EVOLUTIONARY ALGORITHMS-BASED APPROACH OF MULTI-OBJECTIVE PORTFOLIO OPTIMIZATION

Report submitted to

Indian Institute of Technology, Kharagpur

in partial fulfillment of the requirements for the award of the degree

of

Master of Technology

in Financial Engineering

by

Adarsh K

Under the supervision of

Prof. D K Pratihar,

Department of Mechanical Engineering

FINANCIAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY. KHARAGPUR

MAY, 2016

DECLARATION BY STUDENT

I certify that

a) The work contained in this report has been done by me under the guidance of my supervisor.

b) The work has not been submitted to any other institute for any degree or diploma.

c) I have conformed to the norms and guidelines given in the Ethical code of conduct of the Institute.

d) Whenever I used the materials (data, theoretical analysis, figures and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the reference. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

Date:

Signature:

CERTIFICATE BY SUPERVISOR

This is to certify the project report entitled "**Evolutionary algorithms-based approach of multi-objective portfolio optimization**" submitted by **ADARSH K (Roll No: 11MI3FP15)** to Indian Institute of Technology, Kharagpur, is a record of bona fide project carried out by him under my supervision.

(Signature of the Supervisor)

Prof. D. K. Pratihar,

Professor,

Department of Mechanical Engineering,

IIT Kharagpur.

ACKNOWLEDGMENT

With great pleasure and great sense of gratitude, I take this opportunity to express my sense of indebtedness of Prof Pratihar, for his valuable guidance and constant encouragement right from the inception of the problem to the successful completion of this study. He was always more than willing to help me, whenever I needed smallest help. I feel proud and honored to be a student of such personality.

Adarsh K
11MI3FP15,
Fifth year Undergraduate student,
Dual degree in Mining & Financial Engineering,
IIT Kharagpur.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

EA: Evolutionary algorithms

MOP: Multi-objective optimization problem

NSGA-II: Non-dominated Sorting Genetic Algorithm-II

MVT: Mean–variance theory

VaR: Value-at-risk

CVaR: Conditional Value-at-risk

CDaR: Conditional drawdown-at-risk

MOEA: Multi-objective evolutionary algorithm

MAD: Mean-Absolute Deviation

GMD: Gini's Mean Difference

LP: Linear programming

DCC: Dynamic Conditional Correlations

GARCH: Generalized Auto-regressive Conditional Heteroskedasticity

GARCH-SK: GARCH adjusted with Skewness and Kurtosis

EXECUTIVE SUMMARY

In portfolio optimization problem, we maximize the portfolio expected returns for a given amount of risk or minimize the portfolio risk for a given level of expected return. With time, there has been advancement in the financial literature, which led to the adoption of several proxies for risk, returns and the use of various techniques for optimization purpose. Evolutionary algorithms (EAs) have become the methods of choice for multi-objective optimization problems (MOP) that are too complex to be solved using deterministic techniques, as they are inspired by the biological processes which are inherently multi-objective. In this thesis, additional proxies for risk viz., Dynamic Conditional Correlations (DCC) i.e., time-varying correlations, Generalized Auto-regressive Conditional Heteroskedasticity adjusted with Skewness and Kurtosis (GARCH-SK) i.e., time-varying variance, Skewness and Kurtosis have been proposed. These are implemented along with the conventional proxies like Conditional Value-at-risk (CVaR) as multiple objectives. The multi-objective optimization problem has been solved using Non-dominated Sorting Genetic Algorithm (NSGA-II).

CHAPTER 1

PORTFOLIO OPTIMIZATION: INTRODUCTION

A portfolio is the mix of assets held by a portfolio manager. In investment decision making, portfolio optimization is an important process of diversification[1]. It is the process of selecting, from a feasible set of available portfolios, the subset which best achieve some objectives under given constraints. The future outcomes of a given choice of portfolio are almost always stochastic. The choice of objectives must represent the aims and preferences of the manager [12].

According to Markowitz's mean–variance theory (MV), an investor attempts to maximize portfolio expected returns for a given amount of portfolio risk or minimize portfolio risk for a given level of expected return. The MV theory was criticized for unrealistic assumptions. Responding to the critics, researchers incorporated to the portfolio model some real world constraints like bounds on holdings, cardinality, minimum transaction lots and sector capitalization constraints. Moreover they proposed new risk measures such as the value-at-risk (VaR), the conditional value-at-risk (CVaR)[2], and the conditional drawdown-at-risk (CDaR), Dynamic Conditional Correlations (DCC) i.e., time-varying correlations, Generalized Auto-regressive Conditional Heteroskedasticity adjusted with higher order moments-Skewness and Kurtosis (GARCH-SK) i.e., time-varying variance, Skewness and Kurtosis [3]. However, these additional constraints and risk measures made the Portfolio Selection problem difficult to be solved with exact methods.

Evolutionary algorithms (EAs) have become the method of choice for optimization problems that are too complex to be solved using deterministic techniques. EAs are well suited to multi-objective optimization problems (MOP) as they are inspired by the biological processes which are inherently multi-objective. Because the various objectives functions in the portfolio selection problem are usually in conflict with each other, each time that we attempt to optimize further an objective other objectives suffer as a result. Therefore, the objective in MOEAs is to find the Pareto front of efficient solutions that provide a tradeoff between the various objectives.

The key reason for interest in sophisticated optimization procedures is for its real practical applicability to agents for whom it can deliver value. Such agents include financial institutions (banks, insurance firms, hedge funds, investment managers and the like), high net worth individuals and trusts, sovereign wealth funds. For these groups, effective optimization decisions can have significant impact on the nature of their continued existence. Indeed, in the case of mutual funds, it is one of the principal justifications for them being in business at all [10]

1.1 RESEARCH GAP AND OBJECTIVES:

Justification for choosing the objective functions based on literature:

- **CVaR:** According to Uryasev and Rockafellar, all the risk measures don't consider tail risk [28].
- **Skewness, Kurtosis:** Returns are assumed to follow normal distribution, but they don't. Hence, we have to account for skewness (a measure of the level of asymmetry), kurtosis (a measure of the thickness of the tails). From Arrow, Merton, Ralph Scenario based portfolio optimization model [53, 54, 55]
- **Dynamic Conditional correlations:** Risk due to other assets must be considered -Takayuki and Taniguchi [59]
- **Credit ratings:** Because people tend to invest more in high credit rated firms irrespective of risk and return-Chiranjeevi and Sastry[49]

With time, there has been advancement in the financial literature, which led to the adoption of several proxies for risk, returns and the use of various techniques for optimization. Earlier, portfolio optimization problems have taken into account just one or some of these recommended objectives functions. Here in this thesis, we have considered all the above objective functions, tried various combinations of these to determine which of them yield the best risk-adjusted returns.

Also, Now-a-days, Evolutionary algorithms have become the methods of choice for multi-objective optimization problems (MOP) that are too complex to be solved using deterministic techniques, as they are inspired by the biological processes which are inherently multi-objective. Therefore, here Non-dominated Sorting Genetic Algorithm-II (NSGA-II), one of the most popular evolutionary algorithms, was implemented for the optimization purpose.

CHAPTER 2

LITERATURE REVIEW

In 1950's Markowitz [11, 12] proposed the portfolio selection problem in terms of two criteria to be optimized: the mean of the outcomes to be maximized, and the risk, a measure of the volatility of outcomes to be minimized. According to Markowitz, the investor cares about only expected return, volatility and not about the distribution of returns, its skewness (a measure of the level of asymmetry) or its kurtosis (a measure of the thickness of the tails).

The original Markowitz formulation adopts the variance as risk measure, thus resulting in a quadratic programming problem. Actually, solvability of the problem is of crucial importance for financial applications dealing with real-life features that require the introduction of binary and/or integer decision variables, such as minimum transaction lots [13], cardinality constraints [14], and transaction costs [15, 16]. Therefore, since Markowitz, a number of other portfolio optimization models have been proposed in the literature. Following Sharpe [17], many other attempts to linearize the portfolio optimization problem have been made. Though the mean absolute deviation has been earlier considered in portfolio analysis [18], only in 1990's a portfolio LP formulation that minimizes this risk measure has been proposed and analysed by Konno and Yamazaki [19], the so-called Mean-Absolute Deviation (MAD) model. Yitzhaki [20] proposes the Gini's Mean Difference (GMD) as risk measure, leading to the so-called GMD model. Young [21] suggests adopting the worst-case scenario as risk measure (the minimax approach), whereas Ogryczak [22] introduces the multiple criteria LP model that covers all the above as special aggregation techniques. Researchers found that under the assumption of returns normally distributed, the mean absolute deviation and the Gini's mean difference become proportional to the standard deviation [23]. Therefore, under some assumptions, the MAD and GMD models are equivalent to the Markowitz mean-variance model.

As a matter of fact any rational investor is more concerned with underperformances rather than over performances of a portfolio. Therefore, many scholars claim that the volatility of the rate of return over the mean should not be penalized. Markowitz [12] proposes using downside risk measures and suggests to use the semi variance as a risk measure. Some authors point out that the MAD model allows specific modeling of the downside risk [24]. Post-modern Portfolio Theory [33] was another attempt to fix problems with mean variance type analysis, this time by replacing variance with "downside risk" (the negative semi-variance). This, they believe, better represented what was meant by risk. Its introduction is paralleled by replacing the Sharpe ratio with a downside only measure called the Sortino ratio. In fact, most of the LP solvable models may be reformulated as based on some downside risk measures. Additionally, the models may be extended with some piecewise linear penalty functions to provide opportunities for more specific modeling of the downside risk (Carino et al. [25, 26]).

Recently, quantile risk measures - Value at Risk (VaR) or Shortfall, Conditional Value at Risk (CVaR) or Expected Shortfall, introduced by several authors [28, 29] have gained more popularity in various financial applications [27] and representing the shortfall/mean shortfall at a specified confidence level. VaR is a very popular measure of risk, but it has undesirable properties [30] such as lack of sub-additivity, i.e., VaR of a portfolio with two instruments may be greater than the sum of individual VaRs of these two instruments. Also, VaR is difficult to optimize when calculated using scenarios. In some cases, VaR is non-convex. An alternative measure of losses, Conditional Value-at-Risk (CVaR), which is also called Mean Excess Loss, Mean Shortfall, or Tail VaR, is a more consistent measure of risk since it is sub-additive and convex [30]. It can be optimized using linear programming (LP) and non-smooth optimization algorithms, which allow handling portfolios with very large numbers of instruments and scenarios [31].Numerical experiments indicate that the minimization of CVaR also leads to near optimal solutions in VaR terms because CVaR is always greater than or equal to VaR. Moreover, when the return-loss distribution is normal, these two measures are equivalent [32], i.e., they provide the same optimal portfolio. Moreover, the risk of a portfolio, measured by the VaR, can be larger than the sum of the stand-alone risks of its components [33]. Hence, managing risk by VaR may fail to stimulate diversification. Moreover, VaR does not take into account the severity of an incurred damage event. As a response to these deficiencies several authors propose to replace the VaR with the CVaR as a measure of risk.

The second moment of financial returns-variance, is an important measurement of the market fluctuation. Engel [51] introduced an autoregressive conditional heteroskedasticity (ARCH) process to model variance, and the ARCH model had been widely used. As more research were focused on the relation between variance and expected return, the auto-correlated property of variance was mostly believed. Bollerslev [52] developed the ARCH model to a more general one, the GARCH model, with the effect of self regressive considered.

Skewness, measured by the third moment, is found in many economic data such as stock or exchange index returns. The existence of skewness made an asymmetric conditional distribution of returns. The negative skewness indicates a higher probability of negative returns and positive skewness the opposite. Similar to skewness, what the fourth moment of returns concern, kurtosis, has also drawn attention. A large kurtosis implies a sharp peak of the distribution, i.e. the probability of being around the mean is extremely large. However, in most empirical work, the conditional skewness and kurtosis are observed not consistent within a long time period. The original GARCH model is not able to capture the dynamics of time-varying skewness and kurtosis. Harvey [53] presented a new methodology, GARCH with skewness (GARCHS) model, to estimate dynamic conditional skewness by introducing the third moment equation. And Brooks [54] set a GARCH with kurtosis (GARCHK) model to capture dynamic kurtosis via constructing a fourth moment equation. Both of them adapted a noncentral ‘t’ distribution to the error terms which brought also a large amount of calculation. Instead, Leon [55] used a Gram-Charlier series expansion of normality distribution (GCD) to model the conditional skewness and kurtosis simultaneously. The new method, GARCH with skewness and kurtosis (GARCHSK) model, made estimation more easily and a more comprehensive description of financial returns [56].

Many researchers took quite a lot of in-depth studies and developed the univariate GARCH models and multivariate GARCH models. For instance, the Exponential GARCH model (EGARCH model) was pointed out by Nelson [58]. He pointed out that the volatilities aroused by negative news are larger than that by same level positive news. That is to say it is an asymmetry phenomenon. In order to solve this problem, he introduced a dummy variable in the conditional variance part. Then it can reflect different volatilities when the random error takes negative or positive values. Also the Threshold GARCH (TGARCH model) was mentioned by Zakoian and then by Glosten, Jaganathan and Runkle is an asymmetry GARCH models [57].

All the models above are the basis for the multivariate GARCH models like Constant Conditional Correlation (CCC) GARCH model by Bollerslev [59] and is later extended by Jeantheau. Then Engle[60] introduced a Dynamic Conditional Correlation (DCC) GARCH model with which the conditional correlation is not a constant term any more.

Coming to Evolutionary Algorithms, the idea of using techniques based on the emulation of the mechanism of natural selection to solve problems can be traced as long back as the 1930s [4]. Since 1930s, this research field did not see major developments for almost three decades. However, in 1960s three studies [5, 6, 7] set the foundations of what is nowadays denominated ‘‘evolutionary algorithms’’ (EAs). Over the past years researchers developed several approaches for the solution of multi-objective optimization problems with the use of EAs. The first implementation of a multi-objective evolutionary algorithm (MOEA) dates back to the mid-1980s [8, 9]. Since then, a considerable amount of research has been done in this area, now known as evolutionary multi-objective optimization.

CHAPTER 3

THEORITICAL STUDY

(1)Risk Concepts:

JP Morgan [43] defines risk as the degree of uncertainty of future net returns. In the context of finance and insurance there are three main risk categories to be considered, namely, credit risk; market risk and operational risk. Credit risk is the risk of loss of financial capital such as loan repayments due to the failure of a borrower to meet contractual obligations, commonly known as the default. Market risk is the risk of losses in financial positions due to adverse movements in market prices affected by factors such as economic recession, political unrest and events that have substantial impact on the overall performance of the financial markets. This type of risk is omnipresent in various financial sectors and cannot be eliminated, it is however, possible to hedge against market risk by diversification. Operational risk refers to the risk of operational failures due to inadequate internal processes, systems and human errors.

Although insurance companies are certainly exposed to insurance risk that arises from a policyholder's mortality and morbidity status, the most critical risk associated with asset allocation and portfolio management is market risk. Notably, comprehensive market risk management involves techniques such as stress testing, worst scenario analysis combined with careful use of statistical risk measures. In the context of portfolio optimization, the scope of this thesis covers only the statistical approach to risk measurement.

(2) Risk measures [44]**:**

Selection of appropriate risk measures is central to optimization problem. A good risk measure should therefore have a list of desirable properties. This leads to the concept of so-called coherent risk measures presented by Artzner et al. [45]. Let $\rho(\cdot)$ be the risk measure and let R_1 and R_2 be two assets in a portfolio. A risk measure is coherent if it satisfies the following axioms

Axiom T - Translation invariance

$$\rho(R_1 + l) = \rho(R_1) - l \tag{2.1}$$

Axiom S - Subaddivity $$\rho(R_1 + R_2) \le \rho(R_1) + \rho(R_2) \tag{2.2}$$

Axiom PH - Positive homogeneity

$$\rho(l*R_1) = l*\rho(R_1) \tag{2.3}$$

Axiom M – Monotonicity

$$\rho(R_1) \leq \rho(R_2),\ R_2 \leq R_1 \tag{2.4}$$

Axiom T states that adding a quantity l to an asset reduces the risk by the equivalent amount. Axiom S reflects the idea that risk can be reduced by diversification. The total risk faced by a portfolio is less than or equal to the sum of the risks of its individual assets. Axiom PH says that if one increases the amount invested in one asset, one increases the risk with the same factor. Axiom M means that if the value of R_1 in general is larger than that of R_2, then the risk of R_1 is less than or equal to that of R_2. Note that Axiom S and Axiom PH together ensures the convexity of a risk measure, which implies that if there exists a local minimum, it is also the global minimum. Convexity and subadditivity are important properties of risk measures used for solving optimization problems. In particular, convex and continuously differentiable functions are easy to minimize numerically [35].

(a) Value at Risk (VaR):

VaR is a widely used risk measure of the risk of loss on a specific portfolio of financial exposures. For a given portfolio, time horizon, and probability *p*, the *p* VaR is defined as a threshold loss value, such that the probability that the loss on the portfolio over the given time horizon exceeds this value is *p* [34].

The loss function is the negative of the return of the portfolio, where portfolio return is given as the sum of returns on individual instruments $\boldsymbol{r}$ scaled by weights $\boldsymbol{w}$

$$f_L(w, r) = -(w_1 r_1 + \ldots + w_n r_n) = -w^T r \tag{2.5}$$

We define then the probability of the loss $f_L(\boldsymbol{w}, \boldsymbol{r})$ not exceeding $\boldsymbol{l}$ as

$$\Phi(w, l) = \int_{f_L(w,r) \leq l} p(r)dr \tag{2.6}$$

where $p(r)$ is the joint density function of the random returns and $\Phi(w, l)$ is the cumulative distribution loss function associated with $\boldsymbol{w}$ and is continuous and non-decreasing with respect to $\boldsymbol{l}$.

Value at Risk (VaR) with respect to the portfolio weights ***w*** for a given confidence level $\boldsymbol{\alpha} \in (0, 1)$ is given by the smallest ***l*** such that the probability of the loss $f_L(\boldsymbol{w},\boldsymbol{r})$ exceeding ***l*** is at most $1 - \alpha$

$$\mathrm{VaR}_{\alpha}(w) = \min\{l : \Phi(r, l) \geq \alpha\} \tag{2.7}$$

(b) Conditional Value at Risk (CVaR) / Expected shortfall / Expected Tail Loss:

It is an alternative to value at risk that is more sensitive to the shape of the loss distribution in the tail of the distribution. The "expected shortfall at α% level" is the expected return on the portfolio in the worst α% of the cases [35].

CVaR associated with the portfolio weights ***w*** for a given confidence level $\boldsymbol{\alpha} \in [0, 1]$ is defined as

$$\mathrm{CVaR}_{\alpha}(w) = \frac{1}{1-\propto}\int_{f_L(w,r)\geq \mathrm{VaR}_{\propto}(w)} f_L(w,r)p(r)dr \tag{2.8}$$

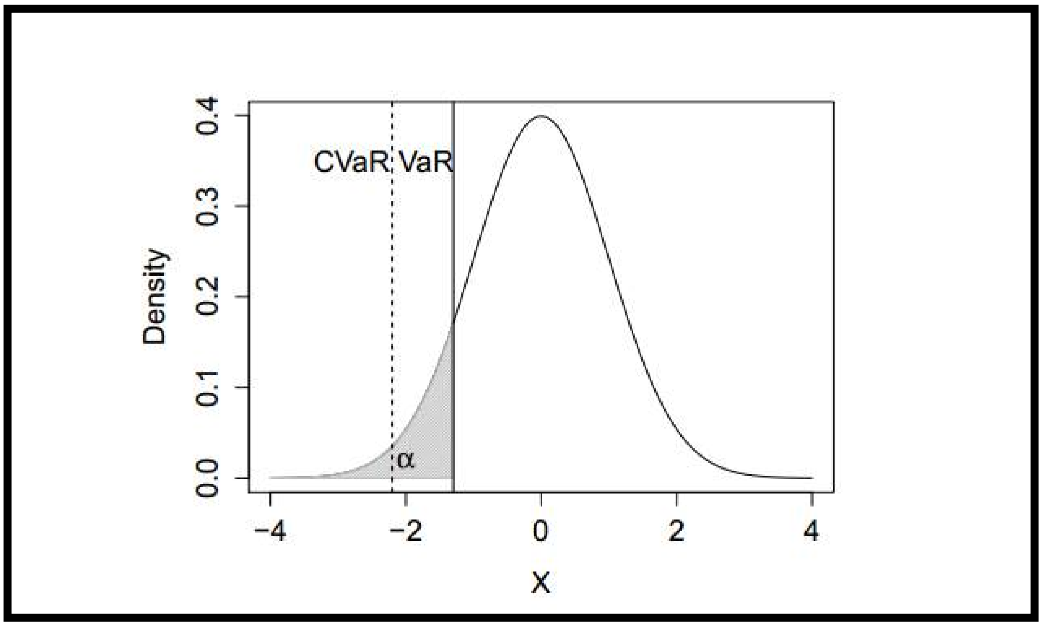


Fig. 1: Graphical representation of VaR and CVaR at confidence level α

(c)Skewness:

skewness is a measure of the asymmetry of the probability distribution of a real-valued random variable about its mean. The skewness value can be positive or negative, or even undefined.

1. negative skew: The left tail is longer; the mass of the distribution is concentrated on the right of the figure. The distribution is said to be left-skewed, left-tailed, or skewed to the left [36].

2. positive skew: The right tail is longer; the mass of the distribution is concentrated on the left of the figure. The distribution is said to be right-skewed, right-tailed, or skewed to the right [36].

Skewness is the third standardized moment. It is denoted γ_1 and defined as

$$\gamma_1 = \mathrm{E}\left[\left(\frac{X-\mu}{\sigma}\right)^3\right] = \frac{\mu_3}{\sigma^3} = \frac{\mathrm{E}\left[(X-\mu)^3\right]}{\left(\mathrm{E}\left[(X-\mu)^2\right]\right)^{3/2}} = \frac{\kappa_3}{\kappa_2^{3/2}}, \tag{2.9}$$

For univariate data $Y_1, Y_2, \ldots, Y_N$, the formula for skewness is:

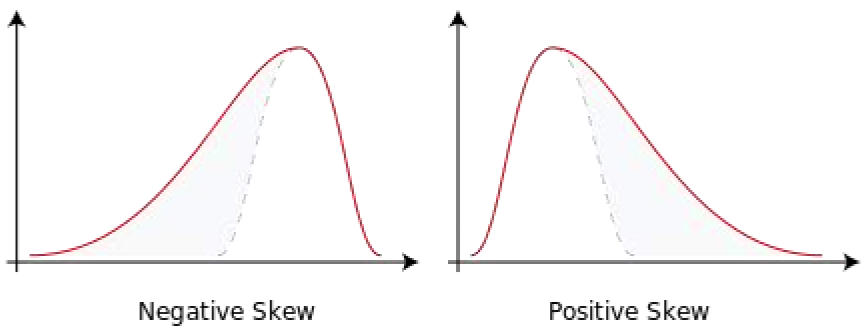


Fig.2 : Graphical representation of positive and negative skewness

$$\gamma 1 = \frac{\sum_{i=1}^{N}(\mathrm{Yi}-\mu)^3/\mathrm{N}}{\sigma^3} \tag{2.10}$$

where μ is the mean, μ_3 is the third moment about the mean, σ is the standard deviation, and N is the number of data points [37].

(d)Kurtosis:

kurtosis is a measure of the thickness of tails of the probability distribution of a real valued random variable [38]. The kurtosis of any univariate normal distribution is 3. Distribution with kurtosis less than 3 are said to be platykurtic. Distributions with kurtosis greater than 3 are said to be leptokurtic. It is also common practice to use an adjusted version of Pearson's kurtosis, the excess kurtosis, which is the kurtosis minus 3, to provide the comparison to the normal distribution.

The kurtosis is the fourth standardized moment, defined as

$$\mathrm{Kurt}[X] = \frac{\mu_4}{\sigma^4} = \frac{\mathrm{E}[(X-\mu)^4]}{(\mathrm{E}[(X-\mu)^2])^2}, \tag{2.11}$$

where μ_4 is the fourth moment about the mean and σ is the standard deviation [37].

Time-varying Skewness and Kurtosis (estimated from GARCH-SK):

Leon et el. developed a GARCH type model assuming a Gram-Charlier series expansion of the normal density function for the error term. The GARCHSK model is expressed as following:

$$\begin{aligned} r_t &= \alpha r_{t-1} + \varepsilon_t \\ h_t &= \beta_0 + \beta_1 \varepsilon_{t-1}^2 + \beta_2 h_{t-1} \\ s_t &= \gamma_0 + \gamma_1 \eta_{t-1}^3 + \gamma_2 s_{t-1} \\ k_t &= \delta_0 + \delta_1 \eta_{t-1}^4 + \delta_2 k_{t-1} \end{aligned} \tag{2.12}$$

where h_t is the conditional variance of r_t, s_t is the conditional skewness of η_t,

k_t is the conditional kurtosis of η_t

$$\eta_t = h_t^{-(1/2)} \varepsilon_t \tag{2.13}$$

η_t follows a conditional distribution of Gram-Charlier series expansion of normal density function.
Therefore the conditional distribution of η_t can be expressed as

$$\begin{aligned} f(\eta_t \mid I_{t-1}) &= \phi(\eta_t)\psi(\eta_t)^2 / \Gamma_t, \\ \psi(\eta_t) &= 1 + \frac{s_t}{6}(\eta_t^3 - 3\eta_t) + \frac{k_t - 3}{24}(\eta_t^4 - 6\eta_t + 3), \\ \Gamma_t &= 1 + \frac{s_t^2}{6} + \frac{(k_t - 3)^2}{24} \end{aligned} \tag{2.14}$$

(e) Credit rating:

A credit rating is an evaluation of the credit worthiness of a debtor, especially a business (company) or a government, but not individual consumers. The evaluation is made by a credit rating agency of the debtor's ability to pay back the debt and the likelihood of default [39].

Credit ratings are determined by credit ratings agencies. The credit rating represents the credit rating agency's evaluation of qualitative and quantitative information for a company or government; including non-public information obtained by the credit rating agencies' analysts.

A poor credit rating indicates a credit rating agency's opinion that the company has a high risk of defaulting, based on the agency's analysis of the entity's history and analysis of long term economic prospects.

(f) ARIMA/GARCH MODEL FOR CONDITIONAL VARIANCE:

The data is checked for serial autocorrelations by studying the ACF and PACF plots of the data. Augmented Dickey Fuller Test (ADF) is used to identify data stationarity, the number of autoregressive parameters, p, the number or integrations, d, the number of moving average parameter, q. Conditional mean model is estimated without considering the heteroskedasticity of the residuals. The best ARIMA model for the data is selected using Akaike Information Criteria (AIC). Model checking is done, by studying the ACF and PACF plots of the residuals. The squared residuals are then tested for conditional heteroskedasticity and conditional variance model is estimated.

Conditional Mean:

To estimate the conditional mean autoregressive integrated moving (ARIMA) model will be estimated. Autoregressive moving average process (ARMA) models are defined by the iteration between an autoregressive process (AR), and moving average process (MA). Equation below presents ARMA (p,q) model with p autoregressive terms and q moving average terms.

$$y_t = \mu + \varphi_1 y_{t-1} + \ldots\ldots + \varphi_p y_{t-p} - \theta_1 \varepsilon_{t-1} - \ldots\ldots - \theta_q \varepsilon_{t-q} + \varepsilon \tag{2.15}$$

y_t is explained by its past values and errors. μ, φ_p and θ_q are the estimated parameters and $\varepsilon_{t-q} \sim N(0,1)$. If Y_t is not stationary, we transform the data using differences to make it stationary, especially in the presence of unit roots. Equation below shows the ARIMA (p,d,q) model with p autoregressive terms, d integrated orders and q moving average terms.

$$\Delta^d y_t = \mu + \varphi_1 \Delta^d y_{t-1} + \ldots\ldots. + \varphi_p \Delta^d y_{t-p} - \theta_1 \varepsilon_{t-1} - \ldots\ldots - \theta_q \varepsilon_{t-q} + \varepsilon_t \tag{2.16}$$

Δ^d is the difference operator, φ_p and θ_q are the estimated parameters and $\varepsilon_{t-q} \sim N(0,1)$

Conditional Variance:

Lagrange Multiplier Test is applied to test for conditional heteroskedasticity effects in the residuals from the mean model. The Equation below gives the specification of a GARCH(p,q) model

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon^2_{t-i} + \sum_{i=1}^{p} \beta_i \sigma_{t-i}^2 \tag{2.17}$$

Akaike Information Criterion (AIC) provides a way to check and identify the model. It can be calculated by the formula:

AIC = N*log(SS/N) + 2(p + q + 1)*N/(N – p – q – 2), if no constant term in model

AIC = N*log(SS/N) + 2(p + q + 2)*N/(N – p – q – 3), if constant term in model

Where, N: the number of items after differencing (N = n – d), SS : sum of squares of differences, p & q : the order of autoregressive and moving average model, respectively. According to this method, the model with lowest AIC will be selected.

Diagnostic checking/Residual Analysis: The procedure includes observing residual plot and its ACF & PACF diagram, and check Ljung-Box result. If ACF & PACF of the model residuals show no significant lags, the selected model is appropriate.

(g) DCC GARCH model: (Dynamic Correlation Model) [61]

The GARCH-DCC involves two steps. The first step accounts for the conditional heteroskedasticity. It consists in estimating, for each one of the *n* series of returns r_{it}, its conditional volatility σ_{it} using a GARCH model. Let D_t be a diagonal matrix with these conditional volatilities, i.e. $D^{i,i}{}_t = \sigma_{it}$ and, if $i \neq j$, $D^{i,j}{}_t = 0$. Then the standardized residuals are:

$$\upsilon_t = D^{-1}{}_t (r_t - \mu) \tag{2.18}$$

Now, define the matrix:

$$R = (1/T) * \sum_{t=1}^{T} \upsilon_t \upsilon_t ' \tag{2.19}$$

This is the Bollerslev's Constant Conditional Correlation (CCC) Estimator.

The second step consists in generalizing Bollerslev's CCC to capture dynamics in the correlation, hence the name Dynamic Conditional Correlation (DCC). The DCC correlations are:

$$Q_t = R + \sum_{i=1}^{p} \alpha_i (\upsilon_{t-i} \upsilon'_{t-i} - R) + \sum_{j=1}^{q} \beta_j (Q_{t-j} - R) \tag{2.20}$$

So, $Q^{i,j}{}_t$ is the correlation between r_{it} and r_{jt} at time t.

(3) Multi-objective optimization:

Multi-objective formulations are realistic models for many complex engineering optimization problems. In many real-life problems, objectives under consideration conflict with each other, and optimizing a particular solution with respect to a single objective can result in unacceptable results with respect to the other objectives [42]. A reasonable solution to a multi-objective problem is to investigate a set of solutions, each of which satisfies the objectives at an acceptable level without being dominated by any other solution. The ultimate goal of a multi-objective optimization algorithm is to identify solutions in the Pareto optimal set. However, identifying the entire Pareto optimal set is practically impossible due to its size. Therefore, a practical approach to multi-objective optimization is to investigate a set of solutions (the best-known Pareto set) that represent the Pareto optimal set as well as possible. With these concerns in mind, a multi-objective optimization approach should achieve the following three conflicting goals[36]:

1. The best-known Pareto front should be as close as possible to the true Pareto front. Ideally, the best-known Pareto set should be a subset of the Pareto optimal set.

2. Solutions in the best-known Pareto set should be uniformly distributed and diverse over of the Pareto front in order to provide the decision-maker a true picture of trade-offs.

3. The best-known Pareto front should capture the whole spectrum of the Pareto front. This requires investigating solutions at the extreme ends of the objective function space.

For a given computational time limit, the first goal is best served by intensifying the search on a particular region of the Pareto front. On the contrary, the second goal demands the search effort to be uniformly distributed over the Pareto front. The third goal aims at extending the Pareto front at both ends, exploring new extreme solutions. One of the common approaches used in multi-objective GA to attain these three conflicting goals while solving a multi-objective optimization problem is Non dominated sorting genetic algorithm (NSGA-II).

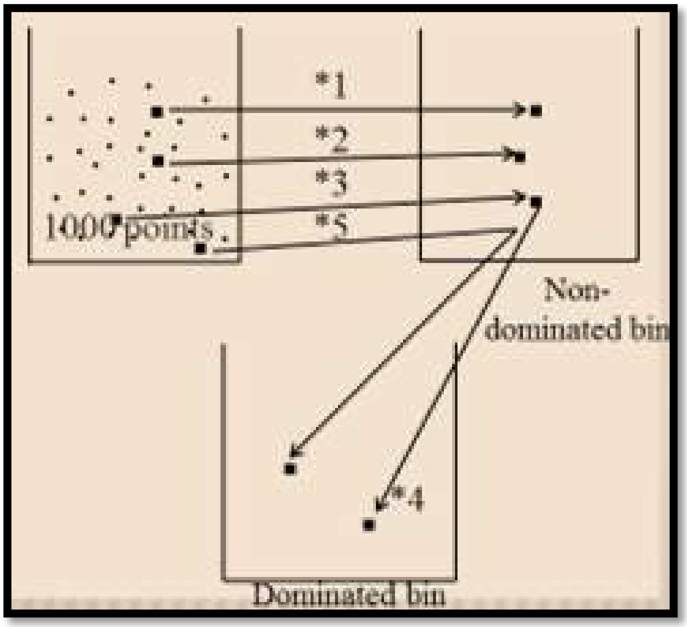
(4) Evolutionary algorithms for optimization:

EAs are population based stochastic optimization heuristics inspired by Darwin's Evolution Theory. An EA searches through a solution space in parallel by evaluating a set (population) of possible solutions (individuals). An EA starts with a random initial population. Then the 'fitness' of each individual is determined by evaluating the objective function. After the best individuals are selected, new individuals for the next generation are created. The new individuals are generated by altering the individuals through random mutation and by mixing the decision variables of multiple parents through crossover. Then the generational cycle repeats until a breaking criterion is fulfilled.

EAs are less susceptible to the shape or continuity of the Pareto front (they can easily deal with discontinuous and concave Pareto fronts), whereas these two issues are known problems with mathematical programming techniques [40, 41]. During multi-objective optimization two goals are to be reached. On the one hand the solutions should be as close to the global Pareto-optimal front as possible and on the other hand the solutions should also cover the whole Pareto-front. The first goal is often achieved through elitism by replacing random individuals with individuals on the Pareto front. The second goal can be achieved by punishing individuals that are too close together (Fitness Sharing).

According to Coello, the traditional evolutionary algorithms cannot efficiently deal with multi-objective optimization problems for two reasons: (1) Due to stochastic noise, evolutionary algorithms tend to converge to a single solution if run for a sufficiently large number of iterations. Thus, it is necessary to block the selection mechanism so that different solutions (non-dominated) are preserved in the population of an evolutionary algorithm. (2) All non-dominated solutions should be sampled at the same rate during the selection stage, since all non-dominated solutions are equally good among themselves.

(5) Non-dominated sorting:



*1: Initially selected as non-dominated point

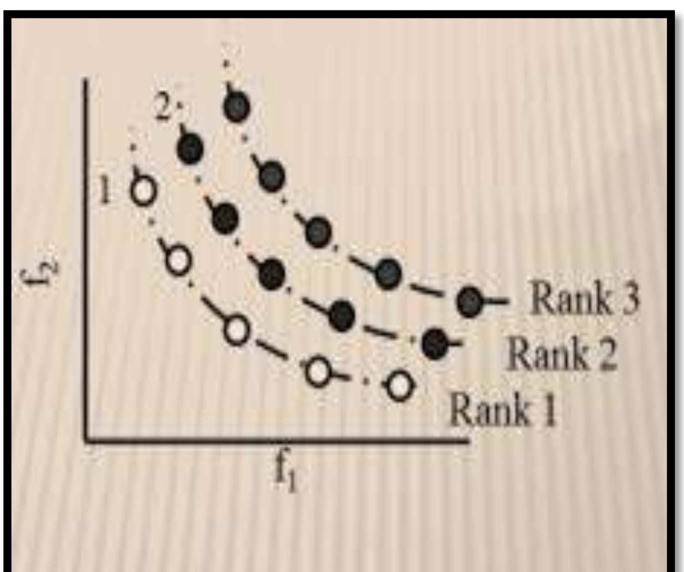
*2: Better than all non-dominated point

*3: Better than some points of non-dominated bin but worse than some other points

*4: Worse points are put into dominated bin

*5: Worse than all non-dominated points Let us consider an optimization problem, where two objectives f_1 and f_2 are to be minimized.

Fig.3: Non dominated sorting mechanism [50]



Point 1 is non-dominated solution and point 2 is a dominated one. All non-dominated points are assigned Rank 1. Average fitness of Rank 1 will be better than that of Rank 2.

The population of solutions is then modified using different operators**: reproduction, crossover, mutation,** and others

Fig.4: Ranks formed after non dominated sorting [50]

(6)Reproduction: It forms a mating pool consisting of good solutions probabilistically various reproduction schemes like proportionate selection (such as Roulette-Wheel selection), ranking selection, tournament selection, and others.

Here, proportionate selection method is used. This is implemented with the help of a Roulette-Wheel.

Probability of getting selected in a mating pool is directly proportional to fitness, $$P = \frac{f_i}{\sum_{i=1}^{N} f_i} \quad ..(2.21)$$

Reproduction will select more number of Rank 1 solutions compared to Rank 2 solutions.

(7) Crossover:

There is an exchange of properties between the parents and as a result of which, new children solutions are created. Various crossover schemes: **single-point, two-point, multi-point, uniform crossovers**

Here, Uniform crossover is employed where at each bit position of the parent strings, we toss a coin (with a probability of 0.5 for appearing head) to decide whether there will be interchanging of the bits. If the head appears there will be a swapping of bits among the parent strings.

(8) **Mutation:**

In biology, it means a sudden change of parameter. It may help to overcome the local minima problems. 1 is converted into 0 and vice versa. Mutation probability (P_m) is generally kept to a low value.

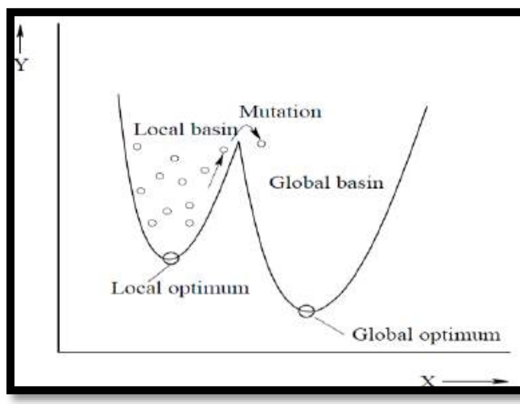


Fig.5: Mutation operating mechanism [50]

(9) Non-Dominated Sorting GA-II (NSGA-II)

Genetic algorithms (GAs) are search and optimization tools, which work differently compared to classical search and optimization methods. Because of their broad applicability, ease of use, and global perspective, GAs have been increasingly applied to various search and optimization problems in the recent past.

Non dominated sorting genetic algorithm is a type of genetic algorithm where the concept of non dominated sorting is applied and connected to GA for further optimization. The steps followed in NSGA-II are discussed below.

a) Sort the population into ranks, say 1, 2 and so on using a ranking procedure. Average fitness of Rank 1 solutions is better than that of Rank 2 solutions & Average fitness of Rank 2 solutions is better than that of Rank 3 solutions

b) Carry out proportionate selection. Thus, there will be more copies of Rank 1 solutions in the mating pool. So, the selection pressure is ensured.

c) Crowding distance of i-th solution lying on the front is the side length of the cuboid. Crowding distance will be less for more crowded solutions. Solutions having more crowding distance values will be preferred in reproduction scheme to form the mating pool

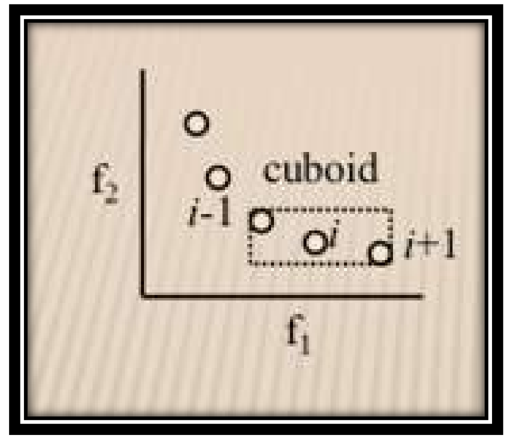


Fig.6: Crowding distance [50]

CHAPTER 4

MULTI-OBJECTIVE PORTFOLIO OPTIMIZATION: MATHEMATICAL FORMULATION

The following figure shows the six more specific objective functions that replaced classical criteria of Markowitz model - risk and return.

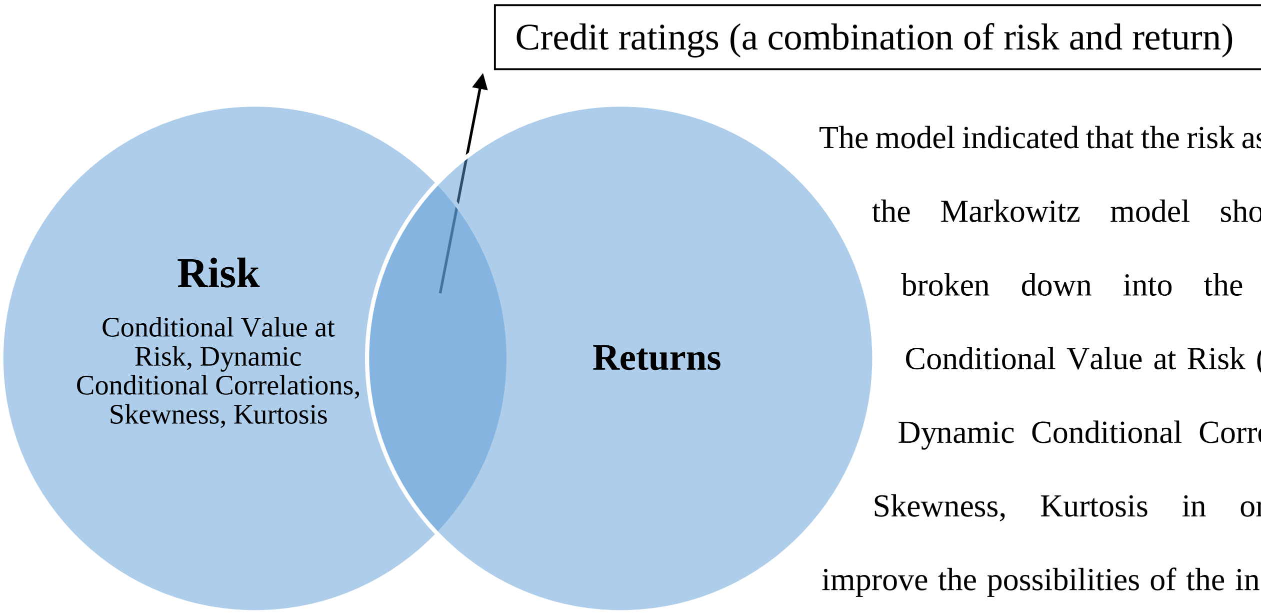


Fig.7. Proxies of Risk and Return

The model indicated that the risk as used in the Markowitz model should be broken down into the criteria Conditional Value at Risk (CVaR), Dynamic Conditional Correlations, Skewness, Kurtosis in order to improve the possibilities of the individual investor to articulate subjective preferences. The fifth objective, the Standard and Poor's star ranking, describes to what extent an investment fund follows a specific market index and is applied particularly in the case that a portfolio consists exclusively of investment funds. The sixth attribute, the fuzziness induced returns, is used as a measure of the return of the portfolio [47]. The resulting model with the six objective functions f1 to f2 as specified above can be formulated as follows:

$$\text{Minimize } \sum_{i=1}^{2} a_i f_i(X), \tag{4.1}$$

where $X = (x_1, x_2, \ldots\ldots, x_5)^T$, x_i are the asset weights or design variables and a_i are the weights of

objective functions lying between 0.0 and 1.0; $\sum_{i=1}^{2} a_i = 1.0$ (4.2)

Objective-1: **Returns** $f_1(X) = r_p = x_1 r_1 + x_2 r_2 + x_3 r_3 + x_4 r_4 + x_5 r_5$ (4.3)

Objective-2: **CVaR**

$$f_2(X) = x_1 f_{21} + x_2 f_{22} + x_3 f_{23} + x_4 f_{24} + x_5 f_{25} \tag{4.4}$$

where, $f_2(X)$ is CVaR associated with the portfolio weights $\boldsymbol{X}$ for a given confidence level $\boldsymbol{\alpha} \in [0, 1]$, which is defined as

$$CVaR_{\alpha}(X) = \frac{1}{1-\alpha} \int_{f_L(X,r) \geq VaR_{\alpha}(X)} f_L(X,r)p(r)dr \tag{4.5}$$

Optimizing **X** with respect to a single objective often results in unacceptable results with respect to the other objectives. Therefore, a perfect multi-objective solution that simultaneously optimizes each objective function is almost impossible. A reasonable solution to a multi-objective problem is to investigate a set of solutions, each of which satisfies the objectives at an acceptable level without being dominated by any other solution [48].

If all objective functions are for minimization, a feasible solution **X** is said to dominate another feasible solution **Y** (**x**>**y**), if and only if, $\mathbf{z_i(x) \leq z_i(y)}$ for least one objective function i. A solution is said to be Pareto optimal if it is not dominated by any other solution in the solution space. A Pareto optimal solution cannot be improved with respect to any objective without worsening at least one other objective. The set of all feasible non-dominated solutions in **x** is referred to as the Pareto optimal set, and for a given Pareto optimal set, the corresponding objective function values in the objective space are called the Pareto front [49].

Similarly, each of the other objective functions viz., Dynamic Conditional Correlations (DCC) i.e., time-varying correlations, Generalized Auto-regressive Conditional Heteroskedasticity adjusted with Skewness and Kurtosis (GARCH-SK) i.e., time-varying variance, Skewness and Kurtosis are optimized with respect to portfolio returns to estimate the optimal weights for each of the assets in the portfolio. The returns for each of the assets are forecasted using ARIMA/GARCH. Then the portfolio returns are determined using these weights for each of the objective functions and the ones with the best risk-adjusted returns are determined.

CHAPTER 5

ANALYSIS, RESULTS AND DISCUSSION

(a) Data and descriptive analysis:

The empirical studies and analysis employ a sample data extending from August 20, 2014 to August 20, 2015, in total 245 trading days. It consists of daily log-returns of the following companies from the National Stock Exchange (NSE, India) data base.

- ITC Ltd, SUN PHARMA, HDFC BANK, INFOSYS, TATA MOTORS

The companies are selected on the basis of their performance in their respective sectors over a period of one year and their growth prospects in the next five years. A thorough market research has also been done about the firm specific information.

The procedure of data analysis, simulation, estimation and optimization is carried out using MATLAB software in addition to RATS & R statistical programming software with functions from packages tseries, forecast and fPortfolio, and Microsoft Excel. See Appendix for a summary of the Matlab, RATS and R codes

Since the data set consists of financial time series, the returns series are expected to display volatility clustering, periods of high volatility and low volatility. But, we can trust the data only if it a stationary series, i.e. mean and variance are time invariant. Only then, can we determine the values of the objective functions and the analysis done be valid. Below are the results of stationarity check done by Augmented Dickey fuller test

```
> data.ts=ts(data,frequency=252)
> log.ret.TATA_MOTORS=diff(log(data.ts))
> adf.test(log.ret.TATA_MOTORS,alternative=("stationary"))

 Augmented Dickey-Fuller Test

data:  log.ret.TATA_MOTORS
Dickey-Fuller = -6.5298, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

Table 1: ADF test result for TATA motors

```
> log.ret.ITC=diff(log(data.ts))
> adf.test(log.ret.ITC,alternative=("stationary"))

 Augmented Dickey-Fuller Test

data:  log.ret.ITC
Dickey-Fuller = -5.9681, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

Table 2: ADF test result for ITC Ltd.

```
> log.ret.SUN_PHARMA=diff(log(data.ts))
> adf.test(log.ret.SUN_PHARMA,alternative=("stationary"))

 Augmented Dickey-Fuller Test

data:  log.ret.SUN_PHARMA
Dickey-Fuller = -5.7521, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

Table 3: ADF test result for Sun pharma

```
> log.ret.HDFC=diff(log(data.ts))
> adf.test(log.ret.HDFC,alternative=("stationary"))

 Augmented Dickey-Fuller Test

data:  log.ret.HDFC
Dickey-Fuller = -5.3503, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

Table 4: ADF test result for HDFC Bank

```
> log.ret.INFOSYS=diff(log(data.ts))
> adf.test(log.ret.INFOSYS,alternative=("stationary"))

 Augmented Dickey-Fuller Test

data:  log.ret.INFOSYS
Dickey-Fuller = -5.5869, Lag order = 6, p-value = 0.01
alternative hypothesis: stationary
```

Table 5: ADF test result for Infosys

From the results, it is clear that the log return series of the stock prices of all the companies are stationary.

(b) Multi-objective optimization:

Here, we test for the conflicting nature of objective functions, because only then can we obtain an optimal front. So, we plot CVaR Vs retuns.

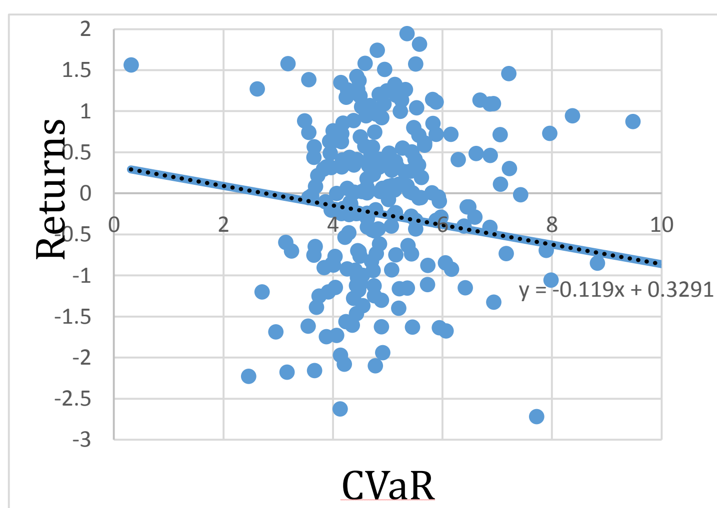


Fig.8. CVaR Vs Returns

From the figure, it is clear that the objective functions are conflicting with each other, hence we can proceed further.

Similarly, below are the graphs of DCC, Skewness, Kurtosis and GARCH against Returns.

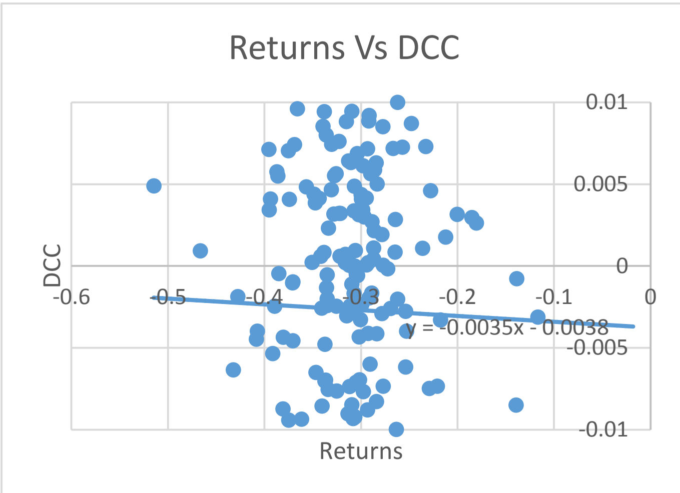


Fig.9. Returns Vs DCC

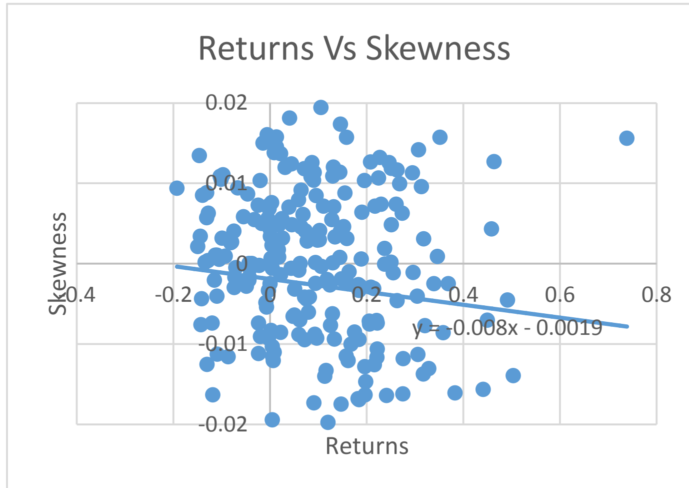


Fig.10. Returns Vs Skewness

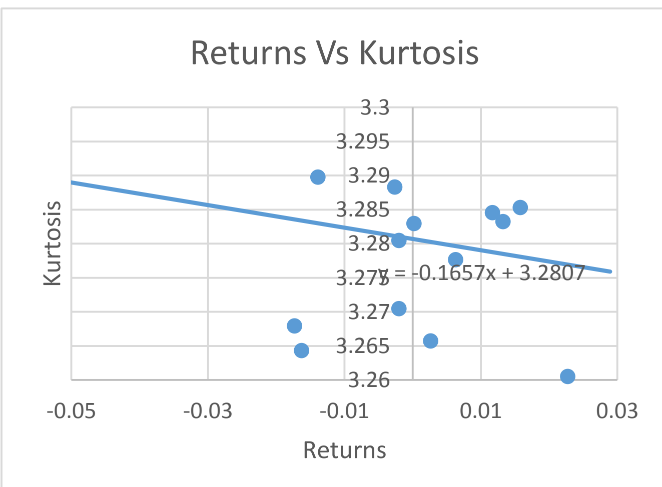


Fig.11. Returns Vs Kurtosis

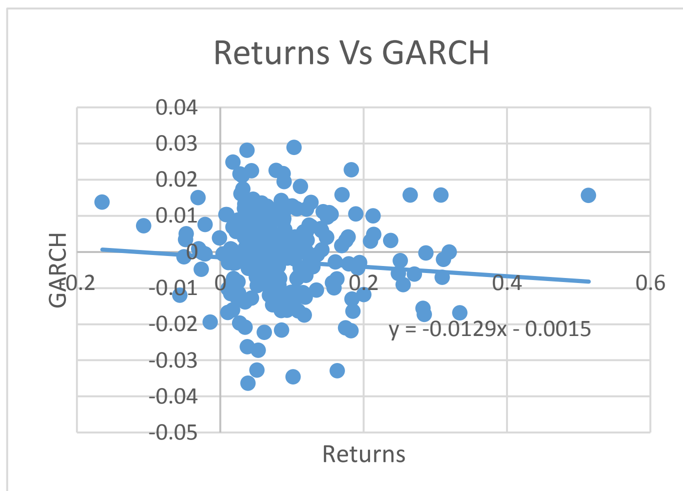


Fig.12. Returns Vs GARCH

(c) Non-dominated sorting:

Non dominated sorting was carried out as discussed in chapter 3 between the objectives, CVaR and returns. Below is a figure showing various ranks of solutions obtained. From the plot, it can be observed that, as the CVaR (risk) value increases, the returns increase. The bottom most front indicated the non dominated rank of solutions (Rank-1) and the rank increases as we move upwards. Each point represents a set of asset weights/design variables (in this case, 5). These solutions are further modified by connecting to genetic algorithm.

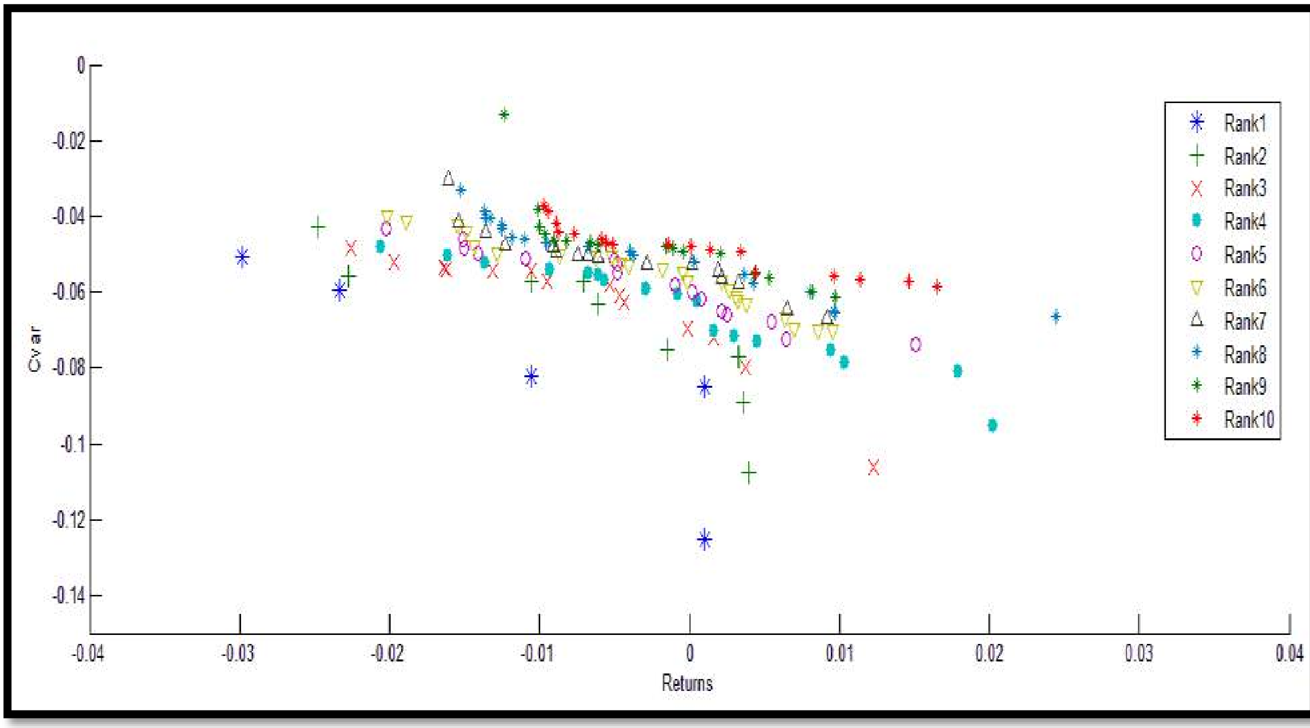


Fig.13. Ranks of solutions obtained after non-dominated sorting

(d) Non-dominated sorting genetic algorithm-II (NSGA-II):

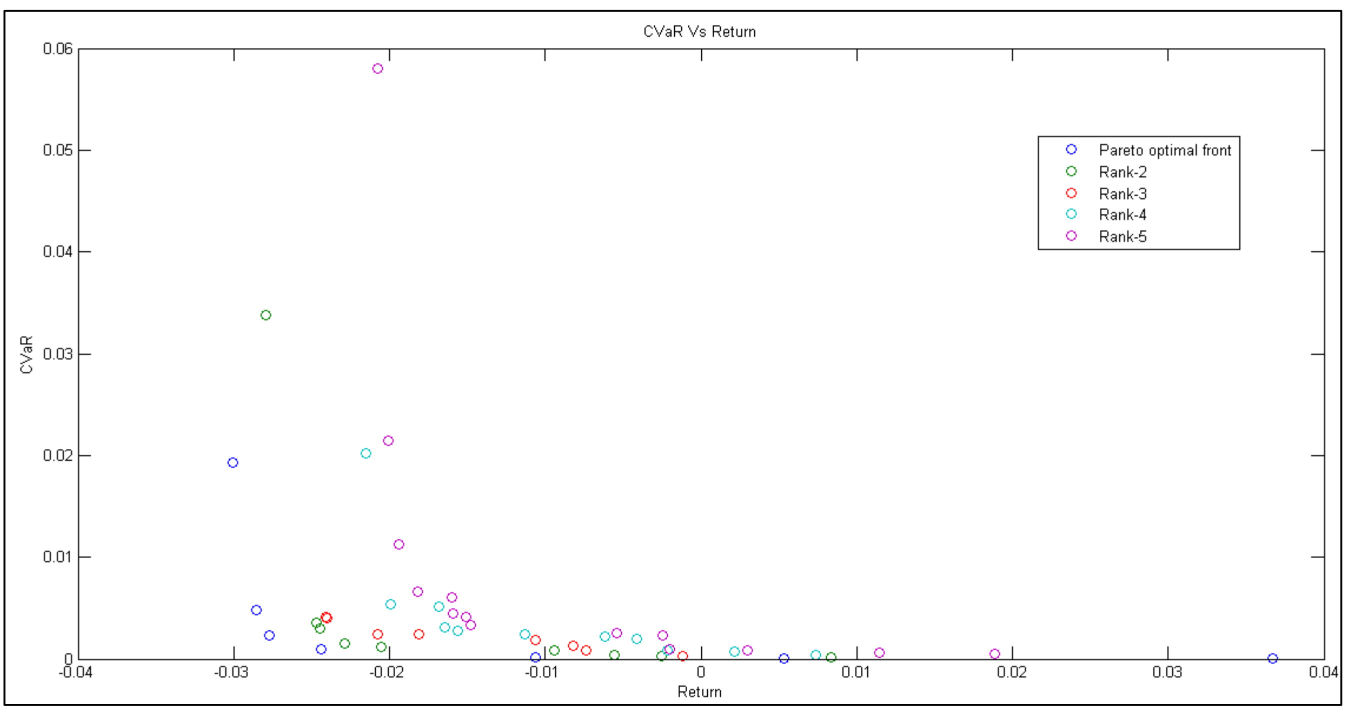


Fig.14. Pareto optimal front obtained by NSGA-II: Return Vs CVaR

From this plot also, it can be observed that, as the CVaR (risk) value decreases as the returns increase. The bottom most front is the pareto optimal front obtained by multi-objective optimization using NSGA-II. It is the locus of all optimal solutions obtained after putting different weights on the objectives. Each point represents a set of asset weights/design variables (in this case, 5)

CVaR	**-Returns**	**Obj. Value**	**HDFC**	**SUN PHARMA**	**TVS MOTORS**	**INFOSYS**	**ITC**
0.0001	**-0.0118**	**0.00595**	**0.3093**	**0.133**	**0.1802**	**0.2268**	**0.1507**
0.0008	**-0.0018**	**0.0013**	**0.0459**	**0.0454**	**0.1248**	**0.7426**	**0.0413**
0.0009	**0.0046**	**-0.00185**	**0.129**	**0.2338**	**0.2232**	**0.2477**	**0.1663**
0.0046	**0.0113**	**-0.00335**	**0.2238**	**0.14**	**0.1456**	**0.2865**	**0.2041**
0.0059	**0.0195**	**-0.0068**	**0.1088**	**0.0565**	**0.2147**	**0.36**	**0.26**
0.0101	**0.0222**	**-0.00605**	**0.223**	**0.228**	**0.1983**	**0.2389**	**0.1118**
0.0131	**0.0262**	**-0.00655**	**0.1105**	**0.3237**	**0.2114**	**0.2243**	**0.13**
0.0261	**0.0302**	**-0.00205**	**0.2089**	**0.129**	**0.259**	**0.2395**	**0.1635**

Table 7. CVaR Vs Returns

There are 8 solutions in the pareto optimal front and the one with the minimum value of overall objective function (-0.0068) yields a return of -1.95% with a CVaR of 0.0059 times portfolio value.

The asset weights of the companies HDFC, Sun pharma, TVS motors, Infosys, ITC Ltd. at this point are 0.1088, 0.0565, 0.2147, 0.36, 0.26 respectively.

Similarly, the NSGA-II is performed on the objective functions as well and the optimal weights are determined. Below are the pareto optimal plots obtained for each of these using NSGA-II.

(ii) DCC:

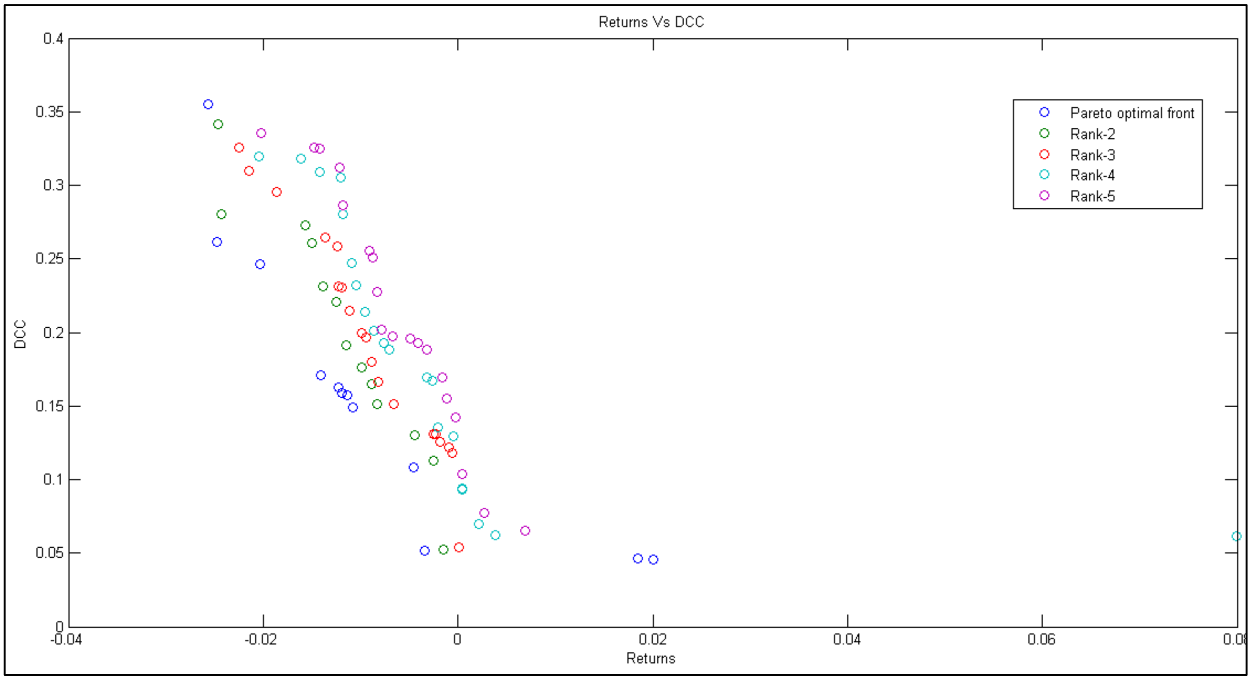


Fig.15. Pareto optimal front obtained by NSGA-II: Return Vs DCC

The optimal asset weights of the companies HDFC, Sun pharma, TVS motors, Infosys, ITC Ltd. here are

0.093917297	0.286644	0.194203	0.1265	0.298735

respectively.

(iii)Skewness:

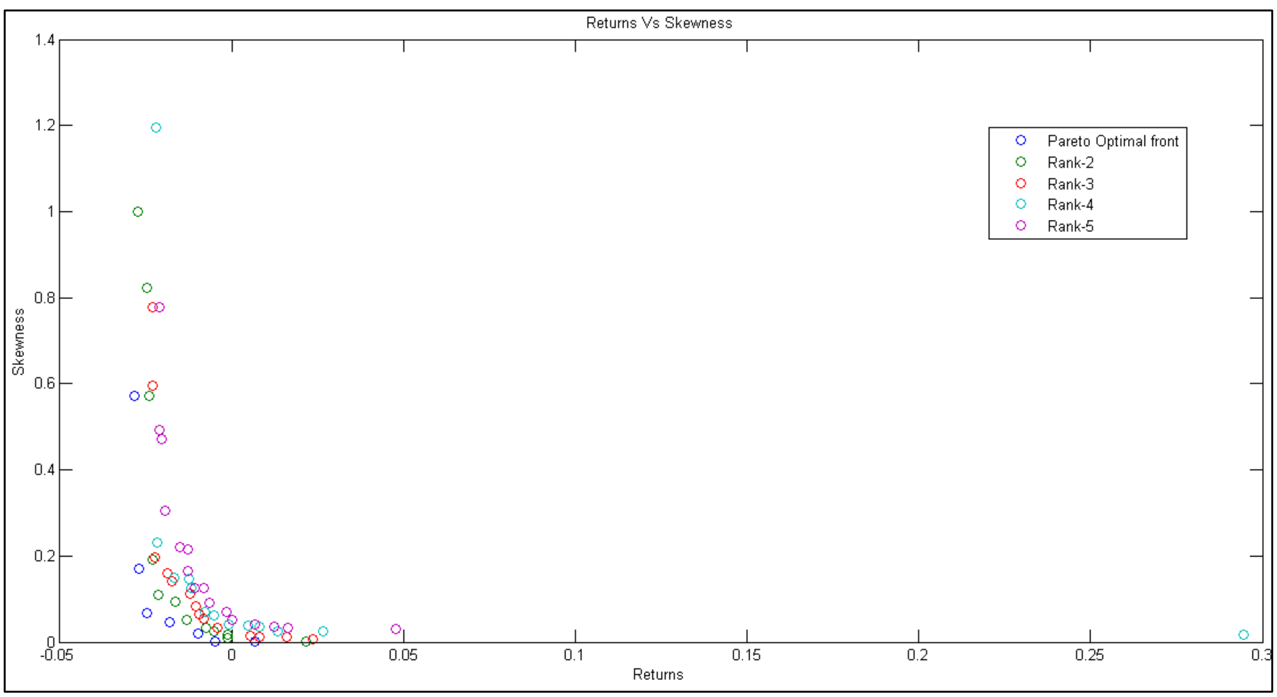


Fig.16. Pareto optimal front obtained by NSGA-II: Return Vs Skewness

The optimal asset weights of the companies HDFC, Sun pharma, TVS motors, Infosys, ITC Ltd. here are 0.072902747 0.067854 0.151002 0.392725 0.315516 respectively.

(iv) Kurtosis:

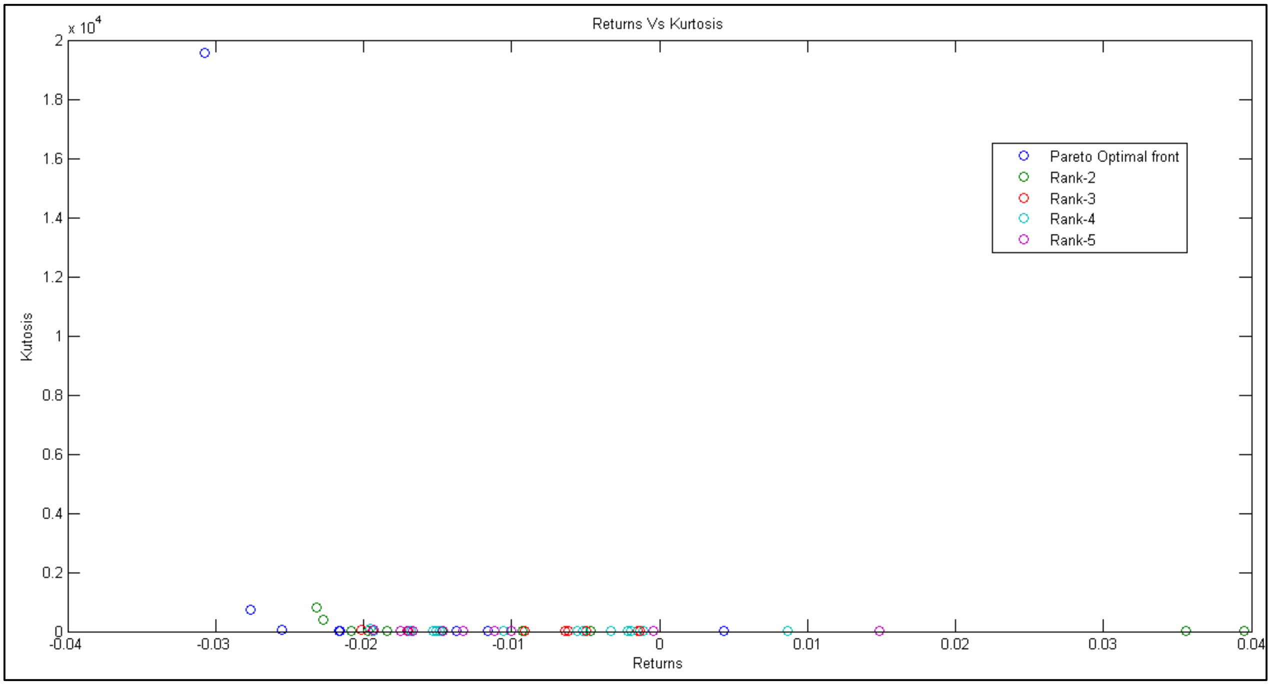


Fig.17. Pareto optimal front obtained by NSGA-II: Return Vs Kurtosis

The optimal asset weights of the companies HDFC, Sun pharma, TVS motors, Infosys, ITC Ltd. here are 0.017538135 0.076107 0.200098 0.251897 0.45436 respectively.

(v) GARCH:

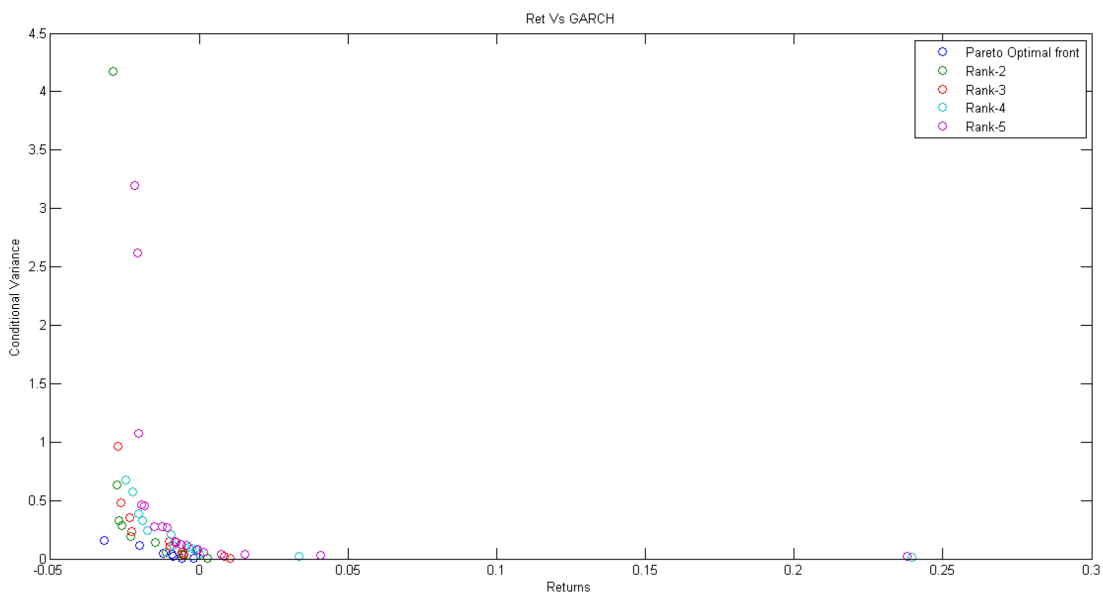


Fig.18. Pareto optimal front obtained by NSGA-II: Return Vs GARCH

The optimal asset weights of the companies HDFC, Sun pharma, TVS motors, Infosys, ITC Ltd. here are 0.227070863 0.349746 0.06713 0.178091 0.177962 respectively.

Results of Regression fit:

A linear regression equation is fitted with returns as dependent variable and DCC, GARCH, skewness, kurtosis, CVaR as independent variables. Below are the results for the fit.

Model	Variables Entered	Variables Removed	Method
1	VAR00006, VAR00005, VAR00004, VAR00002, VAR00001[b]	.	Enter

Table 8. Variables Entered/Removed[a]

a. Dependent Variable: VAR00003

b. All requested variables entered.

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.572[a]	.327	.313	.14161

Table 9. Model Summary

a. Predictors: (Constant), VAR00006, VAR00005, VAR00004, VAR00002, VAR00001

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.329	5	.466	23.233	.000[b]
	Residual	4.793	239	.020		
	Total	7.122	244			

Table 10. ANOVA[a]

a. Dependent Variable: VAR00003

b. Predictors: (Constant), VAR00006, VAR00005, VAR00004, VAR00002, VAR00001

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	3.121	.081		38.679	.000
	VAR00001	-.480	.169	-.168	-2.838	.005
	VAR00002	.409	.067	.355	6.117	.000
	VAR00004	.455	.124	.212	3.653	.000
	VAR00005	-.076	.317	-.013	-.241	.810
	VAR00006	1.339	.785	.106	1.705	.089

Table 11. Coefficients[a]

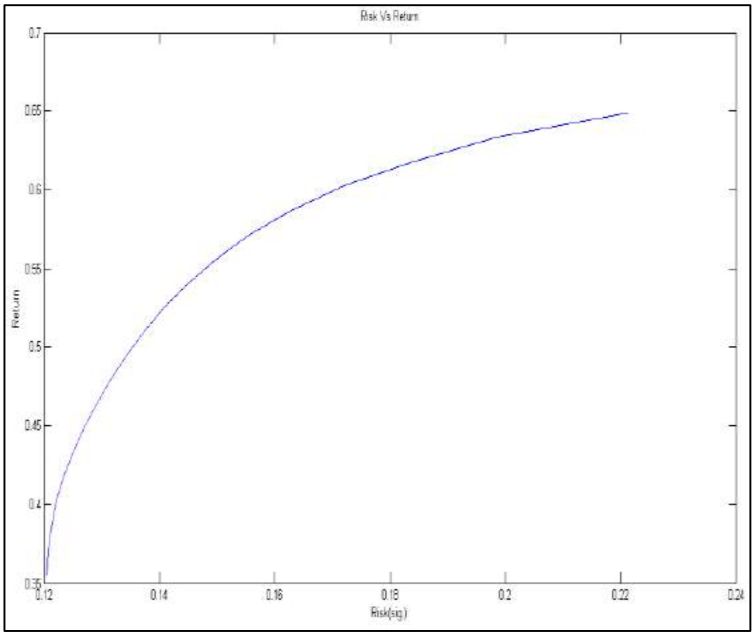
a. Dependent Variable: VAR00003

From the above results, goodness of fit is just 32.7% and only DCC, skewness, GARCH are significant.

Hence, only the above three objective functions are used to estimate the final optimal weights. The optimal asset weights of the companies HDFC, Sun pharma, TVS motors, Infosys, ITC Ltd. here are **0.228901471 0.128222 0.00198 0.470983 0.173874** respectively.

Using these weights and the forecasted returns for the next 100 days, portfolio returns are calculated.

Similarly, optimal weights for the conventional Mean variance portfolio optimization method are estimated. Below are the results for the same.



The optimal asset weights of the companies HDFC, Sun pharma, TVS motors, Infosys, ITC Ltd. here are

HDFC	SUN PHARMA	TVS MOTORS	INFOSYS	ITC
0.9466	0	0	0	0.0534

Fig.19. Efficient frontier: Return Vs Risk

Below is the plot comparing the portfolio returns obtained from the Multi-objective optimization method and the conventional minimum variance optimization technique.

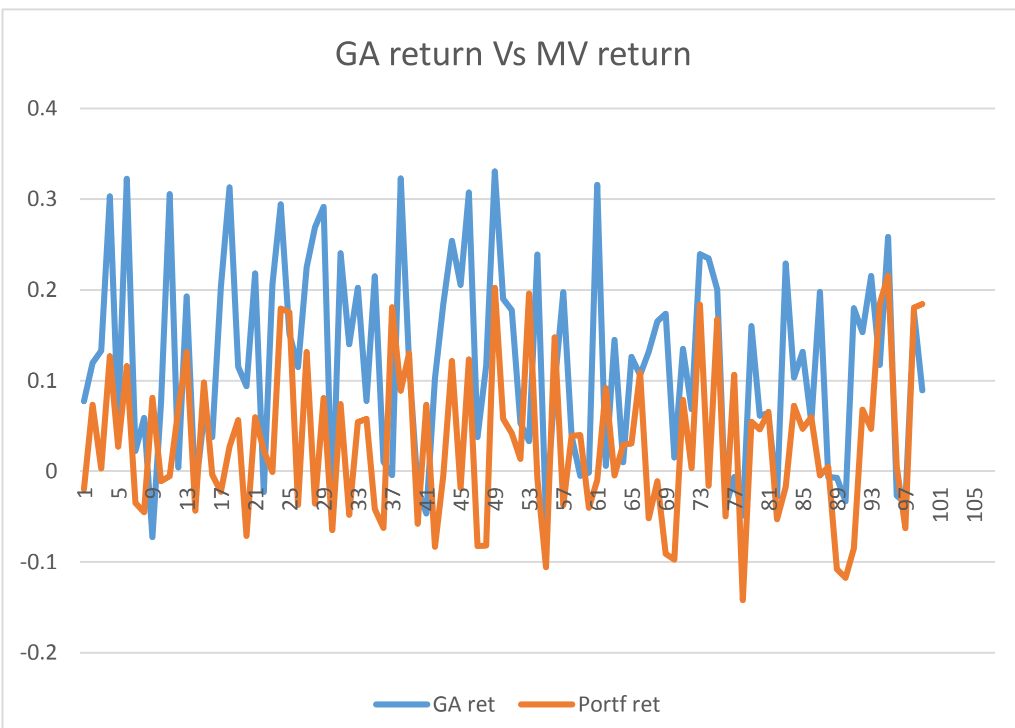


Fig. 20. GA return Vs MV return for next 100 days

CONCLUSION:

From the regression fit, only DCC, skewness and GARCH are significant. Hence, only these three objective functions are used to estimate the final optimal weights, which are obtained from NSGA-II. Using these weights and the forecasted returns for the next 100 days, portfolio returns are calculated.

Similarly, optimal weights and portfolio returns for the conventional Mean variance portfolio optimization method are estimated. The average portfolio return for multi-objective optimization technique came out to be 12.01%, where as the average portfolio return for minimum variance method came out to be 3.178%. Thus, proving that the method adopted in this thesis is better than the conventional method.

In future, to further improve the solutions, we can include transcation costs, inventory cost, liquidity and other asset pricing factors. Also, in this thesis, credit ratings were not included as they do not vary much in short period. Historical asset weights have been optimized in this study. But, as we know companies will be willing to take a certain amount of risk, random weights can be generated and optimized to maximize the specified risk adjusted portfolio returns. In addition to all these, the number of assets can be increased, assets can be chosen from various asset classes across various global markets and fuzziness can be induced in the returns.

BIBLIOGRAPHY

1. Metaxiotis,K. Liagkouras, K.(2012). Multi objective Evolutionary Algorithms for Portfolio Management: A comprehensive literature review. Expert Systems with Applications, 39, pp11685–98
2. A.I. Kibzun, E.A. Kuznetsov(2006). Analysis of criteria VaR and CVaR. Journal of Banking & Finance, 30 (2), pp. 779–796
3. L.A. Balzer(1994). Measuring investment risk. The Journal of Investing, 3 (3), pp. 47–58
4. W.D. Cannon. The wisdom of the body. Norton and Company, New York, NY (1932)
5. L.J. Fogel Artificial intelligence through simulated evolution. Forty years of evolutionary programming John Wiley & Sons, New York, NY (1999)
6. J.H. Holland Outline for a logical theory of adaptive systems' Journal of the Association for Computing Machinery, 9 (1962), pp. 297–314
7. Schwefel, H.P. (1965). Kybernetische evolution alsstrategie der experimentellenforschunginderströmungstechnik. Dipl.-Ing. thesis (in German)
8. Schaffer, J. D. (1984). Some experiments in machine learning using vector evaluated genetic algorithms. Doctoral Dissertation. Nashville, TN: Department of Electrical and Biomedical Engineering, Vanderbilt University
9. J.D. Schaffer Multiple objective optimization with vector evaluated genetic algorithms Genetic algorithms and their applications: Proceedings of the first international conference on genetic algorithms, Lawrence Erlbaum, Hillsdale, NJ (1985), pp. 93–100
10. Joe Staines (2010). Myopic and Dynamic Approaches to Portfolio Optimization. Masters thesis. UK PhD Centre in Financial Computing, University college London.
11. H.M. Markowitz. Portfolio selection. Journal of Finance, 7:77–91, 1952.
12. H.M. Markowitz. Portfolio selection: Efficient diversification of investments. John Wiley & Sons, 1959.
13. R. Mansini and M.G. Speranza. Heuristic algorithms for the portfolio selection problem with minimum transaction lots. European Journal of Operational Research, 114:219–233, 1999
14. A. Cobham. Logic, Methodology and Philosophy of Science, chapter in The intrinsic computational difficulty of functions, pages 24–30. Y. Bar-Hillel eds., Elsevier/North-Holland, 1965.
15. L. Chiodi, R. Mansini, and M.G. Speranza. Semi-absolute deviation rule for mutual funds portfolio selection. Annals of Operations Research, 124:245–265, 2003.
16. H. Konno and A. Wijayanayake. Portfolio optimization problem under concave transaction costs and minimal transaction unit constraints. Mathematical Programming, 89:233–250, 2001
17. W.F. Sharpe. A linear programming approximation for the general portfolio analysis problem. Journal of Financial and Quantitative Analysis, 6:1263–1275, 1971
18. W.F. Sharpe. Mean-absolute deviation characteristic lines for securities and portfolios. Management Science, 18:B1–B13, 1971
19. H. Konno and H. Yamazaki. Mean-absolute deviation portfolio optimization model and itsapplication to Tokyo stock market. Management Science, 37:519–531, 1991.
20. S. Yitzhaki. Stochastic dominance, mean variance, and Gini's mean difference. American Economic Review, 72:178185, 1982
21. M.R. Young. A minimax portfolio selection rule with linear programming solution. Management Science, 44:673–683, 1998.
22. W. Ogryczak. Multiple criteria linear programming model for portfolio selection. Annals of Operations Research, 97:143–162, 2000
23. W.H. Kruskal and J.M. Tanur. International Encyclopedia of Statistics. Old Tappan, 1978.
24. M.G. Speranza. Linear programming models for portfolio optimization. Finance, 14:107–123, 1993
25. D.R. Carino, D.H. Myers, and W.T. Ziemba. Concepts, technical issues and uses of the Russel-Yasuda Kasai financial planning model. Operations Research, 46:450–463, 1998.

26. W. Michalowski and W. Ogryczak. Extending the MAD portfolio optimization model toincorporate downside risk aversion. Naval Research Logistics, 48:185–200, 2001
27. R. Mansini, W. Ogryczak, and M.G. Speranza. LP solvable models for portfolio optimization: A classification and computational comparison. IMA Journal of Management Mathematics, 14:187–220, 2003
28. R.T. Rockafellar and S. Uryasev. Optimization of Conditional Value-at-Risk. Journal of Risk, 2:21–41, 2000
29. C. Acerbi and D. Tasche. On the coherence of expected shortfall. Journal of Banking &Finance, 26:1491–1507, 2002
30. Artzner, P., Delbaen F., Eber, J.M., and D. Heath (1999): Coherent Measures of Risk. Mathematical Finance, 9, 203-228.
31. Rockafellar R.T. and S. Uryasev (2000): Optimization of Conditional Value-at-Risk. The Journal of Risk, accepted for publication. [online] Available at: www.ise.ufl.edu/uryasev/cvar.pdf and http://www.pacca.info/public/files/docs/public/finance/Active%20Risk%20Management/Uryasev%20-%20Cvar%20Optimization%20Algorithms%20and%20Application.pdf)
32. Artzner, F. Delbaen, J.M. Eber, and D. Heath. Coherent measures of risk. Mathematical Finance, 9:203–228, 1999
33. P.K. Swisher and G.W. Kasten. Post-modern portfolio theory. Journal of Financial Planning, 18:6674, 2005.
34. Jorion, Philippe (2006). Value at Risk: The New Benchmark for Managing Financial Risk (3rd ed.). McGraw-Hill. ISBN 978-0-07-146495-6.
35. R. Tyrrell Rockafellar and Stanislav Uryasev. Optimization of conditional value-at-risk. Journal of Risk, pages 2:21–41, 2000.
36. Susan Dean, Barbara Illowsky. Descriptive Statistics: Skewness and the Mean, Median, and Mode. [online] Available at: http://cnx.org/contents/6c4fb0df-8562-40db-8e48-3d91d7dd65f7@9/Descriptive-Statistics-Skewnes [accessed on 10th Oct, 2015]
37. Measures of Skewness and Kurtosis, Engineering statistics handbook. [online] Available at: http://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm [accessed on 10th Oct, 2015]
38. Dodge, Y. (2003). *The Oxford Dictionary of Statistical Terms*, OUP. ISBN 0-19-920613-9
39. Kronwald, Christian (2009). Credit Rating and the Impact on Capital Structure. Norderstedt, Germany: Druck und Bingdung. p. 3. ISBN 978-3-640-57549-7.
40. C.A. Coello Coello, D.A. Van Veldhuizen, G.B. Lamont. Evolutionary algorithms for solving multi-objective problems, Kluwer Academic Publishers, New York (2002).
41. ISBN 0-3064-6762-3
42. C.A. Coello Coello, G. Toscano Pulido. Multi-objective optimization using a micro-genetic algorithm L. Spector (Ed.), et al., Proceedings of the genetic and evolutionary computation conference (gecco'2001), Morgan Kaufmann Publishers, San Francisco, California (2001), pp. 274–282
43. Konak , Coit, Smith (2006). Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering and System Safety 91, pp. 992–1007
44. JP Morgan. RiskMetrics Technical Manual. JP Morgan, New York, 4th edition, 1996.
45. Linda Mon Xi. Portfolio Optimization with PCC-GARCH-CVaR model. Master's Thesis in Statistics Financial Theory and Insurance Mathematics. University of Bergen. [online] Available at: https://bora.uib.no/handle/1956/8555 [accessed on 10th Oct, 2015]
46. P. Artzner, F. Delbaen, J.M. Eber, and D. Heath. Coherent measures of risk. Mathematical Finance, pages 9:203–228, 1999
47. Chiranjeevi, Sastry. Multi Objective Portfolio Optimization Models and its solution using Genetic Algorithms. International Conference on Computational Intelligence and Multimedia Applications, 2007.
48. Konak , Coit, Smith (2006). Multi-objective optimization using genetic algorithms: A tutorial. Reliability Engineering and System Safety 91, pp. 992–1007
49. Zitzler E, Deb K, Thiele L. Comparison of multi-objective evolutionary algorithms: empirical results. Evol Comput 2000;8(2):pp.173–95.
50. Dilip K. Pratihar(2011). Soft computing: Fundamentals and applications (1st ed). Narosa, New Delhi, pp 80-200.

51. Bollerslev, T., Engle, R.F. and Wooldridge, J.M. (1988). A Capital Asset Pricing Model with Time-Varying Covariances, Journal of Political Economy 96: 116-131
52. Tim Bollerslev. Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, Vol. 31, 1986.
53. Campbell R. Harvey and Akhtar Siddique. Autoregressive Conditional Skewness. *Journal of Financial and Quantitative Analysis*, Vol. 34, No. 4, Dec 1999
54. Chris Brooks, Simon P. Burke, Saeed Heram, and Gita Persand. Autoregressive Conditional Kurtosis. *Journal of Financial Econometrics*, Vol. 3, No.3, 2005.
55. Angel Leon, Gonzalo Rubio, and Gregorio Serna. Autoregressive Conditional Volatility, Skewness And Kurtosis. *Working Papers. Serie AD*, 2004-13.
56. Heng wang. 2011. Conditional Skewness and Kurtosis in GARCH Model. [ONLINE] Available at: http://www.statistics.du.se/essays/D11_HengWang.pdf. [Accessed on 1 April 2016]
57. Sisi Peng & Huibo Deng, 2010. Modeling the Dynamic Conditional Correlation between Hong Kong and Tokyo Stock Markets with Multivariate GARCH models. [ONLINE] Available at: http://www.statistics.du.se/essays/D10_SisiPeng_HuiboDeng.pdf [Accessed on 1 April 2016]
58. Nelson Daniel B. (1991). Conditional Heteroskedasticity in Asset Returns: a New Approach. Econometrica, 59(2), pp.347-370.
59. Bollerslev Tim. (1990). Modeling the Coherence in Short-Run Nominal Exchange Rates: A Multivariate Generalized Arch Model. The Review of Economics and Statistics, 72(3), pp.498-505.
60. Engle Robert F.. (2002). Dynamic Conditional Correlation - A Simple Class of Multivariate GARCH Models. Journal of Business and Economic Statistics, 20(3), pp.339-350.
61. Engle, R. F., N. Shephard, and K. Sheppard (2007). Fitting and Testing Vast Dimensional Time-Varying Covariance Models. NYU Working Paper FIN-07-046. [online] available at: *http://vlab.stern.nyu.edu/doc/13?topic=mdls*[accessed on 1 May, 2016].

APPENDIX:

R-code:

```
setInternet2(TRUE)
data1<-read.csv(file.choose(),header=TRUE)
close.data1<-data1[ ,5]
> close.data1.ts<-ts(close.data1,frequency=252)(#f=252 for daily data) (use tseries package)
> log.ret1<-diff(log(close.data1.ts))
> adf.test(log.ret1,alternative=("stationary"))
>acf(log.ret1,type=("correlation"),plot=TRUE)
>pacf(log.ret1,lag.max=NULL,plot=TRUE,na.action=na.omit)
```

Arima:(use forecast package)

```
arima(log.ret1,order = c(2,1,2))
arima212=arima(log.ret1,order=c(2,1,2))
summary(arima212)
```

Residual:

```
  res.arima212=arima212$res
plot(res.arima212)
acf(residuals(arima212),type=("correlation"),plot=TRUE)
pacf(residuals(arima212),lag.max=NULL,plot=TRUE,na.action=na.omit)
```

squared residuals:

```
squared.res.arima212=res.arima212^2
plot(squared.res.arima212,main='Squared Residuals')
acf.squared212=acf(squared.res.arima212,main='ACF Squared Residuals',lag.max=100,ylim=c(-0.5,1))
pacf(squared.res.arima212,main='PACF Squared Residuals',lag.max=100,ylim=c(-
0.5,1),plot=TRUE,na.action=na.omit)
```

ARCH test: Ljung-Box/Portmanteau

```
Box.test(res.arima212,lag=1,type = "Ljung")
```

ARCH/GARCH

> arch01=garch(res.arima212,order=c(0,1),trace=F)

> loglik01=logLik(arch01)

> summary(loglik01)

ARIMA Forecast:

forecast212=forecast(arima212,100,level=95) #forecast for next 100 timeperiods

plot(forecast212)

Prediction:

forecast202=predict(arch01,n.ahead=1,plot=TRUE)

> plot(forecast202)

Conditional variances(ARCH):

ht.arch01=arch01$fit[,1]^2

plot(ht.arch01,main='Conditional variances')

Matlab code

(1)Objective function:

```
function [f,y]=objective(x,rtn,k)
[N,V] = size(x);
f = zeros(N,k);
% Objective function one - Returns
for j=1:N
sum1=0;
for i = 1:V
    sum1 = sum1+(x(j,i)*rtn(j,i));
end
% Decision variables are used to form the objective function.
f(j,k-1) = -sum1;

% Objective function two - CVaR
%Numerator of CVaR
ret1=0;ret2=0;ret3=0;ret4=0;ret5=0;n=floor(0.05*N);
for i=1:n
    ret1 = ret1 + rtn(i,1);
    ret2 = ret2 + rtn(i,2);
    ret3 = ret3 + rtn(i,3);
    ret4 = ret4 + rtn(i,4);
    ret5 = ret5 + rtn(i,5);
end
r(1) = ret1/n;r(2) = ret2/n;r(3) = ret3/n;r(4) = ret4/n;r(5) = ret5/n;
```

```
sum2=0;
for i = 1:V
    sum2 = sum2+(x(j,i)*r(i));
end
% Decision variables are used to form the objective function.
f(j,k) = abs(sum2);

y(j)=(0.5*f(j,k-1))+(0.5*f(j,k));
end
end
```

(2)Non-dominated sorting:

```
function [Rank,L,f] = Nondomsort(par,rtn)
%Non-Dominated Ranks formation
[N,V] = size(par); F = 2;
Sp = zeros(N,N);
Np = zeros(N,1);
[f,y] = objective(par,rtn,F);
for i=1:N-1
    for j=i+1:N
            comp = f(i,:)-f(j,:);
            n = find(comp<=0);
            if length(n)==2
                Sp(i,j) = 1;
                Np(j) = Np(j) + 1;
            end
            if length(n)<1
                Np(i) = Np(i) + 1;
                Sp(j,i) = 1;
            end
    end
end

Rmax = max(Np);
Rank = zeros(Rmax-1,N);
Nrem = [1:N]; i=1;
while max(Np(Nrem))>=2
    Ri = find(Np(Nrem)==0);
    L(i) = length(Ri);
    Rank(i,1:L(i)) = Nrem(Ri);
    for j=1:L(i)
        P = find(Sp(Nrem(Ri(j)),:)==1);
        Np(P) = Np(P) - 1;
    end
    Nrem = setdiff(Nrem,Nrem(Ri));
    i = i+1;
end
R = i-1;
Rank = Rank(1:R,1:max(L));
end
```

(3)Binary to Decimal conversion:

```
function [par] = gadecode(pop,nbits,x)
    [N,L] = size(pop); V = size(x,2); xmin = min(x); xmax = max(x);
    par = zeros(N,V);
    for i=1:N
        n = 1;
    for j=1:nbits:L
        B = bi2de(pop(i,j+nbits-1:-1:j));
        par(i,n) = xmin(n) + (xmax(n)-xmin(n))*B/(2^nbits-1);
        n = n + 1;
    end
```

```
    end
end
```

(4)Fitness of a string:

```
function [cost,rtnpar] = fitness(par,x,rtn)
    [N,V] = size(par);
    rtnpar = zeros(N,V);
    for j=1:N
    for i=1:V
        Hi = find(x(:,i)>=par(j,i));
        Lo = find(x(:,i)<=par(j,i));
        [H,iH] = sort(x(Hi,i));
        [L,iL] = sort(x(Lo,i),1,'descend');
        rtnpar(j,i) = (rtn(iH(1),i)+rtn(iL(1),i))/2;
    end
    end
    for j=1:N
        sum1=0;
    for i = 1:V
        sum1 = sum1+(par(j,i)*rtnpar(j,i));
    end
    % Decision variables are used to form the objective function.
    y1 = -sum1;
    ret1=0;ret2=0;ret3=0;ret4=0;ret5=0;n=floor(0.05*N);
    for i=1:n
        ret1 = ret1 + rtnpar(i,1);
        ret2 = ret2 + rtnpar(i,2);
        ret3 = ret3 + rtnpar(i,3);
        ret4 = ret4 + rtnpar(i,4);
        ret5 = ret5 + rtnpar(i,5);
    end
    r(1) = ret1/n;r(2) = ret2/n;r(3) = ret3/n;r(4) = ret4/n;r(5) = ret5/n;

    sum2=0;
    for i = 1:V
        sum2 = sum2+(par(j,i)*r(i));
    end
    % Decision variables are used to form the objective function.
    y2 = sum2;
    cost(j) = 0.5*(y1+y2);
    end
end
```

(5)Crowding:

```
function [Cd] = Crowding(Rank,f)
    [R,L] = size(Rank);
    m = size(f,2); Cd = zeros(R,L);
    Cd(:,1) = 100;
    for i=1:R
        Ndnum = find(Rank(i,:)>0);
        Ndset = Rank(i,Ndnum);
        for k=2:length(Ndset)-1
            Cd(i,length(Ndset)) = 100;
        for j=1:m
            [F1,I] = sort(f(Ndset,j));
            Cd(i,k) = Cd(i,k) + abs(f(I(k+1),j)-f(I(k-1),j))/abs(F1(end)-F1(1));
        end
        end
    end
end
```

(6)Binary GA:

```
maxit=100;
mincost=-9999999;

x=xlsread('NewData.xlsx','Sheet1','ac2:ag246');
rtn=xlsread('NewData.xlsx','Sheet1','bb2:bf246');
xmin = min(x); xmax = max(x);
npar=size(x,2);
popsize=size(x,1); % set population size
mutrate=0.1; % set mutation rate
selection=0.5; % fraction of population
% kept
nbits=10;  % number of bits in each
% parameter
Nt=nbits*npar;  % total number of bits
% in a chormosome
keep=floor(selection*popsize); % #population members
% that survive
pop = [];
popi = zeros(popsize,Nt);
iga=0; % generation counter
% initialized
for i=1:popsize
    for j=1:npar
        Bi = zeros(1,nbits);
        B = round((x(i,j) - xmin(j))*(2^nbits-1)/(xmax(j)-xmin(j)));
        D = de2bi(B); Bi(1:length(D)) = D;
        popi(i,j*nbits-nbits+1:j*nbits) = fliplr(Bi);
    end
end

par=gadecode(popi,nbits,x); % convert binary to
% continuous values
[cost,rtnpar]=fitness(par,x,rtn); % calculates population
% cost using ff
[Rank,L,f] = Nondomsort(par,rtnpar);
[Cd] = Crowding(Rank,f);
for i=1:size(Rank,1)
    pop = [pop;popi(Rank(i,1:L(i)),:)];
end
N = size(pop,1);
if N<popsize
    k = 1; nr = 1;
    while N+k<popsize
        if L(nr)<=popsize-N
        pop(k+N,:) = popi(Rank(nr,k),:);
        k = k+1;
        if k>L(nr)
            nr=nr+1;
        end
        elseif L(nr)>popsize-N
            [C,id] = sort(Cd(nr,1:L(nr)),2,'descend');
            pop(k+N:popsize,:) = popi(Rank(nr,id(1:popsize-N-k+1)),:);
            k=popsize-N+1;
        end
    end
end

% [cost,ind]=sort(cost); % min cost in element 1
% par=par(ind,:);pop=pop(ind,:); % sorts population with
% lowest cost first
minc(1)=min(cost); % minc contains min of
% population
meanc(1)=mean(cost); % meanc contains mean
% of population
```

```
while iga<maxit
iga=iga+1;
M=ceil((popsize-keep)/2); % number of matings
prob=flipud([1:keep]'/sum([1:keep]));% weights
% chromosomes based upon position in list
odds=[0 cumsum(prob(1:keep))']; % probability distribution function
% PROGRAM I: BINARY GENETIC ALGORTHIM 213
pick1=rand(1,M); % mate #1
pick2=rand(1,M); % mate #2
ic=1;
while ic<=M
for id=2:keep+1
if pick1(ic)<=odds(id) && pick1(ic)>odds(id-1)
ma(ic)=id-1;
end % if
if pick2(ic)<=odds(id) && pick2(ic)>odds(id-1)
pa(ic)=id-1;
end % if
end % id
ic=ic+1;
end % while
ix=1:2:keep; % index of mate #1
xp=ceil(rand(1,M)*(Nt-1)); % crossover point
pop(keep+ix,:)=[pop(ma,1:xp) pop(pa,xp+1:Nt)];
% first offspring
pop(keep+ix+1,:)=[pop(pa,1:xp) pop(ma,xp+1:Nt)];
% second offspring
nmut=ceil((popsize-1)*Nt*mutrate); % total number
% of mutations
mrow=ceil(rand(1,nmut)*(popsize-1))+1; % row to mutate
mcol=ceil(rand(1,nmut)*Nt); % column to mutate
for ii=1:nmut
pop(mrow(ii),mcol(ii))=abs(pop(mrow(ii),mcol(ii))-1);
% toggles bits
end % ii
% The population is re-evaluated for cost
par=gadecode(pop,nbits,x);
% decode
[cost,rtnpar]=fitness(par,x,rtn);
minc(iga+1)=min(cost);
meanc(iga+1)=mean(cost);
popn = [];
[Rank,L,f] = Nondomsort(par,rtnpar);
[Cd] = Crowding(Rank,f);
for i=1:size(Rank,1)
    popn = [popn;pop(Rank(i,1:L(i)),:)];
end
N = size(popn,1);
if N<popsize
    k = 1; nr = 1;
    while N+k<popsize
        if L(nr)<=popsize-N
        popn(k+N,:) = pop(Rank(nr,k),:);
        k = k+1;
        if k>L(nr)
            nr=nr+1;
        end
        elseif L(nr)>popsize-N
            [C,id] = sort(Cd(nr,:),2,'descend');
            popn(k+N:popsize,:) = pop(Rank(nr,id(1:popsize-N-k+1)),:);
            k=popsize-N+1;
        end
    end
end
pop = popn;
if iga>maxit || cost(1)<mincost
```

```
break
end
[iga cost(1)]
end

%plot Ranks
for i=1:size(Rank,1)
    plot(f(Rank(i,1:L(i)),1),f(Rank(i,1:L(i)),2),'o')
    hold all
end
%y=(0.5*f(Rank(1,1:L(1)),1))+(0.5*f(Rank(1,1:L(1)),2));
%[c,i] = min(y);
%x(Rank(1,i),:)
```

RATS CODE:

```
DATA(FORMAT=XLS,ORG=COLUMNS) 1 246 R_HDFC     R_SUNPHARMA
     R_TATAMOTORS R_INFOSYS R_ITC

system(model=var1)

variables R_HDFC   R_SUNPHARMA

lags 2

det constant

end(system)

GARCH(P=1,Q=1,mv=dcc,variances=var1,ITERS=4000,PITERS=15,pmethod=simplex,ROBUST,hmatrices=hh,rvectors=rd) / R_HDFC    R_SUNPHARMA

set z1 %regstart() %regend() = rd(t)(1)/sqrt(hh(t)(1,1))

set z2 %regstart() %regend() = rd(t)(2)/sqrt(hh(t)(2,2))

set rho12dgold %regstart() %regend() = hh(t)(1,2)/sqrt(hh(t)(1,1)*hh(t)(2,2))

set usdgoldf %regstart() %regend() = hh(t)(1,2)

set varusdinrf %regstart() %regend() = hh(t)(1,1)

set vargoldf %regstart() %regend() = hh(t)(2,2)

set hedgeusdgoldf %regstart() %regend() = usdgoldf/vargoldf

print
```

ARIMA AND GARCH RESULTS:

(1) HDFC

ARIMA211

Dependent Variable: SERIES01
Method: ARMA Maximum Likelihood (OPG - BHHH)
Date: 04/24/16 Time: 14:51
Sample: 1 245
Included observations: 245
Convergence achieved after 16 iterations
Coefficient covariance computed using outer product of gradients

Variable	Coefficient	Std. Error	t-Statistic	Prob.
C	0.000149	0.001045	0.142672	0.8867
AR(1)	-0.285532	0.353038	-0.808785	0.4194
AR(2)	-0.152988	0.065099	-2.350067	0.0196
MA(1)	0.275093	0.356504	0.771642	0.4411
SIGMASQ	0.000339	2.80E-05	12.09004	0.0000

R-squared	0.024080	Mean dependent var	0.000123
Adjusted R-squared	0.007814	S.D. dependent var	0.018665
S.E. of regression	0.018591	Akaike info criterion	-5.111822
Sum squared resid	0.082954	Schwarz criterion	-5.040367
Log likelihood	631.1981	Hannan-Quinn criter.	-5.083047
F-statistic	1.480435	Durbin-Watson stat	1.975109
Prob(F-statistic)	0.208747		

Inverted AR Roots	-.14-.36i	-.14+.36i
Inverted MA Roots	-.28	

GARCH11

Dependent Variable: SERIES01
Method: ML ARCH - Normal distribution (BFGS / Marquardt steps)
Date: 04/24/16 Time: 14:54
Sample: 1 245
Included observations: 245
Convergence achieved after 32 iterations
Coefficient covariance computed using outer product of gradients
Presample variance: backcast (parameter = 0.7)
GARCH = C(1) + C(2)*RESID(-1)^2 + C(3)*GARCH(-1)

Variable	Coefficient	Std. Error	z-Statistic	Prob.
		Variance Equation		
C	1.50E-05	1.18E-05	1.268337	0.2047
RESID(-1)^2	0.001554	0.014046	0.110652	0.9119
GARCH(-1)	0.958018	0.037611	25.47154	0.0000

R-squared	-0.000044	Mean dependent var	0.000123
Adjusted R-squared	0.004038	S.D. dependent var	0.018665
S.E. of regression	0.018627	Akaike info criterion	-5.120666
Sum squared resid	0.085005	Schwarz criterion	-5.077794
Log likelihood	630.2816	Hannan-Quinn criter.	-5.103402
Durbin-Watson stat	2.010989		

(2) SUN PHARMA

Dependent Variable: SERIES02
Method: ML ARCH - Normal distribution (BFGS / Marquardt steps)
Date: 04/24/16 Time: 15:08
Sample: 1 245
Included observations: 245
Failure to improve likelihood (non-zero gradients) after 382 iterations
Coefficient covariance computed using outer product of gradients
Presample variance: backcast (parameter = 0.7)
GARCH = C(1) + C(2)*RESID(-1)^2 + C(3)*GARCH(-1)

Variable	Coefficient	Std. Error	z-Statistic	Prob.
		Variance Equation		
C	1.24E-06	1.62E-06	0.762932	0.4455
RESID(-1)^2	-0.043150	0.005102	-8.457973	0.0000
GARCH(-1)	1.058371	0.011289	93.75503	0.0000

R-squared	-0.000005	Mean dependent var	4.94E-05
Adjusted R-squared	0.004077	S.D. dependent var	0.022470
S.E. of regression	0.022424	Akaike info criterion	-5.039645
Sum squared resid	0.123194	Schwarz criterion	-4.996772
Log likelihood	620.3565	Hannan-Quinn criter.	-5.022380
Durbin-Watson stat	1.896575		

(3) TATA MOTORS

Dependent Variable: SERIES03
Method: ML ARCH - Normal distribution (BFGS / Marquardt steps)
Date: 04/24/16 Time: 15:10
Sample: 1 245
Included observations: 245
Convergence achieved after 21 iterations
Coefficient covariance computed using outer product of gradients
Presample variance: backcast (parameter = 0.7)
GARCH = C(1) + C(2)*RESID(-1)^2 + C(3)*GARCH(-1)

Variable	Coefficient	Std. Error	z-Statistic	Prob.
		Variance Equation		
C	0.000274	0.000184	1.484790	0.1376
RESID(-1)^2	0.190084	0.104502	1.818947	0.0689
GARCH(-1)	0.175422	0.491463	0.356939	0.7211

R-squared	-0.007635	Mean dependent var	-0.001807
Adjusted R-squared	-0.003522	S.D. dependent var	0.020720
S.E. of regression	0.020757	Akaike info criterion	-4.924285
Sum squared resid	0.105556	Schwarz criterion	-4.881412
Log likelihood	606.2249	Hannan-Quinn criter.	-4.907020
Durbin-Watson stat	2.045067		

(4) INFOSYS

Dependent Variable: SERIES04
Method: ML ARCH - Normal distribution (BFGS / Marquardt steps)
Date: 04/24/16 Time: 15:12
Sample: 1 245
Included observations: 245
Failure to improve likelihood (singular hessian) after 200 iterations
Coefficient covariance computed using outer product of gradients
Presample variance: backcast (parameter = 0.7)
GARCH = C(1) + C(2)*RESID(-1)^2 + C(3)*GARCH(-1)

Variable	Coefficient	Std. Error	z-Statistic	Prob.
		Variance Equation		
C	7.93E-05	3.53E-06	22.45096	0.0000
RESID(-1)^2	-0.007715	0.000164	-47.17190	0.0000
GARCH(-1)	0.992560	0.001209	820.7609	0.0000

R-squared	-0.005832	Mean dependent var	-0.005007
Adjusted R-squared	-0.001727	S.D. dependent var	0.065702
S.E. of regression	0.065759	Akaike info criterion	-3.027822
Sum squared resid	1.059426	Schwarz criterion	-2.984949
Log likelihood	373.9082	Hannan-Quinn criter.	-3.010557
Durbin-Watson stat	1.989637		

(5) ITC

Dependent Variable: SERIES05
Method: ML ARCH - Normal distribution (BFGS / Marquardt steps)
Date: 04/24/16 Time: 15:14
Sample: 1 245
Included observations: 245
Convergence achieved after 35 iterations
Coefficient covariance computed using outer product of gradients
Presample variance: backcast (parameter = 0.7)
GARCH = C(1) + C(2)*RESID(-1)^2 + C(3)*GARCH(-1)

Variable	Coefficient	Std. Error	z-Statistic	Prob.
		Variance Equation		
C	0.000333	0.000121	2.757754	0.0058
RESID(-1)^2	0.106280	0.065320	1.627075	0.1037
GARCH(-1)	-0.310436	0.437626	-0.709362	0.4781

R-squared	-0.000638	Mean dependent var	-0.000423
Adjusted R-squared	0.003446	S.D. dependent var	0.016785
S.E. of regression	0.016756	Akaike info criterion	-5.347252
Sum squared resid	0.068784	Schwarz criterion	-5.304379
Log likelihood	658.0384	Hannan-Quinn criter.	-5.329987
Durbin-Watson stat	1.902244		