HELLENIC
OPEN
UNIVERSITY

School of Social Sciences

Master in Business Administration(MBA)

<u>Dissertation</u>

"Portfolio Optimization and recent approaches using Artificial Intelligence and Machine Learning techniques"

Author's Name & Surname: Antonios Alexiou

Supervisor A': Papadaki Afroditi

Supervisor B': Bekiaris Michail

Athens, May 2019

ΣΧΟΛΗ ΚΟΙΝΩΝΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

Πρόγραμμα Μεταπτυχιακών Σπουδών στη Διοίκηση Επιχειρήσεων (MBA)

Διπλωματική Εργασία

“Βελτιστοποίηση χαρτοφυλακίου και πρόσφατες προσεγγίσεις με τη χρήση Τεχνητής Νοημοσύνης και Μηχανικής Μάθησης”

Φοιτητής: Αντώνιος Αλεξίου

Επιβλέπουσα Α’: Παπαδάκη Αφροδίτη

Επιβλέπων Β’: Μπεκιάρης Μιχαήλ

Αθήνα, Μάιος 2019

Dedication

Ευχαριστώ θερμά την οικογένειά μου, ήτοι τη σύζυγο μου Εύα, αλλά και τον μικρό Γιάννη για την υποστήριξη και διευκόλυνση που μου παρείχαν καθόλη τη διάρκεια της μελέτης.

Abstract

Portfolio optimization constitutes a rather important sector for research. The issue under question is investment via allocation of stocks or other securities among all available assets with the target of return maximization. The investor always aims at a maximum rate of return re-assuring the lowest possible risk; or, at least, the risk that an investor can afford and might select regarding the investment. This defines, theoretically, the investment procedure and could be classified as a decision-making problem. During the last 70 years, scientists have made remarkable progress with H. Markowitz being the pioneer. His mean-variance model associates overall volatility of a diversified portfolio with the expected rate of return. After Markowitz' Nobel-winning proposal, quite a lot of alternatives were studied over the years. Possibilistic and statistical approaches have been imposed, to further assist on the establishment of portfolio optimization as a separate scientific field. Mathematics has always been the basis of any relative research. However, during the last decade, an immense progress in computing power was held and this posed the possibility that Artificial Intelligence and Machine Learning techniques can be applied in a tremendous scale for any data sets, with magnificent results. It is well-known that historical data of stock exchange markets are populating the web servers globally. Big Data is the fuel for machine learning techniques. In this thesis, we discuss portfolio optimization techniques that were applied so far, and their mathematical background. Also, artificial intelligence and machine learning is studied as a trending field in modern informatics science. The conjunction of classical models and artificial intelligence is presented with use of more than recent scientific examples. A classifier based on machine-learning is developed.

Keywords

artificial-intelligence, portfolio, stock, machine-learning.

Περίληψη

Το πρόβλημα βελτιστοποίησης του χαρτοφυλακίου επισφαλειών είναι ένας πολύ σημαντικός τομέας έρευνας. Το υπό εξέταση ζήτημα, είναι η επένδυση μέσω της διάθεσης μετοχών ή άλλων τίτλων, σε ένα υποσύνολο από διαθέσιμες επισφάλειες μεγιστοποιώντας την απόδοση του χαρτοφυλακίου. Ο επενδυτής στοχεύει πάντοτε σε ένα μέγιστο ποσοστό απόδοσης που εξασφαλίζει βέβαια τον μικρότερο δυνατό κίνδυνο ή τουλάχιστον, έναν κίνδυνο τον οποίο ο επενδυτής μπορεί να δεχτεί και να επιλέξει σχετικά με την επικείμενη επένδυση. Τούτο αποτελεί ακρογωνιαία λίθο για την διαδικασία επένδυσης και μπορεί να χαρακτηριστεί ως πρόβλημα λήψης αποφάσεων. Τα τελευταία 70 χρόνια, οι επιστήμονες έχουν σημειώσει αξιοσημείωτη πρόοδο με τον H. Markowitz να είναι ο πρωτοπόρος. Το μοντέλο της μέσης διακύμανσης συνδέει τη συνολική μεταβλητότητα ενός διαφοροποιημένου χαρτοφυλακίου, με το αναμενόμενο ποσοστό απόδοσης. Μετά την πρόταση του Markowitz που έλαβε βραβείο Νόμπελ μετέπειτα, μελετήθηκαν αρκετές εναλλακτικές λύσεις στο πέρασμα του χρόνου. Έχουν υπάρξει μαθηματικές και στατιστικές προσεγγίσεις για την περαιτέρω συμβολή στην καθιέρωση του ζητήματος της βελτιστοποίησης του χαρτοφυλακίου ως επιστημονικού πεδίου. Τα μαθηματικά αποτελούσαν πάντοτε τη βάση οποιασδήποτε σχετικής έρευνας. Ωστόσο, κατά την τελευταία δεκαετία, πραγματοποιήθηκε τεράστια πρόοδος στην λεγόμενη ισχύ των μηχανών ή αλλιώς στην υπολογιστική ισχύ, και αυτό έδωσε τη δυνατότητα στις τεχνικές Τεχνητής Νοημοσύνης και Μηχανικής Μάθησης να εφαρμοστούν σε μεγάλη κλίμακα για σύνολα δεδομένων, με εκπληκτικά αποτελέσματα. Είναι γνωστό ότι τα ιστορικά δεδομένα των χρηματιστηριακών αγορών συνωστίζονται σε διακομιστές παγκοσμίως. Τα δεδομένα είναι το καύσιμο για τις τεχνικές μάθησης μηχανών. Σε αυτή τη διατριβή, συζητούμε τις τεχνικές βελτιστοποίησης του χαρτοφυλακίου που εφαρμόστηκαν μέχρι τώρα και το μαθηματικό τους υπόβαθρο. Μελετώνται η τεχνητή νοημοσύνη και η μηχανική μάθηση ως ένα πεδίο έντονης τάσης στην σύγχρονη επιστήμη της πληροφορικής. Παρουσιάζεται ο συνδυασμός κλασικών μοντέλων και τεχνητής νοημοσύνης με τη χρήση πρόσφατων επιστημονικών παραδειγμάτων. Τέλος, παρουσιάζεται μια εφαρμογή κατηγοριοποιητή(classifier).

Λέξεις – Κλειδιά

Τεχνητή νοημοσύνη, μάθηση μηχανής, χαρτοφυλάκιο, μετοχή.

Table of Contents

List of Figures

List of Tables

List of Abbreviations & Acronyms

AI	Artificial intelligence
CAPM	Capital Asset Pricing Model
CML	Capital Market Line
CPU	Central Processing Unit
DDPG	Deep Deterministic Policy Gradient
DQN	Deep Q-Network
DDM	Dividend Discount Model
GPU	Graphics Processing Unit
IIE	Identical Independent Evaluator
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
KNN	K-Nearest Neighbours
ML	Machine Learning
MPT	Modern Portfolio Theory
OPRA	Optimal Portfolio of Risky Assets
PG	Policy Gradient
PPO	Proximal policy optimization
RL	Reinforcement Learning
SML	Security Market Line
SNR	Signal to Noise Ratio
SPNE	Subgame Perfect Nash Equilibrium
SVM	Support Vector Machine
TRPO	Trust Region Policy Optimization

1. Introduction

In 1952 H. Markowitz proposed an evolutional theory regarding securities’ portfolio optimization. The stock exchange markets were astonished with this article published in the Journal of Finance. Until the middle of the 20^{th} century, investors had been analyzing historical tendencies for specific securities before proceeding to any move. They had been investing based on rumors or inside information which, sometimes, were not reflecting the reality. The problem of portfolio optimization has been evolving the last 50 years. Many scientists in the economics’ and mathematics’ sectors have proposed different models, from Markowitz and the Modern Portfolio Theory, Sharpe’s Capital Asset pricing model, to recent approaches of fuzzy and possibilistic models. During the last years, computing power has increased tremendously. So, Artificial Intelligence (AI) and Machine Learning (ML) are trending methods for manipulating data or finding patterns in massive data. Inspired by the Big Data sets of historical financial data, many scientists are trying to use the power of Machine Learning and Artificial Intelligence to deal with portfolio optimization in global stock exchange markets.

In this dissertation we present a literature review concerning some common portfolio optimization techniques that have been used. The purpose is to present the literature deviating from a pure mathematical spirit and attempt to make the content affordable by people that are not very familiar with mathematic equations and theorems. Our intention is to bring this complicated knowledge regarding techniques for portfolio theory and machine learning on an entry level, as this trend is popular in many scientific fields and might help people that are interested, get some basic information. Owning this knowledge it would be easier for them to study more complicated scientific content such as papers, journals and books, discussing related issues. What is more, we intend to show that Machine Learning will play an important role as a regulator for next generation portfolio optimization techniques.

Recent studies promote Machine Learning as a strong candidate for use in portfolio optimization field. Having thoroughly studied the “pre-machine-learning” portfolio optimization methods, we make a deep dive in Artificial Intelligence and neural networks’ structure. Yet, we present two cases of machine learning for portfolio optimization. Then, we develop a classifier of asset characteristics with use of SVM ML algorithm.

We faced some impediments during this study as the level of mathematics applicable to machine learning is rather than high involving many sub-fields; from probability to decision algorithms, information and complexity theory, high level algebra and calculus. And always the fear of not being able to show and represent the knowledge acquired during the study of many books and journals. To propose this dissertation for review, many changes have been added and hours of thinking whether this would really seem interesting to any audience. In our epoch, machine-learning is handled by engineers as a hidden infrastructure with many ready-to-use features. So, even if the audience does not understand many of the content, there is at least math-free information to study and get most out of machine-learning. Specifically, the outline of this dissertation is as follows:

In chapter 2 we present the portfolio overview and the investment management procedure as well as an overview of the investment management process.

In chapter 3 we discuss modern portfolio theory by H. Markowitz and some alternatives introduced later such as portfolio performance measures.

In chapter 4 Artificial Intelligence and its evolution is presented, Machine Learning overview, algorithms used and some examples of application of AI in several business sectors.

In Chapter 5, we present Support Vector Machines (SVM) model for machine learning and a related case.

In Chapter 6, Deep Learning models are studied, and we show another quite premature case that is very interesting as it was reviewed on November 2018.

In chapter 7 we provide an overview of text and sentiment analysis for stock prediction.

In chapter 8 we present a simple SVM ML classifier application.

In chapter 9 we focus on discussing the dissertation outcomes and possible future work.

2. Stock Portfolio management overview

2.1 The investor

The majority of people throughout their life, will find it interesting, at least once, to invest in a stock exchange market, even if this is for fun only. They might be characterized as opportunistic investors. They will soon forget their "new habit" and will stop spending money in stock exchange markets. On the other hand, there are investors that pursue gains a major period of their life or career, spending heavy amounts of bills on portfolios. The outcome will not always be the best, but it is based on earning money from stocks' upshifts. It happens that an investor chases the gain despite of being short on liquidity and another investor might spend less money following a conservative policy. So, depending on the stocks' balance, an investor will be borrowing, lending or saving money in order to achieve a long-term portfolio ownership (Cordes et al, 2008). For example, in circumstances when current gains exceed consumption needs, people tend to save money. Of course, the savings could either be deposited into a bank account for future use or be re-invested for more profit. The former is a way to keep savings without risk and only lose the actual risk-free, as always a positive risk-free rate is provided. Otherwise, as stated, the investor re-invests the gains in order to further increase and create gains over gains. So, an investor that can afford the risk of re-investing the gains gathered from previous investments, usually seeks new investment opportunities and a chance of liquidity enlargement. It makes sense for an investor to find the best way, or the most reasonable selection of stocks that does not endanger the investment. Of course, there is no absolute answer to this reasonable situation. An investor might, or might not seek help selecting the best portfolio options. There are plenty of approaches on both the tactics and the selection. An investor is not only an individual. Governments, Organizations, Corporations are investing heavily.

2.2 The investment

Investors should always have in mind the time-value of money; the so-called interest rate. People that invest in stocks should expect gains superior to the current main interest rate which is typical, or else, they should be tending to invest money in bank accounts or Treasury bonds that pay a fixed interest rate. Investing is de facto, a race of gaining more than the interest rates.

Investment is the spending of liquidity on securities that will compensate the investor in the future for many reasons, such as, the period of time the investor holds the money in the investment, a rate of return higher than the inflation rate and the risk that the investment is exposed to. From the above, we might need to focus on the risks that an investor takes when investing his money. This is of critical importance and formulates a vast amount of theoretical and mathematical approaches.

2.3 The portfolio

A portfolio is consisting of a structured combination of financial securities. The term financial securities has a broad meaning and gathers every financial asset that is available for buying or selling. So, an investment portfolio refers to assets of investment that are considered as a single unit. For example, an asset in a portfolio could be the stocks of a specific company. Someone might think that the mixture of a major number of stock holdings, forms a portfolio, and that is very true. It is indeed a portfolio of stocks; a random stock portfolio at least. The investor, however, should select an ameliorated portfolio consisting of a number and a mixture of stocks in such a way, that this portfolio will have a strong probability to compensate more in the future than any other portfolio. In order to manage and obtain the best portfolio, every investor must share his financial and investment profile as a basis of preferences. Primarily, the wealth position of the investor is to be defined, so that an investment policy can be formed. Another constraint is the level of risk that the investor is willing to get exposed to, and the preferred level of return. Then, the formulation of a portfolio is all about minimizing the risk with techniques, that will be presented in the following chapters, while maximizing the profit of the portfolio. The objective of the portfolio formulation is to deal with risk and diversification of stocks as well as with the attitude of the investor. This is dealed with the portfolio management process (Cordes et al, 2008).

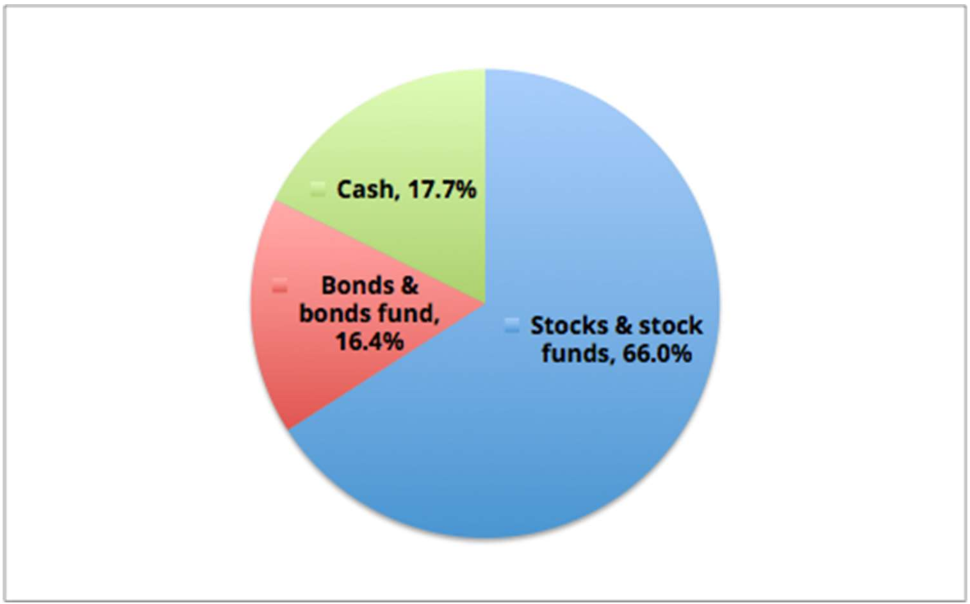


Figure 2.1 An example of a securities' portfolio

(Source: https://www.investx.com/SiteArea/Learn/RetailInvestorPortfolio)

2.4 The portfolio management

Investors must constantly be alerted regarding their portfolios. Thus, a solution to the continuous shifts in the stock's environment might be the management of the portfolio. For big portfolios this process is held by dedicated organizations that control the instrumentation of their customers-investors' portfolios.

The first action of portfolio management is, as aforementioned, the selection of stocks that form the portfolio and also combine the personal or organizational desires of the investor. The major aspects of portfolio management are:

- Planning the portfolio
- Select and construct the portfolio
- Review and evaluate the results

The goal is to maintain a balance according to the objectives of the investor, risk and profitability. Typical tactics of investments when managing portfolios are buying when market is low and sell when market is high. In this process, SWOT (strengths weaknesses opportunities and threats) analysis at the right moment is of great importance. The portfolio manager must always seek low risk and high returns of the portfolio that he controls on an everyday routine. In the literature, portfolio management is described as an art as well as a science of making the right decisions for the perfect mix on time (Cordes et al, 2008). The last sentence describes the volatility of the portfolio mix which is eligible to changes every day by the portfolio manager. Huge organizations with expertise, struggle to attract investors-customers. They put all the best computing tools and the adequate number of financial markets' specialists to achieve the goals for their customers.

Figure 2.2 Portfolio management process

(Source: https://www.liberatedstocktrader.com/stock-market-portfolio-management/)

So, portfolio management is a whole process emphasizing on all of the activities linked with securities or assets of an investor. It is dynamically analyzed and after analysis actions are proposed. The portfolio management-as-a-service helps the investor achieve his goals via constraints, preferences of risk and return even if the investor has a minimal knowledge on investing. The portfolio management involves constant review, adjustments and evaluation, synchronized with the rapid market changes. When markets shift, the portfolio must be altered to meet the new environment situation. In unstable conditions this could be very fuzzy. The portfolio management-as-a-service plays a crucial role on deciding whether to invest on one or another set of portfolios. The right combination of stock assets that are open for investors, is selected with use of mathematical models such as the MPT(modern portfolio theory) and other derivatives of this classical theoretical basis by Markowitz. Software tools depicting current market situation, using state of the art technology, are massively developed and a flavor of them will be presented in the following chapters.

3. Evolution of mathematical models

Before proceeding to the presentation of some basic mathematical models one should understand the history of stock assets selection prior to Markowitz modern portfolio theory. This goes back almost a century.

3.1 Before Markowitz

A completely different approach was taking place on Stock Exchange Markets before MPT. The investor and investments manager did not have any model to, efficiently, select the blend of stocks. News were spreading less rapidly via newspapers and fixed telephony or telegraph. This becomes critical for timing. It is rather than obvious that the investor had more time to decide but lower newsfeeds velocity. The audience of exchange markets tended to think about hot stocks. The story was not about diversification of stocks yet the stocks that hold risks negatively co-related. The DDM (dividend discount model) played a major role on acquiring good stocks at the lowest possible price. This, so called, lowest price policy was an assumption according to historical data of a specific stock price. However, the next day this price might fall again, thus the previous day price was not the lowest possible and the assumption might have been proven wrong. Despite this, an investor could seek for talented portfolio managers that made great success by analyzing the financial status of companies and then make investment decisions. They had been chasing companies with a promising financial future, based on financial statements and analysis, but on low prices. However, they did never focus on risk diversification. Thus, the 20th century investment analyst would recommend a specific set of hot stocks of a pharmaceutical company that would produce gains. Another speculation for investing would be a specific field of the industry that was going up rather than down (Kenneth, 2011). Or from another sector for which non-formal or inside information, presented this as a hot investment despite the fact this was about to form a bubble.

3.2 The Modern Portfolio Theory

As of 1952, a breakthrough was held in the field of portfolio selection and optimization. Harry Markowitz introduced a prototype for the analysis of investment in portfolios. Posting a single article in the Journal of Finance, would drive him to the summit of 1999 Nobel Prize. In this article he presented a new way of developing portfolios by considering the Rate of return and the related risk as well as the co-relation between the different stocks. He proved that by selecting co-related assets, an investor could maximize the Rate of return while minimizing the risk. This model caused a radical change against empirical approaches like the portfolios of hot stocks. Markowitz introduced the diversification of stocks of a portfolio, in such a way, that this model keeps a nice balancing between loss and gain in extreme non-systematic conditions (Markowitz, 1952). At first the Markowitz theory takes one period evaluation. In the beginning of each period an analysis is performed which results to the investment in the optimal portfolio of all possible portfolios. The theory is based in indifference curves that show the risk versus expected return of a portfolio. Risk as standard deviation is shown in the horizontal axis while expected return is marked in the vertical axis. Below you can see an example of such an indifference curve. In this figure we can check the three indifference curves $I_1, I_2, I_3,$ designed according to the preferences of three investors. For every one of the investors all points lying on the indifference curve are acceptable. Investor 1 can select point C on his indifference curve with expected return r_c and risk σ_c. Here we can point out that point B is very risky for investor 2, establishing a high expected return v/s point A where $r_a < r_b$ and $\sigma_a < \sigma_b$.

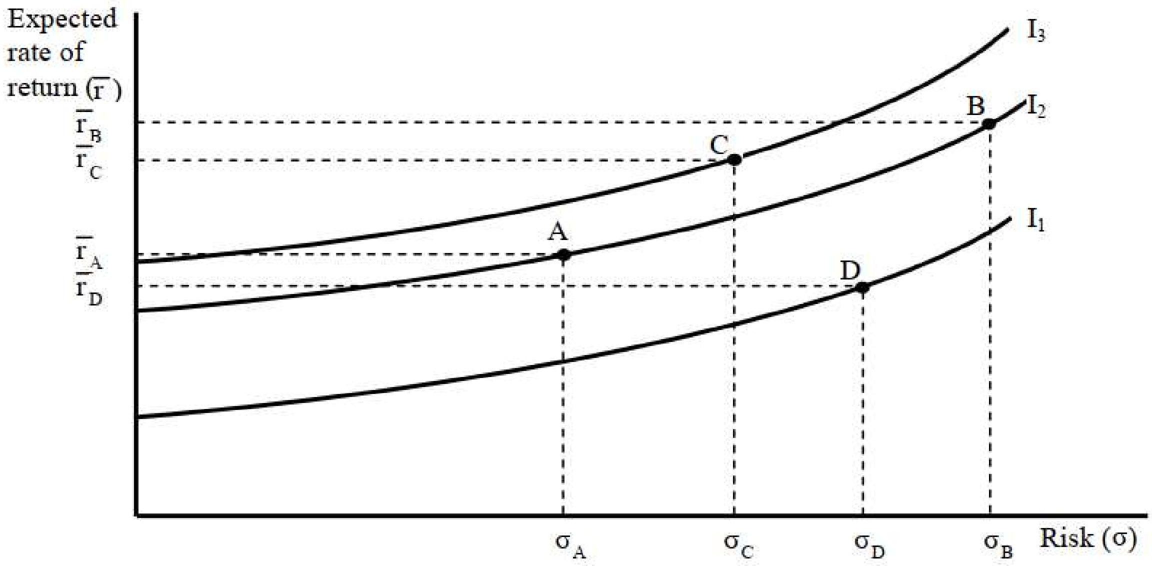


Figure 3.1 Indifference curve

(Source: Levišauskait, 2010)

All the above were shown by Markowitz with the following mathematical model. The return $R_i$of a portfolio at a specific time, let it be t, is defined as the total value T(t) of the portfolio at this specific timestamp, t, divided by the total value of the same portfolio at the timestamp t-1. $R(t) = [\frac{T(t)}{T(t-1)}]$, and multiplying by 100 we get the percentage of return from (t-1) till time t.

Markowitz based his theory on estimating the optimal portfolio by using the variance and the mean values of the rate of return R given that the typical investor 's intention is to anticipate high returns with the lowest possible risks.

Let us consider a portfolio with n different assets while the return of asset 0 is R_0, the return of asset 1 is R_1 and so on, asset n with return $R_n$or equally $R_i$for i=0 to i=n.

Given the same consideration we can use μ_i and ${\sigma_i}^2$as the mean and the variance respectively. Covariance of assets i and j with returns $R_i$and $R_j$is σ_{ij}. The weight with which each asset(i) is invested in the portfolio is x_i.

The total return of the portfolio is a sum of R_i so it is R:

$$\mu = E[R] = \sum_{i=1}^{n} \mu_i x_i$$

$$\sigma^2 = Var(R) = \sum_{i=1}^{n} \sum_{j=1}^{n} (\sigma_{ij} * x_i * x_j)$$

$$\sum_{i=1}^{n} x_i = 1$$

$$x_i \geq 1, i = 1,2 \ldots\ldots n$$

Where:

μ: the mean expected Return of the portfolio.

σ^2: the variance of the returns R.

x_i: the weight invested in portfolio i

The third equation shows that all weights should be 1 or 100% invested.

The fourth equation shows that no sort selling is allowed, and all weights must be greater than zero.

It is obvious that for any different set of $x_i$we get different σ and μ for this portfolio sets. All the possible sets are so called attainable sets. The sets that combine minimum σ for a selected μ and a maximum μ for a selected σ, draw the well-known line of the efficient frontier. Of course, in order to minimize risk and maximize R, an investor should lie on the efficient frontier curve regarding the investment choices for his portfolio. Take a look at figure 3.2 where we can easily define the efficient frontier as a subset of the attainable sets which is an eclipse-like curve.

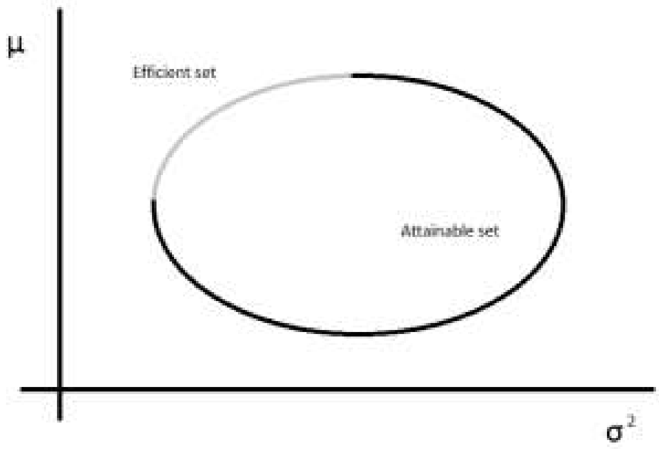


Figure 3.2 Attainable assets

(Source: Marling, 2012)

3.3 Portfolio optimization per Markowitz without risk-free asset

The optimization of a portfolio can be described as follows:

$$min(\sigma^2 - A\mu)\,,$$

$$\sum_{i=1}^{n} x_i \;= 1$$

$$x_i \geq 1, i = 1,2\ldots\ldots n$$

Where A shows how risk-averse is the investor. For example, A=0 represents a very conservative investor wanting the lowest possible risk while $A \to \infty$ shows an investor that wants a maximum R independently of the risk. In the following figure we can examine some possible attainable sets with 3 assets x_1, x_2, x_3 and their characteristics.

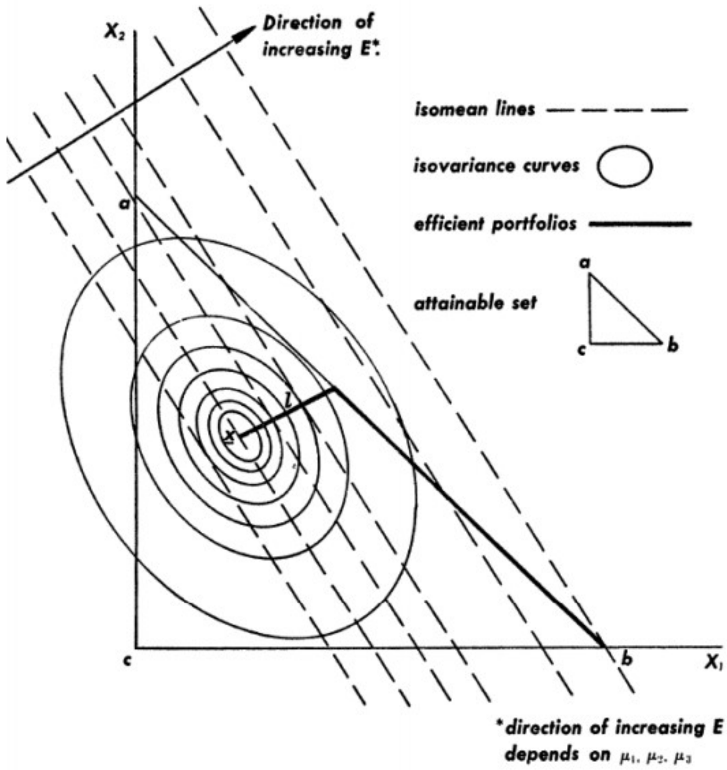


Figure 3.3 Representation of 3 attainable sets

(Source: Marling, 2012)

3.4 Portfolio optimization per Markowitz with risk-free asset

In the special case where there exists a risk-free rate, that shows an asset that can be handled without any risk, the problem is modified as follows:

Weight x is the investments on risky assets while the (1-x) proportion is the investment weight to the risk-free asset.

$$\mu = E[R] = (1-x)r + x\mu_\rho = r + x(\mu_\rho - r),$$

$$\sigma^2 = V(R) = x^2 {\sigma_p}^2$$

From the above:

$$\mu = E[R] = r + \frac{(\mu_\rho - r)}{\sigma_p}\sigma$$

The last equation forms a straight line that intersects the y-axis at (0, r) and has a slope coefficient of $\frac{(\mu_\rho - r)}{\sigma_p}$.

Thinking the same as without a risk-free asset, the selection that maximizes return and minimizes risk make this straight-line steep when σ_p is min and $(\mu_\rho - r)$ is max. Given these we get the Optimal Portfolio of Risky Assets points OPRA1 and OPRA2 as shown in figure 3.4 for different r_1, r_2. This OPRA is the solution to the maximization problem as follows:

$$\sum_{i=1}^{n} \mu_i x_i \max \frac{(\mu_\rho - r)}{\sigma_p} = E(R) = \sum_{i=1}^{n} \mu_i x_i$$

given the following:

$$\sigma^2 = Var(R) = \sum_{i=1}^{n} \sum_{j=1}^{n} (\sigma_{ij} {}_* x_i * x_j)$$

$$\sum_{i=1}^{n} x_i = 1$$

The solution to the above issue results to the well-known Capital Market Line (CML) as show in figure 3.4 which is the line that passes from point $(0, r_i)$ and is tangent to the efficient frontier at [OPRA](i).

Capital Market Line is the line on which, the investor can decide to use any point to invest with a given risk and return by borrowing or lending money, from risk free and investing in OPRA point for the rest percentage of the portfolio respectively.

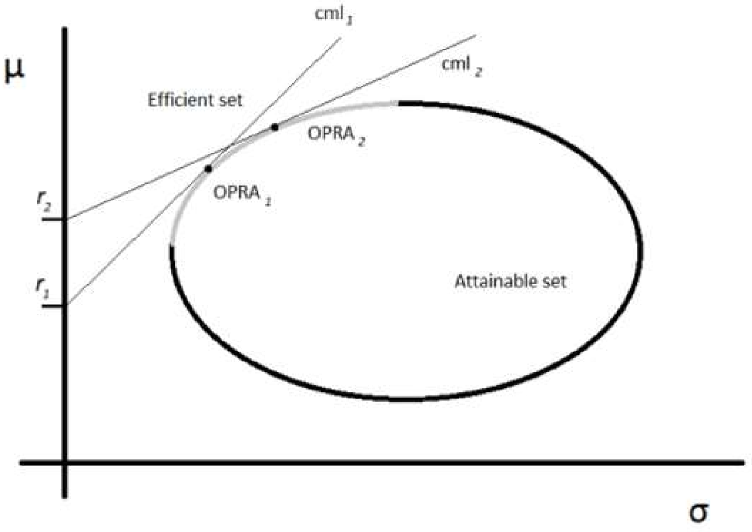


Figure 3.4 Two examples of CML and the OPRA for a specific r.

(Source: Levišauskait, 2010)

The Markowitz portfolio model is the cornerstone of modern portfolio theory. Despite the fact of being a pioneering mechanism for portfolio selection it is not used a lot. This is because it holds massive data with a complexity of $O(2n) + O(n^2)$ hence a quadratic complexity problem. Some less complex adaptations have their origins on Markowitz model and are presented in the following sub sections.

3.5 THE SINGLE-INDEX MODEL

The, so-called Single Index Model was invented by William Sharpe and helps establishing a lower number of parameters against the original Markowitz approach. Thus, it eliminates the calculations needed for estimating the Efficient frontier. The main characteristic of single index model is that it takes as a constraint the fact that returns of each security are correlated as a result of the economic landscape and not their origins, i.e. their business sector. So, this model enhances the idea of a similar response and upshift or downshift following any changes in stock exchange markets. In this way the return of a specific security (stock) is related with the return of a market index related to systemic changes.

In order to get a better idea, at this point, we should focus on understanding systematic risk versus unsystematic risk. Systematic risk is the risk that leads to the same results for all securities or stocks. It cannot be diversified because all stocks are subject to this risk. Unsystematic risk is the residual risk that can be diversified as securities' number in the portfolio, advances. In a generic approach, the risk σ of a security is the sum of the systematic and unsystematic risk. Consequently, Markowitz theory on portfolio optimization eliminates only the unsystematic risk. Regarding systematic risk diversification is not really possible.

The single index model is as follows:

$R_i = \alpha_i + \beta_i \, * \, Rm + e_i$ where:

i: Security number 0 to i

R_i: Rate of return of the ith security

a_i: a variable added to the overall security return

β_i: the coefficient of sensitivity of the security return v/s a change to Rm

Rm: the rate of return of the market index

e_i: a random error in addition which benchmarks the difference between Expected and real return.

As per statistics analysis, the single index model is an estimation line based on Linear Regression with R_i on y-axis and Rm on the x-axis. This relation is shown in the figure below:

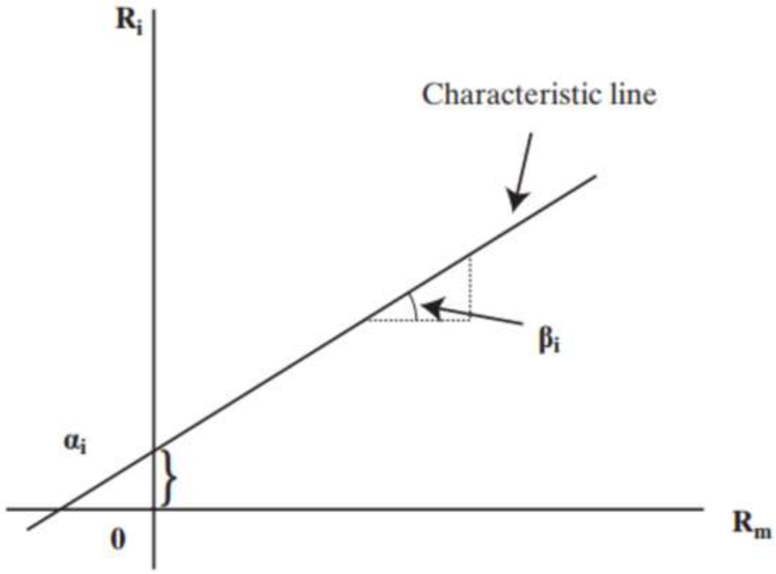


Figure 3.5 Characteristic line example

(Source: Vassiliou, 2005)

The characteristic line depicts the relation of a change on the market return and the related security return change or the market index change. The λ slope of this line is the β-co-efficient which leads to the measurement of the degree of systematical change of the return of the presented security as a response to the market returns changes. Thus, betta is called the coefficient of systematic risk as a response to general market conditions and we cannot diversify this risk. From the latter we can extract the information that betta coefficient for the market portfolio is 1.

On the other hand, α is defining the point (0, α), the intersection of the characteristic line with y-axis and represents the non-systematic risk for the security i.

At this point we should stress that alpha and betta co-efficients in the single index model are a result of the least square method applied:

$$\beta_i = \frac{\sigma_\mu}{{\sigma_\mu}^2}$$

$$\alpha_i = E(R_i) - \beta_i E(Rm)$$

And the correlation coefficient of the security and the index is:

$$\rho_{\ im} = \frac{\sigma_{im}}{\sigma_i \sigma_m} \Rightarrow \rho_{\ im} = \frac{\beta_i {\sigma_m}^2}{\sigma_i \sigma_m} \Rightarrow \rho_{\ im} = \frac{\beta_i \sigma_m}{\sigma_i}$$

We can show that markets betta is 1:

$$\beta_{\ m} = \frac{\sigma_{mm}}{\sigma_m \sigma_m} = 1$$

Now let us summarize on the use of betta which is of great importance theoretically. A stock with betta more than 1 is a very volatile selection because a small change in the market return will cause a much greater change in the security return. Similarly, when betta is lower than one, investing on this security is conservative, as the security moves slower against market changes.

3.6 Single Index model and applications

The Single index model helps with the simplification of the estimation on the Markowitz model. The estimation of a stock return, standard deviation and covariance is much easier with the linear index model for each security. This way, the efficient frontier of portfolios is generated more efficiently. So, the following output of the single index model could be tapped to the Markowitz model:

$$\alpha_i = E(R_i) - \beta_i E(Rm)$$

$${\sigma_i}^2 = {\beta_i}^2 {\sigma_m}^2 {\sigma_{ei}}^2$$

$$\sigma_{ij} = \beta_i \beta_j {\sigma_m}^2$$

Moreover, we can apply the single index model to find and analyze a portfolio return and risk as follows:

$E(R_p) = \alpha_p + \beta_p\ E(Rm)$, where:

$$a_p = \sum_{i=1}^{n} w_i \, a_i$$

$$\beta_p = \sum_{i=1}^{n} w_i \, \beta_i$$

$a_\iota \, \beta_i \; w_i$: the alphas bettas and weights of security i thus

a_p and β_p are the weighted average of all securities in the portfolio.

The risk of the portfolio is :

$$\sigma_p{}^2 = \beta_p{}^2 \sigma_m{}^2 + \sum_{i=1}^{n} w_i^2 \sigma_{ei}{}^2$$

3.7 Actual Market portfolio and separation theorem

The market portfolio is denoted by the point of tangency of the capital market line and the efficient frontier curve. All other portfolios along the efficient frontier curve are less efficient. Therefore, an investor will invest to the Risk-free asset and the market portfolio. So, the preferred investment of each investor is a combination of choices between the risk-free asset and the market portfolio.

The market portfolio consists of all types of risky assets, bonds, T-bills, gold, currencies etc. It cannot be determined a priori, but it is obvious that it is a well-diversified portfolio imposing only systematic risk. To be more realistic, in terms of simplicity, the market portfolio might be defined as the S&P500 or the FTSE20 indices which denote a simplified version of the market portfolio.

James Tobin proposed the Separation Theorem in 1958 saying that the investor selects to invest on the market portfolio which is unique, and then as a second step, the investor selects how he will finance his investment depending on his risk preferences. It is obvious that the market portfolio formation does not depend on the investor choices and characteristics as an investor. An investor's optimal decision for investing is defined by his highest indifference curve and the tangency point to the capital market line. The higher the preferred σ of the investor the less risk averse the investor is and vice versa.

3.8 Security Market Line

The Capital Asset Pricing model (CAPM) was issued first by W. Sharpe and simplified Markowitz theory. As already mentioned, combined with the single index model, it eliminated the complexity of the related quadratic programming problem. Also, the respective elimination of unsystematic risk which can be seen in the figure below has been well presented. However, systematic risk is never eliminated with these models.

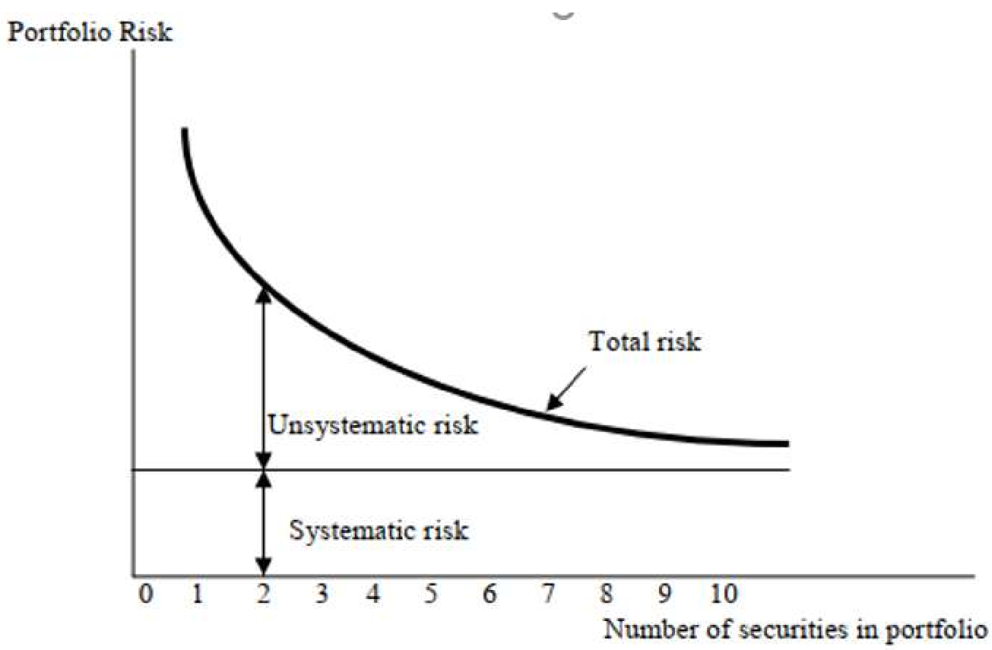


Figure 3.6 Risk versus diversification

(Source: Vassiliou, 2005)

The essence of the CAPM: the more systematic risk the investor carry, the greater is his/her expected return. Based on the CAPM we get a prediction of the expected return based on the return on the market and the systematic risk as follows:

$$E(rj) = Rf + (\beta_j) * (E(r_M) - Rf)$$

$E(rj)$ is the expected return on j stock

Rf is the risk − free rate

β_j betta coefficent of j stock

$E(r_M)$expectedmarket return

CAPM improves the definition of optimal portfolios and the risk analysis but it cannot deal with systematic risk. Companies worldwide, holding huge stock assets, can influence the stock exchange landscape by selling or buying assets thus provoking systematic risk. Somehow CAPM is helpful, as a theoretical and practical approach, rather than the Markowitz model.

As we can see below, the CAPM formula might)be presented as a line having a y-axis intercept of Rf and slope of $\beta * (E(r_M) - Rf)$ in the next figure. This equation between the expected return and betta is called the Security Market Line or SML. Each stock is described by its SML and characteristic line as mentioned before.

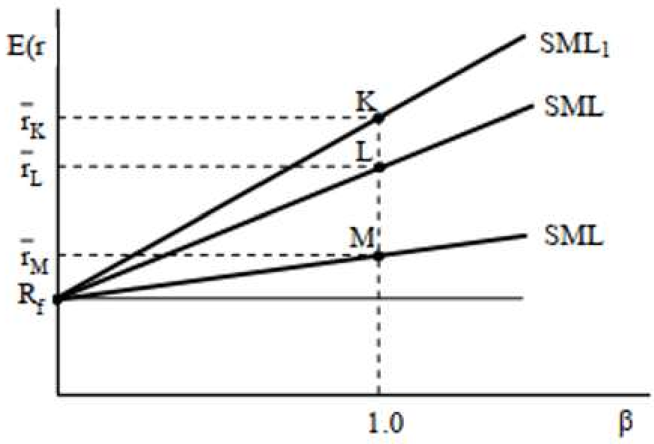


Figure 3.7 Security Market Line

(Source: Levišauskait, 2010)

3.9 Portfolio performance Measures

Portfolio performance is all about performing periodic portfolio performance statistics regarding the return earned, and the risk issued. The market value of a specific portfolio for a timestamp or beginning of a period (Vb) is determined by the sum of the assets' market value at this timestamp. The market value of a specific portfolio after a specific period (Ve) (i.e. 1 year) is the sum of the assets' market value at the end of each period.

So, the portfolio return is:

$$r_p = \frac{(Ve - Vb)}{Vb}$$

As aforementioned, the betta shows us the risk of a portfolio. But there are certain measures for adjusted performance with regards to risk.

Sharpe's ratio measure represents the excess of a return versus risk free rate, by the standard deviation σ.

$$\text{Sharpe's ratio} = \frac{r_p - r_f}{\sigma}$$

r_p *is the actual return*

r_f is the risk-free rate

Treynor's ratio measure represents the excess of a return versus risk free rate, by unit of systematic risk, measured by Beta:

$$\text{Treynor's ratio} = \frac{r_p - r_f}{\beta_p}$$

r_p *is the actual return*

r_f is the risk-free rate

β_p systematic risk

Jensen 's Alpha gives the real return over the required return excess and the difference of actual risk premium compared with risk premium. Jensen's alpha is (using the CAPM method):

$$\text{Jensen's Alpha} = [r_p - r_f] - \beta_p[r_m - r_f]$$

$$r_p \ is\ the\ actual\ return$$

r_f is the risk-free rate

β_p systematic risk

$$r_m \ is\ the\ markets'average\ \ return$$

Of course, we should mention that if a portfolio is fully diversified all these measures of Sharpe Treynor and Jensen will conclude on the same results so there is no difference in portfolios ranking methods. The reason behind this, is that the variance and thus the risk is only the systematic risk. It is known that Jensen's and Treynor measure is diversification tolerant related to the Sharpe measure. The latter measure applies total risk, so systematic and unsystematic risk is included making Sharpe a more conservative measure related to the other two.

3.10 Other mathematical approaches used

The first mathematical model as discussed, was applied for stock optimization by Markowitz. In order to decrease the complexity of this model several scientists proposed models with linear complexity such as the Mean absolute Deviation model. This is very relevant to the original Markowitz model for multivariate distribution of the returns.

Another proposal was made by Young and it is about maximizing the minimum return or minimizing the maximum loss, named minimax and it seems a better approach relative to the mean variance model.

3.10.1 Mean-Semi variance Model

The original thoughts of Markowitz apart from causing a complexity in calculations, have some downsides. As squared, the variance puts limits in terms of extreme values. Extreme values of deviation cause strong effects regarding this model. For example, higher gains than mean, strongly affect the process. So, when the distribution of return is not Gaussian, variance will not work for skewed distributions as well as other models. This model of mean semi variance that take only the negative deviation from an investors point of view, as input, was also proposed by Markowitz. Its main flexible point is that variance beyond the risk threshold is not included.

$$semiVar = \sum_{r_t < Averag}^{n} (Average - r_t)^2$$

$Average\ is$ the observed average

n is the total number observations below mean

r_t the mean or target value of the dataset

3.10.2 Mean-Absolute Deviation Model

Konno and Yamazaki proposed another approach to make improvements opposed to mean semi-variance model which is also computational costly. The new idea introduced is to use the absolute values of deviation to measure risk. This approach can treat Markowitz theory implications and cope well with the advantages of Markowitz theory. At last this proposal uses L1-risk functions for risk and L2-risk function for variance. L1-risk function typically induces a linear notion in this mathematical problem which lowers the complexity. The model can be applied to huge number of securities without impediments on computational level. It can also be applied to tailed distributions of rate of return with success. The mathematical representation is the following:

$$m(x_{1,}x_2,\dots,x_n) = E\left[\left|\sum R_i x_i \; - \; E\left[\sum_{i=1}^{n} \; R_i x_i\right]\right|\right]$$

The absolute deviation of a random variable is the expected absolute of the subtraction between the sum of all random variables and its corresponding sum of means. The risk is $m(x_{1,}x_2,\dots,x_n)$ and is greater than zero.

3.10.3 Mean-Semi Absolute Deviation Model

After Konno & Yamazaki, Sprenza issued the so called semi absolute deviation model to measure risk. He proved that the calculation of semi absolute deviation instead of absolute deviation creates an advantage if the return has a Gaussian distribution. This way by thoroughly selecting the constraints makes the solution to this problem half simple in terms of computational complexity versus the absolute deviation model. The mathematical representation is as follows:

$$W = \frac{1}{2} * \left[\left|\sum r_{it} - r_i \;\right| + \; \left[\sum_{i=1}^{n} \; (r_i - r_{it}) x_i\right]\right].$$

4. Artificial Intelligence Overview

More than 50 years ago, a brilliant young man, Alan Turing was skeptical about mathematical, cryptographic and other types of issues that could influence the globe; which was the outcome of course. Apart from these sectors, he did write some thoughts regarding the possibility of machines to think and learn in the future (Turing, 1950).

The well-known Turing test was presented in MIND magazine Vol LIX No.236 10/1950. The journal entry titled as the "Imitation game" was carrying a pioneering thesis. Can machines think? However, who could define the meaning of "think" and "machine". He rather analyzed that this is not a poll Gallop-type, re-defining the question with a problem-game. Imitation game is played with 3 members a man(A) a woman(B) and the interrogator (C). The purpose of the game is that the interrogator understands who is the man and who is the woman being on a separate place away from them. (A) must avoid presenting his gender trying to deviate from interrogator's questions. (B) must help the interrogator with true answers. Alan Turing proposes other forms of communication other than voice for obvious reasons. The gist out of this, is that suppose (A) is a machine and what will happen to the final assumption of (C) for the genders of (A) and (B). Will the interrogator choose correct? This is per Turing, a similar question to the one asking if machines could think. After this nice point in his article he briefly described the Turing Machine, the cornerstone of computing and information theory.

4.1 Artificial Intelligence today

Artificial intelligence is the answer to the imitation game. (A) convinces (C) that (B) is a man. Game over. IBM®'s Deep Blue computer wins Casparov in a single game of chess. That is, machines are thinking, robots can talk and make decisions like humans.

Artificial intelligence is based on a tremendous development of computing systems that can perform tasks which otherwise would be carried out by a human. It could also be described as the field of computer science that can adapt to environment input changes and produce intelligent outputs such that it seems to own a human-like interaction,

Of course, there are various examples and fields that Artificial intelligence is applied today.

- Robots
- Bots (in the web)
- Visual recognition
- Speech recognition
- Decision making
- Self-driving cars
- Navigation systems
- Computer Games
- Search machines

However, all these examples, theoretically, should be characterized as weak Artificial intelligent machines because Strong AI machine does not yet exist.

In the history of humanity, lots of obstacles have been overcome with use of human intelligence. From the first prototype of an automobile to the intelligent auto-driven vehicle, many leaps have been demolished by our intelligence. Artificial intelligence might certainly become a new leap cruncher for current or future achievements. It acts as an expansion to human intelligence. Overall, humanity could take advantage of AI and go beyond the limits.

The change that can be applied with use of AI to activities of individuals or organizations is tremendous. The solution in problems, existing or new ones, will be dramatically faster. New modes of production and robotics will arise as well as new models for the financial sector. The science is undergoing a rigorous change with the aid of AI. An individual, a manager or a leader should stay tuned to this phenomenon for his own wellness.

4.2 The graphic card evolution

The last 3 decades of the 19th century CPUs had been gaining position in the market. From the first transistor by Bell Labs® in 1947, Intel®'s first microprocessor in 1971, Sun®'s SPARC processor, Intel®'s Pentium in 1995, to Intel® core i9 in 2018, many leaps were demolished; even the atom size leap of the quantum computer. In parallel, there has been a great effort from 1970 and onwards regarding the Graphics Processor Unit or GPU that would be a fast electronic circuit for graphical representation of games in personal computing.

However, GPUs were designed in such a way that the mathematical problems for which they are programmed to solve, remind the mathematical models used in AI (Hurwitz, 2018). Thus, the last 50-60 years the modern GPUs redefined AI breakthrough. Nowadays, such GPU processors are linked in a virtual environment so that they could produce a parallel massive machine power to handle delicate AI operations.

4.3 AI outcomes in business

AI makes our lives easier day by day with use of Deep Neural Networks which are neural networks designed to learn with the help of AI. So, one could program such a network to distinguish a voice of a man from the voice of a woman, a cat from a dog and so on.

Neural Networking is not a new science, but its use was restricted by lower CPU and GPU power till recently. Now it is possible to construct very complex neural networks to surpass this leap. Today's neural networks consist of billions of electronic synapses(connections) which were not a feasible case twenty years before. Complicated algorithms designed for AI, tapped to big data as its fuel. Vast dreams have become true with this combination.

AI is aiding billions of people every day via well-known applications. AI with big data promotes the democracy among industries. Small businesses can fight for their position in the market having cheap and easy access to AI the same way compared to giants like Microsoft® Google® or Facebook®. Many startups embed AI in their products. So, for anyone interested, free software libraries and tools are there, the data required are also available. The experience with the use of AI is unimaginable. AI allows a business or an individual to make things easier faster and efficiently.

4.4 From AI to Machine Learning

In the previous sub-chapter, a reference to Deep Learning was made. This model of Deep learning enables AI to take as input huge amounts of data and learn from the data. The outcome gets the more precise, the more the data volumes. A computer can identify a dog from a cat (visually) using a simple machine-learning scheme. The core idea behind, is that the more dogs it recognizes, the more the efficiency it gains and it minimizes the error factor of an erroneous identification.

Machine Learning or ML is a subset of AI science that designs a computing system to learn with programming and data. On the other hand, one could think that ML is a simple process. Well it is not that simple, and it goes the same as AI is not simple and Neural Networks are not simple. The design of ML is coupled with Big Data.

Big Data is any source of data with the following characteristics:

- Volumes in terms of millions GB.
- High velocity of data networking.
- Data sources expanding constantly.
- Data sets contain true and not misleading data.

4.5 Big Data and Machine Learning

Describing the importance of Big Data does not mean that without Big Data, ML cannot run. Using ML techniques, requires a minimal set of correct data. With expansion of the set of data one can get more accurate results. Improvements in the internet field is of main interest for expanding ML techniques because of the possibility to rapidly transfer data.

Businesses or organizations during their history had a main concern. Data archived in the 20th century in thousands or millions of books. After 1970 some of the data were digital but should be safe and many backups were taken for business continuity purposes. Tapes at first, then Hard Disk drives became cheaper, so companies of all scales could afford saving redundant backups safely. Nowadays it is the cloud, the so-called Storage-as-a-Service (SaS) offered by trusted entities in low prices. Another service in the cloud is HW-as-a-Service or HAS which helps companies to use the virtual HW components on the cloud to perform ML related calculations to their private set of data.

The data in such clouding infrastructure remain intact. So many data for so many years seem to pose a problem for business-as-usual operations. It is not the same way for ML. With ML we can use these old businesses' data to improve company's operation performance and ameliorate its mode of operation, for example, by avoiding historical faults.

Another aspect of ML is the form of data. Apart from the validity of data sources which is subject to a business use of ML and its potential, we need to housekeep our data in ML predefined formats. The latter means that data need to be transposed to a generic format that is accepted by ML models. This is a huge issue concerning specific types of data.

For example, a small business wants to sell a product that embeds utilization of face recognition as a secondary safety mechanism (2-way authentication) or else the product is designed to lock as an anti-theft mechanism. This sounds easy, but any ML model designed in whatever programming language will not accept a jpeg picture as input. The picture snapshot of the owner must be converted to a vector or matrix in order to be handled by the ML core of the exact product.

Another concern for turning data to ML readable format is missing data. For example, in voice recognition how a company should manipulate gaps or Gaussian noise or other nonregular data. The step of data refining is of great importance with regards to the process of integrating ML in a business. Without refining the outcome will not be error-proof.

Data science as well as business analysts, constantly need to make predictions based on analysis of historical sources. So, analytics is thought to be a critical application of data for companies' best practice. Businesses need to find a way to minimize uncertainty during decision taking and construct predictive ML models that are synched up with the current situation of the company. With descriptive analytics, a company can define the situation of the business or a, year over year, analysis can be held. Predictive analytics assist a company to predict business changes based on historical data.

4.6 Statistics and data mining

The fields of Statistics, data mining and ML are tied together to form the overall result of AI/ML. Statistics used to analyze the data and transform them to ML readable data in order to understand the importance and acquire knowledge out of the data. ML as well applies statistical models to make the right prediction; reject or not reject the null hypothesis in a one-class prediction problem,

Data mining is also a field of statistics that aids the exploration of Big Data sets to find any possible pattern in the data (Wallenberg, 2011). Data mining algorithms are used in this scheme to find patterns and then share the output to ML. The aim of data mining is to make data sets more understandable. Data mining is not a prediction process. The latter consists the result of ML as a next step.

There are plenty of vendors offering software solutions for data-mining. The outcome of data mining software is to extract data from a huge dataset with constraints, and to tap it to ML techniques for further analysis. For example, a manager might be curious to know the demographics(age) profile of customers that are buying a certain product in order to have a prediction of an alternative product intended to target a possible age gap. For this reason, data-mining could use as constraint the customers that buy the product to predict what age gap, the sales manager, should fill and help him select the right product. Thus, a new opportunity in the market.

4.7 Types of machine Learning

ML techniques are applied to cover several problems. So, the main types of ML are described in this section as shown in figure 4.1.

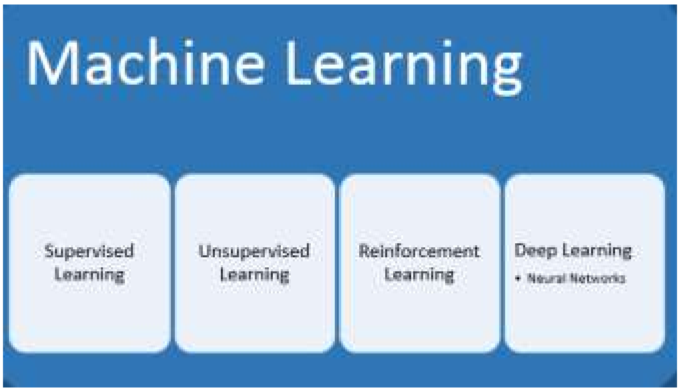


Figure 4.1 Machine Learning types

(Source: the essential handbook for leaders)

4.7.1 Supervised Learning

Supervised learning refers to the type of machine learning that starts with a given data set and the know-how of the classification of data. The data have very distinct labels. For example, as mentioned above, the machine learning procedure to find images of cats (or dogs) is a supervised learning. The dataset, in this case the images, have a certain meta-data description tag suggesting if the animal is a cat. Beginning from this, someone can create a machine learning application which will take as input this well-defined dataset and learn to distinguish cats from other animals. This can also be applied to more than one class of animals. A training dataset with millions of pictures with n animals will have n classes with a defined classification and a meta tag. This is a supervised learning model that can be trained and afterwards, respond to the input of a new picture containing an animal, and as a result it will recognize its species. Another nice example of supervised learning is fraud detection to a network. Many Intrusion Detection Systems (IDS)and Intrusion Prevention Systems (IPS) use historical data patterns to forecast an intrusion to the corporate network or a cyberattack. The model uses current packet flows to determine if this flow matches to any historical pattern of attack. This model is an overfitting model of supervised learning. Overfitting model is a model of supervised learning, in which, the process is defined to find only known patterns and not get trained from new patterns or unknown labels to images as stated above. If the model is trained for overfitting the forecasting results should be much greater.

4.7.2 Unsupervised Learning

Unsupervised learning is another approach to machine learning and refers to occasions where the data are of great volume and unlabeled. Good examples of this situation are Social Media that handle high volumes of data without any label, or which are not at all classified. Thus, a set of algorithms is required, that can iteratively classify and label the data. Unsupervised learning algorithms are designed to find patterns in unknown data. Given the fact that this kind of machine learning does not include a human interference it is named unsupervised learning. A good example of unsupervised learning, apart from social media pictures, is the well-known spam filter feature in our mailbox. Incoming mails contain too many parameters for a human to afford flagging and classifying legitimates from spam mailing. Instead of this, unsupervised machine learning algorithms named classifiers are applied to discover possible spam mails with an extremely accurate prediction which might be more than 95%. The classifier function segments the massive mailing data in to groups of features. Feature is a core term for machine learning. In the case of supervised learning with identification of a picture with a cat, the feature classification is one-dimensional thus 1 class, cat class or not-cat-class. Think about the classification of thousands of feature-classes regarding e-mail parameters and patterns or historical data. So, in unsupervised learning this classification process adds labels to the unlabeled data. From a developer perspective, classifiers are designed to cope with unknown data, unknown mails. An iteration procedure as mentioned, creates labels(features) and classes of data, so that it can conclude to a phase with immense data volumes but somehow labeled. From this point the process is supervised learning because the machine learning algorithm now knows the classes of data and can train itself. In healthcare, unsupervised learning is used to deal with non-labeled data and certain diseases. Historical medical files pass to a classifier that creates the classes of historical records regarding a known condition such as heart attack. In other words, it collects and classifies the patterns and the results that could indicate an alert for a cardiac arrest. After data are classified, for each new patient in question, the health care professional passes patient's medical records in the system. The system analyzes the data and finds patterns matching its training set and responds if the current person's health is in danger or not, for the specific disease.

4.7.3 Reinforcement Learning

Reinforcement learning is different from the other types of machine learning because the mechanism does not use a predefined data set or unclassified data. It is a live training function with use of trial and error techniques. An example of trial and error is robotics section. For example, a robot able to talk with adult men in a mall; and let us consider the primary goal of the robot to correctly distinguish men from general population. When the robot finds someone in the mall using robotic-vision, it can use speech and say, "Good morning Sir". The answer might be none which is a neutral answer and is not used to train the robot. If the answer is "I am a woman" or "I am a girl" or "I am a boy" this means that the robot failed to recognize the age and/or sex. So, its re-trains itself using the robotic-vision and defining the parameters of failure. At the end it will train itself to distinguish adult men in the population. Reinforcement learning is very popular in self-driving vehicles.

4.7.4 Deep Learning

Deep learning is a special way of machine learning that uses the cornerstone of artificial intelligence, the neural networks. It is used for massively non-structured data. So, it goes deeper than unsupervised learning. At the beginning, artificial intelligence used neural networks to reproduce a human being-like perception of certain circumstances. Complex neural networks are used to simulate the human brain and its functionality. In general, deep learning is used as a way to deal with abstract and poorly defined problems or even to identify and solve problems. A smart kid can easily identify the voice of his father or mother, but a complex neural system will take time to train itself for this simple operation. Deep learning is used in artificial vision applications. The deep learning hierarchy consists of 3 main layers that also apply in the other machine learning methods as shown in figure 4.2:

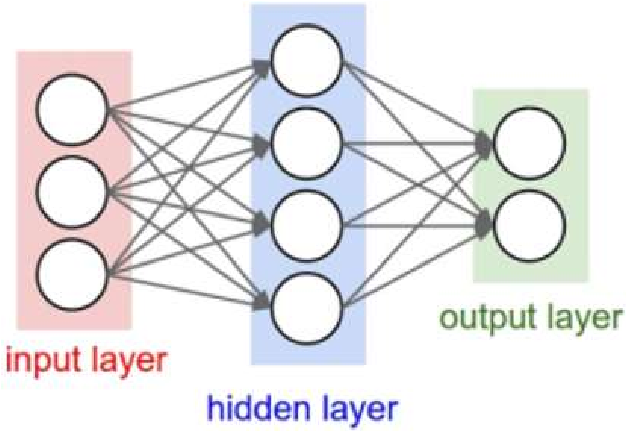


Figure 4.2 The architecture of a neural network

(source: www.pyimagesearch.com)

- Input layer, that analyzes the data sets and prepares them for the next layer
- Hidden Layer(s), that makes the fuzzy classifications and difficult algorithms
- Output layer that takes and represents the final decision

The difference from other learning types is, that in deep learning technique there are several hidden layers. In these layers deep learning consists of supervised and unsupervised technique layers. Deep learning might have impact on business as time pass. Very delicate voice recognition systems act as front-end customer support by phone, in the business sector. From identification of the caller, by collecting data to propose feasible solutions to a client's issue, up to sophisticated solutions.

4.8 Algorithms in ML

Inside machine learning we have algorithms. A simple algorithm is an instruction set. The role of algorithms is to determine how the computer will perform concerning a specific input so as to returning an output. A lot of programming languages exist like Java, Python, C, C++ and all of them support machine learning libraries. In machine learning the libraries are not programmed to give output to specific data input but the process itself creates the machine learning model. As data volumes increase, the machine learning algorithm will output more sophisticated results as well as a more sophisticated algorithm itself. Some types of algorithms exist and can be used in machine learning.

Bayesian algorithms (Naïve Bayes algorithm) are helpful when no massive training dataset is available and known external factors exist that can be used as constraints to the model. So, if the model is partially known, one should use Bayesian algorithms as an approach. For example, if the problem is diagnosis of diabetes based on medical records, apart from known patterns, external factors such as glucose levels results can be added to the model as constraints. This issue is a strong candidate for use of Bayesian algorithm.

Another important algorithmic technique for machine learning is Clustering. Datasets with similar characteristics can be separated from others and become a class or a cluster. From this, it is obvious that clustering is used in unsupervised learning where the machine learning technique, itself, defines the parameters and the classifier classifies the data.

Decision tree algorithms in machine learning are algorithms that formulate a tree structure to present possible outcomes of a problem. Each node – like in Nash SPNE- is assigned to a probability according to its likelihood to happen, as reported by the algorithm. Decision tree algorithms are often used when population segmentation is needed in business. Possible strategies are assigned to a probability or outcome. This machine learning technique helps the manager to take proper decisions.

Dimensionality reduction algorithms help scientists to remove unuseful data from the dataset. With the term unuseful data we also define the duplicate data or irrelevant data. For example, in massive Data Center environments, automated systems that monitor servers and environmental variables, are installed. The operator of the Data Center performs live monitoring and must be able to get an alert for any malfunction. So, Servers that report their health status as active might be ignored as unuseful data. If this info was to be included in this machine learning model it would consume more and more disk space. Thus, dimensionality reduction can reduce the number of data while searching patterns for upcoming trouble in the Data Center.

Linear regression algorithms are used for statistical analysis in current decision-making systems without the use of AI or machine learning. They were found to work well in parallel with machine learning aiding the analysis of relationships among different data sets or features. Along with other machine learning techniques they are strong candidates for predicting outcomes of a process in the future and in decision making. Of course, an analyst should assume that this algorithm takes in to account co-relation. So, in order to use this method in machine learning one must fully understand the data values and types.

Regularization is a class of algorithms that modify existing data sets to counteract the aforementioned overfitting of data sets. These algorithms are normally applicable to any mode of machine learning and aid with not accurate predictions caused by missing or overfitting data.

Rule based machine learning algorithms is a class of algorithms that use logical decisions to explain the labels of data sets. This set of algorithms should not cope with unsupervised mode of machine learning as it fits to supervised learning model. However, we should stress that even if a machine learning model is constructed with millions of rules/constraints, in practice, these rules might disintegrate one by one after applying the model with trial and error mode training or with feedback training.

4.9 Machine learning model

The generic model of machine learning consists of an infinite cycle. An algorithm should be selected based on the problem's nature. However, there are basic steps that such a process should follow:

- Identification of dataset is the fuel to a machine learning system. Sources of data and legal issues should be handled here.
- Preparation of the dataset is of critical importance. Before using data, mechanisms should be applied to avoid misleading decisions and deal with inaccurate or empty data.
- Select the proper algorithm according to the needs of the application and the applicability on the data
- Train your model using one of the known methods (supervised, unsupervised, reinforcement and deep learning)
- Evaluate the model created with different or similar algorithms and modes. An analyst might find that several approaches can be applied but one has the best fit.
- In the field, the model will get more and more trained if designed with feedback
- Get output data and make decisions
- Test model's judgments and even if wrong, tap them back to the data set to ameliorate the accuracy of the model.
- Always find more and more eligible data sources for continuous integration of the model.

4.10 Areas that ML can be applied

Machine learning is gaining the business trust in a variety of sectors. Apart from its current applications there are many fields where machine learning is invading. Having the machinery power of huge cloud computing infrastructure, CPU power is enough to enhance the use of machine learning with big data. In general, machine learning might be rather supportive, from health care to business and public sector. Some areas will need time to access and assess the use of machine learning, some other fields are strong candidates for imminent changes.

4.10.1 Healthcare

Concerning medicine, machine learning is already assisting in diagnosis and decisions for therapeutical schemas. A good example is the diagnosis of diabetes in very early stages. High quality images of the eye sclera, retina and macula are examined via machine learning techniques. The process searches patterns of blood vessels to find a potential pre-diabetic pattern, thus diagnose the disease before its onset. Google® has created a deep learning model that applies to this case.

4.10.2 Education

In education, machine learning can assist in the creation of personalized education programs that fit to a student's characteristics. Artificial intelligence could also be used to automatically grade exams in Massive Open Online Courses (MOOC) or to analyze how students react to an exam.

4.10.3 Transport-Logistics

As already discussed on MBA-61(Management course) of HOU, during the academic year 2017-2018, use of autonomous vehicles is a field with high potential in logistics sector. Machine learning plays a major role in achieving this innovation. Vehicles need to circulate with safety and make decisions on the fly which is a main field where machine learning, and deep learning can apply. Another approach in logistics is Amazon's® drone delivery service that uses machine learning techniques to send delivery drones instead of delivering personnel.

4.10.4 Government sector

Bureaucracy is dominating in public sector at a high level despite modern digitization and computer assisted techniques. In UK a service already exists, based on machine learning; it identifies potential needs of people according to their profile, taking a priori actions to tailor the public services that will be needed for applying a policy, targeting these people. An example is when unemployed young people need help in multiple levels including healthcare, social and psychological or financial to cope with the situation. An algorithm is used in this case to predict and trigger public sector services, proactively, for the situation by simply digging in the person's public records.

4.10.5 Finance

Banking and finance immensely use machine learning applications. In banking, Machine learning checks for patterns that would fit to unusual spending or lending activities and trigger an investigation. Also, robotic bank-tellers are used to respond to customer demands or questions. Voice recognition is used to authorize access to bank accounts via e-banking.

4.10.6 Pharmaceuticals Research and Development

In this sector clinical trials on new medical products create volumes of data and it is a source of data where someone could just claim that it fits to machine learning. These data from trials and even after FDA approval, could help targeting the most valued drug to an individual, maximizing the effectiveness of treatment. Another field where machine learning could fit, is the research for new drugs. Based on historical data of organic chemical substances used for certain diseases, a machine learning-enabled environment could propose the synthesis of new drugs for certain diseases and predict their effectiveness.

4.10.7 Energy

A massive pain-point concerning energy infrastructure failures is the demand on peak hours. Machine learning techniques can analyze the daily and seasonal demand distribution over an energy network and propose additional backup systems or expect and predict failures in the network. Google® has developed DeepMind® that uses machine learning for the amelioration and effectiveness of the micro-climate in its Data Centers predicting temperatures and other variables. This had a positive effect on the energy consumption of 40%.

4.10.8 Tax deviations

This could be characterized of great importance for countries like Greece at the time being. Machine learning can be used to predict tendency to not comply with tax law for individuals and businesses. Based on historical data and patterns that had been proven as tax evasion incidents it might trigger auditing and control. For example, in cases that the machine learning application poses a strong likelihood that an entity is trying to deviate from tax law.

4.10.9 Hardware maintenance

This is a field of major importance in all aspects of our lives. Machine failure was till recently, leveraged with redundancy only. For example, an aircraft has fully redundant systems that can cope with triple and quadruple failures. Of course, machine learning is not the impediment to the further designing redundant systems. Machine learning will rather predict multiple failures or single failures by training on historical data for any hardware equipment and not only for aircrafts. There are certain patterns before a failure, for which we do not need to know the root cause, but we further need to tap on a machine learning input system and it will find the pattern. In this way, business will save capital and other costs related to a maintenance or replacement of multiple failed units. This is because they will be conducted by the machine learning predictive application, to organize a scheduled maintenance because of the likelihood of a certain component to fail. This way it will be less costly to maintain a single component than let it fail, which will subsequently cause more costs for the business.

4.10.10 Advertising and Retail

Machine learning is already used by Advertising Bots which show products, that may interest you in Facebook and other pages, having processed your frequent internet buying habits or by analyzing your browsing habits. So, advertising turns to be very personal and accountable to the needs of an individual. In the retail sector a major improvement was achieved by Amazon® using artificial intelligence and machine deep learning. Automated super market was a milestone in the retail sector. The individual, shops in the store with physical presence but the procedure is not attended by personnel. In this case we have the co-operation of machine learning and artificial intelligence techniques in managing a physical store without personnel.

5. AI and ML in Stock Portfolio management -Case 1

As mentioned, it is profound that stock portfolio optimization and management would not escape from machine learning in terms of huge amounts of data archived.

5.1 Overview of the use of ML

Some machine learning techniques are used to identify singularities or signals on price changes. It is such a big field with plenty of data that no one could ignore the slightest benefit with the use of artificial intelligence methods. The main concern and output of ML techniques is the predictions of volatility, price levels and risk for a variety of time frames. A scientist, on the other hand, should not expect this to work without human interference. Portfolio managers co-operate with scientists designing approaches on machine learning and securities' portfolios to achieve the best results for an investor. Systematic risk is very hard to predict. Singularities over the world market might arise at any time as a result of political or social events. However, as a helper function, machine learning can detect signals, moments before a major event.

5.2 Overview of logistic regression algorithm in ML

Logistic regression is a well-known machine learning algorithm mostly used as classifier. In this case a classification of two classes is expected to be predicted by this ML algorithm. Using the sigmoid function g over a linear equation z gives the following:

$$z = \theta_0 + \theta_1 \chi_1 + \cdots . g(x) = \frac{1}{1 + e^{-x}} g$$

Tapping z as a variable to g, we obtain g(z):

$$g(z) = \frac{1}{1 + e^{-z}}$$

This function returns a value within [0,1] with use of a cost function and its partial derivatives we can obtain the gradients thus θ_0, θ_1 values and predict the class. With regards to the possibilities, if they seem more than 50% for 0 the prediction is marked belonging to class 0. Otherwise it is classified as class 1.

5.3 Support Vector Machines overview

Support Vector Machines is a useful and popular model dealing with data classifiers. It is not the case of Deep Learning or Neural networks and is considered easier to use. A data scientist should rather concentrate on the refining of the data to use with any machine learning model. The machine learning algorithms and their source code are not subject to major changes. Only hyper-parameters' tuning is available.

5.3.1 Overview

We will present a top down approach of libSVM (Hsu et al, 2016) with some focus in its heart algorithm as this library is trending nowadays. SVM classifier is used to separate data in two categories:

- A training set which is a labeled set of features thus it is classified a priori and is analyzed by SVM core during the training phase.
- A testing set which is a labeled set of features. After removing the label, the SVM core should accurately predict the correct label. This is a cross fold validation algorithm.

Suppose that we have i instances of x_i parameters to analyze and their class or label is y_i. (x_i, y_i) with i = 1, . . . ,n with $x_i \in$ R and y $\in \{1, -1\}^L$, the SVM will compute the solution to the following:

$$\min_{w,b,\xi} \quad \frac{1}{2} w^T w + C \sum_{i=1}^{l} \xi_i$$

$$\text{with} \quad y_i(w^T \varphi(x_i) + b) \geq [1 - \xi_i] \ , \quad \xi_i \geq 0$$

The training vectors x_i are mapped to a high dimensional space by $\varphi(x_i)$. SVM computes a linear separating hyperplane with maximum margin. C is as penalty variable of error. There are 4 basic SVM kernels:

- Linear $K(x_i, x_j) = x_i{}^T x_j$
- Polynomial $K(x_i, x_j) = (\gamma x_i{}^T x_j + r)^d, \gamma > 0$
- Radial basis function $K(x_i, x_j) = exp(-\gamma |x_i - x_j|^2), \gamma > 0$
- Sigmoid $K(x_i, x_j) = \tanh(\gamma\, x_i{}^T x_j + r)$

5.3.2 Data processing and scaling for the model

SVM will evaluate a certain data type as an instance. All the numbers must be real, and for categorical data such as images or voice, they need to get transformed to numerical data. The form of data accepted by the algorithm is as follows:

class 1:<value> 2:<value> …. n:<value>

which is the class followed by a TAB and then the number of features n followed by colon and their real value, then TAB as below:

```
-1 1:0.058 2:0.195 3:0.311 4:-0.29 5:-1 6:-0.13 7:-0.85 8:-0.73
```

What is more, comma separated (CSV format) or .xls data are acceptable by some libraries.

Scaling the data is a very important step towards success with SVM. All the data given must be of the same importance and scale. For example, if we scale down training data from [-20, +20] to [-2,2] we need to similarly rescale the testing dataset.

5.3.3 The kernels

RBF kernel is of first choice. It is a nonlinear kernel and has the potential to start mapping class labels to instances even if there is no linear relationship between them. The Linear kernel is mainly used for data with linear relationship. The polynomial kernel has more parameters to deal with. If the number of features is greater than the number of instances Linear kernel is better of RBF kernel. If we deal with huge number of features and instances libSVM will get slower on training.

5.3.4 Cross validation

Regarding RBF kernel we can see that it makes use of 2 parameters C, γ. Before training, it is not evitable to find the most suitable set for these variables. Our ultimate goal is to identify this couple in order to get the most efficient results. The n-cross-validation strategy enters here; we divide our training set in n subsets of equal size. Then subset 1 is used to test the SVM and the remaining subsets are used to train the model. This maximizes the cross-validation accuracy.

5.4 Case with Support Vector Machines and Logistic Regression

In this case (Dai et al, 2018) scientists have worked with libSVM to predict optimized portfolios. Data were downloaded via Bloomberg which poses a credit to the validity of the historical ticker data. The selection of data was based on the following categories:

- Middle priced 10-30 USD
- Last 300 of index SP500
- Average Daily Volume lying on the 33% percentile
- Chose diversified sectors

5.4.1 Data analysis and features

Data were downloaded for 23 tickers and were normalized to be eligible for machine learning techniques as per requirements. The tickers are shown in figure 5.1:

Stock Ticker	Origin
APOL	US Equity
CBG US Equity	
CMA	US Equity
CMS	US Equity
CVS	US Equity
GCI	US Equity
GME	US Equity
GT	US Equity
JBL	US Equity
KIM	US Equity
LNC	US Equity
NFX	US Equity
NI	US Equity
NWL	US Equity
NYX	US Equity
PWR	US Equity
QEP	US Equity
SEE	US Equity
TER	US Equity
THC	US Equity
TIE	US Equity
TXT	US Equity
ZION	US Equity

Figure 5.1 The 23 tickers used in the model

(Source: CS229 Project Report Automated Stock Trading Using Machine Learning Algorithms)

So, they were imposed to 1-minute time discretization from 9.00am to 5.00pm which is the end of business day. Every minute was assigned 8 features:

- Close prices/volumes: 1-minute end prices/volumes
- Open prices/volumes:1-minute interval opening prices/volumes
- High prices/volumes: the highest minute prices/volumes:
- Low prices/volumes: the lowest prices/volumes within the minute

Given the above identifiers the algorithm is used to predict the closing prices/volumes for the 1-minute interval prior to this minute interval. A model evaluation took place for one-day real exchange prices. For each portfolio, the two models (SVM and logistic Regression) invest one share or do not invest according to the simple decision algorithm:

For each minute:

1. Given the opening price Popen predict the closing price Pclose.
2. If predicted Pclose > Popen then invest 1 share to this ticker.
3. Else do not invest
4. Sell the shares obtained at the end of the minute
5. Update the Rolling Profit as: [Predicted] Pclose – Popen

At the end of the business day the same measures were calculated for the two models.

5.4.2 Metrics

The metrics used to evaluate the performance of the model were:

- accuracy = $\frac{correct\ predictions}{total\ predictions}$

- precision = $\frac{accurate\ uptick\ predictions}{uptick\ predictions}$

- recall = $\frac{accurate\ uptick\ predictions}{actual\ upticks}$

5.4.3 Logistic regression model

This model makes use of MNRFIT library and six features as below based on minute t:

[% change in open price] = $\frac{P_{open}^{(t)} - P_{open}^{(t-1)}}{P_{open}^{(t-1)}}$

[% change in high price] = $\frac{P_{high}^{(t-1)} - P_{high}^{(t-2)}}{P_{high}^{(t-2)}}$

[% change in low price] = $\frac{P_{low}^{(t-1)} - P_{low}^{(t-2)}}{P_{low}^{(t-2)}}$

[%change in open Volume]= $\frac{V_{open}^{(t)} - V_{open}^{(t-1)}}{V_{open}^{(t-1)}}$

[%change in high Volume] = $\frac{V_{high}^{(t-1)} - V_{high}^{(t-2)}}{V_{high}^{(t-2)}}$

[%change in low Volume] = $\frac{V_{low}^{(t-1)} - V_{low}^{(t-2)}}{V_{low}^{(t-2)}}$

The model with these features did not give accurate results because as someone can see, more than 2 timestamps get involved and are obscuring any patterns that can be found with this machine learning library.

To better tune the multiple timestamps and thus the dimensions of the feature set, cross-validation was applied indicating that features 1 and 4 can give the best results. So [% change in open price] and [% change in open Volume] were used. Apart from the optimization of the feature set, the training set was also optimized with cross-validation technique and a final training duration of 60 minutes was selected as the most fitted to this model. Performance for different training durations is shown below:

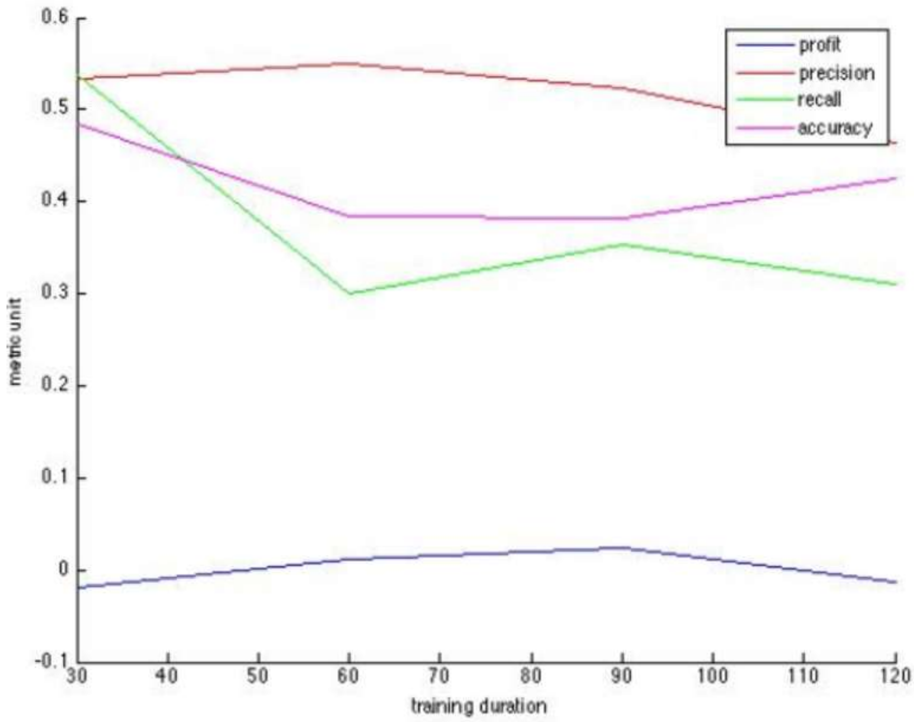


Figure 5.2 Performance for different duration with ML Logistic Regression Algorithm

Source: CS229 Project Report Automated Stock Trading Using ML Logistic Regression Algorithm

From the above, after training had finished, the results were the following:

precision 55.07%, recall 30.05%, accuracy 38.39%, profit 12%

A better approach to the above model was selected based to the fact that time plays a significant role on the model and the results. For example, a massive decision strategy to sell or buy stocks goes beyond the scope of this 1-minute timeframe. That is, signals take time to appear and 1-minute changes are not considered to be effective. For this reason, new features were issued as follows in a λ -minute high-low model with a λ=5. The λ-minute high-low model checks high/low price and high/low volume regarding all portfolios across all the tickers in a λ-minute frame as follows:

$$PH_{\lambda}^{(t)} = \max_{t-\lambda \le i \le t-1} P_{high}^{(i)}$$

$$PL_{\lambda}^{(t)} = \max_{t-\lambda \le i \le t-1} P_{low}^{(i)}$$

$$VH_{\lambda}^{(t)} = \max_{t-\lambda \le i \le t-1} V_{high}^{(i)}$$

$$VL_{\lambda}^{(t)} = \max_{t-\lambda \le i \le t-1} V_{low}^{(i)}$$

The new features with the λ model on top of logistic regression would be:

[% change in open price] = $\frac{P_{open}^{(t)} - P_{open}^{(t-1)}}{PH_{\lambda}^{(t)} - PL_{\lambda}^{(t)}}$

[%change in open Volume]= $\frac{V_{open}^{(t)} - V_{open}^{(t-1)}}{VH_{\lambda}^{(t)} - VL_{\lambda}^{(t)}}$

reflecting the most recent -minute high-low spread.

With a training period of 60 minutes the graphs is shown below:

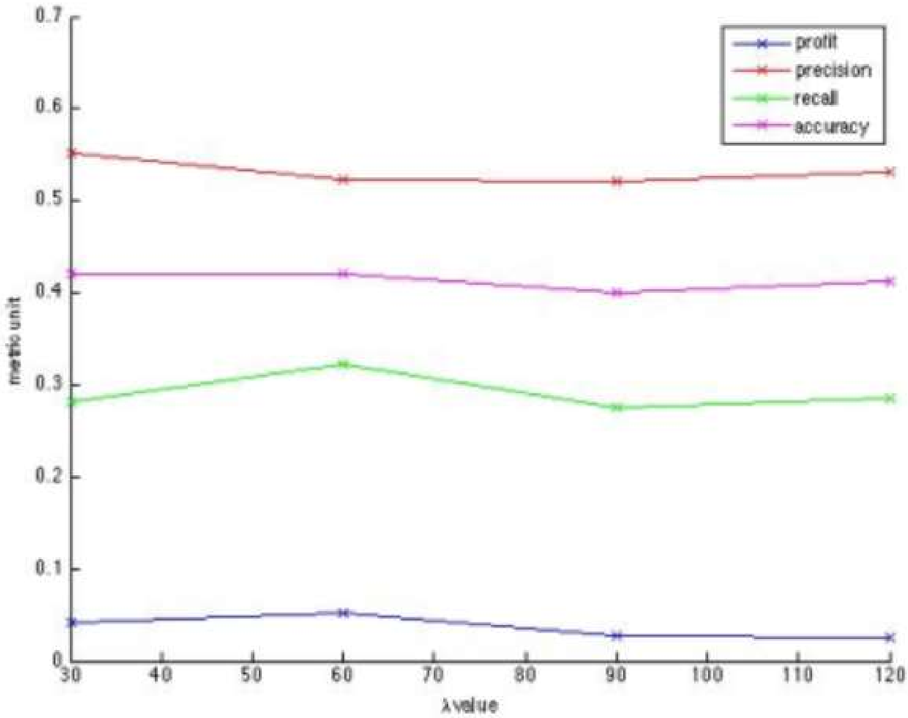


Figure 5.3 Metrics for different λ

(Source: CS229 Project Report Automated Stock Trading Using Machine Learning Algorithms)

From the above, after training had finished, the results were the following:

precision 59.39%, recall 27.43%, accuracy 41.58%, profit 18.6%

Thus, λ model linear regression model over 1-minute linear regression seems having superior results. This proves that time locality of stock exchange data plays a significant role and a time locality of 5 minutes tends to reflect the time deviation of ticker tendency.

5.4.4 Support Vector Machines model (SVM)

The aforementioned SVM model for machine learning was applied for the same data. From the original set of features all possible combinations were tested with the machine learning model and apparently the most promising set of features was to use all 8 features. Using cross validation, C=0.1 and λ=10 was decided for the tests and as for the kernel linear kernel gave better results.

From the above, after training had finished, the results were the following:

precision 47.13%, recall 53.43%, accuracy 47.13%, profit 30.66%

which shows a double better recall rate than the previous logistic regression model

Time-Locality adjustments with λ minutes high-low model applied on top of the SVM model did not reveal major upward changes.

In this study SVM proves to be a better model out of the two models in question. On the other hand, the rates are not adequate to indicate that this model can predict stock values. We should evaluate possible changes to the model proposed by the scientists.

6. AI and ML in Stock Portfolio management -Case 2

6.1 Deep Reinforcement Learning overview

Reinforcement learning as already mentioned, is a machine learning method that communicates with the environment and the model improves with try-and-error algorithms. Another sector of machine learning that was mentioned above, Deep learning, can approximate unknown problems and provide a solution even when data are scattered. The combination of these models of machine learning is called deep reinforcement learning and has contributed to excellent results in robotics and game experiments sectors. For instance, Q-learning is gaining credits with its neural network approximation of Q-value function. Another model of deep reinforcement learning is deep deterministic policy gradient (DDPG) which uses actor-critic techniques to achieve robustness in training process. Proximal policy optimization (PPO) is another technique. Overall deep reinforcement learning is considered to give results that fit to our expectations because the financial and assets datasets contain low SNR.

Deep reinforcement learning has some advantages versus deep learning or reinforcement learning. At first, with datasets as input and a weights vector as output, this type of learning is an, end-to-end, artificial intelligence solution that improves as time pass. What is more, deep reinforcement learning might evaluate and predict stock performance which is a very hard problem. Also, compared with basis reinforcement learning it has the advantage of use of neural networks that can turn this model to a scalable prototype because neural networks are not impeded by high dimension issues, thus a portfolio containing thousands of securities.

6.1.1 Deep Q networks

Q-learning is using the action-valued function. Given a Q (s, a) function where a is the action and s is the state, this machine learning method creates a bi-dimensional matrix with all possible combinations of Q values (Hui, 2018). The dominant action (like SPNE again) is the action that maximizes Q (s, a). All the possible values of Q (a, s) are stored in the matrix. So, given a state's the machine needs to find the Reward, R, and the new state s'. Given the matrix the action to be chosen can also be found as a'.

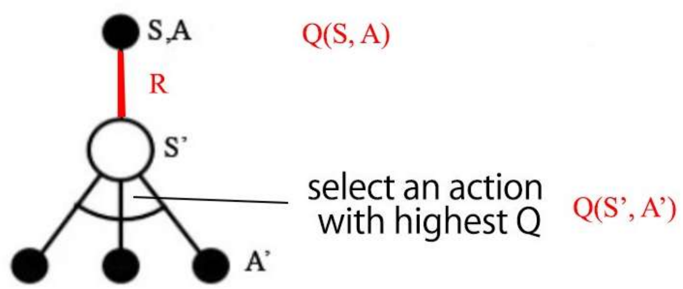


Figure 6.1 Q Network with reward R

(Source: RL—DQN Deep Q-network)

So, the target $is\ R + max[Q(s, a)]$

Algorithm of Q-networks:

Start with $Q_o(s, a)$ for all s, a

Get the initial state.

For $k = 1,2,3....,n \ with \ n \equiv convergence$

Try action a and get next state s'

If terminal, then:

Target = $R(s, a, s')$

Try new initial state s'

Else:

Target = $R(s, a, s')$ +γ max $Q_k(s', a')$

$$\underset{s \leftarrow s'}{Q_{(k+1)}}(s, a) \leftarrow [1 - a]\, Q_k(s, a) + a * [target]$$

If the combinations of Q states and Q actions are high, we could run out of memory. To override this issue Deep Q network can be used instead and is called deep Q learning. Deep Q-learning is approximating the Q function. A deep network is built to learn possible values of Q given that the target values are changing while proceeding. So, it is a Q(Q) iterated-function.

Algorithm of Deep Q-networks:

Start with $Q_o(s, a)$ for all s, a

Get the initial state-s

For $k = 1,2,3....,n \ with \ n \equiv convergence$

Try action a and get next state s'

If terminal, then:

Target = $R(s, a, s')$

Try new initial state s'

Else:

Target = $R(s, a, s')$ +γ max $Q_k(s', a')$

$$s \leftarrow s' \theta_{k+1} \leftarrow \theta_k - a\nabla_\theta E_{s' \sim P(s' \mid s,a)} \{Q_\theta(s,a) - target(s')\}^2 |_{\theta=\theta_\kappa}$$

6.1.2 Deep Deterministic Policy Gradients (DDPG)

Although Deep Q networks are popular, they are not designed to deal with continuous moves and actions. For example, in robotics one cannot resize the dimensionality of the continuous actions needed. Google® Deepmind after re-visiting DQN, designed a policy-gradient actor-critic algorithm which is called Deep Deterministic Policy Gradients (DDPG). Policy-Gradient algorithms use a stochastic policy μ (a, s) distributed over possible actions by probability. There are rewards for better actions taken and penalties for bad actions which cost.

The actor critic algorithms have the following architecture:

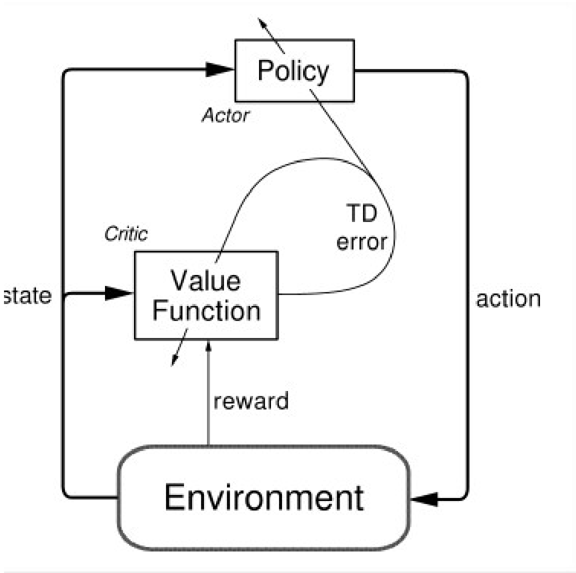


Figure 6.2 DDPG diagram

(Source: RL—DQN Deep Q-network)

The policy in figure 6.2 represents the actor while the value function is the critic. So, as we can see the actor produces an action that fits in the current state of the environment and the critic outputs a Temporal Difference error function given a state and a Reward. Then the model examines whether the Critic has an estimate of the Q (s, a) from Q-networks. Finally, the output of the critic reinforces learning for both the actor and the critic.

DDPG is an actor-critic algorithm using two neural networks; one for the actor and the second for the critic. An error signal (TD) is created by these modules for each timestamp so that the input of the actor is the state and the output is a real number that shows the actions selected from a continuous space in R. The critic's output is the estimated Q-value given the state and the actor's action. Based on the deterministic policy gradient theorem the model gets the updated weights of the actor network while the critic is updated by the TD error.

6.1.3 Proximal policy optimization (PPO)

Many algorithms for policy optimization are classified in three main categories.

- Policy iteration
- Policy gradient
- Derivative-free

Proximal policy optimization belongs to policy gradient optimization category. The PPO is based on Trust Region Policy Optimization (TRPO). TRPO works as follows:

- Start one step towards the Trusted Region for policy improvement, overseen by a subjective function M
- Approximate the expected reward η and enhance the area if η>0 else shrink the area.
- Optimize M and use the new optimal policy for M

It iterates computing the advantage rather than the reward which is more effective. In reinforcement learning the algorithms contain a lot of separate parts very hard to troubleshoot and debug. Furthermore, tuning the parameters has a prerequisite of excellent skills in programming and algorithms implementation. PPO makes the case easier among all the parts of the model including hyper parameters tuning. The fundamental reason that is responsible is the one-step at a time and the update after the step is performed. This ensures that cost function is minimal and that the model did not deviate a lot from the previous policy.

6.1.4 Policy Gradients Adversarial Policy Gradient (PG)

The scope of a Reinforcement Learning agent is to maximize the reward of following a policy π. Like in any Machine Learning model, the user is defining a set of parameters θ to advance the parametrization of (π, θ). If the total reward is projected for a known trajectory τ as $r(\tau)$, we get the following definition.

Reinforcement Learning Objective: Maximize the "expected" reward given a policy parametrization.

The Policy Gradient Theorem: The derivative of the expected reward is the expectation of the product of the reward and gradient of the log of the policy π_θ.

In this stage Markovian Decision Process are to be used along with adversarial training methods that are adding noise in the tickers data. The revised adversarial policy gradient algorithm proposed in the paper is as follows:

1. Actor is initialized
2. initial observation is received
3. Add noise to the price of the data
4. Select an action and observe/save the transition
5. Update the actor policy by the policy gradient

6.2 Case with Adversarial Deep Reinforcement Learning

Deep reinforcement learning is a strong candidate for portfolio management because of its ability to record and manipulate nonlinear features in the set, without any pre-training assumption as a result of use of neural networks. A priori, no deterministic human action exists which also fits the deep reinforcement model. Although these characteristics seem propagating this model for portfolio optimization there are some pain points.

- Stock Exchange markets are very risky and volatile in contrast to robotic vision-AI
- Reinforcement learning is designed for infinite Markowian decision process and in our case, we need a finite horizon for trading and maximizing portfolios returns.
- Stocks have definite metrics for portfolio optimization. So, approximations of portfolio values might give non-conclusive results.

In this case (Liang et al, 2018) and to fit the general rules for portfolio optimization neither DDPG nor PPO model had satisfying results on the training stage, so they were opted out. Instead, adversarial training and adversarial deep learning technique was proposed adding random noise in the market prices.

6.2.1 Data set

The experiments of this case were carried out in China Stock Exchange market. Five assets were randomly chosen from the available assets. To ensure data integrity, the historical data minimum was set to 1200 days to start the learning process. If one stock does not have 1200 days of history another set is rebuilt to match this constraint. Moreover, prices get normalized by division of open, close, high, low prices given last day's closing price. Another pinpoint is the missing data during bank holidays. This issue was handled by filling the missing data with the closing price of the previous days and by setting volume to zero for the specific days.

6.2.2 Network architecture

The model is based on Identical Independent Evaluator model (IIE). This means that networks work in parallel for the assets while sharing the same network parameters. Each network evaluates one ticker and outputs a weight % of the portfolio as an investment reference. Afterwards, all the weights of the assets are normalized and fitted in a weight matrix that is the next period's advised action. IIE has the benefit of scalability which is critical, regarding the portfolio size. Apart from the model Deep Residual Network is applied as the core network component. In the picture below a residual block of Deep Residual Network is demonstrated.

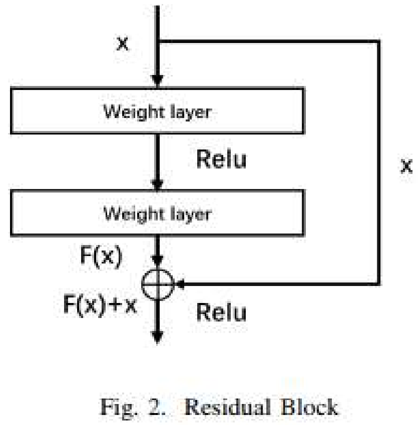


Figure 6.3 Block of Deep Residual network

(Source: Adversarial Deep Reinforcement Learning in Portfolio Management)

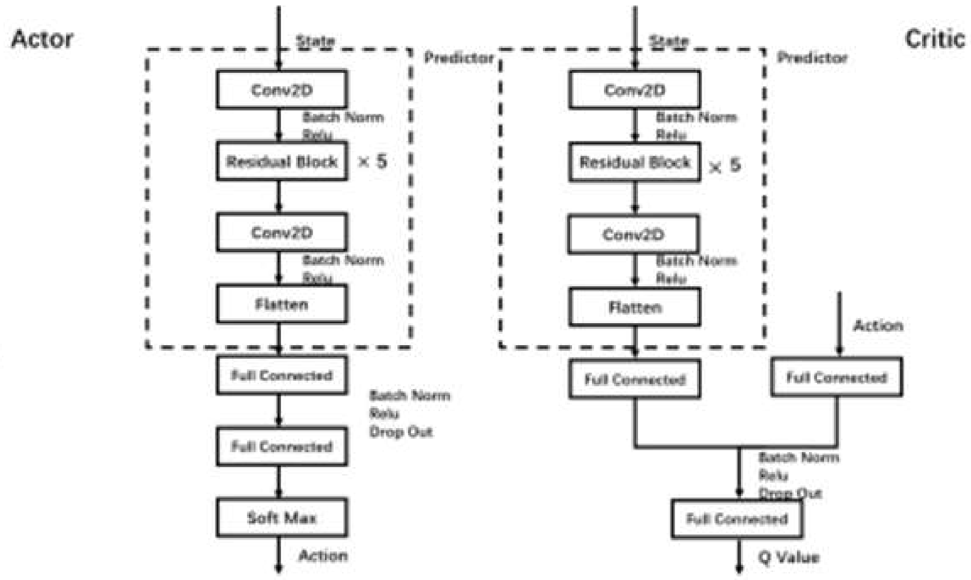


Figure 6.4 DDPG Network architecture

(Source: Adversarial Deep Reinforcement Learning in Portfolio Management)

The main network architecture is represented in the above picture 6.4.

6.2.3 The outcome

Measures of the learning rate were conducted to the models of DDPG and PPO as it is presented in figure 6.5. This rate is essential for neural network training. The learning rate that best suites to the portfolio optimization case is the DDPG case which was issued and implemented using different learning rates. An assumption was made that when actor learns a new pattern, the loss of the critic gets very high. The stable efficient result drops in when actor stabilizes.

Algorithm	DDPG		PPO	
	Actor	Critic	Actor	Critic
Optimizer	Adam	Adam	GradientDescent	GradientDescent
Learning Rate	10^{-3}	10^{-1}	10^{-3}	10^{-3}
τ	10^{-2}	10^{-2}	10^{-2}	10^{-2}

Figure 6.5 Hyper parameters regarding the experiments

The training set of data was limited, so the reinforcement learning agent might underestimate the volatility in testing conditions, unfolding a disaster in the real market condition. There exist some approaches of portfolio evaluation and in this case the objective function of the model was set to:

$$R = \sum_{t=1}^{T} \gamma^t (r(s_t, a_t) - \beta {\sigma_i}^2)$$

That measures the volatility of the returns of asset-i on the last day L.

A constraint was used to minimize the profits from high volatility stocks so that the portfolio is not endangered. The model did not seem to work as modified, and the same results were harvested when using in the subjective function the Sharpe ratio. Concerning the features set, four feature sets were tested for training

- Closing price
- Closing and high price
- Closing and opening price
- Closing and low price

From the results in figure 6.6, we get to the point that feature set plays a significant role in the training process. In this case closing and high price features are to be used:

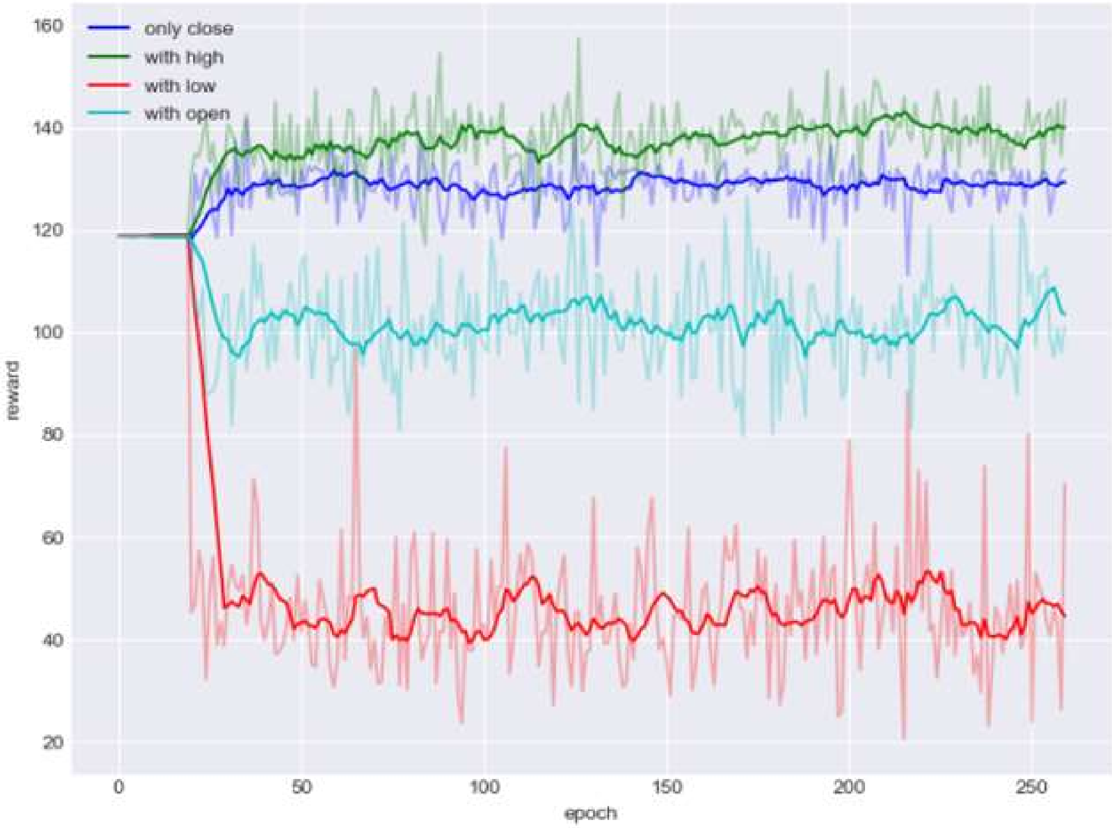


Figure 6.6 Comparison of reward R with different feature sets.

With the given model parameters, as aforementioned, and the subset of features the training was held for 1000 periods on both China and US stocks showing that the accumulative portfolio value gets higher with training:

Figure 6.7 Comparison of portfolio Value before and after DDPG training.

Figure 6.8 Comparison of portfolio Value before and after PG training.

At last the following hypothesis were evaluated with the adversarial learning model:

- H0: ARR1<ARR2 p-value is 0.00076 so with 99% confidence we can approve that adversarial learning improved AAR
- H0: Sharpe1<Sharpe2 p-value is 0.0338 so with 95% confidence we can approve that adversarial learning improved Sharpe ratio
- H0: MaxMarkDown1< MaxMarkDown2 p-value is 2.73e-8 so with 99.99% confidence we can approve that adversarial learning improved MaxMarkDown

Overall the model in this case study proves to be better with Adversarial Reinforcement learning as per figure below:

Figure 6.9 Comparison of portfolio Value with and without adversarial learning.

In this paper one can see how AI can deliver value after training using adversarial PG and DDPG. Further improvements of the algorithm and more testing might prove that this model is heading within a correct direction towards maximizing portfolio value over time.

7. Portfolio optimization using text analysis

7.1 Text and speech recognition

Speech is the most important means of communication among people. Apart from this, it seems to be the most convenient method of communication as it can be spontaneous and might share sentiments. The model of communication with human and computer interaction is called human computer interface. The last decades, progress of speech to text and text to speech conversions are massively used in everyday life. Many scientists had been designing efficient systems used for speech recognition. However, with machine learning and artificial intelligence algorithms, speech is processed more easily. Of course, with use of AI the language that derives from any text can be understood and analyzed by a computer regarding its meaning (Khilari et al, 2015). A speech recognition system overview can be seen below:

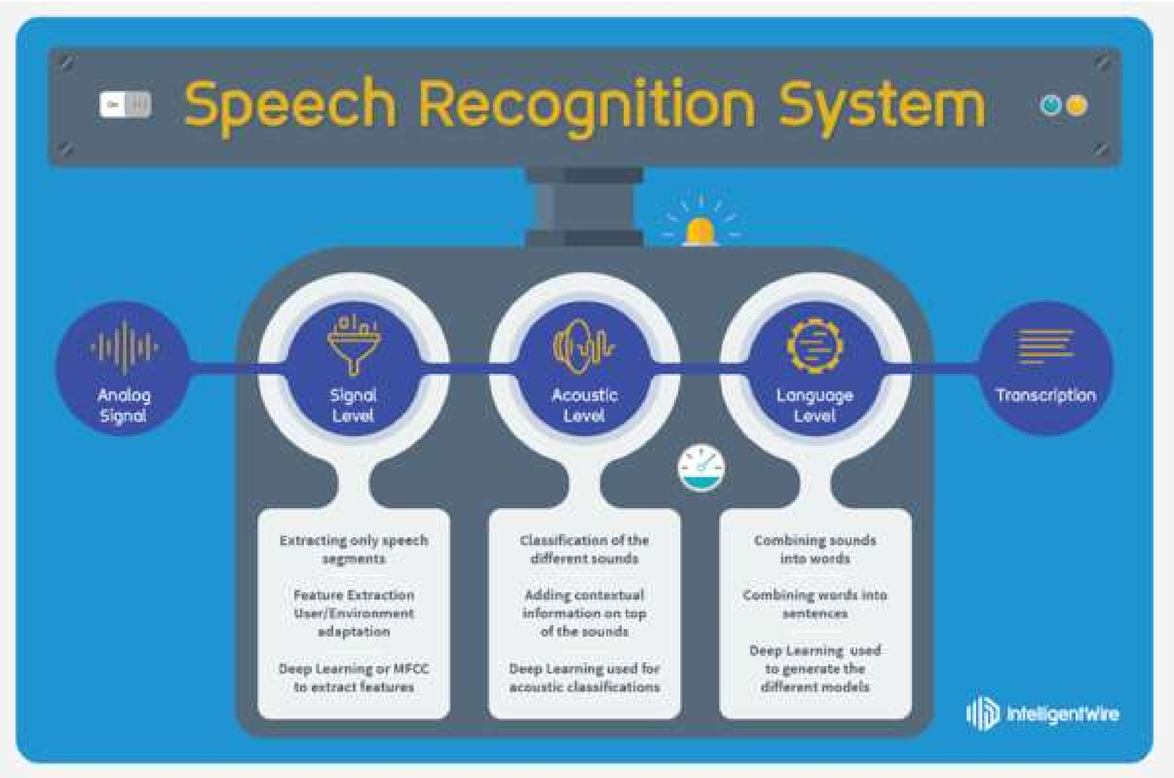


Figure 7.1 Speech recognition system

(source: intelligentwire)

7.2 Sentiment Analysis

Humans produce speech or text with unstructured characteristics of a natural language. To manipulate such data in a manner to extract efficient structured language that simulates a mother language, information beneath text must be dug. This takes place with AI techniques. One main meta-feature that can be revealed from text is the sentiment contained in the context. Sentiment analysis is not a new technique, with applications in the field of statistics and evaluation processes including metrics such as Likert scale.

Nowadays, with machine learning being a pioneer in speech recognition and analysis, combined with big data, scientists have been much interested in the field of sentiment analysis. Before extraction of sentiments from language-text, more basic features must be found and identified. These features are metrics of the presence and frequency of phrases as bag-of-words, occurrence of specific subsets known as noun phrase approach, and parts-of-speech tags (Bohn, 2018). Given these features, the machine learning model can be trained with input the features associated with a certain text and output the sentimental prediction.

A technique for sentiment analysis is the use of sentiment dictionary (Bohn, 2018) which acts as a map of individual words or phrases to a corresponding feeling-sentiment. In a phrase the sentiment of one and every word is measured and classified. Then, the whole sentiment is calculated for the text

In the case of sentiment regarding financial text data, Loughran-McDonald sentiment dictionary is used. This has the most current specifications regarding the language used in financial reports.

7.3 Data mining in the web

In order to get necessary information from vast volumes of text to apply for sentiment analysis, a tool is needed to dig for the data. This is the well-known data mining software or knowledge discovery from data as mentioned in chapter 4. The core of data mining is related with machine learning, statistics and data visualization. Another definition for data mining is data processing using sophisticated data search capabilities and statistical algorithms to discover patterns and correlations in large preexisting databases; thus, a way to discover new meaning in data. Concerning social media, data mining is finding and identifying new information in a constant manner regarding Twitter® or Facebook® or other social media pages. Data mining can deal with big structures of data that are constantly refreshed or posted in social media. Unsupervised or supervised machine learning algorithms are used to fetch and analyze data sets online. Data might have labels or might contain only a small portion of meta-data. When data has labels, supervised learning can be used to extract and classify labels for each set of data. The most common algorithms that are used when data have labels are the classification rules algorithm, SVM and the decision trees algorithm. For data with no label, clustering is used as an unsupervised data mining technique. Also, Bayesian classification can be used. Data mining in social media is a type of social engineering.

7.4 News and stock fluctuations

The financial markets are so complicated and volatile that predictions cannot be identified or be accurate with a viable percentage. At least, capturing some early signals of their volatility and then use this information to produce estimations of a stock exchange market tendency, is something possible. There are lots of sectors attracted by this phenomenon such as economics, computing statistics, data science and machine learning. The conventional approach categorizes stock markets based on historical records or phenomena and tries to find some patterns on the corresponding time series. However, analyzing timeseries is not very feasible and will not take into account the human or public reaction on a stock. With the use of social media where a subject can express his feelings, ideas or opinions there is a field of big data to study. In the past we had sparse news and low velocity in social media and in the exchange market (Guo, 2018). At that time as aforementioned, an individual had the time to make choices. Nowadays news are traveling in high velocities via the Internet causing high volatility fluctuations in stocks and, in our opinion, a reciprocal relationship between news and stock fluctuations exists. Yet, news spreading in the social media might cause fluctuations on a company stock. And the other way fluctuations upwards or downwards of the specific stock market, cause news to spread faster in the social media. Such a cycle of emerging news followed by a positive or negative fluctuation of a certain stock is constant. However, working with data mining on the fly, one can split this vicious cycle of events. Scientists can detect an early signal that might be related with a stock fluctuation. As presented, a sentiment analysis can take place using machine learning. In the literature models exist that can predict notable changes in the market using artificial intelligence with sentiment analysis.

8. ML Algorithm implementation for the Classification of stocks

Given the fact that all the securities in stock exchange markets are appropriate for a certain investor's profile, we might need to divide the stocks in classes depicting their characteristics. This way an investor might select objects from a certain class that fits his investment mentality. In this chapter we are presenting our approach to portfolio clustering applying support vector machines (SVM)machine learning algorithm with Orange® open source.

8.1 Empirical study classes

During an empirical study on multiple assets in Mumbai, India, their risk(σ), liquidity, long and short average return was studied (Dai et al, 2017). Then the assets had been classified empirically within three classes as follows:

- Class I: contains securities that their liquidity mean value is higher than the other classes. In this category we are not expecting high returns but rather moderate.
- Class II: in this class high-yield securities can be found with higher returns. They are estimated to be high in profit with a high standard deviation. On the other hand, their liquidity tends to be low, the lowest among the three classes. This class is suitable for investors with a long investing horizon.
- Class III: which is the class with less risky assets because they have the lowest standard deviation among the three classes and giving a medium return.

8.2 SVM model with fold cross validation

As already mentioned, we will use SVM model to train the dataset and classify the stocks The feature of 10-fold cross validation is used. This is a technique to check how a model will adapt to unseen, new data. The steps for 10-fold cross validation is presented below:

1. Use random sampling to divide the whole training set into 10 sub-folds be F1,F2,…,F10
2. Set an initial i=10 ;
3. while i>0 repeat Steps 4 to 8
4. Remove fold Fi from the training-set.

5. Train with SVM machine learning algorithm using the rest of data folds except Fi as training set.
6. The machine will create a model based on data from training set folds, then it checks its accuracy predicting the Fi data. Errors on the Fi prediction are saved.
7. After the accuracy test on Fi data, Fi is brought back to the original folds of the training set.
8. Set i=i-1

At the end of the training procedure, we get 10 results from the folds that can be convoluted to provide an estimation of the model's prediction accuracy. An advantage of the 10-fold cross validation technique in SVM is that all classified dataset is used for training and prediction/validation, having used each fold only one time for prediction. The latter creates a strong mechanism to measure our model's efficiency in learning from data. The SVM experiments were conducted with use of RBF kernel and different tuples(C,g) or (C,γ). Of course, we chose the parameters that gave the best cross validation accuracy. After many tests we found the tuple of (C,g) = (25,0.02) with an accuracy of 86.7%. After obtaining the optimal (C,γ), the SVM classifier is built for the training data.

8.3 Results of the classification training using SVM

In this point we present the model implemented with Orange® open source that has the libSVM embedded.

In the figure below, we can see the classification using 10-fold cross validation as mentioned above. On the first column we see the SVM prediction and the corresponding probability along with the features used. For example, ASSET A1 is classified as a Class-I asset with 0,97 probability, as class-II asset with 0.01 probability. Also, ASSET A16 is classified as a Class-II asset with 0,89 probability, as class-II asset with 0.09 probability.

	SVM	Class	ASSET	Short term return	Long term return	Risk	Liquidity
1	0.97 : 0.01 : 0.01 → 1	1	A1	-0.060770	0.112130	0.19770	0.006100
2	0.30 : 0.44 : 0.26 → 1	1	A2	0.172390	0.150580	0.18976	0.003590
3	0.63 : 0.05 : 0.33 → 1	1	A3	0.027550	0.117440	0.11776	0.003090
4	0.90 : 0.04 : 0.06 → 1	1	A4	0.056130	0.104970	0.18504	0.005690
5	0.96 : 0.02 : 0.03 → 1	1	A5	0.066380	0.116700	0.21634	0.009360
6	0.98 : 0.00 : 0.02 → 1	1	A6	0.068970	0.105482	0.16645	0.016790
7	0.60 : 0.13 : 0.28 → 1	1	A7	0.181090	0.203610	0.28852	0.011490
8	0.62 : 0.25 : 0.13 → 1	1	A8	0.058930	0.172780	0.23321	0.006330
9	0.88 : 0.07 : 0.04 → 1	1	A9	-0.003000	0.093700	0.19970	0.001840
10	0.29 : 0.08 : 0.63 → 3	1	A10	0.152570	0.115040	0.12004	0.002430
11	0.83 : 0.07 : 0.11 → 1	1	A11	0.149550	0.100500	0.20528	0.005700
12	0.44 : 0.53 : 0.03 → 1	2	A12	0.075060	0.164500	0.31853	0.004890
13	0.04 : 0.89 : 0.07 → 2	2	A13	0.174990	0.192770	0.14009	0.000310
14	0.04 : 0.93 : 0.03 → 2	2	A14	0.156110	0.213650	0.16853	0.001250
15	0.03 : 0.96 : 0.01 → 2	2	A15	0.200950	0.217100	0.20634	0.001230
16	0.09 : 0.89 : 0.02 → 2	2	A16	0.071380	0.198190	0.27252	0.001330
17	0.02 : 0.91 : 0.07 → 2	2	A17	0.339790	0.400860	0.42326	0.003510
18	0.01 : 0.98 : 0.00 → 2	2	A18	0.272900	0.308310	0.38341	0.002400
19	0.10 : 0.36 : 0.54 → 3	2	A19	0.077160	0.174480	0.11526	0.001040
20	0.04 : 0.88 : 0.08 → 2	2	A20	0.149340	0.232020	0.13323	0.002340

Figure 8.1 SVM classifier results.

In the next figure we can see the SVM predicted results versus the actual.

Regarding class-I : 17 assets were correctly predicted, 1 asset was wrongly-predicted to belong to class-II and 3 assets were wrongly-predicted to belong to class-III

Regarding class-II : 18 assets were correctly predicted, 1 asset was wrongly-predicted to belong to class-I and class-III

Regarding class-III : 17 assets were correctly predicted and 2 assets were wrongly-predicted to belong to class-I

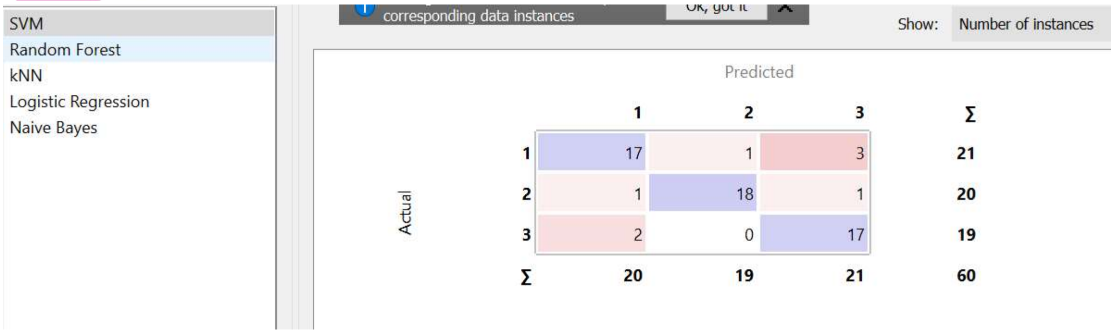


Figure 8.2 Predicted versus actual results.

In the figure 8.3 we can check the widest boundaries created by our model in a bi-dimensional chart where x-axis represents Risk and y-axis the liquidity. Class-I in green, class-II in red and class-III in green. Points' colors show the assets' class respectively.

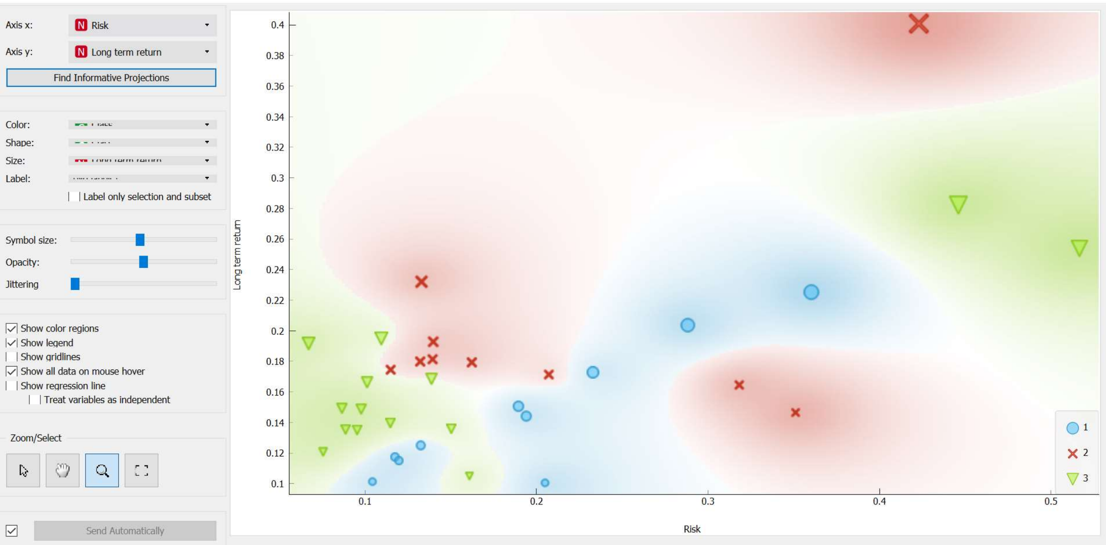


Figure 8.3 boundaries of classes with SVM

Here is the matrix of results of the model. We can see the real class, the asset number, the predicted class and the probabilities P1,P2,P3 that asset #i can belong to class I-II-III.

CLASS	ASSET#	Predicted CLASS	P1	P2	P3
1	A1	1	0.975	0.011	0.014
1	A2	1	0.317	0.401	0.282
1	A3	1	0.601	0.039	0.359
1	A4	1	0.904	0.030	0.065
1	A5	1	0.957	0.013	0.030
1	A6	1	0.981	0.001	0.019
1	A7	1	0.582	0.111	0.307
1	A8	1	0.641	0.221	0.138
1	A9	1	0.895	0.061	0.044
1	A10	3	0.266	0.067	0.667
1	A11	1	0.829	0.055	0.116
1	A34	1	0.976	0.000	0.023
1	A35	1	0.913	0.041	0.045
1	A36	1	0.997	0.001	0.002
1	A37	1	0.863	0.068	0.069
1	A38	1	0.887	0.003	0.110
1	A39	1	0.994	0.001	0.006
1	A40	1	0.884	0.001	0.115
1	A41	1	0.885	0.000	0.115
1	A42	1	0.636	0.058	0.305
1	A43	1	0.657	0.216	0.127
2	A12	1	0.498	0.468	0.034
2	A13	2	0.055	0.877	0.068
2	A14	2	0.048	0.922	0.030
2	A15	2	0.039	0.949	0.012
2	A16	2	0.113	0.861	0.026
2	A17	2	0.031	0.905	0.064
2	A18	2	0.016	0.980	0.004
2	A19	3	0.103	0.340	0.557
2	A20	2	0.052	0.870	0.078
2	A21	2	0.075	0.893	0.031
2	A22	2	0.032	0.961	0.008
2	A44	2	0.261	0.630	0.109
2	A45	2	0.156	0.823	0.021
2	A46	2	0.077	0.792	0.131
2	A47	2	0.012	0.982	0.006
2	A48	2	0.007	0.992	0.001
2	A49	2	0.066	0.835	0.099
2	A50	2	0.030	0.963	0.007
2	A51	2	0.094	0.805	0.101
2	A52	2	0.003	0.997	0.000
3	A23	3	0.067	0.097	0.836

3	A24	1	0.539	0.070	0.391
3	A25	3	0.207	0.065	0.728
3	A26	3	0.095	0.087	0.819
3	A27	3	0.068	0.048	0.884
3	A28	3	0.262	0.131	0.606
3	A29	3	0.050	0.072	0.878
3	A30	1	0.673	0.042	0.285
3	A31	2	0.083	0.491	0.426
3	A32	3	0.033	0.042	0.925
3	A33	3	0.203	0.015	0.782
3	A53	3	0.091	0.040	0.870
3	A54	3	0.071	0.101	0.827
3	A55	3	0.032	0.220	0.748
3	A56	3	0.198	0.037	0.766
3	A57	3	0.132	0.028	0.840
3	A58	3	0.137	0.069	0.794
3	A59	3	0.203	0.090	0.707
3	A60	3	0.291	0.096	0.613

Table 8.1 The results of the SVM classifier

At this point the model is ready to predict the proper class of any unknown asset given that all the three characteristics of liquidity , σ, average Returns are provided.See below 2 random ASSETS that were succesfully classified to belong to class-I

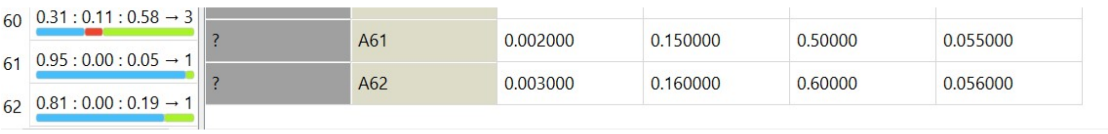
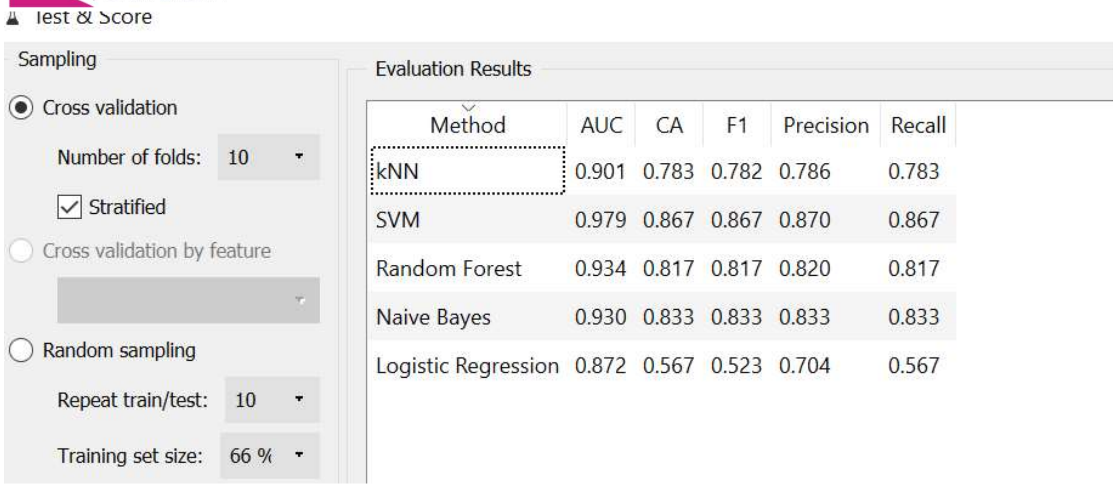


Figure 8.4 Classification of unknown assets A61,A62

8.4 Other ML algorithms used and their results

We also used kNN, Random Forrest and Naïve Bayes algorithms with good results while logistic regression showed the worst results with 56,7% accuracy.

Please check the figure below for the results of all models using 10-fold cross validation.



Method	AUC	CA	F1	Precision	Recall
kNN	0.901	0.783	0.782	0.786	0.783
SVM	0.979	0.867	0.867	0.870	0.867
Random Forest	0.934	0.817	0.817	0.820	0.817
Naive Bayes	0.930	0.833	0.833	0.833	0.833
Logistic Regression	0.872	0.567	0.523	0.704	0.567

Figure 8.5 Comparison of 4 models tested.

9. Conclusions and future work

In this dissertation we studied and presented the main strategies proposed regarding the Modern Portfolio Theory. Scientists have induced several approaches for the portfolio optimization problem. Most of the approaches are consisting of complex mathematical equations which are respected by the community regarding for their philosophy. This does not mean that all the models can be applied in real stock exchange markets. Companies that are dealing with portfolios prefer decent clientele and will never disclose the details of models or algorithms used embedded to proprietary software designed for portfolio optimization.

Artificial intelligence and machine-learning are considered of dominant position in informatics section the last decade. Many applications in real life make use of this modern technology. A lot of studies and papers are written concerning AI algorithms and proposals. The volume of the field is chaotic in terms of programming algorithms introduced. In this thesis we studied and presented two cases of portfolio optimization trials using machine-learning techniques and a classifier of stocks, application with SVM It seems that AI might help to the maximization of a portfolio performance regarding non-systematic risk and is more than obvious that it successfully classifies stocks in classes (our application).

In the future, scientists will continue working with machine-learning and portfolio optimization, achieving more promising results as time passes. In our opinion there is much way to go, but AI is assisting in a positive way the portfolio optimization issue, so it always gives positive feedback in making the right decision against non-systematic risk. Systematic risk, however, applies to all portfolios globally.

A next field of study could be the application of machine-learning techniques in order to predict or to find signals of a major distress in the financial landscape. This is a multi-variate issue involving techniques of web and social media data mining; a problem, the answers of which, lie in global leaders' decisions and diplomacy or major market joint-ventures. Instead of seeking inside information about all these non-disclosable data, we can concentrate on available data in the media. Data such as face analysis, motion analysis, sentiment analysis, body language analysis of important people with use of machine learning and AI. This might help dealing with major situations that can affect the globe financially. However, we cannot deviate from such situations, but we might better manipulate them to prevent a fully unstable financial environment. Hence, any major social or diplomacy incident will have effect in international stock exchange markets, but machine-learning could aid in the direction of a smooth transition preventing financial singularities.

Bibliography

Apt, K. R. (2003). Principles of constraint programming. Cambridge ; New York: Cambridge University Press.

Ban, E. A., & Feigenbaum, E. A. (1979). Handbook of Artificial Intelligence, 762.

Ban, G.-Y. (2017). Machine Learning & Portfolio Optimization, 30.

Beasley, J. E. (2018). Portfolio optimization, 55.

Bodie, Z., Kane, A., & Marcus, A. J. (2002). Investments . Boston, Mass.: McGraw-Hill/Irwin.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (2010). A Training Algorithm for Optimal Margin Classiers, 9.

Cordes, R., O'Toole, B., & Steiny, R. (2008). The art of investing and portfolio management: a proven 6-step process to meet your financial goals (2nd ed). New York: McGraw-Hill.

Dai, T., Shah, A., & Zhong, H. (2018). CS229 Project Report Automated Stock Trading Using Machine Learning Algorithms, 5.

Dunis, C., Middleton, P. W., Theofilatos, K., & Karathanasopoulos, A. (Eds.). (2016). Artificial intelligence in financial markets: cutting-edge applications for risk management, portfolio optimization and economics. London: Palgrave Macmillan.

Frey, A. H. (2013). A beginner's guide to investing: how to grow your money the smart and easy way.

FSB. (2017). Market developments and financial stability implications. FSB.

Gupta, P., Mehlawat, M. K., Inuiguchi, M., & Chandra, S. (2014). Fuzzy portfolio optimization: advances in hybrid multi-criteria methodologies. Heidelberg: Springer.

Hebert, M. (2007). 15-381: Artificial Intelligence. Retrieved from: http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15381-s07/www/

Hsu, C.-W., Chang, C.(2016). A Practical Guide to Support Vector Classification, 16.

Hui, J.(2018).RL—DQN Deep Q-network. Retrieved from:

https://medium.com/@jonathan_hui/rl-dqn-deep-q-network-e207751f7ae4

Hurwitz, J. (2018). Machine Learning For Dummies, IBM Limited Edition, 75.

Irizarry-Nones, A., Palepu, A., & Wallace, M. (2018). Artificial Intelligence (AI), 14.

Kenneth, S. (2011). Successful Investing. Empirical Wealth Management, (Q1 March2011).

Khilari, M. P. (2015). A review on speech to text conversion methods, 4(7), 6.

Kumar, A., & Goyal, R. (2010). Portfolio Management Using SVM, 13.

Levišauskait, K. (2010). Investment Analysis and Portfolio Management, 166.

Liang, Z., Chen, H., Zhu, J., Jiang, K., & Li, Y. (2018). Adversarial Deep Reinforcement Learning in Portfolio Management. Retrieved from http://arxiv.org/abs/1808.09940

Markowitz, H. (1952). Portfolio selection. The Journal of Finance, 7(1), 77–91. https://doi.org/10.1111/j.1540-6261.1952.tb01525.x

Markowitz, H. M. (1999). The Early History of Portfolio Theory: 1600-1960. Financial Analysts Journal, 55(4), 5–16.

Marling, H., & Emanuelsson, S. (2012). The Markowitz Portfolio Theory, 6.

Michaud, R. O., & Michaud, R. O. (2008). Efficient asset management: a practical guide to stock portfolio optimization and asset allocation (2nd ed). New York: Oxford University Press.

Mokhtar, M., Shuib, A., & Mohamad, D. (2014). Mathematical Programming Models for Portfolio Optimization Problem: A Review, 8(2), 8.

Qin, Z. (2016). Uncertain portfolio optimization. New York, NY: Springer Berlin Heidelberg.

Radlak, M. (2008). Artificial Intelligence with Modern Portfolio Theory, 57.

Roudier, F. (2007). Portfolio Optimization and Genetic Algorithms, 70.

Royal Society (Great Britain). (2017). Machine learning: the power and promise of computers that learn by example.

Vasiliou, D. (2005). Financial Management and Accounting Investment Analysis and Portfolio Management (Vol. 3). HELLENIC OPEN UNIVERSITY.

Wallenberg, M. (2018). THE ESSENTIAL AI HANDBOOK FOR LEADERS. Retrieved from: WWW.PELTARION .COM/AI -HANDBOOK

Xidonas, P. (Ed.). (2012). Multicriteria portfolio management. New York: Springer.

Author's Statement:

I hereby declare that, in accordance with article 8 of Law 1599/1986 and article 2.4.6 par. 3 of Law 1256/1982, this thesis/dissertation is solely a product of personal work and does not infringe any intellectual property rights of third parties and is not the product of a partial or total plagiarism, and the sources used are strictly limited to the bibliographic references.