Exploratory Project

Dynamic Portfolio Optimization in Finance

Name: Sanjana Garai
Roll no.: 22124040
Branch: Mathematics and Computing 2nd year
Guide: Dr. Debdas Ghosh

Abstract

- This paper introduces the Normal Boundary Intersection (NBI) method for highly accurate Pareto frontier prediction in multicriteria optimization as compared to the traditional Weighted Sum Scalarization technique.
- It presents dynamic portfolio optimization, highlighting its advantages over static methods in finance.
- It states the mean-variance Portfolio model and introduces a new systematic procedure employing NBI for dynamic portfolio optimization through strategic asset reallocation.
- Additionally, it showcases empirical analysis using curve fitting techniques to predict future stock prices by defining a polynomial function that can predict the price of a stock given any time in the future.

Normal Boundary Intersection Method For Solving Multicriteria Optimization Problems

- **Multicriteria Optimization:** Involves finding solutions that balance multiple conflicting objectives simultaneously. Example: we might need to build a product with minimum cost and pollution

- **DEFINITION (Pareto Point):** *The vector $F(x)$ is said to dominate another vector $F(\bar{x})$, denoted by $F(x) < F(\bar{x})$, if and only if $f_i(x) \leq f_i(\bar{x})$ for all $i \in \{1, 2, \ldots, n\}$ and $f_j(x) < f_j(\bar{x})$ for some $j \in \{1, 2, \ldots, n\}$. A point $x^* \in C$ is said to be globally Pareto optimal or a globally efficient point for (MOP) if and only if there does not exist $x \in C$ satisfying $F(x) < F(x^*)$. $F(x^*)$ is then called globally nondominated or noninferior.*

- Set of all pareto points make the pareto frontier. Our main goal through optimization will be to find this.

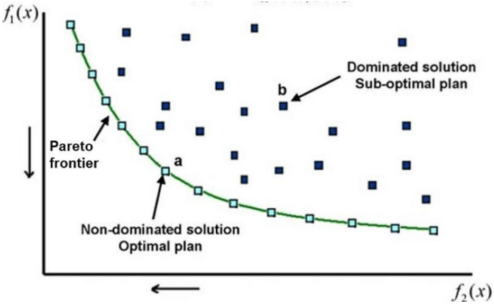
Multicriteria Optimization Problem

- In mathematics, a multicriteria optimization problem can be loosely posed as:

$$\text{“}\min_{x \in C}\text{”} F(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix}, \quad n \geq 2, \quad \ldots (MOP),$$

where C $= \{$ x: h(x) $= 0$, $g(x) \leq y$, $a \leq x \leq b$ $\}$

- Below is a graph demonstrating an MOP with the objective to minimize $f_1(x)$ and $f_2(x)$.



Weighted Sum Scalarization Optimization Approach

- The multicriteria optimization problem is converted into a single objective problem by weighted linear combination

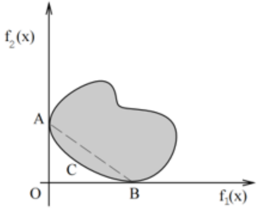
  $$\min \quad \sum_{i=1}^{p} \lambda_i f_i(x)$$
  $$\text{s.t.} \quad x \in X.$$

  where C = { x: h(x) = 0, $g(x) \leq y$, $a \leq x \leq b$ }
  need to be considered for $\lambda \in \mathbb{R}$ with $\lambda_i > 0$,(i = 1,2,...,n)
  and $\sum_{i=1}^{p} \lambda_i = 1$
- Disadvantages of WSS
  (i) Uneven distribution of the Pareto points due to which the exact shape of the pareto frontier cannot be understood
  (ii) The function in the single objective problems are extremely sensitive to change in values of lambda
- Clearly, as the exact shape of the pareto frontier cannot be estimated, we introduce NBI – Normal Boundary Intersection method which generates evenly spaced pareto points.

- **Preliminaries:** *Convex hull of individual minima (CHIM)*. Let x_i^* be the respective global minimizers of $f_i(x)$, $i = 1, \ldots, n$ over $x \in C$. Let $F_i^* = F(x_i^*)$, $i = 1, \ldots, n$. Let ϕ be the $n \times n$ matrix whose i-th column is $F_i^* - F^*$, sometimes known as the *Payoff matrix*. Then the set of points in $\mathbb{R}^n$ that are convex combinations of $F_i^* - F^*$, i.e.,

$$\left\{ \phi\beta : \beta \in \mathbb{R}^n, \sum_{i=1}^{n} \beta_i = 1, \beta_i \geq 0 \right\}$$

is referred to as the CHIM.



In the bi-objective optimization problem to the left, aimed at minimizing $f_1(x)$ and $f_2(x)$, the dotted line connecting A and B represents the CHIM.
A is the global minimizer of $f_1(x) = F_1^*(x)$, and B is the global minimizer of $f_2(x) = F_2^*(x)$.

- *Utopian point:* The utopian point U for a multiobjective minimization problem with n objective functions $f_i(x)$ where $i = 1, 2, \ldots, n$ can be mathematically represented as:

$$U = (U_1, U_2, \ldots, U_n)$$

Here, each U_i represents the ideal minimum value for the corresponding objective function $f_i(x)$. The utopian point is usually practically unattainable as it lies outside the feasible region. In the above graph, the point O is the utopian point.

- *Structure of the payoff matrix:* ϕ*:* The ith column of ϕ is:

$$\phi(:, i) = F(x_i^*) - F^*$$

Since $f_i(x_i^*) = f_i^*$, clearly,

$$\phi(i, i) = 0$$

Moreover, since x_i^* is the minimizer of $f_i(x)$ over x_j^* for $j = 1, \ldots, n$,

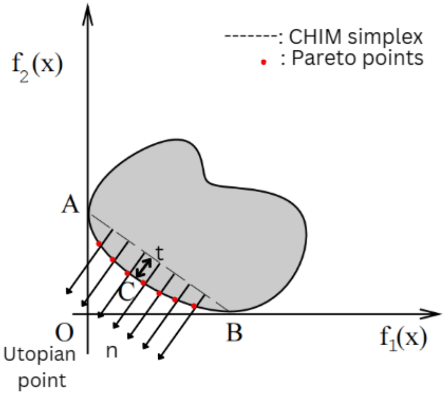
$$\phi(j, i) \geq 0, \quad j \neq i$$

A negative value in the matrix will signify that x_k^* is not the global minimizer of $f_k(x)$.

Formulating the NBI subproblem

- Our objective is to minimize $F = [f_i(x)]$ subject to specific constraints utilizing the NBI method. We first find the values of x in C that individually minimize each of the $f_i(x)$, let them be x_i^*.
- Then $F^*(x_i)$ represents the global minimizers; we obtain n of them.

  On joining the $F^*(x_i)$, we get the CHIM simplex.



We then have to find points of intersection of the normals from the CHIM to the boundary of the feasible region closest to the origin. The distance between the CHIM and the boundary of the feasible region closest to the origin is denoted by t. This is illustrated in the figure above.

The NBI subproblem

- Our main optimization problem is to maximise t. But, as we are defining a minimisation problem, our objective becomes:
  min -t
  s.t. $\phi\beta + t\hat{n} = F(x)$
  Constraints
  h(x) $=$ 0, g(x) $\leq$ 0, a $\leq$ x $\leq$ b
  where,
  $\phi =$ payoff matrix
  $\beta =$ matrix that determines the distribution of points on the CHIM simplex
  $\hat{n} =$ normal unit vector

Comparing WSS and NBI with an example

- Consider the below optimization problem:

$$f_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2$$

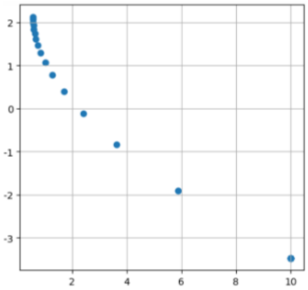
$$f_2(x) = 3x_1 + 2x_2 - \frac{x_3}{3} + 0.01(x_4 - x_5)$$

Minimize $f_1(x)$, $f_2(x)$ subject to:

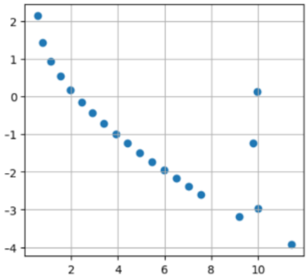
$$x_1 + 2x_2 - x_3 - 0.5x_4 + x_5 = 2$$

$$4x_1 - 2x_2 + 0.8x_3 + 0.6x_4 + 0.5x_5^2 = 0$$

$$x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 \leq 10$$



Solved using WSS



Solved using NBI

Advantages of NBI over WSS

- From the above 2 graphs, it is clear that NBI method provides evenly spaced pareto points and hence, it is easier to determine the exact shape of the pareto frontier
  The code to solve the above problems using WSS and NBI and obtain the graphs is: Link to code

Verifying NBI method on standard multiobjective optimization problems

- Osyczka and Kundu Function

$$\text{Minimize} = \begin{cases} f_1(\boldsymbol{x}) = -25(x_1-2)^2 - (x_2-2)^2 - (x_3-1)^2 - (x_4-4)^2 - (x_5-1)^2 \\ f_2(\boldsymbol{x}) = \sum_{i=1}^{6} x_i^2 \end{cases}$$

$$\text{s.t.} = \begin{cases} g_1(\boldsymbol{x}) = x_1 + x_2 - 2 \geq 0 \\ g_2(\boldsymbol{x}) = 6 - x_1 - x_2 \geq 0 \\ g_3(\boldsymbol{x}) = 2 - x_2 + x_1 \geq 0 \\ g_4(\boldsymbol{x}) = 2 - x_1 + 3x_2 \geq 0 \\ g_5(\boldsymbol{x}) = 4 - (x_3-3)^2 - x_4 \geq 0 \\ g_6(\boldsymbol{x}) = (x_5-3)^2 + x_6 - 4 \geq 0 \end{cases} \qquad \begin{array}{c} 0 \leq x_1, x_2, x_6 \leq 10 \\ , 1 \leq x_3, x_5 \leq 5, \\ 0 \leq x_4 \leq 6. \end{array}$$

Viennet Function

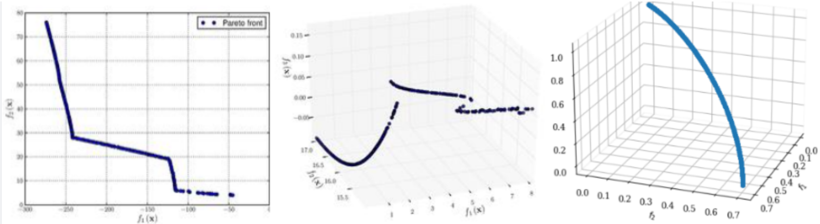
$$\text{Minimize} = \begin{cases} f_1(x,y) = 0.5\left(x^2+y^2\right) + \sin\left(x^2+y^2\right) \\ f_2(x,y) = \frac{(3x-2y+4)^2}{8} + \frac{(x-y+1)^2}{27} + 15 \\ f_3(x,y) = \frac{1}{x^2+y^2+1} - 1.1\exp\left(-\left(x^2+y^2\right)\right) \end{cases}$$

$-3 \leq x, y \leq 3$

DTLZ5 Function

$$\begin{array}{ll}
Min. & f_1(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(\theta_1\pi/2)\cdots\cos(\theta_{M-2}\pi/2)\cos(\theta_{M-1}\pi/2), \\
\text{Min.} & f_2(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(\theta_1\pi/2)\cdots\cos(\theta_{M-2}\pi/2)\sin(\theta_{M-1}\pi/2), \\
\text{Min.} & f_3(\mathbf{x}) = (1+g(\mathbf{x}_M))\cos(\theta_1\pi/2)\cdots\sin(\theta_{M-2}\pi/2), \\
\vdots & \vdots \\
\text{Min.} & f_M(\mathbf{x}) = (1+g(\mathbf{x}_M))\sin(\theta_1\pi/2), \\
\text{with} & \theta_i = \frac{1}{2(1+g(\mathbf{x}_M))}(1+2g(\mathbf{x}_M)x_i), \quad \text{for } i = 2,3,\ldots,(M-1), \\
 & g(\mathbf{x}_M) = \sum_{x_i \in \mathbf{x}_M}(x_i - 0.5)^2, \\
 & 0 \leq x_i \leq 1, \quad \text{for } i = 1,2,\ldots,n.
\end{array}$$

Graphs of the 3 optimization problems



Osyczka and Kundu Viennet DTLZ5

Clearly, the NBI method can very accurately determine the Pareto frontier

Further, it has been discussed how the NBI method can be applied to Dynamic Portfolio Optimization

Portfolio Optimization

- **Portfolio Optimization:** Refers to the process of constructing an investment portfolio that maximizes returns or minimizes risks based on certain objectives, constraints, and statistical measures. It is of 2 types: **Static** and **Dynamic**
- Dynamic portfolio optimization adjusts asset allocation over time based on changing market conditions, while static portfolio optimization maintains a fixed allocation.

Advantages of Dynamic over Static portfolio allocation

- **Adaptive:** Dynamic strategies swiftly adjust to changing markets and risks.
- **Robust Risk handling:** Effectively manage uncertainties in evolving markets.
- **Enhanced Returns:** Flexibility leads to better risk management and higher return potential.

- The most commonly used method for Portfolio Optimization is:
  Markowitz Portfolio Optimization / Mean-variance model
- Consider n risky assets i=1, ..., n
- The mean and covariance of the retuns are as follows:

$$E[\mathbf{R}] = \boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_m \end{bmatrix}, Cov[\mathbf{R}] = \boldsymbol{\Sigma} = \begin{bmatrix} \Sigma_{1,1} & \cdots & \Sigma_{1,m} \\ \vdots & \ddots & \vdots \\ \Sigma_{m,1} & \cdots & \Sigma_{m,m} \end{bmatrix}$$

- The portfolio vector is the vector of weights w_i given by $w = [w_1, w_2, \ldots, w_n]$, where w_i represents the fraction of the total wealth to be invested in asset i and $\sum_{i=1}^{n} w_i = 1$
- The returns vector is defined as $R = [R_1, R_2, \ldots, R_n]$
- *Portfolio Return:* $R_w = w\prime R = \sum_{i=1}^{n} w_i R_i$
  $\alpha_w = E(R_w) = w\prime\alpha$
  $\sigma_w^2 = \mathrm{var}[R_w] = w\prime\Sigma w$

- We now have a biobjective problem:
- To minimize the risk:

$$\min \frac{1}{2} w' \Sigma w$$

- Maximize the return: But, as we are dealing with only minimization problems, we can write:

$$\min -E(R_w) = w' \alpha$$

Subject to constraints:

$$w' \alpha = \alpha_0$$

$$w' \Sigma w = \sigma_0^2$$

$$w' 1_m = 1$$

α_0 is the average return as expected by the investor

σ_0 is the average standard deviation

- Using WSS, the problem becomes:

$$\min \frac{1}{2} \lambda \mathrm{Var}(R_w) - E(R_w) = \frac{1}{2} \lambda w' \Sigma w - w' \alpha$$

Dynamic Portfolio Allocation using NBI method: An Example

- In the code implementation, we have a portfolio of 3 stocks: ADANIENT.NS, EICHERMOT.NS, SBI.NS
- We have a look back period of 6 months. Using this 6 months data, we determine the best portfolio using NBI method.
- We have a review period of 1 month, which means that after every month, we again determine the optimal portfolio but this time with the new month's data appended to our dataset.
- Hence, the strategy is to **invest a finite amount at once and then keep performing asset reallocation according to market trends after every month.**

  Below are the snippets of the intermediaries obtained:

- Individual minima

```
Portfolio weights for minimum risk:  [0.2        0.55159629 0.24840371]
Minimum risk:  0.01407330255971806
Portfolio weights for maximum return:  [0.6 0.2 0.2]
Maximum return:  0.017204351773769992
```

- Utopian point

```
F1* = ( 0.01407330255971806 , -0.012588137432705662 )
F2* = ( 0.02350340705335698 , -0.017204351773769992 )
```

- Payoff Matrix

```
Payoff matrix
[        0                  0.004616214341064330
[ 0.009430104493638920      0                  ]
```

- NBI subproblem definition in Python

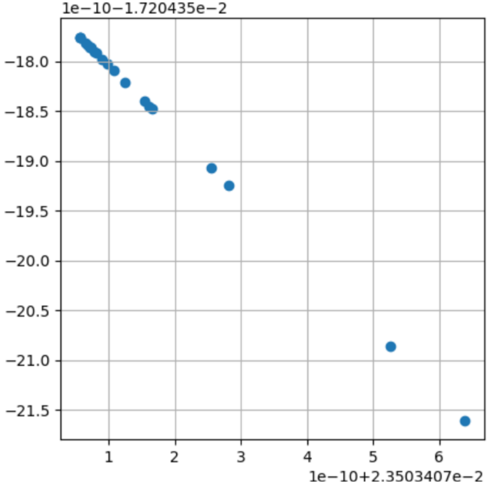
```python
def main_nbi(x):
  return -x[3]

def nbi_con1(x):
  global a
  return (1-a)*y2 - x[3]*0.898 - sd_adani*x[0] + sd_sbi*x[1] + sd_eichermot*x[2] + objective1(min_risk.x)

def nbi_con2(x):
  global a
  return y1*a - x[3]*0.4399 + -(avg_adani_return*x[0] + avg_sbi_return*x[1] + avg_eichermot_return*x[2]) + objective2(max_return.x)

con1 = {'type': 'eq', 'fun': nbi_con1}
con2 = {'type': 'eq', 'fun': nbi_con2}
con3 = {'type': 'eq', 'fun': constraint1}
con4 = {'type': 'ineq', 'fun': constraint2}
constraints = [con1, con2, con3, con4]
b = (0.2, 0.8)
b2 = (-10e5, 10e5)
bnds = (b, b, b, b2)
```

- Graph showing pareto frontier:
  x-axis: risk, y-axis: -return



  Link to Code

  Hence, as demostrated, we can use the NBI optimization method for dynamic portfolio allocation

Time dependency of Stock prices: Volatile nature of Stock prices

- The prices of stocks fluctuate greatly with time but follow certain patterns when we take a large time interval. Hence, we can use curve fitting techniques and built-in polynomial regression techniques in Python to determine a function that can predict the price of a stock given a time in the future.

An example: Code Implementation

- Here, I have determined a 3-degree and a 10-degree polynomial to predict the price of the SBI stock.
- With a lookback period of 1 year, I have got 25 data points, each representing the average closing price for a 10-day interval period. The x data points refer to the time intervals from 1 to 10, and the y data points refer to the average close price of the stock.
- (1) 3-degree polynomial: $ax^3 + bx^2 + cx + d$

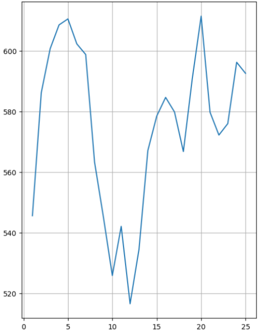
```
The optimised parameters are:  [ 1.46708925e-03  1.84602027e-01 -5.43028547e+00  5.98629011e+02]
```

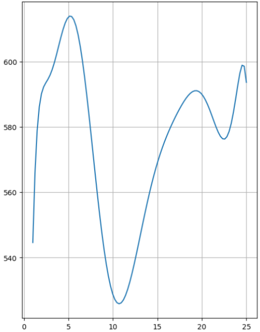
(2) 10 degree polynomial:

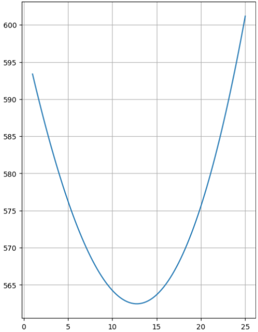
$$ax^{10} + bx^9 + cx^8 + dx^7 + ex^6 + fx^5 + gx^4 + hx^3 + ix^2 + kx + l$$

```
The optimised parameters are:  [-5.22051316e-08  6.95806076e-06 -3.98955852e-04  1.28547625e-02
 -2.55025308e-01  3.20958971e+00 -2.54967405e+01  1.23905368e+02
 -3.49421266e+02  5.24274732e+02  2.68373483e+02]
```

- The graph of the actual data points, the 10-degree polynomial, and 3-degree polynomial are as follows:







- We can see that the 10-degree polynomial curve is a great fit to the actual data points.

- We have predicted the average close price of the 26th time interval and compared it to the actual close price. Link to code

```
Prediction of average close price of 26th time interval
by 10-degree curve:  457.0374328088384
by 3 degree cruve:  608.018195490568
Actual close price: 513.02
```

- Clearly, the 10 degree curve provides a better estimate, however it also has a relative error of 11%, whereas, the 3 degree curve has a relative error of 18.5%.
- We can get a better estimate by taking an even higher order polynomial
- **Hence, we can conclude that the stock market is highly volatile and hence, investment decisions must be dynamic which clearly supports the idea of dynamic portfolio allocation over static portfolio allocation**

Conclusion

- The Normal Boundary Intersection method is a robust way to determine the pareto frontier as it produces evenly spaced pareto points and hence, the shape of the pareto frontier can be determined to great accuracy.
- Dynamic portfolio optimization is much more optimal than static portfolio optimization as it efficiently deals with high market volatility and helps reducing risks and increasing returns.
- The NBI method can be used for dynamic portfolio allocation by applying an asset reallocation strategy.
- Higher order polynomials can be used to predict the price of a stock in future time with good accuracy. Curve-fitting techniques can be used to determine the optimal parameters.

References

- **Dynamic Portfolio Optimization with Liquidity Cost and Market Impact: A Simulation-and-Regression Approach**, Rongju Zhang, Nicolas Langrene, Yu Tian, Zili Zhu, Fima Klebaner, Kais Hamza, Monash CQFIS Working paper 2019
- **Robust multiobjective optimization & applications in portfolio optimization**, Jorg Fliege, Ralf Werner, 26th October, 2013
- **Normal Boundary Intersection method for generation of the efficient frontier**, Indraneel Das and JE Dennis, Vol. 8, No. 3, pp. 631–657, August 1998
- **Stochastic programming approach for dynamic portfolio allocation**, Diana Barro, Giorgio Consigli, Vivek Varun, 20 February, 2022