USING MACHINE LEARNING
TO DEVELOP AN ASSET ALLOCATION APPLICATION

CİHAT ERBAY

BOĞAZİÇİ UNIVERSITY

2020

USING MACHINE LEARNING

TO DEVELOP AN ASSET ALLOCATION APPLICATION

Thesis submitted to the

Institute for Graduate Studies in Social Sciences

in partial fulfillment of the requirements for the degree of

Master of Arts

in

Economics

by

Cihat Erbay

Boğaziçi University

2020

Using Machine Learning

to Develop an Asset Allocation Application

The thesis of Cihat Erbay

has been approved by:

Prof. Burak Saltoğlu ______________________________
(Thesis Advisor)

Assoc. Prof. Tolga Umut Kuzubaş ______________________________

Assoc. Prof. Ahmet Göncü ______________________________
(External Member)

September 2020

DECLARATION OF ORIGINALITY

I, Cihat Erbay, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature.....C. Erbay.....

Date .....23.09.2020.....

ABSTRACT

Using Machine Learning

to Develop an Asset Allocation Application

This paper aims to analyze factor mimicking portfolio (FMP) construction in the context of Machine Learning and use it in a practical application. Inflation is chosen as the target macroeconomic factor in this study. We show that in the FMP construction process, Linear regression model performs better than LASSO and Ridge regression. In order to provide an application with FMP, Random Forest model is utilized in forecasting future inflation values so that asset allocation in the FMP can be done accordingly. While forecast accuracy of the Random Forest model is better than 1-period lag and simple moving average models, Rolling ARIMA model has smaller forecast error. However, FMP which is updated by the inflation forecasts from Random Forest provide more profit, higher Sharpe and Sortino ratios, and smaller maximum drawdown than the one with the Rolling ARIMA. Lastly, we follow a strategy such that we invest 95% of the initial capital to equally weighted portfolio and 5% of the initial capital to the FMP. We show that the proposed model outperforms the market where equally weighted portfolio of the same assets is used as market benchmark.

ÖZET

Varlık Tahsisi Uygulaması Geliştirmek için Makine Öğrenmesi Kullanımı

Bu çalışma sırasında, faktörü takip eden portföy (factor mimicking portfolio (FMP)) oluşturulması sırasında makine öğrenmesinden yararlanılmış ve oluşturulan bu portföy bir uygulama içerisinde kullanılmıştır. Çalışmamızda hedef makro ekonomik faktör enflasyon olarak belirlendi. FMP oluşturulması sürecinde doğrusal regresyon modeli LASSO ve Ridge regresyon modellerinden daha iyi performans göstermiştir. Portföydeki hisselerin yönetimini gerçekleştirmek için ve bu portföyü uygulamalı olarak kullanmak için, gelecek enflasyon değerleri Random Forest makine öğrenmesi yöntemi kullanılarak tahmin edildi. Random Forest modelinin tahmin doğruluğu 1-dönem gecikme (1-period lag) ve basit hareketli ortalama (Simple Moving Average) yöntemlerinden daha iyi olmakla birlikte, Kaydırmalı ARIMA modeli daha az tahmin hatasına sahip. Buna rağmen, Random Forest yönteminin tahmin ettiği enflasyon değerleri ile yönetilen portföy, Kaydırmalı ARIMA'nın tahminleriyle yönetilen portföyden daha fazla getiri getirmekle birlikte hem daha yüksek Sharpe ve Sortino rasyolarına sahip hem de daha düşük en yüksek düşme noktasına sahip. Son olarak, bir yatırım stratejisi takip ederek sermayenin %95'i eşit ağırlıklı portföye ve %5'i oluşturduğumuz portföye yatırıldı. Piyasa karşılaştırmalı değerlendirme noktası olarak eşit ağırlıklı portföy seçildi ve önerilen portföyün piyasadan daha iyi performans verdiği gösterildi.

ACKNOWLEDGMENTS

First of all, I would like to express my deepest thanks and sincere gratitude to my thesis advisor Prof. Burak Saltoğlu for giving me the opportunity to do research and his kind supervision. It was a great honor to work under his guidance.

Besides my advisor, I would like to express my special thanks and appreciation to Dr. Ayhan Yüksel for his endless support and generous advice during the study. I would like to thank my committee members, Assoc. Prof. Tolga Umut Kuzubaş and Assoc. Prof. Ahmet Göncü who kindly accepted to participate my thesis defense and gave me insightful comments.

Also, I would like to thank the Scientific and Technological Research Council of Turkey (TÜBİTAK) BİDEB who supported me financially during the 2210-A Master’s Scholarship Program.

Last but not least, I would like to express my eternal appreciation to my parents and my wife, Elif Erbay, who have always been there for me no matter where I am, for all their supports and patience.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

One of the driving methodologies in the asset management framework is factor investing. Investors generally examine macro factors closely; however, these factors are not investable without an intermediary. Non-tradability of macro variables could be managed by factor mimicking portfolio (FMP) methodology. In this study we elaborate FMP construction in the context of Machine Learning and use it in a practical application. In order to make an application with FMP, future values of the target macroeconomic factor need to be predicted so that asset allocation is done accordingly. We use inflation as the target variable and Machine Learning is utilized in forecasting future inflation values.

Our study is threefold. Firstly, we use several Machine Learning methods such as Linear Regression, LASSO and Ridge Regression to construct factor mimicking portfolios (FMPs) where our target macroeconomic factor is inflation. We use historical data covering the time span from February 1991 to June 2019 to build representative FMP that consists of equities, government bonds and commodities. Then, we compare the performance of these factor mimicking methodologies to find the best model among them, which is Linear model in the specified context.

Secondly, we use one of the most popular Machine Learning methods, Random Forest, to forecast future values of inflation by using 25 distinct factors(overall 44 factors together with lagged variables) from the following 8 classes as predictors: output and income, labor market, housing, consumption, orders and inventories, money and credit, interest and exchange rates, prices, and stock market. Random Forest model overperformed the standard benchmarks like 1-period lag and Simple Moving Average methods while the Rolling ARIMA model has more predictive power by having a smaller forecast error than Random Forest.

Lastly, we combine the outputs of the two stages. Our proposed model relies on updating the asset positions in the FMP (generated in the first stage) according to the inflation forecasts made by Random Forest model in the second stage. When we compare it with Rolling ARIMA, the results show that we generate more profit and higher Sharpe and Sortino ratios with the FMP which is updated using Random Forest forecasts. These findings imply that Random Forest model has ability to add more economic value than Rolling ARIMA although the Random Forest model has worse statistical performance than the Rolling ARIMA model by providing lower sum of squared residuals.

The remaining part of this study is organized as the following: Chapter 2 includes literature review about FMPs, inflation forecasting and Machine Learning usage in these contexts. Chapter 3 consists of data used in the construction of FMP and also macro variables used as predictors to forecast the inflation. This chapter also explain the methodology of well-known Machine Learning models used in this study. Chapter 4 explains the construction of proposed portfolio together with the initial FMP and Machine Learning models to predict inflation. Chapter 5 discusses the FMP performances, accuracy of inflation forecasts and performance of proposed portfolio in the specified context. Finally, Chapter 6 contains the conclusion.

CHAPTER 2

LITERATURE REVIEW

There is an insight that asset pricing models can track future economic variables much more accurately than current variables. Investors generally examine macro factors closely. However, these factors are not investable without an intermediary. Non-tradability of macro variables could be managed by a new approach: factor mimicking portfolio (FMP). Lamont (2001) introduced economic tracking portfolio to track macro factors such as inflation, production, consumption and so on. Lamont mostly utilized the monthly stock and bond returns to forecast these factors and found that we can get more accurate forecast and hedge risk by using economic tracking portfolios. Jurczenko and Teiletche (2019) proposed a framework to improve the construction of mimicking portfolios using machine learning models such as Principal Component Analysis (PCA) and Least Absolute Shrinkage and Selection Operator (LASSO). While the method proposed by Jurczenko and Teiletche reduces the average returns in the course of time, it gives returns with the lower maximum drawdown, lower volatility and higher Calmar and Sharpe Ratio.

There is a huge literature on factor forecasting and inflation is in great demand among those factors. Stock and Watson (1999) used Philips curve models to forecast the US inflation. They found that forecast errors from unemployment-based Philips curve models can be greater than Philips curve models using real activity and production related measures as predictors. In 2001, Atkeson and Ohanian evaluated the performance of Philips curve models and compared it with naïve models. They concluded that even naïve models like random walk are able to outperform the Philips curve models.

After those prior works, there was a motivation in the academic world to improve the inflation forecasts by using alternative models and different type of variables. Chow (2014) investigated the importance of financial variables in inflation forecasting and used an index that composes of the financial variables like stock prices, credit expansions, exchange rates etc. The model proposed by the Chow, found smaller forecast errors with the help of financial variables than simple autoregressive models for inflation. Altug and Cakmakli (2015) developed a statistical model with the help of survey expectations and some monetary policies. They got better forecast results against Philips curve and naïve models.

As we reach more advance technology, researchers have started to take advantage of the computer power to process big data on forecasting problems. Kucukefe (2018) used several machine learning models to predict Turkey's consumer price index. Different kind of machine learning frameworks such as Linear and Ridge regression, Random Forest etc. were used in that study and survey expectations data is used as an input to the models. The study showed that proposed models are superior to the univariate models and also the findings are more accurate than the inflation forecasts by the monetary authority. Hurtado et al. (2013) investigated the inflation of Mexico using another machine learning method, Neural Network, and resulted in better forecast performance than the predictions of the monetary authority. Zhang and Li (2012) tried to forecast inflation using Support Vector Machines and evaluated the model performance by looking at the root mean squared error and mean absolute errors. Results showed that Support Vector Regression model is superior than back propagation and linear model.

CHAPTER 3

DATA AND METHODOLOGY

3.1 Data

3.1.1 Asset universe

In this study, we use US Consumer Price Index (CPI) as inflation measure in the US. In order to track the inflation, it is required to build a representative asset universe in which the base assets are available in the target time period and they also need to be investable. Appendix A shows the selected assets. All these financial variables are obtained from Thomson Reuters-DataStream. They mainly cover three asset classes which are equities, government bonds and commodities.

In order to decide the time span of the sample, I found the common time period of available data for these assets. Since our inflation measure, CPI, is a monthly collected data, daily data of base assets coming from DataStream is converted to monthly data. To eliminate possible trends in the timeseries, monthly change of the variables calculated using discrete return function. Resulting dataset covers the time span from February 1991 to June 2019, which corresponds to 341 observations.

3.1.2 Macroeconomic factors

All macroeconomic factors used in this study to forecast inflation are obtained from the FRED-MD database developed by Michael W. McCracken and Serena Ng who worked with the data desk at Federal Reserve Bank of St. Louis. It consists of monthly collected data and we are using 25 different factors from the following 8 classes: output and income, labor market, housing, consumption, orders and inventories, money and credit, interest and exchange rates, prices, and stock market. Appendix B shows the macroeconomic factors used in this part of the study. As asset universe mentioned in the Section 3.1.1, this dataset also covers the time span from February 1991 to June 2019, which corresponds to 341 observations.

3.2 Models

3.2.1 Benchmark models for forecasting

3.2.1.1 Simple moving average

Simple Moving Average (SMA) can be defined as the unweighted mean of a specific number of recent observations in a timeseries data. Simple moving average can be any order. SMA of order n is defined as:

$$SMA = \frac{p_N + p_{N-1} + p_{N-2} + \cdots + p_{N-(n-1)}}{\text{n}}$$

$$= \frac{1}{n}\sum_{j=0}^{n-1} p_{N-j}$$

where p_N is the observation at period N and n is the number of periods.

3.2.1.2 Autoregressive integrated moving average (ARIMA)

ARIMA models consist of three main components, which are autoregressive component, moving average component and integration component. The autoregressive (AR) component implies that each observation of the timeseries can be defined as the linear combination of previous observations. Autoregressive model of order p can be formulated as following:

$$y(t) = \sum_{j=1}^{p} a_j y(t-j)$$

where $y(t)$ is the observation at period t and a_j's are coefficients.

Moving average (MA) component implies that observations of the timeseries depend on the previous forecasting errors. Moving average model of order q can be defined as:

$$y(t) = \sum_{j=1}^{q} b_j \varepsilon(t-j)$$

where $y(t)$ is the observation at period t, ε is the estimation error and b_j's are coefficients.

If we combine AR and MA models, ARMA model is achieved. We can generalize the ARMA model of order p and q, referred as ARMA(p,q), using following formula:

$$y(t) = \sum_{j=1}^{p} a_j y(t-j) - \sum_{j=1}^{q} b_j \varepsilon(t-j)$$

where $y(t)$ is the observation at period t, ε is the estimation error and a_j and b_j are coefficients.

When we apply additional differencing to the timeseries and integrate it, then ARIMA model is obtained. ARIMA are especially used to filter simple trends in timeseries. The parameter d denotes the degree of differentiation. ARIMA model of order p, d and q, referred as ARIMA(p,d,q), can be generalized as follows:

$$y'(t) = c + \alpha_1 y'(t-1) + \cdots + \alpha_p y'(t-p) + \beta_1 \varepsilon(t-1) + \cdots + \beta_q \varepsilon(t-q) + \varepsilon(t)$$

where $y'(t)$ is the differenced series, ε is the estimation error and α_j and β_j are coefficients. To find the optimal values for p, d and q, the models are tested and ranked with respect to Akaike's Information Criterion (AIC) or Bayesian Information Criterion (BIC). Generally, the model which has the lowest AIC or BIC values is preferred as the best model.

3.2.2 Linear regression

Linear regression models are simply used to find a linear relationship between two group of variables, which are dependent and independent variables. There can be one or more independent variables. If it is one, it is called simple linear regression and if more than one it is called multiple linear regression. Coefficients in the model represents how much an independent variable related to the dependent variable. Additionally, there are two other term in the linear regression equation, which are intercept constant and error term. Linear regression model can be formulated as:

$$y_j = \alpha_0 + \sum_{k=1}^{m} \beta_j x_{jk} + \varepsilon_j$$

where y_j's are observations of dependent variable, x_{jk}'s are independent variables, α_0 is the intercept term and ε_j is the error term for each observation.

3.2.3 Least absolute shrinkage and selection operator (LASSO)

One of the different types of linear regression which uses shrinkage is LASSO. Shrinkage term means to shrink the variables towards a point such as mean or zero. Applying shrinkage leads to reduce the magnitude of the weights and it is possible to provide even sparse models. LASSO utilizes L1 regularization by adding absolute value of the coefficients into the model as penalty term. Using L1 regularization may end up with sparse models that have fewer coefficients since magnitudes of some weights/coefficients can be reduced to zero and removed from the equation. Main purpose of the LASSO algorithm is to minimize the following cost function:

$$\sum_{j=1}^{N}(y_j - \hat{y}_j)^2 = \left(\sum_{j=1}^{N}(y_j - \sum_{i=0}^{p} w_i \times x_{ji})^2\right) + \lambda \sum_{i=0}^{p}|w_i|$$

in which the first term is the same as the cost function for linear regression and additionally there is a penalty term, λ, that regularizes the model by decreasing the coefficients as much as absolute value of magnitudes of the weights.

3.2.4 Ridge regression

Ridge regression is another type of regression model that uses shrinkage. It is very similar to LASSO except for the regularization type. Ridge model uses L2 regularization that apply a penalty term in which magnitudes of the coefficients are squared. Because of using squared values in the penalty term, Ridge model does not end up with the removal of coefficients, instead it just penalizes the model by reducing the magnitude of coefficients. Ridge cost function can be formulated as:

$$\sum_{j=1}^{N}(y_j - \hat{y}_j)^2 = \left(\sum_{j=1}^{N}(y_j - \sum_{i=0}^{p} w_i \times x_{ji})^2\right) + \lambda \sum_{i=0}^{p}{w_i}^2$$

where the first term is the same as the cost function for linear regression and additionally there is a penalty term, λ, that regularizes the model by reducing the coefficients as much as squared value of magnitudes of the weights.

3.2.5 Random Forest model

Random Forest (RF), firstly proposed by Breiman (2001), is some type of Bagging that constructs a collection of so-called decision trees and takes average of them. The main component of the RF is binary decision tree. It consists of 'root' (decision) and 'leaf' nodes. This type of nonlinear model works by searching for correct class and ignoring the others until finding the correct one. The decision tree is searched sequentially and decision is made at every node and data is split into, let's say X_1 and X_2, by considering the condition such as $x_j \leq \delta$ or $x_j > \delta$ respectively where x_j is an individual feature and δ is a threshold value. This separation process is done according to some quality function. It is so-called information gain and can be formulated as the following:

$$\mathrm{IG}(\mathrm{X}, \mathrm{j}, \delta) = IM(X) - \frac{n_{X_1}}{n_X} IM(X_1) - \frac{n_{X_2}}{n_X} IM(X_2)$$

where IM is the impurity measure, δ is threshold value, X, X_1 and X_2 are datasets of root and child nodes, n_X, n_{X_1} and n_{X_2} are the number of observations at root and child nodes respectively. As the number of child nodes increase, impurity of new nodes should also be subtracted from the impurity of root node in a similar manner.

Actual aim of this algorithm is to minimize the variance of the samples in the child nodes and help us to create non-linear dependencies. Nevertheless, decision trees have a significant disadvantage in terms of overfitting. It means that the model may memorize all the training dataset such that it does not make any mistake in the training set in any circumstances. Random Forest model is specially designed to solve this problem. RF has randomization feature in two aspects: First, specified number of virtual subsamples are generated from the original sample. Second, a virtual subsample does not cover all the feature set, but a randomly specified subset. Then a decision tree is constructed for each generated virtual tree. The end result of the algorithm is calculated by averaging the outputs of all decision trees. A critical hyperparameter of RF is number of decision trees. Generally, increasing number of decision trees results in more reliable outputs but the operation time also increases substantially.

CHAPTER 4

MODEL CONSTRUCTION

4.1 Construction of factor mimicking portfolio

As mentioned in the section 3.1.1, firstly the selected macro variables which are shown in Appendix A are extracted from the Thomson Reuters-DataStream. After specifying a common time span on available data and converting the data from daily to monthly, it is left to put the data into the model. In order to construct factor mimicking portfolio (FMP), first of all it is required to find out the relationships of the base assets with our target variable which is inflation. In order to measure those relationships, regression models are utilized. I used three different regression models, which are namely Linear regression, LASSO and Ridge regression, to construct three separate mimicking portfolios.

Instead of in-sample analysis, rolling out-of-sample analysis is used in the models. In the in-sample testing, model uses all available data to make estimation and compare it with the actual data. However, it is very possible for in-sample model to encounter with the risk of overfitting. So, it is better to focus on specific period of time and update this time span in a rolling manner. In this study, I used 120 months-long fixed rolling windows to train the model. By using the model trained with the recent observations, we try to include possible changes in the data structure over time.

After constructing the main cycle of the rolling window mechanism, last 120 observations are used to train Linear regression, LASSO and Ridge regression models. Repeated cross-validation with 10 folds and 2 repeats is applied to both LASSO and Ridge model. Different tuning grids for regularization are created for each model. In each loop of the rolling window analysis, optimal penalty terms are chosen for final models and resulting coefficients of the models give the relationships between inflation and corresponding base asset. The coefficients are normalized to be converted into the portfolio weights. I also apply volatility targeting of 10% level to the resulting asset weights. All the portfolio weights in the given time span are calculated and used to construct the mimicking portfolios. This is the first part of our study and in the second part we build a model to forecast the future inflation values.

4.2 Macroeconomic factor forecasting

The data will be used in this part of the study is described in the section 3.1.2. We obtain 25 macroeconomic timeseries that cover 8 different classes in the economy. After extracting the subset of the data in the specified time period, a stationarity test, Augmented Dickey-Fuller test, is applied to the timeseries. I used first differencing on the nonstationary variables and calculated percent changes. In addition to these timeseries, some lags of selected variables such as CPI, RPI, UNRATE, FEDFUNDS, TWEXMMTH and HOUST are calculated and put into the model as predictors. 1, 2, 3, 4, 6 and 12 months-lagged version of CPI are used in the model. Resulting model consists of 44 predictor variables in total with these lagged timeseries included.

In the inflation forecasting study, I used rolling out-of-sample analysis as in the previous section. With the rolling-window method, last 120 observations are used to train the Random Forest model in each loop and the model makes predictions about the inflation measure, CPI, in the next month. Repeated cross validation with 15 folds and 3 repeats is applied to the model. Since it takes so much time to train the Random Forest model, I take advantage of parallel processing feature using *doParallel* library in the RStudio.

In the machine learning literature, hyperparameter tuning always be an important topic that affects performance and accuracy of the models. In the Random Forest model, we have specifically 2 main hyperparameter to tune. One of them is number of trees grown in the forest, called *ntree*. Generally, the larger number of trees in the model, the more stable and accurate the model is. However, it also costs more memory and CPU power together with the increasing amount of run time. After couple of trials I choose 400 as the number of trees grown in the model. Second main hyperparameter is number of features available to split at each node, which is called *mtry*. Random Forest models use random subspace method, that means rather than choosing all the features available it picks a random subsample of them. *mtry* parameter specifies the size of that random subsample at each node. I define a tune grid for *mtry* that covers the values 35, 38, 41 and 44. Throughout each loop, optimal *mtry* value is calculated and used to construct the best model among others.

In the last part of the study, I created the final portfolios to make investment. We have got the inflation predictions from the Random Forest model on one hand and three factor mimicking portfolios constructed by Linear regression, LASSO and Ridge regression models on the other hand. We updated the mimicking portfolio weights based on the inflation forecast for the next month. The strategy is simply the following: if the CPI forecast is greater than zero, that means if we expect increasing amount of inflation in the next month, then we take long position on assets with the positive weights and short position on assets with the negative weights. On the other hand, if we expect the change of inflation to be smaller in the next month, then we take long position on assets in which short position is taken in the previous case and take short position on assets in which long position is taken in the previous case. So, our constructed portfolio is adjusted consistently, and investments are made based on the forecasts for the inflation in the economy.

CHAPTER 5

RESULTS

5.1 Factor mimicking portfolio evaluation

We used three different regression model to construct three distinct factor mimicking portfolios. Firstly, base assets are regressed on the CPI and coefficients of the model is calculated accordingly. Coefficients are normalized and volatility targeting to 10% level is applied to them in order to get asset weights in the FMP. This process is implemented through each cycle of the rolling window analysis. Resulting cumulative FMP returns for Linear model, LASSO and Ridge model end up with negative returns. However, the important point to notice in these portfolios are the short-term cycles of the returns. Since these are factor mimicking portfolios, we expect to get portfolio returns that may or may not provide positive returns, but instead they track the current position of the inflation in the economy. In order to show this, cumulative portfolio returns are filtered using high-pass filter and trends in the timeseries are eliminated. High-pass filter is also applied to the CPI and relationships of the factor and its mimicking portfolios are shown in the Figure 1, Figure 2, and Figure 3.

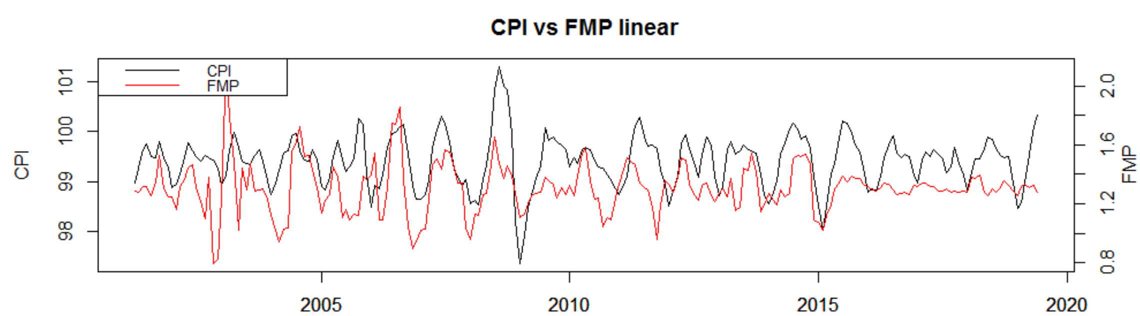

Figure 1. Timeseries data for CPI and FMP constructed by Linear model after HP filter

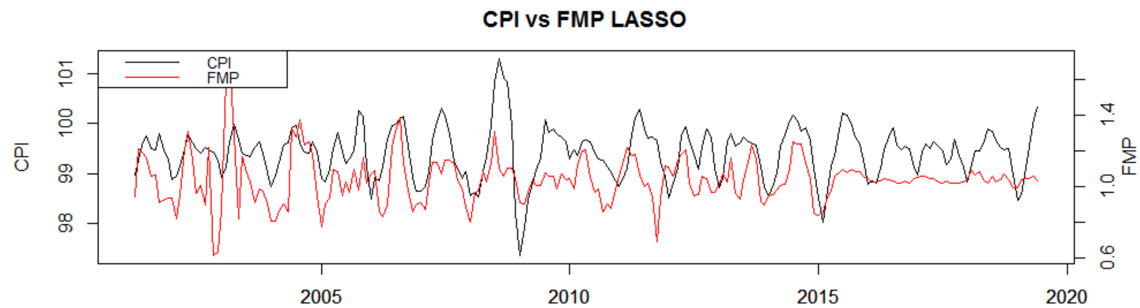

Figure 2. Timeseries data for CPI and FMP constructed by LASSO model after HP filter

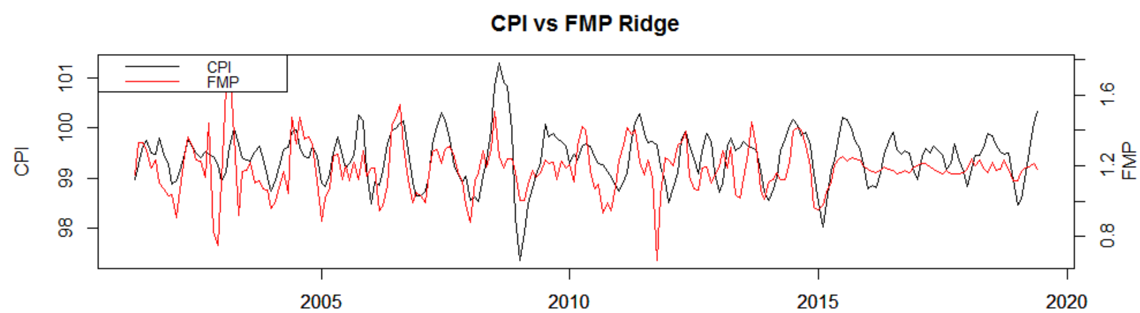

Figure 3. Timeseries data for CPI and FMP constructed by Ridge model after HP filter

In order to compare the tracking performance of the portfolios, we look at the out of sample correlations between high pass filtered portfolio returns and CPI. As it is seen on Table 1 factor mimicking portfolio constructed by Linear model is the most correlated one with CPI. So, it tracks the factor better than other regression models. As correlation of FMP by Ridge model is very close to linear model, LASSO model provides the least correlated portfolio with CPI.

Table 1. Out-of-sample Correlations

Portfolio Type	Correlation
Linear Model	0.438
LASSO	0.403
Ridge Model	0.434

Before evaluating the performance of constructed factor mimicking portfolios using forecasted values of the inflation in the section 5.2, firstly we look at the crystal-ball performance of the portfolios. To analyze the portfolios with perfect future insight about inflation in the economy, actual values of CPI is provided to the model and depending on the direction of change of the target variable we take long or short position on the assets in the portfolio. Resulting crystal-ball performance of the factor mimicking portfolios constructed by Linear regression, LASSO and Ridge regression are shown in the Figure 4. As it is seen on the graph, the most profitable portfolio is created using Linear model as it generates revenue which is 62 times the initial capital. Portfolios constructed by LASSO and Ridge model generates far less than Linear model and give about 40 times the initial capital. Those results show that proposed factor mimicking portfolios, overall, are able to track the target factor successfully and the best factor insights are produced by the Linear model.

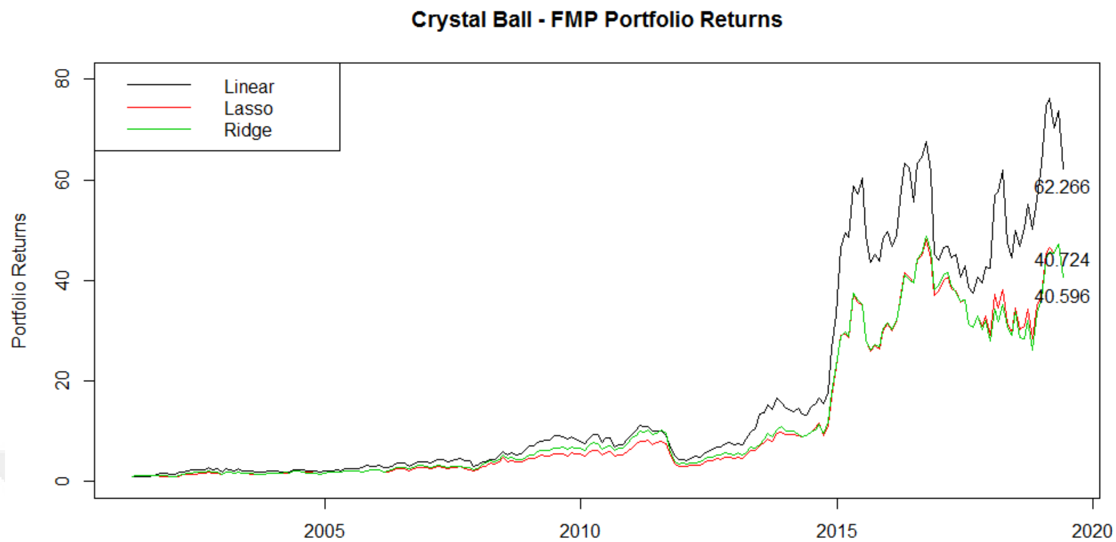

Figure 4. FMP performances with perfect future insight about inflation

5.2 Forecast evaluation and final portfolio returns

In the second part of the study, we try to construct a Random Forest model to forecast inflation using 44 predictor variables in total. Rolling out-of-sample analysis is applied to the model and using recent 120 observations one period forward forecast for monthly change of inflation is estimated. Figure 5 shows the CPI forecasts and actual values on the same graph. We can see that forecast performance is not bad in general. Magnitudes of monthly percentage change of CPI is not forecasted accurately, but the direction estimation of CPI seems acceptable, which is more important for our study because we update our asset positions in the portfolio accordingly.

In order to compare the accuracy of the proposed model, several benchmark models are defined. To see comparable results, we define a very simple model which makes forecast by just copying the 1 period lag of target variable. Second model used as a benchmark to proposed model is Simple Moving Average (SMA) model. To construct this model, we take 12 months simple moving average of target variable, which basically smooth out the timeseries as making estimation by averaging over last 12 observations. In the Figure 6, we compare the forecasts of the proposed model with a simple model which takes 1 period lagged values of monthly change of CPI. Figure 7 shows the comparison of the proposed model with 12 months simple moving average of actual values. We also applied a more complicated benchmark model to the dataset. ARIMA is defined using actual values for percentage change of CPI. In order to specify orders of ARIMA model, we use a built-in function of RStudio in the *forecast* library, called *auto.arima*. We apply ARIMA in a rolling out-of-sample structure, throughout each cycle p, d and q parameters are updated by using *auto.arima* function. Figure 8 shows forecasts by Rolling ARIMA model together with the actual values and Random Forest forecasts for CPI.

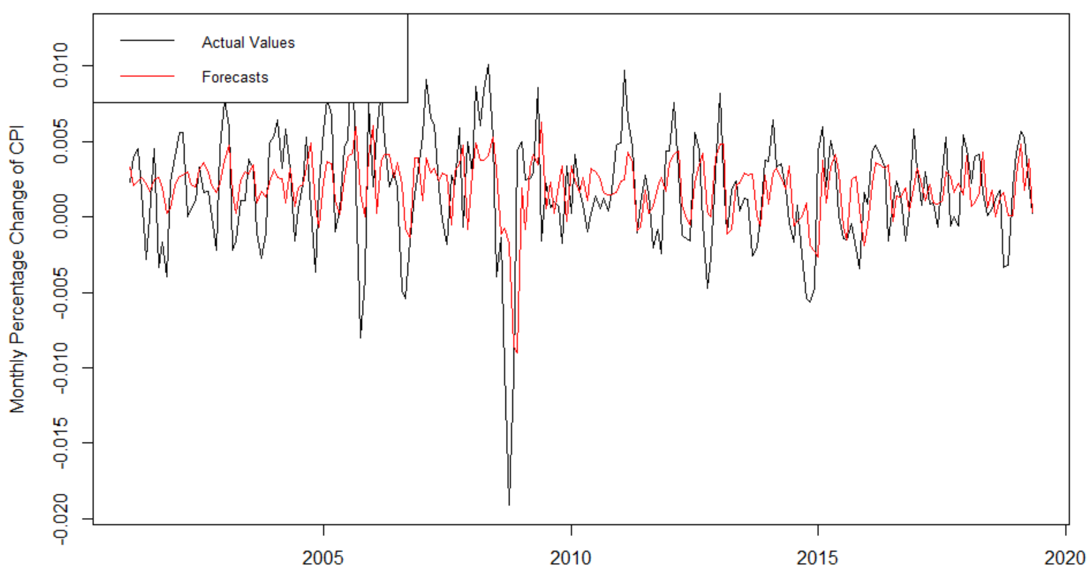

Figure 5. Actual values and RF forecasts for monthly percentage changes of CPI

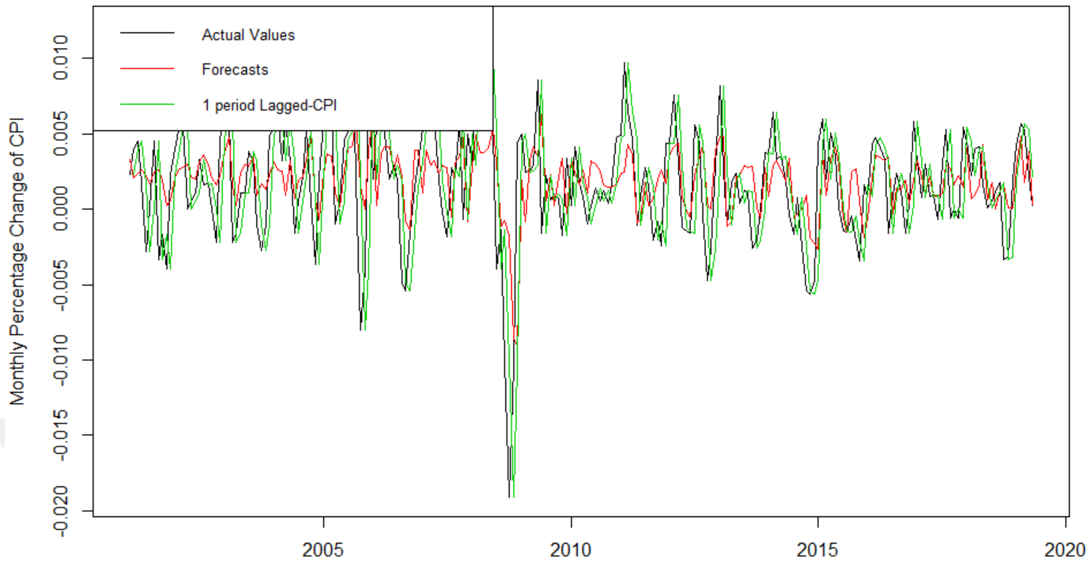

Figure 6. Timeseries data of a simple model (1 period lag of CPI), actual values and Random Forest forecasts for monthly percentage changes of CPI

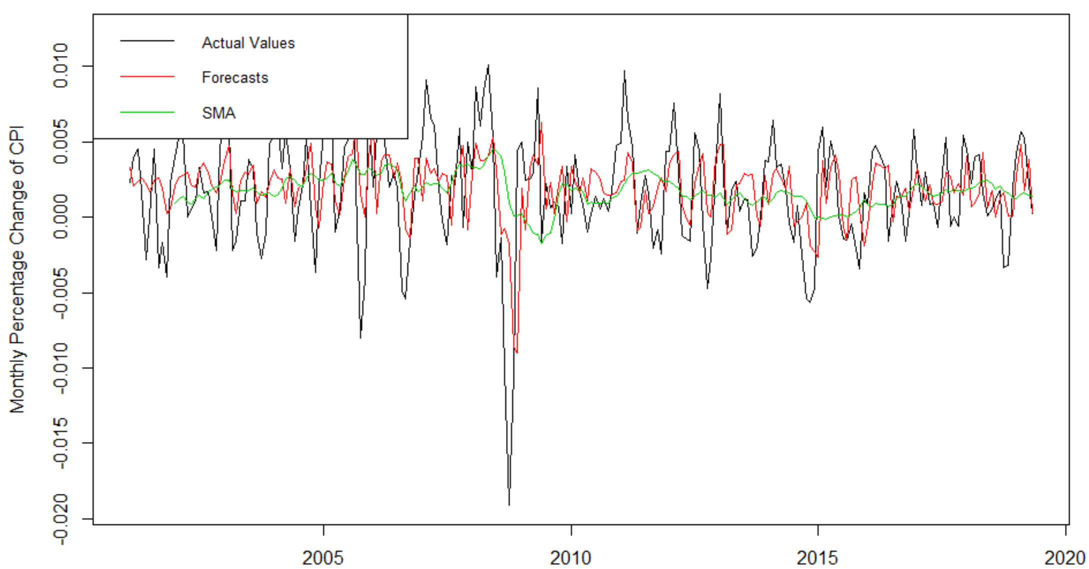

Figure 7. Timeseries data of 12-Months Simple Moving Average, actual values, and Random Forest forecasts for monthly percentage changes of CPI

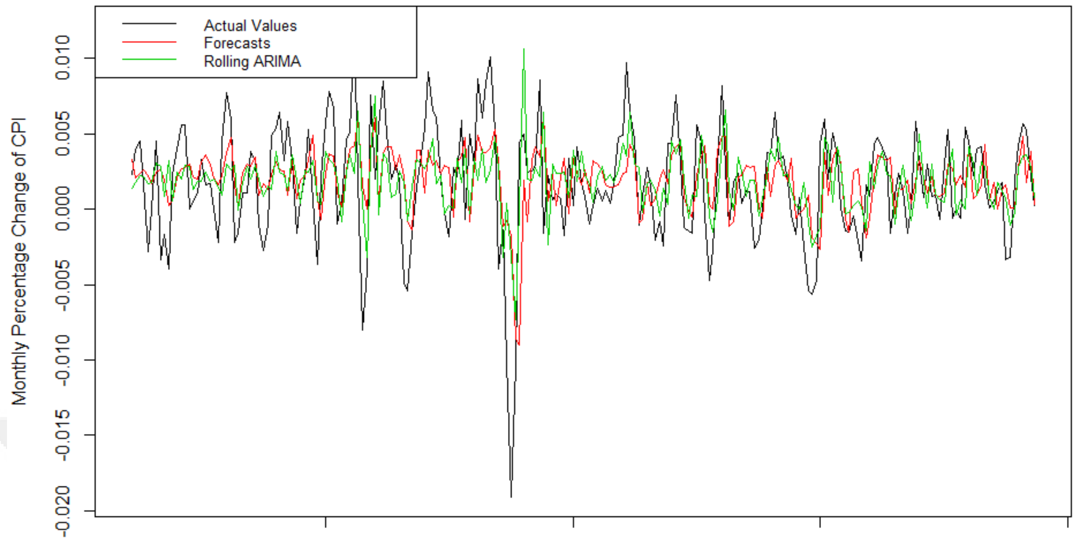

Figure 8. Timeseries data of Rolling Window ARIMA forecasts, actual values, and Random Forest forecasts for monthly percentage changes of CPI

Table 2. Sum of Squared Residuals for Each Forecasting Model

Model	SSR
1 period Lag	0.00319
SMA-12	0.00298
Rolling ARIMA	0.00247
Random Forest	0.00254

As it is seen on Table 2, values for sum of squared residuals show us the forecast accuracy of the benchmark models. We see that ARIMA which applied together with rolling window analysis has the minimum estimation error and Random Forest is very close to it as it is the second most accurate model here. On the other hand, Random

Forest model outperforms both simple models which are 1 period Lag and 12 months Simple Moving Average.

After estimating the future values of inflation, the main task is to update the long-short positions of the assets in the portfolios accordingly. Previously constructed factor mimicking portfolios are reconsidered with the information about future inflation levels and asset positions are updated based on the CPI forecasts generated by the proposed Random Forest model. Resulting portfolio performances are shown in the Figure 9. We can clearly see that FMP constructed by Linear Model is the most profitable portfolio as it gives 2.8 times the initial capital. On the other hand, LASSO and Ridge models seem less profitable than Linear model. While the portfolio with LASSO gives 1.6 times the initial capital, Ridge model does not bring almost any profit at all.

Performance of proposed model can also be compared with the benchmark models. Rolling ARIMA is chosen to make reasonable comparison with the proposed Random Forest model as it has the least forecast error and the most complex one among others. So, constructed factor mimicking portfolios are taken into consideration again and asset positions are updated according to factor forecasts from Rolling ARIMA model. Figure 10 shows the resulting portfolio performances. As we see on the graph, although Rolling ARIMA model has the minimum forecast error (SSR), its portfolio performance is not as good as proposed model. While Linear model brings revenue about 1.3 times initial capital, it is less than half of the profit of proposed Random Forest model. Also, LASSO and Ridge model could not make even positive returns when we apply forecasts from Rolling ARIMA model.

Figure 9. FMP returns with Random Forest forecasts

Figure 10. FMP returns with forecasts from Rolling ARIMA model

We also investigate the other performance metrics of proposed portfolio and one with the Rolling ARIMA model. Details of the comparison are shown on the Table 3. It is very clear that Random Forest model outperforms Rolling ARIMA model in all three metrics. Proposed portfolio has Sharpe ratio of 0.0877, whereas in the benchmark model it is 0.053. Similarly, Sortino ratio is 0.1545 in the proposed model and 0.1072 in the benchmark model. When we look at the maximum drawdowns of the portfolios, proposed model again outperforms the other model as it has lower maximum drawdown. These findings show that Random Forest model has ability to add more economic value than Rolling ARIMA although the Random Forest model has worse statistical performance than the Rolling ARIMA model by providing lower sum of squared residuals. This result is completely consistent with the study conducted by Cenesizoglu and Timmermann (2012) as they evaluate statistical performance and economic significance of return forecasts and found that while they are positively correlated with each other, the relation seems to be weak and only small fraction of the model's economic value is explained by this.

Table 3. Performance Metrics for FMPs Constructed by Forecasts of Rolling ARIMA and Random Forest Models

FMP Type	Rolling ARIMA	Random Forest
Sharpe Ratio	0.0530	0.0877
Sortino Ratio	0.1072	0.1545
Max. Drawdown	0.8527	0.7262

As factor mimicking portfolio with Linear Model gives the best performance using the Random Forest forecasts, we will deeply analyze Linear Model FMP in a more realistic scenario such that we follow a specific strategy to construct the desired portfolio and compare it with equally weighted portfolio. This strategy implies that we invest 95% of the capital to equally weighted portfolio and 5% of the capital to the portfolio created by Linear Model in line with Random Forest forecasts. Return performances of the proposed portfolio and equally weighted portfolio in the test period are shown on the Figure 11. Portfolio performance metrics are shown on the Table 4 and Table 5.

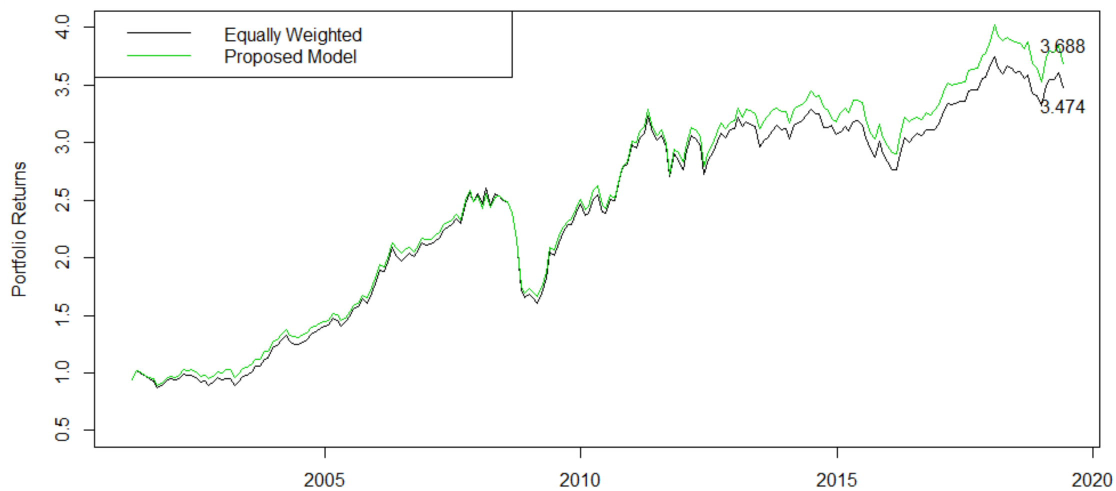

Figure 11. Returns for equally weighted portfolio and proposed model

As it is seen on the Table 4, proposed portfolio has higher Sharpe ratio than equally weighed portfolio. Sharpe ratio is a popular method to measure risk adjusted return. Higher Sharpe ratio means to have higher returns without additional risk and also implies higher level of diversification. On the other hand, Sortino ratio is a variation of Sharpe ratio in a way that it only considers downside deviation whereas Sharpe ratio focuses on both upside and downside risks. We see that proposed model has higher

Sortino ratio such that it gives more return to investment for a specified level of negative risk. Last metric in the Table 4 is maximum drawdown and it is a measure of the largest loss over the specified period of time. We calculated lower maximum drawdown for proposed model.

Table 4. Sharpe Ratio, Sortino Ratio and Max. Drawdown for Equally Weighted Portfolio and Proposed Model

FMP Type	Equally Weighted Portfolio	Proposed Portfolio
Sharpe Ratio	0.1399	0.1503
Sortino Ratio	0.2649	0.2857
Max. Drawdown	0.3851	0.3586

Table 5. Performance Metrics of Proposed Portfolio where Equally Weighted Portfolio is Used as a Baseline Market Portfolio

	Proposed Portfolio
Alpha	0.0005 (p-val: 0.216)
Beta	0.9626*** (p-val: 2e-16)
Tracking Error	0.0200
Information Ratio	0.1443

In order to calculate the portfolio metrics in the Table 5, it is required to construct a benchmark that reflects general moves in the market. Thus, we use equally weighted portfolio as market portfolio to provide a baseline for our analysis. When we look at the results, proposed portfolio gets alpha of 0.0005 which is positive but not significant and beta of 0.9626 which is significant at 1% level. On the other hand, the portfolio has tracking error of 0.02. It shows the divergence between the proposed portfolio and benchmark model. Our finding seems normal as the FMP is only invested with the 5% of the initial capital. Information ratio implies the additional amount of return per extra one unit of risk for the benchmark. Our proposed portfolio has information ratio of 0.1443 which can be considered as a low value, but it is reasonable in the context of investing on a portfolio based on inflation forecast.

CHAPTER 6

CONCLUSION

In this study, we showed that it is possible to create a profitable portfolio by using Machine Learning methods in different aspects. Firstly, we used some regression models on inflation to develop a factor mimicking portfolio (FMP) and the results showed that linear model works best in the specified context. We expected to get this result since linear models are said to be ideal in the case that observation number is much larger than the predictors as in our case. After constructing the FMP, it is left to predict the future values of inflation. Therefore, we applied Random Forest model to forecast inflation. Random Forest model overperformed the standard benchmarks like 1-period lag and Simple Moving Average methods while the Rolling ARIMA model gave more comparable result by having a smaller forecast error than Random Forest model.

In the next part of the study, we make investment on FMP, which is constructed by linear model in the first phase, according to the forecasts made by Random Forest. Resulting portfolio is compared with the one constructed by the forecasts from Rolling ARIMA. The results show that we generate more profit and higher Sharpe and Sortino ratios with the FMP which is updated using Random Forest forecasts. These findings imply that Random Forest model has ability to add more economic value than Rolling ARIMA although the Random Forest model has worse statistical performance than the Rolling ARIMA model by providing lower sum of squared residuals. This result is completely consistent with the study conducted by Cenesizoglu and Timmermann (2012) as they evaluate statistical performance and economic significance of return forecasts and found that while they are positively correlated with each other, the relation seems to be weak and only small fraction of the model's economic value is explained by this.

In addition to this comparable analysis of the models, we followed a specific strategy in a more realistic scenario to construct a portfolio such that we invest 95% of the initial capital to equally weighted portfolio and 5% of the initial capital to FMP generated previously in this study. We proved that the proposed model shows better performance than the market where equally weighted portfolio of the same assets is used as market benchmark. However, resulting information ratio does not seem to be so high. It is mainly due to our target macroeconomic factor selection as the inflation is not the best in this context.

We applied our factor mimicking framework to one of the major factors in the economy which is inflation. It is very possible to extend this study by using growth as a target macroeconomic factor in our methodological framework. In the FMP construction process, using regression models does not provide any restriction for asset weights, however Reinforcement Learning can be applied to the model to limit the weights for the future research. Also, it would be reasonable to include realistic portfolio constraints such as transaction costs to see whether the constructed framework is applicable to the real world.

APPENDIX A

ASSET UNIVERSE USED IN THE FMP

Asset Classes	Symbol	Names	DataStream Symbol
Equities	SPX	US - S&P 500 index	S&PCOMP
	UKX	United Kingdom - FTSE 100 index	FTSE100
	SX5E	Eurozone - Euro Stoxx 50 Price index	DJES50I
	NI225	Japan - Nikkei 225 index	JAPDOWA
	XU100	Turkey - BIST 100 index	TRKISTB
	CNX500	India - Nifty 500 index	ICRI500
Government Bonds	-	US Total All Lives Government Bond Indices	AUSGVAL
Commodities	GOLD	Gold Spot Price (per ton)	GOLDBLN
	SILVER	Silver Spot Price (per ounce)	SILVERH
	COPPER	Copper Spot Price (per ounce)	LCPCASH
	SPGSAG	S&P GSCI Agriculture Index Spot	GSAGSPT
	CL.1 Crude Oil WTI	NYMEX WTI Crude Oil futures	CRUDOIL

APPENDIX B

MACROECONOMIC FACTORS USED AS PREDICTORS IN ML

Factor Class	FRED	Description	GSI Description
Output and Income	RPI	Real Personal Income	PI
	INDPRO	IP Index	IP: total
	CUMFNS	Capacity Utilization: Manufacturing	Cap util
Labor Market	PAYEMS	All Employees: Total nonfarm	Emp: total
	UNRATE	Civilian Unemployment Rate	U: all
Housing	HOUST	Housing Starts: Total New Privately Owned	Starts: nonfarm
Consumption, orders, and inventories	DPCERA3M086SBEA	Real personal consumption expenditures	Real Consumption
	RETAILx	Retail and Food Services Sales	Retail sales
	UMCSENTx	Consumer Sentiment Index	Consumer expect
Money and Credit	DTCTHFNM	Total Consumer Loans and Leases Outstanding	N.A.
Interest and Exchange Rates	FEDFUNDS	Effective Federal Funds Rate	Fed Funds
	TB3MS	3-Month Treasury Bill	3 mo T-bill
	T1YFFM	1-Year Treasury C Minus FEDFUNDS	1 yr-FF spread
	EXJPUSx	Japan / U.S. Foreign Exchange Rate	Ex rate: Japan
	TWEXMMTH	Trade Weighted U.S. Dollar Index: Major Currencies	Ex rate: avg
Prices	WPSFD49207	PPI: Finished Goods	PPI: fin gds
	CPIAUCSL	CPI: All Items	CPI-U: all
	CPIULFSL	CPI: All Items Less Food	CPI-U: ex food
	OILPRICEx	Crude Oil, spliced WTI and Cushing	Spot market price

	CUSR0000SA0L2	CPI: All items less shelter	CPI-U: ex shelter
	CUSR0000SA0L5	CPI: All items less medical care	CPI-U: ex med
	PCEPI	Personal Cons. Expend.: Chain Index	PCE defl
	DDURRG3M086SBEA	Personal Cons. Exp: Durable goods	PCE defl: dlbes
	DNDGRG3M086SBEA	Personal Cons. Exp: Nondurable goods	PCE defl: nondble
Stock Market	S&P 500	S&P's Common Stock Price Index: Composite	S&P 500

REFERENCES

Altug, S., & Cakmakli, C. (2015). Forecasting inflation using survey expectations and target inflation: Evidence for Brazil and Turkey. *International Journal of Forecasting*, *32*(1), 138-153. doi: 10.1016/j.ijforecast.2015.03.010

Atkeson, A., & Ohanian, L.E. (2001). Are Phillips curves useful for forecasting? *The Quarterly Review, 25*, 2-11. doi:10.21034/qr.2511

Cenesizoglu, T., & Timmermann, A. (2012). Do return prediction models add economic value? *Journal of Banking and Finance*, *36*(11), 2974-2987. doi:10.2139/ssrn.1913736

Chavez-Hurtado, J., Luis, J., Fregoso, C., & Hector, J. (2013). Forecasting Mexican inflation using neural networks. *23rd International Conference on Electronics, Communications and Computing*, 32-35. doi:10.1109/CONIELECOMP.2013.6525753

Chow, H.K. (2014). Forecasting inflation with a financial conditions index: The case of Singapore. *Annals of Financial Economics*, *8*(2), 1-18. doi:10.1142/S2010495213500097

Garcia, M., Medeiros, M., & Vasconcelos, G. (2017). Real-time inflation forecasting with high-dimensional models: The case of Brazil. *International Journal of Forecasting*, *33*(3), 679-693. doi:10.1016/j.ijforecast.2017.02.002

Jurczenko, E., & Teiletche, J. (2019). Macro factor mimicking portfolios. *SSRN Electronic Journal*. Retrieved from https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3363598

Kucukefe, B. (2018). Forecasting inflation using summary statistics of survey expectations: A machine-learning approach. *Ekonomi-tek*, *7*(1), 1-16.

Lamont, O.A. (2001). Economic tracking portfolios. *Journal of Econometrics*, *105*(1), 161-184. doi:10.1016/S0304-4076(01)00074-4

Stock, J.H., & Watson, M.W. (1999). Forecasting inflation. *Journal of Monetary Economics*, *44*(2), 293-335. doi: 10.1016/S0304-3932(99)00027-6

Zhang, L., & Li, J. (2012). Inflation forecasting using support vector regression. *4th International Symposium on Information Science and Engineering*, 136-140. doi:10.1109/ISISE.2012.37.