

https://dspace.org/

Forecast based portfolio optimisation using XGBoost

May, Khanya

2022
https://hdl.handle.net/10539/36999

Downloaded from a DSpace repository. https://dspace.org/

Forecast Based Portfolio Optimisation Using XGBoost

Khanya May

Supervisor(s):
Dr Wilbert Chagwiza



A research report submitted in partial fulfillment of the requirements for the degree of Master of Science in the field of e-Science

in the

School of Computer Science and Applied Mathematics
University of the Witwatersrand, Johannesburg

28 October 2022

Declaration

I, Khanya May, declare that this research report is my own, unaided work. It is being submitted for the degree of Master of Science in the field of e-Science at the University of the Witwatersrand, Johannesburg. It has not been submitted for any degree or examination at any other university.



Khanya May

28 October 2022

Abstract

Portfolio optimisation is a vital research field in modern finance. In recent years, a plethora of approaches have been proposed to deal with the increasingly challenging task of portfolio optimisation. In this research, it is demonstrated how using a new methodology that involves using XGBoost regressor chains to forecast stock prices, then incorporating these prices in k-means algorithm, selecting the assets with the highest Sharpe ratio in each cluster then allocating weights to the assets using Monte Carlo simulations. Historical stock price data of the assets in the JSE top 40 index is used. The performance of the model is evaluated using 2 test periods, 2019 as the non-crisis test period and 2020 for the crisis stress test period. The optimal portfolio has the best performance in both periods earning 94.73% returns with a Sharpe ratio of 0.1999 in 2019 and 11.02% returns with a Sharpe ratio of 0.029 in 2020.

Acknowledgements

I would first like to thank the National e-Science Postgraduate Teaching and Training Platform for providing me with funding to complete my degree.
I would like to express my appreciation to Dr. Wilbert Chagwiza, my supervisor, for his valuable and constructive feedback and advice.
I would also like to acknowledge Dr. Helen Robertson for the support while completing my research.

Contents

Declaration i

Abstract ii

Acknowledgements iii

List of Figures vii

List of Tables viii

1 Introduction 1
- 1.1 Background . . . 1
- 1.2 Problem Statement . . . 2
- 1.3 Research Aims and Objectives . . . 3
  - 1.3.1 Research Aims . . . 3
  - 1.3.2 Objectives . . . 3
- 1.4 Limitations . . . 3
- 1.5 Assumptions and Definitions . . . 4
- 1.6 Overview . . . 5

2 Literature Review 6
- 2.1 Introduction . . . 6
- 2.2 Traditional Portfolio Optimization Methods . . . 6
  - 2.2.1 Equal Weighted . . . 7
  - 2.2.2 Minimum Variance . . . 7
  - 2.2.3 Equal Weighted Risk Contribution . . . 8
  - 2.2.4 Maximum Decorrelation . . . 8
  - 2.2.5 Inverse Volatility . . . 9
  - 2.2.6 Maximum Diversification . . . 9

2.3 Machine Learning Heuristic Methods . . . 10
2.3.1 Particle Swarm Optimisation(PSO) . . . 10
2.3.2 Genetic Algorithm . . . 11
2.4 Machine Learning for Predictive Portfolio Optimisation . . . 12
2.4.1 Long Short Term Memory Networks(LSTM) . . . 12
2.4.2 Random Forest . . . 13
2.4.3 Extreme Gradient Boosted Machines (XGBoost) . . . 13
2.5 Risk Measurements . . . 14
2.6 Data Representation . . . 14
2.7 Portfolio Construction . . . 15
2.8 Summary . . . 15

3 Research Methodology **17**
3.1 Introduction . . . 17
3.2 Data . . . 17
3.3 Models . . . 19
3.3.1 Multi-Output Regression . . . 20
3.3.2 Extreme Gradient Boosted Machines (XGBoost) . . . 20
3.3.3 Clustering Model . . . 22
3.3.4 Portfolio Construction . . . 22
3.3.5 Forecast Based Portfolio Optimisation Model . . . 23
3.4 Models Evaluation Metrics . . . 24
3.4.1 Predictive Model . . . 24
3.4.2 Clustering Model . . . 25
3.4.3 Optimal Portfolio . . . 26
3.5 Summary . . . 27

4 Results Analysis **28**
4.1 Introduction . . . 28
4.2 Predictive Model . . . 28
4.2.1 Results for 2019 test . . . 28
4.2.2 Results for 2020 stress test . . . 30
4.2.3 Discussion . . . 34
4.3 Clustering Model . . . 35
4.4 Portfolio Construction . . . 38

4.5 Optimal Portfolio 40
4.5.1 Performance in 2019 test 40
4.5.2 Performance in 2020 stress test 42
4.5.3 Discussion 43
4.6 Summary 44

5 Summary, Conclusions and Recommendations **46**
5.1 Summary 46
5.2 Conclusions 47
5.3 Recommendations 48

A Technical Indicator Formulas **49**

B JSE Top 40 Assets **52**

C Assets Forecast **53**

Bibliography **59**

List of Figures

3.1 Regressor Chain 20
3.2 Methodology flow chart 23

4.1 Forecasts for 2019 32
4.2 Forecasts for 2020 33
4.3 Elbow curve 35
4.4 Efficient Frontier 39
4.5 Monte Carlo Simulated Portfolios 39
4.6 Predicted returns vs Real returns 2019 40
4.7 Portfolio returns 2019 41
4.8 Predicted returns vs Real returns 2020 42
4.9 Portfolio returns 2020 43

List of Tables

3.1 Available Features in data set 18
3.2 Technical indicators 19
3.3 Hyper-parameter tuning 21

4.1 Performance Metrics 2019 29
4.2 Performance Metrics 2020 31
4.3 Asset Clustering 37
4.4 Cluster Analysis 38
4.5 Portfolio Performance 2019 41
4.6 Portfolio Performance 2020 43

B.1 JSE Top 40 Assets 52

Chapter 1

Introduction

1.1 Background

A portfolio is defined as a collection of investment assets [37]. Portfolio optimisation is the method of selecting the best portfolio which results in the most profitable rate of returns for each unit of risk taken. This theory was pioneered by Harry Markowitz [26] and is widely known as modern portfolio theory (MPT). Portfolio optimisation is an essential component of a trading system. The idea is to obtain the optimal weight of each asset by maximising expected return and minimising risk at the same time.

There is a large pool of investment assets to choose from in the market, asset selection is essential to the success of the portfolio. Promoting diversification when selecting assets is paramount in order to achieve an optimized portfolio [43]. The challenge is to select assets that behave differently especially during periods of crashes [30]. The assets need to be least correlated with each other. This is to ensure that the impact of low performance of one asset does not result in the crash of the entire portfolio.

Managing and maintaining a portfolio of investment assets has many difficulties as there is a lot of uncertainty and many hidden variables can influence asset returns [27]. Asset returns can be affected by economical conditions, commodity prices, political events and many other factors [34]. Machine learning algorithms are known to be very effective in many prediction problems. Advancements in machine learning have created opportunities to make use of prediction theory in portfolio optimisation [7]. Since the modern portfolio theory was introduced [26] a number of improvements and changes have been proposed. The most researched

use of machine learning in portfolio optimisation is the prediction of the stock price or returns and using these predictions as expected returns instead of using historical prices. There have been some developments in the measure of risk that most accurately depicts reality. There have also been some innovative ways proposed for constructing and selecting assets to include in the portfolio. To assign the weights researchers have used the traditional mean-variance model, Monte Carlo simulations and predictive models. In this research a machine learning model is proposed to tackle portfolio optimisation by using prediction.

This research has a number of contributions to fill the gap in existing literature: First, this research investigates the performance of a multi output regression chain XGBoost when predicting stock prices. There is not much research available implementing such a method which extends existing research. Second, this researcher uses k-means clustering to achieve diversification, this is done using the forecasted prices. Thirdly, the research creates portfolios using Monte Carlo simulations and compares the performance of the portfolio to that of the equal weighted portfolio and the JSE top 40 index. In addition this research uses JSE stock data which is not widely used in this manner.

1.2 Problem Statement

In the financial investment industry there is a large number of assets available to invest in. As an investor it is essential to select the right assets based on a goal or risk appetite. It would be ideal if a way to select these assets to result in an optimal portfolio was available. The size of the market and large number of assets makes this a difficult decision which can not be efficiently solved with traditional methods. Machine learning has the ability to not only deal with a large amount of data but also detect trends and relationships between assets. The insights obtained from machine learning can be used to maximise returns on a portfolio and thus find more optimal portfolios. In this research a machine learning algorithm is proposed to predict future returns as well as create the optimal portfolios.

1.3 Research Aims and Objectives

1.3.1 Research Aims

The aim of the research is to use forecasted stock prices and clustering to produce an optimal portfolio. The proposed method is to use XGBoost to forecast stock prices, use clustering for portfolio selection and Monte Carlo simulations to obtain optimal weights thereby producing an optimal portfolio.

1.3.2 Objectives

The objectives of the research are:

(a) To develop a XGBoost model to forecast stock prices and use these predictions to estimate expected returns and risk.

(b) To implement a k-means clustering algorithm to achieve diversification and select assets.

(c) To create additional features using technical indicators calculated using the dataset.

(d) To use Monte Carlo simulations to create multiple portfolios and select optimal portfolio.

1.4 Limitations

There are a number of limitations to this research, these limitations are listed below.

(a) Portfolios are self financing and not leveraged. No additional cash is added or extracted during the whole investment period. The change in value of the portfolio comes only from the change in the price of the asset. Cash being added or subtracted from the portfolio will make the measurement of the performance of the portfolio more complex.

(b) Forecasting stock prices can not be 100% accurate due to the volatility of the market. Improving stock price forecast is a major challenge. Machine

learning techniques have shown effectiveness in prediction problems. The use of a multi-output XGBoost regression model proposed in this research may improve the forecasting results.

(c) Computational performance and running time. Training and implementing a model may result in high computational costs and use a lot of memory for storing all the information. A limited data set is selected from the market so that it can be run on the resources at hand. The XGBoost algorithm is computationally inexpensive as it develops simple learners. The running time is manageable.

(d) Lack of prior research done using the JSE dataset. Though there is much research done on portfolio optimisation, there is very little done using the JSE dataset. As such, there is no direct comparisons that can be made with other research done in the field using this dataset. However, the performance of the model will be measured against other markets using universal measures for error and portfolio performance.

1.5 Assumptions and Definitions

A number of assumptions were made when conducting the research, those are listed below:

(a) Stock prices reflect all the available information. In the scope of this research only the stock prices will be used for prediction even though a number of other factors affect the stock prices.

(b) No transactional costs or taxes. In the real world there are additional costs associated with investing in an asset these are referred to as transactional costs. In the scope of this research such costs and taxes will not be considered.

(c) No short selling. Short selling refers to when an investor borrows a stock to sell and then repurchases the stock and returns it. There will be no short selling observed in this research.

(d) Number of assets in the market. The universe of available assets for constructing a portfolio is enormous. A well rounded portfolio consists of stock and

also typically includes bonds and commodities further expending the amount of choices. This research will only consider a limited number of stocks as representative of the entire market.

1.6 Overview

This research report is structured as follows: Chapter 2 presents the literature review of studies in portfolio optimisation. Chapter 3 discusses the methodology proposed in this research. Chapter 4 evaluates the performance of the proposed methodology. Chapter 5 summarizes and presents the conclusion of the research.

Chapter 2

Literature Review

2.1 Introduction

In this section literature on portfolio optimisation is reviewed. Portfolio optimisation remains one of the most challenging problems in the field of finance [1]. It involves using historical stock price data to allocate resources to assets in such a way that it maximises returns while staying in a specified risk level. There have been a number of approaches proposed for solving this problem.

2.2 Traditional Portfolio Optimization Methods

The theoretical model introduced in [26] uses a mathematical framework for assembling a portfolio of assets by solving one of two problems, minimising risk or maximising returns. In the mean-variance model the returns and risks are quantified by means and variances [26]. The two objectives of portfolio optimization are, maximise returns and minimise risk. That is:

$$Max\Big(\sum_{i=1}^{n} w_i \mu_i\Big) \tag{2.1}$$

Maximising the returns

$$Min\Big(\sqrt{\sum_{i}\sum_{j} w_i w_j \sigma_{ij}}\Big) \tag{2.2}$$

Simultaneously minimising the risk

$$\text{Subject to} : \sum_{i=1}^{n} w_i = 1 \text{ and } 0 \leq w_i \leq 1 \tag{2.3}$$

where w_i is the weight of investment in stocks i, μ_i is the expected return of investment in stocks i, σ_{ij} refers to covariance between stocks j and i, n represents the number of stocks.
A flaw with the mean-variance approach is that it assumes that all inputs are known even though they are estimated and not known with certainty. This uncertainty often results in estimation errors that may result in sub-optimal portfolios [29]. A great amount of literature suggested replacing the mean-variance approach with alternative asset allocations that are more focused on risk and diversification as opposed to estimating expected returns.

2.2.1 Equal Weighted

The equally weighted portfolio assigns the same weight to all assets considered in the portfolio [3]. The portfolio is constructed as follows:

$$w_i = \frac{1}{n}$$

where w_i is the weight of investment in stocks i, n is the number of assets in the portfolio.
This method has shown good results, according to [13] the equal weighted portfolio consistently performs better than a value weighted portfolio based on Sharpe ratio. The drawback of this approach is that it can lead to very limited diversification if the assets risks are significantly different [39]. The model is simple and easy to implement since it has no objective function, it is very useful to use as a benchmark.

2.2.2 Minimum Variance

The objective of this model is to generate a portfolio with the least variance. The minimum variance portfolio is computed by solving the following optimisation

problem:

$$w_i = \arg\min_{w \in \mathbb{R}} w' \Sigma w$$
$$\text{subject to:}$$
$$\sum_{i=1}^{n} w_i = 1 \text{ and } w_i \geq 0$$

where Σ represents the covariance matrix.
Minimum variance can be derived from the mean-variance model, its unique property of not requiring information about expected returns makes it easy to compute. According to [2, 11] the minimum variance portfolio outperforms the market capitalization weighted index. It demonstrates higher returns, lower volatilities and therefore better risk adjusted performance. The model only requires an estimate of the covariance matrix as input . [11] suggests that minimum variance leads to a poorly diversified portfolio since it only considers covariance.

2.2.3 Equal Weighted Risk Contribution

The equally weighted risk contributions portfolio is the portfolio for which all asset contributions to portfolio risk are equalized. The equal weighted risk contribution portfolio is computed by solving the following optimisation problem:

$$w_i = \arg\min_{w \in \mathbb{R}} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(w_i (\Sigma w)_i - w_j (\Sigma w)_j \right)^2$$

The aim is to equalise the risk contribution from each portfolio asset results in maximised diversification [32]. In [39] it is shown that the volatility for the equal risk portfolio is located between those of minimum variance and equal weighted portfolio.

2.2.4 Maximum Decorrelation

Maximum decorrelation aims to minimise the correlation between the assets in the portfolio. The maximum decorrelation portfolio is computed by solving the

following optimisation problem:

$$w_i = \arg\min_{w \in \mathbb{R}} w' \Omega w$$

subject to:

$$\sum_{i=1}^{n} w_i = 1 \text{ and } w_i \geq 0$$

where Ω represents the correlation matrix.
The model only requires the correlation matrix as input thereby reducing estimation errors. This approach assumes that the asset volatilities are identical [10].

2.2.5 Inverse Volatility

Inverse volatility assigns weight that is inversely proportional to each assets volatility and is then normalized such that the portfolio weight sums to one. The optimisation problem for inverse volatility is presented by:

$$w_i = \frac{\frac{1}{\sigma_i}}{\sum_{i=1}^{n} \frac{1}{\sigma_i}}, \forall i = 1, \ldots, n$$

Inverse volatility disregards correlation between the assets its main aim is to control the portfolio risk [39].

2.2.6 Maximum Diversification

The maximum diversification portfolio maximises the diversification ratio. The diversification ratio is the weighted average of the volatilities of assets to the volatility of the portfolio of the same assets. The maximum diversification portfolio is

computed by solving the following optimisation problem:

$$w_i = \arg\max_{w \in \mathbb{R}} \frac{w'\sigma}{\sqrt{w'\Sigma w}}$$

subject to:

$$\sum_{i=1}^{n} w_i = 1 \text{ and } w_i \geq 0$$

In [9] maximum diversification outperforms market capitalization weighted portfolio as well as the minimum variance, equal weighted portfolio in delivering higher returns with lower volatilities.

Different computational techniques have been used to solve portfolio optimisation. These computational techniques involve exploring a large number of combination of states, which increase exponentially with the size of the problem becoming computationally expensive. This leads to large amount of computational resources being used [8, 36]. These methods also use mean historical returns as estimates for future expected returns resulting in inaccurate predictions of future [23].

2.3 Machine Learning Heuristic Methods

Some heuristic and evolutionary based techniques can approximate solutions in a reasonably better time than computational solutions. There have been a number of studies that have investigated portfolio optimisation using heuristic techniques. In this section two of the more popular techniques are discussed in relation to portfolio optimisation.

2.3.1 Particle Swarm Optimisation(PSO)

Particle swarm optimisation is a population based search algorithm simulating the social behaviour of birds within a flock. A particle swarm is a population of particles, where each particle is a moving object that is through a search space and is attracted to previously visited locations with high fitness. In the case of portfolio optimisation the particles represent portfolios that consist of the allocated weights

for the assets. The initial portfolio is set randomly. The fitness can be defined by the objective function to either minimise risk or maximise returns or other objective functions. In [42] efficient frontiers are generated using PSO and compared to Monte Carlo simulated frontiers. PSO is found to yield better frontiers, that is, higher returns with lower volatilities. In [12] PSO is implemented on the mean-variance model with cardinality and boundary constraints, that is, the number of of assets and weight allocations are bounded. The comparison was made; using efficient frontiers; between genetic algorithm, simulated annealing, tabu search and the PSO algorithm. The results show that none of the methods clearly outperform the others in all the investment policies.

2.3.2 Genetic Algorithm

The genetic algorithm is a systematic search method for optimisation problems based on the mechanics of natural selection and genetics. The chromosomes with the best fitness survive to create a new generation of chromosomes. In portfolio optimisation the chromosome represent portfolios and the genes are the allocated weights for each asset. The fitness is measured by an objective function to either maximise returns or minimise risk or another objective. In [41] the genetic algorithm is compared to the traditional mean-variance framework and it is found to obtain higher returns with lower volatilities. In [40] the genetic algorithm is integrated into a stochastic sampling procedure and is found to outperform the market benchmark. Different types of genetic algorithms have been implemented in an effort to solve the portfolio optimisation problem. In [36] multiple genetic algorithms and fuzzy theory are applied to the portfolio optimisation problem. Hybrid encoding is used to determine whether a stock is selected for a portfolio and what weight is assigned to it. In [8] the use of fund standardisation to calculate portfolio risk to reduce the amount of risk calculations is proposed. A genetic algorithm is used to construct the portfolio with the Sharpe ratio used to measure performance of the portfolio.

The heuristic and evolutionary techniques cover the stock selection process of portfolio optimisation but do not consider the estimation or calculation of expected returns which is an essential part of solving the problem.

2.4 Machine Learning for Predictive Portfolio Optimisation

Machine learning has been used to tackle the portfolio optimisation problem in two ways, creating prediction based portfolios and portfolio selection or construction process. The first step of portfolio optimisation is to develop estimates for expected returns and volatilities for each asset. Recent developments in Machine Learning has resulted in significant opportunities to incorporate predictive theory into portfolio optimisation. There is a consensus among studies that predictions have the potential to generate high investments returns [7, 23, 37, 21].

2.4.1 Long Short Term Memory Networks(LSTM)

LSTM is an improved version of the traditional recurrent neural network (RNN). It alleviates the vanishing gradient problem of the RNN by setting gate variables to control how much information of the previous time steps is transmitted to the current time step. Thus, the LSTM network is effective at capturing temporal features on sequential data. LSTM algorithms are widely used to forecast stock prices which are then used to build a mean-risk model. In [23] different neural networks are compared using China exchange rate stock prices. [35] also compares different neural networks for predictive based portfolio optimisation. The LSTM model is found to produce the least predictive errors in comparison to the other models. LSTM, logistic regression (LR), and support vector machines (SVM) are compared in [37] and LSTM outperforms the other algorithms in their experiments. Stock price data, macroeconomic and market related data is used as features for the LSTM forecasting model in [34]. Principal component analysis (PCA) is used to transform the macroeconomic and market data. The LSTM model outputs the optimal weight allocations. A novel model for directly optimising the Sharpe ratio without first forecasting the stock prices using a LSTM model is proposed in [43]. This is done by adjusting the parameters of the model. The researcher claims that price forecasting is not guaranteed to result in an optimal portfolio, it only aims to minimise the prediction error. The proposed model directly optimises the Sharpe ratio.

2.4.2 Random Forest

The idea behind random forest is to train several decision trees; each on different subsets of data. This ensures a reduced variance of the predictor while keeping the same bias. Each tree is trained on random samples of the data and the prediction of each tree are combined through a majority vote.In [21] random forest with optimisers are implemented in an effort to prove that predictions provide more reliable input than using historical data, which is found to be true after experiments. the random forest predictions are used to compare different portfolio construction techniques. Mean-variance, equal weighted and hierarchical risk parity (HRP) are compared and it is found that on average mean-variance portfolio outperforms the other techniques. In [16] the Markowitz framework [26] was compared to the Meucci framework. Experiments proved that for low risk tolerance Markowitz performed better and for high risk tolerance the Meucci framework was better on Sharpe and Sortino ratios. In [22] the random forest is used to select stock for the portfolio based on the nine factor model and construct a portfolio. In [38] the random forest is used to predict the direction of the stock price, then a generalized autoregressive conditional heteroskedasticity (GARCH) model is used to forecast magnitude of the move. This model forecasts volatility then deduces the expected returns. The traditional mean-variance and equal weighted portfolios are compared against predicted versions and thee predicted mean-variance outperforms the other models based on returns.

2.4.3 Extreme Gradient Boosted Machines (XGBoost)

XGBoost is a statistical framework that casts boosting as a numerical optimisation problem where the objective is to minimise the loss of the model by adding weak learners using a gradient descent like procedure. In [7] a hybrid extreme gradient boost (XGBoost) with an improved firefly algorithm is proposed, the firefly algorithm is used to tune XGBoost hyper-parameters. Comparison between XGBoost and LR in an automated model that adjusts the risk aversion based on the predictions made is presented in [19].

Classic portfolio optimisation models usually adopt the mean of historical stock returns as expected returns and deduce the risk from these. More recent research attention has focused on implementing predictive models to estimate expected

returns. In [23] it is argued that using mean historical returns as future returns results in imprecise predictions of stock returns. The researchers found that using forecasted prices results in better performing portfolios when compared to using historical prices. In view of all that is mentioned so far, one may suppose that using predictions in portfolio optimisation can result in better portfolio performance.

2.5 Risk Measurements

After Markowitz [26] work the standard mean-variance model has been improved and extended in several ways. Markowitz suggested the use of variance as a measure of risk, this has lead to a number of debates. Several researchers have discovered shortcomings to this risk measure [23, 1, 19, 33, 40] and have investigated other risk measures. [23] uses semi-absolute deviation as a measure of risk stating that it is not reasonable to use historical returns as a measure of risk. [1] uses Conditional Value at Risk (CVaR). [22] proposes a hypothesis that risk set consisting of variance, skewness, and kurtosis is a more effective measure of risk because variance is not sufficient for investment strategies. However, many other researchers continue to use variance as a measure of risk [37, 21, 24, 25]. In this research variance is used as a measure of risk because despite these shortcomings mean-variance still provides an efficient formula for representing the trade off between expected returns and risk.

2.6 Data Representation

Most existing studies represent the current stock state as a vector containing the stock price and technical indicators. There is a lot of data and information about the financial markets publicly available. Researchers have used different types of datasets to build their models for the portfolio optimisation problem. The most commonly used datasets is the stock price data [28, 37, 25]. In [4] the stock data is transformed into a 3D tensor and considered as an image. A number of different researchers has used the stock price data to derive additional features known as technical indicators [19, 21, 24]. Technical indicators assist in making trading actions by sending buy or sell signals. These features are used to extract more information from the stock prices and further improve the predictive model. Stock prices have

also been used in conjecture with economic and market related variables as input to models [21, 34, 16].

2.7 Portfolio Construction

Selecting which assets to include in the portfolio is a crucial action as this could be the difference between finding an optimal portfolio and not. Diversification is essential to achieve an optimal portfolio so how your portfolio is constructed is paramount to its success. In a traditional MV model diversification is achieved using the covariance matrix, which only considers the relationship between two assets. As the size of the data and number of assets in the market increase the size of the matrix grows significantly [8]. This consumes more computation, space and time resources. In [34] weight constraints are used to prevent assets with higher returns being allocated very high weights preventing them from being highly correlated to the performance of the portfolio as this goes against the diversification principle. [25, 34] uses mean-variance approach. [37] selects assets with high returns and low risk to create a portfolio but does not consider any method to achieve diversification. A number of studies use clustering to select assets for the portfolio. In [28, 33] the cluster representatives are fed to the mean-variance model to then obtain the weights. In [18] a comparison of the complete linkage and ward clustering algorithms for portfolio selection with different risk aversions is made. K-means clustering is proposed in [27] using ratios of revenue to assets and net income to assets as similarity measures. The optimal portfolio performs better than the benchmark index but is found to be more volatile. In [28] average linkage agglomerate algorithm is implemented on two environments, realistic and semi-realistic. The focus of the paper is in finding a superior similarity measure by comparing different measures.

2.8 Summary

Portfolio optimisation framework was introduced by Markowitz [26]. Different objectives were studied as opposed to the mean-variance model initially presented, the focus moved from estimating expected returns to risk and diversification. The

studies evolved to implementing heuristic algorithms and machine learning as a result of trying to avoid estimation errors of exact models and increased amounts of data. The growth of the use of prediction introduced an opportunity to use predictive machine learning models in the portfolio optimisation problem. Thus, it can be deduced from the literature reviewed that portfolio optimisation is well researched. It has been noted that there is a need for using forecasted prices, as estimates for returns and risk, instead of historical prices. There has also been success with using clustering in the portfolio selection process as means of achieving diversification. Thus there is a need for the proposed model that implements both these strategies, forecasting and clustering, to produce optimal portfolios.

Chapter 3

Research Methodology

3.1 Introduction

In this research; experimental and statistical research design methods are applied. The experimental design method is implemented when using historical stock prices to predict future prices. The aim is to study the impact of historical prices on future prices. Clustering is used to study the relationship between the stocks being observed in the research. Statistical techniques are used to measure the performance of the developed model.

3.2 Data

The data for this research is collected from the investing website [17] https://www.investing.com/indices/ftse-jse-top-40-components. Investing is a website that provides data in real time about exchanges around the world. The data contains Johannesburg Stock Exchange (JSE) top 40 daily price information from 01/01/2014 to 31/12/2020. The features are described in Table 3.1. The JSE stock market is only open on weekdays and excludes weekends and national holidays.

Pre-processing

Formatting volume: The volume is quantified using K for thousands and M for millions this is converted to numbers to make it easier to work with.

TABLE 3.1: Available Features in data set

Feature	Description
Date	Date of recorded prices.
Close price	Price at which the stock stopped trading during normal trading hours.
Open price	Price at which the stock opened trading on a given day.
High price	Highest stock price traded throughout the day from when the market opened to closing.
Low price	Lowest stock price traded throughout the day from when the market opened to closing.
Volume	Number of shares traded (bought/sold) over a given day.
Change %	The price change percentage from previous day.

Formatting stock price: The stock prices are in South African cents and contain special characters, these are converted to rands and the characters are removed.

Feature engineering: The features in Table 3.1 will be used to compute additional features, which are technical indicators. Technical indicators are pattern based signals produced by the price and volume of a stock [5]. Future price movements can be predicted with the use of technical indicators. The list of technical indicators to be used as features is found in Table 3.2 the formulas used to compute the indicators can be found in Appendix A. The technical indicators and stock price data are then used to create lag features. For each data point the previous three months (63 days) of data is used as lag features and the next month (21 days) of data is used as window features. Window sliding is the strategy of taking the previous time steps to forecast the subsequent time step. Then, using this method, the data can be transformed to solve as a regression problem.

X/Y split: The lag features are used as the input features (x) and the close prices for the window period are used as the output features (y). All other window features are removed from the dataset.

Feature selection: Feature selection is performed on the input features as means to reduce the number of features used in the model. There are initially 1345 features in the dataset, 1147 features are removed using collinearity leaving 198 features. This is done with the aim of reducing training time and improving the generalisability of the model. Collinear features are highly correlated to one another, meaning that one feature can be predicted from another feature. The correlation matrix is used

TABLE 3.2: Technical indicators

Technical indicator	**Description**
Simple Moving Average (SMA)	The SMA is the average price of the given time period, with equal weighting given to the price of each period.
Exponential Moving Average (EMA)	The EMA represents an average of prices, but places more weight on recent prices.
Moving Average Convergence Divergence (MACD)	Based on the differences between two moving averages of different lengths, a Fast and a Slow moving average.
Bollinger Band (BB)	Presents envelope bands, maximum and minimum moving averages, to measure volatility.
Relative Strength Index (RSI)	It represents the current price relative to other recent pieces in the selected look back window length.
Commodity Channel Index (CCI)	Compares current price to average price over a specific time period.
Stochastic Oscillator (SO)	Normalizes price as a percentage.
On Balance Volume(OBV)	Cumulative running total of the amount of volume occurring on up periods compared to down periods.
Average Directional Movement (ADX)	Quantifies trend strength by measuring the amount of price movement in a single direction.

to determine which features to eliminate. If the correlation matrix shows an off diagonal value of more than 0.9 one of the features is removed from the dataset.

3.3 Models

In this section the forecasting algorithm used to predict stock prices is presented. Then portfolio selection and weight allocation are considered.

3.3.1 Multi-Output Regression

Multi output regression aims to simultaneously predict multiple real valued target variables. This provides means to effectively model multi output datasets by considering not only the relationship between features and the corresponding targets but also the relationship between targets, guaranteeing a better representation. Regressor chains (RC) are a type of multi-output regression based on the idea of chaining single target models. The training of RC consists of selecting a chain of target variables, then building a separate regression model for each target following the order of the selected chain. The first model is only concerned with the prediction of the first target in the chain. Then, subsequent models are trained on the transformed data which is augmented to include previously predicted targets in the chain. A representation of regressor chains is shown in Figure 3.1 found in [14].

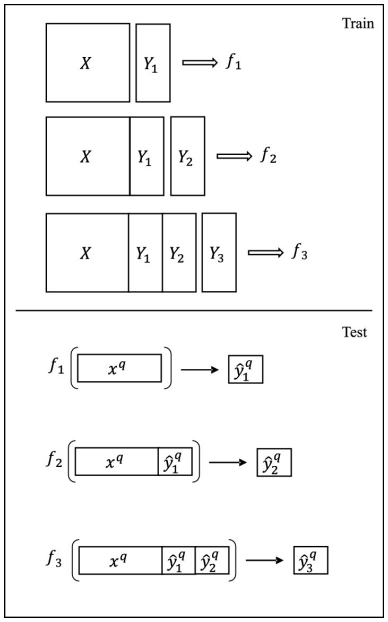


FIGURE 3.1: Regressor Chain

The Regressor chains are implemented using the sklearn package on python. They are used to predict the next 21 days close prices based on the previous 63 days data.

3.3.2 Extreme Gradient Boosted Machines (XGBoost)

XGBoost is a collection of weak classifier decision trees and it primarily focuses to train the new decision tree to learn from the errors committed by the previous trees

[15]. This is done using a gradient descent-like procedure. The learning trees are trained sequentially [6]. Initially, a regression function is drawn which is fitted to the data set and errors occur; which are referred to as residual errors. All the residual errors are considered and another regression function is made to fit the model. The errors are minimized by the combination of the previous regression function and the current regression function. Hence, continuing in this manner; the regression function gets more and more complex in nature and the root mean squared error is observed to be significantly reduced. The following regularized objective function is used to learn in the model.

$$min\Big(\sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k} \Omega(f_k)\Big) \tag{3.1}$$

where $l(y_i, \hat{y}_i)$ refers to the loss function and $\Omega(f_k)$ refers to the regularization term that can be calculated as follows:.

$$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda\|\omega\|^2 \tag{3.2}$$

where γ is L1 regularization coefficient, T is number of leaves, λ is L2 regularization coefficient, ω is the leaf weight.

Hyper-parameter Tuning: The initial model is built using the default parameters in the XGBoost model. In an effort to tune the parameters of the model and improve the performance of the model combinations of different parameter values are tested out in the model using a halve random grid search. The hyper-parameters considered for parameter tuning are found in Table 3.3. The XGBoost model is implemented

TABLE 3.3: Hyper-parameter tuning

Parameter	**Values Considered**	**Value selected**
n_estimators: Number of trees	[100, 200]	100
learning_rate	[0.1, 0.5,0.75]	0.1
max_depth: The maximum depth of the tree	[6,8, 10]	6
colsample_bytree: The portion features to be used to build each tree	[0.7,0.8,1]	0.8

on python. The model is incorporated in the regressor chain model that is used to predict the next 21 days close prices based on the previous 63 days data.

3.3.3 Clustering Model

The proposed method for selecting assets for the portfolio is clustering. The k-means clustering algorithm will be implemented. The primary goal of k-means algorithm is to minimise the distances between the cluster centroid and points within the cluster [20]. The process for implementing k-means is;

(i) Select k the number of clusters.

(ii) K data points are selected randomly as centroids.

(iii) All the data points are assigned to the nearest cluster center.

(iv) Update cluster centers using data points assigned to them.

(v) Repeat steps 3 and 4 until the stopping criterion has been reached.

There are several options for stopping criteria to terminate the k-means algorithm. These include;

(a) Centroids of the newly formed clusters remain the same.

(b) Points stay within the same cluster.

(c) Reached the maximum iterations value.

If after multiple iterations, the clusters have the same centroids or points then the is no learning of patterns taking place and training needs to stop [20]. Also if a maximum number of iterations is stipulated then the algorithm can stop once it has reached the defined threshold.
The k-means clustering model is implemented using the sklearn package on python. It is used to cluster the data based on the predicted stock prices mean and volatilities. This groups the data into different segments used for selecting assets for the portfolio.

3.3.4 Portfolio Construction

To create a portfolio; Monte Carlo simulations will be used to assign weights to the selected assets thereby creating a portfolio. Monte Carlo simulations allow the visualisation of all possible outcomes of a decisions and assess the impact, allowing

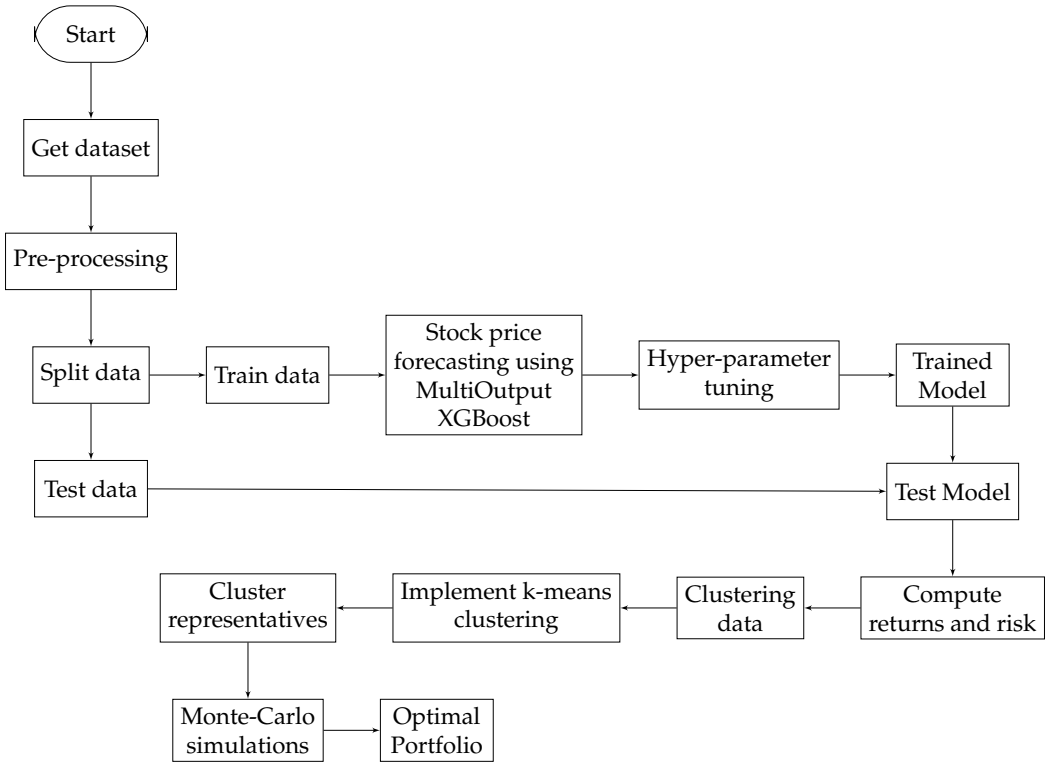


FIGURE 3.2: Methodology flow chart

for better decision making under uncertainty. A variable that has uncertainty is assigned random values at each iteration, the results are recorded. This process is done repeatedly assigning a variety of values to the variable in question. 500000 simulations are run.

The Monte Carlo simulations are implemented on python. They are used to create multiple portfolios consisting of the selected assets. This allows for a wide variety of portfolios that include the optimal portfolio.

3.3.5 Forecast Based Portfolio Optimisation Model

The proposed methodology for this research is to implement the XGBoost algorithm to predict stock prices. This method is selected based on its ability to be less computationally expensive [6] and still achieve a highly accurate results. During training the technical indicators will be used as input to the XGBoost model. The model will build the trees and output a predicted stock price. The testing data will be used in the model and the performance monitored. The k-means clustering algorithm

will use the training dataset to create the clusters for the assets. The test dataset will then be used to evaluate the performance of the clusters. A representative of the cluster will be selected as the asset with the highest Sharpe ratio in each cluster. The assets selected as the representative of the clusters will then be used in the Monte Carlo simulation to obtain weights for each asset creating multiple portfolios. The portfolio with the highest Sharpe ratio will be selected as the optimal portfolio. A flow chart of the proposed methodology can be found in Figure 3.2.

The proposed methodology will be implemented on a Windows 10 pro, 16GB RAM with a 64 Bit operating system. Python version 3.7 is the coding language on a Jupyter notebook used. The XGBoost, Yellowbrick and sklearn packages will be used to develop the model. The data will be split as follows:

- Training dataset will be from 01/01/2014 to 31/12/2018.
- Test dataset will be from 01/01/2019 to 31/12/2019.
- Stress test dataset will be from 01/01/2020 to 31/12/2020.

3.4 Models Evaluation Metrics

Measuring the performance of the proposed model will be done in three stages; measuring the performance of the forecasting model, the clustering model and that of the optimal portfolio.

3.4.1 Predictive Model

The performance of the forecasting model will be reported as an error in those predictions. Error outline on average how close predictions were to the actual values. The error metrics that will be used for evaluating the XGBoost forecasting model are:

(a) Root mean squared error (RMSE). It is a method of measuring the difference between values predicted by a model and their actual values.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}} \tag{3.3}$$

where $\hat{y}_i$ is the predicted price, y_i is the actual price, n is the total number of data points.

(b) Scatter Index (SI). It is the ratio of the RMSE to the mean price of the asset in percentage. It measures the portion of error relative to the mean price of the asset.

$$SI = \frac{RMSE}{\overline{y}} \tag{3.4}$$

where $\overline{y}$ refers to the mean price of the asset.

(c) Mean absolute percentage error (MAPE). It gives an absolute percentage on how much the predicted results deviate from the target.

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{\hat{y}_i - y_i}{y_i} \right| \tag{3.5}$$

3.4.2 Clustering Model

The performance of the k-means clustering model will be measured based on the similarities between data points in same cluster and those in different clusters. The measures used will be:

(a) Davies Bouldin Index (DBI). It is commonly used to evaluate the goodness of a split for a given number of clusters. It is calculated as the average similarity of a cluster with another cluster it is most similar to.The lower the value the better the cluster separation, vice versa.

$$DBI = \frac{1}{n} \sum_{1}^{n} R_i, \tag{3.6}$$

where

$$\begin{aligned} R_i &= \max_{j=1,..,n} (R_{ij}), \\ R_{ij} &= \frac{s_i + s_j}{d(v_i, v_j)}, \\ s_i &= \frac{1}{||c_i||} \sum_{x \in c_i} d(x, v_i) \end{aligned}$$

$d(x, y)$ is the Euclidean distance between x and y, c_i is cluster i, v_i is the centroid of cluster i, n is the number of clusters.

(b) Silhouette Analysis. Is used to determine the degree of separation between clusters.The Silhouette coefficient is calculated as follows

$$s_i = \frac{b_i - a_i}{max(a_i, b_i)} \tag{3.7}$$

where a_i is the mean intra-cluster distance, b_i is the distance from i to nearest cluster that i is not a part of.

3.4.3 Optimal Portfolio

To measure the performance of the optimal portfolio; comparisons will be made with two benchmarks.

(a) The equal weighted portfolio. The weights of the assets selected using clustering will be assigned equally to create this portfolio. The performance of this portfolio will be compared to that of the developed portfolio.

(b) The JSE top 40 index. The performance of the developed portfolio will be compared to that of the broader market index.

Covid-19 resulted in market crash in 2020 and the performance of the portfolio during this period will be evaluated as a stress test period. The following methods will be used when evaluating the model and benchmarks;

1. Sharpe ratio (SR). Computes the excess return of the investment portfolio per unit of total risk of the portfolio [31].

$$SR = \frac{E(R_p) - R_f}{\sigma(R_p)} \tag{3.8}$$

where $E(R_p)$ refers to the return of the portfolio P, R_f refers to the risk free return rate, $\sigma(R_p)$ is the total risk of portfolio P. However, the research report omit the problem of choice between risk-free and risky assets. The research report focus on optimization between risky assets, and our main goal is a demonstration

of machine learning during the risky side of portfolio construction. Thus, we will set risk free rate to 0,

2. Beta coefficient. Measure of sensitivity of a portfolio in the entire market.

$$\beta_p = \frac{Covariance(R_p, R_b)}{Variance(R_b)} \tag{3.9}$$

3. Treynor Ratio (TR). Measures excess return per unit of systematic risk instead of total risk [31].

$$TR = \frac{E(R_p) - R_f}{\beta_p} \tag{3.10}$$

4. Information ratio (IR). Measures the ability of the portfolio to outperform a benchmark portfolio in terms of returns. [25].

$$IR = \frac{E(R_p) - E(R_b)}{\sigma(R_p - R_b)} \tag{3.11}$$

where $E(R_b)$ is the return of the benchmark portfolio, $\sigma(R_p - R_b)$ is the tracking error.

3.5 Summary

The methodology used when conducting this research was presented in this section. To conduct the research JSE top 40 stock price data was used, technical indicators were deduced and used as additional features. Previous 63 days data are used as input in a multi-output XGBoost model to predict the close price for the next 21 days. These predictions are then used to calculate expected return and risk which are then used in a clustering algorithm. After a representative for each cluster is selected the Monte Carlo simulations are used to create portfolios and an optimal portfolio is selected from these. Performance for each of the models is measured based on metrics presented

Chapter 4

Results Analysis

4.1 Introduction

This section encompasses the evaluation of the results obtained through the application of the methodology described in Chapter 3. There are two test periods; the first test period is January 2019 to December 2019 and the second is January 2020 to December 2020. The first period will be referred to as the test period and the second will be the stress test period. This is because during the second period the recent Covid-19 pandemic occurred, resulting in stock markets falling dramatically and experiencing extreme volatility. It can be helpful to see how the method fared over the course of this economic event. The crash started in February 2020 where markets experienced a drop in their performance.

4.2 Predictive Model

In this section the performance of the multi-output XGBoost prediction model will be assessed for both the test and stress test period. The model is tested with the testing sets and comparison between actual and predicted values are made.

4.2.1 Results for 2019 test

Table 4.1 contains the performance of the forecast model for each stock in the JSE top 40 index, which in this case represents the whole market. The stocks with the highest and lowest values are in bold on Table 4.1 and the graphs of their forecasts are in Figure 4.1.

TABLE 4.1: Performance Metrics 2019

Label	Mean Price	RMSE	Scatter Index %	MAPE %
ABGJ	162.97	15.22	9.34	7.05
ANGJ	245.09	44.03	17.96	13.76
APNJ	107.84	28.41	26.34	23.29
ARIJ	160.97	10.22	6.35	4.99
BHPJ	327.85	19.27	**5.88**	**4.67**
BTIJ	536.95	125.85	23.44	14.26
BVTJ	200.03	15.14	7.57	5.98
CCOJ	43.29	16.85	**38.93**	**30.44**
CFRJ	109.12	10.33	9.47	7.66
DSYJ	135.43	11.56	8.54	7.61
EXXJ	148.38	15.72	10.59	8.83
FSRJ	64.76	4.66	7.20	5.61
GRTJ	23.75	1.79	7.52	5.75
IMPJ	77.70	11.97	15.41	13.05
INLJ	85.96	7.40	8.61	7.58
INPJ	85.30	7.03	8.24	6.85
IPLJ	57.68	8.62	14.94	12.05
KIOJ	404.14	76.53	18.94	15.83
LHCJ	24.64	4.87	19.77	16.93
MNPJ	313.64	19.04	6.07	4.76
MTNJ	96.14	10.58	11.00	7.94
NEDJ	248.91	24.32	9.77	7.20
NPNJn	**2931.41**	**447.80**	15.28	14.47
OMUJ	**20.94**	**1.33**	6.37	5.59
REMJ	187.38	16.28	8.69	6.87
RMHJ	79.42	5.55	6.99	5.66
RNIJ	247.43	28.59	11.55	9.58
SBKJ	184.25	15.39	8.35	7.05
SHPJ	151.78	18.87	12.43	10.87
SLMJ	77.45	6.69	8.64	7.36
SOLJ	352.82	69.70	19.76	16.92
TBSJ	230.49	19.16	8.31	7.04
VODJ	119.85	7.60	6.34	5.37
WHLJ	50.85	6.30	12.39	9.84
Mean	**243.96**	**33.31**	**12.26**	**9.96**
Max	**2931.41**	**447.80**	**38.93**	**30.44**
Min	**20.94**	**1.33**	**5.88**	**4.67**

The RMSE ranges from R1.33 (OMUJ) to R447.80 (NPNJn). These assets also have the lowest and highest mean price. OMUJ has a below average scatter index and MAPE, NPNJn has above average scatter index and MAPE. The graph of the asset with the lowest RMSE is in Figure 4.1d. From the graph it can be observed that the model is good at tracking the trend of this asset, it does sometimes respond slowly to the trends and has high variations. In Figure 4.1c the asset with the highest RMSE is shown. The model has periods where it does not follow the trend of the asset (February to March) and periods where it responds slowly to drastic changes in the stock price (September to December).
The Scatter index ranges from 5.88% (BHPJ) to 38.93% (CCOJ). These assets also have the highest and lowest MAPE of 4.67% (BHPJ) to 30.44% (CCOJ). When looking at Figure 4.1b it can be observed that the model over estimates the stock price and has periods where it jumps up and this results in very high error rates. The model does not successfully track the trend of the stock price for this asset. Figure 4.1a shows the asset with the lowest error rates. The model is good at tracking the trend for this asset. However, it has a high variance in the period from September to November where the model has a drastic change that is not reflected in the actual stock price movement.
The average RMSE is R33.31, scatter index is 12.26% and MAPE is 9.96%. This shows that the model is good at predicting the stock prices. Since stock prices are quite volatile achieving these error rates means the model is performing well and is sufficiently predicting stock prices. As can be observed in Figure 4.1 the forecasted prices are close to the actual prices. The forecasted prices are more volatile than the actual prices and are more erratic in their movements. However, they do not move too far from the mean price, the furthest they deviate is 38.93% (CCOJ). The model is most accurate at predicting the assets with the lowest error rates and least accurate with those with the highest error rate. The models response to changes in the asset price is slow.

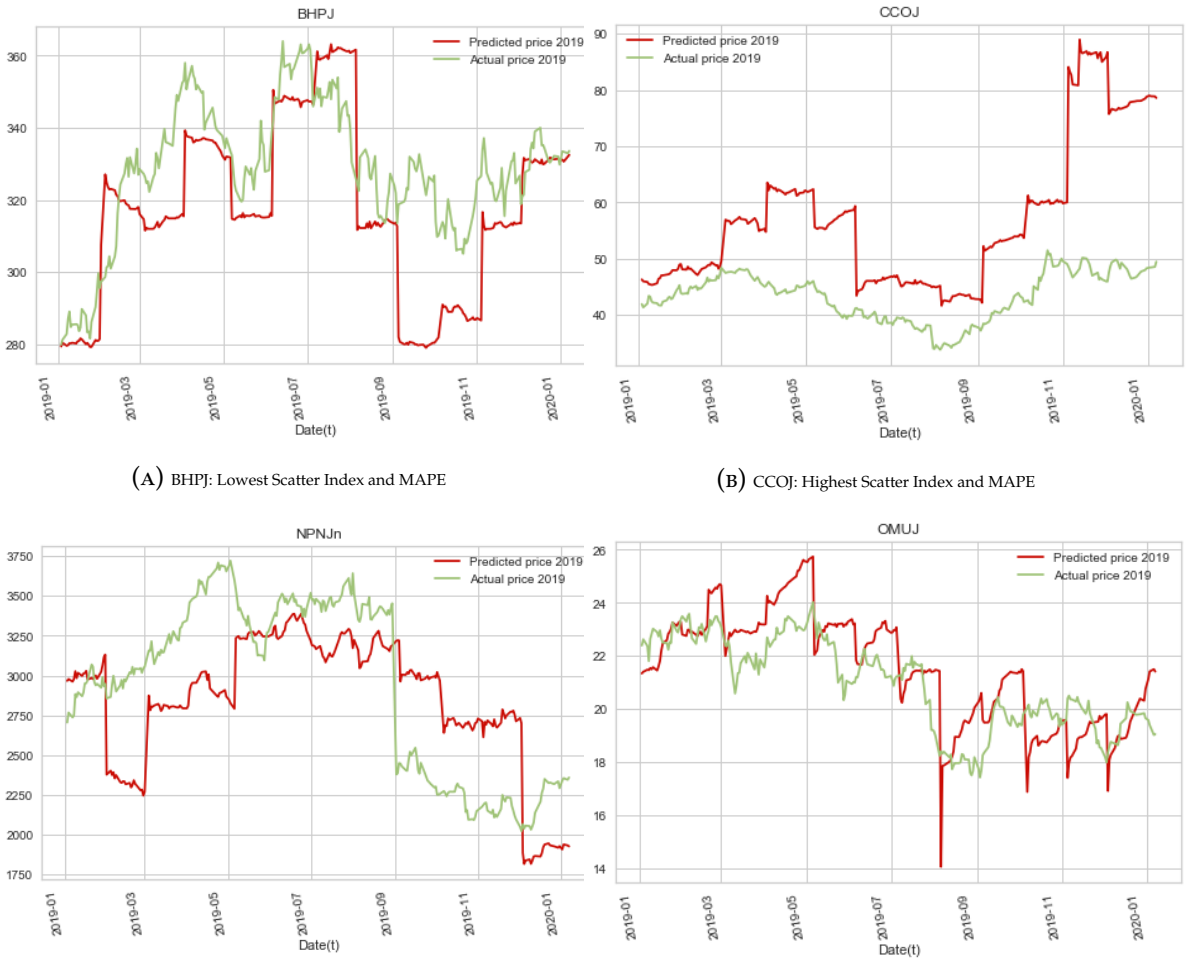
4.2.2 Results for 2020 stress test

In an effort to further test the effectiveness of the prediction model a stress test was performed using the data from the 2020 Covid-19 pandemic. The data from the previous test was added to the training data for the 2020 model. The model was

TABLE 4.2: Performance Metrics 2020

Label	Mean Price	RMSE	Scatter Index %	MAPE %
ABGJ	100.77	25.69	25.50	21.63
ANGJ	406.85	129.76	31.89	26.52
APNJ	123.58	14.19	11.48	9.92
ARIJ	182.41	31.49	17.26	14.32
BHPJ	340.66	41.96	12.32	10.94
BTIJ	609.09	107.90	17.71	13.18
BVTJ	158.84	20.18	12.70	10.82
CCOJ	33.40	13.41	40.15	36.57
CFRJ	111.32	9.70	8.72	6.56
DSYJ	113.11	15.14	13.39	12.42
EXXJ	124.71	14.63	11.73	10.42
FSRJ	44.70	12.63	28.25	24.00
GRTJ	14.35	7.12	49.64	50.98
IMPJ	141.43	28.68	20.28	19.14
INLJ	43.28	31.17	72.03	65.54
INPJ	43.44	27.67	63.71	58.11
IPLJ	39.66	20.41	51.46	35.10
KIOJ	454.51	106.62	23.46	20.06
LHCJ	18.87	7.16	37.94	36.92
MNPJ	322.00	35.49	11.02	9.26
MTNJ	61.72	18.90	30.63	27.68
NEDJ	124.56	36.03	28.93	26.44
NPNJn	**2894.67**	**373.19**	12.89	10.49
OMUJ	**12.89**	7.15	55.47	56.68
REMJ	120.00	20.00	16.67	15.30
RMHJ	29.59	41.99	**141.94**	**2183.79**
RNIJ	294.54	25.85	8.78	6.85
SBKJ	119.85	24.13	20.13	17.10
SHPJ	120.21	14.87	12.37	10.55
SLMJ	59.71	7.41	12.41	9.92
SOLJ	136.17	77.64	57.02	84.24
TBSJ	187.26	18.69	9.98	7.96
VODJ	123.46	**6.76**	**5.48**	**4.24**
WHLJ	35.45	15.53	43.82	30.79
Mean	**227.85**	**40.86**	**29.92**	**87.78**
Max	**2894.67**	**373.19**	**141.94**	**2183.79**
Min	**12.89**	**6.76**	**5.48**	**4.24**

FIGURE 4.1: Forecasts for 2019



(A) BHPJ: Lowest Scatter Index and MAPE

(B) CCOJ: Highest Scatter Index and MAPE

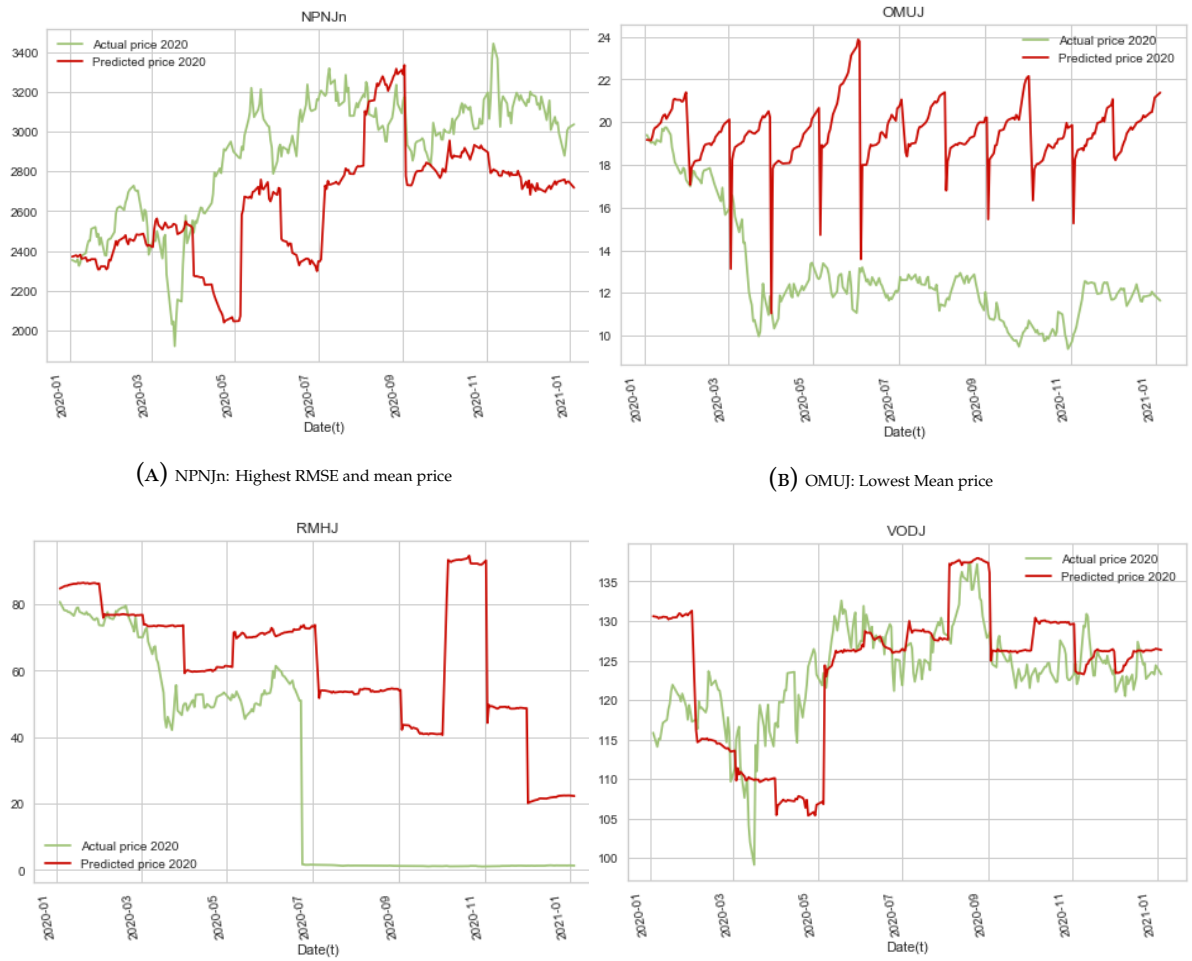
(C) NPNJn: Highest RMSE and mean price

(D) OMUJ: Lowest mean price and RMSE

trained using the same parameters and data features. The performance of the stress test period are presented in Table 4.2 and Figure 4.2. The results show that the error rates of the model were quite high and increased drastically from those achieved during the 2019 test period Table 4.1.

The RMSE ranges from R6.76 (VODJ) to R373.19 (NPNJn). VODJ also has the lowest error rates with scatter index of 5.48% and MAPE of 4.24%. This means that the model is quite accurate in making predictions for this asset. Looking at Figure 4.2d which shows the predictions for VODJ, one can see that the model can track the trends of the asset. The model does not respond to the initial, sharp drop at the beginning of the crisis, it slowly drops the price at a later time. However, after this

FIGURE 4.2: Forecasts for 2020



(A) NPNJn: Highest RMSE and mean price

(B) OMUJ: Lowest Mean price

(C) RMHJ: Highest Scatter Index and MAPE

(D) VODJ: Lowest RMSE, Scatter Index and MAPE

period it follows the trend really well. Figure 4.2a shows the asset with the highest RMSE. The model responds really late to the effect of Covid 19 that occurs in March the model drops in April. The model also has other drastic drops and increases that are not observed in the actual price of the stock, these periods result in the high RMSE.

The highest Scatter index and MAPE are 141.94% and 2183.79%, respectively.These rates are achieved by the same asset, RMHJ, the price for this asset experiences a dramatic drop from R54.18 to R1.57 in a day. This is due to RMHJ unbundling shareholding in Firstrand Limited , in June, which was in the same group as RMHJ. The model takes too long to respond to this drop and does so in small incremental

steps, it also predicts a large increase in October moving the opposite direction to the actual price of the asset. This large variation in price lasts a long period of time and results in an obscenely large scatter index and MAPE.
OMUJ has the lowest mean price meaning this is the cheapest asset in the market. Even tough this asset has a low RMSE it has above average scatter index and high MAPE. Figure 4.2b shows that the model does not follow the trend of the assets stock price. The model predictions are also very volatile and over prices the asset for the whole period.
The average RMSE is R40.86, scatter index is 29.92% and MAPE is 87.78%. These error rates are significantly high and shows that the model struggles to predict the stock prices in this period. There are also large, drastic error for the predictions for RMHJ. Which results in an inflated average error rate, meaning that the scatter index and MAPE are largely distorted by the errors of this asset. When looking at the error rates excluding the outlier values of RMHJ the scatter index is 26.52% and MAPE is 24.26% which is a much better representation of the overall error of the model.

4.2.3 Discussion

Reasonable and accurate forecasts have the potential to generate high investment returns. Which is why forecasts are used in the proposed model. The error is acceptable in the first test period, the model is quite good at making predictions in this period. The accuracy decreases in the crisis period due to unexpected effects of the pandemic. The mean price of the assets decreases from R243.96 (2019) to R227.85 (2020). Approximately 68% of the assets experienced a decline in the average stock price. This is largely due to the Covid 19 pandemic, however, some companies experienced events independent of the crisis that resulted in the drop in their stock prices dropping. The average error rates have also increased from 2019 to 2020, this is expected as the model was trained on non-crisis data and as a result not learned how to react to the unexpected changes that occur during a crisis period. The model has thus successfully been implemented to predict the stock prices. The graphs for all the asset predictions can be found in Appendix C.

4.3 Clustering Model

Diversification has been identified as an important component of portfolio optimization. It enables the portfolio to earn more returns and be more cushioned when a single asset is experiencing a loss. Clustering is used to partition the assets in such a way that assets in the same group are similar and different from assets in another group. Thus, selecting assets from each group should result in a diversified portfolio.

Selecting K: Holding too many different assets is difficult to manage for an investor. As a result many studies consider a portfolio with less stocks. Therefore, the considered number of clusters is 2 to 14. The elbow curve is implemented to decide on the number of clusters. It is presented in Figure 4.3. K, the number of clusters is in the x-axis and the y-axis represents the evaluation metrics, distortion score. The model is trained starting with 2 clusters and the performance is plotted, the number of clusters is increased until it reaches 14. The performance improves the more clusters are added. The cluster value where the improvement becomes constant is chosen as the number of clusters, in this case k=5. The green dotted line represents the computational time incurred when running the model. This method is implemented using yellowbrick KElbowVisualisation.

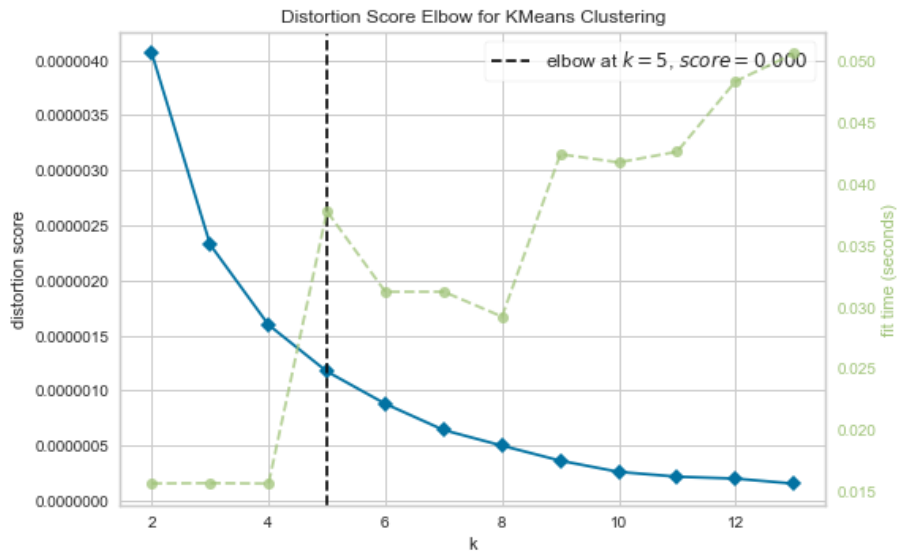


FIGURE 4.3: Elbow curve

Model Initialisation: For this model the Sklearn kmeans++ initialisation is used to initialise the model. Kmeans ++ specifies the procedure to initialise the cluster centers. The intuition behind the approach is that spreading out the k initial cluster centroids is a good thing, the first centroid is chosen randomly, the subsequent centroids are chosen such that they are far away from this centroid and each other. This increases the chance that they are in different clusters. Table 4.3 shows the average daily returns, risk, Sharpe ratio and results of the clustering algorithm with the JSE sectors. The assets with the highest Sharpe ratio in each cluster are selected as the representative and made bold in the table. ANGJ has the best performance in cluster 4, so it is taken as the cluster representative. IMPJ is selected for cluster3, FSRJ for cluster2, CCOJ represents cluster 1 and cluster 0 is represented by BHPJ.

As can be seen in Table 4.4 the size of the clusters varies from 5.88% (cluster 3) to 44.12% (cluster 0). The proportion of the clusters shows that the cluster size is not consistent. The silhouette score is 0.99975, which is close to 1 and the DBI is 0.00034, which is close to 0. The silhouette score indicates that the assets are well matched in their clusters and poorly matched in other clusters. This means that the clusters are dense and well separated. The model is successful at partitioning the data, splitting the data in a good way.

Discussion The clustering algorithm did not separate the assets based on the sectors used by the JSE. Each cluster contains assets from different sectors. Both the healthcare sector assets can be found in the same cluster (Cluster 1), the two telecommunications sector assets can be found in cluster 1 and 0. The basic materials sector assets can be found in 4 of the 5 clusters and the financials sector can be found in 3 of the 5 clusters. The two industrials sector assets can be found in cluster 3 and 4. There is only 1 asset from the technology sector and it is in cluster 2. The assets selected with the highest Sharpe ratio are from the basic materials sector (ANGJ, IMPJ, BHPJ) and the financials sector (FSRJ, CCOJ). Theoretically, making clusters to segregate the data and selecting the best performing asset in each group should create diversification and improve portfolio returns. Assets are most likely to be grouped together if their stock experiences similar movements. By representing each cluster in a portfolio diversification is achieved and performance improves.

TABLE 4.3: Asset Clustering

Label	Returns	Risk	Sharpe Ratio	Cluster	Sector
ANGJ	**0.00354**	**0.00107**	**0.10802**	**4**	**Basic Materials**
KIOJ	0.00150	0.00110	0.04532	4	Basic Materials
ARIJ	0.00073	0.00058	0.03037	4	Basic Materials
EXXJ	0.00038	0.00062	0.01519	4	Basic Materials
BVTJ	0.00026	0.00037	0.01361	4	Industrials
IMPJ	**0.00562**	**0.00193**	**0.12785**	**3**	**Basic Materials**
IPLJ	-0.00158	0.00136	-0.04288	3	Industrials
FSRJ	**0.00006**	**0.00031**	**0.00359**	**2**	**Financials**
SLMJ	0.00003	0.00091	0.00095	2	Financials
SBKJ	-0.00001	0.00066	-0.00038	2	Financials
RMHJ	-0.00011	0.00023	-0.00733	2	Financials
MNPJ	-0.00013	0.00012	-0.01214	2	Basic Materials
NPNJn	-0.00109	0.00117	-0.03189	2	Technology
DSYJ	-0.00125	0.00051	-0.05534	2	Financials
CCOJ	**0.00288**	**0.00166**	**0.07069**	**1**	**Financials**
LHCJ	0.00106	0.00077	0.03834	1	Health Care
MTNJ	0.00053	0.00052	0.02321	1	Telecommunications
OMUJ	0.00080	0.00141	0.02123	1	Financials
APNJ	0.00006	0.00149	0.00159	1	Health Care
BHPJ	**0.00085**	**0.00032**	**0.04783**	**0**	**Basic Materials**
CFRJ	0.00085	0.00045	0.03985	0	Consumer Goods
WHLJ	0.00073	0.00114	0.02167	0	Consumer Goods
INPJ	0.00048	0.00053	0.02111	0	Financials
INLJ	0.00038	0.00040	0.01917	0	Financials
REMJ	0.00052	0.00088	0.01744	0	Financials
RNIJ	0.00028	0.00078	0.01018	0	Financials
ABGJ	0.00017	0.00069	0.00634	0	Financials
BTIJ	-0.00013	0.00175	-0.00321	0	Consumer Goods
TBSJ	-0.00019	0.00036	-0.01009	0	Consumer Goods
VODJ	-0.00016	0.00024	-0.01028	0	Telecommunications
GRTJ	-0.00029	0.00041	-0.01447	0	Financials
NEDJ	-0.00073	0.00024	-0.04709	0	Financials
SOLJ	-0.00169	0.00101	-0.05340	0	Basic Materials
SHPJ	-0.00134	0.00042	-0.06575	0	Consumer Services

Table 4.4: Cluster Analysis

Cluster	Size %	Average Returns	Average Risk	Average Sharpe Ratio
0	44.12	-0.00002	0.000640254	-0.001380027
1	14.71	0.00107	0.00117	0.03101
2	20.59	-0.00036	0.00056	-0.01465
3	5.88	0.00202	0.00164	0.04249
4	14.71	0.00128	0.00075	0.04250

4.4 Portfolio Construction

The Monte Carlo method is applied to generate different portfolios. First, it generates random portfolio weights for the selected assets and calculates the corresponding portfolio measurements such as expected returns, volatility, and Sharpe ratio. The process is repeated 50,000 times. From a statistical point of view, 50,000 random portfolios cover most possible portfolios and different weights are sufficiently represented. The best one is chosen from these 50,000 portfolios, according to the Sharpe ratio.

All possible generated portfolio scenarios can be seen as a colour map as the distribution of random weights in Figure 4.4. The green star represents the portfolio with the lowest risk and the black star is the portfolio with the highest risk. Whereas, the red star represents the portfolio with the highest Sharpe ratio value, this will be called portfolio A.

Figure 4.5 shows some of the Mote Carlo simulated portfolios. The optimal portfolio A is the first on the list with the highest Sharpe ratio of 0.1727. The allocated weights for the selected assets are 32.0% for IMPJ, ANGJ for 24.7%, CCOJ for 26.6%, BHPJ for 16.6% and FSRJ with 0.1%. Since the MC simulations randomly assigns the weights there is no considerations to the returns of the assets this prevents the portfolio having more weights distributed to assets that have higher returns. So optimal weight allocation is not dependent on the performance of the individual asset as it is assigned at random.

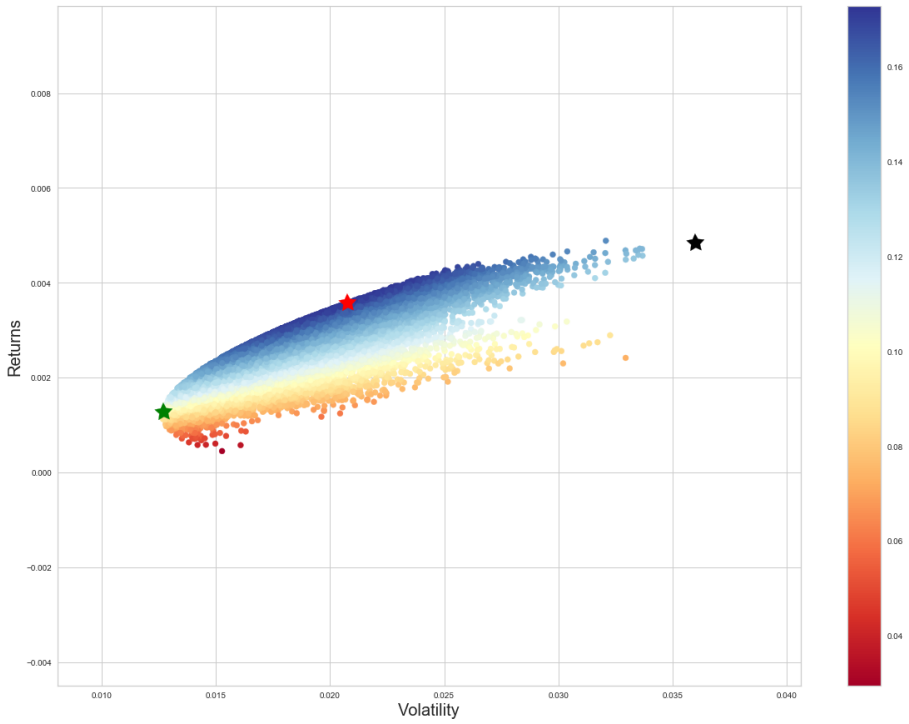


FIGURE 4.4: Efficient Frontier

FIGURE 4.5: Monte Carlo Simulated Portfolios

Return	Risk	Sharpe_Ratio	IMPJ	ANGJ	CCOJ	BHPJ	FSRJ
0.003580	0.020736	0.172665	0.319985	0.246784	0.266580	0.165647	0.001004
0.003672	0.021294	0.172445	0.339507	0.251862	0.259633	0.146874	0.002123
0.003661	0.021241	0.172376	0.314852	0.280276	0.274891	0.127237	0.002744
0.003635	0.021090	0.172353	0.338140	0.241005	0.261611	0.150226	0.009018
0.003521	0.020431	0.172315	0.288257	0.275129	0.274268	0.160831	0.001515
...	...	...	...	...	...	...	...
0.000581	0.014202	0.040917	0.015176	0.067275	0.005972	0.231892	0.679686
0.000608	0.014979	0.040612	0.059302	0.005349	0.021472	0.173413	0.740464
0.000585	0.014567	0.040190	0.016463	0.038770	0.050008	0.196681	0.698077
0.000576	0.016085	0.035796	0.017495	0.030524	0.102423	0.026509	0.823048
0.000451	0.015273	0.029524	0.026655	0.023437	0.017434	0.138481	0.793992

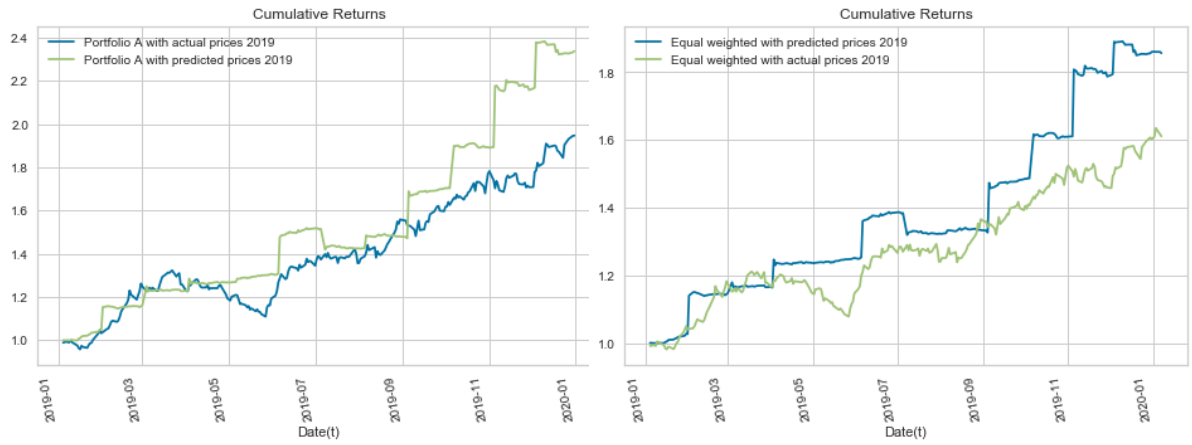
4.5 Optimal Portfolio

To measure the overall performance of the proposed methodology comparisons are made between the portfolio selected (Portfolio A), the equal weighted portfolio and the JSE top 40 index.

4.5.1 Performance in 2019 test

The predicted returns and actual returns are evaluated for both portfolio A and the equal weighted portfolio in Figure 4.6. For portfolio A 4.6a the predicted returns are higher than the actual returns at the end of the period the same can be observed for the equal weighted portfolio 4.6b.

FIGURE 4.6: Predicted returns vs Real returns 2019



(A) Optimal portfolio with forecast prices Vs actual price

(B) Equal weighted portfolio with forecast prices vs actual prices

Figure 4.7 represents cumulative returns for the 3 portfolios; Equal weighted portfolio, portfolio A and the JSE index using the actual prices. The JSE top 40 index remains consistently ranging between 0% to 20% returns whereas both the equal weighted and portfolio A continue to grow above it. It can be observed that portfolio A outperforms both the equal weighted and the JSE index. In the beginning of the test period (January) the JSE index performs better than the created portfolios, however, this is the only period that it does so. At the end of the test period the JSE index has achieved returns of 11.36%, the equal weighted has obtained 61.14% returns and portfolio A has 94.73% returns.

FIGURE 4.7: Portfolio returns 2019

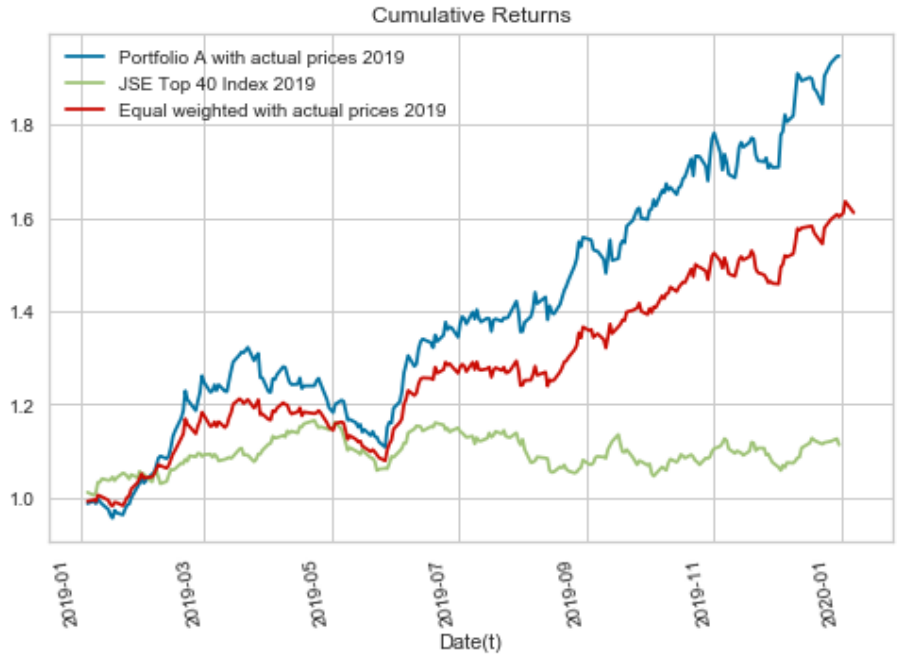


TABLE 4.5: Portfolio Performance 2019

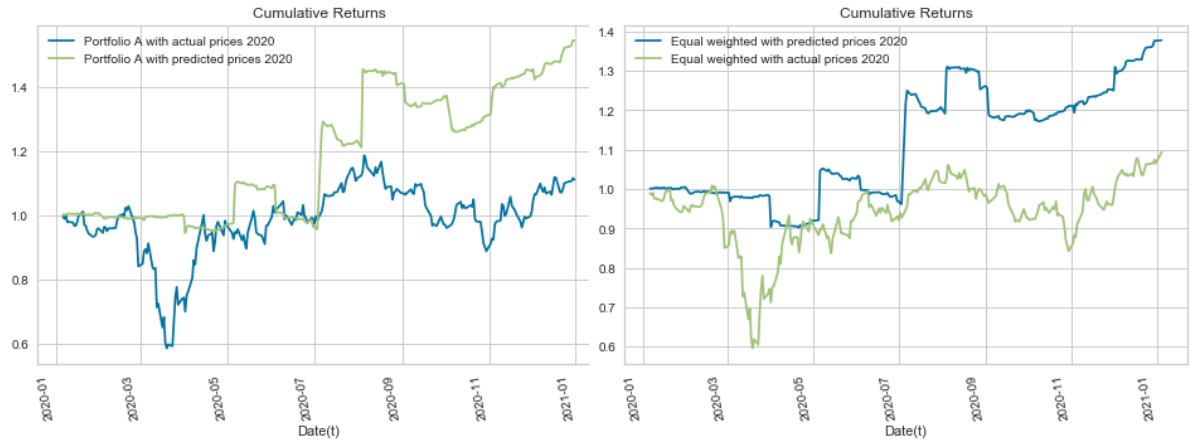
Portfolio	Sharpe Ratio	Beta Coefficient	Treynor Ratio (TR)	Information Ratio (IR)
JSE Top 40 Index	0.056	-	-	-
Equal Weighted Portfolio	0.19	0.41	0.005	0.136
Portfolio A	0.199	0.205	0.014	0.15

Table 4.5 presents the performance of the 3 portfolios considered in this research. The beta coefficient,Treynor ratio and information ratio are not computed for JSE top 40 index because the index is used as the benchmark when calculating these values for the other portfolios. The table shows that portfolio A has the highest Sharpe ratio, Treynor ratio and information ratio and the equal weighted portfolio has the highest beta. Theoretically the equal weighted portfolio is less volatile than portfolio A as it has the lowest beta, this also shows that the equal weighted portfolio is also more correlated to the JSE index. Treynor ratio measures excess returns for each unit of risk and portfolio A has the higher value meaning that it achieves higher returns than the equal weighted portfolio. The higher information ratio achieved by portfolio A shows a better ability to outperform the returns of the JSE index.

4.5.2 Performance in 2020 stress test

The predicted and actual returns are compared of both portfolio A and the equal weighted portfolio in Table 4.8. It can be observed that the predicted returns are higher than the actual returns. The model does not track the drop in March when the crisis is in full effect. The predictions are more volatile and do not track the trend of the actual returns. The actual cumulative returns of the portfolios considered are found in Figure 4.9. It can be observed that all portfolios start at the same point with the JSE index earning more returns till February after which the assets returns are competitive. When the drop occurs in March, JSE does not drop as low as the created portfolios. However, in April portfolio A starts earning more returns than both the JSE index and the equal weighted portfolio. During the sharp drop in the end of October JSE index again does not drop as low as the created portfolios meaning that it is more stable and consistent. At the end of the period portfolio A has earned 11.02% returns, equal weighted portfolio has 6.55% and the JSE has 5.52% returns. Showing that portfolio A has outperformed the other portfolios during this crisis period.

FIGURE 4.8: Predicted returns vs Real returns 2020



(A) Optimal portfolio with forecast prices Vs actual price

(B) Equal weighted portfolio with forecast prices vs actual prices

In Table 4.6 the performance of the portfolios for this period is presented. Portfolio A outperforms the equal weighted portfolios on all the performance measures presented. It can be observed that the beta coefficient for both portfolios has increased drastically which is expected in a crisis period. The beta is higher than 1 on both

FIGURE 4.9: Portfolio returns 2020

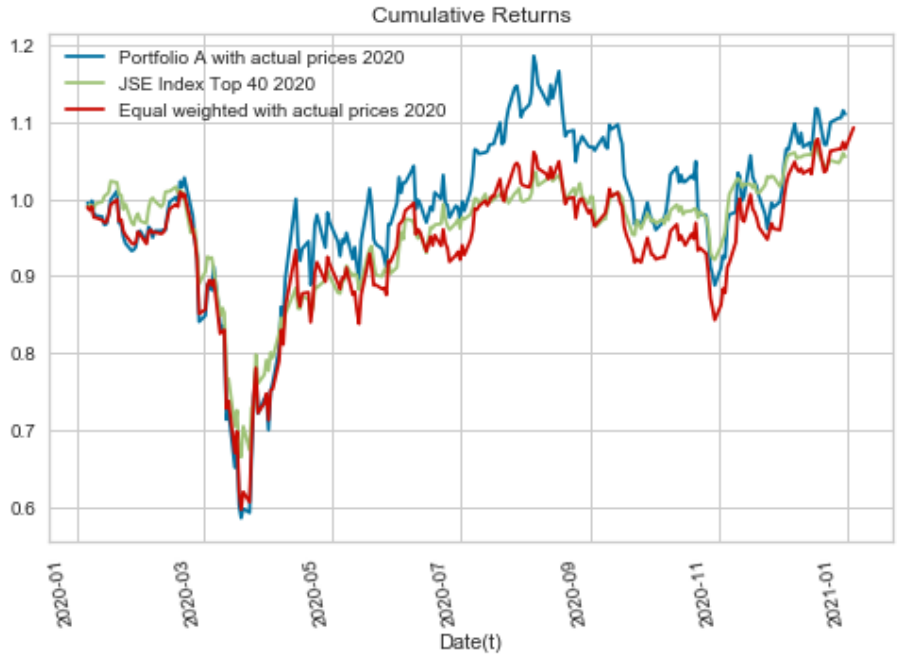


TABLE 4.6: Portfolio Performance 2020

Portfolio	Sharpe Ratio	Beta Coefficient	Treynor Ratio (TR)	Information Ratio (IR)
JSE Top 40 Index	0.021	-	-	-
Equal Weighted Portfolio	0.027	1.14	0.0005	0.014
Portfolio A	0.029	1.207	0.0007	0.024

portfolios, meaning that the portfolios created are more volatile than the JSE index during this stress period. The risk on both these portfolios is thus higher than that of the JSE index during this period. This large amount of risk has also resulted in a drop in the Treynor ratio as compared to previous test period 4.5. The information ratio has also decreased from previous period meaning that the ability for these portfolios to obtain excess returns over the JSE index has also reduced. Portfolio A had the strongest recovery after the crisis event, the portfolio has maintained a high Sharpe ratio during this period of increasing risk.

4.5.3 Discussion

As can be seen in Figure 4.6 and 4.8 the predictive model inflates the returns and the model does not respond to short term dips in the returns. Portfolio A and

the equal weighted portfolio have the same trend because they are created using the same assets just with different weight allocations. However, portfolio A earns more returns meaning that the weight allocation is better on this portfolio. During this crisis period the performance of all the portfolios have dropped compared to those achieved in the previous test period. This is expected as many of crisis are unforeseeable and take the broader market by surprise.

4.6 Summary

The first step of implementing the methodology in Chapter 3 is to forecast the prices for each asset. Evaluating the performance of these predictions showed that the forecasts were good in the test period but suffered greatly in the stress test period. The forecast model responds slowly to unexpected large movements and over prices the stocks, it is also more volatile and erratic in its price movements. This behaviour is heightened in the stress test resulting in larger errors. However, the largest error in this period was not due to the pandemic but to an unrelated event occurring within the asset.

The second step is to use the predictions to group the data into clusters to achieve diversification. K-means clustering is implemented and 5 clusters are used to partition the assets. The cluster partitions did not align with the JSE sectors division. The silhouette score and DBI index achieved by the algorithm suggests that the clusters are well separated. The assets selected for the portfolios are ANGJ, IMPJ, FSRJ, CCOJ, BHPJ all have the highest Sharpe ratio in their respective clusters.

The third step is to create portfolios using Monte Carlo simulations. 50,000 portfolios were created using this method and the portfolio with the highest Sharpe ratio was chosen and referred to as Portfolio A. The weight allocation is not influenced by the performance of the asset since the weights were assigned randomly during the simulations.

After portfolio A is selected the performance is measured against the JSE top 40 index and the equal weighted portfolio. In the 2019 test period portfolio A outperformed the other portfolios in terms of returns achieving 94.73% returns with equal weighted earning 61.14% and JSE having 11.36%. In the 2020 stress test period the performance of all the portfolios reduces and the risk increases. However, it can be observed that the JSE index is more stable and less risky as it experiences less drastic drops and

less deviation in returns. Even though this is the case portfolio A has the strongest recovery during this period and earned 11.02% returns whereas JSE earned 5.52% and equal weighted had 6.55% returns.
The methodology presented in this research has been thoroughly evaluated and tested on two different periods. It can be seen through these tests that the overall model is successful and presents competitive results as the created portfolios outperform the JSE index on both test periods.

Chapter 5

Summary, Conclusions and Recommendations

5.1 Summary

This research focuses on one of the fundamental problems in financial markets, portfolio optimisation, proposing a machine learning model that uses predicted returns as expected returns. The aim of the research is to use these forecasted returns to produce an optimal portfolio. A number of approaches have been implemented to tackle this problem. One of the most common is to predict stock prices and use these predictions as expected returns [23, 37, 21, 25] another is to use clustering to achieve diversification [28, 33]. Thus, the use of both these in one methodology should result in an optimal portfolio. The methodology investigated uses a regressor chain XGBoost model to forecast stock prices and a k-means algorithm to diversify and select assets for the portfolio, a portfolio created with Monte Carlo simulations is selected as the optimal portfolio. The benefits of multi-target regression are under-explored and there has been no other studies in literature that have implemented the developed model. It presents the use of multi-output regression for stock price prediction and combine different models to introduce one machine learning based portfolio optimisation solution.

5.2 Conclusions

Error measures are used to evaluate the performance of the forecasting model. The model produces low average error rates in the test period with RMSE of R33.31, scatter index is 12.26%, and MAPE is 9.96%. However, the error increases during the stress test to RMSE of R40.86, scatter index is 29.92%, and MAPE is 87.78%. The high MAPE error in the stress period is mostly due to an asset experiencing some changes within the company. This drastic drop in the asset price results in a MAPE of 2183.79% which largely distorts the average MAPE. The prediction model performs better during the first test period where there is no large unexpected stress in the market. When tested during the Covid-19 period the performance of the model decreases drastically because the model was not trained using crisis data and thus struggles to predict during this period. This results in the model being too slow to respond to the drastic changes that occur in times of stress and the model is also unable to anticipate these changes resulting in higher error rates.

The predicted results are used as expected returns and risk. These values are fed into the k-means clustering algorithm which divides the assets into 5 groups. Assets with the highest Sharpe ratio are selected in each cluster. When comparing the selected assets with the JSE categories it s clear that the clusters are not grouping them according to sectors. The cluster groups according to returns and risk as opposed to the type of company the asset fall under. This may be because assets in the same sector can have different risk/return movements and assets in different sectors can have the same risk/return movements. Monte Carlo simulations are used to allocate weights to the selected assets creating portfolios, then portfolio with the highest Sharpe ratio is selected as the optimal portfolio (portfolio A). The optimal portfolio weight allocations are 32% in IMPJ, 24.7% in ANGJ, 26.7% in CCOJ, 16.6% in BHPJ, and 0.1% in FSRJ. The assets in the optimal portfolio come from 2 sectors.When allocating weights there is no consideration to which asset have the most returns as the simulations randomly assigns the weights. However, since the portfolio has the highest Sharpe ratio the return and risk adjustment is acceptable.

The experimental results show that portfolio A outperforms the JSE index and equal weighted portfolio. In the 2019 test period the portfolio A achieved Sharpe ratio of

0.199 and 94.73% returns opposed to 0.056 Sharpe ratio with 11.36% returns achieved by the JSE index and Sharpe ratio of 0.19 with 61.14% returns earned by the equal weighted portfolio. In the 2020 stress test period the portfolio A achieved Sharpe ratio of 0.029 and 11.02% returns opposed to 0.021 Sharpe ratio with 5.52% returns achieved by the JSE index and Sharpe ratio of 0.027 with 6.55% returns earned by the equal weighted portfolio. The decrease in returns and Sharpe ratios as well as the increase in risk during the stress test period is expected as this is known to occur during a financial crisis. Portfolio A manages to recover better than the benchmark portfolios during this period. Thus, this portfolio is well selected and can endure stress over a crisis period.

This research further extends the literature concerning portfolio optimisation by using multi-output prediction-based algorithm. The forecasted prices are then incorporated in a clustering algorithm with the aim of creating diversification within the portfolio that is created using Monte Carlo simulations. These 3 methods have not been used together as an approach to tackle portfolio optimisation. The results show that portfolio A delivers the best performance in both test periods. The model performance during the crisis shows the rationality and practicability of the methodology.

5.3 Recommendations

Although this research provides useful results, there are some limitations to this study. The proposed research can further be improved and extended from the following aspects. First, only the JSE top 40 assets data has been considered. However, the size of the market is quite large and these assets do not represent all the possibilities available in the market. Extending this work to include different types of financial assets will further improve the model making it more realistic. Secondly, the model only applies simple historical data as input features. More data can be included such as the financial information of the asset and other economic features. This could further improve the accuracy of the prediction as the model will have more data. Also, there may exist other risk metrics that are more suitable than variance in building portfolio optimization models. Exploring other risk measures could also result in a better performing portfolio.

Appendix A

Technical Indicator Formulas

Simple Moving Average (SMA):

$$SMA = \frac{1}{n}\sum_{i=1}^{n} P_i$$

where $n = 20, 50100$

Exponential Moving Average (EMA):

$$EMA = (P - EMA_{prev}) * K + EMA_{prev}$$

where P is the price for the current period, EMA_{prev} is the Exponential moving Average for the previous period, K is the smoothing constant, equal to $\frac{2}{n+1}$, n is the number of periods in a simple moving average roughly approximated by the EMA. n=9

Moving Average Convergence Divergence (MACD):

$$MACD == FastMA - SlowMA$$

where FastMA is the shorter moving average (12 days) and SlowMA is the longer moving average (26 days).

Bollinger Band (BB): Middle Band = n-period moving average,
Upper Band = Middle Band + (y * n-period standard deviation),

Lower Band = Middle Band - (y * n-period standard deviation)

where n = 20.

Relative Strength Index (RSI):

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

where RS is ratio of smoothed average of n-period(14 days) gains divided by the absolute value of the smoothed average of n-period losses.

Commodity Channel Index (CCI):

$$CCI = \frac{1}{0.015} \frac{P_t - SMA(P_t)}{MD(P_t)}$$
$$P_t = \frac{P_{high} + P_{low} + P_{close}}{3}$$

where P_t is the typical price, MD is mean deviation of the absolute value, and n= 20 for the averages.

Stochastic Oscillator (SO):

$$\%K = \frac{P - Low_n}{High_n - Low_n} * 100$$
$$\%D = \frac{\%K_1 + \%K_2 + \%K_3}{3}$$

where Low_n refers to the lowest price in the last n days, $High_n$ refers to the highest price over n days, n = 14.

On Balance Volume(OBV):

$$OBV = OBV_{prev} + \begin{cases} volume & \text{if } close > close_{prev} \\ 0 & \text{if } close = close_{prev} \\ -volume & \text{if } close < close_{prev} \end{cases} \tag{A.1}$$

Average Directional Movement (ADX): UpMove = $High_t - High_{t-1}$
DownMove =$Low_{t-1} - Low_t$
if UpMove > DownMove and UpMove > 0, then +DM = UpMove, else +DM = 0
if DownMove > UpMove and DownMove > 0, then -DM = DownMove, else -DM = 0
TR = $max(High_t, Close_{t-1}) - min(Low_t, Close_{t-1})$
ATR = $\frac{1}{n}\sum_{i=1}^{n} TR_i$
$ATR_t = \frac{ATR_{t-1}*(n-1)+TR_t}{n}$
+DI = 100 * SMA(+DM)/ ATR_t
-DI = 100 * SMA(-DM)/ ATR_t
ADX = 100 * $\frac{SMA(|(+DI-DI)|)}{(+DI+-DI)}$
Where $High_t$ refers to today's highest price. $High_{t-1}$ is yesterday's highest price, Low_t refers to today's lowest price. Low_{t-1} is yesterday's lowest price, $Close_{t-1}$ is yesterday's closing price, (+/-)DM refers to (positive/negative) directional movement, (+/-)DI refers to the (positive/negative) directional indicator, TR is the true range and ATR is the average true range.

Appendix B

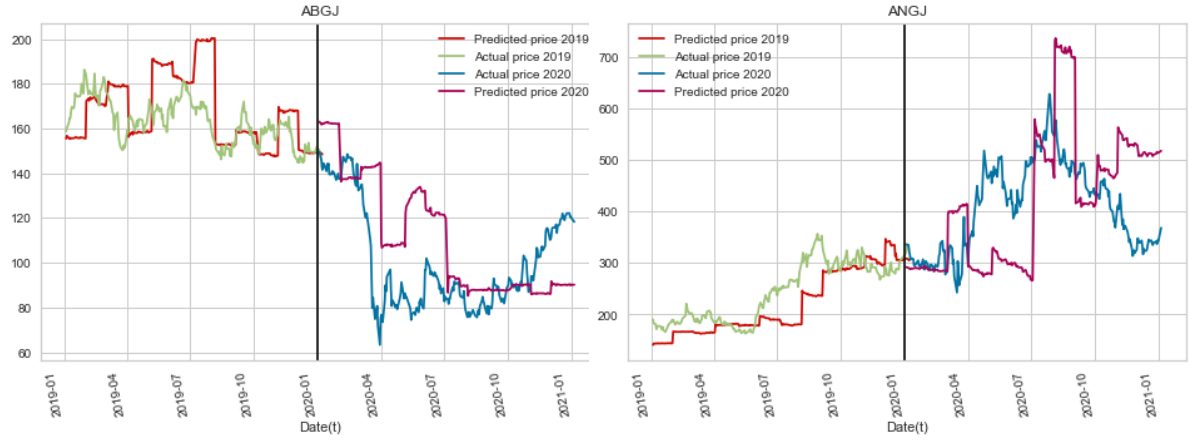
JSE Top 40 Assets

Table B.1: JSE Top 40 Assets

Ticker	Company name	Sector
ABGJ	Absa Group Ltd	Financials
ARIJ	African Rainbow Minerals Ltd	Basic Materials
ANGJ	AngloGold Ashanti Ltd	Basic Materials
APNJ	Aspen Pharmacare Holdings Ltd	Health Care
BHPJ	BHP Group PLC	Basic Materials
BVTJ	Bidvest Group Ltd	Industrials
BTIJ	British American Tobacco PLC	Consumer Goods
CCOJ	Capital & Counties Properties PLC	Financials
DSYJ	Discovery Holdings Ltd	Financials
EXXJ	Exxaro Resources Ltd	Basic Materials
FSRJ	Firstrand Ltd	Financials
GRTJ	Growthpoint Properties Ltd	Financials
IMPJ	Impala Platinum Holdings Ltd	Basic Materials
IPLJ	Imperial Holdings	Industrials
INPJ	Investec PLC	Financials
INLJ	Investec Ltd	Financials
KIOJ	Kumba Iron Ore Ltd	Basic Materials
LHCJ	Life Healthcare	Health Care
MNPJ	Mondi PLC	Basic Materials
MTNJ	MTN Group Ltd	Telecommunications
NPNJn	Naspers Ltd	Technology
NEDJ	Nedbank Group Ltd	Financials
OMUJ	Old Mutual Ltd	Financials
RNIJ	Reinet Investments SCA	Financials
REMJ	Remgro Ltd	Financials
CFRJ	Compagie Financiere Richemont SA DRC	Consumer Goods
RMHHJ	RMB Holdings	Financials
SLMJ	Sanlam Ltd	Financials
SOLJ	Sasol Ltd	Basic Materials
SHPJ	Shoprite Holdings	Consumer Services
SBKJ	Standad Bank Group Ltd	Financials
TBSJ	Tiger Brands Ltd	Consumer Goods
VODJ	Vodacom Group Ltd	Telecommunications
WHLJ	Woolworths Holdings Ltd	Consumer Goods

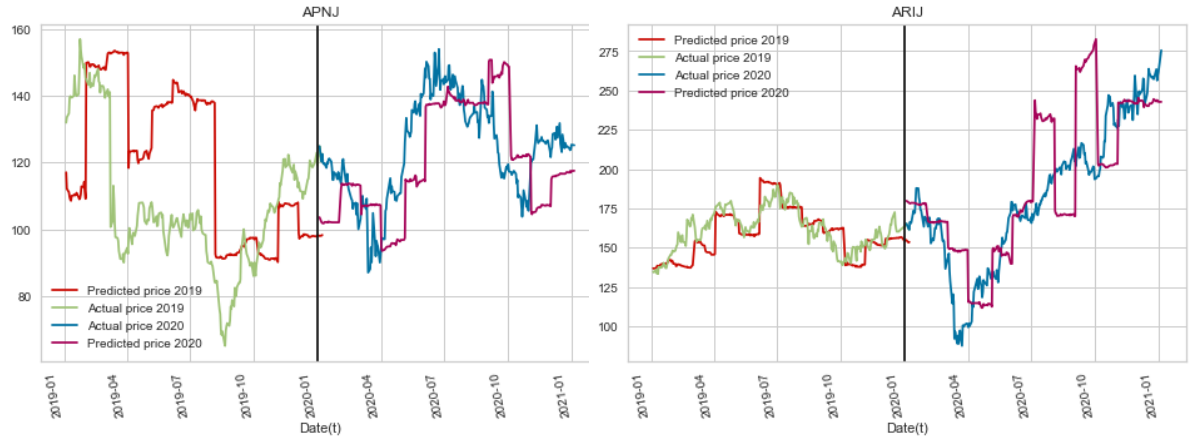
Appendix C

Assets Forecast



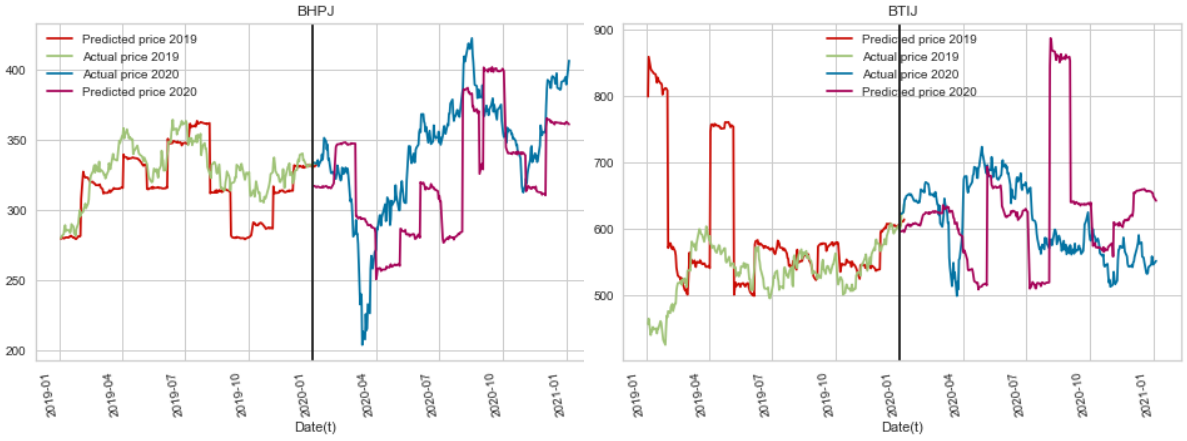
(A) ABGJ forecast

(B) ANGJ forecast



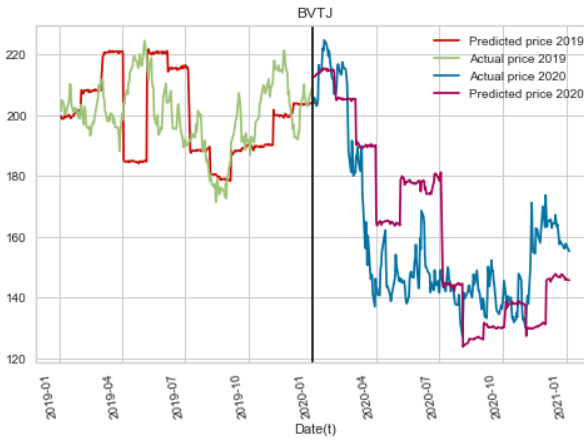
(C) APNJ forecast

(D) ARIJ forecast

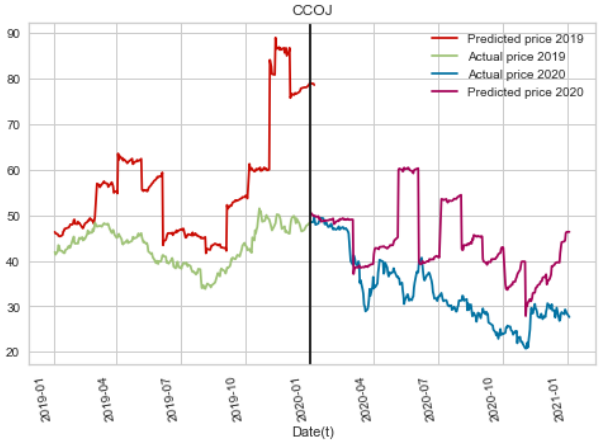


(A) BHPJ forecast

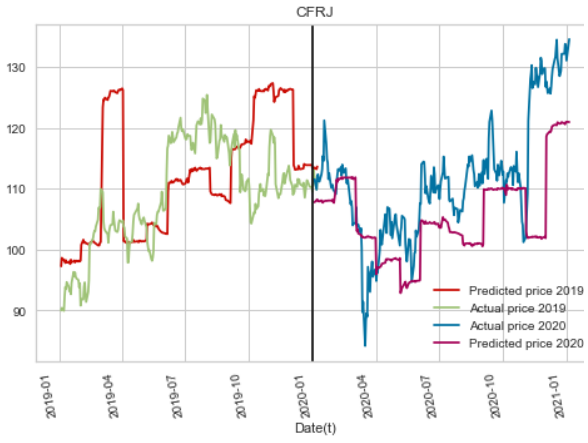
(B) BTIJ forecast



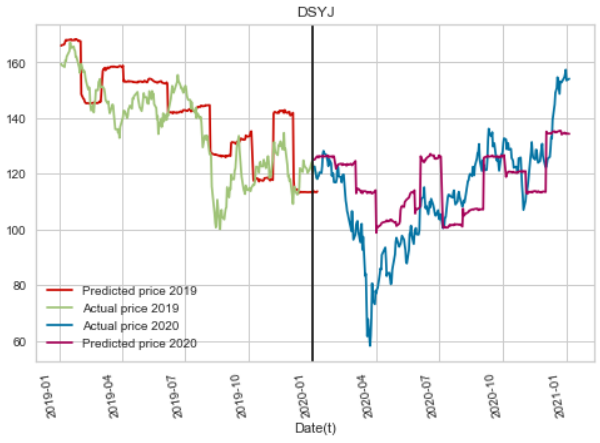
(C) BVTJ forecast



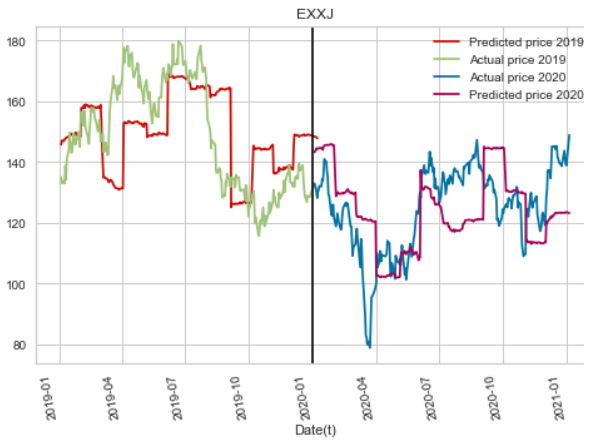
(D) CCOJ forecast



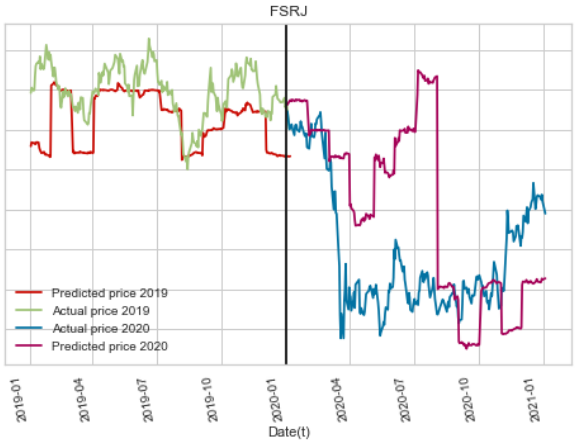
(E) CFRJ forecast



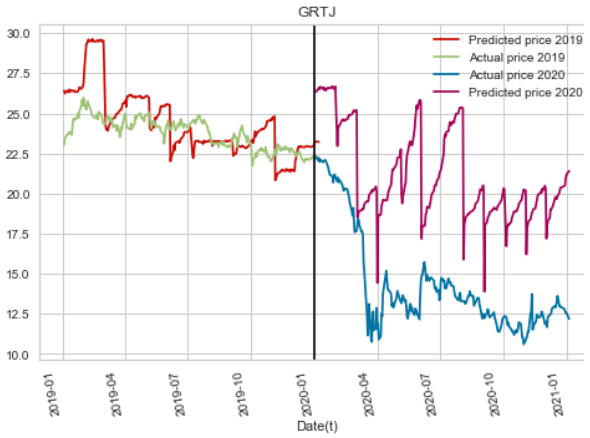
(F) DSYJ forecast



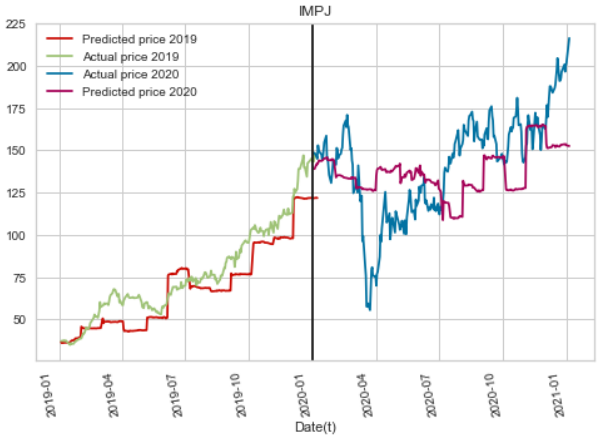
(A) EXXJ forecast



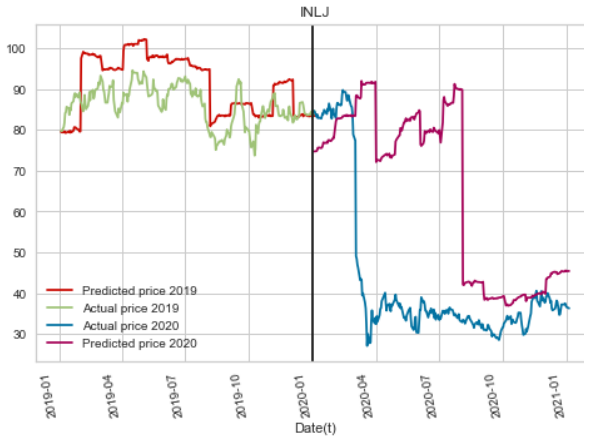
(B) FSRJ forecast



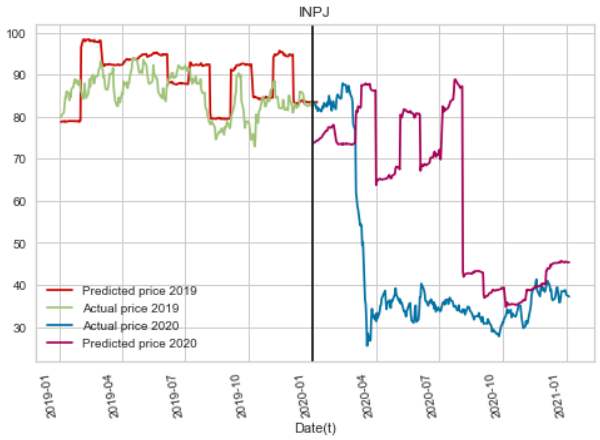
(C) GRTJ forecast



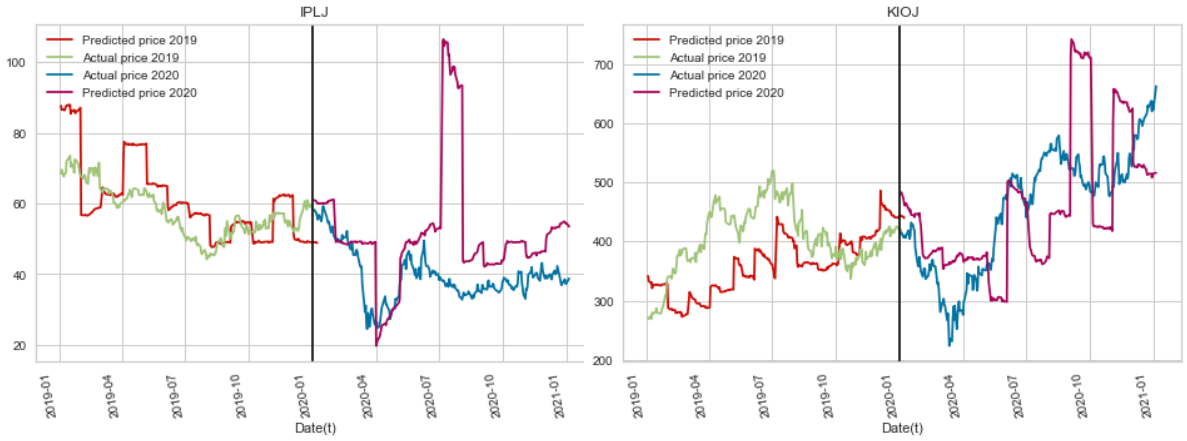
(D) IMPJ forecast



(E) INLJ forecast

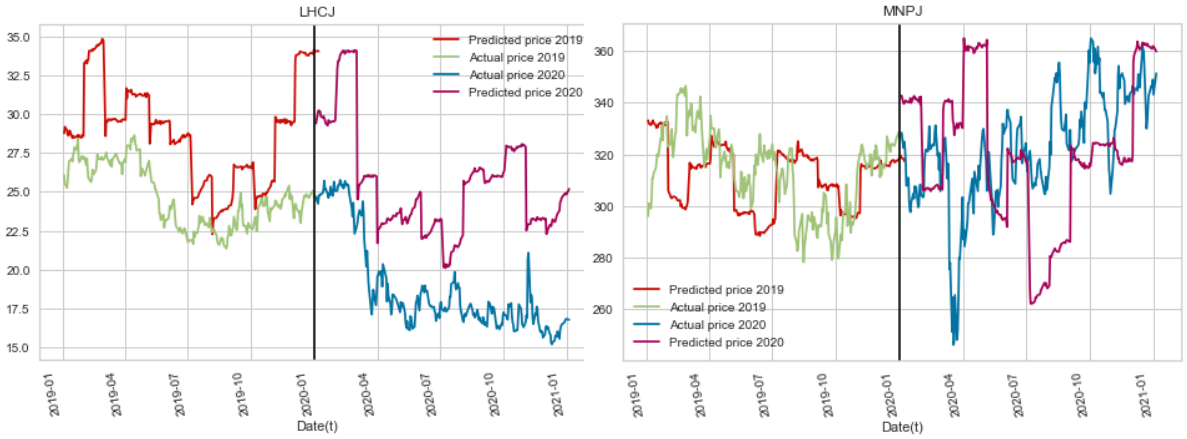


(F) INPJ forecast



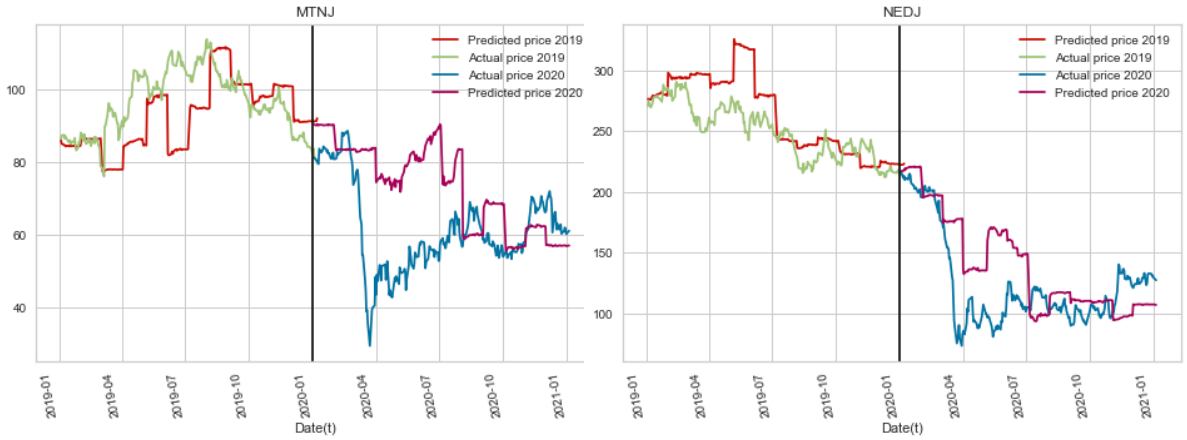
(A) IPLJ forecast

(B) KIOJ forecast



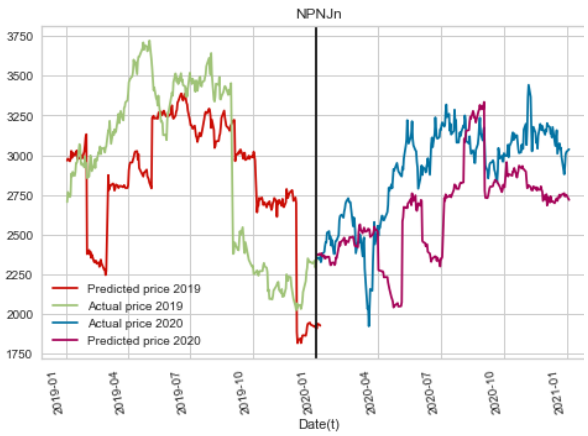
(C) LHCJ forecast

(D) MNPJ forecast

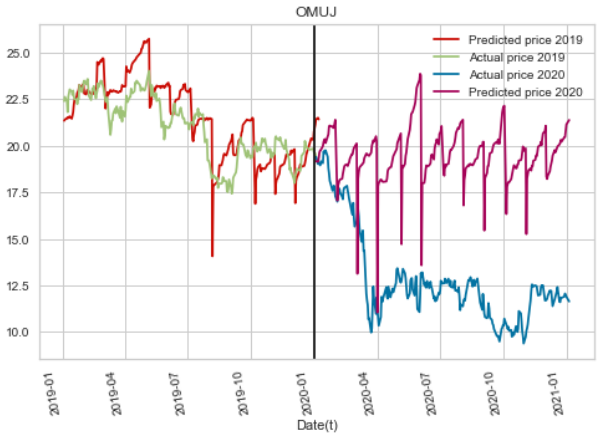


(E) MTNJ forecast

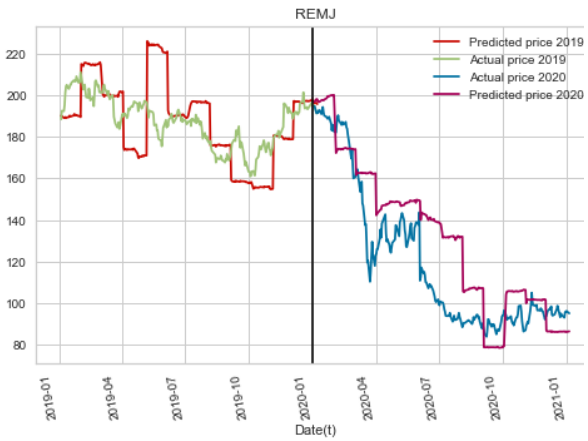
(F) NEDJ forecast



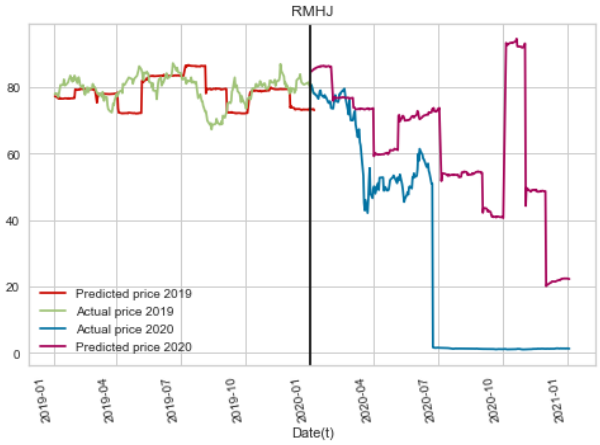
(A) NPNJ forecast



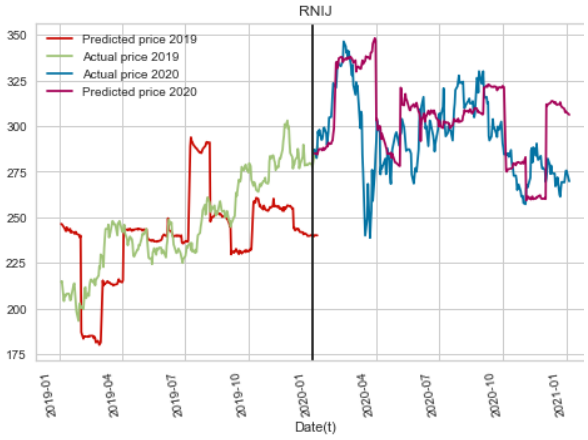
(B) OMUJ forecast



(C) REMJ forecast



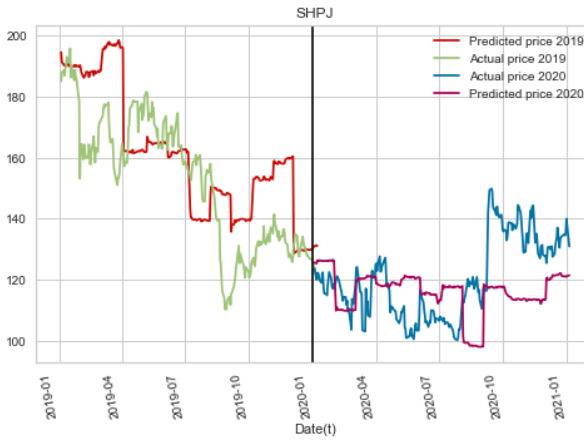
(D) RMHJ forecast



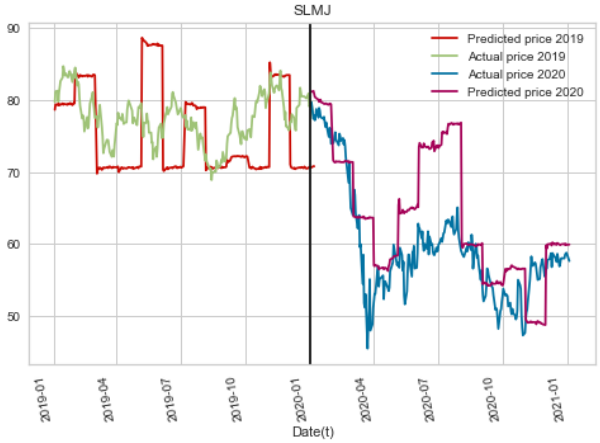
(E) RNIJ forecast



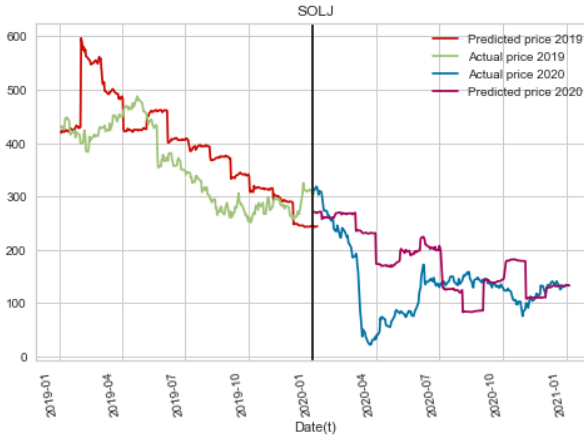
(F) SBKJ forecast



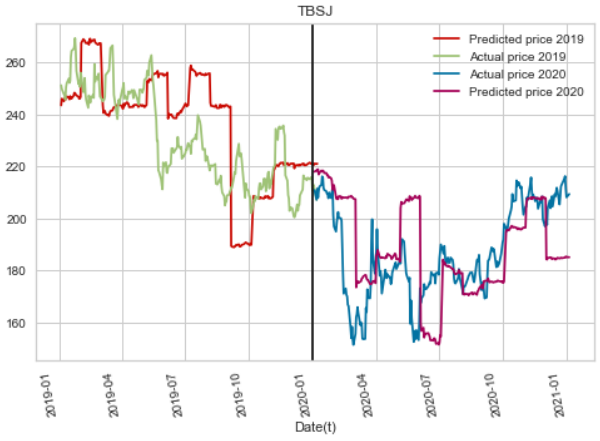
(A) SHPJ forecast



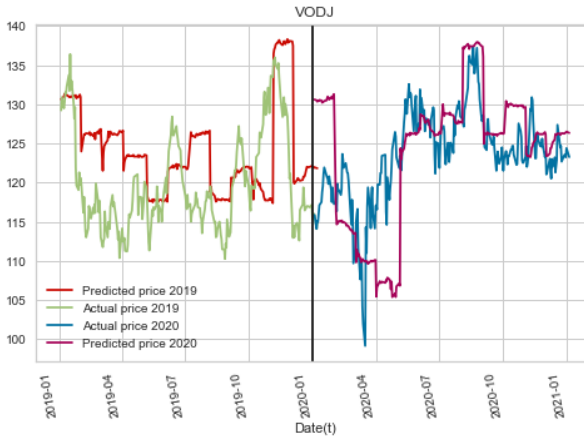
(B) SLMJ forecast



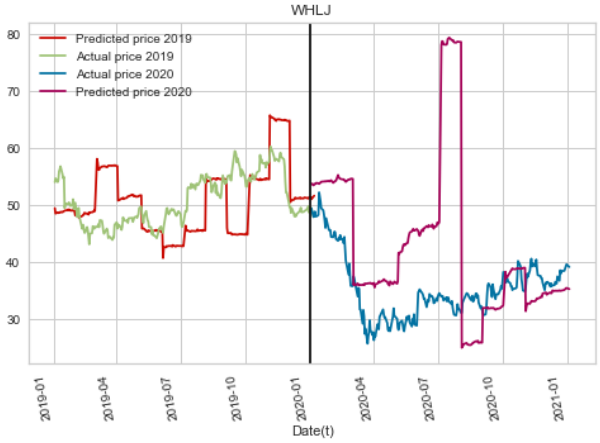
(C) SOLJ forecast



(D) TBSJ forecast



(E) VODJ forecast



(F) WHLJ forecast

Bibliography

[1] Gah-Yi Ban, Noureddine El Karoui, and Andrew E. B. Lim. "Machine Learning and Portfolio Optimization". In: *Management Science* 64.3 (2018), pp. 1136–1154. DOI: 10.1287/mnsc.2016.2644. eprint: https://doi.org/10.1287/mnsc.2016.2644. URL: https://doi.org/10.1287/mnsc.2016.2644.

[2] Patrick Behr, André Güttler, and Felix Miebs. "Is minimum-variance investing really worth the while? An analysis with robust performance inference". In: *EDHEC-Risk working paper* (2008).

[3] Shlomo Benartzi and Richard H Thaler. "Naive diversification strategies in defined contribution saving plans". In: *American economic review* 91.1 (2001), pp. 79–98.

[4] Hieu Cao, Han Cao, and Binh Nguyen. "DELAFO: An Efficient Portfolio Optimization Using Deep Neural Networks". In: May 2020, pp. 623–635. ISBN: 978-3-030-47425-6. DOI: 10.1007/978-3-030-47426-3_48.

[5] James Chen. *Technical Indicator Definition*. URL: https://www.investopedia.com/terms/t/technicalindicator.asp.

[6] T. Chen and Carlos Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016).

[7] Wei Chen et al. "Mean–variance portfolio optimization using machine learning-based stock price prediction". In: *Applied Soft Computing* 100 (2021), p. 106943. ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2020.106943. URL: https://www.sciencedirect.com/science/article/pii/S1568494620308814.

[8] Yao-Hsin Chou, Shu-Yu Kuo, and Yi-Tzu Lo. "Portfolio Optimization Based on Funds Standardization and Genetic Algorithm". In: *IEEE Access* 5 (2017), pp. 21885–21900. DOI: 10.1109/ACCESS.2017.2756842.

[9] Yves Choueifaty and Yves Coignard. "Toward Maximum Diversification". In: *Journal of Portfolio Management - J PORTFOLIO MANAGE* 35 (Oct. 2008), pp. 40–51. DOI: 10.3905/JPM.2008.35.1.40.

[10] Peter Christoffersen et al. "Is the Potential for International Diversification Disappearing? A Dynamic Copula Approach". In: *Review of Financial Studies* 25 (May 2012). DOI: 10.2139/ssrn.2066076.

[11] Roger Clarke, Harindra de Silva, and Steven Thorley. "Minimum-Variance Portfolios in the U.S. Equity Market". In: *The Journal of Portfolio Management* 33 (Sept. 2006), pp. 10–24. DOI: 10.3905/jpm.2006.661366.

[12] Tunchan Cura. "Particle swarm optimization approach to portfolio optimization". In: *Nonlinear Analysis: Real World Applications* 10.4 (2009), pp. 2396–2406. ISSN: 1468-1218. DOI: https://doi.org/10.1016/j.nonrwa.2008.04.023. URL: https://www.sciencedirect.com/science/article/pii/S1468121808001259.

[13] Victor Demiguel, Lorenzo Garlappi, and Raman Uppal. "Optimal Versus Naive Diversification: How Inefficient is the 1/N Portfolio Strategy?" In: *Review of Financial Studies* 22 (May 2009). DOI: 10.1093/rfs/hhm075.

[14] Kenan Cem Demirel, Ahmet Şahin, and Erinc Albey. "A Web-Based Decision Support System for Quality Prediction in Manufacturing Using Ensemble of Regressor Chains". In: *Data Management Technologies and Applications*". Ed. by Slimane Hammoudi, Christoph Quix, and Jorge Bernardino. Cham: Springer International Publishing, 2020, pp. 96–114. ISBN: 978-3-030-54595-6.

[15] Shubharthi Dey et al. "Forecasting to Classification: Predicting the direction of stock market price using Xtreme Gradient Boosting". In: *PESIT South Campus* (Oct. 2016). DOI: 10.13140/RG.2.2.15294.48968.

[16] Alexander Didenko and Svetlana Demicheva. "Application of Ensemble learning for views generation in Meucci Portfolio Optimization Framework". In: *Review of Business and Economics Studies* 1 (Jan. 2013).

[17] *FTSE/JSE Top 40 Index Stocks Prices*. URL: https://www.investing.com/indices/ftse-jse-top-40-components.

[18] La Gubu, Dedi Rosadi, and Abdurakhman. "Classical portfolio selection with cluster analysis: Comparison between hierarchical complete linkage and Ward algorithm". In: *AIP Conference Proceedings* 2192.1 (2019), p. 090004. DOI: 10.1063/1.5139174. eprint: https://aip.scitation.org/doi/pdf/10.1063/1.5139174. URL: https://aip.scitation.org/doi/abs/10.1063/1.5139174.

[19] Zhenlong Jiang, Ran Ji, and Kuo-Chu Chang. "A Machine Learning Integrated Portfolio Rebalance Framework with Risk-Aversion Adjustment". In: *Journal of Risk and Financial Management* 13.7 (2020). ISSN: 1911-8074. DOI: 10.3390/jrfm13070155. URL: https://www.mdpi.com/1911-8074/13/7/155.

[20] *K Means Clustering | K Means Clustering Algorithm in Python*. Aug. 2019. URL: https://www.analyticsvidhya.com/blog/2019/08/comprehensive-guide-k-means-clustering/.

[21] Tomasz Kaczmarek and Katarzyna Perez. "Building portfolios based on machine learning predictions". In: *Economic Research-Ekonomska Istraživanja* 0.0 (2021), pp. 1–19. DOI: 10.1080/1331677X.2021.1875865. eprint: https://doi.org/10.1080/1331677X.2021.1875865. URL: https://doi.org/10.1080/1331677X.2021.1875865.

[22] Liu Lijun, Wei-Kang Shen, and Jia-Ming Zhu. "Research on Risk Identification System Based on Random Forest Algorithm-High-Order Moment Model". In: *Complexity* 2021 (Apr. 2021), pp. 1–10. DOI: 10.1155/2021/5588018.

[23] Yilin Ma, Ruizhu Han, and Weizhong Wang. "Prediction-Based Portfolio Optimization Models Using Deep Neural Networks". In: *IEEE Access* 8 (2020), pp. 115393–115405. DOI: 10.1109/ACCESS.2020.3003819.

[24] Luís Lobato Macedo, Pedro Godinho, and Maria João Alves. "Mean-semivariance portfolio optimization with multiobjective evolutionary algorithms and technical analysis rules". In: *Expert Systems with Applications* 79 (2017), pp. 33–43. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2017.02.033. URL: https://www.sciencedirect.com/science/article/pii/S0957417417301252.

[25] Ahmed Marhfor. "Portfolio Performance Measurement: Review of Literature and Avenues of Future Research". In: *American Journal of Industrial and Business Management* 06 (Jan. 2016), pp. 432–438. DOI: 10.4236/ajibm.2016.64039.

[26] Harry Markowitz. "PORTFOLIO SELECTION*". In: *The Journal of Finance* 7.1 (1952), pp. 77–91. DOI: https://doi.org/10.1111/j.1540-6261.1952.tb01525.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6261.1952.tb01525.x. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1540-6261.1952.tb01525.x.

[27] Karina Marvin. "Creating diversified portfolios using cluster analysis". In: *Princeton University* (2015).

[28] Mahdi Massahi, M. Mahootchi, and A. Arshadi Khamseh. "Development of an efficient cluster-based portfolio optimization model under realistic market conditions". In: *Empirical Economics* (2020), pp. 1–20.

[29] Richard Michaud. "The Markowitz Optimization Enigma: Is 'Optimized' Optimal?" In: *Financial Analysts Journal - FINANC ANAL J* 45 (Jan. 1989), pp. 31–42. DOI: 10.2469/faj.v45.n1.31.

[30] Sarah Perrin and Thierry Roncalli. "Machine Learning Optimization Algorithms & Portfolio Allocation". In: *Machine Learning for Asset Management*. John Wiley and Sons, Ltd, 2020. Chap. 8, pp. 261–328. ISBN: 9781119751182. DOI: https://doi.org/10.1002/9781119751182.ch8. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/9781119751182.ch8. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119751182.ch8.

[31] *PORTFOLIO PERFORMANCE EVALUATION (Finance)*. URL: http://what-when-how.com/finance/portfolio-performance-evaluation-finance/ (visited on 05/04/2021).

[32] Edward Qian. "Risk parity portfolios: Efficient portfolios through true diversification". In: *Panagora Asset Management* (2005).

[33] Zhiwei Ren. "Portfolio Construction using Clustering Methods". Thesis. Worcester Polytechnic Institute, 2005.

[34] Obeidat Samer et al. "Adaptive Portfolio Asset Allocation Optimization with Deep Learning". In: *International Journal On Advances in Intelligent Systems* 11.1 and 2 (2018), pp. 25–34. ISSN: 1942-2679.

[35] Philipp Schiele. "Modern Approaches to Dynamic Portfolio Optimization". In: *Junior Management Science* 6.1 (2021), 149–189. DOI: 10.5282/jums/v6i1pp149-189. URL: https://jums.ub.uni-muenchen.de/JMS/article/view/5102.

[36] Prisadarng Skolpadungket, Keshav Dahal, and Napat Harnpornchai. "Portfolio optimization using multi-obj ective genetic algorithms". In: *2007 IEEE Congress on Evolutionary Computation*. 2007, pp. 516–523. DOI: 10.1109/CEC.2007.4424514.

[37] Van-Dai Ta, Chuanming Liu, and Direselign Addis Tadesse. "Portfolio Optimization-Based Stock Prediction Using Long-Short Term Memory Network in Quantitative Trading". In: *Applied Sciences* 10 (2020), p. 437.

[38] Ghali Tadlaoui. "Intelligent Portfolio Construction: Machine-Learning enabled Mean-Variance Optimization". MA thesis. Master's thesis, Imperial College London, 2017.

[39] Jerome Teiletche, Thierry Roncalli, and Sébastien Maillard. "On the Properties of Equally-Weighted Risk Contributions Portfolios". In: *SSRN Electronic Journal* (Sept. 2008). DOI: 10.2139/ssrn.1271972.

[40] S Wang et al. "Non-linear stochastic optimization using genetic algorithm for portfolio selection". In: *International Journal of Operations Research* 3.1 (2006), pp. 16–22.

[41] Xiaolou Yang. "Improving Portfolio Efficiency: A Genetic Algorithm Approach". In: *Computational Economics* 28 (Aug. 2006), pp. 1–14. DOI: 10.1007/s10614-006-9021-y.

[42] Viriya Yimying and Ohm Sornil. "Portfolio Optimization Using Multi-Objective Particle Swarm Optimization". In: ().

[43] Zihao Zhang, Stefan Zohren, and Stephen Roberts. "Deep Learning for Portfolio Optimization". In: *The Journal of Financial Data Science* 2.4 (2020), pp. 8–20. ISSN: 2405-9188. DOI: 10.3905/jfds.2020.1.042. eprint: https://jfds.pm-research.com/content/2/4/8.full.pdf. URL: https://jfds.pm-research.com/content/2/4/8.