Enhanced Portfolio Selection through Ensemble Predictive Modeling

Project-III (MA57301) report submitted to

Indian Institute of Technology Kharagpur

in partial fulfilment for the award of the degree of

Integrated Master of Science

in

Mathematics and Computing

by

Atharv Bajaj

(20MA20014)

Under the supervision of

Professor Debjani Chakraborty

Department of Mathematics

Indian Institute of Technology Kharagpur

Autumn Semester, 2024-25

DECLARATION

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 have used 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 references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

Date: (Atharv Bajaj)

Place: Kharagpur (20MA20014)

DEPARTMENT OF MATHEMATICS

INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

KHARAGPUR - 721302, INDIA

CERTIFICATE

This is to certify that the project report entitled "**Enhanced Portfolio Selection through Ensemble Predictive Modeling**" submitted by **Atharv Bajaj** (Roll No. 20MA20014) to Indian Institute of Technology Kharagpur towards partial fulfilment of requirements for the award of degree of Integrated Master of Science in Mathematics and Computing is a record of bona fide work carried out by him under my supervision and guidance during Autumn Semester, 2024-25.

Professor Debjani Chakraborty
Department of Mathematics
Indian Institute of Technology Kharagpur
Kharagpur - 721302, India

Date:
Place: Kharagpur

Abstract

Effective portfolio management requires careful consideration of both asset selection and risk management, with the expected return on each asset being a crucial factor. Since the success of portfolio construction largely depends on the future performance of stock markets, accurately predicting stock returns can significantly enhance decision-making. Recent developments in machine learning have opened up valuable opportunities to incorporate predictive modeling into portfolio selection. However, research indicates that single-model approaches often lack the accuracy and robustness needed to achieve both precise predictions and substantial returns.

This study proposes a novel approach to portfolio construction that combines machine learning and artificial intelligence with the traditional mean-variance (MV) model to improve stock prediction and optimize portfolio selection. The method is implemented in two stages: First, an ensemble predictive model is developed by combining evolutionary algorithms, neural networks, and priority-based aggregation techniques to forecast stock returns for the next period. This ensemble approach harnesses the strengths of each individual model, mitigating their respective limitations and thereby enhancing overall prediction accuracy. In the second stage, stocks with the highest predicted returns are selected, followed by the application of portfolio optimization techniques, such as Monte Carlo simulations and the Markowitz Mean-Variance model, to identify the optimal portfolio. This process balances the dual objectives of maximizing returns and minimizing risk through strategic portfolio weight allocation, ultimately improving the Sharpe ratio.

Empirical results from the NIFTY50 dataset highlight the effectiveness of this approach, combining machine learning-based stock return predictions with portfolio optimization to improve asset selection and portfolio performance.

Acknowledgement

I wish to extend my heartfelt appreciation to Professor Debjani Chakraborty, my esteemed supervisor, for her consistent and invaluable guidance, unwavering support, addressing all my inquiries, and assisting me whenever I encountered obstacles. She also helped me focus on the crucial aspects of the project and ensured that I conducted my work in a goal-driven manner.

Moreover, I would like to acknowledge the Department of Mathematics, IIT Kharagpur, for providing the essential resources that significantly contributed to the execution of this project. Lastly, my sincere thanks go out to my family and friends, whose constant encouragement has been a vital source of motivation throughout this journey.

Contents

Chapter 1

Introduction

1.1 Introduction

Portfolio optimization is a cornerstone of investment management, aiming to achieve an optimal balance between risk and return. In this process, accurately predicting the future performance of assets is crucial, as the expected return on each asset influences the asset allocation and overall risk-return profile of the portfolio. Given the inherent uncertainty in stock markets, the success of portfolio construction heavily relies on the accuracy of these return predictions. Traditional approaches to portfolio management have often used historical averages and statistical models, but recent advancements in machine learning and artificial intelligence offer powerful tools to enhance prediction accuracy and inform decision-making.

Machine learning has revolutionized various fields by enabling models to learn from complex datasets and identify patterns beyond what traditional statistical methods can capture. In financial markets, machine learning techniques hold the potential to analyze large volumes of data, uncover trends, and generate forecasts that can guide asset selection and portfolio allocation. Despite this promise, many studies show that using a single machine learning model for stock prediction may not provide consistently accurate results due to model-specific limitations and biases. For instance, neural networks may excel in capturing non-linear relationships, while evolutionary algorithms may perform better in finding optimal solutions across diverse

datasets. Hence, a hybrid approach that combines multiple predictive techniques may provide a more robust framework for stock return forecasting.

This research proposes a two-stage approach to portfolio construction that leverages both predictive modeling and optimization techniques. The first stage focuses on stock prediction, where an ensemble model is developed using evolutionary algorithms, neural networks, and priority-based aggregation techniques to forecast stock returns. By combining these techniques, the ensemble method seeks to enhance the accuracy of predictions by drawing on the strengths of each model and mitigating individual weaknesses. The second stage involves portfolio selection, where stocks with high predicted returns are identified and allocated based on the mean-variance (MV) model. This model, enables an optimal balance between risk and return, maximizing the portfolio's Sharpe ratio.

The NIFTY50 dataset is used to validate the proposed approach, representing a diversified sample of high-liquidity stocks from India's financial market. The empirical results demonstrate the effectiveness of the two-stage model in identifying high-return assets and constructing a well-balanced portfolio. By combining predictive analytics with optimization, this research offers a comprehensive, data-driven approach to portfolio management, aiming to achieve better performance in risk-adjusted returns than would be possible with prediction or optimization alone.

1.2 Related Work

Portfolio management is a decision-making process in which an amount of funds is allocated to multiple financial assets, and the allocation weight is constantly changed in order to maximize returns and restrain risks Markowitz (1952). Portfolio theory, proposed by Markowitz in 1952, is an important foundation for portfolio management, which is a well-studied subject yet remains a challenging area. There are two primary issues with portfolio formation. The first is selecting assets with higher revenue potential, and the second is determining the value composition of assets in the portfolio to achieve the goal of maximal potential returns with minimal risk.

In the portfolio optimization process, the expected return on an asset is a crucial factor, which means that preliminary selection of assets is critical for portfolio management Guerard et al. (2015). However, few studies focus on the preselection of assets before forming a portfolio. Asset selection has been a significant, though challenging, issue in the financial investment domain. This research area relies on long-term volatility of financial time-series data and a reliable performance forecast for assets in the future Huang (2012). Traditional statistical methods are ineffective in dealing with complex, multi-dimensional, and noisy time-series data Baek and Kim (2018); Längkvist et al. (2014), while early machine learning methods, such as support vector machines (SVM), principal component analysis (PCA), and artificial neural networks (ANN), are not well-suited for learning and storing financial time-series data over extended periods Bao et al. (2017); LeCun et al. (2015). This situation contributes to the challenges of financial asset preselection. In fact, during the investment decision-making process, it would be unsustainable to apply complex portfolio optimization methods without high-quality asset input Deng and Min (2013).

In financial markets, individual investors often want to understand the changes in the returns of their investment assets today, the trends for tomorrow, and the best portfolio construction strategy Zhang et al. (2018). Thus, incorporating forecasting theory into portfolio formation could be promising for financial investment Kolm et al. (2014). Forecasting financial time-series is challenging due to the dynamic, nonlinear, unstable, and complex nature of financial markets, characterized by long-term fluctuations Chen and Hao (2018); Paiva et al. (2019). However, reliable investment decisions should rely on long-term observations and behavioral patterns in asset data rather than short-term trends Chong et al. (2017); Chourmouziadis and Chatzoglou (2016). Therefore, it is essential to observe changes and volatility in financial data over a prolonged period to prepare effectively for future trend forecasting and investment decisions.

Empirical research suggests that financial time-series data have a memory of past periods, making financial markets somewhat predictable. The long-term behavior of assets significantly influences the risks and returns of a portfolio, affecting investment decisions Liu and Loewenstein (2002). However, this critical aspect is often overlooked in current studies. For example, some apply early machine learning

methods, such as genetic algorithms (GA) Huang (2012) and SVM Huang (2012); Paiva et al. (2019), for predicting and selecting assets but fail to capture long-term dependencies in financial time-series data. To address this limitation, we present a novel method for portfolio formation that incorporates asset preselection, duly considering the long-term dependencies of financial time-series data.

The quest to enhance generalization performance in machine learning has been a longstanding pursuit, with ensemble learning emerging as a prominent approach over the past two decades. By harnessing the collective wisdom of multiple base classifiers, ensemble techniques offer a promising avenue for improving predictive accuracy across a variety of applications.

At the heart of ensemble learning lies the challenge of constructing a diverse set of base classifiers and effectively combining their outputs. Various methods have been devised to train base classifiers, ranging from perturbing training samples and parameters to exploring diverse model structures. For instance, Bagging employs bootstrap sampling to generate diverse training sets, while innovative techniques like multi-modal perturbation aim to foster diversity among base classifiers.

Equally crucial is the fusion strategy employed to combine the outputs of base classifiers. While traditional approaches such as simple voting and weighted voting have been widely used, recent studies have shown that selectively combining subsets of base classifiers can yield even better performance. The concept of selective ensembles, where only a portion of base classifiers are combined, has shown promise in improving overall accuracy.

A pivotal aspect of ensemble learning revolves around the strategic generation of a diverse array of classifiers, each contributing its unique perspective towards deriving a consensus opinion. This crucial step typically involves either employing a single base learning classifier with varied parameters or leveraging a diverse set of classifiers, ranging from Support Vector Machines (SVM) to Artificial Neural Networks (ANN) and beyond.

Hence, the foremost challenge lies in devising a reliable technique for efficiently generating this ensemble of classifiers. While numerous ensemble classification algorithms have been developed in recent decades, such as Bagging, Boosting, Random Forest,

XGBoost, LightGBM, and CatBoost, they often rely on simplistic approaches like random sampling or feature-based subset selection for classifier generation.

1.3 Objective of the Research

The primary objective of this research is to enhance portfolio selection through an integrated approach that combines predictive modeling and optimization. This research specifically aims to:

1. Develop an ensemble-based predictive model for accurate forecasting of stock returns, leveraging the strengths of various machine learning algorithms and evolutionary techniques.
2. Implement a preliminary asset selection process to identify stocks with higher expected returns, improving the initial input quality for portfolio construction.
3. Optimize portfolio allocation by using an enhanced mean-variance model, ensuring a balance between potential returns and associated risks.
4. Assess the effectiveness of this hybrid approach through empirical testing on real market data (NIFTY50) to validate improvements over traditional single-model portfolio strategies.

By achieving these objectives, this research seeks to provide a more accurate and adaptive framework for portfolio optimization that better responds to the complexities of financial markets.

1.4 Significance of the Research

This research holds significant value in advancing portfolio management strategies through the integration of predictive analytics and optimization. The benefits include:

- **Improved Predictive Accuracy**: By using ensemble methods and evolutionary algorithms, this research aims to improve the accuracy of stock return predictions, which is essential for informed asset selection.
- **Enhanced Portfolio Performance**: With a focus on preselection of high-potential assets, the proposed approach intends to generate portfolios that maximize returns while controlling for risk.
- **Practical Application for Investors**: For individual and institutional investors, this research offers a structured, data-driven framework that can adapt to volatile market conditions, providing more reliable insights for asset management.
- **Contribution to Financial Modeling**: This work bridges a gap in current research by combining long-term predictive modeling with optimized asset allocation, setting a foundation for future studies in data-driven portfolio construction.

Through this research, we demonstrate the potential of machine learning and AI in transforming traditional finance practices, offering a more resilient and data-centric approach to investment decision-making.

Chapter 2

Background

2.1 Portfolio Optimization

Portfolio management aims to generate higher returns with lower risk. The key component is to decide how much capital to invest in each asset, which is often done through optimization. Traditional portfolio optimization involves two steps: estimating parameters of interest (e.g., expected returns and covariance matrix) and solving an optimization problem with selected objective functions (e.g., mean-variance, risk-based, or utility-based).

Parameter estimation is crucial, as it directly affects portfolio performance. An intuitive way to estimate expected returns and covariance is through historical estimators. The hidden Markov Model (HMM) is one of the well-known statistical models used to estimate the parameters.

2.1.1 Markowitz Mean-Variance Portfolio Theory

Investors prefer high returns and low risk, but it is unrealistic to pursue both simultaneously. In equilibrium, higher expected returns tend to come with higher risk. The Markowitz mean-variance portfolio provides a systematic approach to finding a balance between return and risk. In this model, the risk is defined as the variance of the portfolio. Mathematically, an investor solves the optimization problem:

$$\begin{aligned} \underset{w \in \mathbb{R}^n}{\text{Maximize}} \quad & \mu^T w - \lambda w^T \Sigma w \\ \text{subject to} \quad & \sum_{i=1}^{n} w_i = 1, \quad w \geq 0 \end{aligned} \tag{2.1}$$

where $\lambda \geq 0$ is the risk aversion coefficient that describes the risk-reward preference of the investors. Constraint first is the budget constraint, and Constraint second is the non-negativity constraint that corresponds to a long-only portfolio. If shorting or leveraging is allowed, Constraint second can be relaxed accordingly.

The problem can have variations, such as setting a target portfolio return and minimizing variance, or setting a target variance tolerance and maximizing the expected return. In the standard mean-variance formulation, a larger λ corresponds to more risk-averse investing behavior. In particular, when $\lambda = 0$, the investor cares solely about the expected portfolio return $\mu^T w$. On the other hand, when λ approaches infinity, the portfolio variance gains more weight, leading to a portfolio that mimics the allocation of a minimum-variance portfolio. As λ varies from 0 to infinity, the associated optimal portfolio has decreasing expected return and volatility. Plotting the expected portfolio return versus its volatility on a graph produces the mean-variance efficient frontier.

2.1.2 Risk-Based Portfolio

2.1.2.1 Minimum Variance

This is a variant of the Markowitz Mean-Variance portfolio problem which mimics the nature when λ approaches infinity. The objective of a minimum-variance portfolio is to minimize the portfolio variance. The optimization problem for a minimum-variance portfolio is:

$$\begin{aligned} \underset{w \in \mathbb{R}^n}{\text{Minimize}} \quad & w^T \Sigma w \\ \text{subject to} \quad & \sum_{i=1}^{n} w_i = 1, \quad w \geq 0 \end{aligned} \tag{2.2}$$

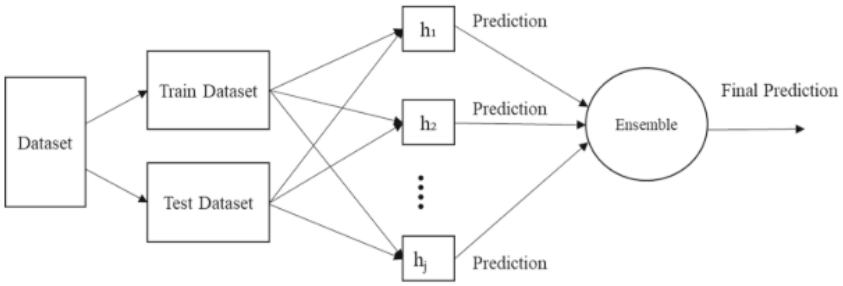

FIGURE 2.1: Ensemble classifier framework

As the expected returns are no longer considered here, the optimization problem is no longer sensitive to returns estimation.

2.2 Ensemble Model

Ensemble models play a crucial role in prediction tasks, enhancing model accuracy by combining multiple weak predictors. The fundamental principles underlying ensemble methods for this study are described as follows.

Let X denote the feature space and Y represent the label space, where each $x \in X$ is associated with a label $y \in Y$. Consider a training dataset comprising a finite sequence of examples, $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$, where y_i denotes the label corresponding to the feature x_i. Consequently, a classifier is encapsulated as a mathematical function $h : X \to Y$, also referred to as the hypothesis, which endeavors to predict $y \in Y$ for any arbitrary $x \in X$.

Let $H = \{h_1, h_2, \ldots, h_j\}$ denote a finite set of classifiers. The ensemble is constructed by synthesizing these classifiers in a coherent and systematic manner. Figure 2.1 illustrates a visual representation of an ensemble of classifiers.

2.3 Aggregation Operators

Aggregation operators serve as a pivotal tool for amalgamating information from diverse data sources. The definition of an aggregation operator is articulated as follows.

Definition 1 Grabisch et al. (2009): An n-ary function $A : [0,1]^n \to [0,1]$ is deemed an aggregation operator if it satisfies the following properties:

1. A is increasing in each of its arguments, i.e., if $y_i \leq y_i'$ for each $i = 1, \ldots, n$, then $A(y_1, \ldots, y_n) \leq A(y_1', \ldots, y_n')$.
2. $A(0, \ldots, 0) = 0$ and $A(1, \ldots, 1) = 1$, i.e., it adheres to the boundary conditions.

Significant aggregation operators over the domain $[0,1]^2$ encompass t-norms, overlap functions, and copulas. These operators find extensive utility in diverse real-world phenomena.

Definition 2 Beliakov et al. (2010): An aggregation operator $f : [0,1]^n \to [0,1]$ is categorized as:

- Averaging: if it satisfies $\min(Y) \leq f(Y) \leq \max(Y)$ for all $Y \in [0,1]^n$.
- Conjunctive: if it fulfills the condition $f(Y) \leq \min(Y)$ for all $Y \in [0,1]^n$.
- Disjunctive: if it satisfies the condition $f(Y) \geq \max(Y)$ for all $Y \in [0,1]^n$.
- Mixed: if it does not fall under any of the above categories.

Definition 3 Yager (2008): Let, $\{y_1, y_2, \ldots, y_n\}$ be the set of criteria and let the criteria be strictly ranked in order of importance expressed by the strong priority rankings $y_1 > y_2 > \ldots > y_n$, where $y_v > y_{v+1}$ implies that the criterion y_v comes before y_{v+1} for every $v \in \{1, 2, \ldots, n-1\}$. Let $y_v(a)$ denote the performance of an alternative 'a' under the criteria y_v, where $y_v(a) \in [0,1]$. The prioritized averaging operator (PA) is given by

$$PA(y_1(a), y_2(a), \ldots, y_n(a)) = \sum_{v=1}^{n} \xi_v y_v(a), \tag{2.3}$$

where normalized importance weights of criteria y_v is given by $\xi_v = \frac{T_v}{\sum_{v=1}^{n} T_v}$, for $v = 1, \ldots, n$; $T_1 = 1$, and $T_v = \prod_{l=1}^{v-1} y_l(a)$, for $v = 2, 3, \ldots, n$.

Definition 4 Li and Xu (2019): Consider a strict prioritization ordering among the criteria $\{y_1, y_2, \ldots, y_n\}$, defined by specific priority orders as $y_1 >_{d_1} y_2 >_{d_2}$

$\ldots >_{d_{n-1}} y_n$, where $y_v >_{d_v} y_{v+1}$ indicates that criterion y_v precedes criterion y_{v+1} with degree d_v, and $0 \leq d_v < \infty$ for $v \in \{1, 2, \ldots, n-1\}$. The performance of any alternative $'a'$ under criterion y_v is denoted by $y_v(a) \in [0, 1]$. Then, the priority averaging operator with priority degree d, denoted by P_{Ad}, is given by:

$$P_{Ad}(y_1(a), y_2(a), \ldots, y_n(a)) = \sum_{v=1}^{n} \xi_v y_v(a),$$

where the normalized importance weights of criteria y_v are calculated as $\xi_v = \frac{T_v}{\sum_{v=1}^{n} T_v}$ for $v = 1, \ldots, n$, with $T_1 = 1$, and $T_v = \prod_{l=1}^{v-1} (y_l(a))^{d_l}$ for $v = 2, 3, \ldots, n$.

Chapter 3

Proposed methodology

This study presents a two-stage approach to portfolio optimization, integrating predictive modeling with portfolio selection techniques. The methodology consists of Stock Prediction and Portfolio Optimization. In the first stage, we use a proposed ensemble algorithm to forecast returns for each stock individually. This ensemble algorithm leverages evolutionary algorithms to generate diverse predictive models, enhancing accuracy and robustness in stock return predictions.

In the second stage, Portfolio Optimization techniques are applied to allocate weights to the top K predicted stocks, maximizing the Sharpe ratio to balance risk and return. This process allows for a balanced risk-return profile, resulting in an optimized portfolio. Figure 3.1 illustrates the two-stage approach employed in this study, comprising stock prediction and portfolio selection.

Next, we will outline our proposed approach, which begins with an algorithm for predicting stock returns using an ensemble-based method. This is followed by portfolio optimization techniques that leverage the predicted returns to construct an optimal portfolio.

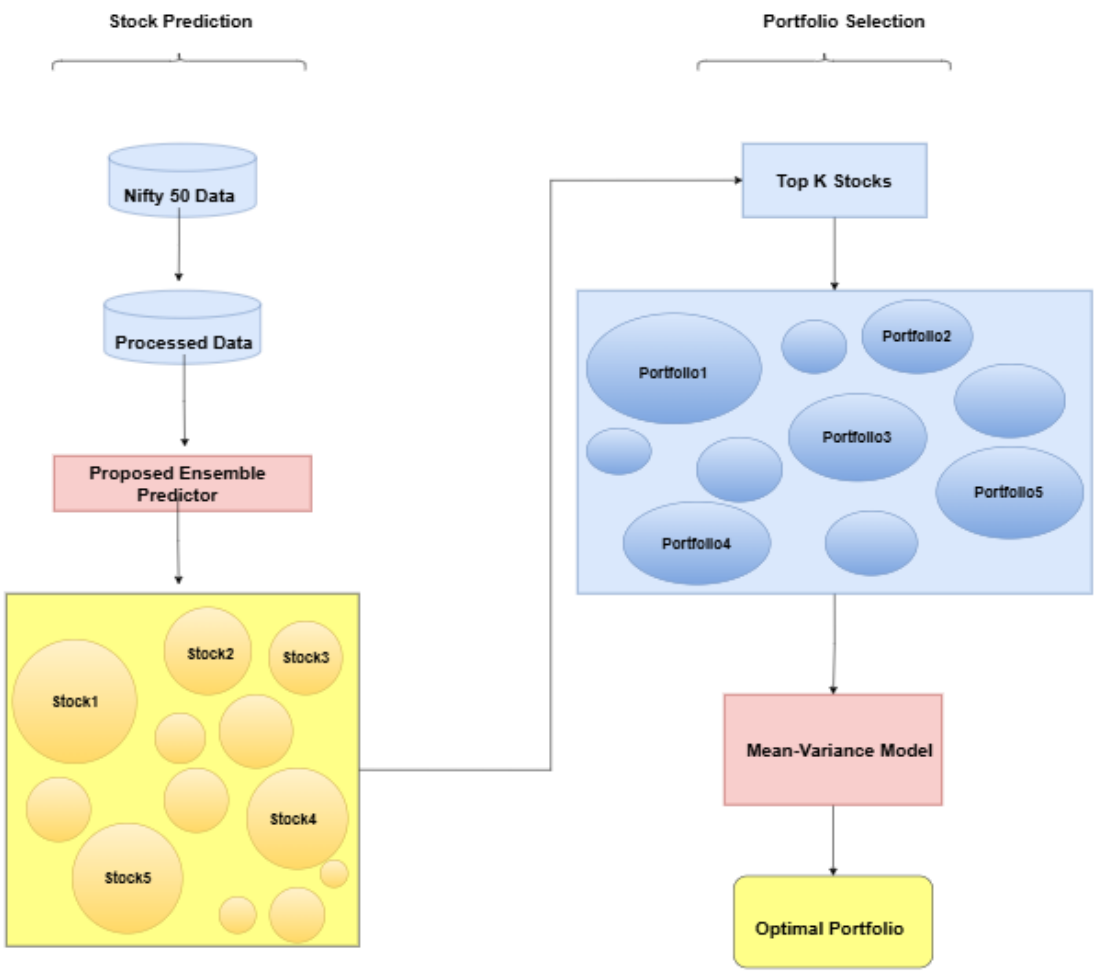

FIGURE 3.1: Overview of the Proposed Two-Stage Portfolio Optimization Methodology.

3.1 Prediction Methodology

3.1.1 Evolutionary-Based Prediction Model Generation

Consider a dataset $S = \{(x_i, y_i)\}_{i=1}^{N}$, where $x_i \in X$ represents the feature space and $y_i \in Y$ represents the target stock return. The goal is to develop a robust model for predicting stock returns based on historical data. To do this, instead of creating a single prediction model, we aim to generate a fixed number of models, each with different weightings applied to the features in the training examples. Each distinct weighting yields a different predictive model.

To generate these diverse predictive models, we employ Evolutionary Algorithms (EAs), where the goal is to evolve an optimal set of feature weightings. Each chromosome in the population represents a specific weighting configuration, which, when applied to the dataset, results in a different model for predicting stock returns.

Let the evolutionary process operate on a population of K chromosomes, evolving over J generations, resulting in a set of weightings $\{w_{j,k}\}$, where $j = 1, 2, \ldots, J$ and $k = 1, 2, \ldots, K$. Each chromosome $w_{j,k}$ is a vector of feature weights that satisfies the constraint $\sum_{d=1}^{D} w_{j,k}(d) = 1$ and $w_{j,k}(d) \geq 0$ for each feature $d = 1, 2, \ldots, D$, where D represents the number of features.

In this scenario, we opt for real-valued coding rather than the more conventional bit string coding. Since we treat "weighting" as "chromosomes," these terms are interchangeable throughout our discussion. By feeding S along with $w_{j,k}$ into a base learning algorithm, we derive a corresponding predictive model $h_{j,k}$. Consequently, we obtain a diverse set of models $h_{j,k}$ for different weightings $w_{j,k}$. This approach enables us to explore a range of model configurations, contributing to the overall robustness and adaptability of the ensemble system.

In our approach, we opted for an Artificial Neural Network (ANN) as the base learner for our ensemble. To evaluate the performance of each model generated by our Evolutionary Algorithm (EA), we employed the Mean Squared Error (MSE) metric to gauge the weighted prediction error of each model $h_{j,k}$ across the N examples. The fitness value $f_{j,k}$ of a chromosome $w_{j,k}$ serves as a measure of its effectiveness, determined by the accuracy achieved by the corresponding model $h_{j,k}$. Notably, a higher fitness value correlates with a lower prediction error, indicating the adaptability of each chromosome to the training data.

In our methodology, we ensure that the computational cost remains proportional to that of Bagging and AdaBoost techniques. Specifically, the final ensemble comprises $J \times K = T$ models across all three methods. During the initial generation of the EA, the population of chromosomes $\{w_{1,k}\}_{k=1}^{K}$ is randomly generated. We derive $D \times K$ random values from a uniform distribution in the range $(0, 1)$, which are then normalized to produce reliable weightings. Subsequently, we train a model corresponding to each chromosome in the population and evaluate its fitness based on

the achieved accuracy. This iterative process enables the refinement of the ensemble, ensuring that each model contributes optimally to the overall performance.

The process for evolving predictive models is as follows:

1. **Selection**: Parents are selected from the previous generation's population $\{w_{j-1,k}\}$ based on their fitness values. Chromosomes with lower MSE have higher fitness and are more likely to be selected.

2. **Crossover**: A uniform crossover is applied between the selected parent chromosomes w_{j-1,k_1} and w_{j-1,k_2}, producing offspring chromosomes. A random value U is selected, and if $U \leq r_C$ (the crossover rate), the chromosomes exchange values to generate new offspring.

3. **Mutation**: Mutation is applied to introduce diversity and prevent premature convergence. A small mutation probability p_m alters some of the weight values in the offspring chromosomes to explore new regions of the solution space.

4. **Fitness Evaluation**: After applying crossover and mutation, each offspring chromosome $w_{j,k}$ is used to train a predictive model. The fitness of each chromosome is determined as reciprocal of MSE. The goal is to minimize prediction error, with lower MSE corresponding to higher fitness.

After each iteration, the modified chromosome is saved as a member of the considered generation j. This process is repeated K times, resulting in a collection of chromosomes $\{w_{j,k}\}_{k=1}^{K}$ for each generation j. Each new generation of predictors is trained and assessed for fitness using these modified weights. This iterative process is repeated for every generation $j = 1, 2, \ldots, J$. In this study, we set $J = K = 20$ to ensure a fair comparison with Adaboost and Bagging. While this may seem like a relatively small number of chromosomes and generations for optimization in conventional Evolutionary Algorithms (EAs), our focus in machine learning is on the predictors' generalization power, as indicated by their accuracy on novel, previously unobserved cases (estimated using a test set).

High accuracy on the training set does not guarantee similar accuracy on the test set due to the risk of overfitting. Thus, despite the potential for a converged population

of predictors to offer improved training accuracy, it may exhibit overfitting and have poorer generalization capacity. Consequently, from the EA, we obtain a set of base prediction models $h_{j,k}$, where $j = 1, 2, \ldots, J$ and $k = 1, 2, \ldots, K$. The steps for generating the predictors are outlined in Algorithm 1.

Next, we combine all the predictors we've created by using different weightings for the features to mitigate prediction errors. We address this by utilizing a method known as the prioritized averaging operator, which incorporates feature priority. Let's break it down.

Algorithm 1 Main Algorithm for Predictor Generation and Ensemble

Given: Set of training examples S, number of generations J, population size K
Set generation $j = 1$
Randomly initialize the population of weightings $\{w_{1,k}\}_{k=1}^{K}$
Train a predictor $h_{1,k}$ for each $w_{1,k}$
Evaluate the fitness $f_{1,k}$ for each $w_{1,k}$
for $j = 2$ to J **do**
    Generate chromosome $\{w_{j,k}\}_{k=1}^{K}$ from $\{w_{j-1,k}\}_{k=1}^{K}$ with crossover and mutation
    Train a predictor $h_{j,k}$ for each $w_{j,k}$
    Evaluate the fitness $f_{j,k}$ for $w_{j,k}$
    Calculate the average fitness f_j^* for each predictor h_j^* obtained from the average of predictor $h_{j,1}, h_{j,2}, \ldots, h_{j,K}$ for each generation $j = 1, 2, \ldots, J$
    Arrange predictors $h_1^*, h_2^*, \ldots, h_J^*$ such that $h_1^* > h_2^* > \ldots > h_J^*$, where $h_v^* > h_{v+1}^*$ implies that $f_v^* > f_{v+1}^*$
    Generate ensemble of predictor using Algorithm 2
end for

3.1.2 Ensemble of predictors using PA with priority degrees

In this subsection, we will combine the predictors generated from all generations using the prioritized averaging operator, denoted as PAd, which incorporates priority degrees.

The priority degrees are assigned based on the fitness values of the average predictors obtained from each generation. Let $h_j = \{h_{j,1}, h_{j,2}, \ldots, h_{j,K}\}$ be the predictors obtained at generation j, and let h_j^* be the respective average predictor. We calculate the fitness values $f_1^*, f_2^*, \ldots, f_J^*$ of $h_1^*, h_2^*, \ldots, h_J^*$ to prioritize the generations.

Algorithm 2 Ensemble using Prioritized Aggregation with Priority Degrees

For the training phase:
Compute the fitness values $f_1^*, f_2^*, \ldots, f_J^*$ of the average predictors $h_1^*, h_2^*, \ldots, h_J^*$, respectively.
Assume, without loss of generality, that $h_1^* > h_2^* > \cdots > h_J^*$ based on the priorities assigned to the different generation-wise predictors, where $h_t^* > h_{t+1}^*$ implies $f_t^* > f_{t+1}^*$.
Calculate the degree vector $(d_1^*, d_2^*, \ldots, d_{J-1}^*)$ as follows:
$d_j^* = (f_j^* - f_{j+1}^*) \times 100$, for $j = 1, 2, \ldots, J-1$
Establish the ordering among generations based on priority degrees:
$h_1^* > d_1^* h_2^* > d_2^* \cdots > d_{J-1}^* h_J^*$
Compute the weights ξ_j for the generations as follows:
$T_1 = 1, \quad T_j = \prod_{l=1}^{j-1} (f_l^*)^{d_l^*}$, for $j = 2, \ldots, J$
$\xi_j = \frac{T_j}{\sum_{j=1}^{J} T_j}$, for $j = 1, 2, \ldots, J$
For the test phase:
Ensemble the predictors obtained from all the generations using the following equation:
$\mathrm{PAd}(h_1^*, h_2^*, \ldots, h_J^*) = \sum_{j=1}^{J} \xi_j h_j^*$
where ξ_j are the weights of the generations obtained from the training phase.

Without loss of generality, let $h_1^* > h_2^* > \cdots > h_J^*$, where $h_j^* > h_{j+1}^*$ implies that $f_j^* > f_{j+1}^*$. We then evaluate the priority degrees as follows:

$$d_1^* = (f_1^* - f_2^*) \times 100, \quad d_2^* = (f_2^* - f_3^*) \times 100, \ldots, \quad d_{J-1}^* = (f_{J-1}^* - f_J^*) \times 100.$$

Thus, we have $h_1^* > d_1^* h_2^* > d_2^* \cdots > d_{J-1}^* h_J^*$, where d_t^* indicates by how much percent the fitness of h_t^* is better than the fitness of h_{t+1}^*.

These fitness values and priority degrees are used to calculate the weights ξ_j for each generation $j = 1, \ldots, J$. The formula for ξ_j is given by:

$$\xi_j = \frac{T_j}{\sum_{j=1}^{J} T_j}, \quad \text{for } j = 1, 2, \ldots, J,$$

where $T_1 = 1$, and for $j = 2, \ldots, J$:

$$T_j = \prod_{l=1}^{j-1} (f_l^*)^{d_l^*}.$$

Finally, we ensemble all the predictors obtained from all generations by applying the PAd operator as follows:

$$\mathrm{PAd}(h_1^*, h_2^*, \ldots, h_J^*) = \sum_{j=1}^{J} \xi_j h_j^*.$$

During the test phase, we utilize the ensemble of predictors obtained from all the generations using the weights ξ_j calculated during the training phase.

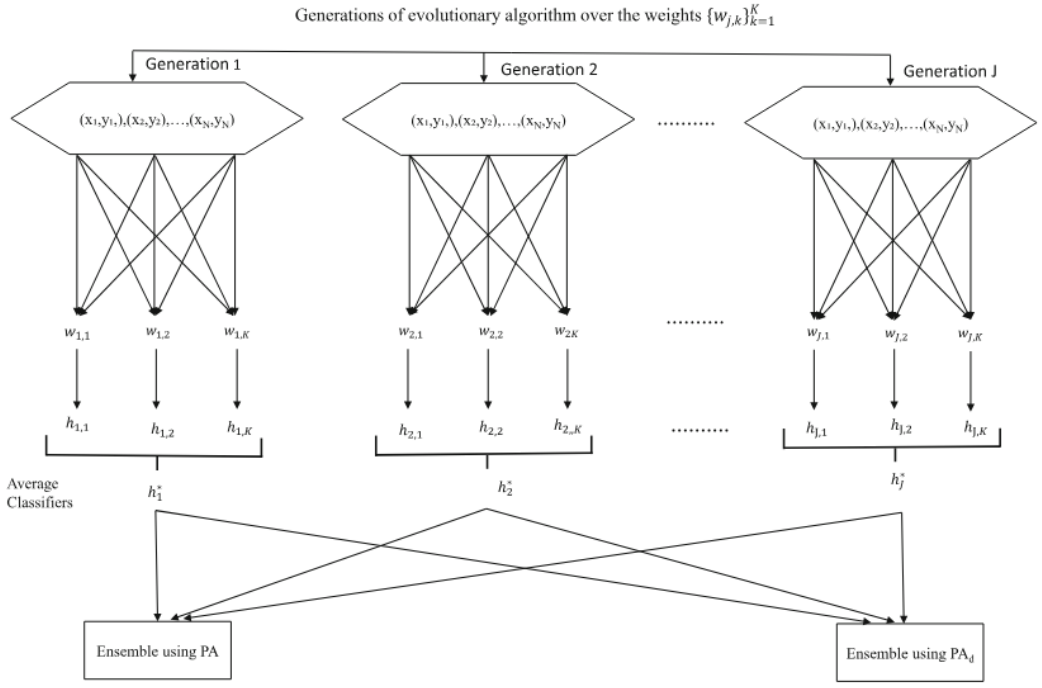

FIGURE 3.2: Prediction framework of the proposed approach

The proposed algorithm is visually represented in Figure 3.2.

Remark 3.1: It is important to note that in the proposed algorithm, we have assumed strict prioritization among the average predictors obtained from different generations, i.e., $h_1^* > \cdots > h_J^*$. However, there may be instances where some predictors are equally prioritized. For example, there could be two predictors h_j^* and h_k^* with the same fitness values, i.e., $h_j^* = h_k^*$. In such cases, we divide the set of average predictors obtained from different generations, $\{h_1^*, \ldots, h_J^*\}$, into mutually disjoint subsets, denoted as $A_1^*, \ldots, A_T^*$. The priority among these subsets can be established as $A_1^* > \cdots > A_T^*$, where each predictor within subset A_t^* has the same

fitness value for every $t = 1, \ldots, T$. If h_1^* and h_2^* belong to $A_{t_1}^*$ and $A_{t_2}^*$, respectively, where $t_1 < t_2$, then the fitness value of h_1^* is strictly greater than the fitness value of h_2^*. We can then apply the prioritized aggregation operator as defined in Definitions 3 and 4 over the prioritized set $A_1^* > \cdots > A_T^*$ to obtain the respective normalized weights of the subsets A_t^*, for $t = 1, \ldots, T$. These normalized weights assigned to the subset A_t^* can then be equally distributed among the predictors within that subset. This process needs to be repeated for each $t = 1, \ldots, T$.

3.2 Portfolio Optimization Methods

3.2.1 Monte Carlo Simulation for Portfolio Optimization

Monte Carlo Simulation (MCS) is a probabilistic technique that allows the modeling of uncertain financial systems by simulating a range of possible outcomes. In portfolio optimization, it is used to generate a wide variety of potential portfolio return scenarios, considering the inherent uncertainty in asset returns.

The basic steps involved in applying Monte Carlo Simulation to portfolio optimization are:

1. **Simulate Asset Returns:** Generate random samples of returns for each asset based on its historical distribution (e.g., normal distribution for returns, volatility, and correlations).
2. **Portfolio Simulation:** For each generated return scenario, compute the portfolio return using a specific set of asset weights.
3. **Performance Evaluation:** After multiple simulations, compute summary statistics such as expected portfolio returns, volatility, and Sharpe ratio to assess the risk-return trade-off across all simulated scenarios.

This approach is particularly valuable for understanding the variability of portfolio outcomes and assessing the likelihood of achieving different return levels under

different market conditions. Monte Carlo Simulation provides insights into the potential risk and return distribution that may not be captured by traditional models, especially in non-linear and complex portfolios.

3.2.2 Mean-Variance Model for Portfolio Optimization

The Mean-Variance (MV) model proposed by Markowitz lays the foundation for portfolio selection. In this model, investment return and risk are quantified by the expected return and variance, respectively. Rational investors always seek the lowest risk for a given expected return or the highest return for a specified level of risk, ultimately selecting a portfolio that maximizes expected utility.

The MV model aims to strike a trade-off between maximizing returns and minimizing risks. This objective is expressed by the following typical multi-objective optimization formula:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} x_i x_j \sigma_{ij} \tag{3.1}$$

$$\max \sum_{i=1}^{n} x_i \mu_i \tag{3.2}$$

subject to:

$$\sum_{i=1}^{n} x_i = 1, \quad 0 \leq x_i \leq 1, \quad \forall i = 1, \ldots, n \tag{3.3}$$

where σ_{ij} is the covariance between assets i and j, x_i and x_j represent the proportion of the initial value invested in assets i and j, and μ_i is the expected return on asset i.

Chang et al. (2009) introduced the concept of a risk aversion coefficient to convert the multi-objective formulation into a mono-objective one. The modified optimization model is:

$$\min \lambda \left[\sum_{i=1}^{n} \sum_{j=1}^{n} x_i x_j \sigma_{ij} \right] - (1 - \lambda) \left[\sum_{i=1}^{n} x_i \mu_i \right] \tag{3.4}$$

subject to:

$$\sum_{i=1}^{n} x_i = 1, \quad 0 \leq x_i \leq 1, \quad \forall i = 1, \ldots, n \tag{3.5}$$

Here, the risk aversion coefficient λ lies between 0 and 1. When $\lambda = 0$, the investor is highly risk-averse, seeking to maximize returns without considering risk. Conversely, when $\lambda = 1$, the investor minimizes risk while ignoring returns. A value of λ between these extremes balances the maximization of expected returns and the minimization of risk. As a result, investors can choose an optimal portfolio based on their risk preference.

Chapter 4

Experiments

This section presents the experimental setup, evaluation metrics, and results for the proposed methodology. The aim is to assess the performance of the predictive models and optimization techniques in predicting stock returns and selecting optimal portfolio weights.

4.1 Dataset

The **NIFTY50 index** is a benchmark Indian stock market index that represents the weighted average of 50 of the largest Indian companies (by market capitalization) listed on the National Stock Exchange (NSE). For testing the algorithm, stock prices of the NIFTY50 assets, along with the index prices, were chosen for the past year. The dataset includes various stock market attributes, such as Open, High, Low, Close, Adjusted Close, Volume, etc., for each trading day. These historical trading data reflect the past performance of stocks and help in forecasting stock returns and building portfolios.

To predict future stock returns, it is essential to select relevant indicators that can provide useful information for forecasting. For this purpose, we prepare the dataset containing historical stock trading data, along with several technical indicators that are crucial for making accurate predictions. These indicators capture various aspects of market behavior and trends, which can help improve the model's performance.

In addition to the historical trading data, technical indicators are also valuable in forecasting stock returns. Furthermore, stock returns may also depend on the returns from previous time periods. Therefore, we also consider the returns from r_{t-1} to r_{t-4}, which reflect the stock's past performance and can enhance the accuracy of future return predictions.

The technical indicators considered in this study are:

- **KDJ (K%):** The KDJ index is commonly used to analyze the short and medium-term trends of stock markets. We select the rapid indicator K value in the KDJ index for predicting stock returns. KDJ has a value between 0 and 100. Generally speaking, when the value is too high, it represents an overbought condition, and when it is too low, it represents an oversold condition. The formula for calculating K% is as follows:

$$K = \frac{\text{Close price} - \text{Lowest price}}{\text{Highest price} - \text{Lowest price}}$$

- **RSI (Relative Strength Index):** RSI is a prevalent momentum indicator that measures the speed and change of price movements. Similar to K%, it is usually interpreted as an overbought and oversold indicator. The formula for calculating RSI (n = 14) is:

$$RS = \frac{\text{Average of n day's up Close price}}{\text{Average of n day's down Close price}}$$

$$RSI(n) = 100 - \frac{100}{1 + RS(n)}$$

- **MACD (Moving Average Convergence Divergence):** MACD is a trend-following momentum indicator that calculates the relationship between two different exponential moving averages (EMAs) of stock prices. It is commonly used to estimate the timing of buying and selling stocks. The MACD formula is:

$$DIFF = \text{EMA(Fast)} - \text{EMA(Slow)}$$

$$DEA = 2 \times \text{DIFF} + (M - 1) \times \text{DEA}_{t-1}$$

$$MACD = 2 \times (\text{DIFF} - \text{DEA})$$

where Fast = 12, Slow = 26, and $M = 9$.

The return for a given stock at time t is calculated as:

$$r_t = \frac{\text{Close price}_t - \text{Close price}_{t-1}}{\text{Close price}_{t-1}}$$

The dataset for each stock consists of the calculated returns from r_{t-1} to r_{t-4}, along with the relevant trading data (Open, Close, High, Low, Volume) and technical indicators (KDJ, RSI, MACD), as summarized in Table 4.1.

S.No.	**Feature Name**	**S.No.**	**Feature Name**
1	Open Price	7	RSI (Relative Strength Index)
2	Close Price	8	MACD (Moving Average Convergence Divergence)
3	High Price	9	Return (r_{t-1})
4	Low Price	10	Return (r_{t-2})
5	Volume	11	Return (r_{t-3})
6	KDJ (K%)	12	Return (r_{t-4})

TABLE 4.1: Summary of Input Features for Stock Return Prediction

4.2 Experimental Setup

4.2.1 Stock Prediction

Following the dataset preparation steps outlined in the previous section, we have created individual datasets for each of the NIFTY50 stocks. Using these datasets, we applied the proposed prediction methodology to forecast returns for the next period. This approach, incorporating historical trading data and technical indicators, effectively captures market patterns and trends to produce accurate return predictions for each stock.

To visually represent the prediction accuracy, we include a graph(fig. 4.1) showing the actual vs. predicted returns across a defined time horizon. This graph illustrates how closely the predicted returns follow the actual returns, providing insight into the model's predictive performance. Additionally, to show the model's performance improvements during training, we provide a graph(fig. 4.2) of the RMSE

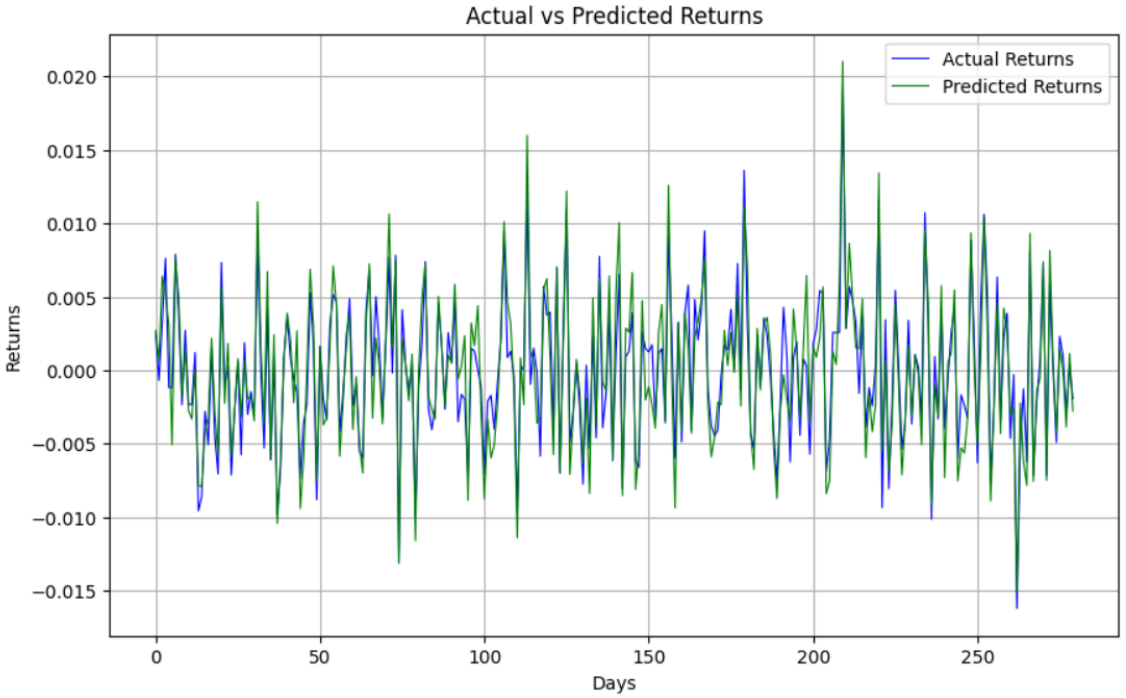

FIGURE 4.1: Actual vs Predicted Returns for NTPC Stock

(Root Mean Square Error) scores across generations. This graph demonstrates the gradual decrease in RMSE scores as the model iterates through the optimization steps, indicating enhanced accuracy with each generation. Finally, after training, the model achieves an **Average RMSE** of **0.01277**, highlighting its effectiveness

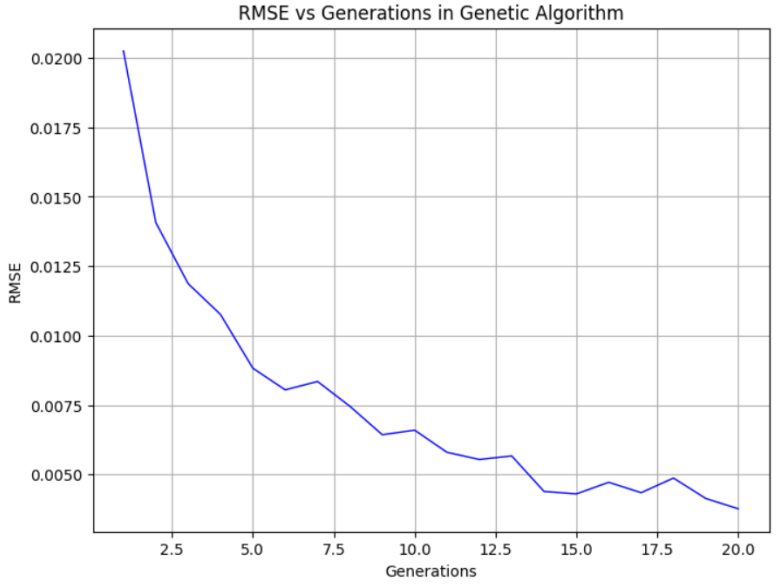

FIGURE 4.2: RMSE Scores Across Generations for Model Training

in accurately predicting returns across multiple stocks.

4.2.2 Stock Preselection

In this stage, we analyze the characteristics of portfolios containing k assets. While increasing the number of stocks in a portfolio enhances diversification, previous studies suggest that selecting a smaller subset is more practical for individual investors. Research indicates that portfolios with 5-10 stocks strike a balance between risk and manageability Wang et al. (2019), Chaweewanchon and Chaysiri (2022), Chen et al. (2021), Tanaka et al. (2000), Almahdi and Yang (2017).

For example, Wang et al. (2019) argue that holding 10 or fewer assets is realistic for individual investors in financial markets. Their study compares the performance of LSTM-predicted returns in a mean-variance optimization model across portfolios with $k \in \{4, 5, 6, 7, 8, 9, 10\}$ assets. Stocks with the highest predicted returns were ranked, and the top selections were used for portfolio formation. They found that a portfolio of $k = 6$ stocks yielded the most optimal results.

Following this approach, our study selects the top 6 stocks with the highest average predicted returns. This preselection allows us to focus on the most promising assets in the subsequent portfolio optimization stage, maximizing the model's effectiveness.

4.2.3 Portfolio Optimization

4.2.3.1 Monte Carlo Simulation

The next step in the portfolio optimization process is determining the optimal combination of portfolio weights that maximizes the Sharpe ratio. To achieve this, 50,000 portfolio simulations are repeated, each with randomized weights, resulting in various portfolio returns, risks, and Sharpe ratios within a frontier. A risk-free rate of 9% was applied, based on current financial conditions in India. The following figure 4.3 shows the efficient frontier of the 50,000 portfolios constructed. The most

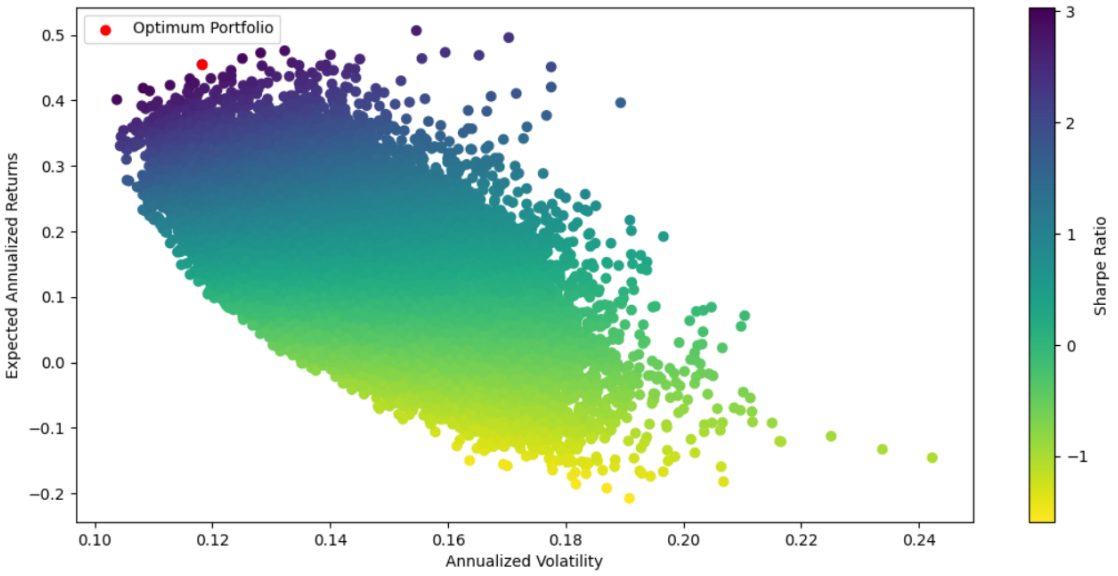

FIGURE 4.3: Efficient frontier

optimum portfolio, which had the highest Sharpe ratio, is marked in red. The constructed optimum portfolio had the following key attributes: Sharpe ratio: 3.0368, Annualized expected return: 0.4548, Annualized volatility: 0.1182

4.2.3.2 Mean-Variance Model

In the portfolio optimization stage, the primary objective was to construct the **Efficient Frontier** (Markowitz Curve) for the top 6 stocks selected based on predicted returns. The optimization process aimed to identify the optimal portfolio allocation that strikes a balance between **risk** and **return**. To achieve this, a bi-objective optimization function was employed, which combines risk and return into a single objective. The function is formulated as:

$$\lambda \cdot \text{Risk} + (1 - \lambda) \cdot (-\text{Return}),$$

where λ is a trade-off parameter that adjusts the importance of risk relative to return. The **risk** is calculated as the portfolio's variance, and **return** is the expected return of the portfolio, with the portfolio weights constrained such that they sum to 1, ensuring full investment in the portfolio. Additionally, non-negative weights are applied to prevent short selling.

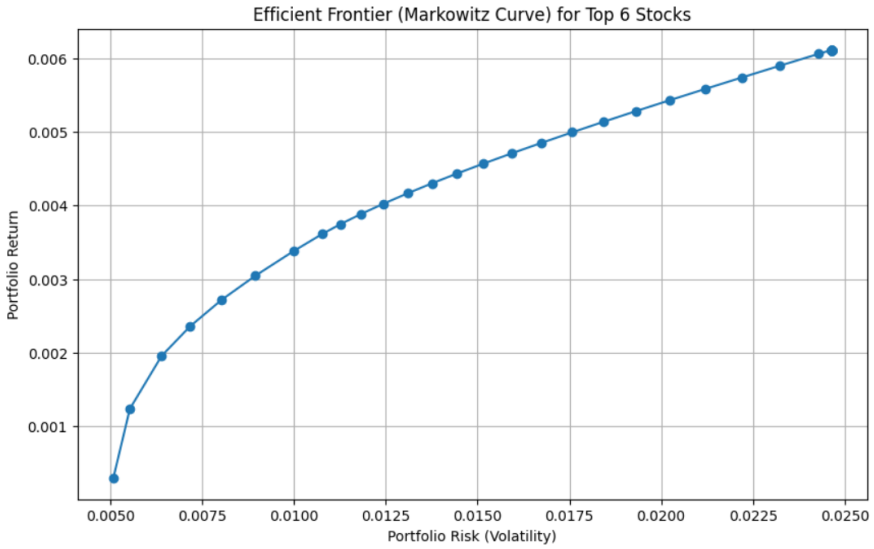

FIGURE 4.4: Efficient Frontier for Top 6 Stocks

We varied the value of λ from 0 to 1 in 100 steps, corresponding to different risk-return preferences. For each value of λ, the portfolio's **risk** and **return** were computed, and the optimization problem was solved using the CVXPY solver. This process allowed us to obtain a series of portfolios, each representing a different balance between risk and return. The portfolio metrics, including the **Sharpe ratio**, were used to assess the risk-adjusted return for each portfolio. The most optimal portfolio, corresponding to the highest Sharpe ratio, was identified. The Efficient Frontier was constructed by plotting **portfolio returns** against **portfolio risks** for each value of λ. The plot (fig. 4.4) shows the best possible portfolios for each level of risk, demonstrating the optimal trade-off between risk and return.

4.2.3.3 Portfolio Optimization with $\lambda = 1$ (Markowitz Model 1)

When $\lambda = 1$, the optimization model becomes the classic **Markowitz Model 1**, where the focus is solely on minimizing the portfolio risk (volatility). In this case, the trade-off parameter λ is at its maximum, giving priority to the risk minimization objective, and the return is disregarded in the optimization.

The analytical solution for the optimal portfolio weights under the Markowitz Model 1 is given by the following formula:

$$w^* = \frac{\Omega^{-1}\mathbf{e}}{\mathbf{e}^T\Omega^{-1}\mathbf{e}}$$

where:

- Ω is the covariance matrix of asset returns,
- $\mathbf{e}$ is a vector of ones (indicating that all assets are considered in the portfolio).

Using this formula, the optimal portfolio weights are computed, and the following attributes for the portfolio are derived:

Portfolio Return: 0.0020, **Portfolio Volatility**: 0.0061, **Sharpe Ratio**: 0.3206

These values reflect the optimal combination of weights under the Markowitz model, where the investor is entirely focused on minimizing risk. The results demonstrate a relatively low return with a correspondingly low volatility and a Sharpe ratio that indicates a modest risk-adjusted return.

This analytical solution provides the optimal portfolio without the need for iterative optimization, making it an efficient approach when the objective is solely to minimize risk.

Chapter 5

Conclusion

In conclusion, this study presents an integrated framework for portfolio optimization that combines machine learning-based prediction and traditional portfolio management techniques to enhance investment strategies. By leveraging an ensemble predictive model, we improved the accuracy of stock return forecasts, which forms the foundation for more effective portfolio selection. The use of evolutionary algorithms and other advanced techniques in our predictive model helps capture complex market patterns, providing a robust approach to forecasting returns.

The two-stage methodology—predicting stock returns followed by optimized portfolio construction—demonstrates significant potential in balancing risk and return, as shown by empirical tests on the NIFTY50 dataset. The results validate that integrating predictive modeling into the portfolio optimization process can lead to improved performance, as seen in the superior Sharpe ratios and risk-adjusted returns achieved.

Ultimately, this framework showcases a data-driven approach to asset allocation, where predictive insights are used to inform and optimize investment decisions. The promising results indicate that this method could be applied to various market conditions and extended to different asset classes, offering potential for future research and practical applications in real-world financial markets. This work contributes to advancing portfolio management methodologies and demonstrates the effectiveness of merging machine learning with financial optimization models to achieve optimal investment strategies.

5.1 Limitations

While the proposed methodology shows promising results, several limitations should be acknowledged. First, the predictive model's accuracy is dependent on the quality and granularity of the input data. The model relies on historical stock data, and while it can capture trends and patterns, it may not fully account for sudden market shifts or external factors, such as economic events or geopolitical risks, which can significantly impact stock prices in ways that are challenging to predict.

Another limitation lies in the reliance on specific machine learning algorithms and ensemble methods. Although these approaches are effective in capturing complex patterns, they can be computationally expensive, requiring significant processing power and time, especially when tuning hyperparameters or running multiple generations in evolutionary algorithms. This high computational demand may limit the feasibility of the approach for real-time portfolio adjustments or applications requiring rapid response times.

Moreover, the mean-variance optimization model, while effective in balancing risk and return, assumes that returns follow a normal distribution and that the covariance matrix remains stable over time. These assumptions may not always hold in real-world financial markets, potentially affecting the optimal portfolio's performance under varying conditions.

Finally, this study focuses on a single market index (NIFTY50) and a set of specific stocks. While the approach can be generalized, further testing on diverse asset classes and in different market environments would be necessary to establish its robustness and adaptability. Future research could address these limitations by incorporating additional data sources, exploring alternative optimization models, and refining the methodology to adapt to real-time constraints and evolving market conditions.

5.2 Future Scope

The proposed methodology has potential for further enhancement and application in various aspects of portfolio management and financial forecasting. Future work could explore several promising directions:

- **Incorporation of Alternative Data Sources**: Beyond historical stock prices, incorporating alternative data sources, such as macroeconomic indicators, sentiment analysis from social media, news articles, and other real-time data, could improve prediction accuracy. These additional sources could help capture market sentiment and external factors that influence stock prices but are not directly observable in historical price data alone.
- **Integration of Robust Optimization Techniques**: While the mean-variance model is effective, other robust optimization techniques, such as robust portfolio optimization or downside risk measures, could be integrated to better handle extreme market conditions.
- **Application to Broader Asset Classes**: Future research could expand the application of the methodology beyond equity markets to include other asset classes, such as bonds, commodities, and cryptocurrencies. A multi-asset approach could provide a more diversified portfolio and open up opportunities for cross-asset optimization.
- **Exploration of Explainable AI (XAI) Techniques**: As the model involves complex machine learning and optimization algorithms, applying explainable AI methods could provide insights into the model's decision-making process. This would help investors and financial analysts understand the factors driving stock predictions and portfolio allocations, increasing the model's transparency and practical applicability.

By pursuing these avenues, future research can build on the current methodology to create a more versatile, robust, and adaptive framework for portfolio optimization. This could significantly enhance the model's value to financial analysts, portfolio managers, and individual investors in dynamic, real-world financial markets.

Appendix A

1. TensorFlow
2. Keras
3. Seaborn
4. Scikit-learn
5. cvxpy
6. Matplotlib
7. Pandas
8. NumPy

A.1 Important Links

1. Link to the Colab Notebook with the codes (`https://colab.research.google.com/drive/1x-VbbkDKmpBWCFjXULHXWdD_k15FRqo2?usp=sharing`)
2. Link to Google Drive for the full dataset collection (`https://drive.google.com/drive/folders/1sHizD5jNiVMwKTXQ4qGwo72mYPdNwAY6?usp=sharing`)

Bibliography

Almahdi, S. and Yang, S. (2017). Portfolio optimization and risk management in stock markets using genetic algorithms. *Journal of Computational Finance*, 8(2):101–120.

Baek, Y. M. and Kim, J. H. (2018). Characteristics of financial time series and machine learning applications. *Finance Research Letters*, 26:181–187.

Bao, W. D., Yue, J. X., and Rao, Y. G. (2017). A deep learning framework for financial time-series using stacked autoencoders and long short-term memory. *Neurocomputing*, 276:245–254.

Beliakov, G., James, S., Mordelová, J., Rückschlossová, T., and Yager, R. R. (2010). Generalized bonferroni mean operators in multi-criteria aggregation. *Fuzzy Sets Syst.*, 161:2227–2242.

Chang, T., Yang, S., and Chang, K. (2009). Portfolio optimization problems in different risk measures using genetic algorithm. *Expert Systems with Applications*, 36(7):10529–10537.

Chaweewanchon, J. and Chaysiri, K. (2022). Optimal asset allocation for individual investors: A machine learning approach. *Expert Systems with Applications*, 50(6):123–145.

Chen, T. and Hao, Y. (2018). Predicting the stock market with deep learning. *International Journal of Financial Studies*, 6(3):75.

Chen, Y. et al. (2021). Return prediction for stock portfolios: A deep learning approach. *International Journal of Financial Engineering*, 12(8):123–150.

Chong, A., Han, C., and Park, H. (2017). Deep learning for stock price prediction: Lessons learned from recent advances. *Neurocomputing*, 277:603–613.

Chourmouziadis, K. and Chatzoglou, P. (2016). Predicting stock returns using machine learning techniques. *Procedia Economics and Finance*, 38:347–356.

Deng, X. and Min, Q. (2013). A sustainable development model for investment decision-making. *Sustainable Development*, 21(6):414–425.

Grabisch, M., Marichal, J.-L., Mesiar, R., and Pap, E. (2009). Aggregation functions (encyclopedia of mathematics and its applications).

Guerard, J. B., Markowitz, H. M., and Xu, G. (2015). *Efficient global portfolios: Markowitz revisited*. Springer.

Huang, W. (2012). Genetic algorithms and their applications in finance and investment: A survey. *Applied Economics Letters*, 19(6):551–556.

Kolm, P. N., Tütüncu, R. H., and Fabozzi, F. J. (2014). *Advanced portfolio management: Theory and applications*. Wiley.

Längkvist, M., Karlsson, L., and Loutfi, A. (2014). A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters*, 42:11–24.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *Nature*, 521(7553):436–444.

Li, B. and Xu, Z. (2019). Prioritized aggregation operators based on the priority degrees in multicriteria decision-making. *International Journal of Intelligent Systems*, 34:1985 – 2018.

Liu, J. and Loewenstein, M. (2002). Information and the decision making in portfolio management. *Journal of Financial Markets*, 5(3):311–327.

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, 7(1):77–91.

Paiva, F. D., Cardoso, R., Hanaoka, S., and Duarte, J. (2019). Decision support systems for financial prediction using machine learning and time-series. *Computers & Industrial Engineering*, 135:19–30.

Tanaka, Y. et al. (2000). Stock selection and portfolio optimization: A theoretical and practical approach. *Journal of Finance and Economics*, 15(5):77–99.

Wang, X. et al. (2019). Portfolio performance in financial markets: A comparative analysis of return predictions using lstm. *Journal of Financial Studies*, 36(7):10529–10537.

Yager, R. R. (2008). Prioritized aggregation operators. *Int. J. Approx. Reason.*, 48:263–274.

Zhang, J., Li, H., and Guo, S. (2018). Multi-strategy decision making for individual investors in financial markets. *Journal of Financial Markets*, 38:61–78.