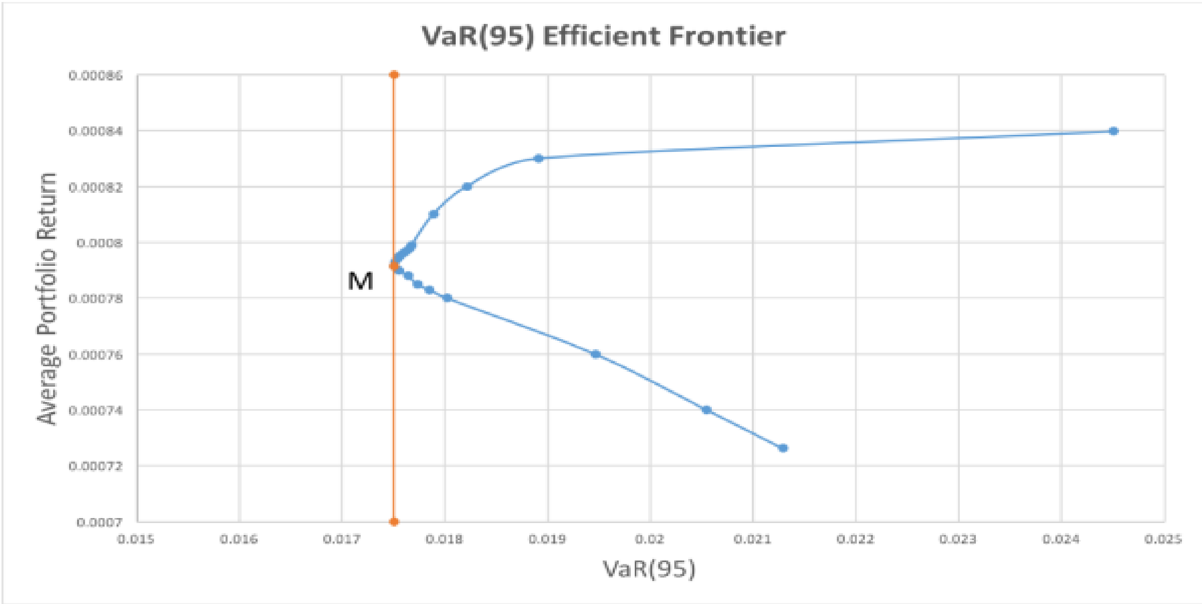

Figure 4 : Minimizing VaR 95%[5]

Here we see the line, which goes through point M. This point is the lowest level of $VaR_{95\%}$, which the portfolio can achieve based on historical data.

4.3 Minimum 95% Conditional Value-at-Risk

The Conditional Value-at-Risk at 95% like the Value-at-Risk at 95% is computed from the daily P&L of portfolio and is defined as[6]:

$$min_h CVaR(A_{P\&L}(t) + hx_{P\&L}(t))$$

where $A_{P\&L}(t) = A(t) - A(t-1)$ and $x_{P\&L}(t) = x(t) - x(t-1)$.

With the Cornish-Fisher expansion[7], we can compute $CVaR$, where

$$CVaR_{95\%}(.) = -[\mu(.) + \sigma(.)\mathbb{E}(z_{cf,1-k}|k > 95\%)]$$

We can express $\mathbb{E}(z_{cf,1-k}|k > 95\%)$ through the values S and K, what are skewness and excess kurtosis from computed the daily P&L of the portfolio:

$$\text{CVaR}_{95\%}(.) = -\mu(.) - \sigma(.)\mathbb{E}\left[\begin{array}{c|c} z_{C(1-k)} + \frac{1}{6}(z^2_{C(1-k)} - 1)S(.) & \\ +\frac{1}{24}(z^3_{C(1-k)} - 3z_{C(1-k)})K(.) & k > 95\% \\ -\frac{1}{36}(2z^3_{C(1-k)} - 5z_{C(1-k)})S(.)^2 & \end{array}\right]$$

[5]The graph was replicated from B.Lee, Investment analytics in the dawn of artificial intelligence, p.45
[6]Definition from B.Lee, Investment analytics in the dawn of artificial intelligence, p.46
[7]https://en.wikipedia.org/wiki/Cornish-Fisher$_{e}xpansion$

Here $z_{C(1-k)}$ is the critical value for probability $1-k$ with standard normal distribution. At $k = 95\%$ $z_{C(1-k)} = -1.64$.

Example 3. As in previous examples we want to construct the portfolio from Starbucks Corp.(SBUX), Apple Inc.(AAPL) and Alphabet Inc.(GOOG). Firstly we have to find the min and max of the portfolio returns(possible values) and select 18 or 19 levels of returns to find an allocation of weights, what minimize $CVaR$. Then we compute $minCVaR_{95\%}$ at each value of daily P%L and draw the efficient frontier, which connects those points.In the end, we find a tangent to this efficient frontier. The point M, where $CVaR_{95\%}$ takes the minimum, is the final goal of this objective function.

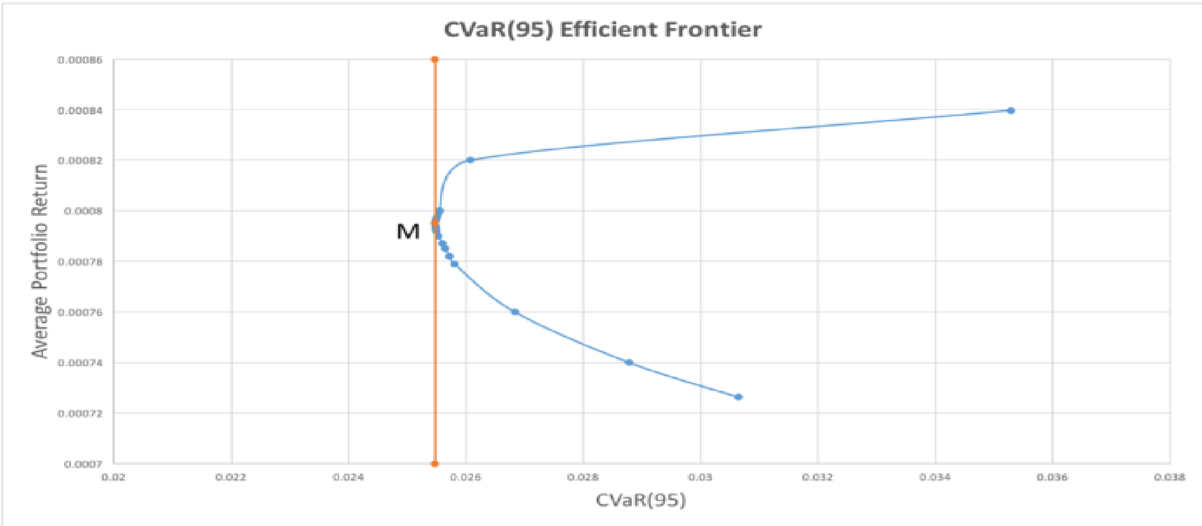

Figure 5 : Minimizing CVaR 95%[8]

4.4 Maximum Sharpe Ratio

In order to provide support investors with the understanding of the return of an investment compared to the risk of the investment, the **Sharpe ratio** is introduced. Sharpe ratio is defined as the average return earned in excess of the risk-free rate per unit of volatility or overall risk. Term 'Volatility' basically means the measure of the price fluctuations of an asset or portfolio[9]. The different option to objective functions presented above, we could maximize the Sharpe Ratio of the traditional portfolio :

$$SR \equiv \frac{\sum_i e_i \pi_i}{\sigma_\pi}$$

[8]The graph was replicated from B.Lee, Investment analytics in the dawn of artificial intelligence, p.48

[9]Definition from https://www.investopedia.com/terms/s/sharperatio.asp

where e_i is excess return rate of the i-th asset of the portfolio π and π_i is i-th position of the portfolio π.

Example 4. We want to construct a portfolio from bunch of IBM Corp.(IBM), Alphabet Inc.(GOOG) and Apple Inc.(AAPL) stocks. We need to find a such combination, that will give to us the portfolio with as high a Sharpe Ratio as possible. First, we calculate the min Standard Deviation at 14 levels of average portfolio returns. Then we will connect them to get the efficient frontier. The point M is where the portfolio achieves its max Sharpe Ratio. Also, this point on the graph intercepts the tangent Capital Market line, which represents the risk-free rate of return, what is assumed to be 2% in this example. Therefore, it achieves the goal of this objective function.

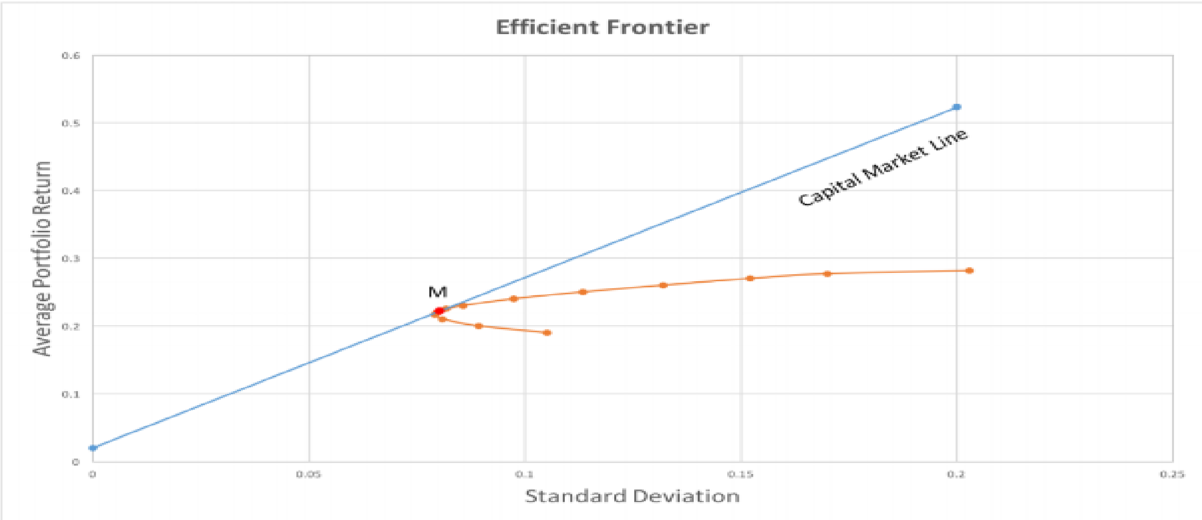

Figure 5 : Maximizing Sharpe Ratio[10]

4.5 Maximum Alternative Sharpe Ratio

The Alternative Sharpe Ratio affords to find an optimal balance between "upside moments" and "downside risk", not only minimizing risk or returns.

$$ASR \equiv \frac{\sum_i e_i \pi_i}{z_\pi^- \sigma_\pi} + \frac{1}{2}\frac{\sum_i \pi_i (z_i^+ \sigma_i)^2}{z_\pi^- \sigma_\pi} - \frac{1}{2} z_\pi^- \sigma_\pi$$

where $z_i^+ = \frac{max(z_{cf}(z_C^+(i)),0)}{z_C^+}$, z_C^+ is the critical value for probability k and $z_i^+ = \frac{min(z_{cf}(z_C^-(i)),0)}{z_C^-}$, z_C^- is the critical value for probability $1-k$

Example 5. We want to construct a portfolio from bunch of IBM Corp.(IBM), Alphabet Inc.(GOOG) and Apple Inc.(AAPL) stocks with as high ASR as possible. In order to calculate ASR we need all the inputs of the model. We calculate for the individual stocks and

[10]The graph was replicated from B.Lee, Investment analytics in the dawn of artificial intelligence, p.49

portfolio their average return, standard deviations and Cornish-Fisher expansion at 99% C.I. Then, we plug other parameters into ASR formula.
We choose 13 levels of return, then we calculate the max of ASR at each level:

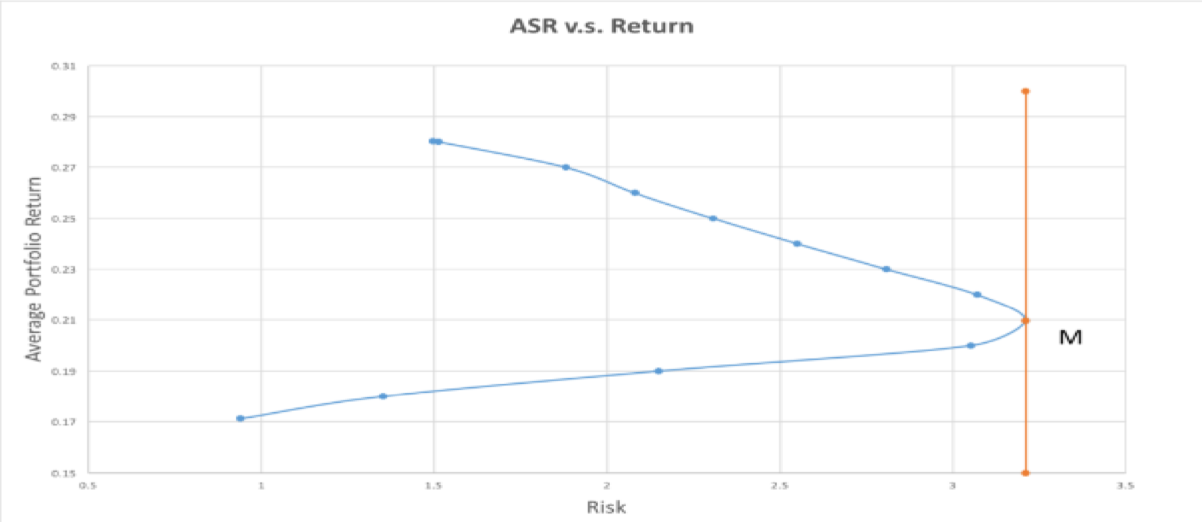

Figure 6 : Maximizing ASR[11]

The line, which connects all these levels, intercepts the point M, what is the goal of this function.

4.6 Minimum Variance

Let $pherr(t)$[12] be the percentage changes in mark-to-market (MTM) value of the portfolio. It is defined as:

$$pherr(t) = \frac{MTM(t)}{MTM(t-1)} - 1 = \frac{A(t) + hx_{cumP\&L}(t)}{A(t-1) + hx_{cumP\&L(t-1)}} - 1$$

where $A(t)$ is MTM-value of the day t, $x_{cumP\&L}(t) = x(t) - x(0)$ with $x(0) = 0$. The minimum variance we can compute, using the formula: $min_h \sum_{1 \leq t \leq T}(pherr(t))^2$

Example 6. We want to construct a portfolio with Starbucks Corp. (SBUX), Apple Inc.(AAPL) and Alphabet Inc.(GOOG). We can calculate the variance of the portfolio with the formula $\mathbb{V} = \frac{\sum(x-\mu)^2}{N}$ with values of the portfolio observations: X is the yearly return, N is the count and μ is the mean, using the historical market data of our stocks, to minimize variance, performing it as an indicator of the volatility. To draw the graph we

[11]The graph was replicated from B.Lee, Investment analytics in the dawn of artificial intelligence, p.51
[12]Definition from B.Lee, Investment analytics in the dawn of artificial intelligence, p.41

have to change the weights of each stock in our portfolio and find min Variance at each level of the portfolio returns. Here 14 levels are enough.

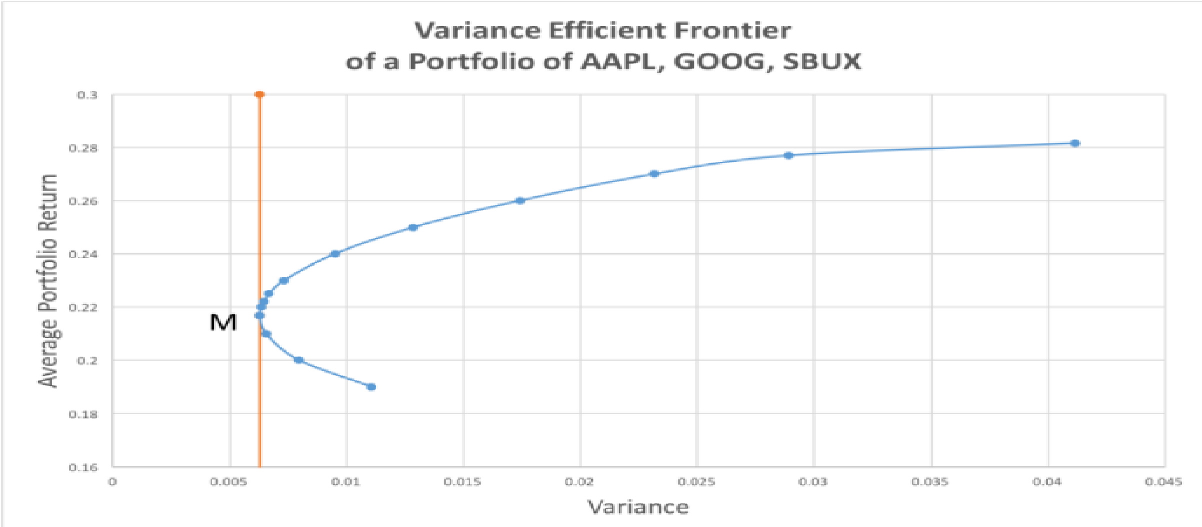

Figure 7 : Minimizing Variance[13]

We connect the points of min Variance at each level and find the point M, which intercepts the efficient frontier line, is our goal.
In the all examples above, we used a historical data to calculate the objective functions. Although this can be valid only under the assumption that no regime shifts have happened. However, this assumption is not always correct. Investment managers can merge these methods to produce more solid portfolio rebalancing recommendations, in the presence of fundamental changes.

5 Brinson Attribution

If we want in our portfolio to identify positions with potential exposure to extreme risk and we want to make sure that portfolio will not be harmed by some extreme scenarios, e.g. financial crisis, changing of world market due to bad economic performance in China and etc. We have to define positions, which can contribute the most to portfolio tail-risk. The better way to do this is using **Risk Attribution** and **Ex-Ante Return Attribution**.

In terms of providing to the investors the opportunity to investigate which investments contribute disproportionately to portfolio tail risk corresponding to their weights, the **Risk Attribution** makes the comparison of the position weight of each investment with its Percentage Contribution to Tail Risk (PCTR). It's defined as:

[13]The graph was replicated from B.Lee, Investment analytics in the dawn of artificial intelligence, p.43

$$\%RC_i = \frac{\omega_i}{R(\vec{\omega})} \frac{\partial R(\vec{\omega})}{\partial \omega_i}$$

Where R is first-order homogeneous risk measure (e.g. $R(\lambda\omega) = \lambda R(\omega)$)

The difference for the calculations made by **Ex-Anter Return Attribution**, which are, however, closely similar to the described previous calculations, defines within everything shown in return space, nor in the risk space. In order to clarify, as an example, the break-even or implied return for an asset, or how much return the asset has to receive for the sake of paying for its tail risk contribution.

$$\sum_i \omega_i e_i \frac{\%RC_i}{\omega_i} = \frac{\sum_i \omega_i e_i}{R(\vec{\omega})} \frac{\partial R(\vec{\omega})}{\partial \omega_i}$$

where e_i can be either the Scenario Return or historical return of an asset.

Typically **Ex-ante Return Attribution** is accompanied by **Brinson Attribution**. This attribution usually is used to explain the difference between portfolio's efficiency and benchmark (*Benchmark*[1] is a standard against which the performance of an asset can be measured). Based on stock selection and asset allocation, it allows to evaluate the performance of the portfolio manager.

Month to Date	2018-01-31	2018-02-07	
Region	ATTRIBUTION		
	Selection	Allocation	Interaction
CN	-0.92%	-0.56%	-0.17%
HK	0.08%	-0.12%	0.01%
IN	-0.14%	0.36%	0.09%
KR	0.00%	-0.55%	0.01%
Other (SG, MY)	0.22%	0.62%	-0.19%
TW	-0.08%	-0.18%	-0.03%
Total	-0.84%	-0.43%	-0.27%
		Active return	-1.54%

Quarter to Date	2017-12-29	2018-02-07	
Region	ATTRIBUTION		
	Selection	Allocation	Interaction
CN	-1.51%	0.23%	-0.28%
HK	-0.11%	-0.04%	-0.02%
IN	-0.33%	0.16%	0.20%
KR	-0.51%	-0.32%	-0.13%
Other (SG, MY)	0.31%	-0.05%	-0.26%
TW	0.04%	0.03%	0.00%
Total	-2.11%	0.01%	-0.48%
		Active return	-2.58%

Figure 8 : Typical worked example of Brinson Attribution

Those techniques can help investors to find strong signals in their portfolio for a potential increase or decrease. We can consider an one worked example. Assume we went to put in 65% of our assets into Apple Inc. (APPL) stocks and remaining 35% into iShares Barclays Aggregate Bond Fund (AGG), where 60% of the portfolio's benchmark is MSCI World (MSCIW) and last 40% is Barclays Aggregate Indices (BA). Our choice is based on assumptions that Apple will exceed MSCI World Index and will be 5% overweight to shares. Let's illustrate this on the table:

[1]Definition from https://www.investopedia.com/terms/b/benchmark.asp

		Table 1			
Benchmark			Portfolio		
	Weight	Return		Weight	Return
MSCI World	60%	9.94%	Apple	65%	28.80%
Barclays Aggregate	40%	-0.84%	AGG	35%	-0.84%
Total	100%	5.62800%	Total	100%	18.42600%
				Outperformance	12.79800%

Figure 9 : Worked Table 1 Example

For the calculations there will be used a method of weighted mean (mwm). Firstly we will define names in the Table a bit shorter. Let it **Total Out-performance** be **TOP**, **Total Portfolio Return** be **TPR**, **Total Benchmark Return** be **TBR**.
In the Table 1 **TOP** = **TPR** - **TBR**, so with the numbers it will be $18.426\% - 5.628\% = 12.798\%$. For calculating **TBR** and **TPR** we can use mwm. It follows **TBR** $= 60\% * 9.94\% + 40\% * (-0.84)\% = 5.628\%$ and **TPR** $= 65\% * 28.80\% + 35\% * (-0.84)\% = 18.426\%$

		Table 2			
Benchmark			Portfolio		
	Weight	Return		Weight	Return
MSCI World	65%	9.94%	Apple	65%	28.80%
Barclays Aggregate	35%	-0.84%	AGG	35%	-0.84%
Total	100%	6.16700%	Total	100%	18.42600%
				Asset Selection	12.25900%
MSCI World	60%	9.94%	MSCI World	65%	9.94%
Barclays Aggregate	40%	-0.84%	Barclays Aggregate	35%	-0.84%
Total	100%	5.62800%	Total	100%	6.16700%
				Asset Allocation	0.53900%
				Outperformance	12.79800%

Figure 10 : Worked Table 2 Example

According to Table 2 let it **Total Adjusted Benchmark Return** be **TABR**, **Asset Allocation** be **AA** and **Asset Selection** be **AS**. The sum of **AA** and **AS** equals to **TOP**. For the first bet, where AAPL performs better than MSCIW, assuming that the weights of MSCIW and BA are same as AAPL with AGG, we can calculate performance of **AS**. It follows that **TABR** $= 65\% * 9.94\% + 35\% * (-0.84\%) = 6.167\%$ and **TPR** $= 65\% * 28.80\% + 35\% * (-0.84\%) = 18.426\%$. Finally, **AS** = **TPR** - **TABR** $= 18.426\% - 6.167\% = 12.259\%$. For the second bet, where **AA** based on 5% overweight to shares, we compare returns in both cases, there are 60% MSCIW with 40% BA and 65% MSCIW with 35% BA. Therefore,**AA** $= 65\% * 9.94\% + 35\% * (-0.84)\% - 60\% * 9.94\% + 40\% * (-0.84)\%$
$= 6.167\% - 5.628\% = 0.539\%$. Finally, in the second table **TOP** = **AS** + **AA** = $12.259\% + 0.539\% = 12.798\%$, which is identical to **TOP** in the first table.

Risk Attribution and **Return Attribution** help investors to decide to increase or decrease weights and to find an appropriate asset, which could give more profit and less risk.

6 Market prediction using AI algorithms

6.1 OLS Regression

With the development of computer technologies investment experts, who usually has a mathematical degree, build complex models to construct and analyze portfolio using a lot of instruments, which a fund, wherein their work, provides a platform, what affords to do it. In the world there are a lot of examples such funds: *Bridgewater Associates*, *Renaissance Technologies*, *BlackRock* and etc. Their success is due to integration of computer-mathematical models. Nowadays to become successful investment manager without programming skills is very problematic. Those computer-mathematical models because of a big amount of date requires titanic computing power. In our time AI technologies represent a "mix" between machine learning(ML) algorithms and neural networks. A lot of hedge fund platforms are based on those algorithms. The idea of the machine learning - to write a self-study algorithm, which computes predictions based on given historical data. ML algorithms use "simple" functions, which with special libraries(depends on programming language) and with given historical data can train a model, that can computes predictions for a new data set, what can give to an investor a clue for constructing portfolio.
To make some investment decisions for constructing the portfolio, investment managers may use ML-algorithms to try to predict the motion of the determined stocks in the future, using their historical data. It allows the investors to decide, what kind of stocks may probably rise up with lower risk. Without a doubt, I can't show the code, which can give us 100% accuracy and of course this code shouldn't be used for investing, although the following example can demonstrate what kind of idea of the AI/ML - algorithms the hedge fund's platforms use for different investment tasks. The following example uses the method of Ordinary Least Squares (OLS) Regression to the performance of the neural networks. The idea of the code was taken from the book *"Yves Hilpisch, Artificial Intelligence in Finance. A Python-Based Guide, 1st edition 2020, p.192-199, chapter 6 Market Prediction based on Returns Data"*
First of all, we will define the meaning of OLS Regression:

Let the $\{X\}_{i=0}^{k}$ be the explanatory variables, Y be a dependent variable, then for each sample n:

$$y_n = \sum_{i=0}^{k} \beta_i x_{ni} + \epsilon_n$$

where β can be found by minimizing the error of prediction. OLS Regression is a linear regression.

6.2 Market Prediction

For the following code the priority goal lies in the comparison to the performance of OLS with the performance of neural networks in order to predict the following's day direction of movement for the distinctive time series. On this stage, the aim can be described as identifying statistical inefficiencies as correlated in a contrast with economic inefficiencies. Statistical inefficiencies are used for a model, while it has the ability to envision the direction of the future price movement with a certain benefit. Moreover, we assume that the prediction holds true in 55% or 60% of all of the cases used. In the only case when the statistical inefficiencies can be productively applied with the use of a trading strategy, the economic inefficiencies would be given. Giving a precise example, a trading strategy may consider transaction costs.

First of all, we import all necessary *Python* libraries, which are used for ML and neural networks:

```
#import necessary libraries
import numpy as np
import pandas as pd
from sklearn.metrics import accuracy_score # computes subset accuracy
from sklearn.neural_network import MLPRegressor #optimizes the squared-loss using LBFGS or stochastic gradient descent
import tensorflow as tf # library for training and inference of deep neural networks
from keras.layers import Dense # regular deeply connected neural network layer
from keras.models import Sequential # is appropriate for a plain stack of layers

data = pd.read_csv('data.csv', index_col=0, parse_dates=True).dropna()#data without N.A. values
```

Screen 1 : Import Python libraries

The *data.csv* contains market prices for Microsoft Inc (MSFT.O), Apple Inc.(AAPL.O), Intel Corp. (INTC.O), Amazon Inc.(AMZN.O), Goldman Sachs Group Inc. (GS.O), SPDR S&P 500 ETF Trust (SPY), S&P 500 Index (.SPX), VIX Index (.VIX), Euro to Dollar (EUR=), Gold Spot US Dollar (=XAU), VanEck Vectors Gold Miners Etf (GDX) and SPDR Gold Trust (GLD) in the period between 1 January 2010 and 1 January 2020:

	AAPL.O	MSFT.O	INTC.O	AMZN.O	GS.N	SPY	.SPX	.VIX	EUR=	XAU=	GDX	GLD
Date												
2010-01-04	30.572827	30.950	20.88	133.90	173.08	113.33	1132.99	20.04	1.4411	1120.00	47.71	109.80
2010-01-05	30.625684	30.960	20.87	134.69	176.14	113.63	1136.52	19.35	1.4368	1118.65	48.17	109.70
2010-01-06	30.138541	30.770	20.80	132.25	174.26	113.71	1137.14	19.16	1.4412	1138.50	49.34	111.51
2010-01-07	30.082827	30.452	20.60	130.00	177.67	114.19	1141.69	19.06	1.4318	1131.90	49.10	110.82
2010-01-08	30.282827	30.660	20.83	133.52	174.31	114.57	1144.98	18.13	1.4412	1136.10	49.84	111.37

Screen 2 : Show the data

In the analysis, the first stage means the creation of data sets, in which lagged log-returns data is established. Moreover, the testing is also applied to the normalized lagged log-returns data. The testing meaning is to analyze for stationarity, which is given, and the not correlated features for correlation. Taking into consideration the following analysis, it relies only on time-series-related data, and, moreover, the data is working with the weak-form market efficiency:

```
def add_lags(data, ric, lags):
    cols = []
    df = pd.DataFrame(data[ric])
    for lag in range(1, lags + 1):
        col = 'lag_{}'.format(lag)
        df[col] = df[ric].shift(lag)
        cols.append(col)
    df.dropna(inplace=True)
    return df, cols
#lagging indicators offer a historical report of background
#conditions that resulted in the current price being where it is
log_ret = np.log(data / data.shift(1))#log returns from the price data
log_ret.dropna(inplace=True)#Drop N.A. values
lags = 7 # The number of lags (in trading days)
modif_data = {}
for sym in data:
    df, cols = add_lags(log_ret, sym, lags)#Lags the log returns data
    mu, std = df[cols].mean(), df[cols].std()#z-score normalization to the features data
    df[cols] = (df[cols] - mu) / std
    modif_data[sym] = df
```

Screen 3 : functions to lag log returns

Lagged data looks like:


```
def add_lags(data, ric, lags):
    cols = []
    df = pd.DataFrame(data[ric])
    for lag in range(1, lags + 1):
        col = 'lag_{}'.format(lag)
        df[col] = df[ric].shift(lag)
        cols.append(col)
    df.dropna(inplace=True)
    return df, cols
#lagging indicators offer a historical report of background
#conditions that resulted in the current price being where it is
log_ret = np.log(data / data.shift(1))#log returns from the price data
log_ret.dropna(inplace=True)#Drop N.A. values
lags = 7 # The number of lags (in trading days)
modif_data = {}
for sym in data:
    df, cols = add_lags(log_ret, sym, lags)#Lags the log returns data
    mu, std = df[cols].mean(), df[cols].std()#z-score normalization to the features data
    df[cols] = (df[cols] - mu) / std
    modif_data[sym] = df
```


Screen 4 : log lag returns

We test new data for stationarity and find the correlation for the features:

```
adfuller(modif_data[sym]['lag_1'])#Tests for stationarity of the time series data
(-51.56825150582556,
 0.0,
 0,
 2507,
 {'1%': -3.4329610922579095,
  '5%': -2.8626935681060375,
  '10%': -2.567384088736619},
 7017.165474260225)
```

```
modif_data[sym].corr()
```

	GLD	lag_1	lag_2	lag_3	lag_4	lag_5	lag_6	lag_7
GLD	1.000000	-0.029691	0.000300	0.012635	-0.002562	-0.005939	0.009852	-0.001343
lag_1	-0.029691	1.000000	-0.030501	0.000814	0.012765	-0.002876	-0.005323	0.009804
lag_2	0.000300	-0.030501	1.000000	-0.031617	0.000320	0.013234	-0.004335	-0.005237
lag_3	0.012635	0.000814	-0.031617	1.000000	-0.031329	-0.000007	0.014115	-0.004387
lag_4	-0.002562	0.012765	0.000320	-0.031329	1.000000	-0.031761	0.000226	0.014067
lag_5	-0.005939	-0.002876	0.013234	-0.000007	-0.031761	1.000000	-0.032289	0.000217
lag_6	0.009852	-0.005323	-0.004335	0.014115	0.000226	-0.032289	1.000000	-0.032351
lag_7	-0.001343	0.009804	-0.005237	-0.004387	0.014067	0.000217	-0.032351	1.000000

Screen 5 : Test for stationarity and show the correlation

At the first point, the OLS regression is implemented in order to generate the final predictions, which are produced from the regression. The important aspect is that the analysis is achieved on the complete data set. The aim of the analyses made is to illustrate the performance of the algorithms in the sample. Moving to the accuracy of the predictions made with OLS regression, the direction of movement for the next day is a few percentage points, even above 50% with one exemption:

```
for sym in data:
    df = modif_data[sym]
    reg = np.linalg.lstsq(df[cols], df[sym], rcond=-1)[0]#the regression
    pred = np.dot(df[cols], reg)#the prediction
    acc = accuracy_score(np.sign(df[sym]), np.sign(pred))#the accuracy of the prediction
```

```
OLS | AAPL.O     | acc=0.5056
OLS | MSFT.O     | acc=0.5088
OLS | INTC.O     | acc=0.5040
OLS | AMZN.O     | acc=0.5048
OLS | GS.N       | acc=0.5080
OLS | SPY        | acc=0.5080
OLS | .SPX       | acc=0.5167
OLS | .VIX       | acc=0.5291
OLS | EUR=       | acc=0.4984
OLS | XAU=       | acc=0.5207
OLS | GDX        | acc=0.5307
OLS | GLD        | acc=0.5072
```

Screen 6 : Accuracy of OLS

For the second point, the analogue of the previous analysis has to be made. The difference is that the neural network from scikit-learn as the model for learning and predicting is used at this stage. Considering the prediction accuracy here, in-sample is reasonably higher than 50% and even higher 60% in some of the cases:

```
for sym in data.columns:
    df = modif_data[sym]
    model = MLPRegressor(hidden_layer_sizes=[512],
    random_state=100,
    max_iter=1000,
    early_stopping=True,
    validation_fraction=0.15,
    shuffle=False)
    model.fit(df[cols], df[sym])
    pred = model.predict(df[cols])
    acc = accuracy_score(np.sign(df[sym]), np.sign(pred))
```

```
MLP | AAPL.O      | acc=0.6005
MLP | MSFT.O      | acc=0.5853
MLP | INTC.O      | acc=0.5766
MLP | AMZN.O      | acc=0.5510
MLP | GS.N        | acc=0.6527
MLP | SPY         | acc=0.5419
MLP | .SPX        | acc=0.5399
MLP | .VIX        | acc=0.6579
MLP | EUR=        | acc=0.5642
MLP | XAU=        | acc=0.5522
MLP | GDX         | acc=0.6029
MLP | GLD         | acc=0.5259
```

Screen 7 : Accuracy of MLP

Repeating the same analysis procedure, however, with the use of the neural network from the *Keras* package. In the third stage, the accuracy evaluation is quite identical to the results from the *MLPRegressor* and the only difference is in the higher average accuracy:

```
def create_model(problem='regression'):#Model creation function
    model = Sequential()
    model.add(Dense(512, input_dim=len(cols),
    activation='relu'))
    if problem == 'regression':
        model.add(Dense(1, activation='linear'))
        model.compile(loss='mse', optimizer='adam')
    else:
        model.add(Dense(1, activation='sigmoid'))
        model.compile(loss='binary_crossentropy', optimizer='adam')
    return model
```

```
for sym in data.columns[:]:
    df = modif_data[sym]
    model = create_model() #Model instantiation
    model.fit(df[cols], df[sym], epochs=25, verbose=False)#Model fitting
    pred = model.predict(df[cols])#Prediction step
    acc = accuracy_score(np.sign(df[sym]), np.sign(pred))#Accuracy calculation
```

```
DNN | AAPL.O      | acc=0.6248
DNN | MSFT.O      | acc=0.6200
DNN | INTC.O      | acc=0.6288
DNN | AMZN.O      | acc=0.6112
DNN | GS.N        | acc=0.6308
DNN | SPY         | acc=0.6065
DNN | .SPX        | acc=0.5562
DNN | .VIX        | acc=0.6093
DNN | EUR=        | acc=0.5682
DNN | XAU=        | acc=0.6033
DNN | GDX         | acc=0.6675
DNN | GLD         | acc=0.6069
```

Screen 8 : Accuracy of DNN

In this elementary sample, we can observe that the neural networks may outperform OLS regression fairly in-sample with predictions of the vector of direction of price movements for the following days of price movements. Nevertheless, we can question ourselves about the difference for the picture possibly changing while performing the test for the out-of-sample performance of the exactly two model types. At this point, the repetition of the analyses has to be made, but with the difference that the training or fitting step is enforced on

data's first 80% and the performance is tested at the same time on the remaining 20%. Also, the OLS regression has to be run before, on the first stage. Then, out-of-sample OLS regression illustrates quite alike the level of accuracy relative to the accuracy levels of 50% in-sample. For the $MLPRegressor$ model, its productivity is out-of-sample is poor if taking the comparison with the in-sample numbers and also with the outcomes of OLS regression. The identical is valid for the Sequential model from $Keras$. For the Sequential model from $Keras$ the out-of-sample numbers will illustrate as well the precision of values between a few percentage points near the 50% threshold (whether above or below this percentage).
Making the evident conclusions formulated on the approaches, used in the following section, the markets are presented in the very inefficient and weak pattern. Hence, the analysis of the historical return patterns, which are positioned on the OLS regression or established on the neural networks is weak, since this analysis is insufficient in order to determine statistical inefficiencies.

6.3 Conclusion

The corner solutions are obtained via the traditional portfolio theory. Furthermore, they can accept only 20-30 investments. There exists the tendency for traditional technology, which can be interpreted as to fir already selected financial products in computed previously asset allocation buckets, which mostly recognized as "two-stage" optimization.The function of the replacement technology consists of providing the ability to diversify the output results via the specification of the customized parameters for every individual investor. Concluding the above, IA+AI are not considered to be a myth anymore. The data access and the opportunity to use the infrastructure which is based on internet resources with speed and computational scale proceed with the massive volume of data from the known easy accessibility for a mobile phone or tablet devices performed massive changes for today's world. Nevertheless, nowadays the formats of the financial data are under massive fragmentation. The meaningful value for the investors is operated via the platform which has such properties, as an automatic collection, integration and to clean data from the relevant sources accessible on the internet. However, if considering it separately, this does not generally have the direct meaning of providing superior investment decisions. The process of creation of valuable IA+AI platforms has artistic pieces in it, as well as most of the issues related to finance. These platforms will prosperously transform the industry of investments over the future decades.

Bibliography

[1] Dr.Bernard Lee. *Investment Analytics in the dawn of Artificial Intelligence, 2019*

[2] https://www.eurekalert.org/pub_releases/2019-08/ws-ia082619.php

[3] https://strategiccfo.com/investment-analysis/

[4] https://www.investopedia.com/terms/a/artificial-intelligence-ai.asp

[5] https://en.wikipedia.org/wiki/Valueatrisk/media/File:VaRdiagram.JPG

[6] https://en.wikipedia.org/wiki/Poisson_distribution

[7] https://en.wikipedia.org/wiki/Empirical_distribution_function

[8] https://en.wikipedia.org/wiki/Cornish-Fisher_expansion

[9] https://www.investopedia.com/terms/s/sharperatio.asp

[10] https://www.investopedia.com/terms/b/benchmark.asp

[11] Yves Hilpisch *Artificial Intelligence in Finance. A Python Based Guide, 1st edition 2020, p.192-199, chapter 6 Market Prediction based on Returns Data*