Irving Fisher Committee on
Central Bank Statistics

BIS

IFC-Bank of Italy Workshop on "Machine learning in central banking"

19-22 October 2021, Rome / virtual event

Machine learning real-time CPI forecasting[1]

Mariam Mamedli,
National Research University, Higher School of Economics, Moscow

[1] This presentation was prepared for the Workshop. The views expressed are those of the author and do not necessarily reflect the views of the Bank of Italy, the BIS, the IFC or the central banks and other institutions represented at the event.

Forecasting Russian CPI with data Vintages and Machine learning techniques[1]

Mariam Mamedli
National Research University Higher School of Economics
Email: mmamedli@hse.ru

Denis Shibitov
Bank of Russia
Email: ShibitovDS@mail.cbr.ru

Abstract

We show, how the forecasting performance of models varies, when certain inaccuracies in the pseudo real-time experiment take place. We consider the case of Russian CPI forecasting and estimate several models on not seasonally adjusted data vintages. Particular attention is paid to the availability of the variables at the moment of forecast: we take into account the release timing of the series and the corresponding release delays, in order to reconstruct the forecasting in real-time. In the series of experiments, we quantify how each of these issues affect the out-of-sample error. We illustrate, that the neglect of the release timing generally lowers the errors. The same is true for the use of seasonally adjusted data. The impact of the data vintages depends on the model and forecasting period. The overall effect of all three inaccuracies varies from 8% to 17% depending on the forecasting horizon. This means, that the actual forecasting error can be significantly underestimated, when inaccurate pseudo real-time experiment is run. We underline the need to take these aspects into account, when the real-time forecasting is considered.

Kew words: inflation, pseudo real-time forecasting, data vintages, machine learning, neural networks.

JEL-classification: C14, C45, C51, C53.

[1] The views expressed in this paper are solely those of the authors and do not necessarily reflect the official position of the Bank of Russia. The Bank of Russia assumes no responsibility for the contents of the paper. Any reproduction of these materials is permitted only with the express consent of the authors.
The authors are grateful to Sergei Seleznev (Bank of Russia), Dmitry Gornostaev (Bank of Russia), participants of 27th Annual Symposium of the Society for Nonlinear Dynamics and Econometrics at FRB Dallas (2019) and the participants of seminars at the Bank of Russia for their helpful comments.

Table of Contents

Introduction

Although it is common in empirical macroeconomics to work with the revised data, a growing body of literature suggests, that analysis using real-time data often leads to the substantially different conclusions, than the work which ignores data revisions (Croushore, Stark, 2001; Orphanides, 2001; Koenig, 2003; Molodtsova et al., 2008). More and more evidence indicate the importance of using real-time data, when constructing forecasting models and performing monetary policy analysis (Fernandez et al., 2011).

In this research, on basis of the unique for Russia real-time dataset we aim to define, how different inaccuracies during pseudo-real time experiments affect the estimates of the model performance. Namely, we focus on the use of not-seasonally adjusted vintages of timeseries (data which include initial values and revisions), consider it as the data for an "ideal" experiment and consequently add one of the inaccuracies to define the impact, each of them has on the estimation of out-of-sample model performance (linear regression, random forest, gradient boosting, neural networks). Our focus is not on the choice of the optimal modelling techniques, but on constructing real-time forecasting exercises.

The ability to forecast accurately inflation is a crucial for the development of monetary policy by the central bank, that is why we consider the CPI forecasting and investigate, how these aspects can affect the CPI forecasting errors. Within this task, we construct the forecasts for four sets of data in order to define an impact of several inaccuracies of pseudo-real-time forecasting. In the *first estimation experiment* we run our models on the vintages of original, not seasonally adjusted, timeseries, taking into account the timing of the data releases. By the *second experiment* we determine, how the results would change, if we include all data without any regard to the data availability, when the forecasts are made. This aspect is important, because macroeconomic and financial data usually have different release lags. However, often macroeconomic forecasting is based on the data, that include all available up to that point timeseries. If the forecasts were constructed in real time, some of these series may not have been yet released. By comparing the first and the second experiment we can quantify the effect of data release timing. By the *third experiment* we define the role of data vintages: how models' forecasting performance would change, if we neglect data revisions and take the final values, which in reality are not available, when the forecast is made. In the *last experiment* the forecasts are based on seasonally adjusted data without vintages and not taking into account the release dates. Seasonal adjustment is a subjective procedure, which leads to the analysis of unobservable series, highly dependent on the applied method of seasonal adjustment. Moreover, the use of seasonally adjusted data complicates the comparison between research results, while not seasonally adjusted data allow to compare the results in terms of observable variables, making the forecasting with not seasonally adjusted data more correct. Nevertheless, the combination of these data characteristics (seasonally adjusted revised data and the neglect of the data availability) often occurs in the forecasting literature, underlining the need to define their contribution to the forecasting error.

Main results of this research come from the comparison of the results between the experiments. *First,* we show that the estimates based on the data, which incorporate the series with different lags (when we take into account the release timing of series) generally have higher out-of-sample forecasting error, in comparison to the case, when all the series are treated as available at the time of forecast (for 8 out of 12 cases). This is an expected result, since in this case for some timeseries we use less recent data: when the forecasts are made, vintages are available with two months lag. This difference amounts to 10,5% on average for all four models for one month in advance (8,0% and 1,7% for three and six lags) *Second,* we compare estimation results based on the vintages and the data after revisions. In this case, the results are inconclusive, and do not point in favor of one experiment or the other. Some models and forecasting horizons are more sensitive to the use of data vintages and have lower forecasting error, some do not (in 7 out of 12 cases forecasting on revised data has an advantage). *Finally,* the last comparison shows, that the use of seasonal adjustment leads to significantly lower forecasting errors for most of the models and forecasting horizons. Overall, the joint effect of all three inaccuracies varies from 7,6% to 16,8% of average monthly CPI depending on the forecasting horizon.[2] This may result in the considerable underestimation of the error in the case of real-time forecasting: when part of timeseries is not published yet (different timing of releases), for some of them only preliminary numbers are available (data vintages) and forecasts are based on the seasonally adjusted data, which lower the error further.

In addition to the error decomposition depending on certain inaccuracies in prediction exercises, we contribute to the literature on Russian CPI forecasting in the following way:

This is the first research, which uses Russian data vintages in economic forecasting.[3]

We propose first reproducible real-time benchmark for Russian inflation forecasting.

In our research we consider a variety of popular machine learning models (including ensemble methods and Bayesian neural networks).

Although, the methodology of the research is relatively new for the forecasting on the Russian data, the application of machine learning (ML) techniques is known to have a great potential for macroeconomic forecasting. Short overview of the related literature on the application of ML models in economic forecasting is presented in the following section.

Alternative, traditional for macroeconomic research approaches to the forecasting in data-rich environment include pooling or averaging of bi-variate forecasts (Stock, Watson (2003); Rossi, Sekhposyan, 2010) and direct pooling of information using a high-dimensional models (see Forni et al. (2000, 2005); Stock, Watson (2002a, b) for DFM, Banbura et al. (2010) for Bayesian VAR).[4] However, an analysis of a wide range of models is out of the scope of this paper, so we do not cover these methods in details here.

The paper is organized as follows. Next section gives an overview of the relevant research on analysis on real-time data and examples of ML applications in economic forecasting. Third section provides an overview of the considered experiments and details on the data. Next cross-validation, optimization procedure and models, applied to the CPI forecasting, are described. Then we provide the estimation results and show the impact of each data characteristic to the out-of-sample forecasting error. The last section concludes.

2. Related research

2.1. Real-time data

Croushore (2011) provides an overview of the existing research on real-time data analysis, dividing it into six areas: data revisions, structural macroeconomic modelling, forecasting, monetary policy, current analysis and revisions to conceptual variables. The research shows that real-time data matter in a variety of contexts. Overall, the results show, that the forecasting ability in real time is much worse, than the forecasting ability,

[2] Average monthly CPI was calculated on the basis of data included in the test set.
[3] The details on the data are provided in Ponomarenko et al. (2021).
[4] Among the recent research on the forecasting of Russian CPI one may name Styrin (2019), where the CPI dynamics is predicted using dynamic model averaging.

when the revised data are used. Moreover, forecasts of levels are very sensitive to data revisions, whereas forecasts of growth rates are much less sensitive (Howrey, 1996). Filardo (1999) shows, how unreliable in real time are models, which attempt to predict recessions, as these models are usually based on revised data. As for the inflation, Koenig (2003) shows, that while markup is a useful predictor of inflation with revised data it fails to predict inflation in real-time. Orphanides, Norden (2005) illustrate, that in real-time the estimation of output gaps is so much affected by uncertainty, that they cannot be reliably used in inflation forecasting. Forecasts of exchange rates are known to be even more sensitive to real-time data issues (Faust et al., 2003; Molodtsova, 2008; Molodtsova et al., 2008).

The question of whether the use of real-time data leads to the different forecasts, than when the latest-available data are used goes back to the first paper by Denton, Kuiper (1965), who on the case of Canada find, that the use of real-time data or latest-available data leads to the significant differences in the forecasts. Cole (1969) also shows, that data errors can reduce forecast efficiency and lead to biased forecasts, as the result, there can be significant differences between forecasts made with different data sets. Similar results were obtained by Trivellato, Rettore (1986), who showed on Italian data, that data errors in a simultaneous-equations model have large effects.

Series of papers afterwards advocated in favor of using real-time data and bring attention to the consequences of using real-time data as opposed to the latest available. Stark, Croushore (2002) discuss, how the forecasts are affected by the use of real-time data rather than latest-available data. They bring attention to the fact, that in the forecasting literature the results are usually obtained using the data set available to the model's developer, but the data would not have been available to him in real-time. They investigate, how the vintage data affect such forecasts and advocate in favour of real-time data rather than latest-available data. Forecasts made for a particular date can be quite different, depending on the vintage of data used: the RMSEs and MAEs of forecasts can differ between real-time data and latest-available data, when only short spans of observations are used, and been misleadingly low when latest-available data are used. They also find, that inflation forecasts are more sensitive to the choice between real-time and latest-available data, than real output forecasts. Stark, Croushore (2002) also show that the choice of lag length depends on whether latest-available data or real-time data are used.

Croushore, Stark (2003) examine the nature of data revisions and investigate the robustness of the results of several key papers in macroeconomics (Kydland, Prescott, 1990; Hall, 1978; Blanchard, Quah, 1989) to different vintages. They show, that only the results of the former paper are robust to the use of different data vintages, underlining an importance of real-time data.

Koenig et al. (2003) argue, that analysts should generally use data of as many different vintages as there are dates in the samples. More specifically, at every date within a sample, right-hand-side variables ought to be measured as they would have been at that time (so called "real-time-vintage data"). They consider three different ways of using real-time data (depending on whether the real-time-vintage data on the left-hand side are used) and show on the example of GDP forecasting that out-of-sample forecasting performance of the model estimated using real-time-vintage data is superior to the one, obtained using conventional estimation. They advocate, that the most popular approach of using end-of-sample-vintage data (estimation strategy 3 in our case) should generally be avoided. Kishor, Koenig (2012) present a method of adopting VAR analysis to account for data revisions. They apply the technique to employment and unemployment rate, real GDP and the GDP/consumption ratio and show, that in each case the proposed procedure outperforms the conventional VAR analysis.

Clements, Galvao (2009) provide evidence in favor of real-time vintage data within MIDAS model. In the following paper Clements, Galvao (2011) show, that a certain class of models can be used to forecast 'fully revised' or 'post-revision' values of past and future observations, estimating the value of those forecasts in terms of their contribution to improving real-time estimates of the output gap, trend inflation and inflation gap. Their research was followed by Clements, Galvao (2013), who conduct an evaluation of vintage-based VAR model forecasts for US inflation and output of a range of maturities of data using a variety of different target variables (forecasting future observations or revisions to past data). They show, that VAR models on a single variable estimated on data vintages can be successful in forecasting the data revisions process of inflation, but are less useful for US output growth. They also find only some evidence, that vintage-based VAR models provide more accurate forecasts of output growth, but clear evidence, that revisions to past inflation data are predictable. In the case of predicting Russian CPI, revisions can play a role only via explanatory variables since CPI series are not subject to any revisions.

Being a brand-new practice for Russia, real-time datasets are more common in other countries. Croushore, Stark (2001) compiled and analyzed a large real-time dataset for macroeconomists on the US economy starting from 1965, bringing the attention to this subject. Later McCraken, Ng (2016) formed a big database for macroeconomic research, updated in real-time through Federal Reserve Economic Data (FRED) database. The data include historical vintages from august 1999 and is widely used in the literature.

There exist several datasets with vintages on European economies as well. Giannone et al. (2012) presented a real-time database for the euro area. Fernandez et al. (2011) introduce a new international real-time dataset on OECD countries and illustrate the importance of using real-time data in macroeconomic analysis by considering several economic applications conducted in real-time perspective on the data on G7 economies. Being one of the first multicounty real-time datasets Fernandez et al. (2011) contributed to the papers with datasets on individual countries. Some examples of the data vintages on other countries include Egginton et al. (2002), who presented a real-time macro dataset for the UK, Clausen, Meier (2005); Sauer, Sturm (2007) and Gerberding et al. (2005), who collected the real-time data for Germany, and Nikolsko-Rzhevskyy (2011), who proposed a methodology of estimation forward-looking Taylor rules in real-time and illustrated it on the example of UK, Germany and Canada.

2.2. Applications of machine learning in economic forecasting

Although been relatively new in application to the Russian data, ML models are widely used in international economic research. Tiffin (2016) along with Chakraborty, Joseph (2017) and Kapetanios, Papailias (2018) outline the potential of ML for central banking and policy analysis and provide a broad overview of the key ML techniques, their advantages and limitations. The comparison of forecasts of the standard and ML models provides evidence in favour of ML models (e.g. Cook, Hallyz, 2017). There is an evidence in favour of ML techniques, especially neural networks (NNs), in forecasting the CPI in many other countries (Moshiri, Cameron, 2000; Chen et al., 2001; Szafranek, 2019; Hanif et al., 2018). The performance of NNs in forecasting inflation was also evaluated within a cross-country comparison with the results in favor of ML models (McAdam, McNelis, 2005; Choudhary, Haider, 2012). At the same time, there is opposite evidence, that NNs in fact does not outperform the standard models (Kock, Tersvirta, 2013; Catik, Karauka, 2012; Ivarez-Daz, Gupta, 2015; Sermpinis et al., 2014; Zhang, Li, 2012).

There is less evidence of applying boosting techniques in the CPI forecasting. However, Buchen, Wohlrabe (2011) apply it to the forecasting US industrial production, showing that boosting can be a serious competitor to other methods in the short run and that it performs best in the long run. Dpke et al. (2017) show that boosting has a better out-of-sample performance, than probit models for the recessions prediction in Germany.

To the best of our knowledge, this is the first research on forecasting abilities of neural networks, gradient boosting and random forest with optimal architectures in the case of Russian CPI. The closest to this research is Baybuza (2018), who shows, that random forest and gradient boosting can outperform standard methods on the horizon of two and more months in forecasting Russian CPI. However, he does not choose optimal hyperparameters, considering the predetermined number of estimators. We consider data vintages and illustrate the impact of the data used on ML models with optimal architectures.

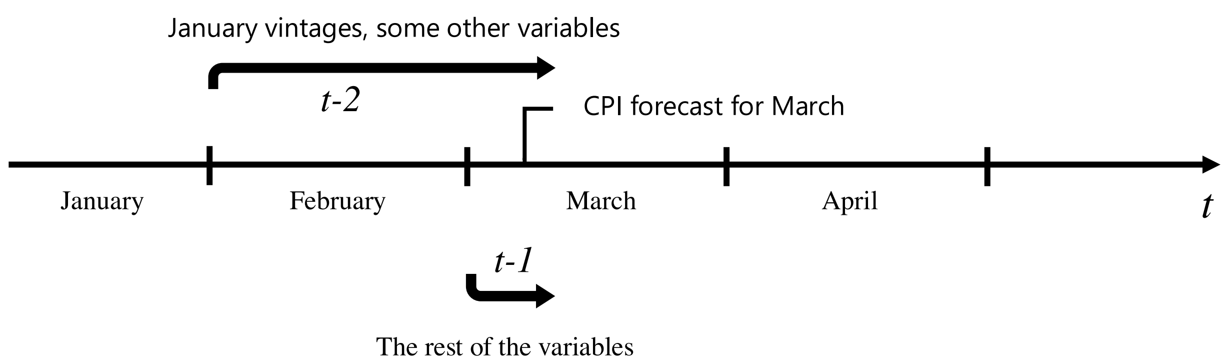
Taking into account, that the results from the other countries are inconclusive with respect to the forecasting abilities of these models, our research, along with its main goal, fills this gap by analysing the performance of ML models in application to the Russian CPI.

3. Experiments and data

In our key estimation experiment we use vintages of official data releases, described in detail in Ponomarenko et al. (2021). The data cover the period from January 2001 to July 2019 and consist of timeseries of macroeconomic variables for each release date, including all revisions. At each point in time, the forecast is made on the basis of the latest (not revised) data. The data include 27 time series on economic activity, interest rates, price indexes, international trade and others. For most of the variables the vintages of data are used: timeseries at each date, as they were initially published by the Federal statistical service. Table A1 in Appendix provides an overview of series included in the dataset.[5] For experiments with not seasonally adjusted data we include months dummies.

We consider the forecasts for four datasets in order to define the impact of main data characteristics. In the *first experiment* we estimate models on the vintages of not seasonally adjusted timeseries, taking into account the timing of the data releases (they are presented in Table A1). In each experiment we assume, that we construct our forecasts at the beginning of each month, when the data on vintages for two months ago are released. By this moment the CPI data and most of the financial variables for the previous month are already released (Figure 1). Therefore, at the moment of forecast we have part of the data for the previous month ($t-1$ lag) and the rest of the data with two months delay ($t-2$ lag). In order to reconstruct properly the real-time forecasting procedure this difference in the data for the latest available dates should be taken into account.

Figure 1. Example of the timeline of data releases



[5] Data sources are Federal State Statistical Service, Central Bank of Russia, EIA, Roskazna.

In the *second experiment* we include timeseries without the regard to the data availability at the moment of forecast: all variables are taken with one month lag. In the *third experiment* we estimate models on the revised data and define, how the use of data vintages affects the forecasting performance. In the *last experiment* the forecasts are based on seasonally adjusted data without vintages and not taking into account the release dates.[6] We advocate for the forecasting of not seasonally adjusted CPI, first, because it is the commonly recognized indicator, as opposed to the seasonally adjusted data, which is the result of the adjustment procedure, which may differ from one research to the other. Second, the original CPI values are not revised, while in the case of seasonal adjustment historical values are revised each time, which may alter the estimations in the unknown manner. The last experiment illustrates the joint effect of all three inaccuracies. Table 1 provides an overview of the experiments with the references to corresponding subsections.

Table 1. Different characteristics of the experiments

Experiment	Starting lags	Vintages	Seasonality
1	$t-1/t-2$	vintages	NSA
2	$t-1$	vintages	NSA
3	$t-1$	regular	NSA
4	$t-1$	regular	SA

Note: yellow – subsection 5.2, blue – subsection 5.3, green – subsection 5.4.

4. Models and estimation procedure

4.1 Cross-validation and estimation procedure

In order to choose the optimal model architecture, we apply the cross-validation procedure. We split the dataset in train and test subsets: train set includes observations up to December of 2012, following observations are included in the test set. On the train set we apply cross-validation with an expanding window.[7] Starting from a minimum size window of 48 observations we train each model and test it on the next observation from the train set.[8] We consequently add next observation to the training data and re-estimate the model. We evaluate the out-of-sample model performance on the test set.

In order to define the optimal architectures of the models, we use Bayesian optimization instead of the grid search. The parameters are chosen so to minimize RMSE on the training set. This procedure allows us to decrease significantly the estimation time (roughly six times faster). Moreover, preliminary results, based on the cross-validation with grid-search, show that Bayesian optimization not only is a faster joint optimization tool, but allows us to define more optimal in terms of out-of-sample RMSE parameter combinations due to the search on continuous intervals instead of fixed combinations. For details on Bayesian optimization see Shahriari et al. (2015).

[6] The seasonal adjustment was conducted in Demetra program with tramoseats method and rsa3 specification.

[7] For the first experiments we have compared the results for both, expanding and rolling window. RMSE levels were lower in the case of
expanding window for all models and both experiments.

[8] The size of minimum window was chosen experimentally.

Each of four models (linear regression, random forest, gradient boosting, and Bayesian neural network) are applied to forecast Russian CPI for one, three and six months in advance. For the last two horizons we predict the accumulated inflation (CPI for three months and for six months in advance).

We estimate each model for each forecasting horizon on the dataset with one, three or six lags. This brings the total number of estimations for each experiment to 36. Overall, 144 estimations were made. Next subsections briefly describe the estimated models.

4.2. Regression with regularization

As a linear model we consider a regression with regularization (elastic net), which is advised, when features are correlated with each other (Friedman et al., 2010). This model combines ridge regularization penalty and Lasso penalty. Via cross-validation we choose the optimal type of regularization as well as other optimization parameters.

Formally it is defined as follows:

$$\min_{w} \frac{1}{2N}\left(\|Xw - y\|_2^2 + \alpha\rho\|w\|_{l_1} + \frac{\rho(1-\alpha)}{2}\|w\|_{l_2}^2\right), \tag{1}$$

where y is a target variable, X is a vector of predictors, β is vector of coefficients and N is the number of observations.

4.3. Random forest

Random forest is an ensemble model proposed by Breiman (2001).[9] The algorithm of random forest is based on decision trees. Each tree is a graph model, which consists of a set of rules on explanatory variables to obtain the target variable. This model has a tree structure with nodes as decision points. The split occurs according to a certain criterion on one of the explanatory variables, while terminal nodes (leaves) contain the value of the target variable. The decision tree is built in a stepwise manner: first, the sample is split into two subsamples according to the specified criterion, then each of subsamples is consequently split further, until a certain stop criterion is not reached.[10]

The random forest model can be expressed as follows. Suppose we have N observations (x_i, y_i) for $i = 1,2,\dots,N$ with $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})$ and M regions $R_1, R_2, \dots, R_M$. The response is modeled as a constant c_m in each region:

$$f(x) = \sum_{m=1}^{M} c_m I(x \in R_m). \tag{2}$$

We consider a splitting variable j and split point s and define the pair of half-planes:

$R_1(j,s) = \{X | X_j \le s\}$ and $R_2(j,s) = \{X | X_j > s\}$.

[9] For the estimation we use scikit-learn Python package.

[10] Algorithm is presented in detail in Friedman et al. (2009), p. 308.

Then the splitting variable *j* and the split point *s* are defined via minimization problem with the chosen minimization criterion. After the best split is defined, the data are divided accordingly and the splitting process is repeated for these two regions. The procedure is repeated on all of the resulting regions.

Via cross-validation we choose the architecture, which ensures the best forecasting performance of the models. Considered parameters are presented in Table A2 in Appendix.

4.4. Gradient boosting

Gradient boosting is another ML algorithm based on the combination of predictive models, decision trees in our case, so to minimize the loss function.[11] In this form it was proposed by Friedman (2001). Gradient boosting can be used both for classification and regression, in our case the problem belongs to the second type. Boosting allows to identify outliers and to exclude them from the training set. However, it is known to have a tendency to overfit, while the stepwise approach of this algorithm can lead to a non-optimal set of weak learners. That is why it is highly important to choose optimally the combination of the number of estimators and learning rate, which can help to avoid overfitting.

Analytically gradient boosting is expressed as an additive sum of simpler models:[12]

$$F(x) = \sum_{m=1}^{M} \gamma_m h_m(x), \tag{3}$$

where $h_m(x)$ are decision trees, and $\gamma_m(x)$ is a step length.

Gradient boosting is built in a stepwise manner in the following way:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x). \tag{4}$$

On each step the decision tree $h_m(x)$ is chosen optimally from the minimization of the loss function L with a given F_{m-1} and $F_{m-1}(x_i)$:

$$h_m(x) = \underset{h,\beta}{\operatorname{argmin}} \sum_{i=1} L\big(y_i, F_{m-1}(x_i) + \beta h(x_i)\big). \tag{5}$$

The minimization problem is approximately solved via fitting steepest descent directions (negative gradient of the loss function evaluated at the current model F_{m-1}) by new weak learner. The step length $\gamma_m(x)$ is chosen according to the equation (6):

$$\gamma_m(x) = \underset{h,\beta}{\operatorname{argmin}} \sum_{i=1} L\big(y_i, F_{m-1}(x_i) - \gamma h_m(x)\big). \tag{6}$$

When defining the optimal architecture, we consider different model characteristics, including the specification of loss function, maximum depth of a tree and the number of trees. The complete set of hyperparameters is presented in Table A3 in Appendix.

---

[11] In the baseline case it is a least squares regression.

[12] For the estimation we use scikit-learn Python package.

4.5. Bayesian neural networks

We consider a sparse Bayesian neural network model described in Khabibullin, Seleznev (2020).[13] The model consists of two hidden layers with 30 and 10 nodes. Analytically the model can be presented as follows:

$$h_{1,t} = f(W_1 x_t + b_1), \tag{7}$$

$$h_{2,t} = f\left(W_2 h_{1,t} + b_2\right), \tag{8}$$

$$y_t = W_3 h_{2,t} + b_{y,t} + \varepsilon_t, \tag{9}$$

where x_t are input data, W_i are weights on the layers $i = 1, 2$ and output layer, b_i and b_y are bias, $f(\cdot)$ is an activation function, h_i is the output of the hidden layer i, y_t is the output of the neural network at time t and ε_t is an error term. We consider two model specifications: Normal distribution of the error term and Tanh activation function and tStudent distribution with ReLu activation function.[14]

5. Results

We estimate elastic net, random forest, gradient boosting and Bayesian neural network on four different datasets. For each dataset we construct forecasts for one, three and six months in advance. Along with the optimal model architectures we consider the different numbers of lags for each horizon. In the next subsection we provide the results for the benchmark experiment (experiment 1), conducted on the vintages of not seasonally adjusted time series, where the release dates are taken into account. In the following subsections it is compared to the results of the experiments. In the second subsection the role of release timing is studied (experiment 1 and 2, 'starting lags' in Table 1). Next, we study, how the estimation with the vintages can affect model performance (experiment 2 and 3, 'vintages' category). In the last subsection we show, how the estimates are affected by the use of revised and seasonally adjusted data (experiment 3 and 4, 'seasonality').

5.1. Benchmark experiment

We start by providing the results for a set of considered models, estimated on the data vintages of not seasonally adjusted data. We also take into account the release dates of timeseries: the official data are released with a delay and different lags. In our case, the publications of vintages have a one-month delay comparing to the financial variables and the CPI, which are not revised. In order to replicate a real-time forecasting experiment, we take these publication lags into account and include lagged variables starting in different point in time: exchange rates, oil price and other financial variables are included in the dataset starting from $t - 1$ (previous month), for vintages data first observations correspond to two months lag ($t - 2$). The details on the data and the delay of first observations are provided in Table A1 in Appendix. In the

[13] The mean-field approximation was used.

[14] Initially for the first experiment all four the combinations of model parameters were considered. These two combinations were chosen as they allow to achieve the lower RMSE level on cross-validation.

cases of three and six months lags we include the equal number of lags starting from $t-1$ or $t-2$ for corresponding series.

Table 2 provides RMSEs, estimated on vintages of not seasonally adjusted timeseries, when the release dates are taken into account (experiment 1 in Table 1). RMSE levels on cross-validation and test for all experiments are provided in Appendix (Tables A7-A10). The results are provided for combination of hyperparameters, which allows to achieve the lowest RMSEs for each forecasting horizon and number of lags (Tables A4-A6, Appendix).

Table 2. The out-of-sample RMSE levels for different models

lags	Elastic Net	Random Forest	Gradient Boosting	Neural network	AR
1 month					
1	0,480	0,368	0,356	0,409	0,404
3	0,403	0,373	0,382	0,431	0,427
6	0,380	0,389	0,396	0,400	0,392
3 months					
1	0,756	0,826	0,820	0,705	1,414
3	0,760	0,883	0,759	0,728	1,393
6	0,789	0,895	0,737	0,860	1,390
6 months					
1	0,758	0,963	0,789	0,687	2,693
3	0,643	0,932	0,716	0,752	2,681
6	0,758	0,986	0,842	0,876	2,664

Note: AR model was estimated with seasonal dummies for proper model comparison.
The lowest RMSE levels for each forecasting horizon are marked with grey.

The results suggest that forecasts on the basis of the gradient boosting model have lower out-of-sample RMSE in the forecasting for one month in advance, random forest forecast is the second best, with both models performing better than AR.[15] For three months neural network forecast has the lowest RMSE, followed by elastic net model. For six month in advance elastic net has the lowest error. In forecasting for three and six months all ML models outperform the AR benchmark, which predicts poorly on these horizons.

Higher AR errors on these horizons are explained by its univariate nature: other models include different macroeconomic variables, informative in predicting future economic developments. Let us consider an elastic net, as more interpretable one among ML models, and a forecasting period of three months with one lag. Additional estimations suggest, that elastic net trained on the whole train set, with optimal hyperparameters “picks” non zero coefficients for variables such as export and import, exchange rate, oil price, production of eggs and meat as well as retail of nonfood goods and others while seasonal dummies have zero or close to zero coefficients. Additional variables capture some seasonal fluctuations and mitigate the autoregressive spikes in CPI if needed. In the case of AR with seasonal dummies this cherry picking is impossible, which makes AR forecasts excessively volatile, when longer horizons are considered due to the change in the seasonal fluctuations of the CPI on the test set with comparison to the train set (see Figure 1 in Appendix).

[15] The models outperform RW benchmark as well, the results of which are omitted in the table with the aim of comparability. For 1, 3 and
6 months in advance the RMSE for RW is 0.41, 1.76 and 3.81, correspondingly.

5.2. The role of release timing

Next, we consider, how taking into account the release timing affects the results. We examine two experiments with not seasonally adjusted data vintages. In the first one we form dataset with the regard to the time, when the latest value of an indicator is released ($t-1/t-2$). In the second experiment we treat all timeseries as available and include them in the dataset with $t-1$ lag. The results for these experiments are provided in Table 3.

Unsurprisingly, for most of the cases an assumption, that the most recent data are available, leads to the lower forecasting errors (for 10,5%, 8% and 1,7% on average for one, three and six months in advance).[16] The exceptions are gradient boosting and random forest forecasts for one month in advance and elastic net forecast for six months in advance. Since usually linear models are used for macroeconomic forecasting the results for elastic net can be more representative: for this model an inclusion of all "available" data can lead to an underestimation of the real forecasting RMSE from 5,5% to 10,6% for one and three months in advance, correspondingly.

Table 3. Comparison of RMSEs of models estimated on data with the same and different lags

	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	$t-1$	$t-1/t-2$	$t-1$	$t-1/t-2$	$t-1$	$t-1/t-2$	$t-1$	$t-1/t-2$
1 month								
1	**0,359**	0,480	0,393	**0,368**	0,378	**0,356**	**0,338**	0,409
3	0,362	0,403	0,446	0,373	0,404	0,382	0,349	0,431
6	0,447	0,380	0,428	0,389	0,375	0,396	0,411	0,400
3 months								
1	**0,676**	0,756	**0,772**	0,826	**0,710**	0,820	**0,626**	0,705
3	0,697	0,760	0,816	0,883	0,789	0,759	0,787	0,728
6	0,720	0,789	0,814	0,895	0,790	0,737	0,765	0,860
6 months								
1	0,722	0,758	0,914	0,963	0,735	0,789	**0,685**	0,687
3	0,701	**0,643**	**0,903**	0,932	**0,716**	0,716	0,698	0,752
6	0,743	0,758	0,927	0,986	0,809	0,842	0,916	0,876

Note: The lowest RMSE levels for each forecasting horizon and number of lags are marked with grey.
The lowest RMSE among two experiments for each model is marked with bold.

5.3. The role of data vintages

We can define the role of data vintages in macroeconomic forecasting by comparing the estimates based on not seasonally adjusted data vintages and model results estimated on not seasonally adjusted data with final releases (experiments 2 and 3 in Table 1). The RMSE levels on test sets for four models are presented in Table 4.

For one and three months in advance gradient boosting and random forest provide lower errors, when vintages are not used. For elastic net and neural network, on the contrary, for these forecasting horizons the use of data vintages allows to achieve lower errors. For longer horizon of six months the neural network

[16] Estimations were made for each model with the optimal number of lags. The change is calculated as the difference between 2nd and 1st experiment's RMSEs relative to the latter one.

estimated on the data without vintages provides the lowest RMSE with the significant difference comparing to its results on vintages data. Elastic net and random forest also provide lower errors, when the vintages data are not used, yet the difference between errors is lower and insignificant in the latter case. For gradient boosting the use of vintage data leads to lower RMSE.

Table 4. Comparison of RMSEs of models estimated with and without data vintages

	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	vintages	no vintages	vintages	no vintages	vintages	no vintages	vintages	no vintages
1 month								
1	**0,359**	0,369	0,393	**0,373**	0,378	**0,334**	**0,338**	0,348
3	0,362	0,399	0,446	0,386	0,404	0,366	0,349	0,402
6	0,447	0,392	0,428	0,399	0,375	0,392	0,411	0,414
3 months								
1	**0,676**	0,697	0,772	**0,732**	0,710	**0,677**	0,626	**0,615**
3	0,697	0,711	0,816	0,824	0,789	0,803	0,787	0,766
6	0,720	0,751	0,814	0,881	0,790	0,848	0,765	0,822
6 months								
1	0,722	0,767	0,914	**0,901**	0,735	0,767	0,685	0,669
3	0,701	**0,690**	0,903	0,940	**0,716**	0,731	0,698	**0,657**
6	0,743	0,690	0,927	0,972	0,809	0,781	0,916	0,889

Note: The lowest RMSE levels for each forecasting horizon and number of lags are marked with grey.
The lowest RMSE among two experiments for each model is marked with bold.

Overall, the results suggest, that the sensitivity to the use of data vintages varies among the models and the considered forecasting horizons. In these two experiments, if the choice of a model with the lowest errors has to be made, neural network has the lowest or comparable out-of-sample errors in comparison to the other models. For the forecasts for one month in advance the use of vintages data leads to the lower error. For three and six months, on the contrary, the use of data vintages only increases the error.

Overall, in 19 cases out of 36 (for three forecasting horizons, three lags for each and four models) the use of data vintages leads to the lower RMSE. However, when the model with the optimal number of lags is considered in 8 out of 12 cases (four models and three forecasting horizons) the forecasts on the revised data provide lower RMSEs. Anyway, usually forecaster is not faced with the choice what data to use. In real time only preliminary releases and data published with delay are available. The comparison of these two estimation experiments shows, that in real-time forecasting RMSE levels can differ significantly from the estimates, obtained on the revised data. Depending on the model, number of lags used and forecasting horizon this impact may vary. However, the results suggest, that model performance on all available data is not representative in real-time estimations and should be considered with caution.

5.4. The role of seasonal adjustment

Next, we consider two experiments on data, which incorporate all revisions (no vintages). In the one experiment all variables with seasonal fluctuations are seasonally adjusted (experiment 4 in Table 1), while in the second one the variables are not altered and seasonal dummies are added in the dataset (experiment 3 in Table 1).[17] This comparison is aimed to show, how forecasting results depend on the use of unobservable data transformations and whether it can alter the forecasting error and the choice of the model. Table 5 summarizes the results.

In almost all cases the use of seasonally adjusted timeseries leads to the lower forecasting errors. The difference in the value errors varies from 12,1% for one month in advance to 8,5% and 9,3% for three and six months, correspondingly. The results suggest, that the use of seasonal adjustment in the CPI forecasting can lead to a significant underestimation of forecast errors.

Table 5. RMSEs estimated on seasonally adjusted and not seasonally adjusted data

	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	sa	nsa	sa	nsa	sa	nsa	sa	nsa
1 month								
1	0,348	0,369	**0,326**	0,373	**0,323**	0,334	**0,304**	0,348
3	**0,295**	0,399	0,346	0,386	0,344	0,366	0,347	0,402
6	0,377	0,392	0,353	0,399	0,344	0,392	0,391	0,414
3 months								
1	**0,622**	0,697	**0,686**	0,732	0,700	**0,677**	0,625	**0,615**
3	0,675	0,711	0,715	0,824	0,682	0,803	0,652	0,766
6	0,664	0,751	0,738	0,881	0,732	0,848	0,716	0,822
6 months								
1	0,666	0,767	**0,843**	0,901	**0,671**	0,767	0,837	0,669
3	**0,599**	0,690	0,846	0,940	0,791	0,731	0,804	**0,657**
6	0,658	0,690	0,880	0,972	0,827	0,781	0,921	0,889

Note: The lowest RMSE levels for each forecasting horizon and number of lags are marked with grey.
The lowest RMSE among two experiments for each model is marked with bold.

5.5. Joint effect of three discrepancies

Here we compare two 'corner' experiments: the benchmark one, when the estimates are based on vintages of not seasonally adjusted data and release timing is taken into account (experiment 1), and the last one, estimated on series with final revisions and seasonally adjusted data, treating all variables as available. Table 6 summarizes the results.

Table 6. Comparison of RMSEs of models the first and fourth experiment

	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	exp. 4	exp.1	exp. 4	exp.1	exp. 4	exp.1	exp. 4	exp.1
1 month								
1	0,348	0,480	**0,326**	0,368	**0,323**	0,356	**0,304**	0,409
3	**0,295**	0,403	0,346	0,373	0,344	0,382	0,347	0,431
6	0,377	0,380	0,353	0,389	0,344	0,396	0,391	0,400
3 months								

[17] The seasonal adjustment is conducted in Demetra program with tramoseats method and rsa3 specification.

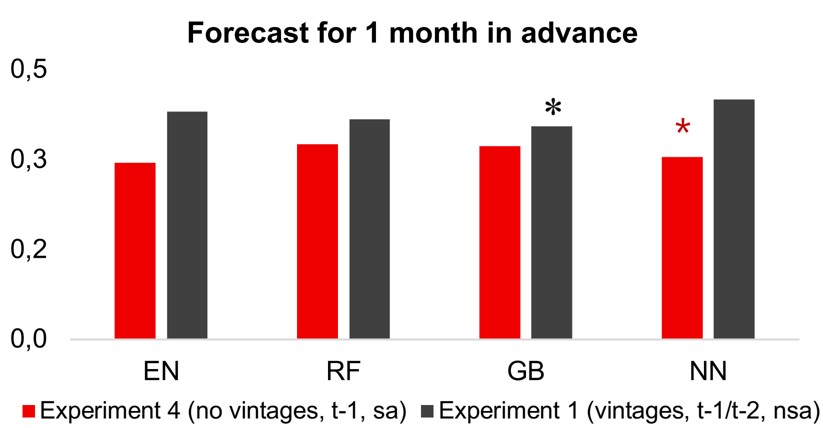
1	**0,622**	0,756	**0,686**	0,826	0,700	0,820	**0,625**	0,705
3	0,675	0,760	0,715	0,883	**0,682**	0,759	0,652	0,728
6	0,664	0,789	0,738	0,895	0,732	0,737	0,716	0,860
6 months								
1	0,666	0,758	**0,843**	0,963	**0,671**	0,789	0,837	**0,687**
3	**0,599**	0,643	0,846	0,932	0,791	0,716	0,804	0,752
6	0,658	0,758	0,880	0,986	0,827	0,842	0,921	0,876

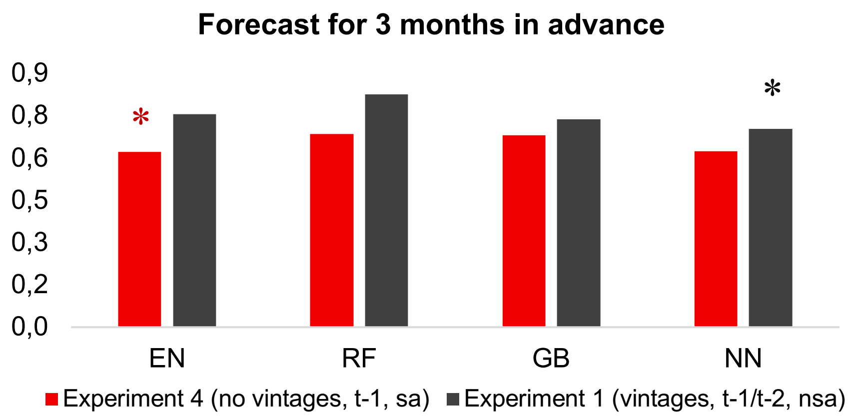
Note: Exp. 1 does not include vintages or take into account of release timing, with seasonal adjustment;
Exp.4 is based on vintages, in it we take into account release timing and use not seasonally adjusted data.
The lowest RMSE levels for each forecasting horizon and number of lags are marked with grey. The lowest RMSE among two experiments for each model is marked with bold.

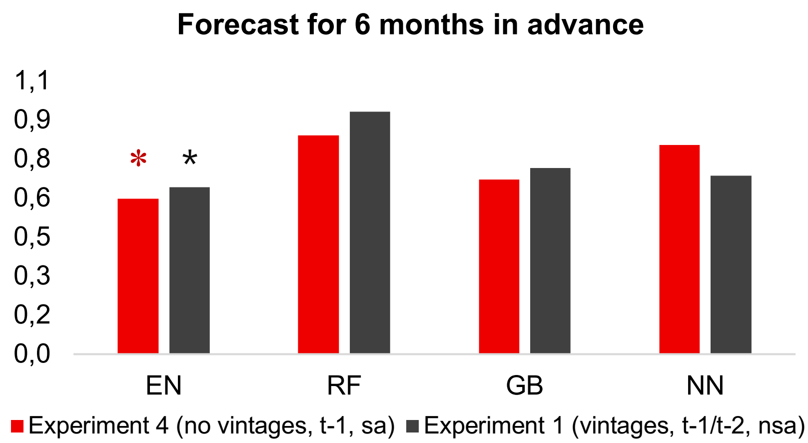
We see, that for all four models experiment 4, which incorporates all inaccuracies, has a lower forecasting error than the 1 experiment (32 out of 36 cases), which reconstructs the real time forecasting procedure, how it would have been at each point in the past.

If we consider the forecasts with the optimal for each horizon number of lags, the evidence is even more conclusive (Figure 2). The RMSE gap of the models within two experiments varies depending on the model and the forecasting horizon. On average the difference between the experiments is 16,8% for one month in advance, 13,4% for the three months and 7,6% for forecasts for six months. Moreover, the choice of the optimal model for two out of three horizons changes depending on the experiment, so these results may also affect the comparison of models, based on their performance. Therefore, neglecting these three inaccuracies in the actual real-time forecasting can lead to a significant underestimation of the forecasting error and the incorrect choice of the forecasting model.

Figure 2. Comparison of the RMSEs of the models with optimal number of lags for first and fourth experiment for different forecasting horizons







Conclusion

We construct four forecasting experiments in order to define, how different inaccuracies in pseudo real-time forecasting affect the performance of several machine learning models. We analyze these effects in the case of forecasting Russian CPI. As a benchmark for the comparison, we use the case, where the forecasts are based on the data with three main characteristics. First, we use data vintages, which is the first case of their use in the forecasting excises on Russian data. Second, in our real-time experiment we take into account, when each variable is released, fix at what part of the month we make our forecasts and include in the estimation only available data with the corresponding release delay. Finally, we do not apply any seasonal adjustment procedures, and include seasonal dummies in the dataset along with timeseries.

Main results come from the comparison of the benchmark experiment with others. First, we show, that the neglect of the release timing of series lead to the significant underestimation of the forecasting error. Second experiment provides inconclusive evidence concerning an impact of the use of vintages. For most of the models the estimation on the revised data leads to the lower errors. We show, that for some models the use of ordinary data can lead to an artificially low forecasting error. In reality, all we have at each point in time are preliminary data before any revisions occur. With the help of the last experiment we show, that the use of seasonally adjusted data lowers artificially the forecasting error. This means, that when a particular model is considered, lower forecasting errors can be misleading and partly be the result of the use of seasonally adjusted data. Finally, we compare the results of the benchmark experiment with the results of the estimations, conducted on revised time series (not vintages) of seasonally adjusted data and with no account for the difference in release dates. Overall effect of these three discrepancies is 16,8% for the best number of lags for each model for one month in advance (13,4% and 7,6% for three and six months, correspondingly). Therefore, these aspects could lead to a significant underestimation of forecasting error in actual real-time forecasting and should be taken into account in the macroeconomic forecasting.

Within the benchmark experiment we also compare the performance of ML models and show, that these models have a great potential in forecasting macroeconomic timeseries. For forecasting for one month in advance gradient boosting and neural network have the comparable RMSE, for three months in advance neural network has the best performance, for six months elastic net provides lower error. In all three cases AR forecasts are outperformed by these ML models.

References

1. Bańbura, M., Giannone, D., & Reichlin, L. (2010). Large Bayesian vector auto regressions. Journal of applied Econometrics, 25(1), 71-92.
2. Baybuza I. (2018). Inflation Forecasting Using Machine Learning Methods. Russian Journal of Money and Finance 77(4) pp. 4259. doi: 10.31477/rjmf.201804.42.
3. Blanchard, O. J., & Quah, D. (1989). The dynamic effects of aggregate demand and supply disturbances: Reply. The American Economic Review, 79, 655-673.
4. Buchen T. & Wohlrabe K. (2011). Forecasting with many predictors: Is boosting a viable alternative?. Economics Letters 113(1) 16-18.
5. Catik A. N. & Karauka M. (2012). A comparative analysis of alternative univariate time series models in forecasting Turkish inflation. Journal of Business Economics and Management 13(2) 275-293.
6. Chakraborty C. & Joseph A. (2017). Machine learning at central banks. Staff Working Paper No. 674 Bank of England
7. Chen X. Racine J. & Swanson N. R. (2001). Semiparametric ARX neural-network models with an application to forecasting inflation. IEEE Transactions on neural networks 12(4) 674-683.
8. Choudhary M. A. & Haider A. (2012). Neural network models for inflation forecasting: an appraisal. Applied Economics 44(20) 26312635.
9. Clausen, J. R., & Meier, C. P. (2005). Did the Bundesbank Follow a Taylor Rule? An Analysis Based on Real-Time Data. Swiss Journal of Economics and Statistics (SJES), 141(II), 213-246.
10. Clements, M. P., & Galvão, A. B. (2009). Forecasting US output growth using leading indicators: An appraisal using MIDAS models. Journal of Applied Econometrics, 24(7), 1187-1206.
11. Clements, M. P., & Galvão, A. B. (2011). Improving Real-time Estimates of Output Gaps and Inflation Trends. VAR models. Discussion paper, Department of Economics, University University of Warwick.
12. Clements, M. P., & Galvão, A. B. (2013). Forecasting with vector autoregressive models of data vintages: US output growth and inflation. International Journal of Forecasting, 29(4), 698-714.
13. Cole, R. (1969). Data errors and forecasting accuracy. In Economic forecasts and expectations: analysis of forecasting behavior and performance (pp. 47-82). NBER.

14. Cook T. R. & Smalter Hall A. (2017). Macroeconomic Indicator Forecasting with Deep Neural Networks (No. RWP 17-11).
15. Croushore, D. (2011). Frontiers of real-time data analysis. Journal of economic literature, 49(1), 72-100.
16. Croushore, D. and T. Stark (2001) A real-time data set for macroeconomists. Journal of Econometrics, 105 pp. 111-130.
17. Croushore, D., & Stark, T. (2003). A real-time data set for macroeconomists: Does the data vintage matter?. Review of Economics and Statistics, 85(3), 605-617.
18. Denton, Frank T., and John Kuiper. The Effect of Measurement Errors on Parameter Estimates and Forecasts: A Case Study Based on the Canadian Preliminary National Accounts, Review of Economics and Statistics 47 (May 1965), pp. 198-206.
19. Dpke J., Fritsche U. & Pierdzioch C. (2017). Predicting recessions with boosted regression trees. International Journal of Forecasting 33(4) 745-759.
20. Egginton, D. M., Pick, A., & Vahey, S. P. (2002). 'Keep it real!': A real-time UK macro data set. Economics Letters, 77(1), 15-20.
21. Faust, J., Rogers, J. H., & Wright, J. H. (2003). Exchange rate forecasting: the errors we've really made. Journal of International Economics, 60(1), 35-59.
22. Fernandez, A. Z., Branch, H., Koenig, E. F., & Nikolsko-Rzhevskyy, A. (2011). A real-time historical database for the OECD. Globalization and Monetary Policy Institute Working Paper, 96.
23. Filardo, A. J. (1999). How reliable are recession prediction models?. Economic Review-Federal Reserve Bank of Kansas City, 84, 35-56.
24. Forni, M., Hallin, M., Lippi, M., & Reichlin, L. (2000). The generalized dynamic-factor model: Identification and estimation. Review of Economics and statistics, 82(4), 540-554.
25. Forni, M., Hallin, M., Lippi, M., & Reichlin, L. (2005). The generalized dynamic factor model: one-sided estimation and forecasting. Journal of the American Statistical Association, 100(471), 830-840.
26. Friedman J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics 1189-1232.
27. Friedman J. Hastie T. & Tibshirani R. (2010). Regularization paths for generalized linear models via coordinate descent. Journal of statistical software 33(1) 1.
28. Gerberding, C., Seitz, F., & Worms, A. (2005). How the Bundesbank really conducted monetary policy. The North American journal of economics and finance, 16(3), 277-292.
29. Giannone, D., Henry, J., Lalik, M., & Modugno, M. (2012). An area-wide real-time database for the euro area. Review of Economics and Statistics, 94(4), 1000-1013.
30. Hall, R. E. (1978). Stochastic implications of the life cycle-permanent income hypothesis: theory and evidence. Journal of political economy, 86(6), 971-987.
31. Hanif M. N., Mughal K. S. & Iqbal J. (2018). A Thick ANN Model for Forecasting Inflation (No. 99). State Bank of Pakistan Research Department.
32. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media.

33. Howrey, E. P. (1996). Forecasting GNP with noisy data: a case study. Journal of Economic and Social Measurement, 22(3), 181-200.

34. Kapetanios G. & Papailias F. (2018). Big data & macroeconomic nowcasting: Methodological review. ESCoE Discussion Paper (12).

35. Khabibullin R. & Seleznev S. (2020) Stochastic Gradient Variational Bayes and Normalizing Flows for Estimating Macroeconomic Models. Bank of Russia working paper series No. 61.

36. Kishor, N. K., & Koenig, E. F. (2012). VAR estimation and forecasting when data are subject to revision. Journal of Business & Economic Statistics, 30(2), 181-190.

37. Kock A. B. & Tersvirta T. (2013). Forecasting the Finnish Consumer Price Inflation Using Artificial Neural Network Models and Three Automated Model Selection Techniques. Finnish Economic Papers 26(1) 13-24.

38. Koenig, E. (2003) Is the makeup a useful real-time predictor of inflation? Economic Letters, 80 pp. 261-267.

39. Koenig, E. F., Dolmas, S., & Piger, J. (2003). The use and abuse of real-time data in economic forecasting. Review of Economics and Statistics, 85(3), 618-628.

40. Kydland, F. E., & Prescott, E. C. (1990). Business cycles: Real facts and a monetary myth. Real business cycles: a reader, 383.

41. lvarez-Daz M. & Gupta R. (2015). Forecasting the US CPI: Does Nonlinearity Matter?. Department of Economics University of Pretoria Working Paper No 12.

42. McAdam P. & McNelis P. (2005). Forecasting inflation with thick models and neural networks. Economic Modelling 22(5) 848-867.

43. McCracken, M. W., & Ng, S. (2016). FRED-MD: A monthly database for macroeconomic research. Journal of Business & Economic Statistics, 34(4), 574-589.

44. Molodtsova, T., & Ince, O. (2008). Real-Time Exchange Rate Predictability with Taylor Rule Fundamentals. Manuscript, Emory University.

45. Molodtsova, T., A. Nikolsko-Rzhevskyy, and D. Papell (2008) Taylor rules and real-time data: A tale of two countries and one exchange rate. Journal of Monetary Economics, 55 pp. S63-S79.

46. Moshiri S. & Cameron N. (2000). Neural network versus econometric models in forecasting inflation. Journal of forecasting 19(3) 201-217.

47. Nikolsko-Rzhevskyy, A. (2011). Monetary policy estimation in real time: Forward-looking Taylor rules without forward-looking data. Journal of Money, Credit and Banking, 43(5), 871-897.

48. Orphanides, A. (2001) Monetary policy rules based on real-time data. American Economic Review, 91(4) pp. 964-985, September.

49. Orphanides, A., & Van Norden, S. (2005). The reliability of inflation forecasts based on output gap estimates in real time. Journal of Money, Credit and Banking, 583-601.

50. Ponomarenko A., Gornostaev D., Seleznev S., Sterhova A. (2021) A Real-Time Historical Database of Macroeconomic Indicators for Russia. Bank of Russia working paper series.

51. Rossi, B., & Sekhposyan, T. (2010). Have economic models' forecasting performance for US output growth and inflation changed over time, and when?. International Journal of Forecasting, 26(4), 808-835.
52. Sauer, S., & Sturm, J. E. (2007). Using Taylor rules to understand European Central Bank monetary policy. German Economic Review, 8(3), 375-398.
53. Sermpinis G. Stasinakis C. Theofilatos K. & Karathanasopoulos A. (2014). Inflation and unemployment forecasting with genetic support vector regression. Journal of Forecasting 33(6) 471-487.
54. Shahriari B. Swersky K. Wang Z. Adams R. P. & De Freitas N. (2015). Taking the human out of the loop: A review of Bayesian optimization. Proceedings of the IEEE 104(1) 148-175.
55. Stark, T., & Croushore, D. (2002). Forecasting with a real-time data set for macroeconomists. Journal of Macroeconomics, 24(4), 507-531.
56. Stock, J. H., & W Watson, M. (2003). Forecasting output and inflation: The role of asset prices. Journal of Economic Literature, 41(3), 788-829.
57. Stock, J. H., & Watson, M. W. (2002a). Forecasting using principal components from a large number of predictors. Journal of the American statistical association, 97(460), 1167-1179.
58. Stock, J. H., & Watson, M. W. (2002b). Macroeconomic forecasting using diffusion indexes. Journal of Business & Economic Statistics, 20(2), 147-162.
59. Styrin K. (2019). Forecasting Inflation in Russia by Dynamic Model Averaging. Russian Journal of Money and Finance 78(1) 3-18.
60. Szafranek, K. (2019). Bagged neural networks for forecasting Polish (low) inflation. International Journal of Forecasting, 35(3), 1042-1059.
61. Tiffin A. (2016). Seeing in the dark: A machine-learning approach to nowcasting in Lebanon. WP/16/56 IMF Working paper.
62. Trivellato, U., & Rettore, E. (1986). Preliminary data errors and their impact on the forecast error of simultaneous-equations models. Journal of Business & Economic Statistics, 4(4), 445-453.
63. Zhang L. & Li J. (2012). Inflation Forecasting Using Support Vector Regression. In Information Science and Engineering (ISISE) (pp. 136140). IEEE.

Appendix

Tables

Table A1. Variables and data sources[18]

Name of series	Type	Source	Period
Consumer price index	regular	FSSS	t-1
Industrial production	vintages	FSSS	t-2
Unemployment rate	vintages	FSSS	t-2
Real wage	vintages	FSSS	t-2
Real agricultural production	vintages	FSSS	t-2
Eggs production	vintages	FSSS	t-2
Meat production	vintages	FSSS	t-2
Milk production	vintages	FSSS	t-2
Freight	vintages	FSSS	t-2
Railway freight	vintages	FSSS	t-2
Commercial freight	vintages	FSSS	t-2
Real retail output	vintages	FSSS	t-2
Food retail	vintages	FSSS	t-2
Nonfood retail	vintages	FSSS	t-2
Services	vintages	FSSS	t-2
Public catering	vintages	FSSS	t-2
Construction	vintages	FSSS	t-2
Export of goods	vintages	RSSS	t-2
Import of goods	vintages	RSSS	t-2
Nominal exchange rate	regular	CBR	t-1
Real effective exchange rate	regular	BIS	t-1
Interbank interest rate	regular	CBR	t-1
Deposit interest rate	regular	CBR	t-2
International reserves	regular	CBR	t-1
Monetary aggregate M2 (real)	regular	CBR	t-1
Total government deficit	regular	Roskazna	t-2
Crude oil (Brent) price	regular	EIA	t-1

[18] FSSS – Federal State Statistical Service, CBR – Central Bank of Russia, BIS – Bank for international settlements, EIA – Energy Information Administration.

Table A2. Values of the parameters used in the estimation of random forest

Hyperparameters	Values
criterion	Mse, mae
maximum number of features	Auto,log2, sqrt
number of estimators	(5, 300)
maximum depth of a tree	None, (1, 14)
min. number of samples required to split a node	(2, 10)

Note: default values are criterion – mse, number of estimators = 100, maximum depth – None, maximum number of features – auto, maximum depth of a tree – None, min. samples split = 2.

Table A3. Values of the parameters used in the estimation of gradient boosting

Hyperparameters	Values
loss function	least squares least absolute deviation Huber function
criterion	Friedman mse mse mae
learning rate	(0.001 0.2)
number of estimators	(10 400)
maximum depth	(2 10)
subsample	(0.2 1.0)

Note: default values are loss function - least squares, criterion - Friedman mse, learning rate = 0.1, number of estimators = 100, maximum depth = 3, subsample = 1.0.

Table A4. Optimal parameters of random forest chosen on cross-validation

hyperparameters	one month	three months	six months
first experiment			
criterion	mae	mae	mae
maximum number of features	auto	auto	auto
number of estimators	21	30	33
maximum depth of a tree	8	10	14
min. samples split	8	2	2
max. leaf nodes	47	50	50
number of lags	1	1	3
second experiment			
criterion	mse	mse	mse
maximum number of features	auto	auto	auto
number of estimators	10	39	34
maximum depth of a tree	9	14	14
min. samples split	4	2	4
max. leaf nodes	30	None	None
number of lags	1	1	3
third experiment			
criterion	mae	mae	mae
maximum number of features	auto	auto	auto
number of estimators	39	244	204
maximum depth of a tree	10	13	14
min. samples split	2	5	6
max. leaf nodes	34	45	25
number of lags	1	1	1
fourth experiment			
criterion	mae	mse	mse
maximum number of features	auto	auto	auto
number of estimators	300	194	92
maximum depth of a tree	12	7	14
min. samples split	7	3	2
max. leaf nodes	47	41	43
number of lags	1	1	1

Note: default values are criterion – mse, number of estimators = 100, maximum depth – None, maximum number of features – auto, maximum depth of a tree – None, min. samples split = 2.

Table A5. Optimal parameters of gradient boosting chosen on cross-validation

hyperparameters	one month	three months	six months
first experiment			
loss function	ls	ls	ls
criterion	mse	friedman_mse	mse
learning rate	0,041	0,223	0,145
number of estimators	323	348	314
maximum depth	2	2	2
subsample	0,386	0,768	0,995
number of lags	1	6	3
second experiment			
loss function	ls	ls	huber
criterion	mse	mse	mae
learning rate	0,068	0,181	0,103
number of estimators	399	163	344
maximum depth	2	2	2
subsample	0,552	0,889	0,792
number of lags	6	1	3
third experiment			
loss function	ls	ls	huber
criterion	mse	friedman_mse	mse
learning rate	0,086	0,171	0,205
number of estimators	128	334	211
maximum depth	3	2	2
subsample	0,620	0,619	0,982
number of lags	1	1	3
fourth experiment			
loss function	huber	ls	ls
criterion	friedman_mse	friedman_mse	mse
learning rate	0,050	0,4	0,4
number of estimators	238	10	28
maximum depth	9	2	2
subsample	0,733	1	1
number of lags	1	3	1

Note: default values are loss function - least squares, criterion - Friedman mse, learning rate = 0.1, number of estimators = 100, maximum depth = 3, subsample = 1.0.

Table A6. Optimal parameters of elastic net chosen on cross-validation

hyperparameters	one month	three months	six months
first experiment			
alpha	0,005	0,108	0,044
l1_ratio	0,547	0,907	0,999
fit_intercept	TRUE	TRUE	TRUE
normalize	TRUE	FALSE	FALSE
max_iter	268	237	1821
selection	random	cyclic	cyclic
number of lags	6	1	3
second experiment			
alpha	0,003	0,061	0,067
l1_ratio	0,138	0,987	1,000
fit_intercept	TRUE	TRUE	TRUE
normalize	TRUE	FALSE	FALSE
max_iter	1280	1669	463
selection	random	cyclic	cyclic
number of lags	1	1	3
third experiment			
alpha	0,017	0,201	0,007
l1_ratio	0,653	0,965	1,000
fit_intercept	TRUE	TRUE	TRUE
normalize	FALSE	FALSE	TRUE
max_iter	568	1447	1261
selection	random	random	random
number of lags	1	1	6
fourth experiment			
alpha	0,022	0,057	0,069
l1_ratio	0,836	0,923	0,922
fit_intercept	TRUE	TRUE	TRUE
normalize	FALSE	FALSE	FALSE
max_iter	488	1009	1239
selection	cyclic	random	random
number of lags	3	1	3

Note: default values are alpha = 1 , l1_ration = 0,5, fit_itercept = True, normalize = False, max_iter = 1000, selection = 'cyclic'.

Table A7. RMSE levels on cross-validation and test in the first experiment

Lags	Gradient Boosting		Random Forest		Elastic Net		Neural network	
	CV	test	CV	test	CV	test	CV	test
1 month								
1	0,382	0,356	0,371	0,368	0,377	0,480	0,421	0,409
3	0,400	0,382	0,378	0,373	0,387	0,403	0,439	0,431
6	0,394	0,396	0,405	0,389	0,387	0,380	0,447	0,400
3 months								
1	0,598	0,820	0,644	0,826	0,631	0,756	0,694	0,705
3	0,648	0,759	0,715	0,883	0,673	0,760	0,739	0,728
6	0,646	0,737	0,717	0,895	0,598	0,789	0,891	0,860
6 months								
1	0,837	0,789	0,922	0,963	0,649	0,758	0,769	0,687
3	0,853	0,716	0,939	0,932	0,632	0,643	0,756	0,752
6	0,805	0,842	0,949	0,986	0,586	0,758	0,724	0,876

Table A8. RMSE levels on cross-validation and test in the second experiment

Lags	Gradient Boosting		Random Forest		Elastic Net		Neural network	
	CV	test	CV	test	CV	test	CV	test
1 month								
1	0,360	0,378	0,379	0,393	0,363	0,359	0,363	0,338
3	0,393	0,404	0,387	0,446	0,397	0,362	0,425	0,349
6	0,385	0,375	0,401	0,428	0,370	0,447	0,464	0,411
3 months								
1	0,566	0,710	0,641	0,772	0,587	0,676	0,672	0,626
3	0,650	0,789	0,689	0,816	0,632	0,697	0,847	0,787
6	0,644	0,790	0,700	0,814	0,561	0,720	0,757	0,765
6 months								
1	0,830	0,735	0,961	0,914	0,646	0,722	0,689	0,685
3	0,890	0,716	1,031	0,903	0,684	0,701	0,729	0,698
6	0,839	0,809	1,003	0,927	0,604	0,743	0,712	0,916

Table A9. RMSE levels on cross-validation and test in the third experiment

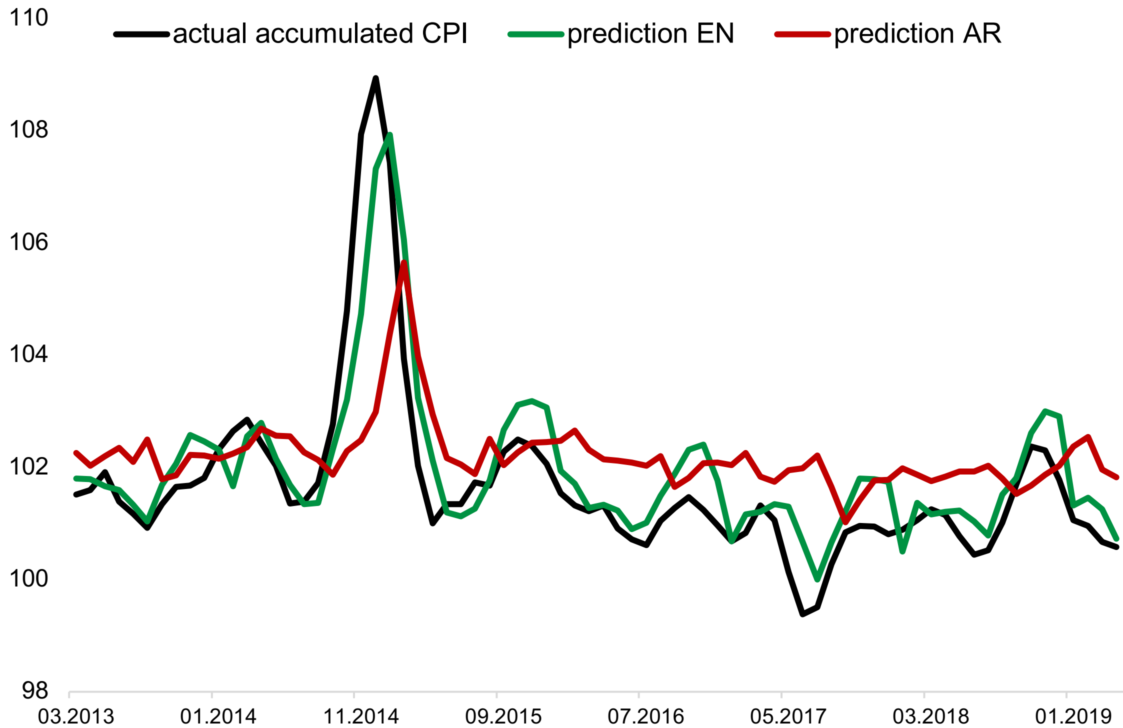
Lags	Gradient Boosting		Random Forest		Elastic Net		Neural network	
	CV	test	CV	test	CV	test	CV	test
1 month								
1	0,348	0,334	0,378	0,373	0,395	0,369	0,378	0,348
3	0,387	0,366	0,393	0,386	0,387	0,399	0,407	0,402
6	0,390	0,392	0,396	0,399	0,353	0,392	0,471	0,414
3 months								
1	0,576	0,677	0,666	0,732	0,592	0,697	0,712	0,615
3	0,655	0,803	0,697	0,824	0,632	0,711	0,729	0,766
6	0,629	0,848	0,720	0,881	0,610	0,751	0,731	0,822
6 months								
1	0,829	0,767	0,929	0,901	0,610	0,767	0,731	0,669
3	0,877	0,731	0,972	0,940	0,639	0,690	0,743	0,657
6	0,842	0,781	0,963	0,972	0,590	0,690	0,723	0,889

Table A10. RMSE levels on cross-validation and test in the fourth experiment

Lags	Gradient Boosting		Random Forest		Elastic Net		Neural network	
	CV	test	CV	test	CV	test	CV	test
1 month								
1	0,293	0,323	0,290	0,326	0,314	0,348	0,313	0,304
3	0,304	0,344	0,302	0,346	0,374	0,295	0,310	0,347
6	0,300	0,344	0,307	0,353	0,313	0,377	0,349	0,391
3 months								
1	0,460	0,700	0,458	0,686	0,444	0,622	0,541	0,625
3	0,480	0,682	0,488	0,715	0,494	0,675	0,561	0,652
6	0,478	0,732	0,494	0,738	0,505	0,664	0,581	0,716
6 months								
1	0,596	0,671	0,627	0,843	0,502	0,666	0,513	0,837
3	0,656	0,791	0,663	0,846	0,512	0,599	0,640	0,804
6	0,636	0,827	0,692	0,880	0,553	0,658	0,788	0,921

Figures

Figure A1. Comparison of AR and elastic net forecasts for three months in advance with one lag



Bank of Russia

Machine learning real-time CPI forecasting

Mariam Mamedli
NRU Higher School of Economics
Denis Shibitov
Bank of Russia

IFC workshop ”Data science in central banking”
Machine learning applications
October, 2021

This views expressed herein are those of the author and do not necessarily represent the position of the Bank of Russia

Aim:

To replicate the construction of CPI forecasts in real time and to define how different aspects of forecasting affect the model accuracy:

- The use of vintages, data availability and the use of SA procedure.

Tasks:

- to build ML models with the optimal architecture (hyperparameters, lags, CV).
- to define how the model performance changes depending on:
    - the type of ML model applied;
    - data availability;
    - the use of vintage data;
    - the use of seasonal adjustment.

Relevant research

- Importance of vintage data (Koenig et al., 2003; Clements, Galvao, 2009)
- The role of data revisions (Stark, Croushore, 2002)

Application of ML models in CPI forecasting:

- Potential of ML models in economic forecasting (Chakraborty, Joseph, 2017)
- The use of neural networks:
  - one country cases (Moshiri, Cameron, 2000; Chen et al., 2001; Szafranek, 2017, Hanif et al., 2018);
  - panel data (Choudhary, Haider, 2012, McAdam, McNelis, 2005);
  - results not in favor of neural networks (Kock, Terasvirts, 2013; Catik, Karacuka, 2012).
- Other models:
  - random forest (Chakraborty, Joseph, 2017; Butavyan, 2019, Baybuza, 2018);
  - gradient boosting (Baybuza, 2018);
  - SVR (Zhang, Li, 2012; Sermipinis et al., 2014; Plakandaras et al., 2017).
- Only one paper on the application to the Russian data (Baybuza, 2018).

Value added

Data:

- Unique for Russia vintage data.

Methodology:

- Application of ML models (one of the first for Russia) + Bayesian optimization;
- The choice of the optimal architecture via CV and different lags.

Concept:

Model comparison in a series of experiments so define the role of:

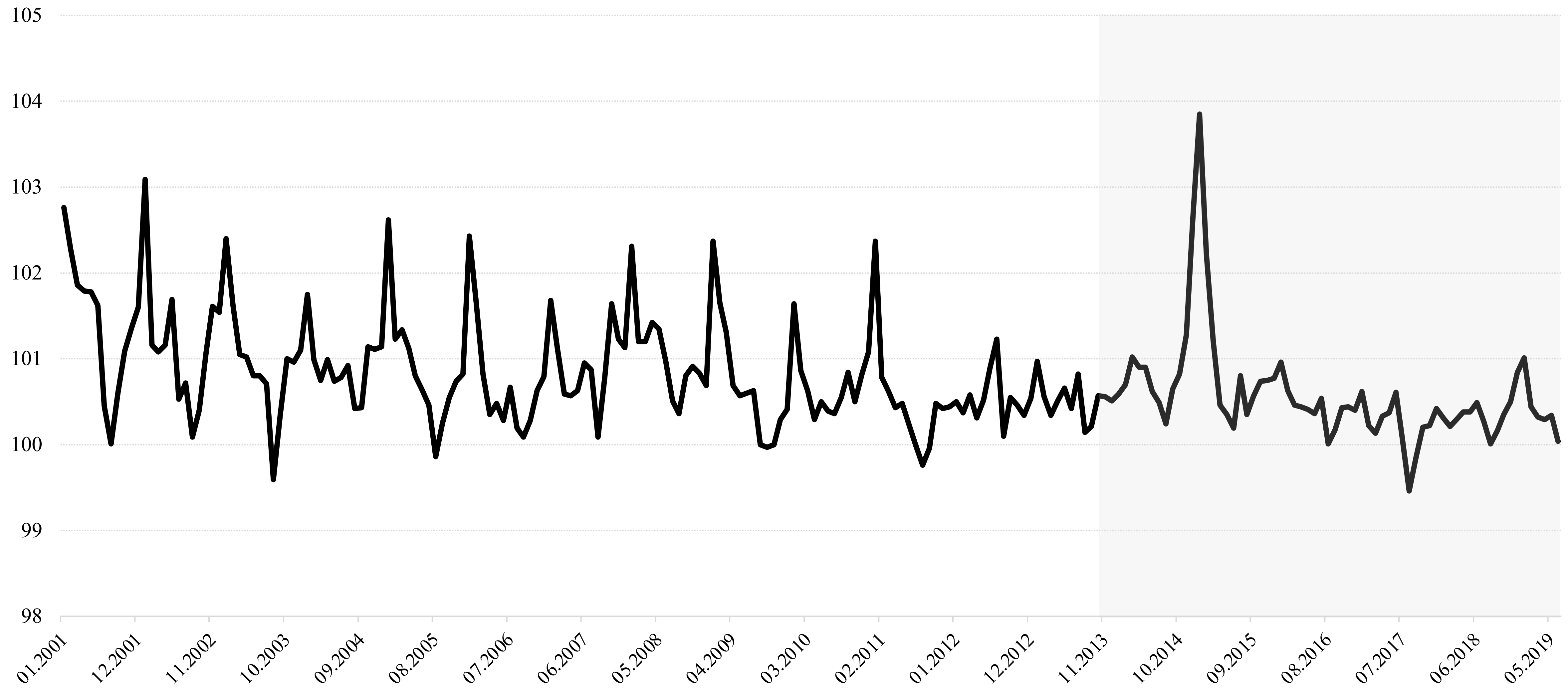
- Data availability;
- Data vintages;
- Seasonal adjustment.

Experiments:

Experiment	Starting lags	Vintages	Seasonality
1	$t-1/t-2$	vintages	NSA
2	$t-1$	vintages	NSA
3	$t-1$	regular	NSA
4	$t-1$	regular	SA

- 1st experiment is considered as benchmark;
- 2d and 3d experiments aimed at the replication of the forecast in real time;
- 4th experiment is devoted to the role of seasonal adjustment.
- The use of NSA data:
  - The forecast of an observed variable;
  - Is not revised (replicability);
  - Commonly used and accepted indicator.

CPI (NSA)

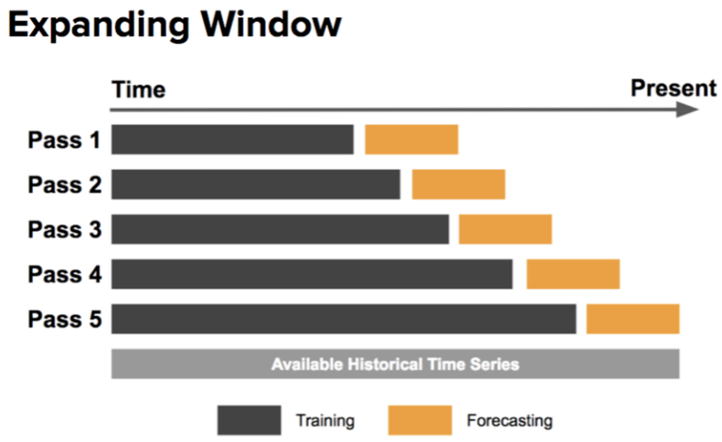

105
104
103
102
101
100
99
98
01.2001
12.2001
11.2002
10.2003
09.2004
08.2005
07.2006
06.2007
05.2008
04.2009
03.2010
02.2011
01.2012
12.2012
11.2013
10.2014
09.2015
08.2016
07.2017
06.2018
05.2019

Data:

- Vintage data including revisions + other macro and financial variables;
- Data from January 2001 to June 2019;
- 27 variables + 'dummy' for each month;
- NSA, minimal series transformations;
- Forecast for 1, 3 and 6 months with 1, 3 and 6 lags each.

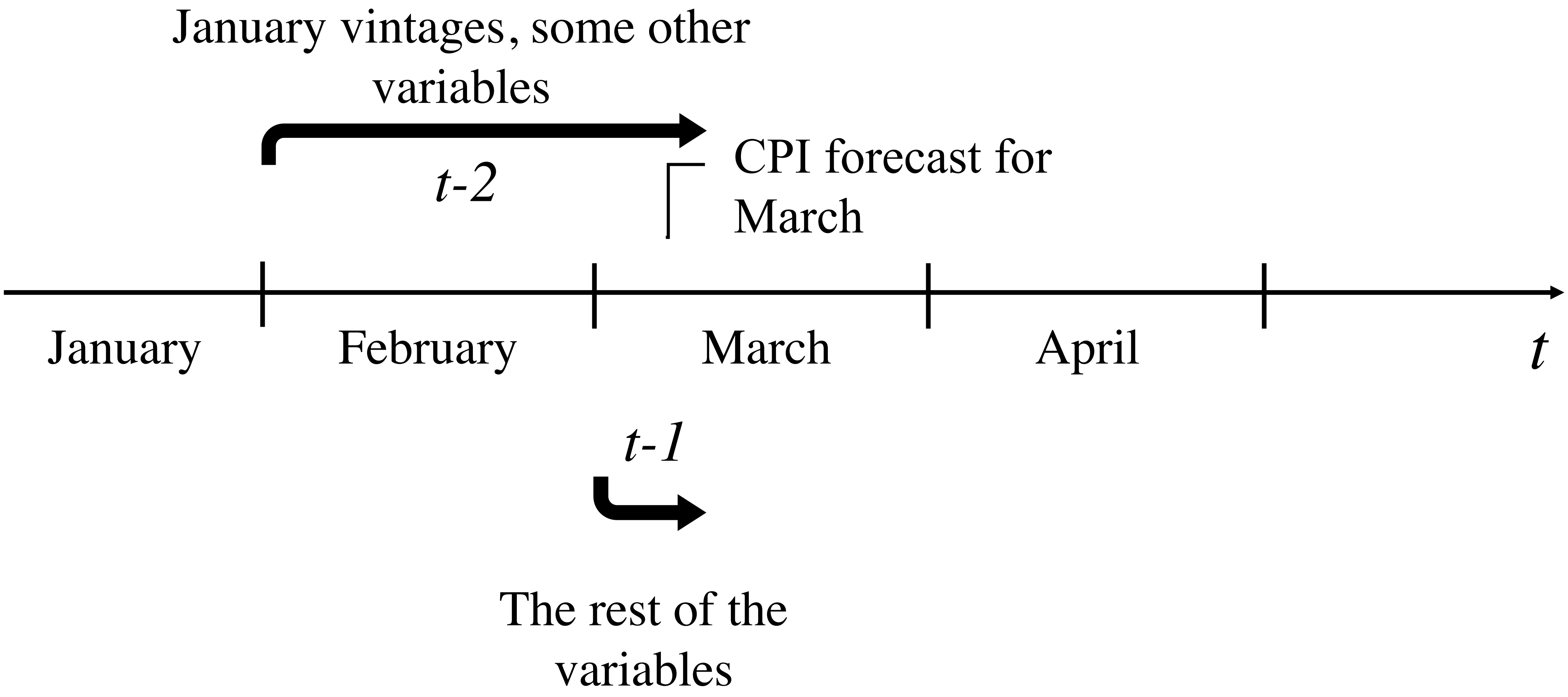
Cross-validation:

- Expending window



Name of series	Variable name	Type	Source	Period
Consumer price index	cpi	regular	FSSS	t-1
Industrial production	ip	vintages	FSSS	t-2
Unemployment rate	unempl_15_72	vintages	FSSS	t-2
Real wage	real_wages	vintages	FSSS	t-2
Real agricultural production	agriculture	vintages	FSSS	t-2
Eggs production	agr_eggs	vintages	FSSS	t-2
Meat production	agr_meat	vintages	FSSS	t-2
Milk production	agr_milk	vintages	FSSS	t-2
Freight	freight_total	vintages	FSSS	t-2
Railway freight	freight_railway	vintages	FSSS	t-2
Commercial freight	freight_com	vintages	FSSS	t-2
Real retail output	retail_total	vintages	FSSS	t-2
Food retail	retail_food	vintages	FSSS	t-2
Nonfood retail	retail_nonfood	vintages	FSSS	t-2
Services	services	vintages	FSSS	t-2
Public catering	restaurants	vintages	FSSS	t-2
Construction	construct_l	vintages	FSSS	t-2
Export of goods	export	vintages	RSSS	t-2
Import of goods	import	vintages	RSSS	t-2
Nominal exchange rate	ner	regular	CBR	t-1
Real effective exchange rate	reer	regular	CBR	t-1
Interbank interest rate	miacr	regular	CBR	t-1
deposit interest rate	deposit_rate	regular	CBR	t-2
International reserves	reserves	regular	CBR	t-1
Monetary aggregate M2 (real)	m2	regular	CBR	t-1
Total government deficit	gov_dev	regular	Roskazna	t-2
Crude oil (Brent) price	oil_price	regular	EIA	t-1

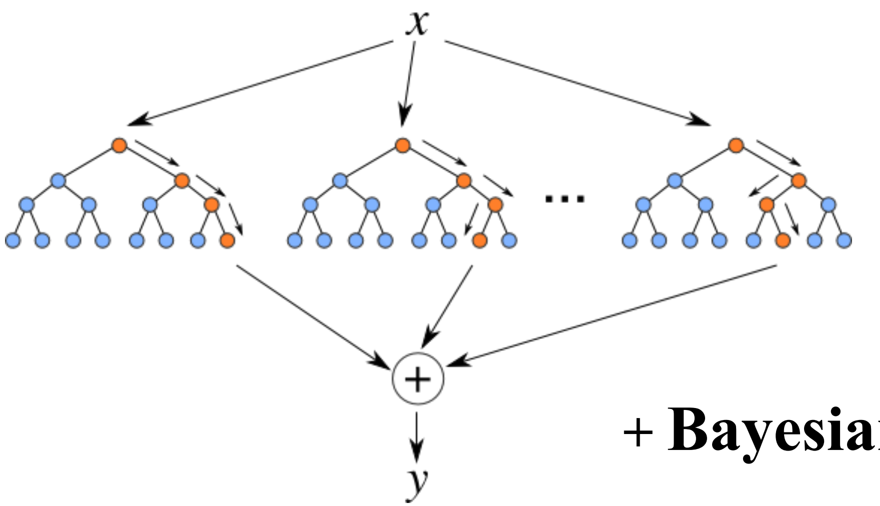
Data:


January vintages, some other variables
t-2
CPI forecast for March
January
February
March
April
t
t-1
The rest of the variables

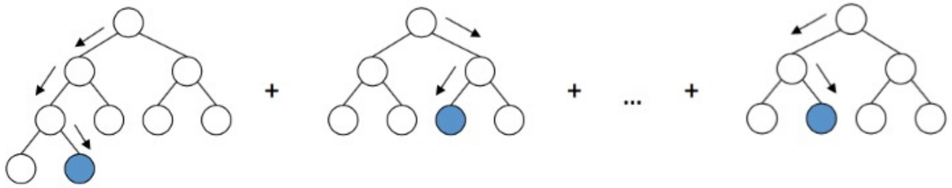
Models

Elastic net

Random forest



Gradient boosting



+ Bayesian optimization

Bayesian Neural Network

- BNN from Khabibullin, Seleznev (2020);
- Network size (30 and 10 neurons);
- Normal error distribution with Tanh activation function and tStudent distribution with ReLu activation function.

$$h_{1,t} = f(W_1 x_t + b_1),$$

$$h_{2,t} = f(W_2 h_{1,t} + b_2),$$

$$y_t = W_3 h_{2,t} + b_{y,t} + \varepsilon_t,$$

where x_t are input data, W_i are weights on the layers $i=1, 2$ and output layer, b_i and b_y are bias, $f(\cdot)$ is an activation function, h_i is the output of the hidden layer i, y_t is the output of the neural network at time t and ε_t is an error term.

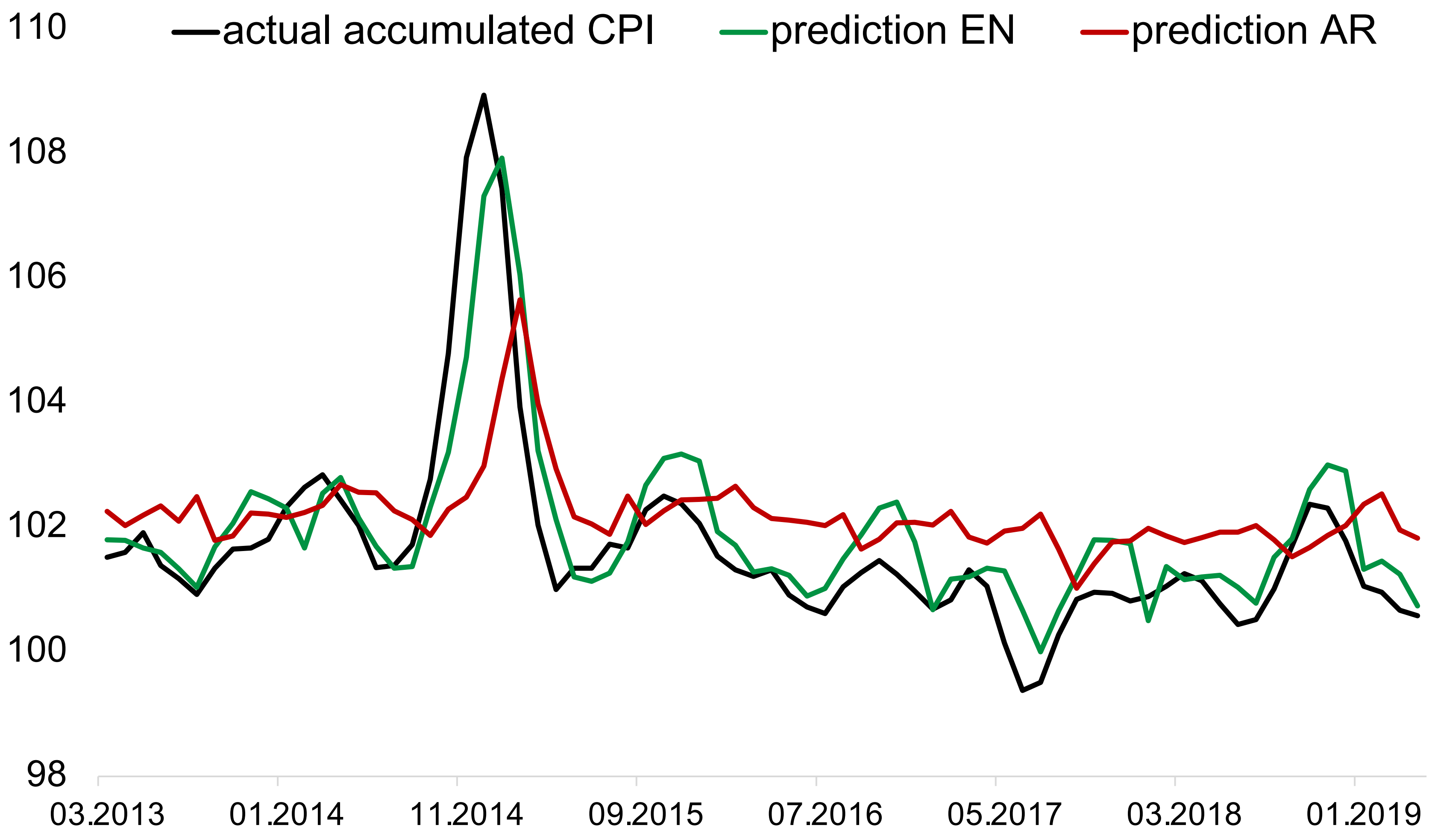
Model comparison, Exp.1:

RMSE levels for different models

lags	Elastic Net	Random Forest	Gradient Boosting	Neural network	AR
1 month					
1	0,480	0,368	0,356	0,409	0,404
3	0,403	0,373	0,382	0,431	0,427
6	0,380	0,389	0,396	0,400	0,392
3 months					
1	0,756	0,826	0,820	0,705	1,414
3	0,760	0,883	0,759	0,728	1,393
6	0,789	0,895	0,737	0,860	1,390
6 months					
1	0,758	0,963	0,789	0,687	2,693
3	0,643	0,932	0,716	0,752	2,681
6	0,758	0,986	0,842	0,876	2,664

Note: AR model was estimated with seasonal dummies for proper model comparison. The lowest RMSE levels for each forecasting horizon are marked with grey.

Forecast comparison of EN and AR (for 3 month in advance with 1 lag):



The role of data availability:

Comparison of models RMSE, with and without taking into account data availability

	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	$t-1$	$t-1/t-2$	$t-1$	$t-1/t-2$	$t-1$	$t-1/t-2$	$t-1$	$t-1/t-2$
1 month								
1	**0,359**	0,480	0,393	**0,368**	0,378	**0,356**	**0,338**	0,409
3	0,362	0,403	0,446	0,373	0,404	0,382	0,349	0,431
6	0,447	0,380	0,428	0,389	0,375	0,396	0,411	0,400
3 months								
1	**0,676**	0,756	**0,772**	0,826	**0,710**	0,820	**0,626**	0,705
3	0,697	0,760	0,816	0,883	0,789	0,759	0,787	0,728
6	0,720	0,789	0,814	0,895	0,790	0,737	0,765	0,860
6 months								
1	0,722	0,758	0,914	0,963	0,735	0,789	**0,685**	0,687
3	0,701	**0,643**	**0,903**	0,932	**0,716**	0,716	0,698	0,752
6	0,743	0,758	0,927	0,986	0,809	0,842	0,916	0,876

The role of data vintages:

Comparison of models RMSE, with and without taking into account data availability

	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	vintages	no vintages	vintages	no vintages	vintages	no vintages	vintages	no vintages
				1 month				
1	**0,359**	0,369	0,393	**0,373**	0,378	**0,334**	**0,338**	0,348
3	0,362	0,399	0,446	0,386	0,404	0,366	0,349	0,402
6	0,447	0,392	0,428	0,399	0,375	0,392	0,411	0,414
				3 months				
1	**0,676**	0,697	0,772	**0,732**	0,710	**0,677**	0,626	**0,615**
3	0,697	0,711	0,816	0,824	0,789	0,803	0,787	0,766
6	0,720	0,751	0,814	0,881	0,790	0,848	0,765	0,822
				6 months				
1	0,722	0,767	0,914	**0,901**	0,735	0,767	0,685	0,669
3	0,701	**0,690**	0,903	0,940	**0,716**	0,731	0,698	**0,657**
6	0,743	0,690	0,927	0,972	0,809	0,781	0,916	0,889

The role of seasonal adjustment:

Comparison of models RMSE, with and without seasonal adjustment

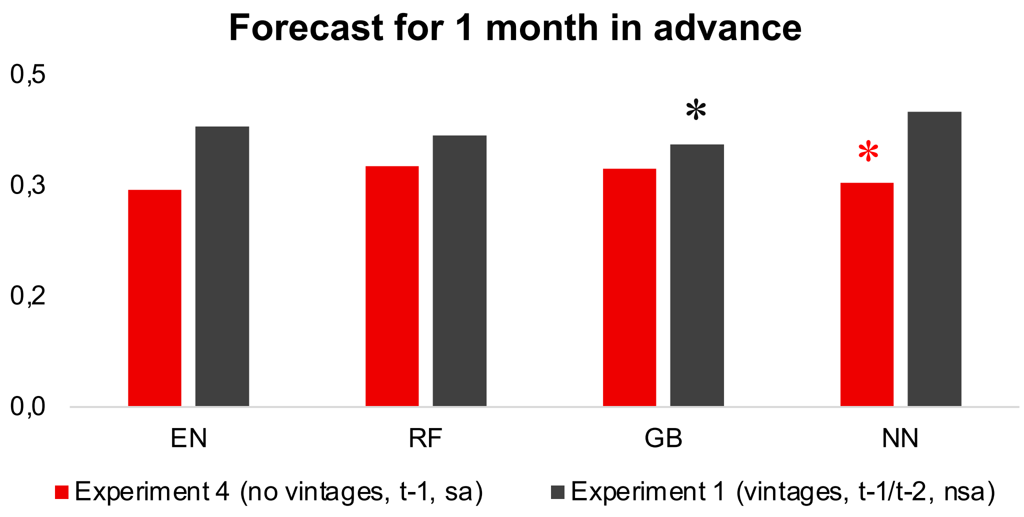
	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	sa	nsa	sa	nsa	sa	nsa	sa	nsa
1 month								
1	0,348	0,369	**0,326**	0,373	**0,323**	0,334	**0,304**	0,348
3	**0,295**	0,399	0,346	0,386	0,344	0,366	0,347	0,402
6	0,377	0,392	0,353	0,399	0,344	0,392	0,391	0,414
3 months								
1	**0,622**	0,697	**0,686**	0,732	0,700	**0,677**	0,625	**0,615**
3	0,675	0,711	0,715	0,824	0,682	0,803	0,652	0,766
6	0,664	0,751	0,738	0,881	0,732	0,848	0,716	0,822
6 months								
1	0,666	0,767	**0,843**	0,901	**0,671**	0,767	0,837	0,669
3	**0,599**	0,690	0,846	0,940	0,791	0,731	0,804	**0,657**
6	0,658	0,690	0,880	0,972	0,827	0,781	0,921	0,889

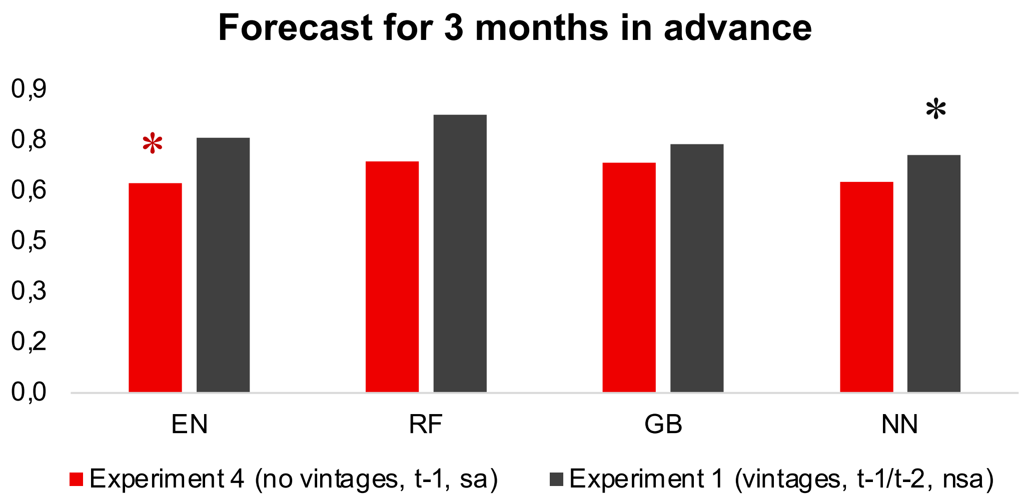
Joint effect of all data transformations:

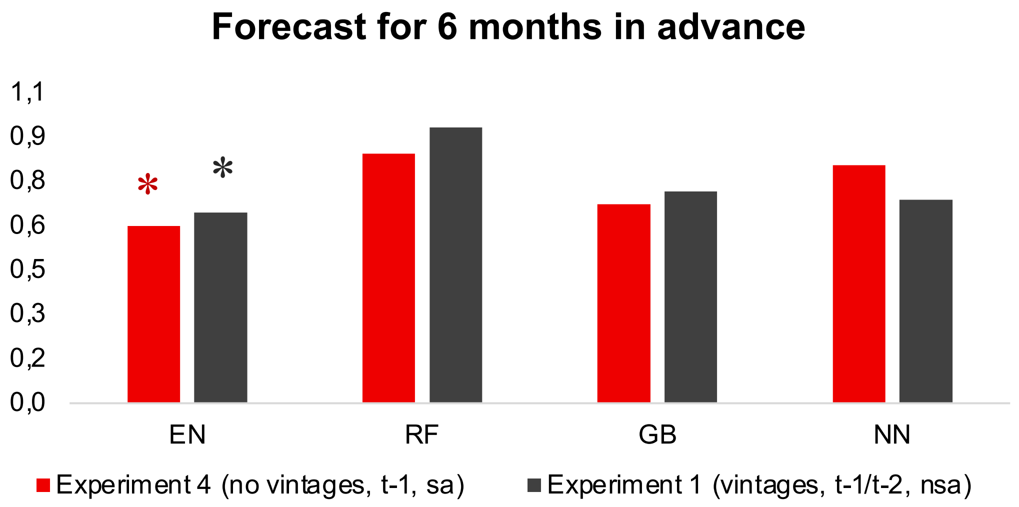
Comparison of models RMSE in two experiments

	Elastic Net		Random Forest		Gradient Boosting		Neural network	
lags	exp. 4	exp.1	exp. 4	exp.1	exp. 4	exp.1	exp. 4	exp.1
1 month								
1	0,348	0,480	**0,326**	0,368	**0,323**	0,356	**0,304**	0,409
3	**0,295**	0,403	0,346	0,373	0,344	0,382	0,347	0,431
6	0,377	0,380	0,353	0,389	0,344	0,396	0,391	0,400
3 months								
1	**0,622**	0,756	**0,686**	0,826	0,700	0,820	**0,625**	0,705
3	0,675	0,760	0,715	0,883	**0,682**	0,759	0,652	0,728
6	0,664	0,789	0,738	0,895	0,732	0,737	0,716	0,860
6 months								
1	0,666	0,758	**0,843**	0,963	**0,671**	0,789	0,837	**0,687**
3	**0,599**	0,643	0,846	0,932	0,791	0,716	0,804	0,752
6	0,658	0,758	0,880	0,986	0,827	0,842	0,921	0,876

Comparison of Exp.1 and Exp.4:







Results

- The use of all data (including unavailable) underestimate an error by 11%, 8% and 2% in average for the forecasting for 1, 3 and 6 months in advance correspondingly.
- Forecasts based on the final data have lower error on average (8%, 5% and 2%), yet the results are very sensitive to the model type, number of lags and forecast horizon.
- The use of seasonally adjusted data leads to the lower error (12%, 9% and 9%).
- The joint effect of all three transformations is on average 17%, 13% and 8% depending on the forecasted horizon.
- The results point in favor of GB (1 month), NN (3 months) and EN (6 months) comparing to AR.