DEVELOPING ALGORITHMIC TRADING STRATEGIES USING MACHINE LEARNING METHODS

RAJEEV KUMAR SINGH

Final Thesis Report

JULY 2022

Dedication

This report is dedicated firstly to god almighty for giving me energy and good health to continue the hard work.

Then I dedicate this report equally to my thesis supervisor Mr Praveen Chougale whose guidance and support had made the research come through and small family who gave me support,energy and motivation to strive for better things in life.

Abstract

There has been an attempt for decades to intelligently and safely operate in Equity markets. Equity Markets are a vital component of market economy. Traditionally various techniques like Mean Variance Optimisation have been tried by portfolio managers for portfolio optimization. However with the advent of Machine Learning and Artificial Intelligence in this field there have been multiple strategies that have become popular.

This research aims at exploring the existing portfolio optimisation techniques and compare them with multiple machine learning methods.A pipeline of models was

Used to forecast stock prices from ten representative stocks from Nifty50.

Stock portfolio was constructed as per the forecasted returns for the next day with weights assigned to each stock in proportion to their return.This portfolio with ML based forecast was compared to with a portfolio where mean variance optimisation based weights were used.These retults were compared with actual retrurns.

First simulation for data from 26/10/2017 till 09/06/2022 resulted in ML based method predicting stock direction 9 out of 10 stocks correctly whereas MVO weight based method made correct prediction only 50% times for daily average returns.cumulative error for ML based method was 1.5% whereas for the MVO weight based method was 6.5%

Simulations for before covid and after covid data also showed ML based method perfoming better that MVO based method in stock direction prediction and lower cumulative errors.

TABLE OF CONTENTS

LIST OF FIGURES

List of Tables

CHAPTER 1

INTRODUCTION

1.1 Background of the Study

Equity markets are popular as well as risky investment hubs for both retail and institutional investor. Certain major Stock exchanges of the have market capitalization in Trillions of Dollars. Given the direct and indirect involvement of masses in equity market they are an important Area of Study.

Investments in the stock markets have been on the rise due to competitive economy. Mutual Funds, which put certain percentage of their investment in stock markets, have become popular with small investor recently. If we compare the data from Statistical Abstract of US from previous years we can see that not only direct ownership but also indirect ownership in form for Retirement accounts rose from 39% in 1992 to 52% in 2007.These suffice to say given these trends stock markets will increasingly become important parts of our financial investment and planning.

The growth in Information and Technology specially digital connectivity have provided platforms in this Era of globalisation where world markets are gradually being integrated in their movement, global citizens have access to all the market and opportunities around the globe.In this vein this area has witnessed keen studies early on, seminal paper of Harry Markowitz provided a framework for optimum return calculation that balanced risk and returns.Now traditional and Modern Portfolio optimisations theories and a lot of Algorithmic trading models try to solve the above problem.Multiple efforts have been made to come up with accurate prediction of future based on historical data study through Statistical methods and analysis like Mean Variance optimisation.

This research is aimed at developing Machine learning(ML) based alogorthmic trading methodology for day trading.Here in historical data is enriched calculated features.These forecasted values are used to create a portfolio which is adjusted daily as per expected returns.The returns from ML based method is compared with Mean variance optimization(MVO) based returns.The study also compares the performance of Traditional method with ML method in two timeframes separated by the covid incident.

1.2 Problem statement

Algorithmic trading also referred to as portfolio optimisation is a problem that has attracted much research in past and in current times. Algorithmic trading refers to the automation of trading

strategies that require minimal human intervention during the trade day.This is a wide term that encompasses both the hardware and software related to stock trading.Given multi-dimensional problem space in stock trading mathematical theorisation and practical solution of this problem for maximum profit has been a continuing endeavour. In this light various methods have been developed over time.During current times machine learning methods have gained popularity.Efforts have been made to model the stock trading problem as real world as possible This study aims at creating ML models for day trading.

The aim of this research is to develop Algorithmic trading strategies using Machine learning frameworks, improving upon the existing methods to generate profit and provide early warning signal to save losses under crisis situations like the Credit Crisis of 2007,Market crash During COVID-19.

1.3 Aims and Objective

The research Objectives arrived in congruence with the aim of research are below:

- Data Collection from Yahoo Finance and manipulating it to arrive at important features for analysis.
- Performing analysis to come up with representative list of stocks.
- Comparing Traditional and Current Portfolio optimization techniques
- Creating Algo-Trading models in Trading Environment based on Machine learning Algorithms.
- Evaluating the efficacy of Models through popular Testing Strategies and Mertics inclding the scenarois arising of crisis like COVID.

1.4 Scope of the study

The Research is aimed at analysing and developing models based on stock market data.

Ten representative stocks from Nifty50 were chosen to develop a diversified portfolio.

Traditional portfolio optimisation technique,MVO, will be compared with ML based stock price forecasting models developed during the study.Returns will also be compared over time periods before and after Covid-19.

1.5 Significance of study

Equity Markets are an important component of market economy. It's a meeting ground for Buyers and sellers of different capacities be it retail whose capacity is few thousand Dollars to institutional players whose capacity is in millions.

The promise of reward at the market also comes with a huge risk of losses. Study of these markets have been more than half a century old. One of the seminal works in this area of Markowitz is appreciated because it tries to balance the risk in process of maximizing reward.

Current research aims at bringing more intelligence in the process through use of traditional as well as advanced Mathematical and Machine Learning models.The Research is aimed at developing ML based models to bring in better decision for market positions.It will be tested for scenarios that cause loss of life savings and frustration of millions of investors during financial crisis situations.

1.6 Structure of the study

This study is aimed at portfolio optimisation using machine learning.This section of the report is followed literature review which deals with the study of current research work in portfolio optimisation area.Literature review tries to identify the gaps in traditional portfolio optimisation technique and highlighting the work to overcome those gaps.A special emphasis is given to machine learning based methods which the core theme of this study.Chapter 3 deals with a high level explanation of tools,techniques and methods involved in the study.It highlights how data is extracted,features created,MVO method basics and ML models' theoretical premise.Chapter 4 deals with detailed Implementation of ML models and MVO methods.Chapter 5 contains the results of simulations done with optimised portfolio and their evaluation.Chapter 6 contains conclusion of the study and future recommendations.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This approach illustrated relation between “Expected returns and variance of returns” culminating into the concept of “Efficient Frontier” to give a desired combination of Expectation of return and Variance.(Markowitz, 1952).

The approach leads to concentration of weights in few stocks that leads to high risk in investment.This lack of risk diversification is fraught with failures during occurrence of extreme events like market falls.(Min *et al.*, 2021).For stocks with sparse historical data estimation mean and variance of stocks could have high errors resulting in high risk.(Jobson and Korkie, 1980).For large sample size the computational complexity increases with temporal overhead.MVO approach discourages portfolio selection with increase in returns or variance.(Weng *et al.*, 2020).Markowitz’s modern portfolio theory makes assumptions which are breached in real time like market being perfect without Short sales or shares can be divided into fractions exempted from taxes and transaction costs.(Simos, Mourtas and Katsikis, 2021).MVO model is also criticised due to absence of real word scenarios like boundary and cardinality constraints.(Kalayci, Polat and Akbay, 2020).

Above limitation, needs of changing times coupled with exponential growth in computational power have led many approaches that have solved the problem of portfolio optimization in a much better manner.

2.2 Developments in portfolio optimization techniques.

Developments in portfolio optimization techniques can be studied through two dimensional framework one focussing on the Problem and other on Methodology.(Doering *et al.*, 2019)

Problem specific developments relate to more real, accurate problem formulation, diversified definition of risk,use of hybrid simulations coupled with machine learning.

Additional constraints are added the problem and objectives.(Doering *et al.*, 2019)

On the other hand methodology specific developments concentrate on improving computational times and methodological complexity,computational time increase exponentially when the problem solving becomes multi-objective.These focus on meta-heuristic approaches, increasing

computational capability through better hardware support,distributed and parallel computing.Methodological complexity related development focuses on lacunae of population based metahueirstics by proposing single point metahueristic approaches.Hybridiation of methods is an important trend which can be further analysed given its richness and diversity.(Doering *et al.*, 2019)

One other important classification for analysing developments that requires mention on the trading strategy side is passive and active trading strategy depending on the investor behaviour of replicating the index or trying to actively beat it.(Shi *et al.*, 2022).More efforts have been made on beating the market through active trading strategies.

The number of such development and strategies is vast enough to be beyond the scope of discussion in this paper.Certain techniques have been discussed following text to enrich the perspective on this area.

2.3 Traditional Machine learning based Methods

Machine learning based methods have proved useful overtime to solve multi-dimensional prediction problems.Stock data contain time series data in various forms like stock open/close price,volume,return,volatility.(Enke and Thawornwong, 2005).Stock price movement over the temporal dimension provide necessary features required to train machine learning models. Variables like technical indicators,financial variables and macro-economic variables are considered the most important variables.(Tsai and Hsiao, 2010).However over the years the Portfolio optimization as an area of study has got lot of directed research over the decades.

Given the utility and commercial viability of portfolio optimization there has been renewed and continued research through cutting edge tools, techniques.

Treatment of portfolio optimization from Mean Variance optimization(MVO) approach which logically divided the problem into two parts viz. 1.Empirical Observation of past performance 2.Belief about the future performance and optimal choice in that respect.(Markowitz, 1952).

exponential growth in social media and web based content site has been found to be an important influencer of stock market.(Li *et al.*, 2020).The search more data new data influencer is an on-going one.

Ensemble models like Adaptive Boosting(Adaboost), Gradient boosted decision trees(GBDT) and Extreme Gradient boosted tress(XGBoost)(Chen *et al.*, 2021) are both as for classification and regression problem solving.

The scheme of use includes use of these models as binary classifiers to predict direction of market.Experiments have resulted in higher returns than market.(Nobre and Neves, 2019).Use of Extreme Gradient Boosted trees to classify stock trend.(Dey *et al.*, 2016).XGboost model has predominantly been used for classification some experiments have also used XGBoost as regression in stock price forecasting.(Chen *et al.*, 2021).In this study XGBoost model's hyperparameters are optimised using a meta-heuristic algorithm called firefly.The resulting model is a hybrid model namely IFAXGBoost.(Chen *et al.*, 2021)

Another Tree based ensemble model Random forest has been used to predict the stock price direction.Using technical indicators like relative strength indicator(RSI),stochastic oscillator to predict increase or decrease in price of stock after 'n' days.Being a classification problem metric like precision and recall has been used.(Khaidem, Saha and Dey, 2016)

Another study that add more classes to vanilla binary classification problem was conducted by Lohrmann and Luukka,random forest model was used on S&P500 for predicting intraday returns.The four classes were 'strong positive', 'slightly positive', 'slightly negative' and 'strongly negative' returns.136 variables ranging from technical indicators,fundamental indicators to engineered features were selected chosen which were later pruned using fuzzy similarity entropy measure(FSAE)(Lohrmann *et al.*, 2018).Multiple strategies involving permutation and combination of long(buy) and short(sell) were compared with the benchmark of long(buy) and hold.The study indicated that classes with strong signals to buy and sell were the most accurately predicted than those with slight buy and sell signals.(Lohrmann and Luukka, 2019).Study combining multiple feature selection methods like principal component analysis(PCA),genetic algorithm(GA) and decision tress(CART) has shown how intersection of PCA and GA and multintersection of PCA,CART and GA for feature selection results in accuracy in the range of 79%.(Tsai and Hsiao, 2010).Bayesian networks have been used to determine the influence of one closing market on the other opening market over 24 hour and 48 hour window around the globe keeping the main index of Sao Paulo stock exchange iBOVESPA as the pivot.This research proves that the markets around the world are interrelated.Results from the research provide mean accuracy of 71%.(Malagrino, Roman and Monteiro, 2018).

2.4 Deep Learning Models

Deep Learning models are a popular choice for predicting the direction or value stocks.In the recent times technological advancements have led to increased use of Deep learning models due to two primary reasons:first being the availability of enormous volumes of structured and unstructured data at least latency,second being the increase in compute power given advance processing systems like Graphical processing unit(GPU),cloud computing to state a few.

Deep learning models tend to solve for the lacunae of stock data that pose challenges traditional time series models,they work well on non-stationary,non-linear,noisy data.(Niaki and Hoseinzade, 2013).Most popular deep learning models could be classified into three categories viz.standard models,their variants,hybrid models and other models.Popular standard models include feedforward neural network,convolutional neural network and recurrent neural network.Hybrid models are a combination of one or more standard deep learning models or a combination of standard deep learning models and traditional linear models.Others catergory has models like generative adversarial network, transfer learning.Re-inforcement learning models.(Jiang, 2021).

Feedforward neural networks have been popularly used in Artificial Neural Networks(ANN) and Deep Neural Networks(DNN).ANN was used in predicting the closing price of PETR4, traded on BM&FBOVESPA, feedforward multilayer perceptron composed of three layers input,hidden and output layer was used.Best performing model had a time window size of 3 with a prediction of change direction(POCID) accuracy on test set at 93.62% and performed much better than the baseline logit model.(De Oliveira, Nobre and Zárate, 2013). DNN is an advanced version of ANN with more than one hidden layers.Recurrent Neural Network(RNN) is a variation of ANN where the output of previous step is used as input of current step thus capable of remembering immediate past value.However RNN have problem in handling long term dependencies that are solved by Long Short Term Memory(LSTM) which include memory cell that can maintain information in memory for long period of time.

LSTM models have shown promise over years and multiple research attest to the fact.LSTM compared to traditional time series auto regressive integrated moving average (ARIMA) model gives good results in terms of higher accuracy and lower forecast errors.(Siami-Namini and Namin, 2018).Another enhanced use of LSTM model through attention mechanism was used for stock price forecasting after extracting news information in auxiliary role to gauge price movement.Stock price

passed through wavelet transform and attention based LSTM was found promising.(Qiu, Wang and Zhou, 2020).A LSTM-DNN based time-series model for stock price prediction was developed using new auto-regression scheme,autoregressive moving pointer model(AMPM)..Herein input output that are fed to LSTM-DNN model are generated through the AMPM model on NIFTY50 data.(Rather, 2021).Ensemble model consisting of deep neural network(DNN),gradient boostd tree(GBT),Random forest(RF) for equal weights strategy with top 10 stock of S&P500 provide daily returns of 25% and 73% per annum.Research also points to the importance of hyperparameter tuning and combining base learners as per the compute power,so does the need to use advanced ensemble integration methods like stacking or superlearning.(Krauss, Anh and Huck, 2017)

Reinforcement learning(RL) frameworks aims at solving Markov decision problems. RL framework has three major formal elements (i)Sequential decision making(ii)Scalar reward(iii)Delayed feedback.Major components of RL configuration are agent-environment,action-reward.The baseline for RL is Markov decision process(MDP) wherein the reward depends on last state and action,current state is the sufficient representation of past.

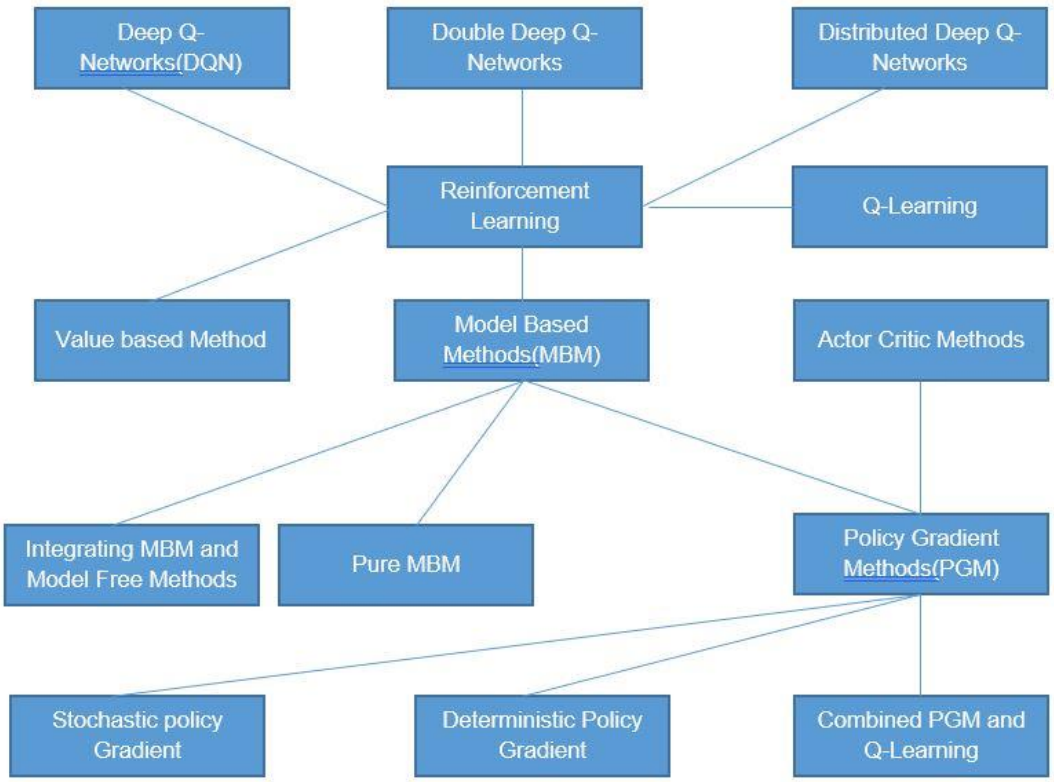


Fig 1.Dominant Algorithms in Modern RL used in Finance.(Singh et al., 2022)

There is a assumption that the agent acts to maximise reward. RL algorithms have two popular classifications viz:value based and policy based.RL have helped in modelling the stock prediction problem closer to real world scenarios be it accommodating constraints or moving away from strict model formalisations.(Singh *et al.*, 2022).

Use of Policy or Value based RL methodology is determined by the nuances of problem at hand.Q-learning a type of value based approach has been tried very early in combination with nueral networks to achieve better results than the nuero-fuzzy model.The experiment gave 25% better returns in test phase and kept capital out of market in case of volatility or absence of trends.(Neuneier, 1996).Q-Learning compared with policy based method recurrent reinforcement learning(RRL) does not fare equally good on efficiency and simplicity often leading to unnatural problem representation.RRL method was found to be more explainable than Q-learning as well.Run time of Q-learning was almost 150 times that of RRL method.(Moody and Saffell, 2001).In Current times there is a big push on sustainable development,various supra-national organisations including United nations have come up with sustainable development goals.In this vein global investors and organisations want to invest under the paradigm of socially responsible investing(SRI).Enterprises that promote sustainable development score high on Environmental,Social and Governance(ESG) metrics.

SRI has been enabled through an experiment wherein multivariate bidirectional LSTM is used to multiple time series prediction for stock returns for a multi-objective portfolio construction.Reinforcement learning is used to tune hyperparamaters as per changing market dynamics.(Vo *et al.*, 2019).

2.5 Summary

The chapter starts with an overview of traditional portfolio optimization theory and methods.

This provides a baseline over which development in the portfolio optimisation areas have happened.Limitations of MVO methods have been highlighted which broadly relate to their too simplistic view of portfolio optimisation, which has additional nuances added when portfolio optimisation plays in real world scenarios.There is a discussion on how multiple shortcoming have been addressed by new methods due to growth in research and emergence of new computational power in technology.Development in portfolio optimization theory have been logically explained

under two emerging trends one being problem specific development where portfolio optimisation has been treated in more complexity nearing the real world situations.Other being the methodology specific development that has seen new methods,hybrid methods and computationally intensive methods gaining popularity.

Two areas of developments have been described in detail from dimensions of problem and method enhancements viz. Traditional machine learning methods and Deep learning methods.

Major research in the area has happened in supervised learning space where historical data is used to either forecast the value or direction of stock at various frequencies.Regression and classification methods have been described highlighting major development in those areas.Ensemble models like XGboost,Random Forest have been illustrated with their use cases of research.Linear models like decision trees have also been highlighted for their utility.Finally in discussion about the deep learning models which provide solutions to the shortcoming of traditional Machine learning models standard deep learning models like ANN,CNN,RNN and their utility in portfolio optimization space are discussed.Certain variants of standard model and hybrid model find mention.In other category RL models are discussed with their variants and how they help in imbibing constraints in their formalisation.

CHAPTER 3

METHODOLOGY

3.1 Introduction

Portfolio optimisation is a data intensive problem solving method.Historical data from stock market is used to derive historical patterns that show the vital statistics of a stock.Indicate the direction of market, reveal a lot of micro and macro relationships.Data granularity can go down to second level information to daily open-close prices some of which is freely publicly available.Traditional method of portfolio optimization relates to Mean Variance optimisation(MVO) that can be programmatically solved used computational tools.MVO tries to find a sweet spot of maximum returns at optimum risk or variance which can be measured through the metric of Sharpe ratio.This techniques leads to weights or proportions that could be invested said stocks of portfolio.In this technique temporal pattern of close price of the stock could be used to get both a measure of return and variance in stock.

Machine learning methods of portfolio optimisation on the other hand mostly deal with portfolio optimisation as a supervised learning problem wherein either the direction or value of stock is forecasted thus it turns into either a classification or a regression problem.As per the choice the metric of model performance also change.Metric in a classification problem range from Area Under the Curve(AUC) to metrics like recall,sensitivity or accuracy derived from confusion metrics.In a regression problem metric like Root Mean Square Error(RMSE),Mean Absolute Percentage Error(MAPE) are preferred.

The methodology followed in this study compares the traditional portfolio optimisation methods of mean variance optimization with multiple standards and ensemble machine learning. Forecasts thus obtained are used to simulate trades using both the baseline methodology derived weights and the weights derived from the top ML models developed using AutoML tool datarobot.

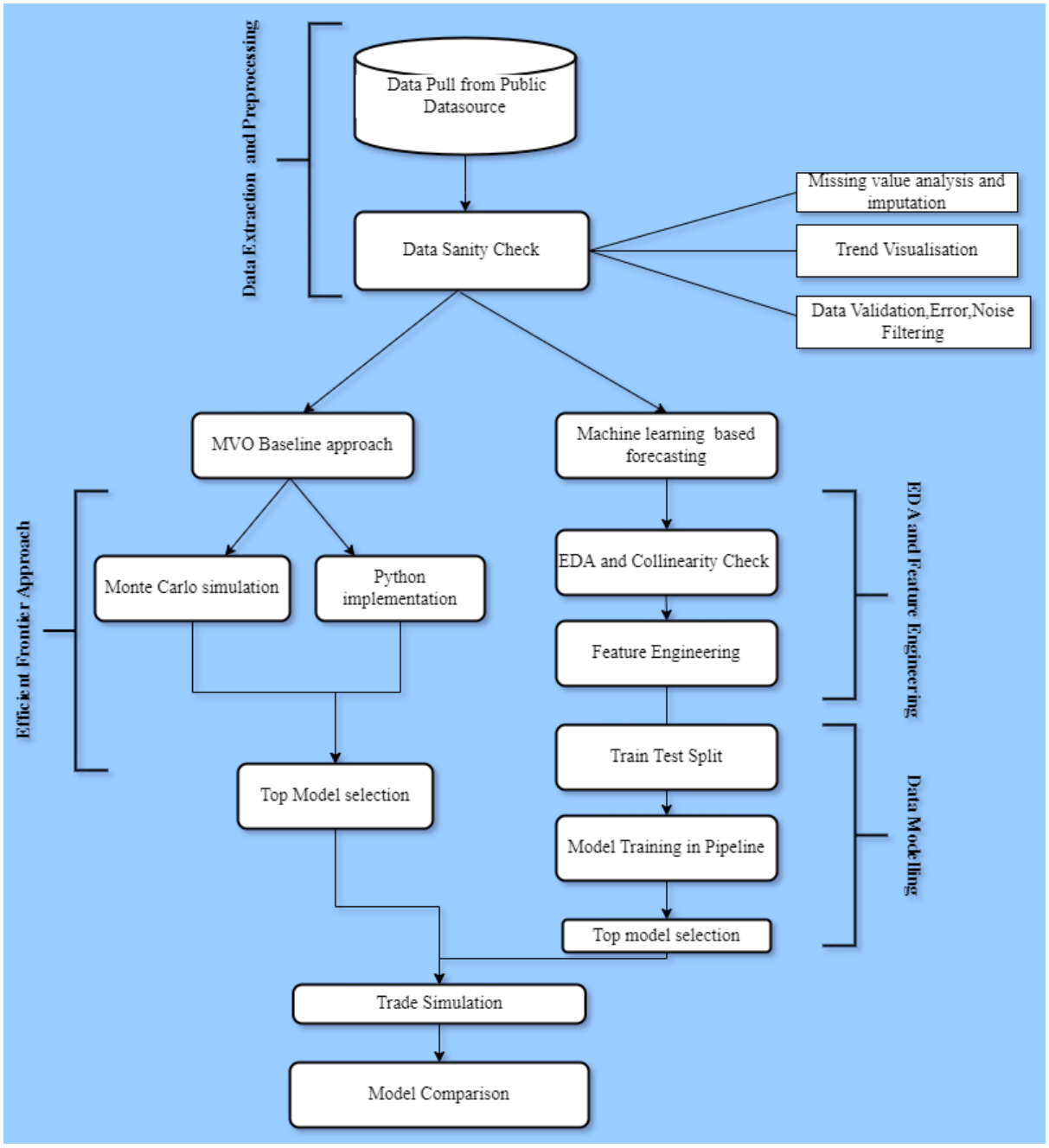


Fig 2.Flow of work

3.2 Markowitz Portfolio Selection Method

Markowitz theory of portfolio optimization is based on mean variance optimization.

For theoretical treatment let's define the expected returns R_i as:

$$R_i = \sum_{t=1}^{\infty} d_{it}\, r_{it} \text{ return for } i^{\text{th}} \text{ security} \quad \text{...3.2.1}$$

$$R = \sum x_i\, R_i \quad \text{is the total return} \quad \text{...3.2.2}$$

Where the variables r_{it} is the anticipated return at time *t* per dollar for security *i*.

x_i is relative amount invested in each stock,

$x_i \geq 0$ this excludes short sales, $\sum x_i = 1$ (Markowitz, 1952)

Thus for a portfolio with N assets expected return(*E(r)*) can be expressed as:

$$E(r_p) = \sum_{i=1}^{N} x_i r_i \quad \text{...3.2.4}$$

Variance for such a portfolio can be expressed as:

$$\sigma_p^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} x_i x_j \sigma_{ij} \quad \text{...3.2.5}$$

Where $\sigma_{ij}$is the covariance between the return of *i*th asset and *j*th asset.Mean variance optimization has three constraints under which the variance is minimised with constraints as:

$$Min.(\sigma_p^2) = \sum_{i=1}^{N} \sum_{j=1}^{N} x_i x_j \sigma_{ij} \quad \text{...3.2.6}$$

$$Return = \sum x_i R_i \quad \text{...3.2.7}$$

$$\sum x_i = 1$$

$$x_i \geq 0$$

The solution to problem leads the efficient frontier for N risky assets.(Gökgöz and Atmaca, 2012)

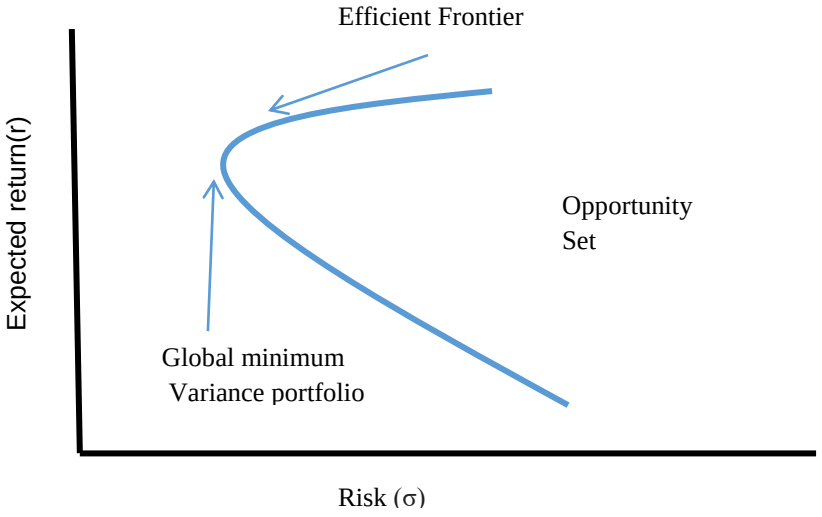


Fig 3.Efficient Frontier representation

Defining the investor's utility function (U):

$$U = E(r_p) - \frac{1}{2} A \sigma_p{}^2 \quad \text{...3.2.8}$$

here *"A"* is an index of investor risk aversion.(Gökgöz and Atmaca, 2012)

Thus the final equation looks like:

$$U = E(r_p) - \frac{1}{2} A \sigma_p{}^2 \quad \text{...3.2.9}$$

Under constraints:

$$\sum_{i=1}^{N} x_i = 1 \quad \text{...3.2.10}$$

$$x_i \geq 0, \forall x_i \in [i = 1,2,3 \dots N]$$

Where ...3.2.11

$$E(r_p) = \sum_{i=1}^{N} x_i r_i$$

$$\sigma_p^2 = \sum_{i=1}^{N} \sum_{j=1}^{N} x_i x_j \sigma_{ij}$$

3.3 Supervised learning

The challenger methodology to Markowitz mean variance optimization method is supervised learning models. Machine learning is a subfield of Artificial intelligence where the aim is to develop "Agents", "Mathematical functions" that can iteratively perform better through information gather from world.(Laperrière-Robillard, Morin and Abi-Zeid, 2022).In supervised learning usually a large enough set of learning examples are used to establish a mathematical relationship between the independent and dependent variable.Independent variables are a set of statistically or business defined variable that are considered to be significant enough to correlated to the dependent or target variable.Cases when the dependent variable is categorical are called classification problems and where the dependent variable is continuous are called regressions problems.Often choice of independent variables get restricted to data available for a particular phenomenon.There are tests to determine which variables are of importance and which are redundant or duplicate.Phenomenon where the variables are scarce or have relationship which are not explicit to mathematical function in such scenarios uni-variate,bi-variate or multi-variate analysis is done to find hidden or implicit relationship.This technique called feature creation or feature engineering is of much importance in supervised machine learning area.Supervised learning has two important components the learning phase and prediction phase.In learning model or mathematical function is tuned or trained are per training data.In prediction phase the learned model is used to make predictions on test data.These test and train data are part of the same data set which are split as data sampling methodology apt for problem at hand.These sampling strategies could like random sampling, stratified sampling or as in this case sequential split.These choices is made to keep the model accurate and generalised for unseen data or reality. There are multiple metric which are used to select the best fit, which will be discussed in later sections.

Supervised learning phases could be summarised into logical phases. (Rohaan, Topan and Groothuis-Oudshoorn, 2022)

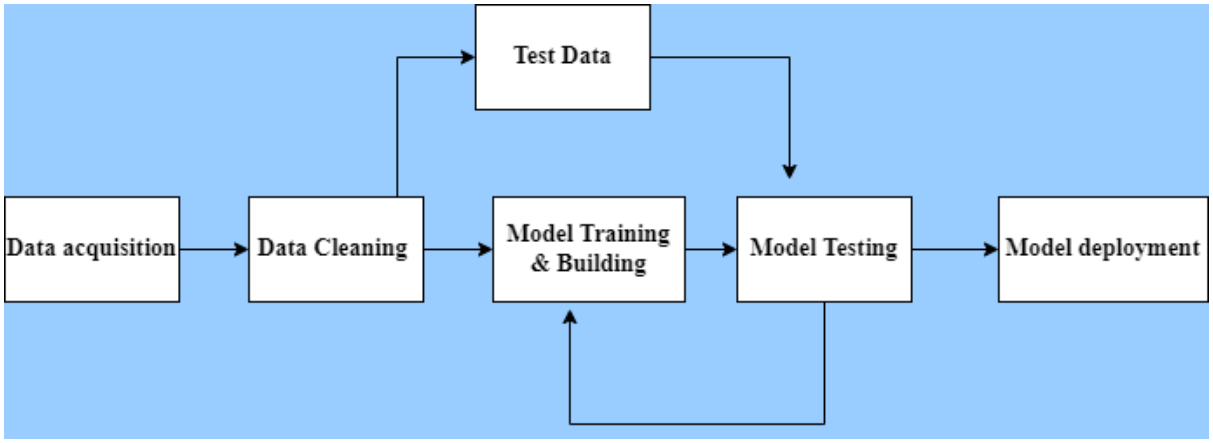


Fig 4.Supervised machine learning framework

3.3.1 Input Data and Feature Engineering

Input data in supervised learning is use case specific data.Data could be structured i.e having a defined format like int, float,string or time series data as in this case,data could be semi-structured like java script object notation(json),hypertext markup language(html) etc. or data could be unstructured like images,audio,ge-spatial data.In case of semi-structured and unstructured data cleaning and pre-processing more intensive than in case of semi-structured data.However data validation,veracity check is mandatory for all forms of data.The choice of feature engineering depends on the input constraints data of the machine learning algorithm being used and the specific underlying relationship needs expression.Input variable for Stock prediction problem could be classified broadly into fundamental indicators, macroeconomic indicator and technical indicators.(Tsai and Hsiao, 2010).Fundamental indicators are derived from analysis of company fundamentals like annual statement report,balance sheet,growth forecast in the area where company operates,market capitalisation etc.Economic indicators are the macro-economic factors that indicate the health and direction of economy where the particular stock operates as the stock indexes are impacted due macro-economic scenarios.These indictors influence the demand-supply forces, investors both domestic and international confidence.Ecomonic indicators are numerous and not limited to Gross Domestic Product(GPD),GDP growth,Index of industrial production(IIP),Wholesale Price Index(WPI),monetary rates etc.

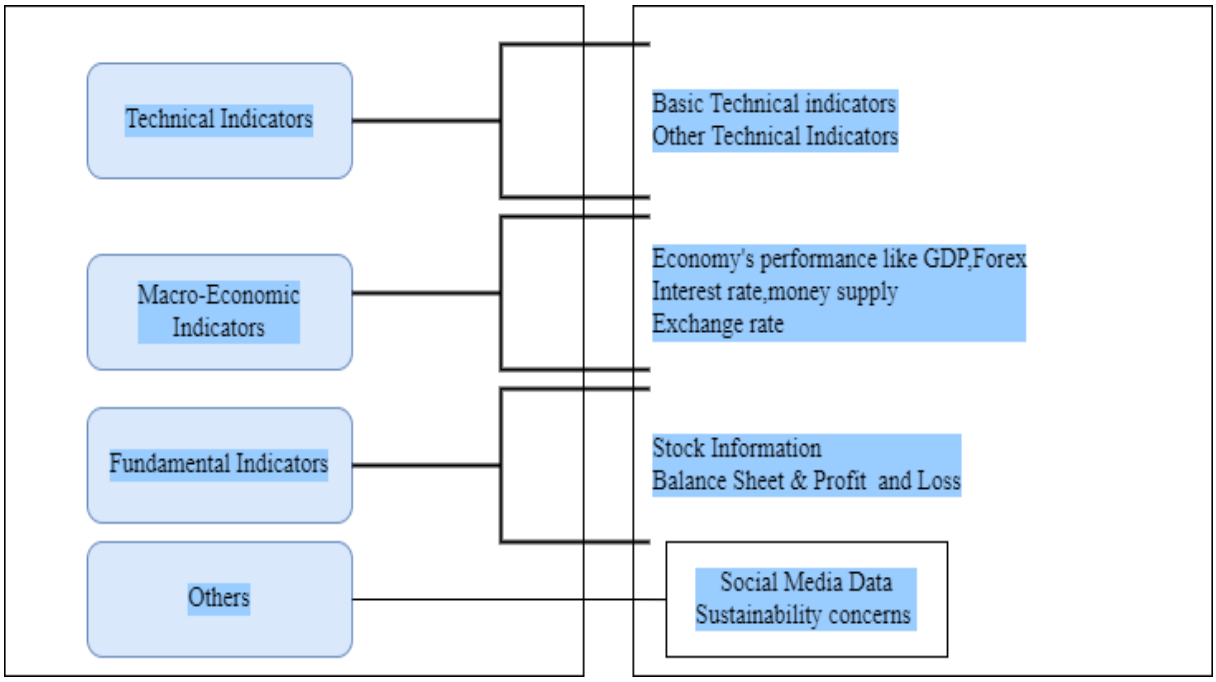


Fig 5.Input Variable types and sub-types.(Kumbure et al., 2022)

Input data is of the format received from yahoo finance through yfinace package of python is of the form where daily dynamics of stock are indicated in s time-series format.

Table 1.Input stock data structure.

Dt.	Open	High	Low	Close	Adjusted Close	Vol.
1/1/2020	121	122	120	120	120	538
1/1/2020	120	121	119	119	119	771

Input data attributes definition are detailed in table 2.

Table 2.Attribute definition of Stock data.

Attribute	Description
Date	Date of Transaction
Open	Opening Price of stock at the beginning of Trading Day.
High	Highest traded price of the Stock in the day

Low	Lowest traded price of the Stock in the day
Close	Closing Price of a stock at end of Trading Day.
Adj Close	Adjusted close is the closing price after adjustments for all applicable splits and dividend distributions. Data is adjusted using appropriate split and dividend multipliers, adhering to Center for Research in Security Prices (CRSP) standards
Volume	These are the Physical number of shares traded on that stock in a Day.

Technical indicators have traditionally been used study dynamics of stock market at granular and aggregated levels.Stock price movement is charted,measures of central tendencies derived over time to state few.They include moving average(MA),moving average convergence and divergence(MACD),relative strength indicator(RSI) etc.(Tsai and Hsiao, 2010).

For the purpose of this study simple moving average(SMA) and relative strength indicator(RSI) are being derived from stock data.Simple moving average is the simple average of closing price of a stock for past "N" days.

$$SMA(N) = \frac{CP_1 + CP_2 + CP_3 + CP_4 \ldots\ldots\ldots\ldots\ldots + CP_N}{N} \quad \ldots 3.4.1$$

Where CP_i denotes price at i^{th} day.

SMA provides a reliable metric providing smoothened pattern of price movement with minimal noise.SMA with *"N"* values 14,30,50,100 are popularly used.

RSI is another popular technical indicator.It is a momentum indicator that indicates the intrinsic strength of the stock.

$$RSI = 100 - \left(\frac{100}{1+RS}\right) \quad \ldots 3.4.2$$

$$RS = \frac{Average\ of\ X\ days'\ up\ closes}{Average\ of\ X\ days'\ down\ closes} \quad \ldots 3.4.3$$

Where RS is the relative strength.

3.4 Machine Learning Models

Supervised Machine learning models used to solve stock price prediction problem here are regression based models.Sequential data split was done to keep the time series pattern intact.

This suits the scheme because model should learn from historical patterns,in this scheme older data is used to train the model which is tested on recent data.As it is a regression problem the Target

variable is one day future closing price of stock.The models used in the study are advanced Boosting or ensemble algorithms. Such algorithms have a characteristic of having multiple hyperparameters.Hyperparamters are arguments that get supplied mostly manually by model developers.To find parameters that lead to best performance a process of hyperparamter tuning is carried out.There are multiple hyperparameter tuning techniques like random search,Bayesian optimization,Evolutionary optimization,Grid search etc.Grid search was used for hyperparameter tuning during the study due to its generalizability and efficiency as intuitive grid values are searched. Other methods might provide results that are better and non-intuitive but require a lot of time due higher search space.Another technique that needs mention due to the nature of algorithms used in study is early stopping support,this is a technique which checks for incremental gains by adding more tree to the model training process if the model performance decreases by adding more iterations beyond a point then the training process stops.This is a deterrent against overfitting.

3.4.1 eXtreme Gradient Boosting(XGBoost)

XGBoost is tree Boosting Algorithm that has high computational efficiency and low complexity.XGBoost has proven been on leaderboard of various computational challenges due to its high accuracy.Multiple classification and regression tree(CART) act as slow learners to iteratively reach results with higher accuracy.XGBoost models have regularisation term that control the variance of fit in order to keep model flexible and generalizable to avoid overfitting.(Nobre and Neves, 2019).Given the XGboost trees develop in parallel model provides for parallel computing with faster execution time.Given the data points in real trading environment could increase exponentially if the granularity set to second or hour level XGBoost model could prove robust.XGBoost can use tweedie loss for zero-inflated positive distributions,for right skewed positive distributions gamma loss function could be used,poisson loss for count problems though the default loss function is least squares loss.

3.4.2 Random Forest

Random forest belongs to class of ensemble models where multiple similar models are used to arrive at consensus output.Random Forest model is composed multiple decision trees which are run on bootstrapped data on random set of “N” features this leads to creation weakly correlated trees as opposite to decision trees.These “N” features is one important parameter of this algorithm that can be configured.This leads to generalised model that performs well on unseen data.Power of parallel

computing is also an advantage and model accuracy is high.Another important additional feature is that of “ExtraTrees” model which increases the randomness in splits.As no sorting is done on input data for splits thus this model is even more efficient.

3.4.3 Light Gradient Boosting

Light Gradient Boosting Models(LightGBM) have host of benefits that range from faster training time,parallel learning,better accuracy to state few.These models apart from handling missing data also balance bias and variance appropritelty.GBM models are similar to random forests in the sense that they fit multiple decision trees but these trees are not fit independently as in Random forest rather the new tree is fitted on the residual errors of all previous trees combined.Two critical parameters to these models are learning rate and number of trees,these parameters should be set carefully as this could lead to overfitting.GBM models have a quality of extracting maximum latent patterns from the dataset as they have a capability of finding a point from where overfitting of models begins and stop training just a step before that state.

3.4.4 Eureqa Generalised Additive model

Eureqa model is a dependent model it uses Eureqa’s engine to come near GBM predictions.

Eureqa models are AI-powered proprietary models of Datarobot that leverages automated evolutionary algorithm.Model is explainable in the sense that models are represented in form of mathematical form.Eureqa models have concept of expressions that are human readable mathematical equations that highlight features interactions.Basic Blocks are mathematical operators within those symbols.These are advanced tuning variables that can impact model evolution and have an impact of model complexity with a related metric of complexity score.The hyperparameters of this model are permutation combination of building blocks and proxy hyperparameters of XGBoost.

3.4.5 Elastic net regressor

Elastic net regressor is based on Lasso and ridge regularisation trained linear regression.This provides for advantages of both lasso and ridge.Elstic nets are used for scenarios where there are large number of correlated features.This comination of L1 and L2 regularizers leads to sparse models where only some weights are finite.Elastic net model allows the dependent variable to have error distributions that are different from normal like poisson,gamma etc.

3.5 Data Split

Data sampling for the particular study is sequential for model to learn from historical patterns and predict future stock prices.

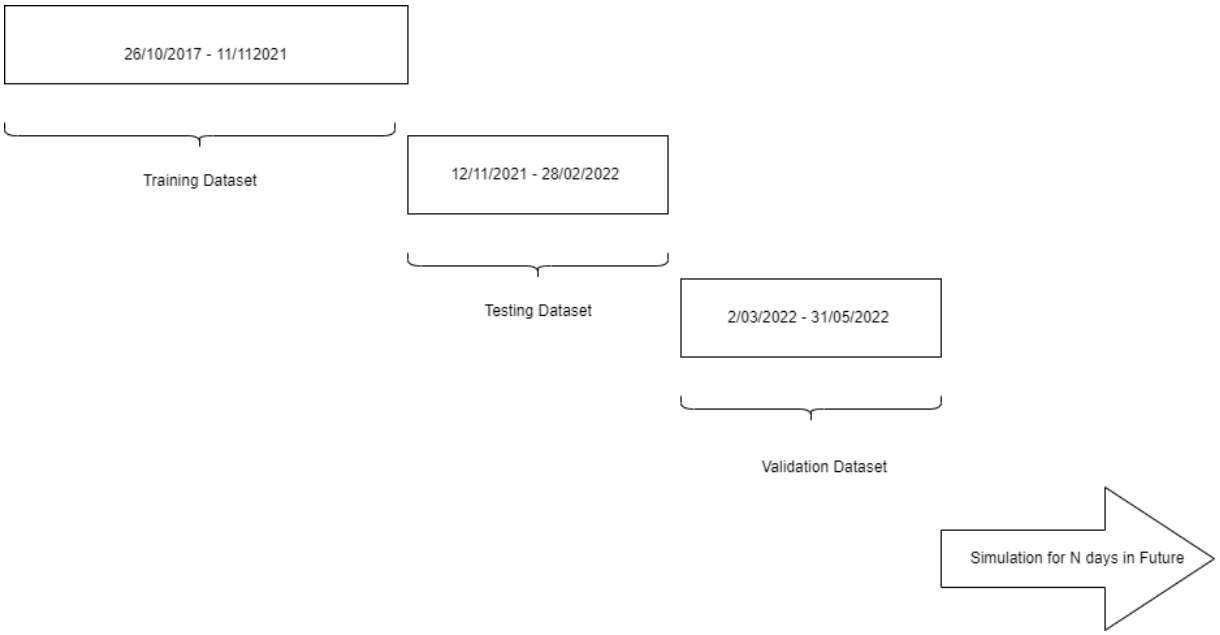


Fig 6.Data Split framework

Same scheme for data splits is followed for all the stocks and multiple models that are build for forecasting stock prices.More than four years data is used for model training and three months each data is used as validation and test set.Test set is used as holdout sample because model does see this data till the end of training and hyperparameter tuning steps.

3.6 Model Metrics

Mean Variance optimisation method primarily utilised three measures to gauze method's performance are the return at optimum risk or variance in intuitive terms.Sharpe ratio is an important metric in portfolio optimisation.

Sharpe ratio in generic terms measures the risk and reward of a portfolio. Its formula can be expressed as:

$$\frac{Excess\ return\ of\ the\ portfolio}{Volatility\ of\ the\ porfolio} = \frac{R_p - R_f}{\sigma_p} \quad \text{...3.5.1}$$

In equation 3.5.1:

R_p is return of the portfolio

R_f is risk free rate

σ_p is standard deviation of portfolio's excess return.

Sharpe ratio provides a cue into performance of a portfolio and further comparison, higher Sharpe ratio indicates higher returns under optimum risk.

For supervised machine learning models that have been used in the study the metric of choice is mean absolute percentage error (MAPE):

$$MAPE = \frac{Absolute\ Forecast\ Error}{Actual} \times 100 \quad \ldots 3.5.2$$

$$MAPE = \frac{1}{N} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right| \times 100 \quad \ldots 3.5.3$$

Where A_t is the actual value for t observation and F_t is the forecasted value.The error percentage absolute forecast error to actual is divided by total observation in the set under study to obtain mean absolute percentage error.MAPE has been chosen to compare multiple Machine learning models that require different data manipulations. Lower value of MAPE is preferred.Also a significant difference is MAPE of training and test set or unseen data with MAPE value higher for test data is an indicator of model Overfitting.As no data normalisation or standardisation was conducted for any model thus MAPE provides good baseline for comparison.

One other calculation which is also used as metric is the percentage return.

$$\%\ Return = \frac{(Close\ Price - Open\ Price)}{Open\ Price} \times 100 \quad \ldots 3.5.4$$

Provided the stock is bought at open price at the start of session and sold at end of session.

3.7 Tools

Tools used in this study were anaconda provided python version 3.6.1.Python is an opensource functional programming language that among other things is very popular for data analytics and Machine learning.Python comes with multiple packages supporting various fuctions.Pandas package is a popular package in python this package is used for data reading,writing and manipulations has been used during this study.Matplotlib is a python package used for visualisations.At places google collab has also been used to speed data processing.Certain other propriety softwares have also been used for the process of analysis and modelling. Tableau version 2020.4 has been used to create advanced visualisations.Tableau is beneficial for creating custom plot,dashboards with multiple option in charting and data manipulations.AutoML tool Datarobot has been used to build challenger models.Datarobot has an advantage of faster model executions due computing supported by multiple

cores of spark cluster that has proved to be industry reliable.Datarobot also allows easy and robust deployment of Machine learning models.Finally Microsoft excel was used to prepare returns calculator that helped in calculating portfolio weight,daily returns etc.

3.8 Simulations

Under this study the final part is to conduct simulations with forcasted values.In first simulation Finding the returns from portfolio by creating the portfolio with weights assigned from MVO methods and comparing it with returns received from portfolio created for the forecasted share prices through ML models. Portfolio weights for ML based model will in proportion to the percentage returns received from the stock.Here a rational investor will be considered who will invest in the proportion of expected returns a day in future through the ML algorithm.

Analysis of portfolio is done holdout set where in the forecasted return is compared with actual return.On the days where a negative return is predicted the return is considered as zero and share does not get any weight.

Table 3.Negative return scenario

% return predicted	% Predicted non-negative Return	Weight in portfolio
-2.163715464	0	0
-2.157792262	0	0
-1.910558374	0	0
-0.874059988	0	0
-0.653542372	0	0

Here one condition laid out which relates to stop loss.As the data available has only a day level granularity thus it’s difficult to see if stop loss condition was met earlier in the day.So there is an assumption that stocks close price is the best indicator of its intraday trend. Any day for the stock where actual loss is below 1% is thus being capped to 1% considering a stop loss at 1%.

Table 4.Stop loss at 1%

Date	Day+1 forecast	Actual day Adj close price	A actual day+1 price	% return predicted	% Predicted non negative Return	Asianpaints weight	Actual Return %	Loss due to stop loss %1
02-03-22	3020.864	3011.594971	2855.4446	0.307762596	0.307762596	7%	-5.18497316	-1
03-03-22	2878.763	2855.44458	2722.4609	0.816628276	0.816628276	9%	-4.65719569	-1
04-03-22	2748.24	2722.460938	2692.9312	0.946899554	0.946899554	12%	-1.08467253	-1

Under this condition the return of the portfolio was calculated for ML based foreacasts.

For MVO model the assumption was that the portfolio weights would remain constant over the holdout period and a long strategy will be followed where the profits will be calculated at end of period comparing the percentage returns. In the second simulation portfolio optimisation for pre-covid and post covid days.

Second simulation is tried for which the data is split into pre-covid and post-covid time frames.

Table 5.Data split into pre and post covid periods

Data time frame	Train	Validation	Test
Pre-covid	26/10/2017-26/10/2019	27/10/2019-12/12/2019	13/12/2019-14/02/2020
Post-covid	15/02/2020-15/02/2022	16/02/2022-02/04/2022	02/04/2022-10/06/2022

Implementation steps and conditions for the second simulation remain same as first simulation only that the number of models created increases to 20,ten each for pre-covid and post covid time frames.

3.9 Summary

Methodology of the study has been driven by two choices and its leading consequences.Portfolio optimisation through MVO technique bases itself on annualised return and variance wherein sharpe ratio is used as a metric.Data being sourced from free public source has a daily granularity.MVO does not require feature engineering and uses data in raw form.MVO method is implemented in two ways first being random simulations of weights for each stock,with corresponding return and risk being captured.The other method uses python custom package to find optimum portfolio.

Supervised learning methods require feature engineering to extract information from data.Use of ensemble models help in gaining optimum efficiency as proven in number of studies.This study is designed to forecast future value of stock thus regression models are used.A pipeline of models is created using AutoML tool Datarobot and top performing models are chosen as challengers on the basis of MAPE as data not scaled before predictions.

Comparison of returns from the portfolio created through weights arrived at by MVO method is done with forecasts from the top ML models.These simulations are conducted and discussed in "Results and Conclusion" section.

CHAPTER 4

IMPLEMENTATION

4.1 Introduction

Data was sourced from freely available public data source of yahoo finance.A set of ten stocks were chosen from the National stock exchange(NSE) benchmark index Nifty50.Past four and half year data was read through python package.This data was analysed for trends in Tableau,data was smoothened through data manipulation in python.Stock data was in time series format with open,close price ,volume etc.This data was used to create optimlal portfolio using Monte Carlo simulation in python to find the efficient frontier.Mean variance optimisation was also done using Polyoft package of python.Optimum portfolios for maximum returns,maximum varianace and maximum sharpe ratio were obtained.

Challenger models,mostly ensemble models were developed in datarobot.These models were fed with data that was enriched through feature engineering.New features representing momentum indicators like simple moving average(SMA),Relative strength index(RSI) were created for multiple time periods like 7,14,50,100,200 were created.This data was split into train-test–validation set and fed into Datarobot to create select models which were further tuned to get optimal results.The predicted values from the test set which is unseen data is further used simulate trades in one data future.

4.2 Data Extraction

A representative set of stocks chosen from Nifty50 was selected for portfolio optimisation.The list includes stocks from different sectors of economy like services,consumer durables,telecom,power etc.Certain notable like State Bank of India is the largest public sector bank of India,Relaince Industries with a market cap of US$203.74 billion and a revenue of US$97 billion,Tata consultancy services is the largest Information technology and enabled services company in the world with a market capitalisation of US$200 billion.

Table 6.Representative Stock List

Company Name	Industry	Symbol
Adani Ports and Special Economic Zone Ltd.	Services	ADANIPORTS
Asian Paints Ltd.	Consumer Durables	ASIANPAINT

Bharti Airtel Ltd.	Telecommunication	BHARTIARTL
Hindustan Unilever Ltd.	Fast Moving Consumer Goods	HINDUNILVR
NTPC Ltd.	Power	NTPC
Reliance Industries Ltd.	Oil Gas & Consumable Fuels	RELIANCE
State Bank of India	Financial Services	SBIN
Sun Pharmaceutical Industries Ltd.	Healthcare	SUNPHARMA
Tata Consultancy Services Ltd.	Information Technology	TCS
UltraTech Cement Ltd.	Construction Materials	ULTRACEMCO

Data is downloaded from open source yahoo finance database.Yfinance package of python.

```
import yfinance as yf
ticker= "HINDUNILVR.NS"
stock_Hul = yf.download(ticker, start="2017-10-26", end="2022-06-10")#Past 4.5 years data
stock_Hul.head()
```

```
[**********100%**********]  1 of 1 completed
```

Date	Open	High	Low	Close	Adj Close	Volume
2017-10-26	1285.000000	1285.000000	1253.500000	1273.699951	1175.047363	2321903
2017-10-27	1273.699951	1273.699951	1246.000000	1260.150024	1162.546875	875032
2017-10-30	1260.000000	1269.449951	1230.000000	1234.199951	1138.606689	1063165
2017-10-31	1241.000000	1249.900024	1230.449951	1237.449951	1141.605103	722440
2017-11-01	1238.699951	1263.000000	1238.000000	1258.000000	1168.115112	1243886

Fig 7.Data Download

Data is received through “download” function to save data in a dataframe which is similar to a two-dimensional table.Data is downloaded for the period 26/10/2017 to 10/06/2022 for all ten stocks.data sanity is checked by looking at the top five records through .head() function.
Data was downloaded for 10 stocks and downloaded into .csv files.

4.3 Initial Data Visualisation

Visualisation of data from all the ten stocks on close price shows the trend of shared over past years.Data at daily granularity has lot of noise leading to uneven pattern.

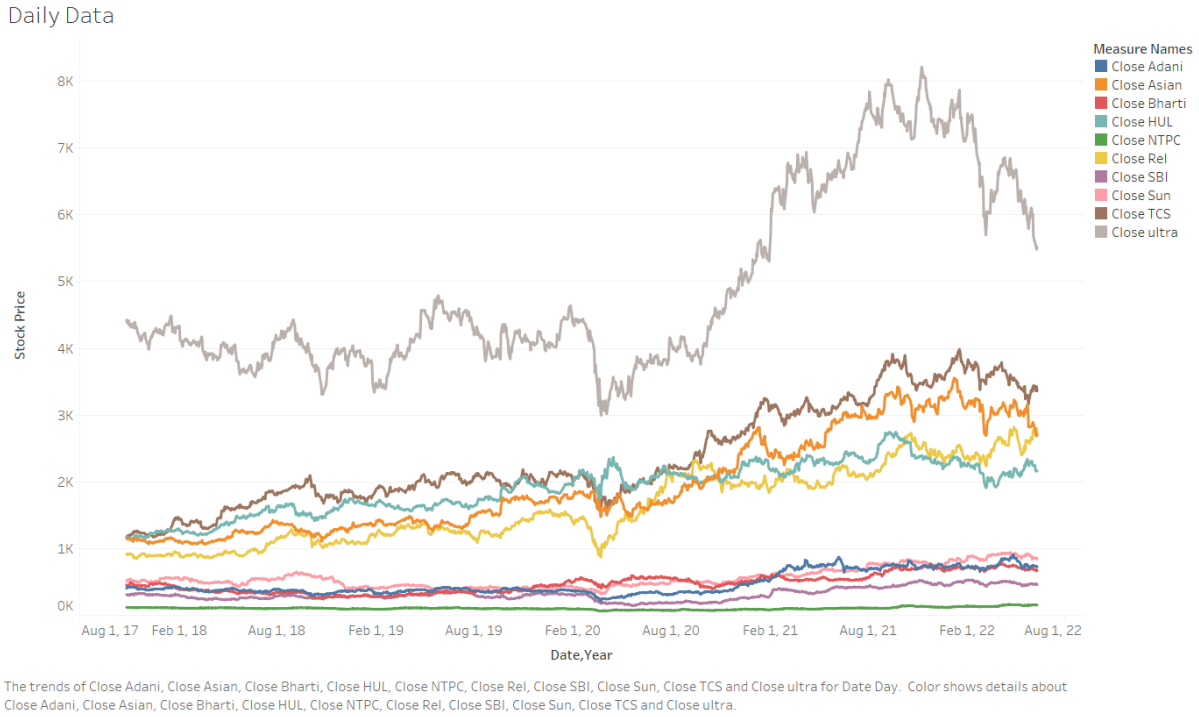


Fig 8.Daily stock price movement for 10 stocks.

The sharp dip in February of 2020 was due to impact of Covid-19 which led to sharp decline in markets all over the world.The sharp peaks in September of 2021 till November of 2021 are due to businesses picking up in post lockdown period.Coming June 2022 markets again begin to decline which is visible in all the stocks dropping for said period.

4.4 Mean Variance Optimisation implementation

Mean variance optimisation is implemented in python in two ways.First method involves use of python libraries to create annualised returns and co-variance matrix.Further calculating maximum returns, minimum variance, maximum Sharpe ratio portfolios.

After reading the stocks data, First step involves calculating percentage change on returns.

Pct_change function is used to find the percentage change is previous day's adjusted close price to today's adjusted close price.

```
for ticker in tickers:
    data=web.DataReader(ticker,'yahoo',start,end)
    data=pd.DataFrame(data)
    data[ticker]=data['Adj Close'].pct_change()
    if returns.empty:
        returns=data[[ticker]]
    else:
        returns=returns.join(data[[ticker]],how='outer')
```

Fig 9.Code snippet for Percentage change calculation

```
print(returns.head())
```

```
            HINDUNILVR.NS  ADANIPORTS.NS  ASIANPAINT.NS  BHARTIARTL.NS  \
Date
2017-10-26            NaN            NaN            NaN            NaN
2017-10-27      -0.010638       0.042843      -0.005541      -0.050851
2017-10-30      -0.020593      -0.005208      -0.001688       0.015660
2017-10-31       0.002633       0.001280       0.000805       0.008724
2017-11-01       0.023222       0.006390      -0.004149       0.084272

             NTPC.NS  RELIANCE.NS   SBIN.NS  SUNPHARMA.NS    TCS.NS  \
Date
2017-10-26       NaN          NaN       NaN           NaN       NaN
2017-10-27 -0.014622    -0.018524 -0.029485      0.037254  0.017902
2017-10-30  0.003572     0.013604  0.003054      0.005819  0.012344
2017-10-31 -0.007393    -0.001751 -0.019872     -0.000361  0.002982
2017-11-01 -0.002759     0.012436  0.045945     -0.007596 -0.008060

            ULTRACEMCO.NS
Date
2017-10-26            NaN
2017-10-27      -0.009824
2017-10-30       0.007298
2017-10-31      -0.015921
```

Fig 10.Code Snippet rows display.

Monte-Carlo simulation is done to get multiple combination of weights for each stock that add upto one.1000 such rounds are runs to assign random weights this is done by using random.random_sample of numpy.

```
##Generate portfolio weights##
for portfolio in range(rounds):
    weights = np.random.random_sample(len(tickers))
    weights = np.round((weights/np.sum(weights)),3)
    portfolio_weights.append(weights)
```

Fig 11.Code Snippet for randomised weights generation

Next steps are looped to create mean annualised return under the conditions of 252 trading days in year for 1000 combination of portfolio weights. This is appended to an empty list ‘portfolio_returns’

```
    ##calculate annualised returns##
    annualized_return=np.sum(returns.mean() * weights) * 252
    portfolio_returns.append(annualized_return)
```

Fig 12.Code snippet for annualised return calculation

Covariance in returns is calculated for trading days in an year and resulting variance in the portfolio is calculated to randomly generated portfolio weights.sqrt() function of numpy is used to calculate the standard deviations which is attended in empty list portfolio_risk .

```
    ##Matrix co-variance and portfolio risk calculation##
    matrix_covariance = returns.cov() * 252
    portfolio_variance=np.dot(weights.T,
                              np.dot(matrix_covariance,weights))
    portfolio_standard_deviation=np.sqrt(portfolio_variance)
    portfolio_risk.append(portfolio_standard_deviation)
```

Fig 13.Code snippet for covariance and standard deviation calculation

In the last step of block code sharpe ratio is calculated using annualised_return and standard deviation generated combinations.

```
##sharpe ratio##
sharpe = (annualized_return - RF)/portfolio_standard_deviation
sharpe_ratio.append(sharpe)
```

Fig 14.Code snippet for Sharpe ratio calculation

These lists are converted into numpy arrays and collated to obtain a list that contains corresponding randomised portfolio weight based entries of return,risk,sharpe ratio and weights for ten stock in the portfolio

```
portfolio_returns=np.array(portfolio_returns)
portfolio_risk =np.array(portfolio_risk)
sharpe_ratio =np.array(sharpe_ratio)
portfolio_weights =np.array(portfolio_weights)
portfolio_metrics=[portfolio_returns,portfolio_risk,sharpe_ratio,portfolio_weights]
```

Fig 15.Code snippet for structuring results in list format

These are further saved in dataframe for easy reference and search.

```
portfolio_df.columns=['Return','Risk','Sharpe','Weights']
portfolio_df.head()
```

	Return	Risk	Sharpe	Weights
0	0.192117	0.195493	0.982733	[0.1, 0.125, 0.039, 0.059, 0.004, 0.15, 0.081,...
1	0.16763	0.198688	0.843685	[0.03, 0.128, 0.063, 0.111, 0.179, 0.049, 0.15...
2	0.190776	0.1832	1.04136	[0.163, 0.139, 0.137, 0.047, 0.131, 0.115, 0.0...
3	0.173568	0.20405	0.850614	[0.033, 0.139, 0.04, 0.023, 0.128, 0.132, 0.15...
4	0.187301	0.199886	0.937041	[0.172, 0.082, 0.111, 0.174, 0.001, 0.101, 0.1...

Fig 16.: Code snippet for saving results in dataframe and results display

Next step involves finding optimum porfolios as per return and risk appetite.

Three combinations of portfolios are chosen for display viz:minimum risk,maximum return and maximum sharpe keeping other two measures unconstrained during the search.

```
min_risk=portfolio_df.iloc[portfolio_df['Risk'].astype(float).idxmin()]
max_return=portfolio_df.iloc[portfolio_df['Return'].astype(float).idxmax()]
max_sharpe=portfolio_df.iloc[portfolio_df['Sharpe'].astype(float).idxmax()]
```

Fig 17.Code Snippet for selection of minimum risk and maximum return, Sharpe portfolio

These are printed in next step get combination of return,risk and sharpe for above conditions.
First output relates to minimum risk constraint.

```
Minimun Risk
Return                                          0.170258
Risk                                            0.183006
Sharpe                                          0.930341
Weights    [0.111, 0.011, 0.08, 0.16, 0.122, 0.028, 0.09,...
Name: 30, dtype: object
['HINDUNILVR.NS', 'ADANIPORTS.NS', 'ASIANPAINT.NS', 'BHARTIARTL.NS', 'NTPC.NS', 'RELIANCE.NS', 'SBIN.NS', 'SUNPHARMA.NS', 'TCS.N
S', 'ULTRACEMCO.NS']
```

Fig 18.Code Snippet minimum risk output

Here the risk is 18.3% and return is 17% which is suboptimal.
These are printed in next step get combination of return,risk and sharpe for above conditions.
First output relates to maximum return constraint.

```
Maximum Return
Return                                          0.196958
Risk                                            0.200756
Sharpe                                          0.981081
Weights    [0.109, 0.228, 0.138, 0.008, 0.079, 0.141, 0.0...
Name: 41, dtype: object
['HINDUNILVR.NS', 'ADANIPORTS.NS', 'ASIANPAINT.NS', 'BHARTIARTL.NS', 'NTPC.NS', 'RELIANCE.NS', 'SBIN.NS', 'SUNPHARMA.NS', 'TCS.N
S', 'ULTRACEMCO.NS']
```

Fig 19.Code Snippet maximum return output

Here the return is 20% and risk is 19.69%,as the return increases so does the risk.
Finally the combination of return,risk and sharpe is received for maximum return constraint.

```
Maximum Sharpe
Return                                          0.194487
Risk                                            0.190158
Sharpe                                           1.02276
Weights    [0.081, 0.094, 0.174, 0.178, 0.037, 0.102, 0.0...
Name: 8, dtype: object
['HINDUNILVR.NS', 'ADANIPORTS.NS', 'ASIANPAINT.NS', 'BHARTIARTL.NS', 'NTPC.NS', 'RELIANCE.NS', 'SBIN.NS', 'SUNPHARMA.NS', 'TCS.N
S', 'ULTRACEMCO.NS']
```

Fig 20.Code Snippet maximum Sharpe ratio output

Here the return is 19.44% and risk is 19.0%, and sharpe ratio value is maximum i.e 1.02.

The portfolio weights are seen concentrated under five stocks Adaniports-9.4%,Asianpaints 17.4,Bhartiairtl at 17.8%,Reliance at 10% and Tcs at 16% these sum upto 70% of total weights.

Table 7.MVO based portfolio weights

Stock	Portfolio Weight in percentage
HINDUNILVR.NS	8
ADANIPORTS.NS	9.4
ASIANPAINT.NS	17.4
BHARTIARTL.NS	17.8
NTPC.NS	3.7
RELIANCE.NS	10.2
SBIN.NS	6.9
SUNPHARMA.NS	5.7
TCS.NS	16
ULTRACEMCO.NS	4.9

On the lower end ntpc,ultracemco and sunpharma have only 14% weights.There is clear concentration on weights on certain stocks.

The efficient frontier achieved is visualised through Matplotlib.Efficient frontier is a two dimensional space where the effort is to get maximum return with a minimum risk.The aim is to find a point with lower value of x i.e volatility and higher value of y i.e return.

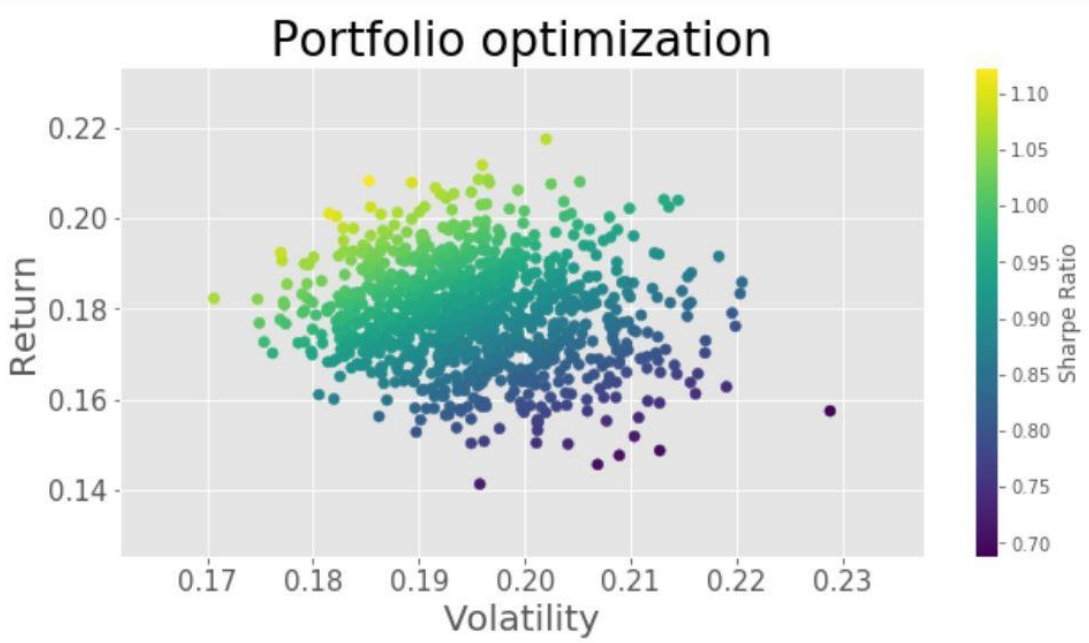


Fig 21.Efficient portfolio visualisation in Matplotlib

Place on the efficient portfolio graph which is desirable is closer to y(lower Volatility value) axis with higher value of y(higher return)

Another simulation is done for portfolio optimisation of these ten representative stocks through an open source python package by the name PyPortfolioOpt.This package is used to arrive at efficient frontier for maximum sharpe ratio.

Pyportfolioopt has inbuilt functions to calculate historical annualised return and covariance.

```
from pypfopt.efficient_frontier import EfficientFrontier
from pypfopt import risk_models
from pypfopt import expected_returns

# Calculate expected returns and sample covariance
mu = expected_returns.mean_historical_return(df)
S = risk_models.sample_cov(df)
```

Fig 22.Code Snippet pypfopt function call

In further steps the sharpe ratio is maximised to achieve optimum portfolio.

```
# Optimize for maximal Sharpe ratio
ef = EfficientFrontier(mu, S)
raw_weights = ef.max_sharpe()
cleaned_weights = ef.clean_weights()
ef.save_weights_to_file("weights.csv")  # saves to file
print(cleaned_weights)
ef.portfolio_performance(verbose=True)
```

Fig 23.Code Snippet for Sharpe ratio maximisation

The return and sharpe ratio are higher than the monte carlo random simulation method.

```
OrderedDict([('HINDUNILVR.NS', 0.16947), ('ADANIPORTS.NS', 0.0), ('ASIANPAINT.NS', 0.2899), ('BHARTIARTL.NS', 0.0), ('NTPC.NS',
0.0), ('RELIANCE.NS', 0.14507), ('SBIN.NS', 0.0), ('SUNPHARMA.NS', 0.01707), ('TCS.NS', 0.3785), ('ULTRACEMCO.NS', 0.0)])
Expected annual return: 21.9%
Annual volatility: 19.3%
Sharpe Ratio: 1.03
```

Fig 24.Code pypfopt MVO optimisation output

Resulting return using this method is 21.8% for the portfolio allocation.

Table 8.Stock weights in predicted portfolio

Stock	Portfolio Weight in percentage
HINDUNILVR.NS	16.92
ADANIPORTS.NS	0
ASIANPAINT.NS	28.9
BHARTIARTL.NS	0
NTPC.NS	0
RELIANCE.NS	14.59
SBIN.NS	0
SUNPHARMA.NS	1.71
TCS.NS	37.88
ULTRACEMCO.NS	0

Here interestingly the portfolio allocation is even skewed to only 4 stocks weighting almost 98%.This is typical of traditional MVO method.A Closer look at stock visualisation table N will reveal that there is a comparatively higher variance in left out stocks.

For second simulation using where data is split into pre and post covid timeframes pypfopt package is used.For Pre-covid simulation MVO method based run on test sample.Results from MVO method for pre-covid period is viz. Expected annual return: 31.2%,Annual volatility: 14.3%,Sharpe Ratio: 2.04.The returns here are higher than the previous experiment where combined dataset was used.

Table 9.Portfolio weights through MVO method for pre-covid time.

Stock	Portfolio Weight in percentage
HINDUNILVR.NS	41.40
ADANIPORTS.NS	0
ASIANPAINT.NS	12.8
BHARTIARTL.NS	0
NTPC.NS	0
RELIANCE.NS	10.8
SBIN.NS	0
SUNPHARMA.NS	0
TCS.NS	35
ULTRACEMCO.NS	0

For Post-covid simulation MVO method based run on data till validation sample.Results from MVO method for pre-covid period is viz. Expected annual return: 41.3%,Annual volatility: 23.5%,Sharpe Ratio: 1.67.The returns here are higher than the previous experiment where combined dataset was used.The portfolio distribution for post covid time is completely different from pre-covid so much so that the maximum weight recommended shares do not even find a place portfolio.Hindusunilvr get 0% from a lion's share of 41.4%,Sunpharma on the other hand which was 0% gets a whopping 45%.

Table 10.Portfolio weights through MVO method for post-covid time.

Stock	Portfolio Weight in percentage
HINDUNILVR.NS	0
ADANIPORTS.NS	7.6
ASIANPAINT.NS	21.4
BHARTIARTL.NS	0
NTPC.NS	0
RELIANCE.NS	0
SBIN.NS	0
SUNPHARMA.NS	45

TCS.NS	26
ULTRACEMCO.NS	0

4.5 Challenger models implementation in datarobot

Challenger models in datarobot being machine learning models will be fed with data enriched with features relating to RSI and SMA are created in colab.

“talib” a custom package in python is used to calculate SMA and RSI for time periods 14,30,50 and 200.These new features are appended column wise to the original dataset.

```
ticker= "HINDUNILVR.NS"
stock_Hul = yf.download(ticker, start="2017-10-26", end="2022-06-10")###Past 4.5 years data Train sample
feature_names_Hul = []
for n in [14, 30, 50, 200]:
    stock_Hul['ma' + str(n)] = talib.SMA(stock_Hul['Adj Close'].values, timeperiod=n)
    stock_Hul['rsi' + str(n)] = talib.RSI(stock_Hul['Adj Close'].values, timeperiod=n)

    feature_names_Hul = feature_names_Hul + ['ma' + str(n), 'rsi' + str(n)]
stock_Hul.tail()
```

Fig 25.Code Snippet for SMA,RSI feature creation

Which leads to new data set with increased features.

```
[**********100%**********]  1 of 1 completed
```

Date	Open	High	Low	Close	Adj Close	Volume	ma14	rsi14	ma30	rsi30	ma50	rsi50	ma200	rsi200
2022-06-03	2284.000000	2324.949951	2265.149902	2291.949951	2251.781982	1845303	2267.638375	54.421870	2195.491138	53.948856	2139.851643	52.070046	2312.413569	50.322333
2022-06-06	2272.199951	2301.949951	2247.600098	2280.500000	2240.532715	2131405	2270.129656	52.929968	2200.519767	53.280637	2145.784805	51.668699	2311.876581	50.208549
2022-06-07	2250.000000	2259.850098	2207.050049	2211.600098	2172.840332	2034730	2264.529576	44.945388	2204.150008	49.466578	2150.612686	49.333808	2310.894207	49.531235
2022-06-08	2215.000000	2218.949951	2186.000000	2197.050049	2158.545166	1506843	2259.792637	43.454586	2205.744889	48.704914	2155.408142	48.858037	2309.560107	49.389826
2022-06-09	2197.000000	2208.350098	2170.750000	2197.699951	2159.183838	1497867	2250.792480	43.544683	2207.467489	48.741392	2159.782119	48.880512	2308.218218	49.396313

Fig 26.Code Snippet for new dataset display

New features added are ma14,ma30,ma50 and ma200 for the moving average,for RSI new features added were rsi14,rsi30,rsi50,rsi200.

Adding Moving average helps in smoothening the data and removes unwanted noise.

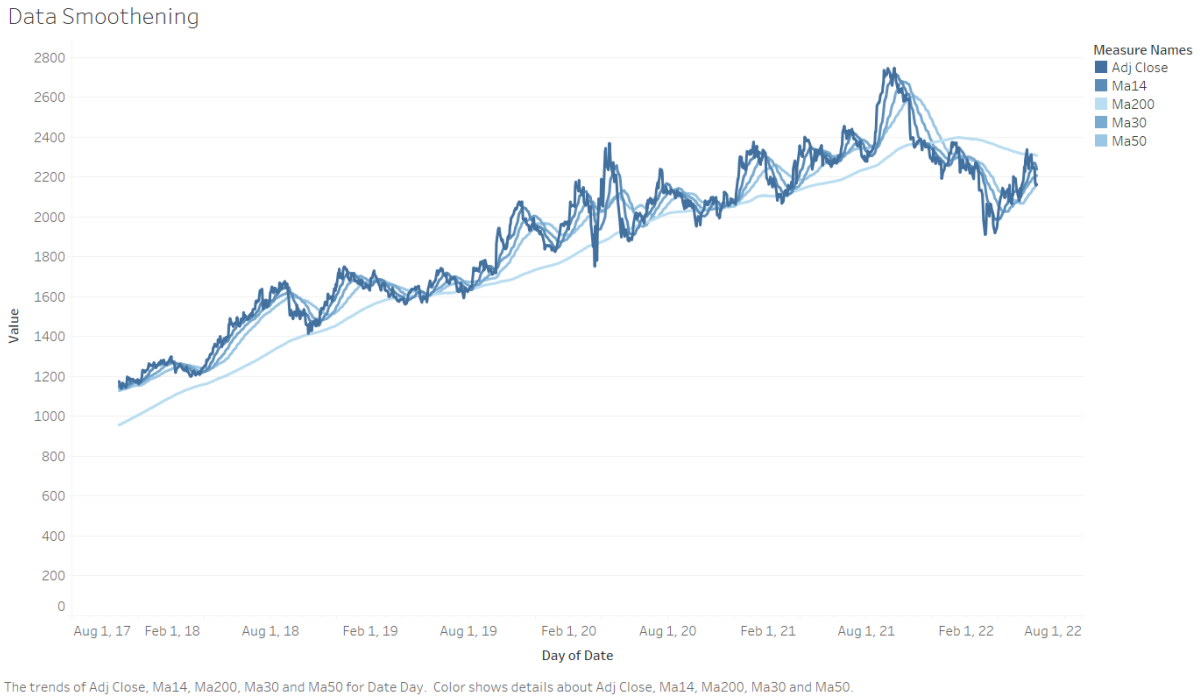


Fig 27.Smoothened Data

Resulting data is split onto train,validation and test dataset.Each set having 1000,70 and 76 records respectively.A total of 10 data sets are created for which 5 models each are created.This results in a total of 50 models being created.Stock data is loaded into AutoML tool datarobot to create a model pipeline.Certain important Settings relating to sampling are made by creating a “partition” feature in dataset.

Table 11.Train-Validation-Test split datewise

Feature name	Data	Count	Time
Partition	Train	1000	26/10/2017-11/11/2021
Partition	Val	76	12/11/2021-28/02/2022
Partition	Test	70	02/03/2022-31-05-2022

These setting are saved through patitioning tab in additional properties.

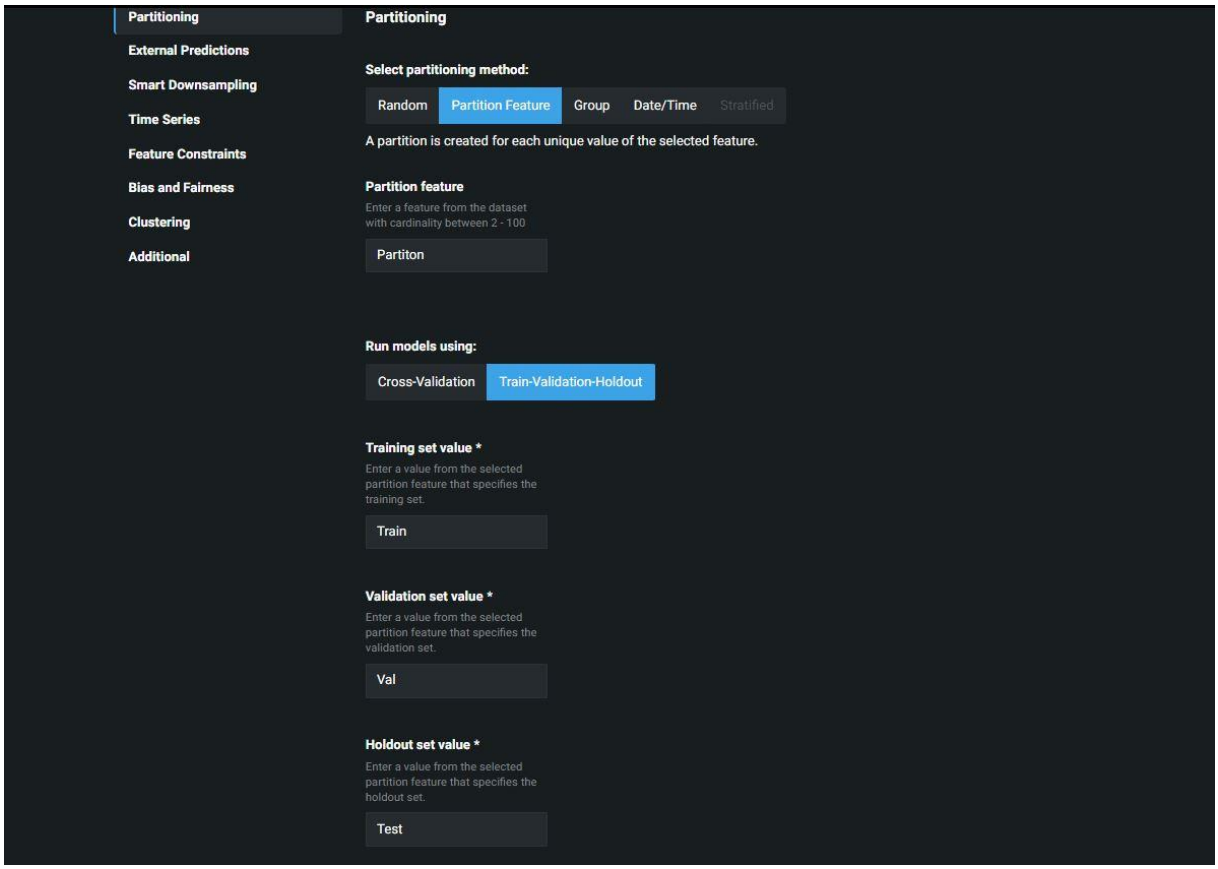


Fig 28.Data partitioning properties in datarobot

As mentioned in methodology MAPE is the metrics of choice thus optimization metric is chosen as MAPE,blender models are allowed and models are created with Data leakage features removed.

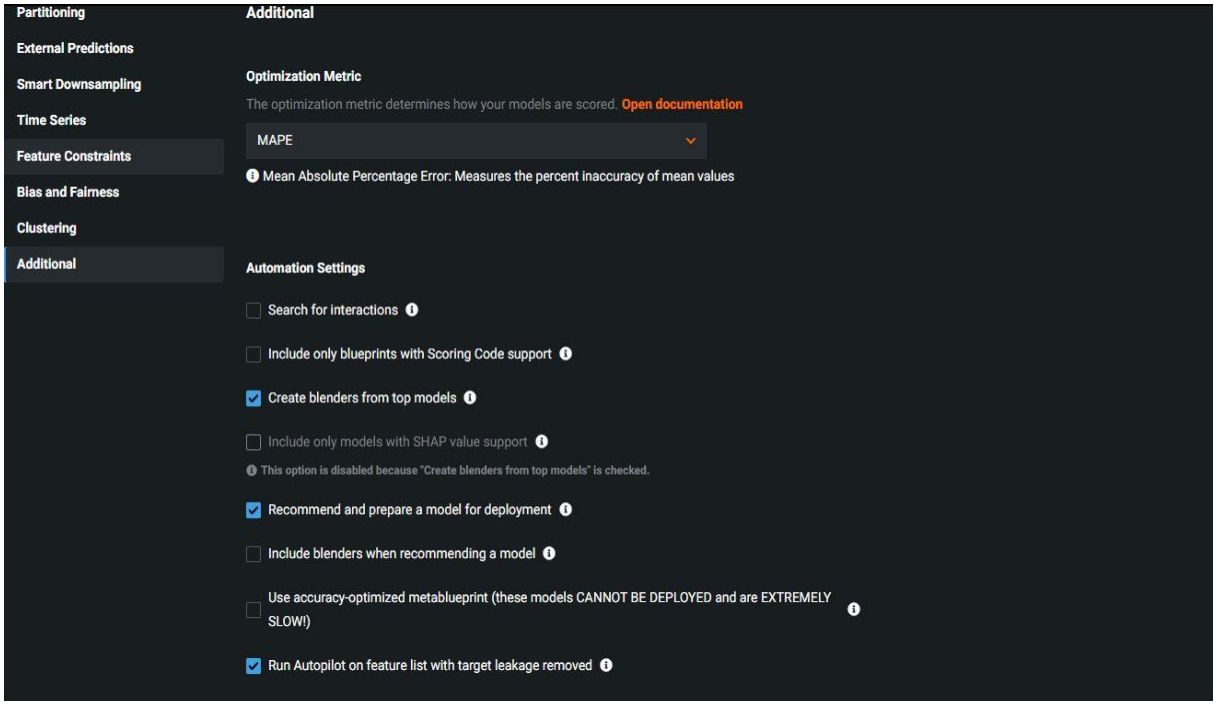


Fig 29.Additional properties setting for optimization metric

Highly correlated features are removed,certain time based features like Date,Month are also removed.A new feature list is created and saved for model building phase to begin.

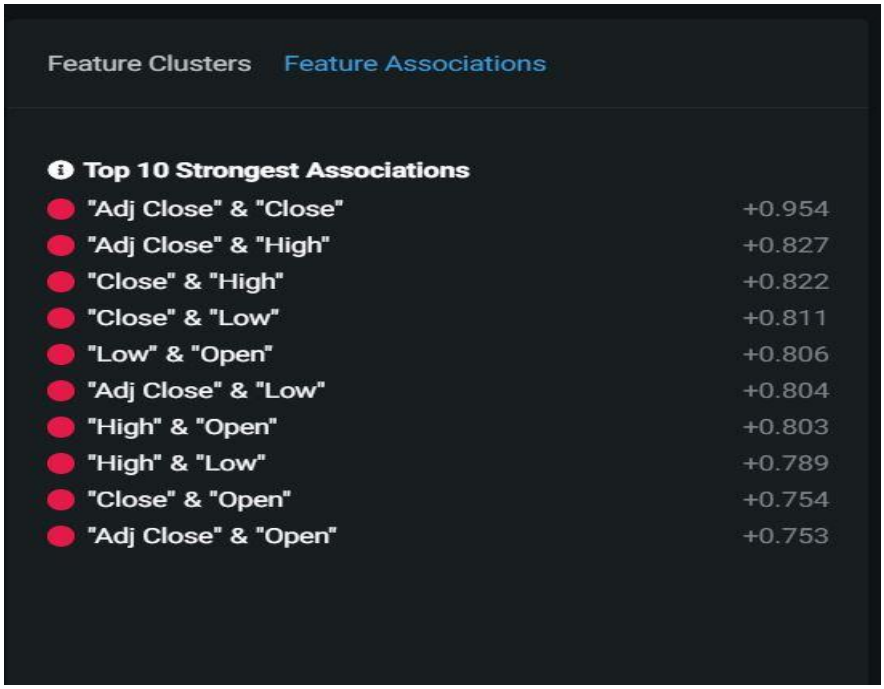
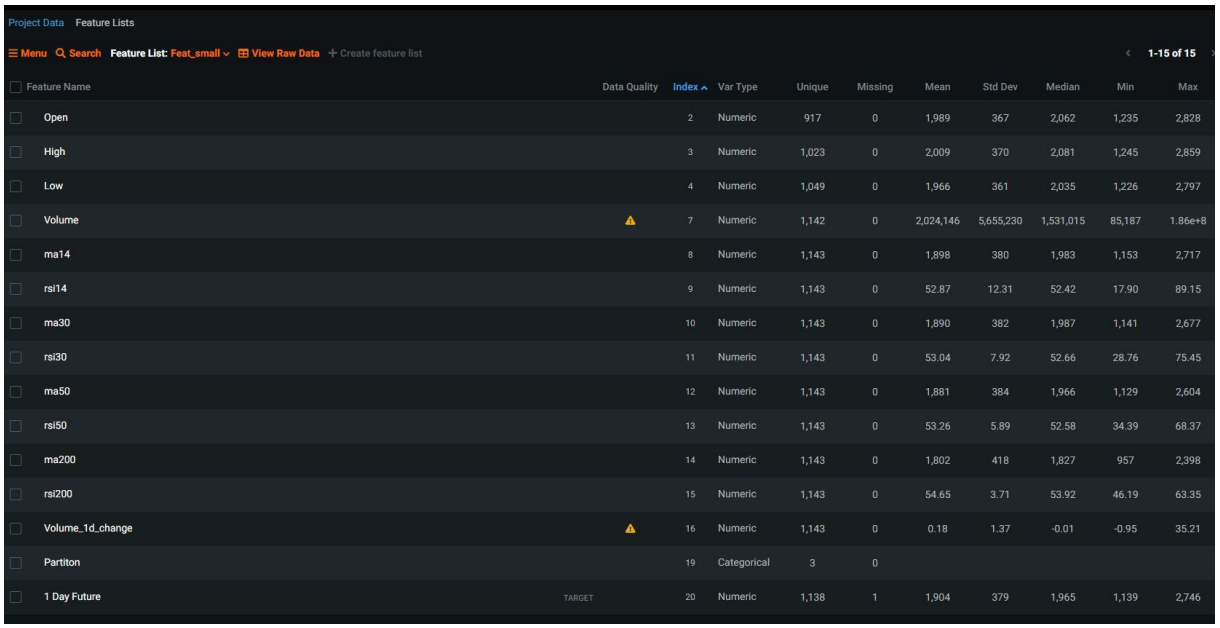


Fig 30.Feature association

Feature that have high correlation are removed, prominent features that are correlated are in various pairs.The feature Adj.Close and close have a high feature association.These two features in atleast 7 pairs of the top 10,they are removed from the list of features that are used in model building.

Final feature list is a reduced list of 13 features that is arrived at after feature elimination is displayed in figure 30.



Project Data Feature Lists

Menu Search Feature List: Feat_small View Raw Data Create feature list 1-15 of 15

Feature Name	Data Quality	Index	Var Type	Unique	Missing	Mean	Std Dev	Median	Min	Max
Open		2	Numeric	917	0	1,989	367	2,062	1,235	2,828
High		3	Numeric	1,023	0	2,009	370	2,081	1,245	2,859
Low		4	Numeric	1,049	0	1,966	361	2,035	1,226	2,797
Volume	⚠	7	Numeric	1,142	0	2,024,146	5,655,230	1,531,015	85,187	1.86e+8
ma14		8	Numeric	1,143	0	1,898	380	1,983	1,153	2,717
rsi14		9	Numeric	1,143	0	52.87	12.31	52.42	17.90	89.15
ma30		10	Numeric	1,143	0	1,890	382	1,987	1,141	2,677
rsi30		11	Numeric	1,143	0	53.04	7.92	52.66	28.76	75.45
ma50		12	Numeric	1,143	0	1,881	384	1,966	1,129	2,604
rsi50		13	Numeric	1,143	0	53.26	5.89	52.58	34.39	68.37
ma200		14	Numeric	1,143	0	1,802	418	1,827	957	2,398
rsi200		15	Numeric	1,143	0	54.65	3.71	53.92	46.19	63.35
Volume_1d_change	⚠	16	Numeric	1,143	0	0.18	1.37	-0.01	-0.95	35.21
Partiton		19	Categorical	3	0					
1 Day Future	TARGET	20	Numeric	1,138	1	1,904	379	1,965	1,139	2,746

Fig 31. Feature list for modelling

Top models are chosen from the leaderboard on the basis of MAPE,also models with significant difference in validation and test error are excluded from use as this is an indicator of model overfitting.Top models on the leaderboard that have similar MAPE errors for validation and test are Eureqa regressors, Their MAPE are in the range of 1.2 %.Further details will be discussed in Results and conclusion section.

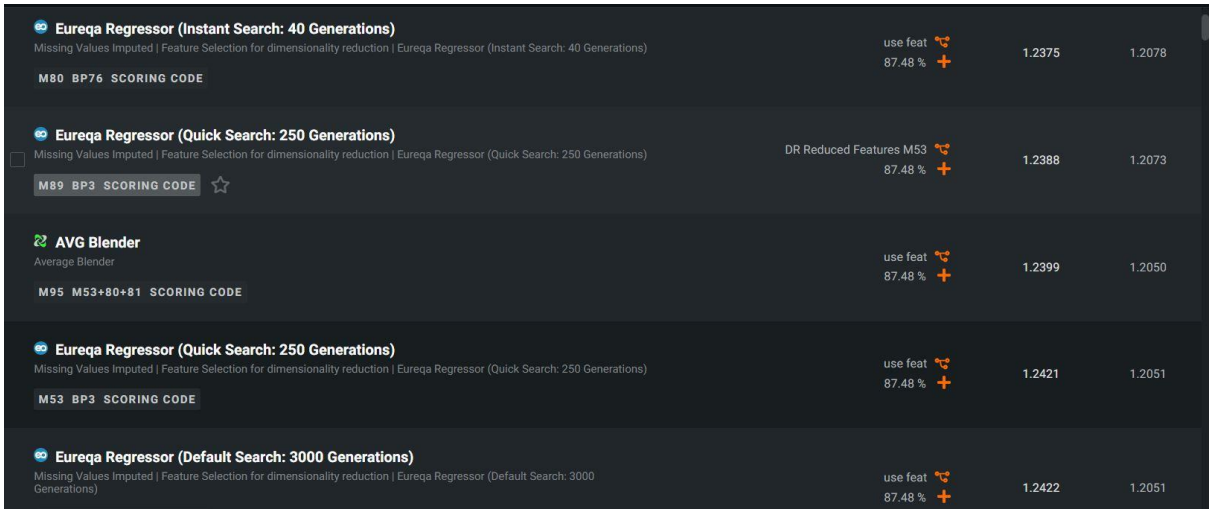


Fig 32. Leader board for stock Adaniports

Feature importance is analysis is important to understand the impact of features on target variable.This helps in intuitive understanding of data behavoiur and provided scope for performance improvement.

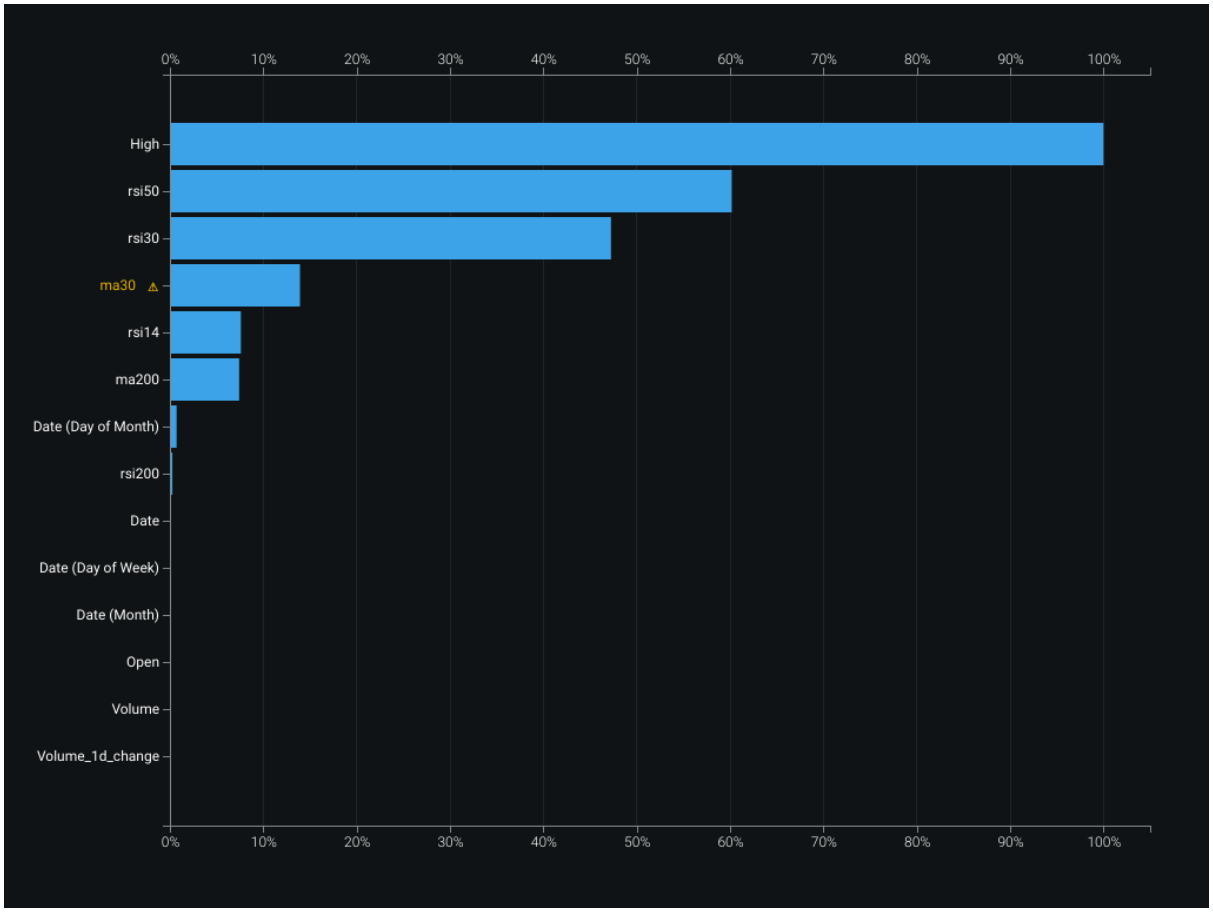


Fig 33.Feature importance chart

Through similar method five top models were analysed for each of the ten stocks.Out of which top model for each stock was selected to create an optimum portfolio on the basis of their future return and compared with baseline MVO method in next section.

4.6 Summary

Portfolio optimisation was implemented key steps involving data downloading from Yahoo Finance for ten representative stocks.Data used in ML models was enriched through feature engineering which led to data smoothening and extracting more information from raw data.Past four and half years data was used for training the supervised ML models,remaining 2 months each data was used for validation and testing,Top five models with minimum MAPE developed in datarobot for ten stocks were analysed.Models with similar validation and test errors were chosen for portfolio selection through weighted method.

Mean variance optimization method was implemented in python through two approaches one being a Monte-Carlo simulation method and other using python package pyportfoliopt.Monte-Carlo simulation method gave portfolio weignts for stock for in scenarios of minimum risk-return,maximum return-risk and maximum sharpe ratio.The python method was used to maximise Sharpe ratio to arrive at portfolio weights.

CHAPTER 5

RESULTS AND EVALUATION

5.1 Introduction

Portfolio allocation received from both traditional and ML based techniques would be assessed here.Portfolio allocation from the pyportfoliopt method will be used for MVO method allocation given it returns are higher than Monte-Carlo method.For ML models top model will be decided for each stock on the basis of MAPE and generalisation.The simulation methodology defined in methodology section will be used to create multiple forecasts

5.2 Models from Datarobot

Datarobot provides various metric related to a particular model which might be helpful in choosing that particular model from the pipleline of models.

Table 12.Leader board for stock Asianpaints.

Rank	Modeling Results	Model Type
1	FVE Gamma: [0.73608], FVE Poisson: [0.7388], FVE Tweedie: [0.73746], Gamma Deviance: [0.00039], Gini Norm: [0.8611], MAE: [47.28689], MAPE: [1.47834], Poisson Deviance: [1.24622], Quantile Loss: [23.64345], R Squared: [0.74135], RMSE: [63.4645], RMSLE: [0.01974], SMAPE: [1.46489], Tweedie Deviance: [0.02193], labels: ['(0,-1)'], metrics: ['FVE Gamma', 'FVE Poisson', 'FVE Tweedie', 'Gamma Deviance', 'Gini Norm', 'MAE', 'MAPE', 'Poisson Deviance', 'Quantile Loss', 'R Squared', 'RMSE', 'RMSLE', 'SMAPE', 'Tweedie Deviance']	Ridge Regressor

2	FVE Gamma: [0.40807], FVE Poisson: [0.40544], FVE Tweedie: [0.40679], Gamma Deviance: [0.00087], Gini Norm: [0.85221], MAE: [73.49674], MAPE: [2.22507], Poisson Deviance: [2.83675], Quantile Loss: [36.74837], R Squared: [0.40254], RMSE: [96.45654], RMSLE: [0.0292], SMAPE: [2.25947], Tweedie Deviance: [0.04955], labels: ['(0,-1)'], metrics: ['FVE Gamma', 'FVE Poisson', 'FVE Tweedie', 'Gamma Deviance', 'Gini Norm', 'MAE', 'MAPE', 'Poisson Deviance', 'Quantile Loss', 'R Squared', 'RMSE', 'RMSLE', 'SMAPE', 'Tweedie Deviance']	Gradient Boosted Trees Regressor (Least-Squares Loss)
3	FVE Gamma: [0.10938], FVE Poisson: [0.1117], FVE Tweedie: [0.11057], Gamma Deviance: [0.0013], Gini Norm: [0.8493], MAE: [95.64513], MAPE: [2.89405], Poisson Deviance: [4.23827], Quantile Loss: [47.82257], R Squared: [0.11376], RMSE: [117.47751], RMSLE: [0.03576], SMAPE: [2.95282], Tweedie Deviance: [0.0743], labels: ['(0,-1)'], metrics: ['FVE Gamma', 'FVE Poisson', 'FVE Tweedie', 'Gamma Deviance', 'Gini Norm', 'MAE', 'MAPE', 'Poisson Deviance', 'Quantile Loss', 'R Squared', 'RMSE', 'RMSLE', 'SMAPE', 'Tweedie Deviance']	RandomForest Regressor

4	FVE Gamma: [-0.04275], FVE Poisson: [-0.02322], FVE Tweedie: [-0.03285], Gamma Deviance: [0.00153], Gini Norm: [0.69897], MAE: [110.67931], MAPE: [3.37966], Poisson Deviance: [4.88197], Quantile Loss: [55.33966], R Squared: [-0.00472], RMSE: [125.08338], RMSLE: [0. 03874], SMAPE: [3.44889], Tweedie Deviance: [0.08628], labels: ['(0,-1)'], metrics: ['FVE Gamma', 'FVE Poisson', 'FVE Tweedie', 'Gamma Deviance', 'Gini Norm', 'MAE', 'MAPE', 'Poisson Deviance', 'Quantile Loss', 'R Squared', 'RMSE', 'RMSLE', 'SMAPE', 'Tweedie Deviance']	Auto-tuned K-Nearest Neighbors Regressor (Euclidean Distance)
5	FVE Gamma: [0.28978], FVE Poisson: [0.28814], FVE Tweedie: [0.28895], Gamma Deviance: [0.00104], Gini Norm: [0.87199], MAE: [81.69133], MAPE: [2.47178], Poisson Deviance: [3.39644], Quantile Loss: [40.84566], R Squared: [0.28624], RMSE: [105.42774], RMSLE: [0.03195], SMAPE: [2.51578], Tweedie Deviance: [0.0594], labels: ['(0,-1)'], metrics: ['FVE Gamma', 'FVE Poisson', 'FVE Tweedie', 'Gamma Deviance', 'Gini Norm', 'MAE', 'MAPE', 'Poisson Deviance', 'Quantile Loss', 'R Squared', 'RMSE', 'RMSLE', 'SMAPE', 'Tweedie Deviance']	eXtreme Gradient Boosted Trees Regressor (Gamma Loss)

Herein all relevant metric related to regression problem are tracked.Important metric for regression like root mean squared error(RMSE),mean absolute error(MAE),MAPE,R squared to name a few.The models are mentioned with their names to a detail of their variant like XGBoost model has multiple variants like early stopping,learning rate 0.001 etc. herein the model gaining performance is XGBoost with Gamma loss this is due the fact that datarobot finds data to be Gamma distributed.

For each stock another test was conducted which entailed checking the difference between the expected %returns and actual %returns for the validation data set.The objective of this experiment was to use the model which resulted in maximum returns yet was generalisable.

Table 13.Top models compared for returns over validation time period for Reliance

Model	Return
Eureqa	3.53%
XGBoost	5.24%
Ridge regressor	5.86%

Post this analysis models with maximum returns were chosen from the subset of models are obtained for all 10 stocks.However for this experiment the top model is to be chosen. Top models from Datarobot for all 10 stocks are selected for portfolio simulations.

Table 14.Top models for each of 10 stocks with their return of validation set.

Nify50 Stock	ML Model	MAPE
Adani Ports and Special Economic Zone Ltd.:ADANIPORTS	Lasso regressor	1.2738
Asian Paints Ltd.:ASIANPAINT	Eureqa Regressor(3000 Generations)	1.2431
Bharti Airtel Ltd.:BHARTIARTL	Ridge regressor	1.4783
Hindustan Unilever Ltd.:HINDUNILVR	Eureqa Generalised Addtive model(1000 Generations)	0.8799

NTPC Ltd.:NTPC	ExtraTrees Regressor	1.6404
Reliance Industries Ltd.:RELIANCE	AVG Blender(Ridge regressor,Keras Slim Residual Neural Network)	1.3332
State Bank of India:SBIN	Eureqa Regressor(3000 Generations)-M93	1.3099
Sun Pharmaceutical Industries Ltd.:SUNPHARMA	Eureqa Regressor(3000 Generations)-M81	1.5383
Tata Consultancy Services Ltd.:TCS	Eureqa Generalised Addtive model(1000 Generations)	1.6781
UltraTech Cement Ltd.:ULTRACEMCO	Eureqa Regressor(Instant search:40 Generation)	1.2375

5.3 Simulation results

Predictions from top models for the holdout set for each stock are arranged excel calculator to get the returns.These weights are adjusted dynamically for each days predictions.

Such calculations are done for all the stocks in portfolio and returns obtained.Similarly the MVO based returns are calculated for weights arrived through MVO optimisation in Table 8 .

5.3.1 First simulation

Average daily returns for the period 02/03/2022 to 09/06/2022 are compared with Actuals as well both the methods.

Table 15.Average daily return per stock for holdout sample

Nify50 Stock	Average daily return for forecast based weights for ML model	Average daily return for MVO based weights for ML model	Actual average daily return
ADANIPORTS	-0.1	0	-0.2
ASIANPAINTS	-0.64	-1.5	-0.75
BHARTIARTL	0.21	0	0.01
HINDUNILVR	-1.93	-2.5	-1.81
NTPC	0.15	0	0.08
RELIANCE	0.20	0.23	0.28
SBIN	-1.53	0	-1.32
SUNPHARMA	-0.68	-0.57	0.09
TCS	-0.5	-3.2	-0.6
ULTRACEMCO	-0.48	0	-0.33

The results show that the general trend of market during the holdout phase for selected stocks is bearish with average daily returns being negative for most of the stocks.There perceptible difference between the performance of ML model based returns in comparison to MVO model based returns.

The ML based model predicts 9 out of 10 directions of average daily returns correctly in comparison to MVO which only predict 5 out 10 directions of average daily return.Cumulative error in prediction of daily average return for the ML based model is 1.5% whereas MVO weight based method has cumulative error of 6.5%

One important feature of ML based model's output is that it is effectively able to predict both the positive and negative direction of stock movement which is useful for short-selling the stocks.This is unique to ML based models however MVO based method used here does assume short selling as possible trade.

5.3.2 Second simulation

The results for second simulation where data is split into pre-covid time frame is

Table 16Average daily return per stock for holdout sample

Nify50 Stock	Average daily return for forecast based weights for ML model	Average daily return for MVO based weights for ML model	Actual average daily return
ADANIPORTS	-0.1	0	-0.2
ASIANPAINTS	-2.73	-0.9	-2.53
BHARTIARTL	-1.8	0	-0.4
HINDUNILVR	-4.48	-3.7	-5.54
NTPC	-14.45	0	-17.1
RELIANCE	-1.46	-0.7	-1.8
SBIN	-4.1	0	-3.38
SUNPHARMA	-4.5	0.1	-4.05
TCS	-4.9	-1.2	-4.5
ULTRACEMCO	-1.5	0	-1.2

Interestingly the time period selected for as the holdout sample 16/12/2019 to 17/02/2020 saw market declines due to global trend.Thus majority trends come negative for actual returns as well.Profits could be made by short selling here too.

The ML based model predicts 10 out 10 directions of average daily returns correctly in comparison to MVO weight based method which only predict 4 out 10 directions of average daily return.Cumulative error in prediction of daily average return for the ML based model is 7.62% whereas MVO weight based method has cumulative error of 34.3%

The results for second simulation where data is split into post-covid time frame is

Table 17.Average daily return per stock for holdout sample

Nify50 Stock	Average daily return for forecast based weights for ML	Average daily return for MVO based weights for ML model	Actual average daily return

	model		
ADANIPORTS	-2.75	0	-0.58
ASIANPAINTS	0.77	1.86	-0.732
BHARTIARTL	0.74	0	-0.2
HINDUNILVR	-1.66	-3.77	-1.72
NTPC	0.15	0	-0.1
RELIANCE	-1.1	0.50	0.15
SBIN	-1.53	0	-1.32
SUNPHARMA	1.2	2.3	-0.6
TCS	-0.5	0.8	-0.7
ULTRACEMCO	-0.48	0	-0.33

Time period selected for as the holdout sample 05/04/2022 to 09/06/2022,this period also had negative daily average return for 90% stocks in the selected portfolio.Here the performance of ML based model did not perform as well as it was able to correctly predict actual directions of only 5 out 10 stock correctly. Cumulative error in prediction of daily average return for the ML based model is 8.5% whereas MVO weight based method has cumulative error of 11.5%.

The study was conducted in a scenario that saw multiple ups and down in the market.Towards the end of quarter one in 2020 there is a big dip in market,this dip started during January 2020

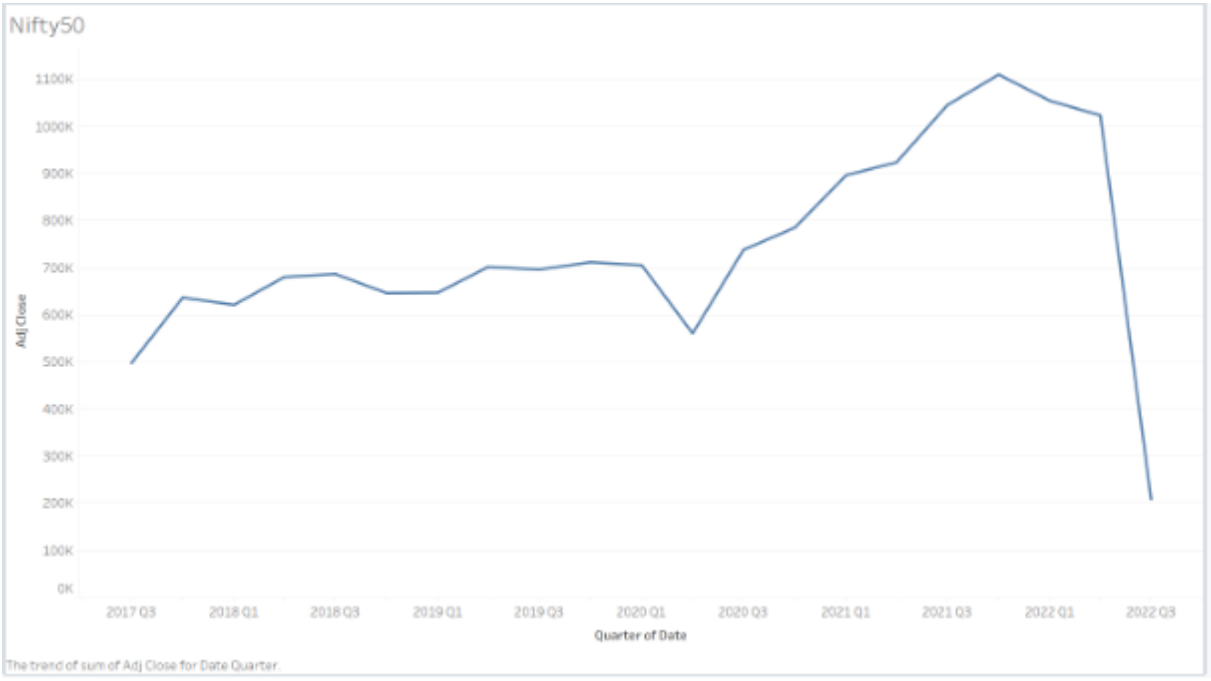


Fig 34. Nifty50 Chart

And continued quarter 2 on 2020.Similar fall has been seen in the month may-june of 2022.
This period has seen daily negative returns.However the method developed in this study was correct more often in predicting the direction of stock movement.

5.4 Summary

Stock forecasting is a complex problem which can draw couple surprises.It would be evident from common knowledge that the markets fell during past 5 years but the consistency of negative returns point to the possibility of profit on short selling side more.The study also showed the daily weight adjusted ML forecasting methods coming closer to actual returns.ML forecast based weights simulation and MVO weight based simulations were comapared.ML forecast based simulation was better performing than the baseline method.
During covid crisis simulation also ML weight based method placed investor in better position with with cumulative errors being 7.6% and 8.52% in pre-covid and post covid simulations.

CHAPTER 6

CONCLUSION AND RECOMMENDATION

6.1 Introduction

Research conducted to compare traditional mean variance optimisation method and Machine learning based method algorithmic trading presented multiple insights and findings.The study systemtically analysed 10 representative stocks for past four and half years in first simulation.

In the second simulation data was split into pre-covid and post covid periods taking two year window at both ends.More than 100 models were built in through AutoMl tool datarobot out of which 10 were used in first simulation and 20 were used second simulation.

6.2 Discussion and Conclusion

The results were in favour of dynamic stock forecasting with daily weights adjustment method.This provides a cue into the complexity of stock forecasting and portfolio optimisation problem.Important finding of the research was that given the consistent negative returns in the scenario of study ML based forecasting method can be used for short selling.A more complex strategy with certain dynamic threshold rules could help in better returns.

Lack of hourly or more granular data is a limitation as the more information could be derived through these dynamic patterns.Data enrichment through use of macroeconomic indicators and sentiment data could help in improving the ML method more as in current study only price trends turned out be significant variables.

6.3 Contributions

Research was able to meet the objective developing ML based trading strategy and comparing it with traditional MVO based method.Covid scenario simulation was also done.This study confirms the previous assertions of other studies where ML based especially ensemble models were found to be useful in predicting direction of stock.This study also showed forecasting stock prices with good accuracy.

6.4 Future work

One area of future work is having dynamic MVO weights that get updated daily as static MVO weights were used in the study.Another promising are would be datasource augmentation with more daily macro-economic and social media feeds.

REFERENCES

Chen, W. *et al.* (2021) 'Mean–variance portfolio optimization using machine learning-based stock price prediction', *Applied Soft Computing*, 100, p. 106943. doi:10.1016/j.asoc.2020.106943.

Dey, S. *et al.* (2016) 'Forecasting to Classification : Predicting the direction of stock market price using Xtreme Gradient Boosting Forecasting to Classification : Predicting the direction of stock market price using Xtreme Gradient Boosting', (October), pp. 1–10. doi:10.13140/RG.2.2.15294.48968.

Doering, J. *et al.* (2019) 'Metaheuristics for rich portfolio optimisation and risk management : Current state and future trends', *Operations Research Perspectives*, 6(February), p. 100121. doi:10.1016/j.orp.2019.100121.

Enke, D. and Thawornwong, S. (2005) 'The use of data mining and neural networks for forecasting stock market returns', *Expert Systems with Applications*, 29(4), pp. 927–940. doi:10.1016/j.eswa.2005.06.024.

Gökgöz, F. and Atmaca, M.E. (2012) 'Financial optimization in the Turkish electricity market : Markowitz ' s mean-variance approach', *Renewable and Sustainable Energy Reviews*, 16(1), pp. 357–368. doi:10.1016/j.rser.2011.06.018.

Jiang, W. (2021) 'Applications of deep learning in stock market prediction: Recent progress', *Expert Systems with Applications*, 184(June), p. 115537. doi:10.1016/j.eswa.2021.115537.

Jobson, J.D. and Korkie, B. (1980) 'Estimation for Markowitz efficient portfolios', *Journal of the American Statistical Association*, 75(371). doi:10.1080/01621459.1980.10477507.

Kalayci, C.B., Polat, O. and Akbay, M.A. (2020) 'An efficient hybrid metaheuristic algorithm for cardinality constrained portfolio optimization', *Swarm and Evolutionary Computation*, 54(February), p. 100662. doi:10.1016/j.swevo.2020.100662.

Khaidem, L., Saha, S. and Dey, S.R. (2016) 'Predicting the direction of stock market prices using random forest', 00(00), pp. 1–20. Available at: http://arxiv.org/abs/1605.00003.

Krauss, C., Anh, X. and Huck, N. (2017) 'Deep neural networks , gradient-boosted trees , random forests : Statistical arbitrage on the S & P 500 R', *European Journal of Operational Research*, 259(2), pp. 689–702. doi:10.1016/j.ejor.2016.10.031.

Kumbure, M.M. *et al.* (2022) 'Machine learning techniques and data for stock market

forecasting: A literature review', *Expert Systems with Applications*, 197(April 2021), p. 116659. doi:10.1016/j.eswa.2022.116659.

Laperrière-Robillard, T., Morin, M. and Abi-Zeid, I. (2022) 'Supervised learning for maritime search operations: An artificial intelligence approach to search efficiency evaluation', *Expert Systems with Applications*, 206(November 2021), p. 117857. doi:10.1016/j.eswa.2022.117857.

Li, Y. *et al.* (2020) 'The role of text-extracted investor sentiment in Chinese stock price prediction with the enhancement of deep learning', *International Journal of Forecasting*, 36(4), pp. 1541–1562. doi:10.1016/j.ijforecast.2020.05.001.

Lohrmann, C. *et al.* (2018) 'A combination of fuzzy similarity measures and fuzzy entropy measures for supervised feature selection', *Expert Systems with Applications*, 110, pp. 216–236. doi:10.1016/j.eswa.2018.06.002.

Lohrmann, C. and Luukka, P. (2019) 'Classification of intraday S&P500 returns with a Random Forest', *International Journal of Forecasting*, 35(1), pp. 390–407. doi:10.1016/j.ijforecast.2018.08.004.

Malagrino, L.S., Roman, N.T. and Monteiro, A.M. (2018) 'Forecasting stock market index daily direction: A Bayesian Network approach', *Expert Systems with Applications*, 105, pp. 11–22. doi:10.1016/j.eswa.2018.03.039.

Markowitz, H. (1952) 'PORTFOLIO SELECTION', *The Journal of Finance*, 7(1), pp. 77–91. doi:10.1111/j.1540-6261.1952.tb01525.x.

Min, L. *et al.* (2021) 'Robust mean-risk portfolio optimization using machine learning-based trade-off parameter', *Applied Soft Computing*, 113, p. 107948. doi:10.1016/j.asoc.2021.107948.

Moody, J. and Saffell, M. (2001) 'Learning to trade via direct reinforcement', *IEEE Transactions on Neural Networks*, 12(4), pp. 875–889. doi:10.1109/72.935097.

Neuneier, R. (1996) 'Optimal Asset Allocation using Adaptive Dynamic Programming', *Advances in Neural Information Processing Systems 8*, 32(2), pp. 952–958. Available at: http://www.siemens.de/zfe.Jlll/homepage.html%0Ahttp://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1039198%0Ahttp://papers.nips.cc/paper/1121-optimal-asset-allocation-using-adaptive-dynamic-programming.pdf.

Niaki, S.T.A. and Hoseinzade, S. (2013) 'Forecasting S&P 500 index using artificial neural networks and design of experiments', *Journal of Industrial Engineering International*, 9(1),

pp. 1–9. doi:10.1186/2251-712X-9-1.

Nobre, J. and Neves, R.F. (2019) 'Combining Principal Component Analysis, Discrete Wavelet Transform and XGBoost to trade in the financial markets', *Expert Systems with Applications*, 125, pp. 181–194. doi:10.1016/j.eswa.2019.01.083.

De Oliveira, F.A., Nobre, C.N. and Zárate, L.E. (2013) 'Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index - Case study of PETR4, Petrobras, Brazil', *Expert Systems with Applications*, 40(18), pp. 7596–7606. doi:10.1016/j.eswa.2013.06.071.

Qiu, J., Wang, B. and Zhou, C. (2020) 'Forecasting stock prices with long-short term memory neural network based on attention mechanism', *PLoS ONE*, 15(1), pp. 1–15. doi:10.1371/journal.pone.0227222.

Rather, A.M. (2021) 'LSTM-based Deep Learning Model for Stock Prediction and Predictive Optimization Model', *EURO Journal on Decision Processes*, 9(May), p. 100001. doi:10.1016/j.ejdp.2021.100001.

Rohaan, D., Topan, E. and Groothuis-Oudshoorn, C.G.M. (2022) 'Using supervised machine learning for B2B sales forecasting: A case study of spare parts sales forecasting at an after-sales service provider', *Expert Systems with Applications*, 188(April 2021), p. 115925. doi:10.1016/j.eswa.2021.115925.

Shi, S. *et al.* (2022) 'GPM: A graph convolutional network based reinforcement learning framework for portfolio management', *Neurocomputing*, 498, pp. 14–27. doi:10.1016/j.neucom.2022.04.105.

Siami-Namini, S. and Namin, A.S. (2018) 'Forecasting Economics and Financial Time Series: ARIMA vs. LSTM', pp. 1–19. Available at: http://arxiv.org/abs/1803.06386.

Simos, T.E., Mourtas, S.D. and Katsikis, V.N. (2021) 'Time-varying Black–Litterman portfolio optimization using a bio-inspired approach and neuronets', *Applied Soft Computing*, 112, p. 107767. doi:10.1016/j.asoc.2021.107767.

Singh, V. *et al.* (2022) 'International Journal of Information Management Data Insights How are reinforcement learning and deep learning algorithms used for big data based decision making in financial industries – A review and research agenda', *International Journal of Information Management Data Insights*, 2(2), p. 100094. doi:10.1016/j.jjimei.2022.100094.

Tsai, C.F. and Hsiao, Y.C. (2010) 'Combining multiple feature selection methods for stock prediction: Union, intersection, and multi-intersection approaches', *Decision Support*

Systems, 50(1), pp. 258–269. doi:10.1016/j.dss.2010.08.028.

Vo, N.N.Y. *et al.* (2019) 'Deep learning for decision making and the optimization of socially responsible investments and portfolio', *Decision Support Systems*, 124(July), p. 113097. doi:10.1016/j.dss.2019.113097.

Weng, L. *et al.* (2020) 'Portfolio trading system of digital currencies: A deep reinforcement learning with multidimensional attention gating mechanism', *Neurocomputing*, 402, pp. 171–182. doi:10.1016/j.neucom.2020.04.004.

APPENDIX A: RESEARCH PROPOSAL

Abstract

There has been an attempt for decades to intelligently and safely operate in Equity markets. Equity Markets are a vital component of market economy. Traditionally various techniques like Mean Variance Optimisation have been tried by portfolio managers for portfolio optimization. However with the advent of Machine Learning and Artificial Intelligence in this field there have been multiple strategies that have become popular.

This research aims at exploring the existing portfolio optimisation techniques and compare them with certain modern techniques like Asset Graphs based portfolio optimisation to start with. Algorithmic Trading algorithms like Deep Q Learning (DQN) and variants of Reinforcement Learning will be tried and models will be developed, Performance of these Algorithms will be compared for metric like sharp Ratio, Annualised Return Percentage etc.

Models will be developed to have a capability to provide as early warning signal with an emphasis on saving the portfolio on Crisis like situations: Financial Credit crisis, COVID-19.

List of Figures

List of Tables

1. Background

Equity markets are a popular as well as risky investment hubs for both retail and institutional investor. Certain major Stock exchanges of the have market capitalization in Trillions of Dollars. Given the direct and indirect involvement of masses in equity market they are an important Area of Study.

Investments in the stock markets have been on the rise due to competitive economy. Mutual Funds, which put certain percentage of their investment in stock markets, have become popular with small investor recently. If we compare the data from Statistical Abstract of US from previous years we can see that not only direct ownership but also indirect ownership in form for Retirement accounts rose from 39% in 1992 to 52% in 2007.These suffice to say given these trends stock markets will increasingly become important parts of our financial investment and planning.

The growth in Information and Technology especially digital connectivity have provided platforms in this Era of globalisation where world markets are gradually being integrated in their movement, global citizens have access to all the market and opportunities around the globe. In this vein this area has witnessed keen studies early on, seminal paper of Harry Markowitz is considered by many as a systematic analysis of Portfolio Optimisation.

Problem of Portfolio selection could broadly be studied as consisting two parts, one relating to having confidence on certain stocks and second coming with a choice of their proportions to put stake on. Now traditional and Modern Portfolio optimisations theories and even some Algorithmic trading models try to solve the above problem. Multiple efforts have been made to come up with accurate prediction of future based on historical data study through Statistical methods and analysis like Mean Variance optimisation. However, these efforts are often academic in nature which difficult to reproduce for real future scenarios or too complex to implement. Thus, the current development in machine learning and deep learning frameworks provide an opportunity to efficiently model these scenarios and come up with predictive models. The same is being attempted in this research.

2. Related Research

The field of Portfolio optimization and Algorithmic trading is very dynamic. Literature review done for the research revolved around exploring the traditional as well as contemporary portfolio optimisation techniques.(Doering et al., 2019) to start with. Traditional Methods like Mean Variance Optimization were compared with a host of methods from cross functions like Heuristic and Meta Heuristic methods.(Doering et al., 2019).It was found that the drawbacks of traditional methods in terms of solving the problem in real setting as well as their intractability(Perrin, 2019) left the need to come up with newer methods. Optimization algorithms like Coordinate descent, the alternating direction method of Multipliers, the proximal gradient method and the Dykstra's algorithm were studied.(Perrin, 2019).

Work on Asset Graphs and Trees(J.-P. Onnela, A. Chakraborti, K. Kaski, J. Kertesz, 2003) that based on correlation of asset returns was studied this scheme of analysis could also benefit in providing a cue to act as early warning signal to Financial Crisis and minimising losses.

The Literature review with references is detailed below:

Reference	Problem(s)	Purpose	Algorithm
(Doering et al., 2019)	Most of real life problem in computational finance are complex and NP hard in nature. These cannot be solved as in the given framework and are often solved with simplifying assumptions.	Metaheuristic methods try to overcome drawbacks of traditional as well as Heuristic methods. Group of Methods like single solution and Population based are good alternative to Markowitz method.	NA
(Perrin, 2019)	Poor Control on Academic	Exploration of alternate	Coordinate descent, the

	Portfolio optimisation methods in practical settings in spite of their rigorous theorisation.	Portfolio Optimisation methods in Machine learning, Coordinate descent, The alternating Direction method to state a few.	alternating direction method of multipliers, the proximal gradient method and the Dykstra's algorithm
(J.-P. Onnela, A. Chakraborti, K. Kaski, J. Kertesz, 2003)	Asset Trees and Assets graphs find correlation between asset returns but both are not suited for all scenarios. Study probes on the scenarios where each could be suitably used.	Dynamic asset graphs based on Correlation between Asset returns.	Spearman correlation, Pearson Correlation.
(Ghali Tadlaoui, 2018)	Compare the traditional trading strategy returns with ML based algorithmic technique.	Developing a predictive algorithm using Machine learning to Find Stock Direction, Model Volatility of returns and compare the returns over time.	Random Forest, Time Series(GARCH)

(Yang et al., 2020)	Inherent complexity and cost of implementation of traditional strategy involving maximising returns and minimising risk. Strategies like Markov decision Process that use Dynamic programming to derive optimal strategy have scalability issues.	Training a Deep Reenforcement Learning Agent further obtain an ensemble trading strategy. It has benefit of Adjustability, Light weight in memory consumption.	Three actor-critic based algorithms: Proximal Policy Optimization (PPO), Advantage Actor Critic (A2C), and Deep Deterministic Policy Gradient (DDPG).
(Li et al., 2019)	Compare the traditional Portfolio allocation strategies with Deep Learning Methods to optimise Sharpe Ratio	Deep Learning Methods for portfolio optimisation to outperform traditional strategies. Four US Market Indices Total stock index (VTI),aggregate bond index (AGG), commodity index (DBC) ,Volatility Index (VIX).	LSTM with 64 Units

(Théate & Ernst, 2020)	Zeroing in on Optimal Trading position in time.	Problem Specific Deep Re-enforcement learning Agent. Developed	Deep Q Learning.
		using Artificial Trajectories of Limited stocks.	

Table 1: Related work list

In the second half of literature survey an attempt was made to understand how the Trading environment in Theorised and modelled through traditional machine learning models. It was found that a mix of Mean Variance optimisation with Feature Engineering is used to create Random Forest and time series forecasting models.(Ghali Tadlaoui, 2018).

Certain advanced techniques in the area of deep learning were analysed. These techniques were more accurate and real in Trading environment reconstruction. They revolved around the use of Re-enforcement learning in general, however three actor critic based ensemble strategy(Li et al., 2019) was found of particular merit. Sequential LSTM model was evaluated to enhance trading predictions(Li et al., 2019).Deep Q Network that aimed at optimal trading position prediction were found value adding to research.(Théate & Ernst, 2020)

3. Aims and Objective

The aim of this research is to develop Algorithmic trading strategies using Machine learning frameworks, improving upon the existing methods to generate profit and provide early warning signal to save losses under crisis situations like the Credit Crisis of 2007,Market crash During COVID-19.

The research Objectives arrived in congruence with the aim of research are below:

- Yahoo Finance Data collection,manipulation to arrive at important features for analysis.
- Performing Statistical analysis to come up with selected list of stocks.
- Comparing Traditional and Current Portfolio optimization techniques
- Creating Algo-Trading models in Trading Environment based on Machine learning Algorithms.
- Evaluating efficacy of Models through popular Testing Strategies and Mertics including the scenarois arising of credit crisis,COVID.

4. Significance of Study

Equity Markets are an important component of market economy. It's a meeting ground for Buyers and sellers of different capacities be it retail whose capacity is few thousand Dollars to institutional players whose capacity is in millions.

The promise of reward at the market also comes with a huge risk of losses. Study of these markets have been more than half a century old. One of the seminal works in this area of Markowitz is appreciated because it tries to balance the risk in process of maximizing reward.

Current research aims at bringing more intelligence in the process through use of traditional as well as advanced Mathematical and Machine Learning models.The Research is aimed at developing models using Random Forest, Time Series, Re-enforcement Learning to bring in better decision for market positions.It will be tested for scenarios that cause loss of life savings and frustration of millions of investors during financial crisis situations.The research will try to explore and develop a scheme that could act as an early warning signal to investors to take safeguarding decision in case of an impending crisis.

5. Scope

The Research will be aimed at analysing and developing models based on stock market data.Combinations of Index and individual stocks will be tried.

It will tried to create a stock portfolios that will have characteristics like Diversified in terms of sectors, High Growth Stocks Traditional as well as new age stocks.

Index price like Nifty 50 will also be tried for modelling purposes.

Algo-Trading models have varying are significant as they provide consistent returns.

Unlike manual trading strategies they are run by statistical decision making in turn providing high returns, given the high volume of trading data generated, which is impossible to analyse through traditional methods Algo-Trading provides an efficient solution to steady returns.Given the customisable nature of models it is possible to generate returns on the basis of risk appetite of the portfolio.

Limitation

Financial Markets are uncertain in their nature of being, the information that is available for public use is often too high level and aggregated.

Thus the research would work not work on a ticker/Minute level data and provide recommendations that are sufficient and satisfying.

6. Research Methodology

Research Steps are detailed below:

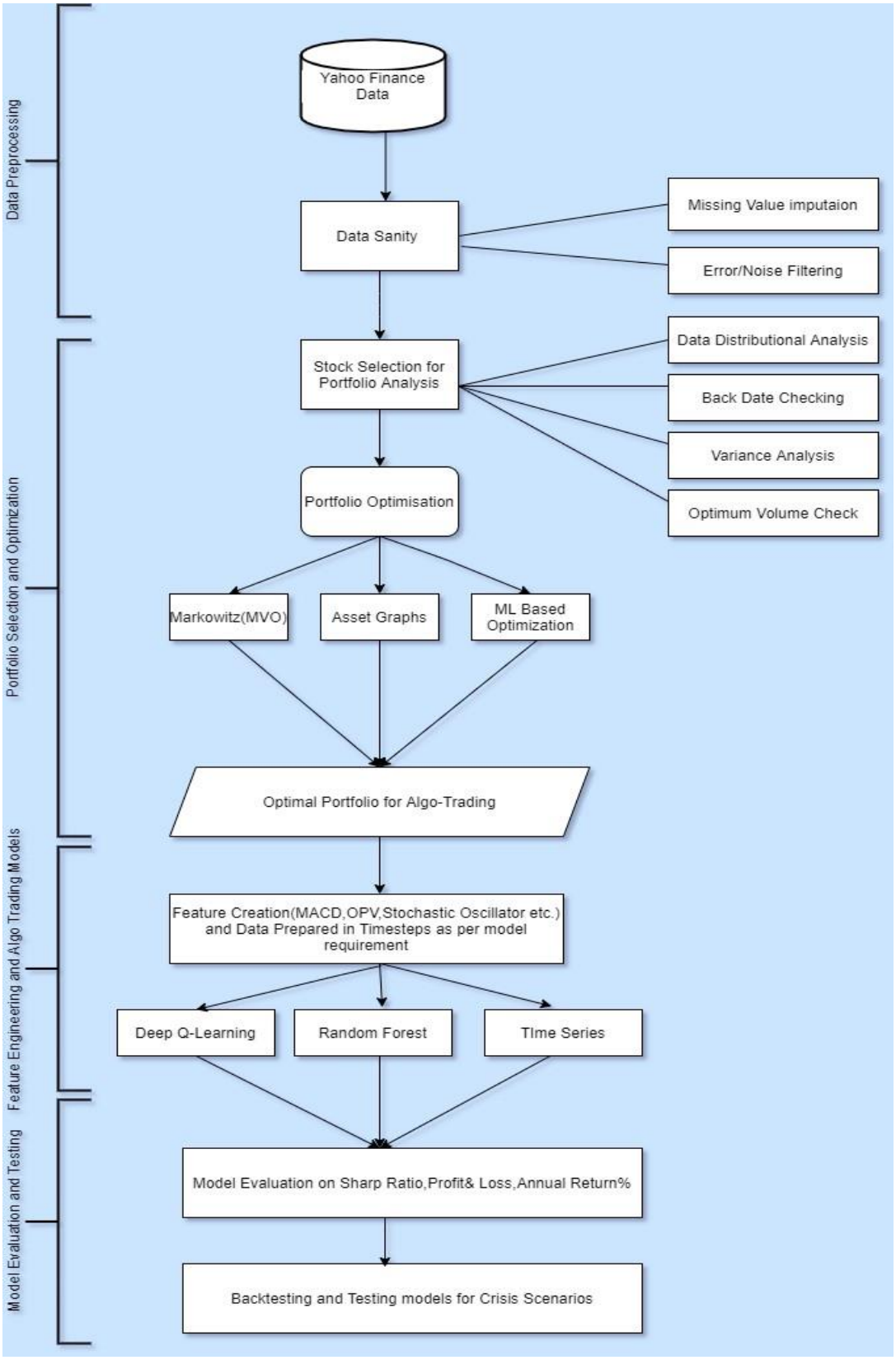


Figure 1: Research Methodology Flowchart

Introduction

The research is aimed at exploring different portfolio optimization techniques which are traditional like Mean Variance Optimization of Markowitz or the new techniques which have gained acceptance lately.

Using the publicly available Data from yahoo Finance an attempt will be made to compare the portfolio optimization techniques and their resulting returns.

Asset Graphs and Asset tree based portfolio correlation (J.-P. Onnela, A. Chakraborti, K. Kaski, 2002) will be analysed so will be other challenger techniques of optimization like Coordinate descent, the alternating direction method of multipliers, the proximal gradient method and the Dykstra's algorithm (Perrin, 2019)

Generating an active trading model will be the next part of research wherein feature based prediction model where technical indicators like Open balance Volume, Stochastic Oscillator, Moving Average Convergence Divergence etc will be used in a Random Forest algorithmic setting or Time series(Ghali Tadlaoui, 2018) setting to maximise returns.

In the Later stages Variations of Deep learning Algorithms will be used to create algorithmic models.

This area of research is wide and dynamic thus as per current information and expertise Deep Q Learning (DQN) based models will be generated.

Also the current research on Deep Reinforcement learning will be tried to generate algorithmic trading model.(Théate & Ernst, 2020).

Dataset Description

Data set used for modelling will be Yahoo Finance Data.

Data will look like below:

Dt.	Open	High	Low	Close	Adj Close	Vol.
1/1/20	121	122	120	120	120	538
1/1/20	120	121	119	119	119	771

Table 2: Data Snapshot

Input Data Attributes:

Attribute	Description
Date	Date of Transaction
Open	Opening Price of stock in morning.
High	Highest intraday traded price of the stock
Low	Lowest intraday traded price of the stock
Close	Closing Price of a stockss

Adj Close	Adjusted close is the closing price after adjustments for all applicable splits and dividend distributions. Data is adjusted using appropriate split and dividend multipliers, adhering to Center for Research in Security Prices (CRSP) standards
Volume	Number of shares of said scrip traded in a Day.

Table 3: Attribute Description

Past 20 years NIFTY data will be downloaded for Index as well as stock relating to Technology (Infosys,TCS,HCL),Manufacturing(Tata Motors, Ashok Leyland, Mahindra) ,Banking(SBI,ICICI,HDFC)

Data pre-processing and Feature Engineering

After evaluating the sample Data it was found that the Data from Yahoo Finance mostly clean. Few Null Values Rows have to be filtered from the Data.
Feature Engineering will be done to derive features like:

Feature	Description
Balance Volume	BV: indicator relating the traded volume.
Stochastic Oscillator	Compares the closing price with range of price over a given period of time.
Moving Average Convergence Divergence	Momentum indicator

Table 4: Few Features

Model Building and Model Evaluation

Host of models that are planned to be built are: Random Forest, Time series Model (GARCH), Deep Q Learning Model other variants of Re-enforcement learning models. Widely Accepted Metric like sharp Ratio, Profit and Loss, Annualised Return etc. will be used to evaluate the model performance. Model will be back tested to check for robustness, out of sample testing will also be performed. Model will be tested additionally for the credit crisis induced crash of 2008 and Pandemic related crash of 2020.

Required Resources and Hardware

- Open source software like R,Python.
- Visualisation tools like Microsoft power BI, Tableau.
- High Computation Power GPU

7. Research Plan

Research plan is illustrated below in the Gantt chart.

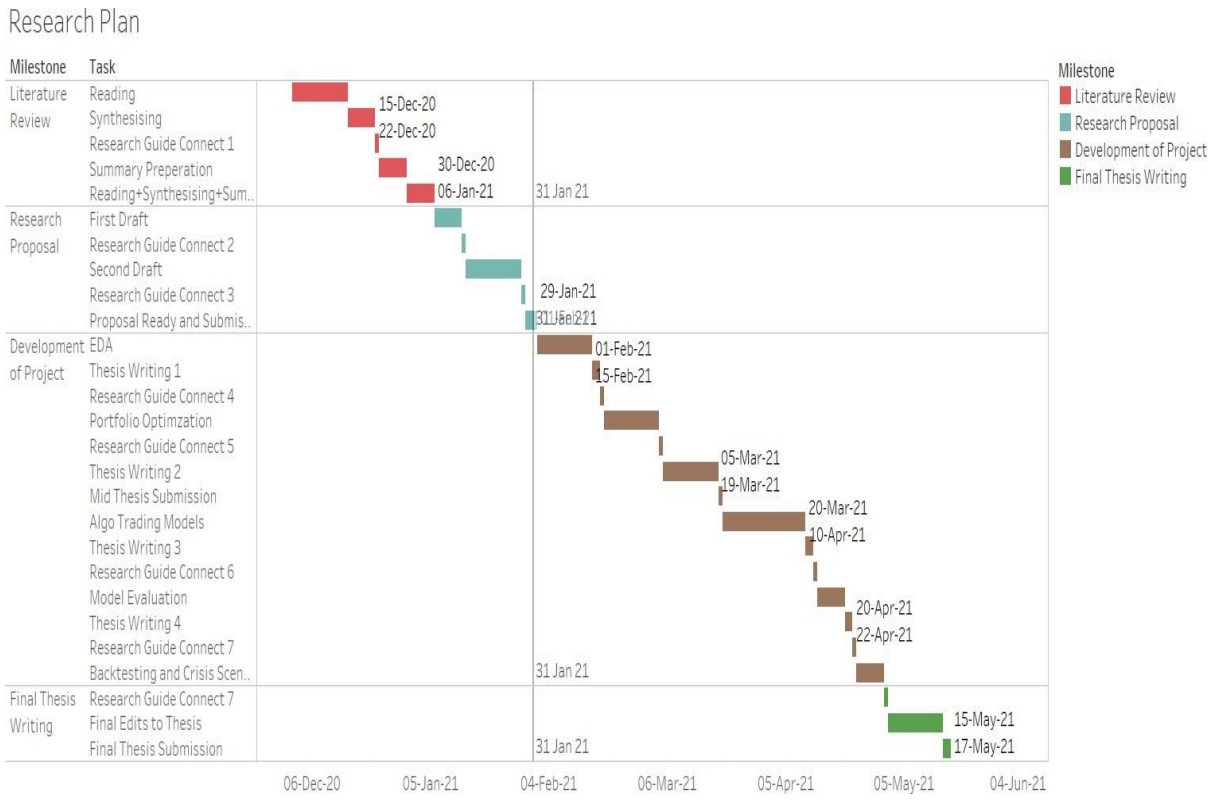


Figure 2 Research Plan and Timelines

8. References

Doering, J., Kizys, R., Juan, A.A., Fitó, À. & Polat, O. (2019). Metaheuristics for rich portfolio optimisation and risk management : Current state and future trends. *Operations Research Perspectives*. [Online]. 6 (February). p.p. 100121. Available from: https://doi.org/10.1016/j.orp.2019.100121.

Ghali Tadlaoui (2018). *Intelligent Portfolio Construction : Machine-Learning enabled Mean-Variance Optimization*. p.pp. 2017–2018.

J.-P. Onnela, A. Chakraborti, K. Kaski, J. Kertesz, A.K. (2003). Asset Trees and Asset Graphs in Financial Markets. *Physica Scripta T106*. 48.

J.-P. Onnela, A. Chakraborti, K. Kaski, J.K. (2002). Dynamic asset trees and portfolio analysis. *Eur. Phys. J. B.* 30.

Li, Y., Zheng, W. & Zheng, Z. (2019). Deep Robust Reinforcement Learning for Practical Algorithmic Trading. *IEEE Access*. 7. p.pp. 108014–108021.

Perrin, S. (2019). *Machine Learning Optimization Algorithms & Portfolio Allocation*. p.pp. 1–66.

Théate, T. & Ernst, D. (2020). *An Application of Deep Reinforcement Learning to Algorithmic Trading*. [Online]. Available from: http://arxiv.org/abs/2004.06627.