UNIVERSITY OF TARTU

Faculty of Social Sciences

School of Economics and Business Administration

Kateryna Kolomiiets

FORECASTING US INFLATION USING MACHINE LEARNING METHODS

Master's thesis

Supervisors:

M. Hakan Eratalay (PhD)

Aleksei Netšunajev (PhD)

Tartu 2018

Name and signature of supervisor: .....

Allowed for defense on .....
(date)

I have written this master's thesis independently. All viewpoints of other authors, literary sources and data from elsewhere used for writing this paper have been referenced.

.....
(signature of author)

Abstract

This thesis considers application machine learning methods like multivariate adaptive regression splines, enhanced adaptive regression through hinges, group method of data handling, regression trees and random forests in order to forecast US inflation. Performance of these methods in forecasting of inflation is poorly investigated. The benchmark model is AR(2). As a result, random forests, group method of data handling multivariate and adaptive regression splines prove to be methods which can be applied in order to forecast US inflation. Experiments are done on two time series samples: stationary and non-stationary. In addition, the results of feature selection were also analyzed.

Keywords: Artificial neural networks, machine learning, statistical learning, inflation, multivariate adaptive regression splines, regression trees, random forests, group method of data handling, forecasting.

JEL classification: C32.

Contents

1 Introduction

Inflation is one of the key indicators of the economic situation since it influences all areas of economy and finance. It is important to know the future path of inflation to make appropriate decisions in monetary policy. Therefore economists continuously try to find robust models which would be able to forecast the future rates of inflation. In recent decades machine learning algorithms became widespread and are already being applied to any data which may be gathered and processed.

The main points uncovered in this thesis are the goodness of forecasting US inflation done with the considered machine learning methods and the results of an automated feature selection process. To the best of our knowledge, there are few machine learning methods which are poorly investigated when dealing with macroeconomic data. And so, this study is aiming to figure out if these methods can do forecasting inflation. To do it the accuracy of U.S. inflation forecasts done with machine learning and autoregressive models are compared. The hypotheses that are tested in this study: multivariate adaptive regression splines outperform AR(2) in forecasting US inflation; group method of data handling can construct models which are outperforming AR(2) in forecasting US inflation; decision trees based methods outperform AR(2) in forecasting US inflation.

Conventional approaches like time series analysis require the previous setting of the model. The methods considered in this study do not need any previous precise model setting and perform automated feature selection, only the general view of the structure of the future model is known at the beginning, the complexity will change.

The novelty of the thesis is that it considers forecasting of the U.S. inflation using methods of machine learning, such as multivariate adaptive regression splines, group method of data handling, regression trees, and random forests. All these methods construct non-linear models for predicting numerical variables but until nowadays stay poorly investigated regarding application for inflation forecasting. Doing this research may be necessary because these methods conduct automated feature selection for constructing predictive models, which gives us an opportunity to notice simultaneous changes of different macroeconomic indexes, which are occurring

in reality.

In this study, the AR(2) model is used as the benchmark model and pseudo out-of-sample forecasting approach. Model estimation is recursive. Forecasts are direct, meaning that forecasted at the step h value of inflation at the period $(t+h)$ defined by values of variables known at the period t. Comparison of the models is made by observing the values of mean squared forecasted error at the step h - $MSFE(h)$. The formula for MSFE(h) is

$$MSFE(h) = \frac{1}{m} \sum_{t=T-m+h}^{N-h} (\hat{y}(\bar{x}_t) - y_t)^2 \tag{1}$$

where h is a step of forecast, m is a number of observations on which the pseudo-out-of-sample exercise is run. All the scripts for this work are developed using programming language R.

The main results of this study are following: the mentioned machine learning methods for constructing forecasting models can be used for forecasting US inflation; multivariate adaptive regression splines outperform AR(2) when taking as input non-stationary data and when having modifications from Friedman (1993) with input stationary data as well as non-stationary; group method of data handling can construct models which are outperforming AR(2) independently from stationarity of the data; regression trees and random forests outperform AR(2) on stationary data; the most often used predictors for CPI are lags of CPI, Wage and Salary Disbursements, money stock M1, Real Personal Consumption Expenditures (Services), Unit Labor Cost, Industrial Production Index Manufacturing, Real GNP/GDP and the number of new residential construction projects that have begun during the quarter.

The overall structure of the study takes the form of six sections, including this introductory Section 1. tehSection 2 provides an overview of research papers devoted to forecasting inflation of US and other countries using as conventional as well as non-conventional approaches. In the Section 3 the explanations for all the used methods are presented: multivariate adaptive regression splines (MARS)*, group

* Friedman (1991), implemented in package `mda` developed by Tibshirani (2016) and package `earth` by Milborrow (2011) - more sophisticated version of MARS, based on Friedman (1993)

method of data handling (GMDH) [†], regression trees[‡] and random forests[§]. The datasets description is provided in the Section 4; results and discussions are located in the Sections 5, which are followed by the Section 6 with conclusions; illustrations for produced forecasts are provided in appendix A.

2 Literature review

2.1 Forecasting Inflation

Forecasting American inflation was a point of interest since 1980's. Fama and Gibbons (1984) compare interest rate models for forecasting inflation are with univariate time series model. Bernanke and Woodford (1997) discuss the inflation forecasting role in monetary authority policy making.

Banerjee and Marcellino (2006) compare factor models for forecasting US inflation (Stock and Watson (1999)) with models using automated feature selection method *PcGets* developed by Hendry and Krolzig (1999).

A considerable contribution to forecasting US inflation was made by Mark W. Watson. According to Stock and Watson (2008) the most common way to forecast inflation is applying the instruments of times series analysis involving as predictors: the lags of inflation (ARIMA univariate models); the unemployment rate, output gap, output growth according to Phillips curve; forecasts of other variables; and some other variables. The last approach is considered to be the least precise if compared to aforementioned when time series analysis methods are used.

2.2 Application of neural networks for forecasting inflation in different countries

Alongside with progress in computational capacity development, new methods for data processing came under the spotlight. In the following lines, the papers devoted

[†] using package `GMDH` by Dag and Yozgatligil (2016) and code that was developed for this study
[‡] using package `rpart` by Therneau et al. (2017)
[§] using package `randomForest` by Liaw and Wiener (2002)

to application these approaches for forecasting inflation of different countries and comparison of their performance with traditional methodologies will be discussed.

Ajoy K. Palit (2005) indicates that models which are constructed using neural networks have such distinguished features as:

- general nonlinear mapping between a subset of the past time series values and the future time series values;
- the capability of capturing essential functional relationships among the data, which is valuable when they are not a priori known or are very difficult to describe mathematically and/or when the collected observation data are corrupted by noise;
- universal function approximation capability that enables modeling of arbitrary nonlinear continuous functions to any degree of accuracy;
- the capability of learning and generalization from examples using the data-driven self-adaptive approach.

There is a set of papers considering the comparison of the performance of artificial neural networks and traditional methods used for time series forecasting.

First to mention, a work by Aiken (1999) where a general neural network for forecasting *US inflation* was applied. Here predictive variables were: Producer Price Index (PPI), crude materials; PPI, intermediate materials; PPI, capital equipment; PPI, finished consumer goods; PPI, finished goods; PPI, finished goods less food and energy; change in sensitive materials prices; change in money supply M1, and change in money supply M2.

Moshiri and Cameron (2000), in which the performance of Back Propagation Artificial Neural Network models was compared with the traditional econometric approaches (structural reduced-form model, an ARIMA model, a vector autoregressive model, and a Bayesian vector autoregression model) for *out-of-sample* forecasting the inflation rate of *Canada*. The results show the hybrid BPN models can forecast as well as all the traditional econometric methods and outperform them in some cases.

Another paper of Kock and Teräsvirta (2016) analyzes using artificial neural networks for inflation forecasting for *Finland*. Here Finnish inflation rate is being forecasted by using linear autoregressive and nonlinear neural network models. Another example is a work of Nakamura (2005), where neural networks outperform traditional approaches. And another example of paper was a comparison of artificial neural networks and $AR(1)$ is Choudhary and Haider (2012). Here forecasts are estimated for 28 countries of OECD, in 45% artificial neural networks could predict inflation better than $AR(1)$. Besides, a similar exercise was done by Haider and Hanif (2009). Here feed-forward neural network with back-propagation training algorithm is outperforming $AR(1)$ and $ARIMA$. The conclusion is done after analysis of RMSE.

In the technical report Szafranek (2017) considers single hidden-layer feed-forward artificial neural networks. The forecast combination of bagged single hidden-layer artificial neural networks outperforms many competing models. Here the competing models are: RW - the pure random walk model for the seasonally adjusted monthly headline inflation, AO - the random walk model for the monthly headline inflation closely related to the Atkeson and Ohanian (2001) specification, the unobserved-component stochastic volatility model (UCS) à la Stock and Watson (2007), the recursive autoregressive process of order one (AR1), the autoregressive process of order twelve (AR12), the seasonal autoregressive moving average model (BS), the factor augmented vector autoregression (FAV), the judgment forecast (JD), the dynamic factor model (DFM), the Bayesian vector autoregressive model with the Sims-Zha priors (SZ), the Bayesian vector autoregressive model with the Villani steady-state prior (VI), the forecast combination of the bagged single hidden-layer feed-forward artificial neural networks described in this paper (ANN), the combination of the two best performing models (CB1), the convolution of the two best performing models (in terms of the RMSFE) and the ANN model (CB2).

2.3 Machine learning for forecasting macroeconomic data

In recent years, there has been an increasing amount of literature on the usage of neural networks for forecasting macroeconomic data.

Lin et al. (2008) introduces a hybrid causal model for predicting the occurrence of currency crises by using the combination of learning ability of neural networks with the inference mechanism of fuzzy logic. The empirical results show that the proposed neuro-fuzzy model leads to a better prediction of crisis.

A large-scale comparison study for the major machine learning models for time series forecasting is done by Ahmed et al. (2010). The models which were studied are multilayer perceptron, Bayesian neural networks, radial basis functions, generalized regression neural networks, K-nearest neighbor regression, CART regression trees, support vector regression and Gaussian process. These models were applied on the monthly *M3 time series competition data* (including 731 time series of macroeconomic data). The best two models turned out to be multilayer perceptron and the Gaussian process regression.

Garcia et al. (2017) apply for forecasting *Brazilian inflation* in real time such models as LASSO, adaptive-LASSO, Random Forest, Complete Subset Regression with Targeted Predictors. Here forecast mean absolute errors and root mean squared errors for out-of-sample forecast were taken into consideration while comparing different models. For $h = 1$, the LASSO and FOCUS (expert) forecasts deliver the best predictions. For the second horizon, the adaptive LASSO is superior to all other models considered. For the remaining horizons, the complete subset regression dominates all other alternatives. The model that had the smallest errors in most horizons of the forecast was the complete subset regression.

Many papers are investigating using multivariate adaptive regression splines in various fields, as this method was applied in Rounaghi et al. (2015) for stock price forecasting. A similar situation is with group method of data handling for constructing artificial neural networks.

As a conclusion, there are very few papers discussing the application of machine learning methods for forecasting US inflation. To the best of our knowledge, no previous study has investigated the performance of multivariate adaptive regression splines on macroeconomic data, and no previous research has examined the performance of random forests and GMDH on US inflation. To fill the research gap, a motivation for this study is to apply these machine learning methods and compare

their performance with other methods.

After all, there were not many papers studying the performance of forecasting models created with the help of the machine or statistical learning models using data about US inflation.

3 Methods used for forecasting

Suppose there is a connection between the values of inflation which are denoted as y_t and a set of other variables at current and previous periods of time $\bar{x}_t = (x_{1,t}, ..., x_{n,t})^T$. Here $x_{i,t}$ may be lagged value of y_t or lagged value of some other variable taking part in forecasting, index t denotes correspondence to the value of y_t in period t. The task is to approximate the relation as a function of these variables - that is to model the dependence between response variable y_t and data $\bar{x}_t = (x_{1,t}, ..., x_{n,t})^T$, while having realizations $\{y_t, x_{1,t}, ..., x_{n,t}\}_{t=1}^T$ where T is the total number of periods when the observations were available:

$$y_t = f(x_{1,t}, x_{2,t}, ..., x_{n,t}) + \varepsilon_t \tag{2}$$

The additive stochastic component ε_t, whose expected value is defined to be zero, usually reflects the dependence of y_t on quantities other than $\bar{x}_t = (x_{1,t}, ..., x_{n,t})^T$ that are neither controlled or observed.

3.1 Multivariate adaptive regression splines (MARS)

The first method which is applied in this study is multivariate adaptive regression splines, developed by Friedman (1991). This method has an automates feature selection and deals with many numerical variables as input and numerical responses. The main reason why MARS is chosen for applying is its advantage to deal with low as well as high dimensional settings.

There are two approaches to function approximation in high dimensional settings: parametric and non-parametric. Here the main historical steps in the development of non-parametric function approximation are provided.

In parametric approximation the principal approach has been to fit known function $g(\bar{x}_t|\{a_j\}_{j=1}^p)$ to the data most often by least squares. In general case:

$$\begin{cases} \hat{f}(\bar{x}_t) = g(\bar{x}_t|\{\hat{a}_j\}_{j=1}^p) \\ \{\hat{a}_j\}_{j=1}^p = \underset{\{a_j\}_{j=1}^p}{argmin} \sum_{i=1}^T [y_i - g(\bar{x}_i|\{a_j\}_{j=1}^p)]^2 \end{cases} \tag{3}$$

where $\{a_j\}_{j=1}^p$ is a set of p parameters, $j = 1, ..., p$. In low dimensional settings ($n \leq 2$), global parametric modeling was generalized using piecewise and local parametric fitting and roughness penalty methods. The most popular piecewise polynomial fitting procedures are based on splines where parametric functions are taken to be polynomials of degree q and derivatives to order $(q-1)$ are required to be continuous. Procedure is implemented by contracting a set of (globally defined) basis functions that span the space of qth order spline approximations and fitting the coefficients of the basis function expansion to the data by ordinary least squares. Direct extension of piecewise parametric modeling to higher dimensions ($n > 2$) is straightforward in principle but difficult in practice (Friedman (1991)). Local parametric approximations take the form: $\hat{f}(\bar{x}_t) = g(\bar{x}_t|\{\hat{a}(\bar{x}_t)_j\}_{j=1}^p)$ but the set of parameters is different at each evaluation point $\bar{x}_t$ and is obtained by locally weighted least squares fitting: $\{\hat{a}_j(\bar{x}_t)\}_{j=1}^p = \underset{\{a_j\}_1^p}{argmin} \sum_{i=1}^T \omega(\bar{x}_t, \bar{x}_i)[y_i - g(\bar{x}_i|\{a_j\}_{j=1}^p)]^2$ where $\omega(\bar{x}, \bar{x}')$ is a weighting function which is chosen to put the dominant mass on points $\bar{x}'$ close to $\bar{x}$.

The roughness penalty approximations are defined by:

$$\hat{f}(\bar{x}_t) = \underset{g(\cdot)}{argmin}\{\sum_{i=1}^T [y_i - g(\bar{x}_i)]^2 + \lambda R(g)\} \tag{4}$$

where $R(g)$ is a functional that increases with increasing roughness of the function $g(\bar{x})$. The parameter λ regulates the trade off between the roughness of $g(\cdot)$ and its fidelity to the data. Non-parametric approximations take the form of low

dimensional expansions:

$$\hat{f}(\bar{x}_t) = \sum_{j=1}^{J} \hat{g}_j(\bar{z}_{j,t}), \tag{5}$$

where each $\bar{z}_{j,t}$ is a preselected subset of $\{x_{1,t}, ..., x_{n,t}\}$, such that $\bar{z}_{j,t} \neq \bar{z}_{i,t} \iff i \neq j, \forall i, j \in \{1, ..., J\}$, J is a total number of $\bar{z}_{j,t}$. Dimension of each subset $\bar{z}_{j,t}$ has to be ≤ 2. After selecting the variable subsets $\{\bar{z}_{j,t}\}_{j=1}^{J}$ the corresponding functions are obtained, for example, using least squares:

$$\{\hat{g}_j(\bar{z}_{j,t})\}_{j=1}^{J} = \underset{g_j}{argmin} \sum_{i=1}^{T} [y_i - \sum_{k=1}^{J} g_k(\bar{z}_{k,i})]^2 \tag{6}$$

Projection pursuit computation uses approximation of the form:

$$\hat{f}(x) = \sum_{m=1}^{M} f_m(\sum_{i=1}^{n} a_{i,m} x_i) \tag{7}$$

where M is a number of functions $f(\cdot)$ sum of which is sufficient for approximation.

Recursive partitioning regression model takes the form:

$$\bar{x}_t \in R_m \implies \hat{f}(\bar{x}_t) = g_m(\bar{x}_t | \{a_j\}_{j=1}^{p}) \tag{8}$$

where $\{R_m\}_1^M$ is a set of disjoint subregions representing a partition of domain for function $f(\bar{x})$. Most commonly used are constant functions (Breiman et al. (1984)): $g_m(\bar{x}_t | a_m) = a_m$. The partitioning is accomplished through the recursive splitting of previous subregions. The starting region is entire domain of $f(\bar{x})$. At each stage of the partitioning, all existing subregions are each optimally split into two subregions, which are separated by some point $\bar{b}$ using goodness-of-fit criterion on the resulting approximation (8). The recursive partitioning is continued until a large number of subregions is generated. The subregions are then recombined reversely until an optimal set is reached, based on a criterion that penalizes both for lack-of-fit and increasing number of regions.

Recursive partitioning regression model is more viewed as geometrical procedure

and Friedman (1991) casted this approximation with arithmetic notions of adding and multiplying:

$$\hat{f}(\bar{x}_t) = \sum_{m=1}^{M} a_m B_m(\bar{x}_t) \tag{9}$$

where M is the number of subregions. Basis function takes form: $B_m(x) = I[x \in R_m]$, where $I[\cdots]$ is an indicator function such that it takes value 1 if $x \in R_m$ and 0 otherwise.

Coefficients $\{a_m\}_{m=1}^{M}$ are jointly adjusted to give the best fit to the data. $\{R_m\}_{m=1}^{M}$ are the same as in (8). Here the aim of recursive partitioning is not only to adjust the coefficient values to fit the data best, but also to derive a good set of basis functions.

MARS is conducted in two stages:

- Forward stepwise algorithm. At this point basis functions are produced but corresponding regions are not disjoint but overlap.
- Backward stepwise algorithm. At this point basis functions which do not improve the fit are eliminated.

As a result, approximation made by MARS takes a form:

$$\hat{f}(\bar{x}_t) = a_0 + \sum_{\substack{K_m=1 \\ i \in V(m)}} a_m B_m(x_{i,t}) + \sum_{\substack{K_m=2 \\ (i,j) \in V(m)}} a_m B_m(x_{i,t}, x_{j,t}) + \sum_{\substack{K_m=3 \\ (i,j,k) \in V(m)}} a_m B_m(x_{i,t}, x_{j,t}, x_{k,t}) + ... \tag{10}$$

where $V(m) = \{v(k,m)\}_{k=1}^{K_m}$ - is a (predictive) variable set, associated with the mth basis function $B_m(\cdot)$, K_m is number of variables which take part in a spline

$$B_m(x_{v(1,m)}, ..., x_{v(K_m,m)}) = \prod_{k=1}^{K_m} [s_{k,m} \cdot (x_{v(k,m)} - b_{k,m})]_+ \tag{11}$$

where $s_{k,m} = \pm 1$, $b_{k,m}$ is a point which separates subregions - knot, $[x]_+$ is a function which takes value x if $x > 0$ and otherwise 0. After first stage backwards deletion process is applied. The terms whose removal causes the smallest increase in the

residual squared error are deleted. New model is denoted as $\hat{f}_\lambda$ where λ is the number of terms in the model estimation. As the criterion, generalized cross-validation is used:

$$GCV(\lambda) = \frac{\sum_{t=1}^{T}(y_t - \hat{f}_\lambda(\bar{x}_t))^2}{(1 - M(\lambda)/T)^2} \tag{12}$$

where $M(\lambda)$ is the effective number of parameters in the model. Both numbers of terms in the model and number of parameters used in selecting the optimal positions of the knots are accounted. If there are r linearly independent basis functions in the model and K knots were selected at the forward process, the formula is $M(\lambda) = r + cK$, where $c = 3$ according to Hastie et al. (2001)

In this study the packages which implement MARS in R developed by Tibshirani (2016) and Milborrow (2011) are used. The last is based on Friedman (1993) and is enhanced version of traditional MARS. In the later version, the computational costs are reduced due to using the parameter of the priority queue search depth and the frequency with which the optimization over input variables is performed. In this thesis, the key development is in introducing memory into MARS so that results from earlier iterations are encountered while doing optimization for the later ones.

3.2 Group method of data handling (GMDH)

The second method to be applied in this study is a group method of data handling. According to Kartal Koc and Bozdogan (2015) underlying idea of MARS is similar to group method of data handling introduced by Ivakhnenko (1966). This algorithm is designed to model the functional relationship between the response and predictor variables which is settled directly from self-organization of the data as well.

In GMDH approximation is made in a form of high order Kolmogorov-Gabor polynomial:

$$y_t = a_0 + \sum_{i=1}^{m} a_i \cdot x_{i,t} + \sum_{i=1}^{m}\sum_{j=1}^{m} a_{ij} \cdot x_{i,t} \cdot x_{j,t} + \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{m} a_{ijk} \cdot x_{i,t} \cdot x_{j,t} \cdot x_{k,t} + ... \tag{13}$$

where y_t is a value of the dependent variable at the moment t, $\bar{x}_t = (x_{1,t}, ..., x_{n,t})^T$ is the vector of predicting variables, $\bar{a}$ is a set of coefficients or weights. This function is linear in parameters $\bar{a}$ and non-linear in $\bar{x}_t$. GMDH algorithm considers all pairwise polynomial combinations of external inputs:

$$y = a_0 + a_1 \cdot x_i + a_2 \cdot x_j + a_3 \cdot x_i \cdot x_j. \tag{14}$$

In this algorithm polynomial-type neurons are organized in such neural network architecture: GMDH yields a model consisting of nested polynomials. In terms

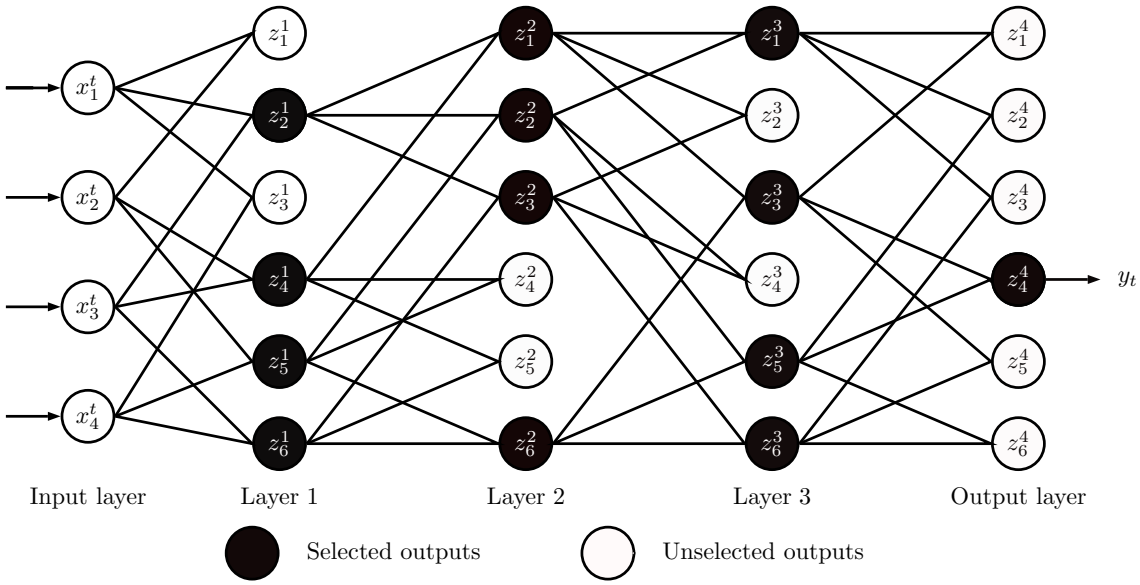

Figure 1: Example of neural network with 3 hidden layers produced by GMDH algorithm

of artificial neural networks each neuron is a polynomial pairwise combination of inputs, which at the Layer 1 is an input vector, but at further layers inputs are results of these polynomial combinations of previous layers. At each iteration of the GMDH algorithm, the outputs of each neuron of a new layer are compared with values of predicted variable y_t. The data, available for model estimation is divided into two parts - training and testing set in proportion 70 : 30. All the coefficients are estimated using the training set. Those outputs yielding the smallest mean squared error compared to values from the testing set are considered as inputs for the next layer. At some iteration, the minimum difference between the neuron output and

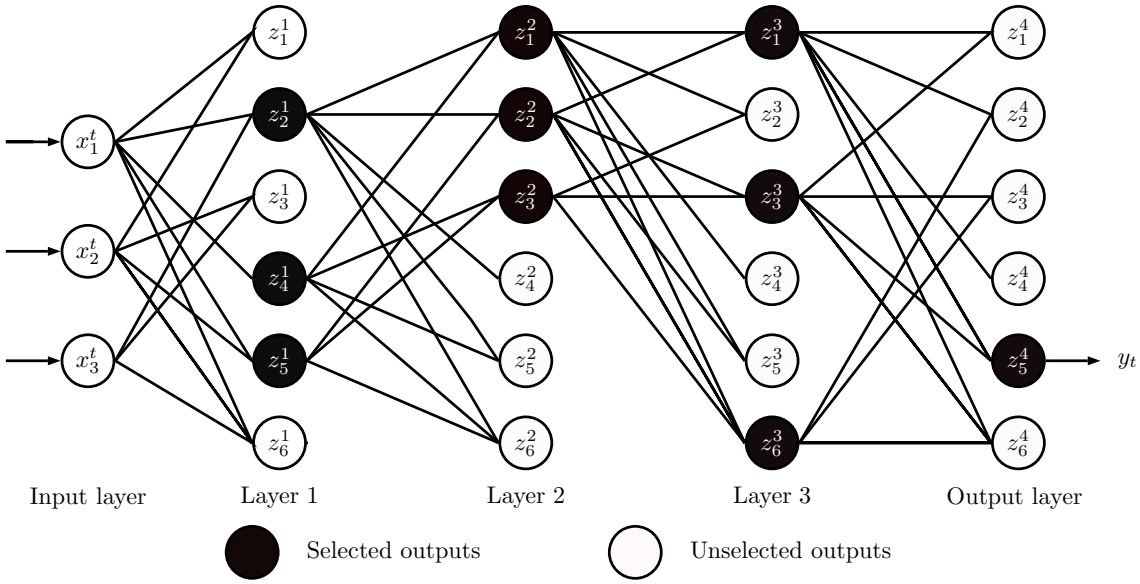

Figure 2: Example of neural network with 3 hidden layers produced by RGMDH algorithm

the predicted variable starts to increase, and algorithm selects the model which gave the smallest mean squared error compared to data from the testing set on the previous layer. The number of surviving neurons is chosen by the developer. For this study, a script with GMDH that takes many variables as inputs and has one layer to avoid overfitting was developed. As a result, GMDH yields a neural network of feedforward type.

In this study the package `GMDH` by Dag and Yozgatligil (2016) which has a function of short-term forecasting `fcast` for univariate time series up to 5 steps ahead is used. Here GMDH-type algorithms which build neural networks involve sigmoid $z = \frac{1}{1+e^{-y}}$, radial basis function $z = e^{-y^2}$, polynomial $z = y$ that simply forwarding the polynomial for the next layer and tangent function $z = tan(y)$ as activation function. Activation functions are computations which are made on the activation (Lopes and Ribeiro (2014)). Activation of a neuron is a function which takes a vector as input and produces scalar value as output. In case of GMDH algorithm activation is $y = f(x_1, x_2) = a + b \cdot x_1 + c \cdot x_2 + d \cdot x_1 \cdot x_2$. There are no quadratic terms since algorithm produces a model of nested polynomials so that in case of need they appear on later iterations. `Fcast` gives an opportunity to choose a type of activation

function or apply all of them and select the best fit. In addition, recursive GMDH (RGMDH) is available in the package as well. The last one considers individual effects of inputs on the result. In RGMDH there are two types of neurons. First type of neurons is the same as in conventional GMDH, in the second type activation functions take the form $y = a + \sum_{i=1}^{r} b_i \cdot x_i$. Here $r \leq n$ is the number of inputs in a neuron of the second type. This package can work only with univariate time series and so a decision to create a script which utilizes conventional GMDH which was described earlier was made.

3.3 Decision making trees

3.3.1 Regression trees

Regression trees are based on the principle of recursive partitioning which may be described as well as MARS, in basis functions. Likewise, regression trees use recursive partitioning in order to create a subset of predicting data so that values of responsive variables in each subset are as similar as possible. The main advantage of regression trees is easy interoperability of the output. In this thesis the R package `rpart` by Therneau et al. (2017) is applied. According to Berk (2008) the key difference between MARS and regression trees lies in the nature of basis functions used. Unlike in (11), basis functions for regression trees are indicator functions and overall the approximation model takes the form:

$$\hat{f}(\bar{x}_t) = \sum_{m=1}^{M} a_m I(\bar{x}_t \in R_m) \tag{15}$$

Having as a criterion minimization of $\sum_{t=1}^{T}(y_t - \hat{f}(\bar{x}_t))^2$, in regression trees best a_m is defined as:

$$a_m = \frac{1}{T}\sum_{t=1}^{T}(y_t|\bar{x}_t \in R_m) \tag{16}$$

That is, for each observation $\bar{x}_t$ can be assigned to some set R_m which is defining terminal node m. The value of dependent variable y ia assigned to that $\bar{x}_t$ as a mean

of that node. "The collection of means for all of the terminal nodes are, therefore, fitted values analogous to the fitted values from conventional parametric regression. They represent how the numerical response is related to the predictors."Berk (2008).

The whole procedure starts with considering the whole domain as one set, which is partitioned afterwards. Suppose a predictive variable j is considered and a split point b, a pair of half-planes is defined:

$$R_1(j,b) = \{\bar{x}_t | x_{j,t} \leq b\}, R_2(j,b) = \{\bar{x}_t | x_{j,t} > b\} \tag{17}$$

After division a pair (j,b) that solves a problem

$$\underset{j,b}{min}[\underset{a_1}{min} \sum_{x_t \in R_2(j,b)} (y_t - a_1)^2 + \underset{a_2}{min} \sum_{x_t \in R_1(j,b)} (y_t - a_2)^2]. \tag{18}$$

Similarly, the inner minimization problems are solved by

$$\begin{cases} a_1 = \frac{1}{T} \sum_{t=1}^{T} (y_t | \bar{x}_t \in R_1(j,b)) \\ a_2 = \frac{1}{T} \sum_{t=1}^{T} (y_t | \bar{x}_t \in R_2(j,b)) \end{cases} \tag{19}$$

After finding the best split, the process is continued on each partition. Tree size belongs to tuning parameters and has to be chosen depending on the data to deal with. There is a strategy first to grow a tree K, stopping the growing process after some specific node size is reached. After that a procedure of cost-complexity pruning has to be implemented. Cost complexity criterion:

$$C_\alpha(K) = \sum_{m=1}^{|K|} \sum_{x_i \in R_m} (y_i - \frac{1}{N_m} \sum_{x_i \in R_m} y_i)^2 + \alpha \cdot |K| \tag{20}$$

where N_m is the number of elements in $\{x_i \in R_m\}$, $|K|$ is a number of terminal nodes in a tree K. The main idea is to find such α that minimizes the cost complexity criterion.

Regression trees answer a question: what is the average value of a given target for all the examples for which a given set of conditions on the input attributes is

true? This causes problems when dealing with non-stationary data since the mean value is continuously changing.

3.3.2 Random forests

The last method used in this study is random forests. Random forests were developed by Breiman (2001). It is a substantial modification of bagging that builds a large collection of de-correlated trees and then averages them. According to Hastie et al. (2001) bagging or bootstrap aggregation is a technique for reducing the variance of an estimated prediction function. In this study the R package `randomForest` by Liaw and Wiener (2002) used.

The essential idea in bagging is to average many noisy but approximately unbiased models, and hence reduce the variance. The bootstrap approach here means that there are randomly drawn datasets with replacement from the training data, each sample having the same size as the original training set. And so, B different training datasets are created. But, when the random forests are applied, according to Berk (2008), at each node split construction, a new sample of predicting variables without replacement is considered. The tree construction is finished when the needed tree size is reached. Afterward, the newly constructed tree is applied to the testing data, which was not present in the training set and the assignment value of output is stored along with values on the nodes. The tree construction is repeated many times which yields a set of trees $\{K_b\}_{b=1}^{B}$. For an observation $\bar{x}_t$ out of testing sample or a newcomer observation and estimation model takes the form:

$$\hat{f}(\bar{x}_t) = \frac{1}{B} \sum_{b=1}^{B} K_b(\bar{x}_t) \tag{21}$$

Since the main construction unit for Random Forests is a decision tree, this algorithm does not work well with non-stationary data.

4 Data

As it was mentioned before, the data used in this study was taken from Korobilis (2017). As an alternative, this study could be conducted using the dataset by McCracken and Ng (2016) which contains observations of 134 monthly macroeconomic U.S. indicators.

Table 1: Data description

Mnemonic	Description	Tcode	Source	Range
CPI	Consumer Price Index, Quarterly Vintages	1	Philly	1947Q1-2015Q3
IPM	Industrial Production Index, Manufacturing	1	Philly	1947Q1-2015Q3
HSTARTS	Housing Starts	2	Philly	1947Q1-2015Q3
CUM	Capacity Utilization Rate, Manufacturing	1	Philly	1948Q1-2015Q3
M1	M1 Money Stock	1	Philly	1947Q1-2015Q3
RCOND	Real Personal Consumption Expenditures, Durables	1	Philly	1947Q1-2015Q3
RCONS	Real Personal Consumption Expenditures, Services	1	Philly	1947Q1-2015Q3
RG	Real Government Consumption & Gross Investment, Total	1	Philly	1947Q1-2015Q3
RINVBF	Real Gross Private Domestic Investment, Nonresidential	1	Philly	1947Q1-2015Q3
ROUTPUT	Real GNP/GDP	1	Philly	1947Q1-2015Q3
RUC	Unemployment Rate	3	Philly	1948Q1-2015Q3
ULC	Unit Labor Costs	1	Philly	1947Q1-2015Q3
WSD	Wage and Salary Disbursements	1	Philly	1947Q1-2015Q3
DYS	Default yield spread (Moody's BAA - AAA)	4	St Louis	1947Q1-2015Q3
NAPM	Purchasing Manager's Index	4	St Louis	1947Q1-2015Q3
NAPMII	Inventories Index	4	St Louis	1947Q1-2015Q3
NAPMNOI	New Orders Index	4	St Louis	1947Q1-2015Q3

Tcode: 1 - first difference of log-transformed data, 2 - log transformation, 3 - first difference, 4 - no transformation. Source: Philly - Philadelphia Fed; St. Louis - FRED

Table 2: Dataset with 1 lag of all variables, untransformed.

	CPI(t)	IPM(t)	HSTARTS(t)	$\cdots$	NAPMII(t)	NAPMNOI(t)	CPI(t+1)
1948:Q2	23.63	13.90	1321.33	$\cdots$	44.60	49.40	24.00
1948:Q3	24.00	14.00	1464.00	$\cdots$	49.30	51.30	24.40
1948:Q4	24.40	14.10	1350.33	$\cdots$	47.60	45.70	24.20
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
2015:Q1	237.07	104.90	1055.33	$\cdots$	49.80	61.00	235.20
2015:Q2	235.20	104.73	978.00	$\cdots$	51.70	52.40	236.93
2015:Q3	236.93	105.13	1157.67	$\cdots$	51.30	55.10	237.87

In this study dataset by Korobilis (2017) is used because it is a shrunk subset of McCracken and Ng (2016). And so, applying the considered algorithms on the second dataset would require more computational recourses like computing time and memory space. As a result, it is more convenient to deal with the small dataset by Korobilis (2017). McCracken and Ng (2016) dataset exploration is left for future research.

Table 3: Dataset with 1 lag of all variables, transformed.

	CPI(t)	IPM(t)	HSTARTS(t)	NAPMII(t)	···	NAPMNOI(t)	CPI(t+1)
1948:Q3	6.16	2.87	29.16	49.30	···	51.30	6.61
1948:Q4	6.61	2.85	28.83	47.60	···	45.70	-3.29
1949:Q1	-3.29	-4.76	28.36	41.70	···	43.30	-4.43
...	...	...	...	...	...	...	...
2015:Q1	-0.79	3.19	27.85	49.80	···	61.00	-3.16
2015:Q2	-3.16	-0.64	27.54	51.70	···	52.40	2.94
2015:Q3	2.94	1.52	28.22	51.30	···	55.10	1.57

Table 4: Dataset with 4 lags of all variables, not transformed.

	CPI(t-3)	IPM(t-3)	···	CPI(t-2)	IPM(t-2)	···	CPI(t-1)	IPM(t-1)	···	CPI(t+1)
1949:Q1	23.63	13.90	···	24.00	14.00	···	24.40	14.10	···	23.93
1949:Q2	24.00	14.00	···	24.40	14.10	···	24.20	13.93	···	23.90
1949:Q3	24.40	14.10	···	24.20	13.93	···	23.93	13.50	···	23.73
...	...	...	...	...	...	...	...	...	...	...
2015:Q1	235.43	101.60	···	236.83	103.10	···	237.53	104.07	···	235.20
2015:Q2	236.83	103.10	···	237.53	104.07	···	237.07	104.90	···	236.93
2015:Q3	237.53	104.07	···	237.07	104.90	···	235.20	104.73	···	237.87

Table 1 shows the list of variables used, their sources, the dates when the observations were available, and transformations should be applied. Here the annualized transformation of CPI was considered. The annualization was made with multiplying transformed to stationary data by 4. As a result, obtained a time series which is interpreted as an annual increase if all the conditions stay the same is obtained.

According to the results of Dickey-Fuller test forecasted variable CPI is not stationary while benchmark model $AR(2)$ has a stationarity requirement. For that reason, stationarity transformations should be applied. The results of models estimated both on transformed and untransformed data are taken into consideration as if they are outcomes of models built on two different datasets. All the variables in the dataset containing stationary times series of inflation CPI are transformed as well. These transformations are done according to transformation data provided by Korobilis (2017).

All the mentioned methods for estimating forecasting models are applied on four sets of observations: two datasets contain first lags of all the predictors available including the CPI with corresponding original value of $\{y_{t+h}\}_{h=1}^{8}$ - formed from stationary and non-stationary data and two datasets contain lagged up to four step values of all the predicting variables: $X = \{\bar{x}_{t-4}, \bar{x}_{t-3}, \bar{x}_{t-2}, \bar{x}_{t-1}\}$. All the models

Table 5: Dataset with 4 lags of all variables, transformed.

	CPI(t-3)	IPM(t-3)	···	CPI(t-2)	IPM(t-2)	···	CPI(t-1)	IPM(t-1)	···	CPI(t+1)
1949:Q2	6.16	2.87	···	6.61	2.85	···	-3.29	-4.76	···	-0.56
1949:Q3	6.61	2.85	···	-3.29	-4.76	···	-4.43	-12.64	···	-2.80
1949:Q4	-3.29	-4.76	···	-4.43	-12.64	···	-0.56	-14.07	···	-1.13
···	···	···	···	···	···	···	···	···	···	···
2015:Q1	2.10	0.66	···	2.37	5.86	···	1.18	3.73	···	-3.16
2015:Q2	2.37	5.86	···	1.18	3.73	···	-0.79	3.19	···	2.94
2015:Q3	1.18	3.73	···	-0.79	3.19	···	-3.16	-0.64	···	1.57

are constructed so that they forecast the CPI values at future periods $\{t+h\}_{h=1}^{h+8}$. The validation is conducted using the testing subset of the sample available. The forecast is direct, i.e. knowing the values of predictive variable at the time period t the models forecasting $y_{t+1}, y_{t+2}, \ldots, y_{t+8}$ are being constructed by machine learning methods. Structure of datasets used in this study is provided in Tables 2, 3, 4 and 5. When constructing a forecasting model on each of them, $CPI(t+1)$ is dependent variable. In the mentioned tables $h = 1$ estimating models for forecasting h steps ahead is realized by shifting the variable CPI(t+h) relatively to the rest of present variables.

According to mentioned features of the package `GMDH`, which was possible to apply only to univariate time series. Results of applying this package are denoted in Tables 7, 6 as "GMDH, n lagged CPI". Other GMDH results were obtained after applying the script with GMDH, which takes many variables as input and was developed for this study.

5 Results and discussions

5.1 Best methods

Table 6 and Table 7 present the results of this study. The first line in both tables is an array of values of the MSFE(h) for forecasting CPI for each quarter starting from the next quarter up to eight quarters (two years) ahead done with AR(2) models. In similar investigations Korobilis (2017) uses as benchmark model AR(2), Stock and Watson (2007) use AR(AIC), where the order of the lags is defined according to the Akaike Information Criterion. For this thesis a decision to use AR(2) was

Table 6: Results for the stationary data

	h=1	h=2	h=3	h=4	h=5	h=6	h=7	h=8	Total $\sum_{i=1}^{5}$
AR(2)	4.777	6.049	6.303	7.042	7.471	7.892	8.737	9.354	-
RGMDH,3 lagged CPI	0.913**	0.919	0.89	0.871	0.914	-	-	-	4.51
GMDH, 4 lagged CPI	0.963	0.912	0.896	0.907	0.93	-	-	-	4.61
MARS, 1 lag of all	2.387	1.414	1.535	1.117	1.187	0.982	0.997	1.202	7.64
EARTH, 1 lag of all	1.123	1.265	0.925	0.989	1.053	1.126	1.152	1.083	5.35
MARS, lagged values	1.986	2.339	1.981	1.439	1.789	1.588	1.692	1.499	9.53
EARTH, lagged values	1.015***	1.072**	0.962	1.015	1.301	1.214	1.042	1.012	5.37
GMDH, lagged values	1.055**	0.951	0.96	0.975	1.097	1.042	0.964	0.888	5.04
RT, 1 lag of all	1.238	1.231	1.292	1.189	1.239	0.834	0.892	1.252	6.19
RT, lagged values	1.485	1.129	0.977	0.912	1.334	1.464	1.254	1.195	5.84
Random forest, 1 lag of all	1.028	0.885	0.925	0.841	0.822	0.692	0.72	0.727	4.5
Random forest, lagged values	1.007**	0.839**	0.837**	0.74**	0.756**	0.743	0.722	0.7	4.18
GMDH, 1 lag of all	0.991	1.088	0.885	0.931	1.066	0.945	1.082	0.99	4.96

Asterisks next to the relative MSFEs provide the level of statistical significance with which the compared model gives more precise forecast than $AR(2)$, according to the Diebold-Mariano test: 1%(***),5%(**), and 10%(*); see Diebold and Mariano (1995).

made, taking into consideration insignificant difference of the performance of these two methods on the dataset considered. Coefficients for AR(2) were estimated by `ar.ols` from the package `stats`. Forecasted values were obtained using a function `predict` from the same package.

All the lines starting from the second contain normalized values of the MSFE(h) for all the rest of applied models. Normalization is conducted in the following way:

$$a_{i,h} = \frac{MSFE_i(h)}{MSFE_{AR(2)}(h)} \tag{22}$$

where i belongs to the set of applied methods, $MSFE_{AR(2)}(h)$ is a $MSFE$ for the model $AR(2)$ performing the forecast for a step h. That is, the table shows the results of division MSFE(h) for certain method by MSFE(h) for AR(2). Those methods having values in the tables less than one indicate better performance to construct forecasting models comparing to $AR(2)$. The MSFE(h) is calculated according to the

Table 7: Results for the raw data

	h=1	h=2	h=3	h=4	h=5	h=6	h=7	h=8	Total $\sum_{i=1}^{5}$
AR(2)	3.05	4.054	5.556	7.129	9.19	12.396	16.563	21.974	-
RGMDH,3 lagged CPI	1.212	1.274	1.397	4.684	23.709	-	-	-	32.28
GMDH, 4 lagged CPI	1.147	1.256	1.447	1.775	2.269	-	-	-	7.89
MARS, 1 lag of all	0.335***	0.882	0.831	0.641***	0.443***	0.386***	0.259***	0.207***	3.13
EARTH, 1 lag of all	0.337***	0.697**	0.805	0.871	0.887	0.748	0.718	0.71	3.6
MARS, lagged values	0.316***	1.06	0.842	0.784**	0.443***	0.495***	0.306***	0.207***	3.44
EARTH, lagged values	0.311***	0.625**	0.721**	0.853	0.891	0.916	0.802	0.761	3.4
GMDH, lagged values	0.323***	0.616***	1.046	0.761	0.896	0.67**	0.652**	0.682	3.64
RT, 1 lag of all	32.709	24.7	18.114	14.188	11.059	8.241	6.2	4.809	100.77
RT, lagged values	32.201	24.316	17.831	13.967	11.133	8.311	6.11	4.633	99.45
Random forest, 1 lag of all	5.804	4.33	3.266	2.612	2.047	1.539	1.187	0.878	18.06
Random forest, lagged values	6.621	5.111	3.767	3	2.305	1.742	1.255	0.943	20.8
GMDH, 1 lag of all	0.294***	0.59***	0.777**	0.832	0.817	0.71	0.597**	0.506**	3.31

Asterisks next to the relative MSFEs provide the level of statistical significance with which the compared model gives more precise forecast than $AR(2)$, according to the Diebold-Mariano test: 1%(∗ ∗ ∗),5%(∗∗), and 10%(∗); see Diebold and Mariano (1995).

Formula 1. In order to examine the accuracy of forecasts, the Diebold-Mariano(DM) test (Diebold and Mariano (1995)) was used as well.

For one-step-ahead forecasting, when dealing with *stationary data*, the best performance was demonstrated by models constructed with recursive group method of data handling from the package `GMDH` by Dag and Yozgatligil (2016) taking as input three lagged values of CPI. Results of DM test show that forecasts made by models constructed with this method are more accurate than forecasts made using AR(2) with statistical significance. MSFE(h) for this method is the lowest as well. The hypothesis that group method of data handling can construct models which are outperforming AR(2) independently from stationarity of the data is supported. The possible reason why it performed better than AR(2) might be such that this method yields non-linear models with lags of CPI as predictors and that the linear model is not enough for the description of the inflation growth. Good results were

demonstrated by GMDH taking as inputs lagged values of CPI. Overall, as for the first step forecasting, dealing with stationary data, all the applied methods proved to be able to construct forecasting models which are not significantly worse than the ones yielded by AR(2).

Random forests yielded models which provided statistically significantly according to DM test and MSFE(h) the best forecasts for two, three, four and five steps ahead. It may be happening due to the principle of averaging results of many trees, which helps to find the closest to the real description of the model. In this study, these criteria in forecast assessment were taken into account as they had been justified in Diebold (2015). Here those methods yielding the smallest MSFE and being statistically significantly more accurate than forecasts produced by AR(2), according to results of DM test as the best forecasts, are considered as the best.

The worst performance was demonstrated by multivariate adaptive regression splines for the package `mda` by Tibshirani (2016). But the version with modification by Friedman (1993) called "Fast MARS" which is enhanced version of traditional MARS. The reason why it happened this way may that MARS produces overfitting models due to possible loss of accuracy in the enhanced version of MARS Friedman (1993). If raw data is considered, multivariate adaptive regression splines and group method of data handling using many predictors as an input show outstanding results in producing forecasting models, considering statistical significance provided by DM test and smaller values of MSFE(h). In addition, in case of non-stationary data regression trees cannot help. As it was said before, these methods work well when the range of response values is limited. Random forests fail the task not so strong but nevertheless cannot be considered as methods which can construct good models for forecasting non-stationary data. Overall, multivariate adaptive regression splines do not outperform AR(2) when taking as input stationary data, but outperform when taking as non-stationary input data and when having modifications from Friedman (1993). Regression trees and random forests outperform AR(2) on stationary data.

It is worth to mention, that in case of forecasting CPI when having raw data, the best results were shown by models taking not only CPI and its lags as predictors. This may happen because of losing some information due to transformations applied

to the data. Another possible reason may be such that non-linear interactions of other macroeconomic indexes can more precisely describe the level of inflation. In addition, it is crucial to remember autoregressive models cannot perform well on non-stationary data (Mann and Wald (1943)) as well as regression trees and random forests. Multivariate adaptive regression splines and GMDH so not have a similar restriction.

Graphs illustrating forecasts for $h = 1, 4, 8$ for both stationary and raw data are provided in appendix A. The worst forecasting performance for stationary data was demonstrated by multivariate regressive adaptive splines (Friedman (1991)) from the package `mda` developed by Tibshirani (2016). The worst results for the raw data were delivered by Regression trees from the package `rpart` by Therneau et al. (2017) and Random forests from the package `randomForest` by Liaw and Wiener (2002).

5.2 Selected predictors

As it was mentioned previously, machine learning algorithms possess automated feature selection. After applying the methods, the selected predicting variables were retrieved, and the total number of occurrences was counted.

Illustrations showing frequencies are provided in Figures 3, 4, 5, 6 for data retrieved from EARTH, in Figures 7, 8, 9, 10 for data retrieved from GMDH, in Figures 11, 12 for data retrieved from random forest. Title for each figure indicates which model was applied to which dataset. All the following conclusions are based on the visual analysis of provided figures.

Paying attention to approaches which show a good performance according to MSFE(h) and results of DM test, it is worth to highlight the frequency results for such methods: GMDH and EARTH using one lag of all (Figures 3, 7) data and GMDH using many lags for all variables (Figure 9) on *raw data* and random forest only on lagged values of *stationary data* (Figure 11). And so, the most often used predictors for CPI are lags of CPI, Wage and Salary Disbursements WSD, money stock M1, Real Personal Consumption Expenditures (Services) RCONS, Unit Labor Cost ULC, Industrial Production Index Manufacturing IPM, Real GNP/GDP ROUTPUT and HSTARTS - the number of new residential construction projects

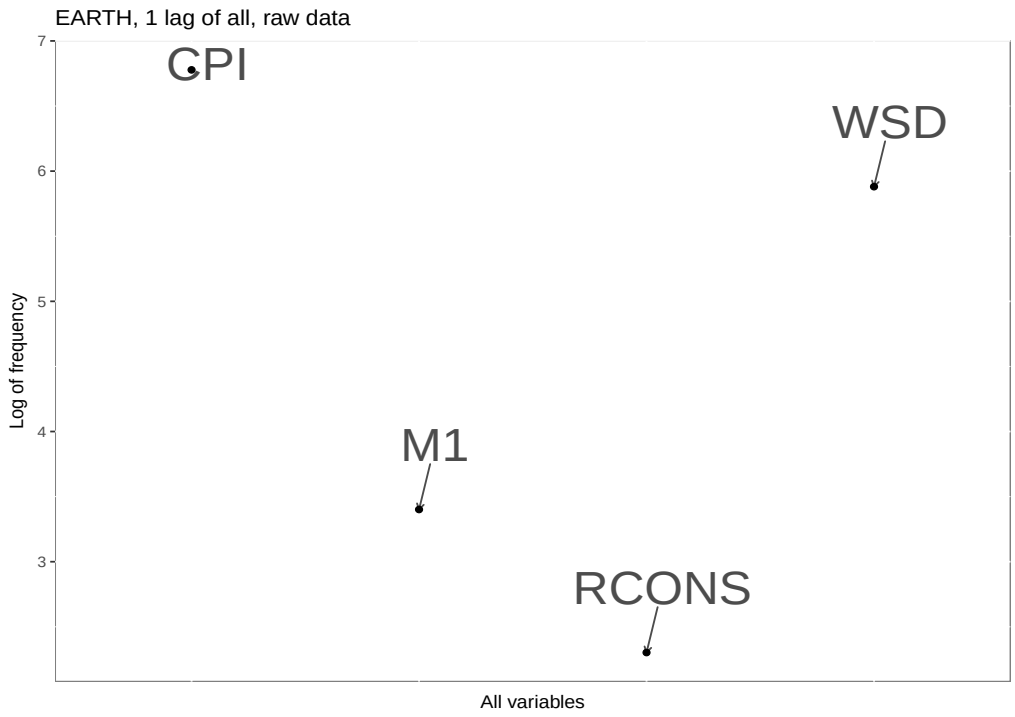

Figure 3: Frequencies retrieved from EARTH

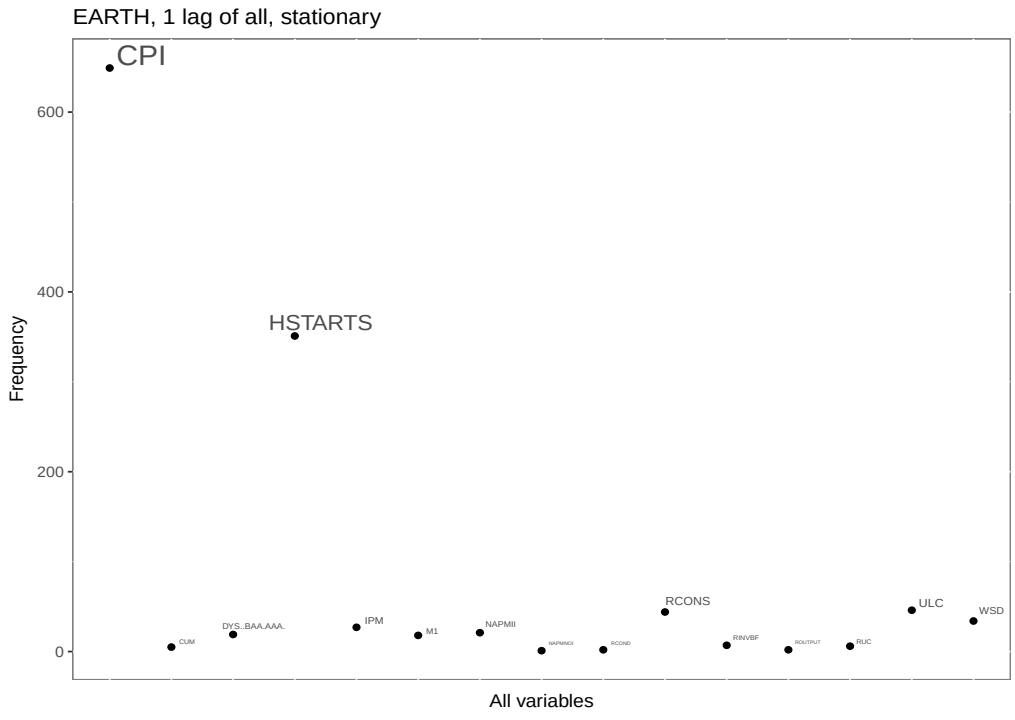

Figure 4: Frequencies retrieved from EARTH

that have begun during the quarter.

The presence of lagged values of CPI in results for both stationary and untrans-

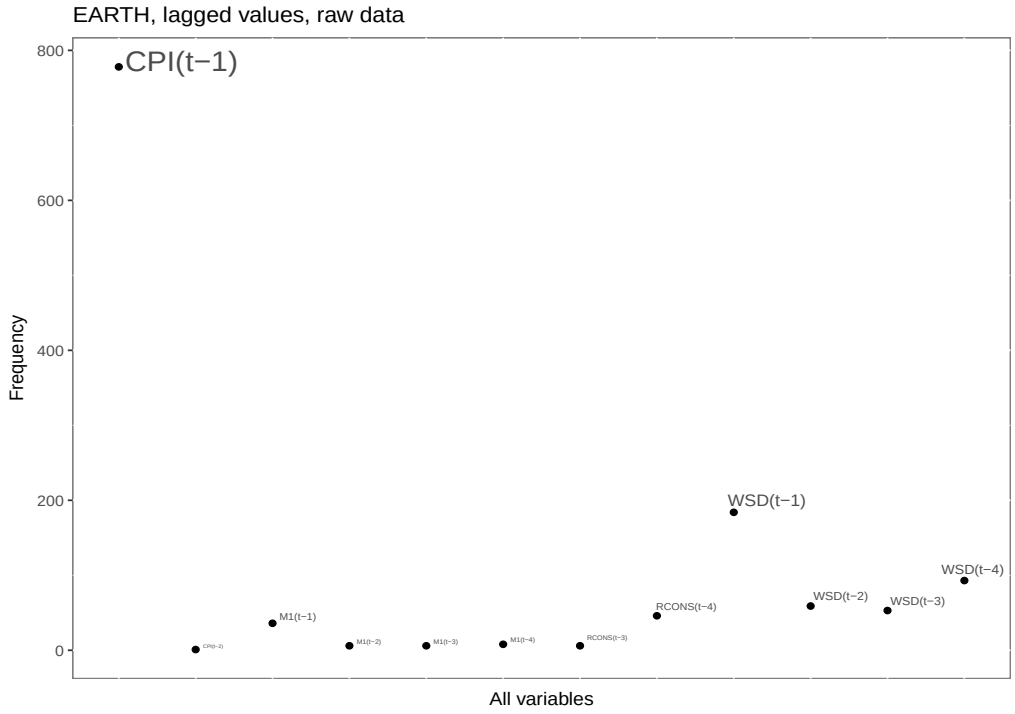

Figure 5: Frequencies retrieved from EARTH

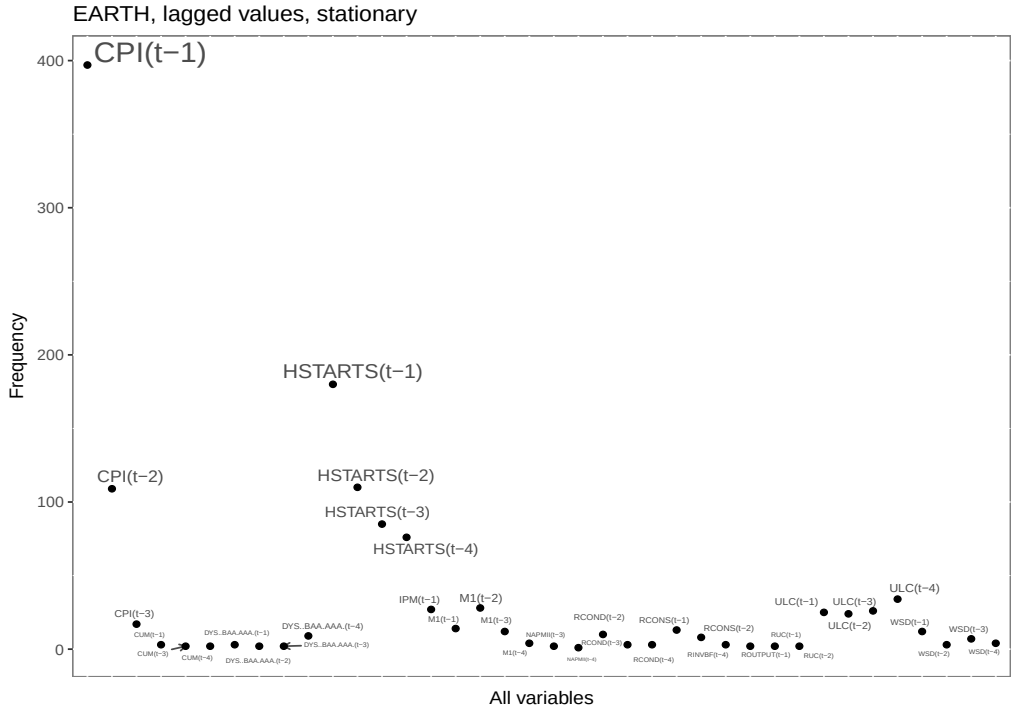

Figure 6: Frequencies retrieved from EARTH

formed data is not surprising as well, it does not disagree with the theory, according to Stock and Watson (2008). Using ARIMA type univariate models for forecasting

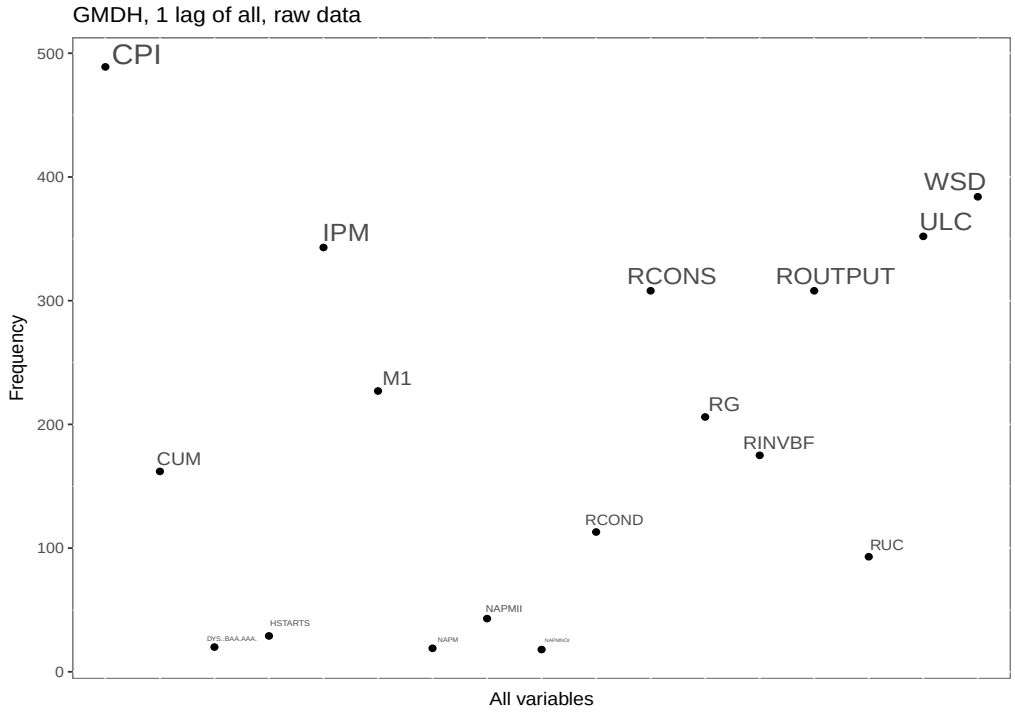

Figure 7: Frequencies retrieved from GMDH

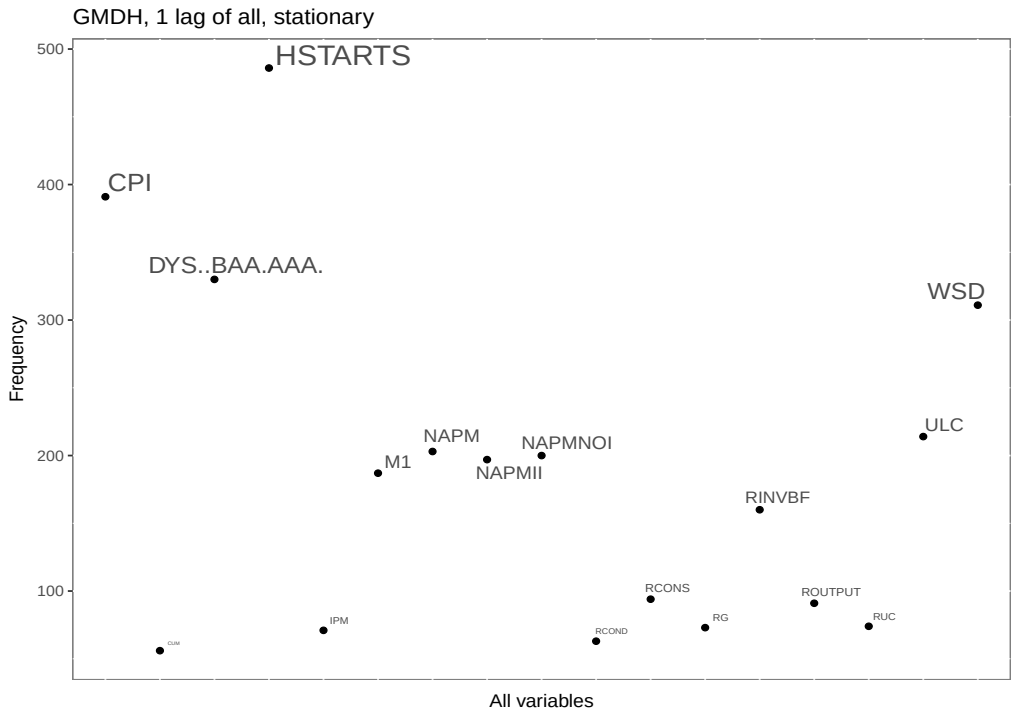

Figure 8: Frequencies retrieved from GMDH

CPI was mentioned first among conventional approaches. The connection between inflation and Wage and Salary Disbursements and Unit Labor Cost as well as num-

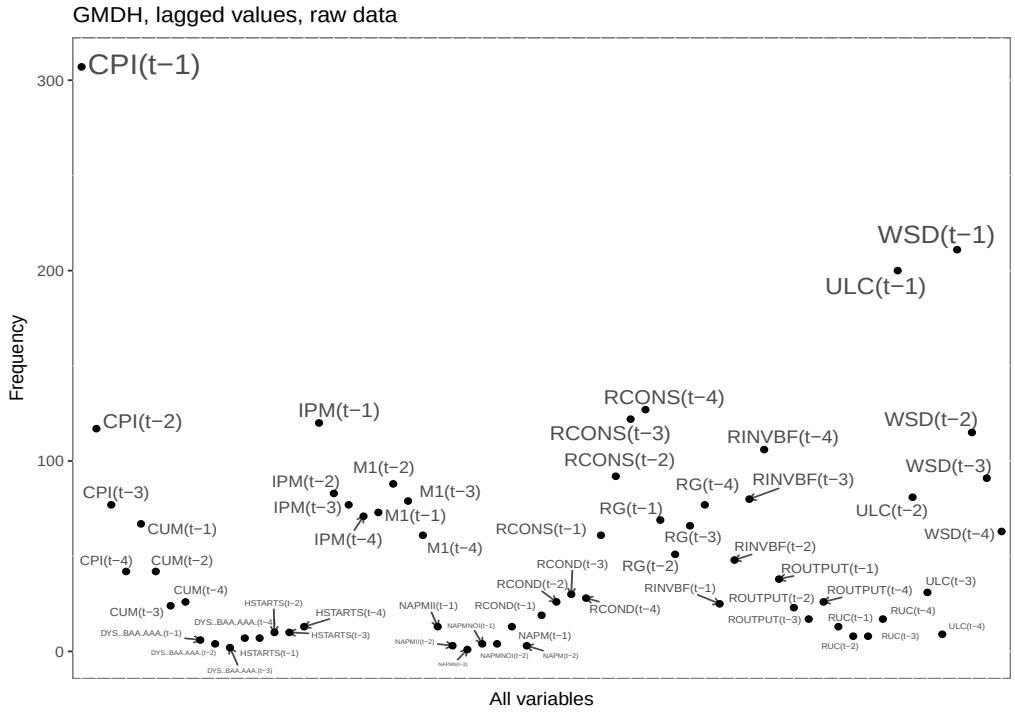

Figure 9: Frequencies retrieved from GMDH

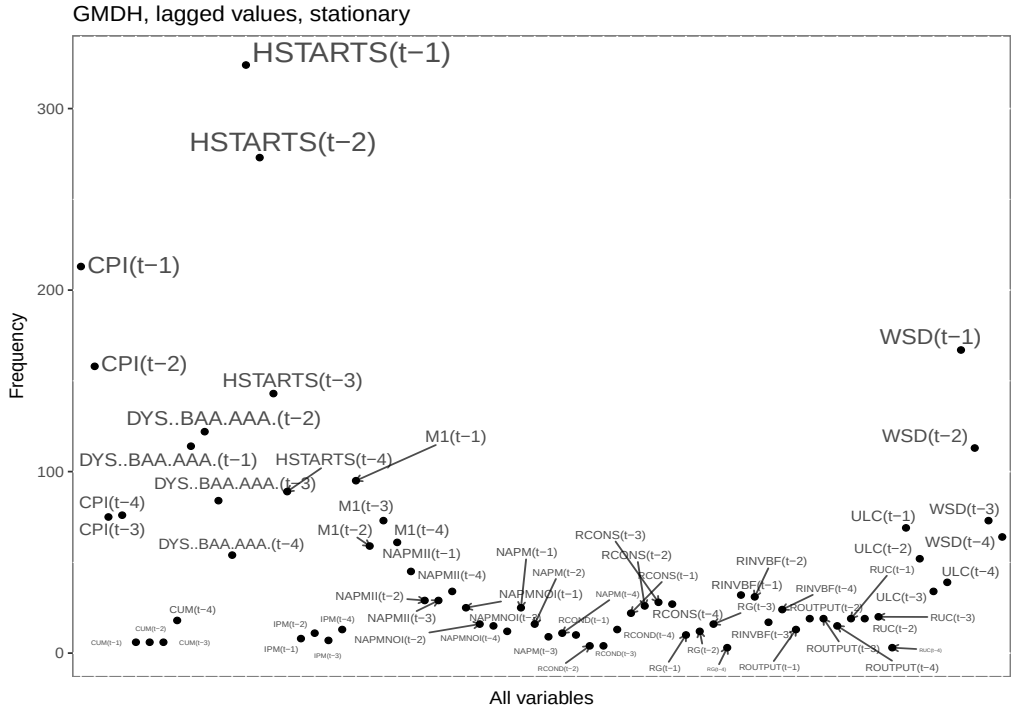

Figure 10: Frequencies retrieved from GMDH

ber of houses started to be constructed Real Personal Consumption Expenditures (Services) and Industrial Production Index Manufacturing, and Real GNP/GDP

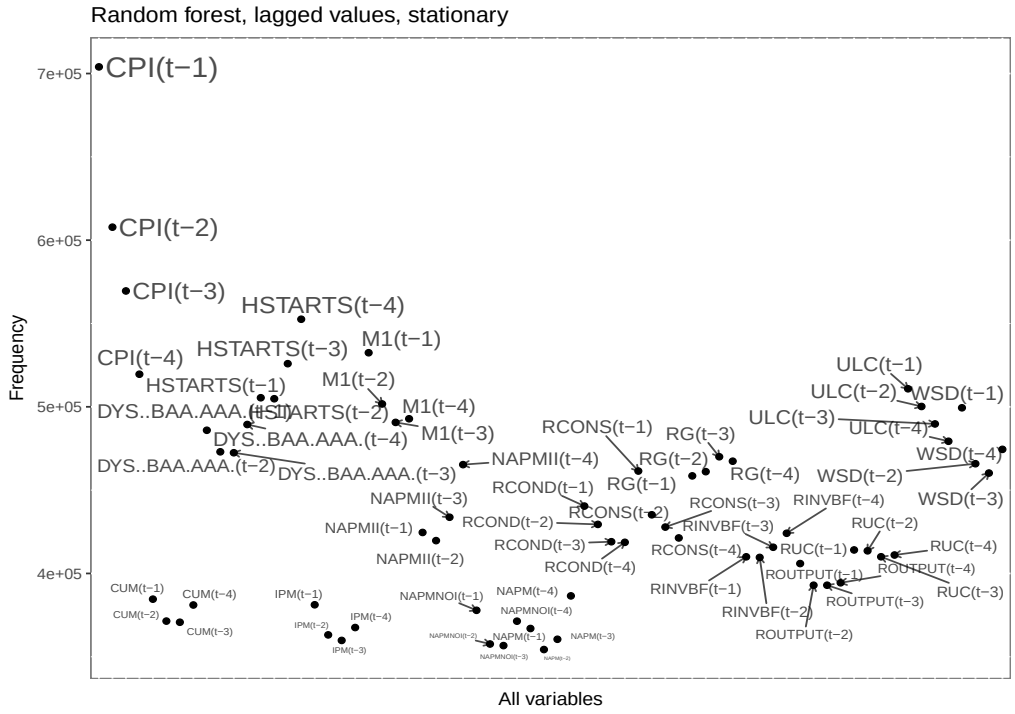

Figure 11: Frequencies retrieved from random forest

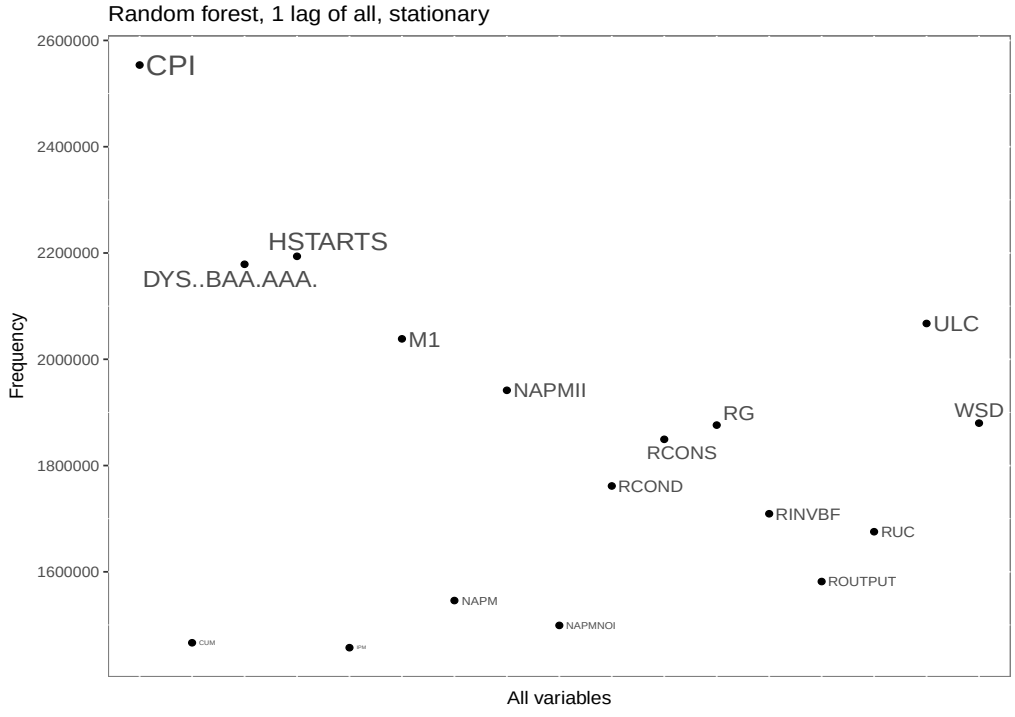

Figure 12: Frequencies retrieved from random forest

possibly can be explained by demand shock. Regarding money stock M1 similar results were obtained by a researcher of Central Bank of Canada Atta-Mensah (1996)

when modifying a model P^* proposed by Hallman et al. (1989). The presence of HSTARTS is not surprising since housing is among eight groups of consumer goods which are taken into consideration while calculating CPI. However, when looking into reports of Bureau of Labor Statistics (BLS (2017)) housing as a component in CPI is represented by "Rent of shelter" having relative importance in July 2017 of 33.374 and "Owners' equivalent rent of residence" having relative importance 24.529 according to News Release Consumer Price Index for July 2017 from Bureau of Labor Statistics.

All the rest of figures - 4, 5, 6, 8, 10, 12 provide frequencies of features for the rest of methods, where these values could be retrieved. On these figures predicting variables are: lags of CPI, HSTARTS, Default yield spread, WSD, M1, ULC.

6 Conclusion

The aim of this study was to figure out if the machine learning methods like multivariate adaptive regression splines, group method of data handling, regression trees, random forests can do forecasting inflation. In order to do it the accuracy of U.S. inflation forecasts done with these machine learning methods and autoregressive models of the second order were compared.

As a result, these methods can construct forecasting models for CPI. Multivariate adaptive regression splines do not outperform AR(2) when taking as input stationary data, but outperform when taking as input non-stationary data and when having modifications; group method of data handling can construct models which are outperforming AR(2) independently from stationarity of the data; regression trees and random forests outperform AR(2) on stationary data. It is crucial to remember AR(2) cannot perform well on non-stationary data. Regression trees and random forests have the similar restriction. Group method of data handling and multivariate adaptive regression splines do not have such restrictions and show excellent performance when dealing with raw data.

Methods belonging to machine learning approaches possess automated feature selection which is used to identify best predictors. The most often used predictors

for CPI are lags of CPI, Wage and Salary Disbursements WSD, money stock M1, Real Personal Consumption Expenditures (Services) RCONS, Unit Labor Cost ULC, Industrial Production Index Manufacturing IPM, Real GNP/GDP ROUTPUT and HSTARTS - the number of new residential construction projects that have begun during the quarter. These results do not contradict economic theory and common sense.

Acknowledgments

I would like to acknowledge the supervision of Senior Economist in Economics and Research Department of Eesti Pank Ph.D. Aleksei Netšunajev and Research Fellow at the School of Economics and Business Administration at the University of Tartu Ph.D. M. Hakan Eratalay, the useful comments of Senior Research Fellow and Programme Manager of Master's programme in quantitative Economics at the School of Economics and Business Administration at the University of Tartu Ph.D. Jaan Masso and Senior Economist in Economics and Research Department of Eesti Pank Ph.D. Lenno Uusküla, the help in mastering R of Economist in Economics and Research Department of Eesti Pank Nicolas Reigl, valuable comments and moral support of Associate Professor at Université d'Orléans as well as useful comments of my opponent Ph.D. student at the University of Tartu Luca Alfieri.

A Forecasting illustrations

Here are the graphs illustrating forecasts for $h = 1, 4, 8$ for both stationary and raw data.Title of each figure contains the information which method yielded the smallest MSFE(h) for certain step.

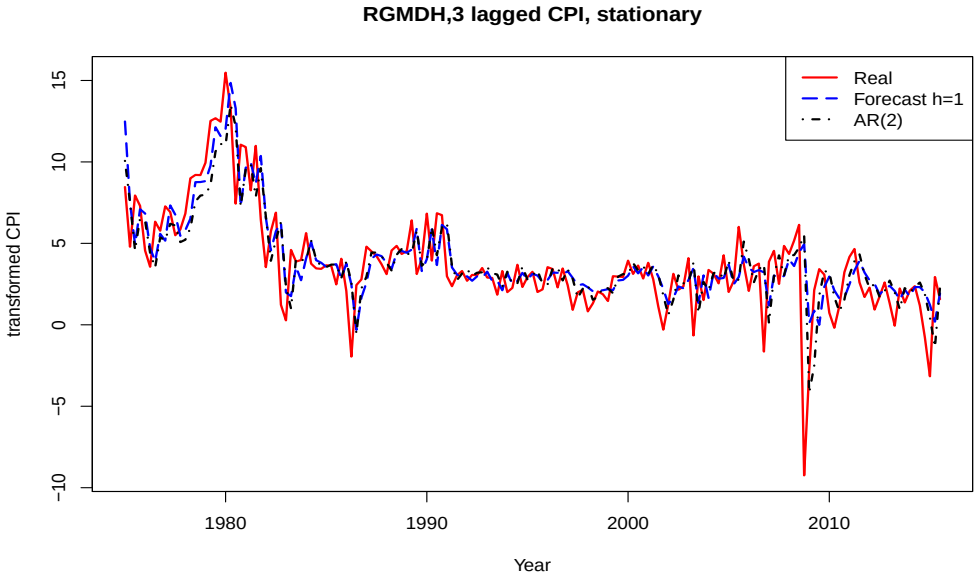

Figure 13: Forecasting stationary series of CPI 1 step ahead

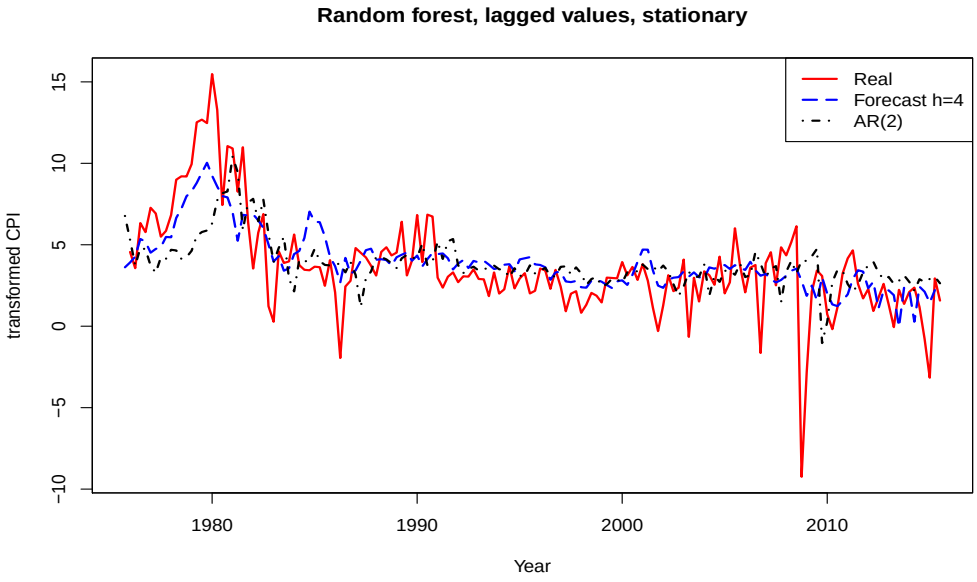

Figure 14: Forecasting stationary series of CPI 4 steps ahead

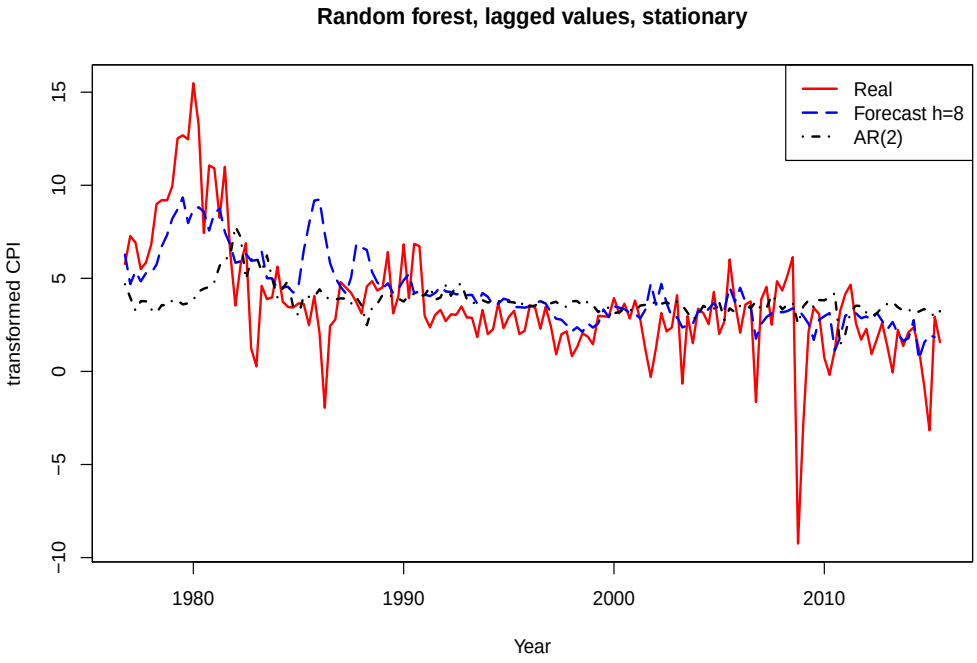

Figure 15: Forecasting stationary series of CPI 8 steps ahead

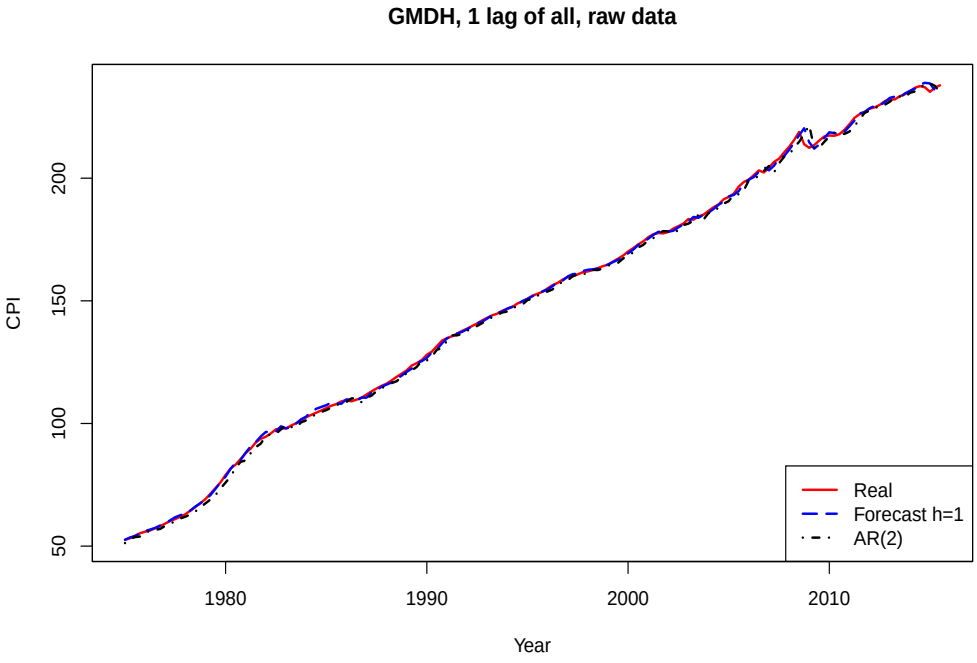

Figure 16: Forecasting raw series of CPI 1 step ahead

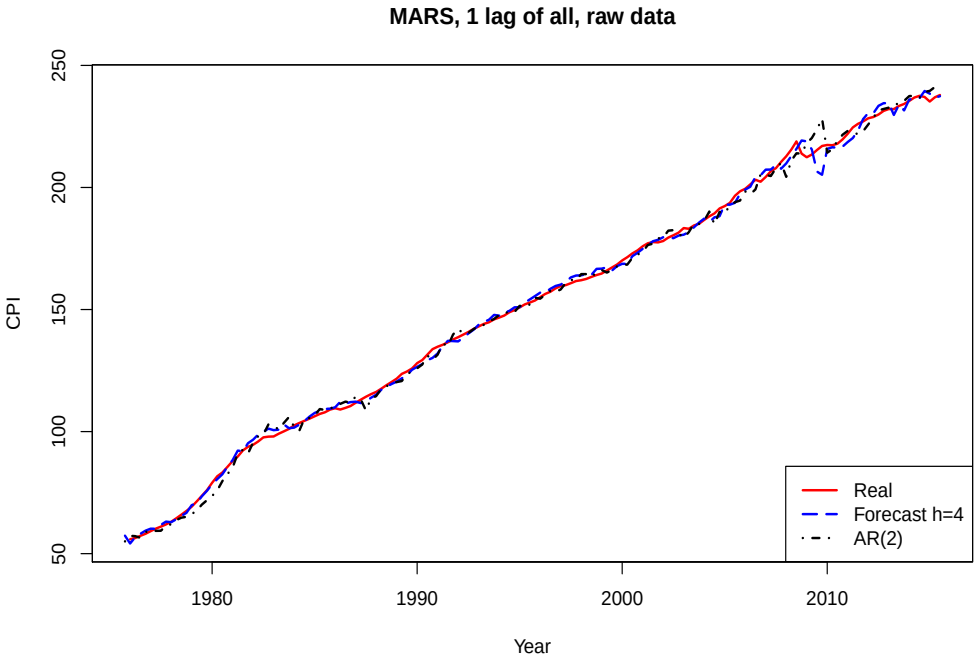

Figure 17: Forecasting raw series of CPI 4 steps ahead

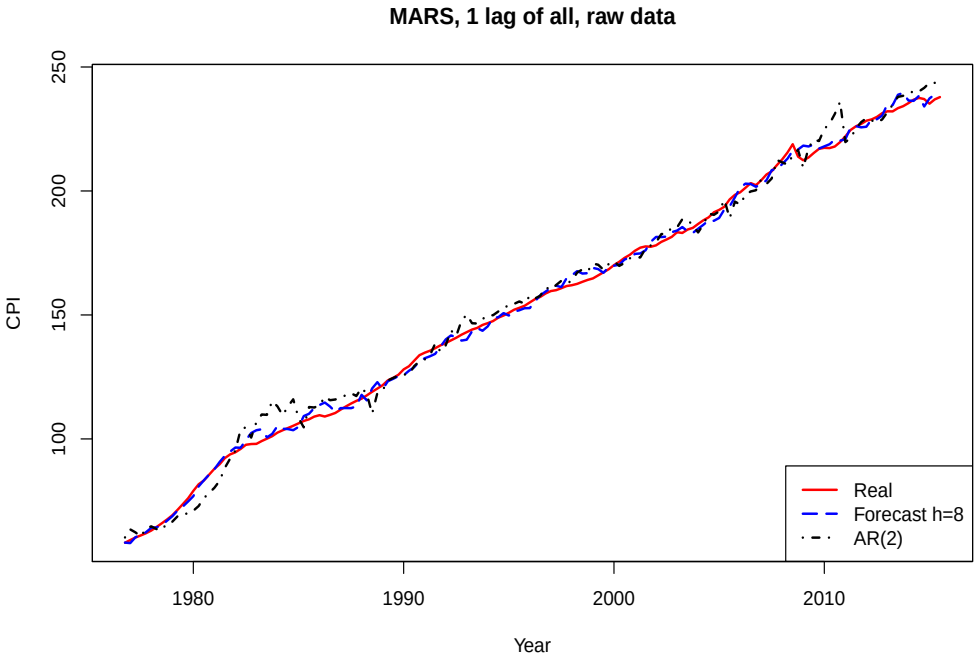

Figure 18: Forecasting raw series of CPI 8 steps ahead

References

Ahmed, N. K., A. F. Atiya, N. E. Gayar, and H. El-Shishiny (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews 29*(5-6), 594–621.

Aiken, M. (1999). Using a neural network to forecast inflation. *Industrial Management & Data Systems 99*(7), 296–301.

Ajoy K. Palit, D. P. (2005). *Computational Intelligence in Time Series Forecasting*. Advances in Industrial Control. Springer-Verlag London.

Atkeson, A. and L. E. Ohanian (2001). Are Phillips curves useful for forecasting inflation? *Quarterly Review* (Win), 2–11.

Atta-Mensah, J. (1996, 02). The empirical performance of alternative monetary and liquidity aggregates.

Banerjee, A. and M. Marcellino (2006). Are there any reliable leading indicators for us inflation and gdp growth? *International Journal of Forecasting 22*(1), 137 – 151.

Berk, R. (2008). *Statistical Learning from a Regression Perspective*. Springer Series in Statistics. Springer New York.

Bernanke, B. S. and M. Woodford (1997). Inflation forecasts and monetary policy. *Proceedings*, 653–686.

BLS (2017). Consumer price index - july 2017. Technical report USDL-17-1100, Bureau of Labor Statistics.

Breiman, L. (2001, Oct). Random forests. *Machine Learning 45*(1), 5–32.

Breiman, L., J. Friedman, C. Stone, and R. Olshen (1984). *Classification and Regression Trees*. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis.

Choudhary, M. A. and A. Haider (2012). Neural network models for inflation forecasting: an appraisal. *Applied Economics 44*(20), 2631–2635.

Dag, O. and C. Yozgatligil (2016). *GMDH: Short Term Forecasting via GMDH-Type Neural Network Algorithms*. R package version 1.6.

Diebold, F. X. (2015). Comparing predictive accuracy, twenty years later: A personal perspective on the use and abuse of diebold-mariano tests. *Journal of Business & Economic Statistics 33*(1), 1–1.

Diebold, F. X. and R. S. Mariano (1995). Comparing predictive accuracy. *Journal of Business & Economic Statistics 13*(3), 253–263.

Fama, E. F. and M. R. Gibbons (1984). A comparison of inflation forecasts. *Journal of Monetary Economics 13*(3), 327 – 348.

Friedman, J. H. (1991). Multivariate adaptive regression splines. *The Annals of Statistics 19*(1), 1–67.

Friedman, J. H. (1993, May). Fast mars. Technical report 110, Stanford University Department of Statistics.

Garcia, M. G., M. C. Medeiros, and G. F. Vasconcelos (2017). Real-time inflation forecasting with high-dimensional models: The case of brazil. *International Journal of Forecasting 33*(3), 679 – 693.

Haider, A. and M. N. Hanif (2009). Inflation forecasting in pakistan using artificial neural networks. *Pakistan Economic and Social Review 47*(1), 123–138.

Hallman, J. J., R. D. Porter, and D. H. Small (1989). M2 per unit of potential gnp as an anchor for the price level. *Fed. Res. Bull. 75*, 263.

Hastie, T., R. Tibshirani, and J. Friedman (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. New York, NY, USA: Springer New York Inc.

Hendry, D. F. and H.-M. Krolzig (1999). Improving on 'data mining reconsidered' by k.d. hoover and s.j. perez. *The Econometrics Journal 2*(2), 202–219.

Hyndman, R. J. (2017). *forecast: Forecasting functions for time series and linear models*. R package version 8.1.

Ivakhnenko, A. (1966). Group method of data handling: A rival of the method of stochastic approximation. *Soviet Automatic Control 13*, 43–71.

Kartal Koc, E. and H. Bozdogan (2015, Oct). Model selection in multivariate adaptive regression splines (mars) using information complexity as the fitness function. *Machine Learning 101*(1), 35–58.

Kock, A. B. and T. Teräsvirta (2016). Forecasting macroeconomic variables using neural network models and three automated model selection techniques. *Econometric Reviews 35*(8-10), 1753–1779.

Korobilis, D. (2017). Quantile regression forecasts of inflation under model uncertainty. *International Journal of Forecasting 33*(1), 11 – 20.

Liaw, A. and M. Wiener (2002). Classification and regression by randomforest. *R News 2*(3), 18–22.

Lin, C.-S., H. A. Khan, R.-Y. Chang, and Y.-C. Wang (2008). A new approach to modeling early warning systems for currency crises: Can a machine-learning fuzzy expert system predict the currency crises effectively? *Journal of International Money and Finance 27*(7), 1098 – 1121.

Lopes, N. and B. Ribeiro (2014). *Machine Learning for Adaptive Many-Core Machines - A Practical Approach*. Studies in Big Data. Springer International Publishing.

Mann, H. B. and A. Wald (1943). On the statistical treatment of linear stochastic difference equations. *Econometrica, Journal of the Econometric Society*, 173–220.

McCracken, M. W. and S. Ng (2016). Fred-md: A monthly database for macroeconomic research. *Journal of Business & Economic Statistics 34*(4), 574–589.

Milborrow, S. (2011). *earth: Multivariate Adaptive Regression Splines*. R package.

Moshiri, S. and N. Cameron (2000). Neural network versus econometric models in forecasting inflation. *Journal of Forecasting 19*(3), 201–217.

Nakamura, E. (2005). Inflation forecasting using a neural network. *Economics Letters 86*(3), 373 – 378.

Rounaghi, M., M. Abbaszadeh, and M. Arashi (2015, 10). Stock price forecasting for companies listed on tehran stock exchange using multivariate adaptive regression splines model and semi-parametric splines technique. *438*.

Stock, J. H. and M. Watson (2007). Why has u.s. inflation become harder to forecast? *Journal of Money, Credit and Banking 39*(s1), 3–33.

Stock, J. H. and M. W. Watson (1999). Forecasting inflation. *Journal of Monetary Economics 44*(2), 293 – 335.

Stock, J. H. and M. W. Watson (2008, September). Phillips curve inflation forecasts. Working Paper 14322, National Bureau of Economic Research.

Szafranek, K. (2017). Bagged artificial neural networks in forecasting inflation: An extensive comparison with current modelling frameworks. Technical report.

Therneau, T., B. Atkinson, and B. Ripley (2017). *rpart: Recursive Partitioning and Regression Trees*. R package version 4.1-11.

Tibshirani, T. H. . R. (2016). *mda: Mixture and Flexible Discriminant Analysis*. R package version 0.4-9.

Trapletti, A. and K. Hornik (2017). *tseries: Time Series Analysis and Computational Finance*. R package version 0.10-41.

Non-exclusive licence to reproduce thesis and make thesis public

I, Kateryna Kolomiiets (date of birth: 08.07.1994),

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

   1.1. reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright, and

   1.2. make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives until expiry of the term of validity of the copyright, "Forecasting US inflation using machine learning methods", supervised by M. Hakan Eratalay and Aleksei Netšunajev.

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 24.05.2018