FUNDAÇÃO GETÚLIO VARGAS
ESCOLA DE PÓS-GRADUAÇÃO EM ECONOMIA

Leonid Garnitskiy

Nowcasting Brazilian Inflation with Machine Learning

Rio de Janeiro
2020

Leonid Garnitskiy

Nowcasting Brazilian Inflation with Machine Learning

Dissertação submetida à Escola de Pós-Graduação em Economia como requisito parcial para obtenção do grau de Mestre em Economia.

Área de concentração: Econometria

Orientador: João Victor Issler

Rio de Janeiro
2020

Dados Internacionais de Catalogação na Publicação (CIP)
Ficha catalográfica elaborada pelo Sistema de Bibliotecas/FGV

Garnitskiy, Leonid
Nowcasting Brazilian Inflation with machine learning / Leonid Garnitskiy. – 2020.
83 f.

Dissertação (mestrado) - Fundação Getulio Vargas, Escola Brasileira de Economia e Finanças.
Orientador: João Victor Issler.
Inclui bibliografia.

1. Inflação – Brasil - Modelos macroeconômicos. 2. Aprendizado do Computador. 3. Modelagem de dados. I. Issler, João Victor. II. Fundação Getulio Vargas. Escola Brasileira de Economia e Finanças. III. Título.

CDD – 332.410981

Elaborada por Márcia Nunes Bacha – CRB-7/4403

LEONID GARNITSKIY

"NOWCASTING BRAZILIAN INFLATION WITH MACHINE LEARNING''.

Dissertação apresentado(a) ao Curso de Mestrado em Economia do(a) EPGE Escola Brasileira de Economia e Finanças - FGV EPGE para obtenção do grau de Mestre(a) em Economia.

Data da defesa: 24/4/2020

ASSINATURA DOS MEMBROS DA BANCA EXAMINADORA

Presidente da Comissão Examinadora: Profº/ª **JOÃO VICTOR ISSLER**

João Victor Issler
Orientador

Felipe Saraiva Ichan

Wagner Piazza Gaglianone

Em cumprimento ao DECRETO nº 46.970 de 13/03/20 - Poder Executivo do Estado do Rio de Janeiro, DOE nº 047-A em 13/03/20, Art 4ª e Portaria MEC nº 343 de 17/03/20, DOU nº 53 de 18/03/20, que dispõe sobre a suspensão temporária das atividades acadêmicas presenciais e a utilização de recursos tecnológicos (em conformidade à legislação vigente), face ao COVID-19, as apresentações das defesas de Tese e Dissertação, de forma excepcional, serão realizadas de forma remota, inclui-se nessa modalidade membros da banca e discente.

Lucas Jóver Maestri
Coordenador

Antonio de Araujo Freitas Junior
Pró-Reitor de Ensino, Pesquisa e Pós-Graduação FGV
Antonio Freitas, PhD
Pró-Reitor de Ensino, Pesquisa e Pós-Graduação
Fundação Getúlio Vargas

Instrução Normativa nº 01/19, de 09/07/19 - Pró-Reitoria FGV
Em caso de participação de Membro(s) da Banca Examinadora de forma não-presencial*, o Presidente da Comissão Examinadora assinará o documento como representante legal, delegado por esta I.N.
*Skype, Videoconferência, Apps de vídeo etc

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

This study was also financed by Brazilian Financial and Capital Markets Association, ANBIMA, through **XV Prêmio ANBIMA de Mercado de Capitais**.

Abstract

Many efforts are made by economists to track key macroeconomic variables in real time. This paper aims to make a contribution to economic forecasting research by employing Machine Learning techniques to perform a daily nowcasting of Brazilian inflation. The original results obtained are encouraging. The benefit of making a daily nowcast of inflation instead of one-month-ahead forecast is found to be of 50%-60% on average for almost all ML models considered. The best-performing ML techniques are Complete Subset Regression and Random Forest. The results also show that using ML methods instead of univariate benchmarks reduces the nowcasting error in at most 20%.

Keywords: inflation nowcasting, Machine Learning, Complete Subset Regressions.

Resumo

A importância de monitoramento das principais variáveis macroeconômicas é evidenciada pelo grande esforço que os agentes devotam a esta tarefa. O presente trabalho propõe-se a contribuir para a literatura de previsão econômica, aplicando os modelos de aprendizado de máquina para monitorar diariamente a inflação brasileira medida pelo IPCA. Os resultados obtidos são promissores. O benefício de fazer monitoramento diário da inflação em vez da previsão uma vez por mês é na ordem de 50%-60% em média para quase todos os modelos de aprendizado de máquina considerados. Os modelos que apresentam o melhor desempenho são Regressão de Subconjunto Completo e Floresta Aleatória. Os resultados também mostram que usar técnicas multivariadas de aprendizado de máquina em vez de simples modelos univariados reduz o erro da previsão em até 20%.

Palavras-chave: monitoramento da inflação, Aprendizado de Máquina, Regressão de Subconjunto Completo.

Contents

List of Tables

List of Figures

1 Introduction

Inflation – defined as an increase in the general price level – is one of the most central macroeconomic phenomena that affects an economy pervasively. In 1980's and the beginning of 1990's years, Brazil experienced hyperinflation process, which has been overcome since 1994; nevertheless, inflation in Brazil always remains under close attention of market agents.

Meanwhile, in the first two decades of the XXI century the applied research in economic science has been enriched with two innovations: the emergence of big data – extremely large data sets – and computer machines capable of processing this data. Hence, economists have more and more frequently resorted to *machine learning* techniques. Machine learning (ML) is generally defined as a science (and art) of pattern recognition from a given dataset. Supervised[1] machine learning is beneficial for prediction (Varian (2014), Mullainathan and Spiess(2017)). An important prediction task faced by macroeconomists is a *nowcasting* of key economic variables.

Nowcasting is the prediction of the present, the near future and the recent past (Banbura et al., 2013). This definition highlights the fact that economic statistics are published with some delay after the reference period: for example, Brazilian Broad Consumer Price Index (IPCA) of a month is usually released on the eighth working day of the subsequent month. However, market participants and policy makers need to monitor the economy in real time (every day) to make correct decisions. The monetary authority depends on accurate inflation nowcast to formulate adequate responses in order to reach the inflation target (in Brazil, the target in 2020 is 4%, with tolerance band of ± 1,5 percentage point), as well as to anchor expectations[2]. Firms and consumers are also interested in nowcasting inflation, since almost all the contracts in the economy are established in nominal terms. For Brazilian Financial Market participants, inflation nowcasting is important for asset pricing. In fact, asset prices are affected daily by releases of new data on macroeconomic series when this data brings some surprise to the market (Flannery and Protopapadakis, 2002). In Brazil, inflation exhibits high short-term volatility, which makes financial institutions allocate significant efforts to nowcast inflation (Garcia et al., 2017). As there is a delay in publication of the relevant data, nowcasting attempts to predict this data using already available series.

According to Stock and Watson (2017), during the last two decades the researchers in the field of Econometrics have made efforts to elaborate scientific methods of forecasting. A scientific method is the one that is transparent, replicable, capable of quantifying uncertainty, has well-defined properties and can have its performance evaluated out of sample. Thus, the aim of this research program has been to develop reliable algorithms of prediction in order to reduce a degree of subjectivity and "expert judgment", that forecasting used to rely on.

[1] Machine learning methods can be classified into three groups: 1) *supervised learning*, when the goal is, given the set of predictors (*features*), predict the value of dependent (*target*) variables: e.g., regression, classification and class probability estimation, causal modeling; 2) *unsupervised learning*, when the answers are not given or even do not exist, and the goal is to recognize data patterns: e.g., clustering analysis, co-occurrence grouping, profiling and dimensionality reduction (such as Principal Component Analysis, PCA); 3) *reinforcement learning*, which aims to iteratively search for input variables that optimize some reward function: e.g., dynamic programming techniques.

[2] Medeiros et al. (2019) mention high welfare costs arising from Central Banks' errors in forecasting inflation.

The present paper aims to make a contribution to this research program. This paper has 3 main objectives: 1) to compare, according to defined criteria, the performance of different Machine Learning and Econometric methods in nowcasting Brazilian inflation; 2) to measure the benefits of working with high-frequency (daily) data instead of forecasting the inflation once a month; 3) to assess which features have the greatest impact on the update of the target variable. The target in this paper is the Brazilian Broad Consumer Price Index (IPCA) – a monthly inflation index calculated by Brazilian Institute of Geography and Statistics (IBGE). This index is the most important one among other inflation indexes that exist in Brazil, not only due to being the reference in many contracts but also because it is used by the Central Bank of Brazil (BCB) in inflation targeting.

The literature on economic nowcasting is recent (XXI century), but some of its statistical techniques date back to the XX century. The statistical problem that permeates all this research is the following: in the face of a large number of potential predictors in the data (sometimes even larger than number of observations), it is necessary to reduce the number of predictors. The idea that the most part of behavior of a large set of variables can be explained by few variables which govern the whole dataset is quite old. For instance, Principal Component Analysis (PCA) was first developed by Pearson (1901) and Hotelling (1933). Closely related to it, the Factor Model has been extensively used in macroeconomic research, since Sargent and Sims (1977) showed that a great part of variance of important macroeconomic variables, such as GDP, prices and unemployment, could be explained by 2 dynamic factors. Stock and Watson (2011) remark 3 generations of Dynamic Factor Model (DFM): 1) estimation of the Gaussian likelihood via the Kalman Filter (Engle and Watson (1981,1983), Stock and Watson (1989), Sargent (1989), and Quah and Sargent (1993)); 2) nonparametric averaging methods (Chamberlain and Rothschild (1983), Stock and Watson (2002a, 2002b), Forni et al. (2009)); 3) hybrid principal components and state space methods, implemented in Giannone et al. (2008), Giannone et al. (2004) and discussed in Doz et al. (2011). In the former paper, the term "nowcasting" was introduced to the economics, and two-step estimation of DFM was applied. To improve the Factor Model, Bai and Ng (2008) proposed Targeted Factor Model (TFM). Meanwhile, Machine Learning methods were also being developed to deal with high dimensionality of data: Support Vector Machines (SVM) (Cortes and Vapnik (1995)), Least Absolute Shrinkage and Selection Operator (LASSO) (Tibshirani (1996)), adaptive LASSO (adaLASSO) (Zou (2006)), Elastic Net (ElNet) (Zou and Hastie 2005), adaptive Elastic Net (adaElNet), Ridge Regression (RR) (Hoerl and Kennard (1970)), Bayesian VAR (BVAR) (Banbura et al. (2010)), Bagging (Breiman (2008)), Factor Boosting (Bai and Ng (2008)), Complete Subset Regressions (CSR) (Elliott et al. (2013, 2015)), Jackknife Model Averaging (JMA) (Hansen and Racine (2012); Zhang et al. (2013)), and Random Forests (RF) (Breiman (2001)).

Above mentioned techniques have been used to forecast several economic variables, such as GDP (Giannone et al. (2008), Banbura et al. (2013), Richardson et al. (2018)), inflation (Modugno (2013), Chakraborty and Joseph (2017)), commodities prices (Xie et al. (2006)), stock prices (Huang et al. (2005)) and exchange rates (Colombo and Pelagatti (2019)). As to Brazilian inflation forecasting, Arruda et al. (2011) employed ARMA, VAR, TAR (Threshold Autoregressive) and Phillips Curve models and showed that AR models deliver smaller

forecasting errors than Phillips Curve models; later, Medeiros and Vasconcelos (2016) considered some high-dimensional ML models (LASSO, Bagging, TFM and CSR) to conclude that high-dimensional models have, on average, smaller forecasting errors than autoregressive and factor models. Also, Medeiros and Vasconcelos (2016) compared performance of AR, Factor Model, LASSO and adaLASSO to forecast Brazilian inflation indexes IPCA and IGP-M with high-dimensional data and found that LASSO performs better[3] for shorter forecasting horizons, but for longer ones AR model is still better. Further, Garcia et al. (2017) contributed to this literature by considering *real-time* forecasting (i.e., each forecast is computed using only the information that was available to the forecaster at the point of time when he made this forecast) and by using AR, Random Walk (RW) and Unobserved Components Stochastic Volatility (UCSV) models as benchmarks to assess the performance of LASSO, adaLASSO (and its flex version), post-OLS, TFM, CSR, RF and combination of professional forecasts[4] (based on the model confidence sets) in forecasting Brazilian inflation index IPCA. The data they used refers to the period from January 2003 to December 2015 and consists of 156 observations on 59 macroeconomic variables and 34 variables linked to specialist forecasts. They found that, for the shortest horizon (5 days ahead), LASSO and FOCUS perform better; for 1 month and 5 days ahead horizon, adaLASSO is the best model; for further forecasting horizons, CSR is superior to other models. Moreover, the average of the models included in the confidence set was found to be the best model.

More recently, Medeiros et al. (2019) assessed the performance of ML methods in real-time forecasting of U.S. inflation. Their paper, in comparison with Garcia et al. (2017), employs more ML models (10 ML models, 3 Factor Models, 3 benchmark models and 3 combinations of forecasts), has richer database (monthly observations from January 1960 till December 2015, total of 672 observations, and 122 variables) and discusses in more details the variable selection. The best performing model was found to be the RF. According to the authors, this success can be due to possible nonlinearities between inflation and its predictors; also, can be due to the RF's variable selection mechanism. Moreover, Medeiros et al. (2019) point out that the gains of using ML techniques to forecast inflation can achieve 30% in terms of Mean Squared Error (MSE).

The present paper aims to make a contribution to this discussion by assessing the above mentioned models in a real-time *nowcasting* exercise. The prediction is made for the *current* month and is updated *daily* using new data that emerges in the information set. This paper uses larger database than Garcia et al. (2017) and considers the models studied in Medeiros et

[3]The metric to evaluate the quality of performance of a model is: the smaller is forecast error (for instance, Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) or Median Absolute Deviation from the median (MAD)) the better is forecasting power of the model.

[4]In Brazil, professional forecasts on variables such as GDP, IPCA, Exchange Rate, Interest Rate SELIC etc are collected by the Central Bank of Brazil via on-line system, and the medians of these forecasts are published every Monday in form of a report called FOCUS (thus, henceforth in this paper, professional forecast data in Brazil will be referred to as FOCUS). According to the Central Bank of Brazil, approximately 140 financial institutions (and non-financial ones) – such as banks, asset management companies, brokerage firms and consulting companies – are actually registered in this system. These market agents can update their forecasts at any time, but many of them update on Fridays. In order to encourage market agents to update their forecasts, the Central Bank also publishes Top5 ranking of the most accurate forecasters to promote their businesses.

al. (2019), but applies them to Brazilian economy in order to search for the best performing model(s) and to quantify the benefits of daily nowcasting. The innovation is the following: while Garcia et al. (2017) makes *forecasting* of Brazilian inflation, issuing the prediction (for the current month's and 11 subsequent months' inflation) *once* a month (5 days before the release of IPCA), this work makes *nowcast* of this variable, issuing the prediction of the current month's inflation *every day*. Thus, it makes possible to assess if the usage of high frequency data improves the accuracy of the forecast.

Following this Introduction, the paper is organized as follows. Section 2 describes the Dataset. Section 3 reports the Methodology used in this study. Section 4 implements the Kalman Filter to balance the Dataset. Section 5 presents, succinctly, all the Models used. Section 6 discusses the Results obtained. Section 7 concludes.

2 Data

This research deals with supervised learning: there is a target variable (IPCA inflation rate), and the task is to nowcast this variable. Given this task, the researcher proceeds to *feature engineering*, that is, selection of features that make this task be well performed by ML algorithms. The features must be informative about the target, i.e., must reduce the uncertainty about the IPCA rate. The dataset used in this research is the object-feature matrix which has 174 macroeconomic time series as columns (features) and 2974 observations, from 01/12/2006 to 03/10/2018, as rows (objects). The choice of time window is due to the availability of the four Monitors data. The dataset is detailed in Table A in Appendix. The data has mixed frequency: there are daily, weekly and, mostly, monthly series. The criteria for including a feature in the dataset were its possible relatedness to inflation process (according to Economic Theory) and availability of the data for the relevant time window. The dataset describes the behavior of the economy as a whole and contains the most important economic series in its respective sector: (I) Prices; (II) Money and Finance; (III) Production ans Sales; (IV) External Sector; (V) Public Sector; (VI) Labor, Employment and Income; and (VII) Expectations. The innovation of this paper is the inclusion of four Monitors for Brazilian inflation in the dataset. These Inflation Monitors have daily frequency, are produced by the FGV and are used by Brazilian Financial Market institutions as a *proxy* for the current month's inflation rate. The inclusion of these features is supposed to improve the nowcast (relatively to other researches) because these features are the nearest to the target variable among all other available features up to now. For the purpose of implementation of the Models of Section 5, the dataset was augmented by 4 principal components computed from the original dataset, as well as encompassed 4 lags for each variable, following Medeiros et al. (2019). Thus, the Models in Section 5 were applied to the dataset with $174 \times 4 + 16 = 712$ variables and 2974 observations. The rest of this Section provides the description of the Monitors.

Monitor IPCA is time series with daily frequency produced by FGV that simulates the IPCA inflation rate of the last 30 days. This Monitor (like the three others) uses the collection of prices made by the FGV for calculation of the IPC (Consumer Price Index) inflation index but applies the official IBGE weighting. Unlike the official IPCA index, which has one collection

of prices for each calculation period, the IPC collection occurs continuously, so the mean of the collected prices is calculated and compared to the analogous mean in the previous period. The geographic coverage of the collection of prices encompasses the cities comprising most of the weight (more than 90%) in the IPCA; for cities not covered, there is monitoring of administered prices. The collection calendar is similar to that used for the IPCA by IBGE. Each month, the collection period of IPCA ends approximately on the 27th day; hence, the 27th day Monitor data is the best *proxy* to the inflation rate that will be published by IBGE on the 8th day of subsequent month. The weights are updated in two stages. The first stage takes place in the last day of the IBGE collection, according to the official calendar, and uses the relative prices collected by FGV. The second stage occurs when the IPCA index is published by IBGE, so that the variations of subitems' prices become known.

Differently from the Monitor IPCA, the Monitor IPCA Ponta is not based on comparison of the means, but on direct comparison of the prices collected in the last 7 days with the same period of the previous month. Hence, the Ponta result that best anticipates the current month's inflation rate is each month's 7th day result. Also, the Ponta has the capacity of anticipating the upper bound of the current month's inflation.

Monitor IPCA-15 is the analogous Monitor for the IPCA-15 inflation index. This index uses the same prices collected for the IPCA, but the calculation period is different: it encompasses from 15th day of a month till 14th day of the subsequent month and is published by IBGE approximately on the 22th day of each month. It is worth noting that, until the publication of the official IPCA-15 inflation index, the Monitor is the best variable to nowcast the current month's inflation. When published, the IPCA-15 is a good indicator of what can be the current month's IPCA inflation rate, because this index has already covered the first half of the month.

Monitor IPCA-15 Ponta is the analogous Ponta version for the IPCA-15.

3 Methodology

After collection of the time series data of the features, the next steps of the research are *data preparation* and *data cleaning*. The goal of *data preparation* step of this work is to construct a balanced panel where all the variables (the target and features) have daily frequency. This interpolation of monthly and weekly series to daily frequency is accomplished using the Kalman Filter in Section 4. As for the *data cleaning*, the calendar was created, eliminating all the weekends and national holidays, then some observations were deleted (for example, the Monitors have some data on weekends) and some missing data in daily series were inferred (for example, if a daily series has data in dates t and $t+2$, the data in $t+1$ was filled in as a simple arithmetic mean between them). Moreover, although the ML methods are powerful, some of them require specific data format, such as the series be stationary. All the series were also standardized to have the same scale (it is needed by Factor and Shrinkage Models). To test stationarity, the augmented Dickey–Fuller test (ADF) was used. Table A in the Appendix shows which of the following transformations was applied to each series: (0) no transformation; (1) Δx_t; (2) $\Delta^2 x_t$; (3) $\log x_t$; (4) $\Delta \log x_t$; (5) $\Delta^2 \log x_t$; (6) $\Delta^3 \log x_t$; (7) $\Delta\Big(\frac{x_t}{x_{t-1}} - 1\Big)$.

The next step of the research is the *modeling* itself, that is, using the Database, the goal is to create Models that, given new data on the features, are capable of nowcasting the value of the target so that the error of the nowcast is as small as possible. This is the *learning* stage: the machine is learning the correct answers that were given, in the past, to the following question: "Given those feature values, what was the IPCA inflation rate in that date?" to be able to answer it in the present when there is no known answer. The whole dataset is divided in two parts: the learning sample (from 01/12/2006 to 08/10/2014), which has 1974 observations (approximately two thirds of the whole sample) and on which the learning is performed, and the testing sample (from 09/10/2014 to 03/10/2018), which has 1000 observations.

The next step is *testing (assessment)* stage, which is performed over the testing sample.

The assessment of models' performance here is different from that of Medeiros et al. (2019) in the following sense: For every ML technique, Medeiros et al. (2019) run 12 models, corresponding to each forecasting horizon (in months) $h = 1, 2, ..., 12$, so they present 12 RMSE (MAE) results for each technique. Here, this can not be performed because the exercise is not forecasting, but is *nowcasting*, which implies that, during the month, the researcher does not observe the IPCA rate, so can not compare the predicted value with the realized value *online*, as it is made in Medeiros et al. (2019), where, in each line, the researcher observes the IPCA rate (since the database is monthly). For example, on 17/08/2016, the researcher nowcasts the value for the month inflation rate, but to assess the performance of the model, this number can only be compared to the official IPCA rate published on 08/09/2016. Therefore, in this work the following procedure was adopted: firstly, it was verified that each month in Testing Sample has, *on average*, 21 days. Then, for every ML technique, 20 models were run, corresponding to each forecasting horizon (in days) $h = 1, 2, ..., 20$. The Testing Sample has 1000 daily observations, each one indexed by $t = 1, 2, ..., 1000$. From this sequence of indexes, define the subsequence $t_j = 21, 42, 63, ..., 966, 987$; hence $j = 1, 2, ..., 47$. Evidently, on each day indexed by t_j, the true IPCA rate becomes publicly known, and there is no need to nowcast it on that day. Hence, to assess the nowcasting performance of the ML techniques, the following quality functionals are defined:

$$RMSE(\hat{y}_{h,t_j}, y_j) = \sqrt{\frac{1}{940}\sum_{h=1}^{20}\sum_{j=1}^{47}(\hat{y}_{h,t_j} - y_j)^2} \tag{1}$$

and

$$MAE(\hat{y}_{h,t_j}, y_j) = \frac{1}{940}\sum_{h=1}^{20}\sum_{j=1}^{47}\left|\hat{y}_{h,t_j} - y_j\right|, \tag{2}$$

where $\hat{y}_{h,t_j}$ is the prediction of the IPCA rate made h days ago for the date t_j, and y is the official IPCA rate published on the day j (note that every month the publishing date is different, but it is possible to affirm that, *on average*, the publishing day is j). Thus, differently from Medeiros et al. (2019), in the present work each ML technique is assessed not by a vector of 12 errors, but by an integer that represent the mean error of 20 models corresponding to forecasting horizons $h = 1, 2, ..., 20$.

Following Medeiros et al. (2019), the nowcasting exercise performed in this paper also uses

the *rolling window* technique. Its main advantage is that it smoothes the effects of possible structure breaks and outliers in the series. The rolling windows have fixed size m within each model, but it differs across the models, because it depends on the forecasting horizon h and on the lag p of the model:

$$m(h,p) = 1974 - h - p - 1,$$

where 1974 is the size of the Learning Sample. Hence, the number of subsamples in the rolling window scheme is given by $N = 2974 - m(h,p) + 1$.

Computer codes used in Section 4 were written in Python 3.7, Anaconda distribution. The codes used in Section 5 for implementation of ML methods, are in R language and use the same packages and functions as Medeiros et al. (2019)[5].

4 Kalman Filter

For the sake of exposition, inflation x is thought of as an object moving in two-dimensional space: one dimension is the IPCA rate and the other is the IPCA-15 rate. It moves continuously, nevertheless the economist observes its position in $\mathbb{R}^2$ only once a month, when the IBGE publishes the official statistics. However, the economist wants to track the moving of this object every day (nowcast it). To do this job, he receives every day a signal – the Inflation Monitor[6] – that shows approximately where the target object is, but this Monitor data has a lot of noise. The Kalman Filter (KF)[7] is an unsupervised algorithm that helps to reduce this uncertainty about where the inflation is now (it filters noisy data). Below follows the explanation of how the KF works and how it was used in this research to make the target variable and the monthly or weekly features all be of daily frequency.

Step 1. Initial belief (prior)

Suppose, without loss of generality, that today is 09/01/2012. The economist does not know what are the IPCA and IPCA-15 rates today (the true location of x in $\mathbb{R}^2$), but he must form some belief about it. Three days ago (in 06/01/2012), the IBGE published the official IPCA rate of December/2011: it was 0,50%. Another official data that he knows is the IPCA-15 rate of December/2011, published in 21/12/2011 by the IBGE: it was 0,56%. So this economist's initial guess $\hat{x}_0$ is that today the inflation must be in some region nearby the point $\hat{x}_0 = \begin{pmatrix} 0,50 \\ 0,56 \end{pmatrix} \in \mathbb{R}^2$. The Figure 1 illustrates it. The assumption of the classical (linear) KF algorithm is that x has Normal probability density (in our case, bivariate). Denote $p = N(\hat{x}_0, \Sigma_0)$ as the prior for x, where the initial guess for the covariance matrix let be $\Sigma_0 = \begin{pmatrix} 0,4 & 0,3 \\ 0,3 & 0,45 \end{pmatrix}$. The colored areas in Figure 1 show probability levels of the object being

[5] See https://github.com/gabrielrvsc

[6] More generally, the signal may be n-dimensional, where n is number of daily features.

[7] For this topic, the references are Harvey(1989) and Hamilton(1994). For the sake of succinct exposition of the KF, the material from https://python.quantecon.org/kalman.html was consulted.

located in that area, e.g., $0,48 = \int_M p(x)dx$ is the probability of the inflation being in the central (Maroon) ellipse.

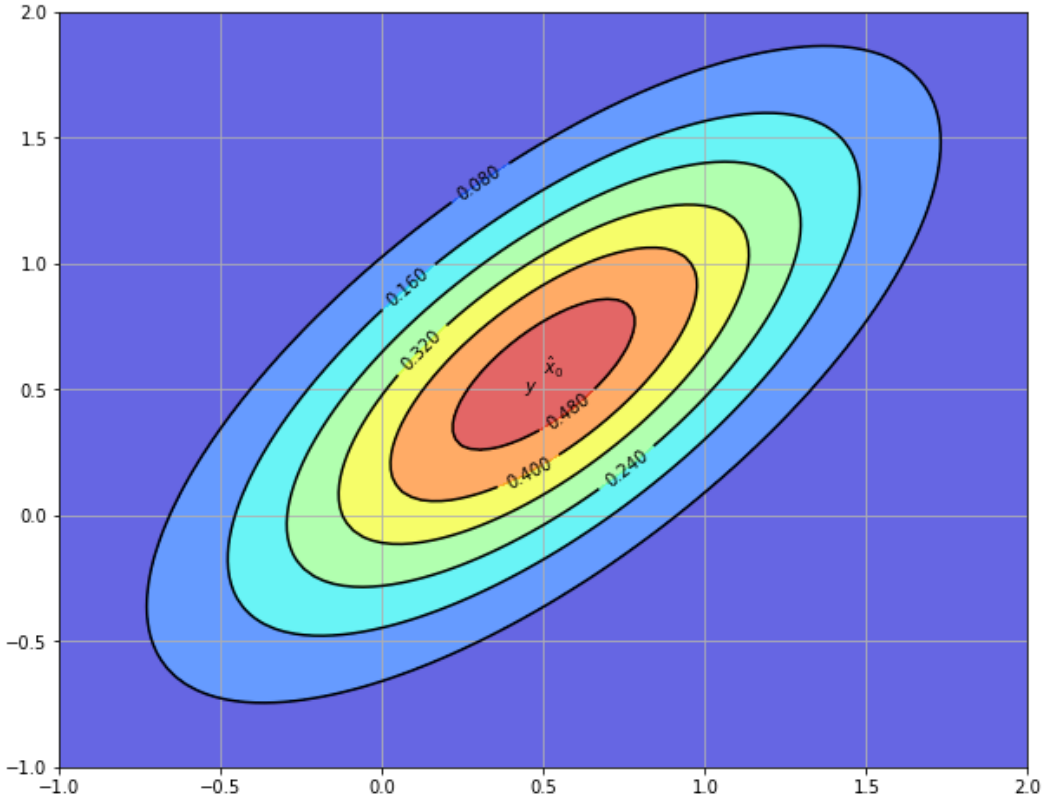


Figure 1: Prior density of the random variable x.

Step 2. Filtering

Now suppose that the economist received new information: today's inflation Monitor data, which is noisy but delivers a signal of where the inflation can be today. Denote $y = \begin{pmatrix} \text{Monitor IPCA} \\ \text{Monitor IPCA-15} \end{pmatrix}$. In 09/01/2012, the economist observed $y = \begin{pmatrix} 0,44 \\ 0,49 \end{pmatrix}$. This point is depicted in Figure 1. But this observed sensor is imprecise, so it is assumed that

$$y_t = Cx_t + d_t + \epsilon_t \qquad \text{[Observation Equation]} \tag{3}$$

where C is 2×2 observation matrix, d_t is observation offset (by default, set it to zero) and $\epsilon_t \sim N(0, R)$ is independent of x, where $\underset{(2\times2)}{R}$ is called observation covariance matrix and is positive definite. For now, assume that C is identity matrix and $R = 0,5 \cdot \Sigma_0$.

Kalman Filter is a bayesian method: it uses the new information y to update the prior $p(x)$ according to the Bayes' Rule:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int p(y|x)p(x)dx}.$$

By (3), $p(y|x) = N(Cx, R)$. Using the formulas for marginal and conditional distributions of Gaussian variables (Bishop(2006), p.93) and also the Woodbury matrix identity

$$(A+UCV)^{-1} = A^{-1} - A^{-1}U\left(C^{-1}+VA^{-1}U\right)^{-1}VA^{-1},$$

the solution is $p(x|y) = N(\hat{x}^F, \Sigma^F)$, where

$$\hat{x}^F = \hat{x}_0 + \Sigma_0 C^T (C\Sigma_0 C^T + R)^{-1}(y - C\hat{x}_0) = \begin{pmatrix} 0,46 \\ 0,51 \end{pmatrix} \tag{4}$$

and

$$\Sigma^F = \Sigma_0 - \Sigma_0 C^T (C\Sigma_0 C^T + R)^{-1} C\Sigma_0 = \begin{pmatrix} 0,13 & 0,1 \\ 0,1 & 0,15 \end{pmatrix}. \tag{5}$$

Equation (4) tells that the new position $\hat{x}^F$ is the initial guess $\hat{x}_0$ corrected by the news $y - C\hat{x}_0$ weighted by the matrix $\Sigma_0 C^T (C\Sigma_0 C^T + R)^{-1}$. Equation (5) tells that new information y must reduce the uncertainty by the amount of $\Sigma_0 C^T (C\Sigma_0 C^T + R)^{-1} C\Sigma_0$. In fact, this can be observed in Figure 2, where the filtering distribution $p(x|y)$ is depicted. The contour lines of the prior distribution appear in black, while those of the new density are colored. Note that the uncertainty (the areas of ellipses) reduced significantly, and the mean shifted towards the point y indicated by the Monitor.

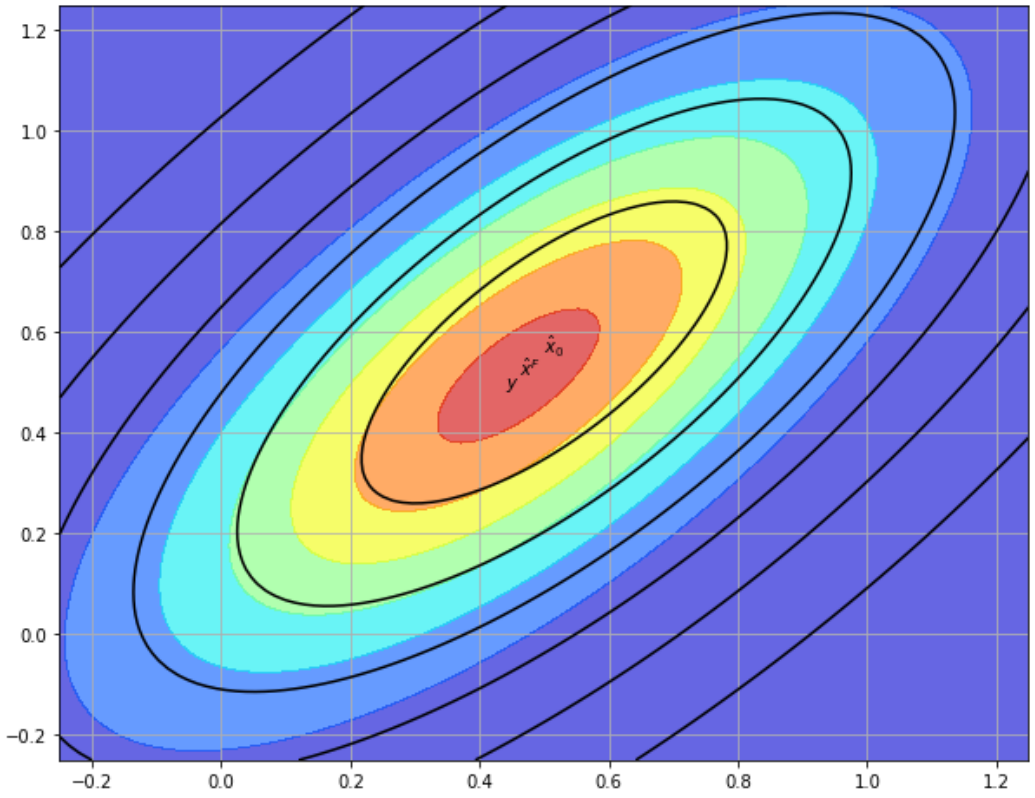


Figure 2: Filtering distribution $p(x|y)$.

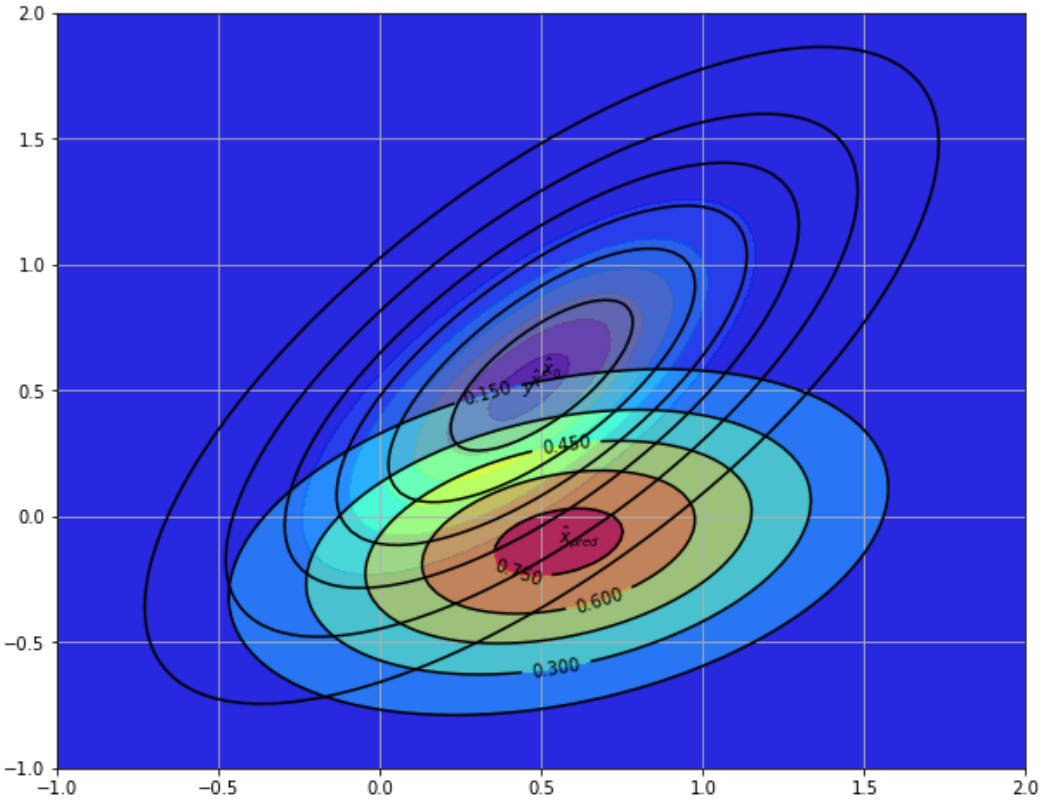


Figure 3: Predictive distribution $p_{pred}(x)$.

Step 3. Forecasting

Now suppose that the economist's next aim is to predict where the inflation will be tomorrow (or suppose that tomorrow the sensor data will be missing, and it will be necessary to nowcast inflation having only the data up to today). For that, it is needed to make assumption about the dynamics of the true inflation process x:

$$x_{t+1} = Ax_t + b_t + \varepsilon_{t+1} \qquad \text{[State Equation]}, \tag{6}$$

where $\underset{(2\times 2)}{A}$ is transition matrix, b_t is transition offset (by default, set it to zero) and $\varepsilon_{t+1} \sim N(0, Q)$ is independent of x, where $\underset{(2\times 2)}{Q}$ is called transition covariance matrix. For now, assume that $A = \begin{pmatrix} 1,2 & 0 \\ 0 & -0,2 \end{pmatrix}$ and $Q = 0,3 \cdot \Sigma_0$. Jointly, equations (3) and (6) form the so-called state-space model, that has wide usage in forecasting.

Hence, now, given the filtering distribution $p(x|y)$ and the law of motion (6), the task is to calculate predictive distribution $p_{pred}(x)$ of the location of inflation tomorrow. Substituting random vector $x^F \sim N(\hat{x}^F, \Sigma^F)$ in (6), it comes out that $(Ax^F + \varepsilon) \sim N(\hat{x}_{pred}, \Sigma_{pred})$ because it is linear combination of two Normal variables. Moreover,

$$\hat{x}_{pred} := \mathbb{E}[Ax^F + \varepsilon] = A\mathbb{E}x^F = A\hat{x}_0 + A\Sigma_0 C^T (C\Sigma_0 C^T + R)^{-1}(y - C\hat{x}_0) = A\hat{x}_0 + K_\Sigma(y - C\hat{x}_0)$$
$$= \begin{pmatrix} 0,55 \\ -0,1 \end{pmatrix},$$

where $K_\Sigma := A\Sigma_0 C^T (C\Sigma_0 C^T + R)^{-1}$ is called Kalman Gain; and

$$\Sigma_{pred} := \mathbb{V}[Ax^F + \varepsilon] = A\mathbb{V}[x^F]A^T + Q = A\Sigma_0 A^T - A\Sigma_0 C^T (C\Sigma_0 C^T + R)^{-1} C\Sigma_0 A^T + Q$$
$$= A\Sigma_0 A^T - K_\Sigma C\Sigma_0 A^T + Q = \begin{pmatrix} 0,31 & 0,066 \\ 0,066 & 0,14 \end{pmatrix}.$$

The predictive density $p_{pred}(x) = N(\hat{x}_{pred}, \Sigma_{pred})$ is depicted in Figure 3 in colors, in comparison with prior and filtering distributions.

Implementation

To obtain the target and feature variables in daily frequency, the above described Steps 1, 2 and 3 were implemented in *pykalman*[8] package in Python. As seen above, the main advantage of this KF is that it does not require any labeled training data neither demands any previous transformations of data (such as making it stationary etc); on the other hand, it has the computational cost of $O(Td^3)$, where T is total number of time steps and d is the dimension of the state space, and is not able to handle non-gaussian noise, according to the documentation of the package.

To interpolate monthly[9] variables to daily frequency via KF, this research used the following state-space framework[10]: observation equation is

$$\underset{(1\times 1)}{y_t} = \underset{(1\times 24)}{C_t} \underset{(24\times 1)}{\begin{pmatrix} x_t \\ x_{t-1} \\ \vdots \\ x_{t-22} \\ u_t \end{pmatrix}}, \tag{7}$$

where y_t is the interpoland series,

$$C_t = \begin{cases} \begin{bmatrix} 1/23 & 1/23 & \dots & 1/23 & 0 \end{bmatrix}, & t = 23, 46, 69, \dots, T \\ \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \end{bmatrix}, & \text{otherwise} \end{cases}$$

[8] See https://pykalman.github.io/

[9] For weekly variables, all the reasoning is analogous to that presented below, but the ragged edge gap between two releases of observation series becomes 5 days instead of 23.

[10] For references, see Bernanke et al. (1997) and Mönch & Uhlig (2005). Issler & Notini (2016) apply this technique to nowcast Brazilian GDP.

and transition equation is

$$x_{t+1} = \underset{(24\times1)}{\begin{pmatrix} x_{t+1} \\ x_t \\ x_{t-1} \\ x_{t-2} \\ \vdots \\ x_{t-21} \\ \varepsilon_{t+1} \end{pmatrix}} = \underset{(24\times24)}{\begin{pmatrix} \phi & 0 & 0 & \dots & 0 & 0 & \rho \\ 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & \rho \end{pmatrix}} \underset{(24\times1)}{\begin{pmatrix} x_t \\ x_{t-1} \\ x_{t-2} \\ x_{t-3} \\ \vdots \\ x_{t-22} \\ \varepsilon_t \end{pmatrix}} + \underset{(24\times1)}{\begin{pmatrix} z_t'\beta \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}} + \underset{(24\times1)}{\begin{pmatrix} \eta_{t+1} \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \eta_{t+1} \end{pmatrix}}, \quad (8)$$

where $\underset{(1\times k)}{z_t'}$ is a vector of k feature series with daily frequency that are used as covariates to increase the accuracy of the filter, $\underset{(k\times1)}{\beta}$ is a vector of weights for these covariates, and

$$\eta_{t+1} \sim N(0, \sigma^2), \quad \text{with} \quad \sigma^2 = 999999.$$

Equations (7)-(8) incorporate two main specifications to the framework (3) and (6):

(i) The transition equation error is generalized to allow for autoregressive structure:

$$\varepsilon_t = \rho\varepsilon_{t-1} + \eta_t;$$

(ii) The transition matrix C is specified in such a way that the following *sum restriction* must be satisfied:

$$y_t = \begin{cases} \frac{1}{23}\sum_{i=0}^{22} x_{t-i}, & t = 23, 46, 69, \dots, T \\ 0, & \text{otherwise.} \end{cases}$$

Thinking of y_t as IPCA inflation rate, this restriction imposes that the state inflation x_t that the Kalman algorithm generates every day during the month must be such that the mean of these states is *exactly* equal to the the official IPCA inflation rate that IBGE will publish relatively to that month. Thinking of y_t as any other monthly variable from the dataset, the reasoning is analogous. In order to respect this restriction, Kalman *Smoother* is implemented, because it estimates the states in a batch, while Kalman Filter performs only *online* estimation. As empirical results of Issler and Notini (2016) show, the use of this restriction in the in-sample period leads to better performance out-of-sample. Hence, Kalman Smoother is used to generate states in the Learning Sample and Kalman Filter is used in the Testing Sample.

Moreover, any month can have at most 23 working days, hence, to perform the interpolation exercise, the new calendar is created: additional days (holidays or weekends) are added to the months in the original dataset calendar that had less than 23 days. After obtaining the interpolation result, these additional observations are excluded, and the calendar is back to its original form.

When implementing the system (7)-(8) in *pykalman*, the values of y_t for $t \neq 23, 46, 69,$ $..., T$ are *missing* and the goal is to fill this ragged edge; to accomplish that, these cells

must be *masked*, so that the algorithm considers them as missing and makes *prediction* (Step 3 of the explanation above) using the *state equation* for all $t \neq 23, 46, 69, ..., T$.

As pointed out above, the KF assumes that the model parameters $\theta = (\hat{x}, \Sigma_0, C, d, R, A, b, Q)$ are previously estimated and can be specified by hand. These parameters define a probabilistic model from which the unobserved states and observed measurements are assumed to be sampled from. There are some possibilities concerning specification of θ.

The first possibility is to not specify θ at all (in this case, the code uses sensible default values: zeros for all 1-dimensional arrays and identity matrices for all 2-dimensional arrays) and use the Expected Maximization algorithm to learn θ while running the Kalman Filter or Smoother (as can be seen from Table 1, algorithms with EM perform better). The EM algorithm is implemented as follows (Hastie et al. (2001)):

- the algorithm starts with the initial guess for $\hat{\theta}^{(0)}$;
- *expectation step j*: compute $\mathbb{E}[\ell_0(\theta, \mathbf{T})|\mathbf{Y}, \hat{\theta}^{(j)}]$, where ℓ is log-likelihood function, $\mathbf{T} = (\mathbf{Y}, \mathbf{X})$, $\mathbf{Y}$ are observed variables and $\mathbf{X}$ are unobserved ones;
- *maximization step j+1*: find $\hat{\theta}^{(j+1)} \in argmax\, \mathbb{E}_\theta[\ell_0(\theta, \mathbf{T})|\mathbf{Y}, \hat{\theta}^{(j)}]$;
- iterate two previous steps until convergence.

To avoid overfitting, number of iterations is set to 5. By Jensen's inequality, each new iteration never decreases the log-likelihood of the observed data; however, this is a nonconvex optimization problem, so when the algorithm converges, nothing guarantees that it converges to the global (and not only local) extremum.

The second possibility is to specify θ according to (7)-(8) and to learn ϕ, β and ρ on the learning sample and then use these estimations to run the filter on the testing sample. It is worth reiterating that the sum restriction is not satisfied on the Testing Sample; however, using this restriction on the Learning Sample not only improves the quality of interpolation there, but also helps to estimate more accurately parameters that will be used to interpolate the Testing Sample.

To initialize the values of ϕ, β and ρ in the algorithm, the daily and weekly features in z_t are aggregated to monthly frequency, and the following regressions are run[11]:

$$\begin{aligned} (1 - \phi L) y_t &= z_t' \beta + \varepsilon_t, \\ \varepsilon_t &= \rho \varepsilon_{t-1} + \eta_t, \end{aligned} \tag{9}$$

where y_t are monthly observed series. After that, the estimated values of these parameters are initialized in the Kalman Smoother algorithm, and then the Expected Maximization mechanism optimizes them at each iteration step t.

After interpolation of Learning Sample, it is necessary to specify θ for Kalman Filter that will be run on Testing Sample. To initialize θ, the interpolated Learning Sample is used to search for the optimal values of ϕ and ρ. For example, the following algorithm was used for

[11] Using SARIMAX class of package *statsmodels* (see https://www.statsmodels.org/) in Python.

this purpose on the IPCA target series: Fix $\rho = 0$ and create a grid in the interval $[-2; 2]$, with 121 potential values for ϕ. Run a KS loop 121 times and then choose ϕ_* such that

$$\phi_* = \underset{\phi}{argmin}\ RSE = \underset{\phi}{argmin} \sqrt{\sum_{\substack{t=27/12/2006,\\ 29/01/2007,\\ ...,\\ 29/09/2014}} (\pi_t - m_t)^2}, \tag{10}$$

and

$$\phi_* = \underset{\phi}{argmin}\ AE = \underset{\phi}{argmin} \sum_{\substack{t=27/12/2006,\\ 29/01/2007,\\ ...,\\ 29/09/2014}} |\pi_t - m_t|, \tag{11}$$

where RSE and AE are Root Squared Error and Absolute Error, respectively, π_t is official IPCA rate published by IBGE, and m_t is the daily state inflation calculated by the KS. When the optimal ϕ according to RSE criterion is different from the optimal ϕ according to AE, take the mean of these ϕ's. As can be seen from the indexes of summation, it is considered that the best KS result is that is the "nearest" to the true IPCA inflation rate. Since the collection period of IPCA ends approximately on the 27th day of each month, hence the KS result on that day that produces the least error when compared to the true IPCA rate for that month is considered the one where ϕ is optimal. After that, fixing the optimal ϕ, introduce autoregressive observation error to the model and search for the optimal ρ analogously. The interval for the grid was chosen to be $[-2; 2]$ because numbers out of this interval produce infinite errors. The initial value of β was that estimated by (9). The initial state means $\hat{x}$ were set equal to the last observed value of the interpoland variable, both in the Learning Sample Smoother and in the Testing Sample Filter.

Instead of interpolating the IPCA inflation rate using this method with sum restriction, the researcher could only use the Monitor IPCA (without applying any filter to it) as state inflation; he could also apply Kalman Smoother to Learning Sample and Kalman Filter to Testing Sample of several Monitors and daily covariates, without using the sum restriction or autoregressive error, but using the default parameters for θ. Table 1 shows the errors produced by these different alternatives and the conclusion is that using the sum restriction reduces the RSE (AE) in 13% (46%) in comparison with using the Monitor IPCA as daily state inflation rate and reduces the RSE (AE) in 36% (46%) in comparison with smoothing and filtering all the Monitors and another 14 daily covariates without the use of the sum restriction. The errors presented are total errors, that are calculated summing the error on the learning sample produced by the smoother and the error on the testing sample produced by the filter. The model applied in the last line of the Table 1 employs 18-dimensional (all the Monitors + 14 daily covariates) Kalman Smoother with the sum restriction (using the loop described above, the optimal values for parameters ϕ and ρ were found to be $\phi_* = 0,4$ and $\rho_* = 0$) with EM in the Learning Sample and the 18-dimensional Kalman Filter with EM in the Testing Sample. As shows the graph in Figure 4, the interpolated to daily frequency IPCA inflation rate accompanies well the official monthly IPCA rate. As expected, the nowcasting error of daily interpolated series is higher in the Testing Sample than in the Learning Sample.

Table 1: Root Squared Errors and Absolute Errors of the Monitor, Filtered/Smoothed Monitors and KF/KS model with sum restriction.

Time series	**RSE (AE)**
Monitor IPCA	1,03 (10,09)
1-dimensional KF (Monitor IPCA) without EM	1,45 (10,15)
1-dimensional KS/KF (Monitor IPCA) with EM	1,44 (10,10)
2-dimensional KS/KF (Monitor IPCA and Monitor IPCA-15) without EM	1,45 (10,15)
2-dimensional KS/KF (Monitor IPCA and Monitor IPCA-15) with EM	1,44 (10,02)
2-dimensional KS/KF (Monitor IPCA and Monitor IPCA Ponta) without EM	1,45 (10,15)
2-dimensional KS/KF (Monitor IPCA and Monitor IPCA Ponta) with EM	1,44 (10,05)
4-dimensional KS/KF (all the Monitors) without EM	1,45 (10,15)
4-dimensional KS/KF (all the Monitors) with EM	1,44 (10,04)
18-dimensional KS/KF (all the Monitors + 14 daily covariates) without EM	1,45 (10,15)
18-dimensional KS/KF (all the Monitors + 14 daily covariates) with EM	1,44 (10,04)
18-dimensional KS/KF with sum restriction with EM	**0,9 (5,45)**

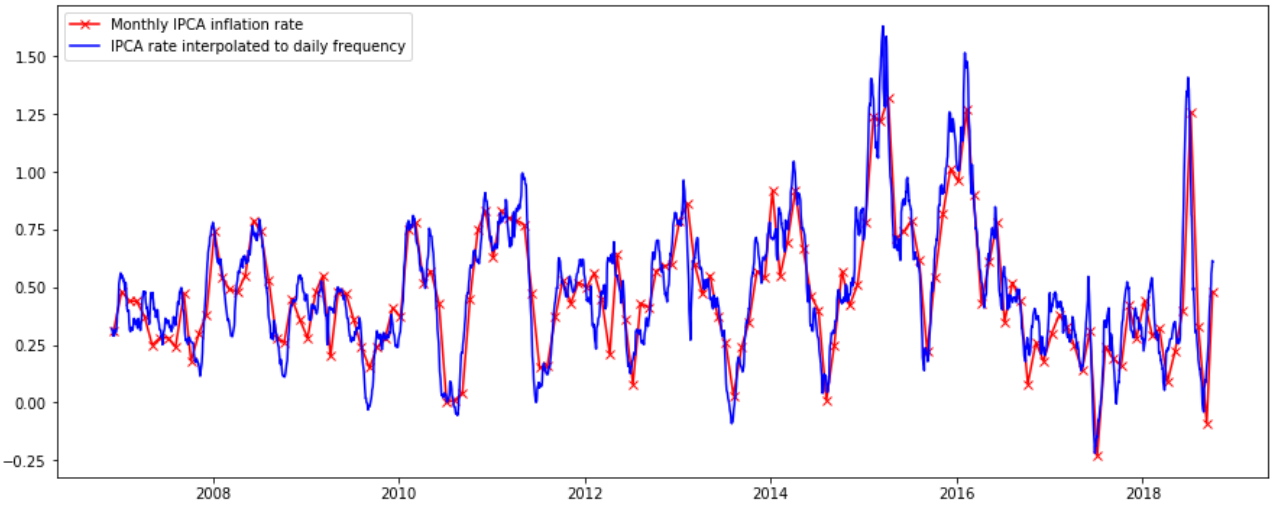


Figure 4: Monthly IPCA inflation rate and IPCA rate interpolated to daily frequency using Kalman Smoother and Filter.

5 Models

The present work considers 2 Benchmark models, 3 Factor models, 5 Shrinkage methods, 4 Averaging (Ensemble) methods and 2 Hybrid models. This Section makes an exposition of these Models[12], following Medeiros et al. (2019).

5.1 Benchmarks

5.1.1 Random Walk (RW)

Let π_t denote IPCA monthly inflation rate at date t, and $h = 1, .., 20$ be forecasting horizon (in days). RW model resembles the adaptive expectations theory in economics:

$$\pi_t = \pi_{t-1} + \epsilon_t, \tag{12}$$

where $\{\epsilon_t\}$ iid $(0, \sigma^2)$. The model in (12) is without drift. Hence, inflation forecast h periods ahead is

$$\widehat{\pi}_{t+h} := E_t(\pi_{t+h}) = \pi_t. \tag{13}$$

5.1.2 Autoregressive model (AR)

Another univariate benchmark model is Autoregressive Model (AR) with lag p and h prediction horizons:

$$\pi_t = \varphi_{0,h} + \varphi_{1,h}\pi_{t-1} + \varphi_{2,h}\pi_{t-2} + ... + \varphi_{p,h}\pi_{t-p} + \epsilon_t, \tag{14}$$

where $E(\epsilon_t) = 0$, $V(\epsilon_t) = \sigma^2$ $\forall$ t and $E(\epsilon_t\epsilon_\tau) = 0$ for $t \neq \tau$. The lag p is defined using the Bayesian Information Criterion (BIC) and the parameters are estimated by Ordinary Least Squares. Thus, the forecast h periods ahead is

$$\widehat{\pi}_{t+h} = \widehat{\varphi}_{0,h} + \widehat{\varphi}_{1,h}\pi_t + \ldots + \widehat{\varphi}_{p,h}\pi_{t-p+1}. \tag{15}$$

5.2 Factor Models

5.2.1 Dynamic Factor Model with Principal Component Analysis

Dynamic Factor Model (DFM) is an important unsupervised technique of dimensionality reduction. In macroeconometrics, researchers frequently deal with object-feature matrices where $T \ll N$, that is, the number of features N is far greater than the number of time series observations T. Implementing DFM, a great part of variation of these observed variables can be summarized in the dynamics of a small number of unobserved (latent) common factors. Differently from Shrinkage methods, the Factor Models do not discard uninformative

[12] A *model* (or *algorithm*) is a function $a : \mathbb{X} \to \mathbb{Y}$, that for every object $\underset{(N\times 1)}{X_t}$ returns the predicted target variable value $\hat{\pi}_{t+h}$.

features; indeed, they condense all the features in a joint parsimonious structure that has good asymptotic properties.

Dynamic Factor Model can be written in two equivalent forms: the *dynamic* form, where the vector of observed features $\underset{(N\times 1)}{X_t}$ depends on the lags of factors explicitly, and the *static* form, where this dynamics is implicit (Stock and Watson (2016)). The two forms are exposed below. The lag operator notation is used: $a(L) := \sum_{i=0}^{\infty} a_i L^i$, and so $a(L)X_t = \sum_{i=0}^{\infty} a_i X_{t-i}$.

Dynamic form of DFM:

$$\begin{aligned} \underset{(N\times 1)}{X_t} &= \underset{(N\times q)(q\times 1)}{\lambda(L)\, f_t} + \underset{(N\times 1)}{e_t} \\ \underset{(q\times 1)}{f_t} &= \underset{(q\times q)\ (q\times 1)}{\Psi(L)\, f_{t-1}} + \underset{(q\times 1)}{\eta_t} \end{aligned} \tag{16}$$

where $q \ll N$, the polynomial matrix $\lambda(L)$ is referred to as matrix of factor *loadings* and $\lambda_i(L)f_t$ is called *common component* of the ith series, $i = 1, 2, ..., N$. Also, η_t is zero-mean serially uncorrelated vector of innovations to the factors, and it is assumed that $Ee_t\eta'_{t-k} = 0 \quad \forall k \in \mathbb{Z}$. Moreover, the (zero-mean) idiosyncratic disturbances vector e_t is generally assumed to be serially correlated according to

$$e_t = \delta(L)e_{t-1} + \upsilon_t, \tag{17}$$

where υ_t is serially uncorrelated.

Letting p be the degree of lag polynomial matrix $\lambda(L)$, it is possible to rewrite (16) in the *companion form* (18), obtaining the

Static form of DFM:

$$\underset{(N\times 1)}{X_t} = \underbrace{\begin{bmatrix} \underset{(N\times q)}{\lambda_0} & \underset{(N\times q)}{\lambda_1} & \cdots & \underset{(N\times q)}{\lambda_{p-1}} \end{bmatrix}}_{:=\ \underset{(N\times qp)}{\Lambda}} \underbrace{\begin{pmatrix} f_t \\ f_{t-1} \\ \vdots \\ f_{t-p+1} \end{pmatrix}}_{:=\ \underset{(qp\times 1)}{F_t}} + \underset{(N\times 1)}{e_t} = \Lambda F_t + e_t$$

$$\begin{aligned} \underbrace{\begin{pmatrix} f_t \\ f_{t-1} \\ \vdots \\ f_{t-p+1} \end{pmatrix}}_{\underset{(qp\times 1)}{F_t}} &= \underbrace{\begin{bmatrix} \Psi_1 & \Psi_2 & \cdots & \Psi_{p-1} & \Psi_p \\ \mathbb{I}_{(q)} & 0_{(q)} & \cdots & 0_{(q)} & 0_{(q)} \\ 0_{(q)} & \mathbb{I}_{(q)} & \cdots & 0_{(q)} & 0_{(q)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0_{(q)} & 0_{(q)} & \cdots & \mathbb{I}_{(q)} & 0_{(q)} \end{bmatrix}}_{:=\ \underset{(qp\times qp)}{\Phi(L)}} \underbrace{\begin{pmatrix} f_{t-1} \\ f_{t-2} \\ \vdots \\ f_{t-p} \end{pmatrix}}_{\underset{(qp\times 1)}{F_{t-1}}} + \underbrace{\begin{bmatrix} \mathbb{I}_{(q)} \\ 0_{(q)} \\ \vdots \\ 0_{(q)} \end{bmatrix}}_{:=\ \underset{(qp\times q)}{G}} \eta_t \\ &= \Phi(L)F_{t-1} + G\eta_t, \end{aligned} \tag{18}$$

where Ψ_1, Ψ_2, ..., Ψ_p are $(q \times q)$ matrices of loadings such that the state equation in (16) holds when rewritten in terms if F_t.

Working with *dymanic* DFM implies parametric estimation, while rewriting it to *static* form leaves to nonparametric estimation (such as PCA), that is, without imposing any structure

on transition equation in (18) and without assuming any functional form, such as (17), for disturbances. Instead, only the data $\underset{(N\times 1)}{X_t}$ is used to estimate $\underset{(r\times 1)}{F_t}$, where $r = qp$ such that $q \leq r \ll N$. The present work employs static form of DFM to make Principal Components estimation of factors, following Stock & Watson (2002a, 2011, 2016).

Consider estimating F_t by *cross-sectional averaging* of X_t, where the weighted average of X_t is calculated using a matrix of weights $N^{-1}\Lambda'$, that is,

$$\hat{F}_t = N^{-1}\Lambda' X_t. \tag{19}$$

Cross-sectional averaging is used to eliminate influence of idiosyncratic disturbances e_t on X_t, preserving only variation attributed to factors. This elimination is possible under two hypothesis made by Chamberlain and Rothschild (1983):

(i) $N^{-1}\Lambda'\Lambda \xrightarrow[N\to\infty]{} \underset{(r\times r)}{D_\Lambda}$, where D_Λ has full rank;

(ii) $maxeval(\Sigma_e) \leq c < \infty \quad \forall N$, where *maxeval* denotes the maximum eigenvalue, and $\Sigma_e = E(e_t e_t')$.

Condition (i) guarantees that the factors are pervasive, that is, that they affect most or all of the features, and that factor loadings are heterogeneous, that is, the columns of Λ are not too similar. Condition (ii) assures that cross-sectional correlation among the idiosyncratic disturbances $\{e_{it}\}$ is limited (Stock and Watson (2011, 2016)).

Hence, (ii) assures that, by (weak) Law of Large Numbers, $N^{-1}\Lambda' e_t \xrightarrow[N\to\infty]{P} 0$, and, using (i), it comes out that $N^{-1}\Lambda' X_t - N^{-1}\Lambda'\Lambda F_t \xrightarrow[N\to\infty]{P} 0$. Thus, $N^{-1}\Lambda' X_t$ asymptotically spans the space of factors. However, the weights $N^{-1}\Lambda$ are infeasible because Λ is unknown. That is when the Principal Component Analysis comes into play: in the weighted averaging estimator (19), instead of Λ, use $\hat{\Lambda}$ (computing, thus, the sample version of this weighted average), where $\hat{\Lambda}$ is the matrix of eigenvectors of the sample variance-covariance matrix of X_t, $\hat{\Sigma}_X = T^{-1}\sum_{t=1}^{T} X_t X_t'$, associated with r largest eigenvalues of $\hat{\Sigma}_X$. Note that these $\hat{F}$ and $\hat{\Lambda}$ can be derived as solutions to the following least squares problem:

$$\begin{aligned} \underset{F_1,F_2,\ldots,F_T,\Lambda}{min} \quad & \frac{1}{NT}\sum_{t=1}^{T}(X_t - \Lambda F_t)'(X_t - \Lambda F_t) \\ s.t. \quad & N^{-1}\Lambda'\Lambda = \mathbb{I}_r. \end{aligned} \tag{20}$$

The constraint in (20) imposes orthogonality on Λ to preclude infinite solutions to the optimization problem. To solve (20), first consider Λ as known and derive the OLS estimator for F_t, which is

$$\hat{F}_t = (\Lambda'\Lambda)^{-1}\Lambda' X_t = N^{-1}\Lambda' X_t. \tag{21}$$

Secondly, substitute (21) in (20), and note that the problem can be written as:

$$\begin{aligned} \underset{\Lambda}{max} \quad & \Lambda'\hat{\Sigma}_X\Lambda \\ s.t. \quad & N^{-1}\Lambda'\Lambda = \mathbb{I}_r, \end{aligned} \tag{22}$$

which solution is $\hat{\Lambda}$. Having calculated $\hat{\Lambda}$, substitute it in (21) to obtain $\hat{F}_t$.

Considering (22), note that $\Lambda'\hat{\Sigma}_X\Lambda = D$, where $\underset{(r\times r)}{D}$ is diagonal matrix of the eigenvalues of $\hat{\Sigma}_X$, and Λ is a matrix of eigenvectors associated with those eigenvalues. As the objective is to find Λ that maximizes D, the program searches for r eigenvectors associated with r *largest* eigenvalues of $\hat{\Sigma}_X$ (the matrix $\hat{\Sigma}_X$ has, in total, N eigenvalues). The eigenvalues in D at the solution are disposed decreasingly, that is, d_{11} is the largest eigenvalue, d_{22} is the second largest and so on. Let us consider this in some details (Dhrymes (1974)).

The task of an economist is to nowcast inflation rate π_t, which depends on observable features $\underset{(N\times 1)}{X_t}$; however, when N is very large, he wants to reduce the dimensionality of X_t, but this reduction must preserve the most of the variability of X_t. So, he searches for r (mutually uncorrelated) linear combinations of the elements of X_t that capture most of their variability. These linear combinations are called *principal components*.

To search for a linear combination $f_1 = \lambda'_{(1)}X_t$ that maximizes the sample variance of the elements of X_t, where $\underset{(N\times 1)}{\lambda_{(1)}}$ is a vector of weights, one solves the following problem:

$$\begin{aligned} \underset{\lambda}{max} \quad & \lambda'\hat{\Sigma}_X\lambda \\ s.t. \quad & \lambda'\lambda = 1, \end{aligned} \tag{23}$$

which has the following Lagrangian associated:

$$\mathcal{L} = \lambda'\hat{\Sigma}_X\lambda + \mu_1(1 - \lambda'\lambda), \tag{24}$$

and the following First-Order Conditions:

$$\begin{aligned} \frac{\partial\mathcal{L}}{\partial\lambda} &= 2\hat{\Sigma}_X\lambda_{(1)} - 2\mu_1\lambda_{(1)} = 0 \quad \therefore \quad \hat{\Sigma}_X\lambda_{(1)} = \mu_1\lambda_{(1)} \\ \frac{\partial\mathcal{L}}{\partial\mu} &= 1 - \lambda'_{(1)}\lambda_{(1)} = 0. \end{aligned} \tag{25}$$

From (25), it is clear that μ_1 must be one of eigenvalues of $\hat{\Sigma}_X$; moreover, as the objective function $\lambda'\hat{\Sigma}_X\lambda = \mu_1$, then this problem maximizes μ_1 and so the solution vector $\lambda_{(1)}$ is the eigenvector associated with the largest eigenvalue of $\hat{\Sigma}_X$. The linear combination $f_1 = \lambda'_{(1)}X_t$ is called the first principal component of X_t, and the eigenvector $\lambda_{(1)}$ is stored in the first column of the matrix $\hat{\Lambda}$.

To search for another linear combination $f_2 = \lambda'_{(2)}X_t$ that also maximizes the sample variance of the elements of X_t, but is uncorrelated with the combination f_1, one solves the analogous problem, but with additional restriction that assures this uncorrelatedness:

$$\begin{aligned} \underset{\lambda}{max} \quad & \lambda'\hat{\Sigma}_X\lambda \\ s.t. \quad & \lambda'\lambda = 1 \quad \text{and} \quad cov(f_1, f_2) = \lambda'_{(1)}\hat{\Sigma}_X\lambda = 0. \end{aligned} \tag{26}$$

The solution of the Lagrangian associated to (26) is the vector $\lambda_{(2)}$, which is the eigenvector associated with the second largest eigenvalue of $\hat{\Sigma}_X$. The linear combination $f_2 = \lambda'_{(2)} X_t$ is called the second principal component of X_t, and the eigenvector $\lambda_{(2)}$ is stored in the second column of the matrix $\hat{\Lambda}$.

This procedure is repeated r times, until obtaining $\underset{(N\times r)}{\hat{\Lambda}}$ and $\underset{(r\times 1)}{\hat{F}_t}$. If the economist admits the following relation between the inflation rate and the factors:

$$\pi_{t+h} = \beta'_F F_t + \beta'_w w_t + \varepsilon_{t+h}, \tag{27}$$

then, the nowcast of the inflation rate is calculated as

$$\hat{\pi}_{T+h} = \hat{\beta}'_F \hat{F}_T + \hat{\beta}'_w w_T, \tag{28}$$

where $\underset{(M\times 1)}{w_t}$ is a vector of observed variables (e.g. lags of π_t) that help to improve the nowcasting; $\underset{(r\times 1)}{\hat{\beta}_F}$ and $\underset{(M\times 1)}{\hat{\beta}_w}$ are vectors of weights that were estimated by applying OLS to (27) using the estimated factors $\hat{F}_t$; and ε_{t+h} is the resulting zero-mean nowcasting error[13]. The economist makes this nowcast (28) at T and has the data available for $\{\pi_t, X_t, w_t\}_{t=1}^{T}$.

Stock and Watson (2002a) demonstrated that principal components of X_t have good asymptotic properties. Firstly, they are consistent estimators of the true latent factors F when $N \to \infty, T \to \infty$. Secondly, the feasible forecast $\hat{\pi}_{T+h}$ in (28) converges (as $N \to \infty, T \to \infty$) to infeasible forecast what would be obtained if Λ and F were known. This means that the feasible forecast is first-order asymptotically efficient.

To finalize this concise exposition of DFM with PCA, let us prove a proposition (Dhrymes (1974)) that highlights the importance of standardizing the dataset before implementing this model, for not to assign different weights to the series due to measurement units differences.

Proposition 1. *Principal components are not independent of the units in which the elements of X_t are measured.*

Proof. Given a vector of observable features $\underset{(N\times 1)}{X_t}$, consider another vector $\underset{(N\times 1)}{X_t^*}$ whose elements differ from those of X_t only in their scale of measurement: $X_t^* = UX_t$, where $U = diag(u_1, u_2, ..., u_N)$. Then, the variance-covariance matrix of X_t^* is $U\hat{\Sigma}_X U$, the ith principal component of X_t^* is $f_i^* = \lambda^{*\prime}_{(i)} X_t^*$, and thus $U\hat{\Sigma}_X U\lambda^*_{(i)} = \mu_i^* \lambda^*_{(i)}$, where $i = 1, 2, ..., N$ and $\lambda^*_{(i)}$ is the eigenvector associated to the eigenvalue μ_i^* of the matrix $U\hat{\Sigma}_X U$. Note that $\prod_{i=1}^{N} \mu_i^* = det(U\hat{\Sigma}_X U) = |U|^2 |\hat{\Sigma}_X| \neq |\hat{\Sigma}_X| = \prod_{i=1}^{N} \mu_i$. Hence, $\exists i \in \{1, 2, ..., N\}$ such that $\mu_i \neq \mu_i^*$.

Now, $f_i = f_i^*$ if and only if $\lambda'_{(i)} X_t = \lambda^{*\prime}_{(i)} UX_t$, which implies $\lambda_{(i)} = U\lambda^*_{(i)}$. But this is contradiction with the fact that, by construction, $\lambda_{(i)}$ and $\lambda^*_{(i)}$ are both orthonormal vectors. □

In the present work, the computational implementation of the DFM/PCA model described above considered $q = 4, p = 4$, thus $r = 16$; and $M = 4$, where

[13] The measurement equation in (18) and the equation (27) constitute what is known in literature on economic forecasting as the *diffusion index* framework.

$$w_t = \begin{pmatrix} \pi_t \\ \pi_{t-1} \\ \pi_{t-2} \\ \pi_{t-3} \end{pmatrix}. \tag{29}$$

5.2.2 Target Factors

This method was proposed by Bai and Ng (2008) to refine the DFM/PCA technique described above. The motivation for this refinement is the following: When factors are estimated by (20), the goal is to find linear combinations of X_t that maximize the sample variance of its elements. The predictive ability of x_{it}, $i = 1, 2, ..., N$ for π_{t+h} is not being taken into account. Thus, the set X_t can contain uninformative features (i.e., features that do not reduce the uncertainty about π_{t+h}), and calculating factors from these features can result in noisy factors with poor predictive ability. To overcome this issue, Bai and Ng (2008) proposed to, first, select only the features $\tilde{X}_t^h \subset X_t$ that have predictive power for π, and then compute the factors based on $\tilde{X}_t^h$ and not on all the features X_t. These factors are called *targeted* because the subset $\tilde{X}_t^h$ is different for each forecasting horizon h and sample period t.

The subset $\tilde{X}_t^h$ is selected, in the present work, using the method of *hard thresholding*[14]: employing the statistical t test to determine if the ith feature is *marginally* significant, without accounting for joint significance with other features, and discarding this feature if it is not significant at the 95% confidence level. As π_{t+h} depends not only on X_t, but also on w_t defined in (29), so w_t must be treated as control variables. The algorithm is the following:

1) For each $i = 1, 2, ..., N$, regress π_{t+h} on x_{it} and w_t. Denote t_i the t-statistic associated with the coefficient of x_{it}.

2) Make a ranking of the marginal predictive power of x_{it}, $i = 1, 2, ..., N$, by disposing their respective t-statistics in a decreasing order: $|t_1|, |t_2|, ..., |t_N|$.

3) Define the significance level $\alpha = 0,05$, select the features whose $|t_i| > t_\alpha$, where $t_\alpha = 1,96$, and let k_α^* be the number of these features.

4) Let $\tilde{X}_t^h(\alpha) = (x_{1t}, x_{2t}, ..., x_{k_\alpha^* t})'$ be the set of the features selected in (3). Estimate the factors F_t from $\tilde{X}_t^h(\alpha)$ using the method of Principal Components and obtain $\hat{F}_t$.

5) Run the regression (27) using $\underset{(r\times 1)}{\hat{F}_t}$ instead of $\underset{(r\times 1)}{F_t}$. Recall that $r = qp$. In this algorithm, $p = 4$ and q is defined using the Bayesian Information Criterion (BIC).

6) The nowcast is then obtained by (28).

Bai and Ng (2008) perform forecasting of inflation using the technique of Target Factors and show that it reduces substantially the forecasting errors relatively to the DFM/PCA model.

[14] Bai and Ng (2008) also discuss application of *soft thresholding* techniques, that perform subset selection and simultaneously shrink the values of the coefficients towards zero: LASSO, Elastic Net, and Least Angle Regressions.

The results of the daily nowcasting exercise carried out here show that Targeted Factors are slightly better than the DFM/PCA model (see Section 6).

5.2.3 Boosting Factors

Bai and Ng (2009) proposed this method as an alternative to the above discussed factor models to select variables and lags, with focus on selection of lags. The motivation for using boosting is that, if the pth lag of f or of π has high predictive power for π, the typical selection algorithms include all the previous lags, from 1 to $p-1$, in the model, even if these lags have no predictive power for π. The comprehensive boosting method treats each lag as a separate variable.

Following Bai and Ng (2009), define a function $\Phi : \mathbb{R}^N \to \mathbb{R}$, which takes the vector of observed variables $\underset{(N\times 1)}{X_t}$ and returns the predicted value for π. Let $C(\pi_t, \Phi(X_t))$ be the loss function that penalizes the deviation of $\Phi(X_t)$ from π_t. The objective is to estimate the function Φ that minimizes the expected loss $E[C(\pi_t, \Phi(X_t))]$. Under quadratic loss function, $C(\pi_t, \Phi(X_t)) = \frac{1}{2}(\pi_t - \Phi(X_t))^2$, the optimal solution is $\Phi(X_t) = E(\pi_t|X_t)$. Let us consider the quadratic loss function. Let $\underset{(5N\times 1)}{z_t}$ be the vector of all the N factors computed by PCA from X_t and four lags for each factor. The boosting factors algorithm used here to estimate $\Phi(z_t)$, for $t = 1, 2, ..., T$, is the same as in Medeiros et al. (2019):

1) For $t = 1, 2, ..., T$, initialize $\hat{\Phi}_{t,0} = \frac{1}{t}\sum_{i=1}^{t} \pi_i = \bar{\pi}$.

2) For $m = 1, ..., M$:

   a) Compute $\hat{u}_t = \pi_t - \hat{\Phi}_{t-h,m-1}$ for all t.

   b) For each candidate variable $i = 1, 2, ..., 5N$, regress $\hat{u}_t$ on z_{it} to obtain the coefficient $\hat{b}_i$, for all t. Compute $\hat{e}_{t,i} = \hat{u}_t - z_{i,t}\hat{b}_i$, for all t, take $\hat{e}_i = (\hat{e}_{1,i}, \hat{e}_{2,i}, ..., \hat{e}_{T,i})'$ and compute $SSR_i = \hat{e}_i'\hat{e}_i$.

   c) Select i_m^* the index of variable that delivers the least SSR and let $\hat{\phi}_{t,m} = z_{i_m^*,t}\hat{b}_{i_m^*}$.

   d) Update $\hat{\Phi}_{t,m} = \hat{\Phi}_{t,m-1} + \nu\hat{\phi}_{t,m}$, where ν is the step length set to $\nu = 0, 2$.

3) Stop the algorithm after the Mth iteration or when the BIC starts to increase. In the R code, M is set to $M = 10N$, where N is the number of columns in the database. This step is to avoid overfitting.

The resulting model is

$$\hat{\Phi}_{t,M} = \hat{\Phi}_{t,0} + \nu\sum_{m=1}^{M}\hat{\phi}_{t,m}. \tag{30}$$

As can be noted from the algorithm, Boosting is an ensemble algorithm (see Subsection 5.4) that constructs models *sequentially* (differently from Bagging, that constructs models *independently*), and each subsequent model corrects the errors of the previous one. The base learners $\hat{\phi}_{t,m}$ are trained in such a way that each weighting coefficient depends on the performance of previous learners.

5.3 Shrinkage techniques

When the model has learned too well the learning sample and is has low prediction capability out-of-sample, it is said that this model has been *overfitted*. One of the most frequent cases when overfitting of a linear model can occur is when the object-feature matrix $\underset{(T\times N)}{\mathbf{X}}$ has linearly dependent features (the database used in this work certainly has linearly dependent features: e.g., Brazilian Federal Government Debt and Brazilian Federal Government Domestic Debt). In this case, by definition, always exists a vector $v \neq 0$ such that $\langle v, X_t\rangle = 0$ for any object $\underset{(N\times 1)}{X_t}$. Suppose that, with this database $\mathbf{X}$, the linear model has been learned (e.g., using gradient descent) and the optimal vector of weights w_*, that minimizes the RMSE, has been obtained. But then models with any vector of weights $w_* + \alpha v$, $\alpha \neq 0$, will return, on all objects, the same answers as the optimal model, because $\langle w_* + \alpha v, X_t\rangle = \langle w_*, X_t\rangle + \alpha\underbrace{\langle v, X_t\rangle}_{=0} = \langle w_*, X_t\rangle$, so, these models will have the same RMSE as the optimal one, and will also be optimal. Hence, this optimization method can find optimal solutions with arbitrarily large weights. Usually, occurrence of large weights indicates that the model has been overfitted. One of the ways to strive against overfitting is penalizing large weights by adding to the quality functional a penalty function (a *regularizer*), so the objective function becomes:

$$Q_\lambda(w) = Q(w) + \lambda R(w), \tag{31}$$

where $Q(w)$ is the MSE quality functional and $R(w)$ is penalty function. This work considers two regularizers:

$$R(w) = \|w\|_1 = \sum_{i=1}^{N} |w_i| \qquad [L_1 \text{ regularizer}] \tag{32}$$

and

$$R(w) = \|w\|_2^2 = \sum_{i=1}^{N} w_i^2 \qquad [L_2 \text{ regularizer}]. \tag{33}$$

In (31), λ makes the balance between the accuracy of fitting of the learning sample and the penalty for excessive complexity of the model. λ is a *hyperparameter*, because it controls the process of learning and, differently from parameter w, cannot be fitted on the learning sample (because on the learning sample the optimal value for it is $\lambda = 0$). Thus, λ is adjusted out-of-sample, using BIC. Larger λ implies more simplicity of the model.

Though L_1 regularizer presents some difficulties by not having derivative at $w = 0$ (this difficulty can be overcome by running, for example, the sub-gradient descent algorithm, but it converges more slowly), nevertheless it has a good property: it is *sparsity*-inducing, because it sets some weights to zero, thus eliminating uninformative variables from the model. Sparse models can be desirable by different reasons: they do not contain irrelevant features that add noise; they demand less computational cost, etc. Their drawback is that parameters' consistency holds only under strong conditions. L_2 regularizer does not induce sparsity, that is, it usually

does not eliminate irrelevant variables from the model. To see why this occurs, let us consider two possible explanations.

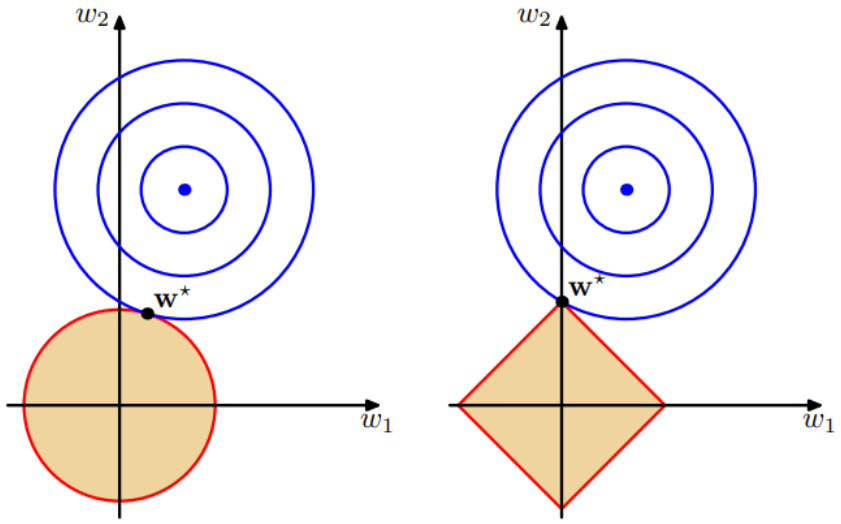


Figure 5: Level curves of the quality functional and constraint sets of L_1 and L_2 regularizers. Source: Bishop (2006).

Firstly, note that the unconstrained minimization problem $\min_w$(31) can be written as constrained problem

$$\begin{aligned} &\min_w Q(w) \\ &s.t.\ R(w) \leq C, \end{aligned} \tag{34}$$

where one-to-one relation between λ and C is assumed. Figure 5 depicts, in blue, the level curves of a convex quality functional, and, in red, the bounds of the constraint set (the left-side graph corresponds to L_2 regularizer, and the right-side one, to L_1 regularizer). The solution is given by the point on the bound of the constraint set that is the nearest to the unconstrained minimum. It can be thought that, in most cases, w_* of the L_1 constrained problem will be localized at one of the vertexes of the rhombus, thus, setting to zero one of the weights; while w_* of the L_2 constrained problem probably will not set any weight to zero.

To see this in a numeric example, let $w = (1, \epsilon)$ be some vector of weights, where ϵ is arbitrarily small. Let us see what is more advantageous from the point of view of minimization of L_1 and L_2: to diminish the first component of w or to diminish the second component? Diminishing the first component by δ, with $\delta < \epsilon$, gives

$$\|(1, \epsilon) - (\delta, 0)\|_1 = \|1 - \delta, \epsilon\|_1 = 1 - \delta + \epsilon \tag{35}$$

and

$$\|(1, \epsilon) - (\delta, 0)\|_2^2 = \|1 - \delta, \epsilon\|_2^2 = 1 - 2\delta + \delta^2 + \epsilon^2; \tag{36}$$

while diminishing the second component of w by the same δ gives

$$\|(1, \epsilon) - (0, \delta)\|_1 = \|1, \epsilon - \delta\|_1 = 1 + \epsilon - \delta \tag{37}$$

and

$$\|(1,\epsilon)-(0,\delta)\|_2^2 = \|1,\epsilon-\delta\|_2^2 = 1+\epsilon^2-2\epsilon\delta+\delta^2. \tag{38}$$

As can be seen from (35) and (37), for minimization of L_1 it does not matter what component to diminish. But as (36)<(38), the L_2 regularizer will prefer to diminish the biggest components of w and not the smallest, so that, choosing L_2 regularizer gives less chances to have small weights set to zero. Thus, the advantage of using L_1 regularizer is that it does variable selection.

This paper implements the following 5 Shrinkage Models, which choose different penalty functions.

5.3.1 LASSO

Proposed by Tibshirani (1996), this model uses L_1 regularizer and minimizes the following objective function:

$$\min_w \frac{1}{T}\left\|\mathbf{X}w-\pi\right\|_2^2 + \lambda\|w\|_1. \tag{39}$$

In the present paper this problem is solved for every forecasting period $h = 1, 2, ..., 20$:

$$\min_{w_h} \sum_{t=1}^{T-h}(\pi_{t+h} - w_h' X_t)^2 + \lambda\sum_{i=1}^{N}|w_{h,i}| \tag{40}$$

to obtain the forecast $\hat{\pi}_{t+h} = w_{h*}' X_t$.

5.3.2 Ridge Regression

Proposed by Hoerl and Kennard (1970), this model uses L_2 regularizer and minimizes the following objective function:

$$\min_w \frac{1}{T}\left\|\mathbf{X}w-\pi\right\|_2^2 + \lambda\|w\|_2^2. \tag{41}$$

The solution can be written explicitly as

$$w_* = (\mathbf{X}'\mathbf{X}+\lambda\mathbb{I})^{-1}\mathbf{X}'\pi. \tag{42}$$

To see that this solution is unique, note that a singular matrix $\mathbf{X}'\mathbf{X}$ suffers a small perturbation when to each diagonal element of it is being added the number λ, so each eigenvalue of $\mathbf{X}'\mathbf{X}$ is being lifted by λ and thus this matrix becomes nonsingular and hence invertible.

Solving (41) for every forecasting period $h = 1, 2, ..., 20$:

$$\min_{w_h} \sum_{t=1}^{T-h}(\pi_{t+h} - w_h' X_t)^2 + \lambda\sum_{i=1}^{N} w_{h,i}^2, \tag{43}$$

the forecast $\hat{\pi}_{t+h} = w_{h*}' X_t$ is obtained, where the optimal weights are given by (42).

5.3.3 Elastic Net

Elastic Net (Zou and Hastie (2005)) is a convex combination of L_1 and L_2 regularizers. The objective is

$$\min_{w} \frac{1}{T}\left\|\mathbf{X}w - \pi\right\|_2^2 + \alpha\lambda\|w\|_1 + (1-\alpha)\lambda\|w\|_2^2 \quad , \tag{44}$$

where $\alpha \in [0,1]$. In this work, the hyperparameter α is set *ad hoc* to be 0,5. The same value for this parameter is defined in other works, e.g. Chakraborty & Joseph (2017) and Medeiros et al. (2019).

Solving (44) for every forecasting period $h = 1, 2, ..., 20$:

$$\min_{w_h} \sum_{t=1}^{T-h}(\pi_{t+h} - w_h' X_t)^2 + \alpha\lambda\sum_{i=1}^{N}|w_{h,i}| + (1-\alpha)\lambda\sum_{i=1}^{N} w_{h,i}^2, \tag{45}$$

the forecast $\hat{\pi}_{t+h} = w_{h*}' X_t$ is obtained.

5.3.4 Adaptive LASSO

As pointed out above, LASSO estimator is consistent only under very strong conditions (Zou (2006)). Zou (2006) proposed the adaptive version of LASSO, where different weights are assigned to different coefficients. The objective is

$$\min_{w} \frac{1}{T}\left\|\mathbf{X}w - \pi\right\|_2^2 + \lambda\left\|\omega \odot w\right\|_1, \tag{46}$$

where $\omega_i = \dfrac{1}{|\hat{w}_i| + \dfrac{1}{\sqrt{T}}}$ for $i = 1, ..., N$, where $\hat{w}_i$ is the estimate from non-adaptive version of the model. Note that (46) is a convex optimization problem and has a global minimum. Zou (2006) show that (46) has oracle properties, that is, performs as well as if the underlying subset model was known. Solving (46) for every forecasting period $h = 1, 2, ..., 20$:

$$\min_{w_h} \sum_{t=1}^{T-h}(\pi_{t+h} - w_h' X_t)^2 + \lambda\sum_{i=1}^{N}\omega_i\left|w_{h,i}\right|, \tag{47}$$

the forecast $\hat{\pi}_{t+h} = w_{h*}' X_t$ is obtained.

5.3.5 Adaptive Elastic Net

This technique is analogous adaptive version of the Elastic Net model (44), where the coefficients w are weighted by previously estimated by OLS weights $\hat{w}$.

5.4 Ensemble methods

Ensemble methods make an average of the predictions of a group of models. The motivation for using these techniques is that often a combination of different models (*committee*) performs better than each model separately (Bishop (2006)).

5.4.1 Bagging

Bagging is an acronym for "**b**ootstrap **agg**regation" and was proposed by Breiman (1996). It constructs independently M models and averages their predictions. The algorithm used is the following (and illustrated in Figure 6):

1) From the original Learning Sample, generate M bootstrap samples. In this algorithm, $M = 100$ and the length of bootstrap samples is the same as of the original learning sample. Because this work deals with time series, the block bootstrapping is used, that is, block resampling with fixed block lengths of $l = 5$.

2) On every bootstrap sample $m = 1, 2, ..., M$, learn a linear model by OLS and make a variable selection by hard thresholding: eliminate the variables whose weights have t-statistic $|t| < c$ where c is 95% confidence level threshold. Then use only the selected variables to learn again a linear model by OLS.

3) Compute the ensemble model as the average of these M base models:

$$a(X)_{BAG} = \frac{1}{M} \sum_{m=1}^{M} b_m(X). \tag{48}$$

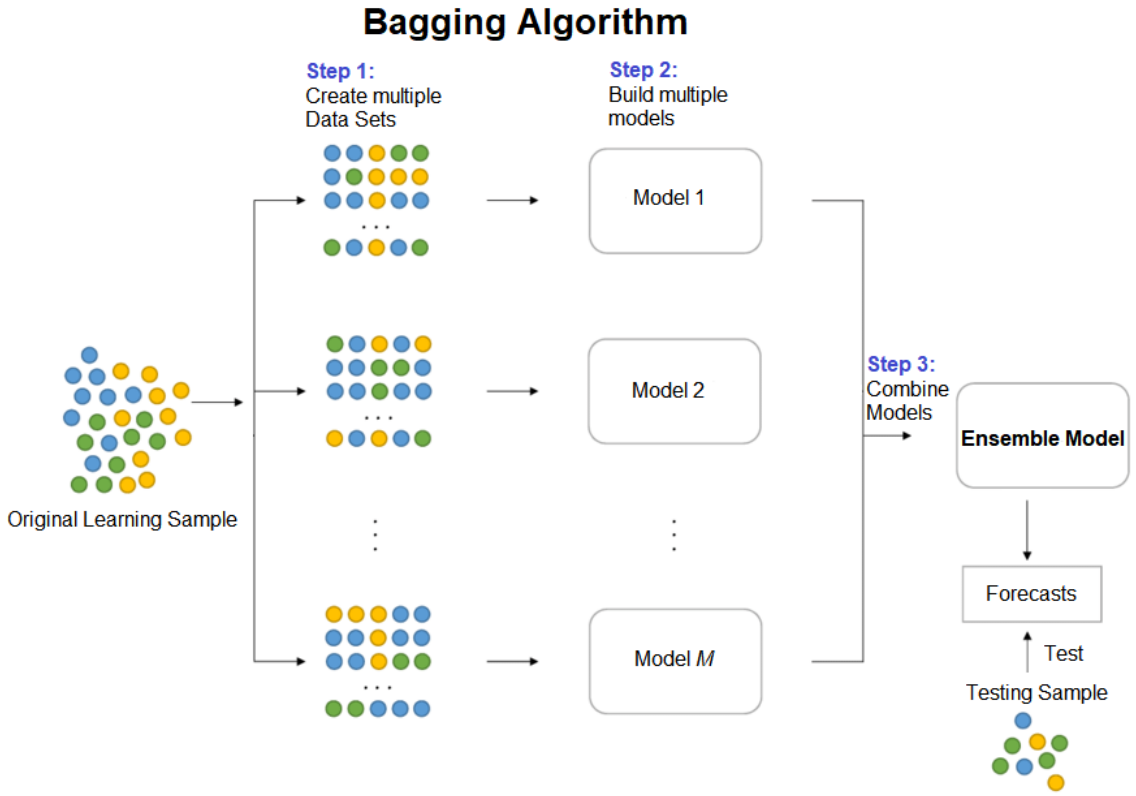


Figure 6: Bagging Algorithm.

An important issue is how bagging affects the bias-variance decomposition. It can be shown that the bias of any individual model is the same as the bias of the ensemble model, so,

bagging does not worsen the bias of the model. As for the variance, if the errors of the base models are uncorrelated, it can be shown that

$$E_{BAG} = \frac{1}{M} E_1, \tag{49}$$

where E_1 is the average of the expected squared errors of the base models working individually, and E_{BAG} is the expected squared error of the ensemble model (48). That is, the bagging technique can reduce in M times the expected squared error. However, the assumption of uncorrelatedness of the errors of base models is unlikely to hold in time-series framework. The more the base algorithms are correlated, the less is the reduction of the variance of a base algorithm. But it can be shown that $E_{BAG} \leq E_1$, that is, the expected squared error of bagging is never greater than the expected squared error of individual models. More sophisticated ensemble techniques, like Boosting and Random Forests, achieve more significant improvements (Bishop (2006)).

5.4.2 Random Forests

Random Forests technique (Breiman (2001)) consists of bagging of regression trees.

A regression tree is a recursive binary partition of the set of features. To exemplify how a regression tree is grown in a bi-dimensional features space (see Figure 7), let X_1 and X_2 be the features, and π is the target variable. As the sample is finite and every X_i takes values in an interval of $\mathbb{R}$, the idea is to partition these intervals in some regions where each region R_k gives the same prediction c_k for π. That is, given the target variable π_{t+h}, the set of N features X_t and a number of terminal nodes K, the splitting aims to minimize

$$\left\| \pi_{t+h} - \sum_{k=1}^{K} c_k I_k \{ X_t \in R_k(\theta_k) \} \right\|_2^2 \quad , \tag{50}$$

where the constant c_k is estimated as a sample average of realizations of π that fall into the region R_k; $I\{\cdot\}$ is indicator function which is equal to 1 if the condition in brackets is satisfied and is equal to zero if not; and θ_k is the vector of parameters that define the region R_k, $k = 1, ..., K$. In Figure 7, for example, the feature space was partitioned in five regions R_k, $k = 1, ..., 5$.

Regression trees are rather complex and can achieve zero error on learning sample (thus, are low-biased), but at the same time they are very subject to overfitting. The Random Forests technique makes bagging of regression trees. It was argued that bagging allows to join low-biased but highly sensitive to learning sample algorithms into a low-biased committee with low variance. Regression trees, thus, are a good family of base algorithms that bagging can be applied to. As pointed out above (see (49)), bagging can substantially reduce the variance of base algorithms, provided that they are weakly correlated. In Random Forests, the correlation between trees is reduced by two mechanisms. The first one is that, in each node, the feature that is being split is selected from a random subset of $n = \left\lfloor \frac{N}{3} \right\rfloor$ features and not from the whole set of N features. The tree is grown until perfect quality of learning is reached (in this regression framework, until each leaf has 5 objects). The second mechanism is that each tree

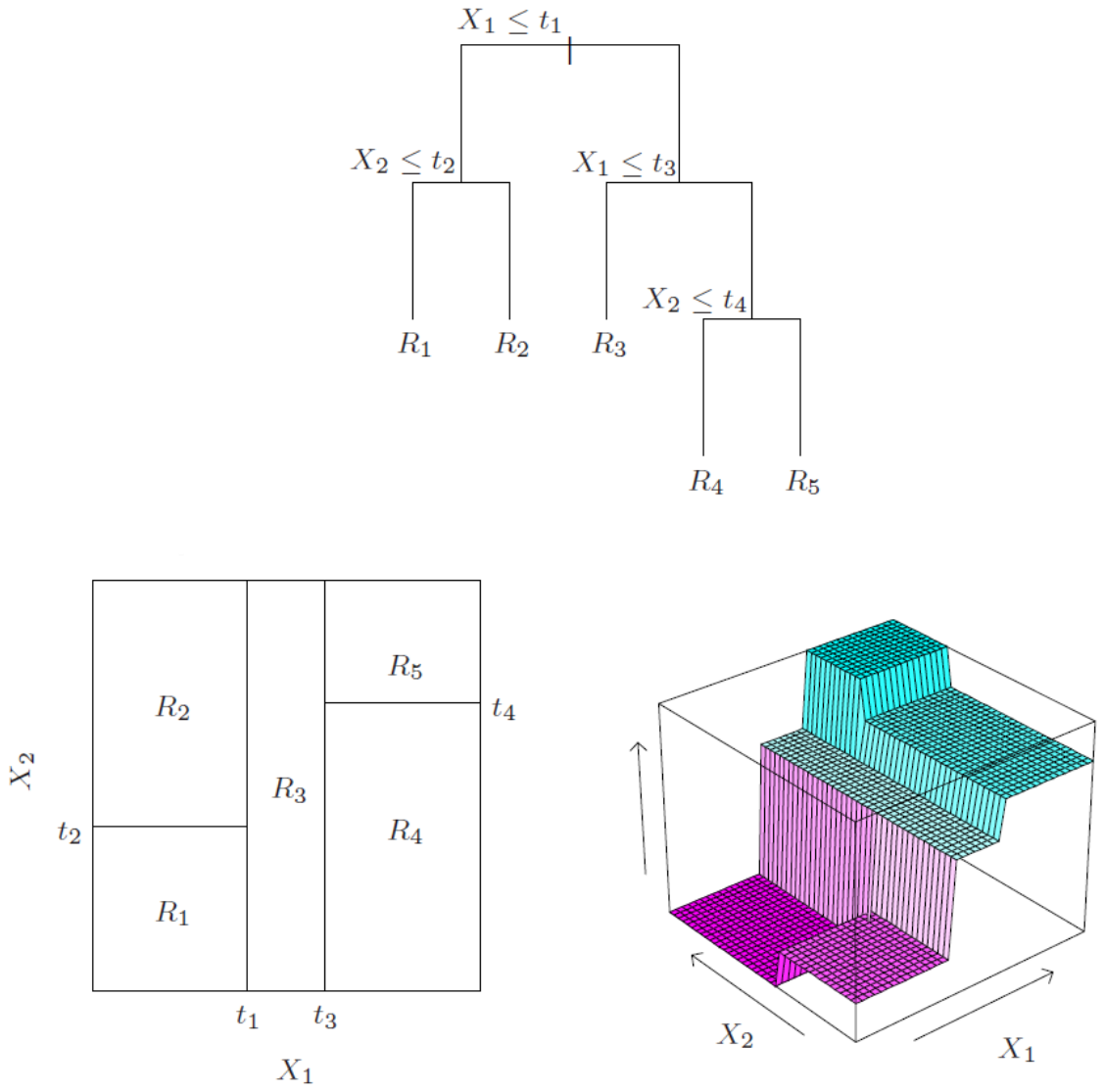


Figure 7: Decision tree. Source: Hastie et al. (2001).

is learned on a bootstrapped subsample. So, the Random Forests algorithm employed in this work is the following:

1) From the original Learning Sample, generate M bootstrap samples. In this algorithm, $M = 500$ and the length of bootstrap samples is the same as of the original learning sample. Because this work deals with time series, the block bootstrapping is used, that is, block resampling with fixed block lengths of $l = 5$.

2) On every bootstrap sample $m = 1, 2, ..., M$, learn a regression tree with K_m regions.

3) The final model is the average of the forecasts of each tree applied to the original data:

$$\hat{\pi}_{t+h} = \frac{1}{M} \sum_{m=1}^{M} \left[\sum_{k=1}^{K_m} \hat{c}_{k,m} I_{k,m} \left\{ X_t, \hat{\theta}_{k,m} \right\} \right]. \tag{51}$$

5.4.3 Complete Subset Regressions

Proposed by Elliott et al. (2013, 2015), this technique is motivated by the fact that, having N features in the dataset, it would be computationally infeasible to run regressions with all possible combinations of features. Thus, from $K < N$ candidate variables only a subset $k < K$ of the features is used and $\frac{K!}{(K-k)!k!}$ regressions with all possible combinations of these features are run. This set of models for a fixed value of k is called a *complete subset*. Then, the average of predictions of these models is taken. So, the algorithm implemented here is the following:

1) Pre-selection step: For each $i = 1, 2, ..., N$, regress π_{t+h} on x_{it} (here, the control variables w_t are not being considered). Denote t_i the t-statistic associated with the coefficient of x_{it}. Make a ranking of the marginal predictive power of x_{it}, $i = 1, 2, ..., N$, by disposing their respective t-statistics in a decreasing order: $|t_1|, |t_2|, ..., |t_N|$. Select $K = 20$ variables with the greatest predictive power for π_{t+h}.

2) Let $k = 4$ and run $\frac{K!}{(K-k)!k!} = 4845$ complete subset regressions of π_{t+h} on these features.

3) Let $b_i(X)$ be the forecast of the ith subset regression, $i = 1, 2, ..., 4845$. The final forecast of the Complete Subset Regressions is given by averaging the individual forecasts with equal weights:

$$\hat{\pi}_{t+h} = \frac{1}{4845} \sum_{i=1}^{4845} b_i(X). \tag{52}$$

5.4.4 Jackknife Model Averaging

Jackknife Model Averaging, proposed by Hansen and Racine (2012), also performs model averaging, but, instead of using simple average, it employs weighted average, where the weights are selected by minimizing a leave-one-out cross-validation criterion.

Let $\{\hat{\mu}^1, \hat{\mu}^2, ..., \hat{\mu}^M\}$ be M linear models that are candidates to predict π_{t+h}. Considering the Least Squares estimation, the mth model ($m = 1, ..., M$) is given by

$$\begin{aligned} \hat{\mu}_t^m &= X_t^{m\prime} \hat{w}_*^m \\ &= X_t^{m\prime} \left(\mathbf{X}^{m\prime} \mathbf{X}^m\right)^{-1} \mathbf{X}^{m\prime} \pi. \end{aligned} \tag{53}$$

The mth Jackknife model is then given by

$$\begin{aligned} \tilde{\mu}_t^m &= X_t^{m\prime} \tilde{w}_*^m \\ &= X_t^{m\prime} \left(\mathbf{X}_{(-t)}^{m\prime} \mathbf{X}_{(-t)}^m\right)^{-1} \mathbf{X}_{(-t)}^{m\prime} \pi_{(-(t+h))} \end{aligned} \tag{54}$$

where $\tilde{w}_*^m$ is the estimator $\hat{w}_*^m$ computed when the observations (π_{t+h}, X_t) are deleted. The mth Jackknife model has the vector of residuals given by $\tilde{e}^m = \pi - \tilde{\mu}^m$.

The Jackknife Models Averaging is

$$\tilde{\mu}(\omega) = \sum_{m=1}^{M} \omega^m \tilde{\mu}^m, \tag{55}$$

subject to

$$\omega \in \mathcal{H} := \left\{ \omega \in \mathbb{R}^M : \omega^m \geq 0, \quad \sum_{m=1}^{M} \omega^m = 1 \right\}. \tag{56}$$

The Jackknife Models Averaging vector of residuals is then given by

$$\begin{aligned} \tilde{e}(\omega) &= \pi - \tilde{\mu}(\omega) \\ &= \sum_{m=1}^{M} \omega^m \tilde{e}^m \\ &= \tilde{e}'\omega. \end{aligned} \tag{57}$$

To estimate the weights ω, the following program is computed:

$$\begin{aligned} \min_{\omega} \quad & \frac{1}{T}\tilde{e}(\omega)'\tilde{e}(\omega) = \omega' \mathbf{S}_T \omega \\ s.t. \quad & \omega \in \mathcal{H}, \end{aligned} \tag{58}$$

where $\underset{(M\times M)}{\mathbf{S}_T} = \frac{1}{T}\tilde{e}'\tilde{e}$. The expression minimized in (58) is known as *cross-validation criterion*.

Thus, the JMA forecast is given by $\hat{\pi}_{t+h} = \hat{\mu}'\omega_*$, where ω_* is the solution to (58). Hansen and Racine (2012) show that ω_* is asymptotically efficient, that is, achieves the lowest possible expected squared error.

5.5 Hybrid models

The following two models were designed in Medeiros et al. (2019) to understand if good forecasting performance of the RF model is due to nonlinearities in inflation process that are being captured or to the method of variable selection that RF employs.

5.5.1 RF/OLS

The algorithm is the following:

1) Do the Step 1 of the Random Forest algorithm.

2) On every bootstrap sample $m = 1, 2, ..., M$:

   a) Learn a regression tree with 20 nodes and save $n \leq 20$ split variables.

   b) Run the OLS on the selected splitting variables.

   c) Compute the forecast $\hat{\pi}^m_{t+h}$.

3) The final forecast of the model is given by

$$\hat{\pi}_{t+h} = \frac{1}{M}\sum_{m=1}^{M}\hat{\pi}_{t+h}^{m}. \tag{59}$$

The motivation to implement this model is the following. If it performs as well as RF, this indicated that nonlinearities in inflation are not important to explain good performance of RF. If it performs not so well as the RF, this indicates that nonlinearities are important. If it performs better than bagging – which, like RF, is an ensemble method, but is linear while the RF is nonlinear –, this indicates that variable selection made by RF is important to explain its performance.

5.5.2 adaLASSO/RF

This model firstly performs variable selection using the adaLASSO method, and then employs these selected variables to implement the RF model. The motivation is that if adaLASSO/RF performs as well as the RF, this indicates that variable selection made by RF is not likely to explain performance of RF.

6 Results and discussion

6.1 Comparison of the performance of ML methods

To compare the performance of above described ML models in the task of daily nowcasting, the quality functionals (1) and (2) were defined, and calculated for each model's results. To evidence the benefits[15] of using high-dimensional ML models instead of the univariate benchmarks, Table 2 reports each model's nowcasting error relatively to the nowcasting error of the Random Walk model: $\frac{\text{Model's error}}{\text{RW error}}$.

As evidenced by the results in Table 2, all the Machine Learning methods (except Bagging) perform better than the univariate benchmarks. The benefit of using *big data* instead of univariate models is of approximately 4% for Shrinkage methods, 10% for Factor Models, 12% for Complete Subset Regression and comes to be of 20% for Random Forest. The best performing models are Random Forest, Complete Subset Regression and Target Factors.

Within Shrinkage methods, LASSO, Ridge and ElNet perform better than adaptive versions of LASSO and ElNet. Another finding is that LASSO has the same nowcasting error as Ridge Regression in this exercise, so L_1 regularizer delivers the same error as the L_2 regularizer.

Factor models perform much better that Shrinkage methods, and targeting the factors reduces slightly the nowcasting error, as well as boosting.

On the other hand, Bagging has the worst performance, deteriorating the quality of prediction (in terms of RMSE) in more than 50% compared with RW benchmark. Jackknife Model Averaging has the third worst performance, but it is still slightly better than RW.

[15] To be precise, "benefit" means "reduction of nowcasting error relatively to benchmark".

Table 2: Nowcasting performance of Machine Learning models in terms of RMSE and MAE of Random Walk benchmark.

Model	**RMSE (MAE) ratio**
RW	1,00 (1,00)
AR	1,039 (1,061)
LASSO	0,954 (0,952)
adaptive LASSO	0,963 (0,961)
Elastic Net	0,957 (0,955)
adaptive Elastic Net	0,965 (0,963)
Ridge Regression	0,954 (0,953)
Bagging	1,592 (1,352)
Complete Subset Regression	**0,881 (0,874)**
Jackknife Model Averaging	0,973 (0,982)
DFM with PCA	0,908 (0,903)
Target Factors	**0,889 (0,899)**
Boosting Factors	0,904 (0,900)
Random Forest	**0,808 (0,789)**
Random Forest / OLS	0,945 (0,936)
Adaptive LASSO / Random Forest	0,934 (0,937)

Random Forest presents the best performance. As in Medeiros et al. (2019), the question is: is this performance due to nonlinearities in inflation process that are being successfully captured by the model or due to variable selection that RF makes? As Table 2 shows, the hybrid model RF/OLS performs worser than RF; this indicates that nonlinearities are important to nowcast inflation. On the other hand, Random Forest/OLS performs rather better than Bagging, which is a linear ensemble method; this indicates that the method of variable selection that RF employs is more efficient than other methods of dimensionality reduction considered in this paper. Another result which indicates it is that the hybrid model adaLASSO/RF performs rather worser than RF. Hence, the variable selection made by RF matters for the quality of inflation nowcasting.

As for nonlinearities, Medeiros et al. (2019) show that in periods of high volatility of inflation RF performs better than in periods of low volatility. Garcia et al. (2017) argue that Brazilian inflation exhibits high short term-volatility. One of the reasons for this is uncertainty, which accounts for nonlinearities in the economy. This can possibly explain good performance of the RF, which is a highly nonlinear model.

The results presented in Table 2 are compatible with the empirical evidence found in the literature about inflation forecasting. Especially, in Medeiros et al. (2019), RF was found to yield the best performance; in Garcia et al. (2017), CSR dominated another models. Here, both of them are the two best models for inflation nowcasting. As in Medeiros et al. (2019), Shrinkage methods here perform very similarly, JMA performs poorly, and boosting or targeting factors improves only slightly the quality of DFM/PCA model. On the other hand, there are several findings in this work that are different from those encountered in the literature. Firstly, the maximum of nowcasting error reduction that ML models can do is 20% in this work; in Medeiros et al. (2019), RF reduced in 30% the forecasting error relatively to the RW benchmark.

Secondly, Medeiros et al. (2019) and Garcia et al. (2017) report poor performance of Factor Models and show that Shrinkage methods perform better; in the present work, all the Factor Models perform better than Shrinkage, especially, Target Factor model reduces the nowcasting RMSE in 11%, while Shrinkage techniques reduce it only in about 4%. Thus, the present exercise suggests that, for the purposes of nowcasting of Brazilian inflation, it can be better to use all the available information than to discard features that are less informative.

6.2 Benefits of daily nowcasting *vis-à-vis* monthly forecasting

Compared to the existing literature, a contribution that this work aims to do is to perform a *daily* real-time nowcasting of inflation, instead of forecasting it monthly as in Garcia et al. (2017), Medeiros et al. (2019) and other works that employ ML methods to forecast inflation. To measure quantitatively the benefits of daily nowcasting *vis-à-vis* monthly forecasting, the following monthly forecasting exercise was performed:

From the original sample described in Section 2, with 174 features and 2974 observations, only monthly frequency features were considered in their original (monthly) frequency, without interpolation. Thus, the sample employed in monthly forecasting exercise has 142 observations (from 30/11/2006 till 31/08/2018)[16] and 137 features. Its learning sample has 95 observations and the testing sample has 47 observations (from 31/10/2014 till 31/08/2018).

The only forecasting horizon that is of interest in this work is one-month-ahead, because the question is: "What are the benefits of working on nowcasting of inflation every day, monitoring it and issuing the prediction daily, considering informational news that appear in the dataset every day with updating of the features, *vis-à-vis* calculating the predicted inflation for the current month only once a month, using monthly frequency data?" Thus, only $\tilde{h} = 1$ is implemented for all the models in monthly-frequency exercise, where $\tilde{h}$ is monthly-frequency time horizon.

The computer codes used in daily exercise were modified to not proliferate the number of features. Recall, from Section 2, that daily nowcasting considered also four lags for each feature and four principal components, augmenting the number of features from 174 to 712. But while the number of observations in daily framework is big enough to permit this, in monthly framework with only 142 observations this proliferation of features precludes the implementation of the most part of algorithms. Thus, in monthly framework, only the original 137 monthly features were used, without augmenting the dataset by lags or factors.

The Appendix contains graphs that show the inflation predicted by daily nowcasting and by one-month-ahead forecasting for each of the models implemented in this paper. In the graphs of daily nowcasting, the predicted inflation out-of-sample is depicted in red. As can be observed graphically, the nowcasting is more accurate than one-month-ahead forecasting.

The quality functionals defined to assess the algorithms in monthly exercise are $\text{RMSE}\left(\hat{y}_t, y_t\right) = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(\hat{y}_t - y_t\right)^2}$ and $\text{MAE}\left(\hat{y}_t, y_t\right) = \frac{1}{T}\sum_{t=1}^{T}\left|\hat{y}_t - y_t\right|$, where $T = 47$, as stated above. Note that the errors in monthly exercise are calculated *online*, which is made possible by the

[16] The sample stops in August 2018 because that month's inflation was only known in September, and, as the daily sample stops in 03/10/2018, September's inflation was not already known on 03/10/2018.

fact that the true inflation rate is observed monthly, that is, in each row of the dataset. This is not the case in daily nowcasting framework, where the true inflation is observed only once a month, so the errors have to be calculated using the formulas (1) and (2). The results are presented in Table 3, where the values of errors are shown, and not their ratio to the RW errors as in Table 2. The benefit of daily nowcasting for each model m is calculated as $\left(1 - \frac{\text{Error of daily nowcasting for model } m}{\text{Error of monthly forecasting for model } m}\right) \times 100$.

Table 3: RMSE and MAE of Machine Learning models in daily nowcasting and monthly forecasting of inflation.

Model	**RMSE (MAE) of *daily nowcasting***	**RMSE (MAE) of *monthly forecasting***	**Benefit of daily nowcasting**
RW	0,152 (0,114)	0,313 (0,238)	51% (52%)
AR	0,158 (0,121)	0,306 (0,239)	48% (49%)
LASSO	0,145 (0,109)	0,305 (0,236)	52% (54%)
adaptive LASSO	0,146 (0,110)	0,295 (0,240)	50% (54%)
Elastic Net	0,145 (0,109)	0,319 (0,249)	54% (56%)
adaptive Elastic Net	0,147 (0,110)	0,296 (0,239)	50% (54%)
Ridge Regression	0,145 (0,109)	0,313 (0,233)	54% (53%)
Bagging	0,242 (0,154)	0,307 (0,236)	21% (35%)
Complete Subset Regression	0,134 (0,1)	0,320 (0,254)	**58% (61%)**
Jackknife Model Averaging	0,148 (0,112)	0,310 (0,241)	52% (54%)
DFM with PCA	0,138 (0,103)	0,284 (0,221)	51% (53%)
Target Factors	0,135 (0,102)	0,290 (0,224)	53% (54%)
Boosting Factors	0,137 (0,103)	0,298 (0,224)	54% (54%)
Random Forest	0,123 (0,090)	0,290 (0,219)	**58% (59%)**
Random Forest/OLS	0,144 (0,107)	0,281 (0,209)	49% (49%)
Adaptive LASSO/Random Forest	0,142 (0,107)	0,311 (0,240)	54% (55%)

The findings are encouraging: it is verified that the gains of nowcasting the inflation daily may reach 60% and are higher than 48% for all models except Bagging. Note, from the first line of Table 3, that just filtering the target variable daily via Kalman Filter is 51% better than forecasting inflation once a month. The most substantial reduction in RMSE is presented by CSR and RF. Among Shrinkage methods, it is worth noting that Ridge Regression reduces the error more than LASSO; furthermore, adaptive versions reduce the error less than their original models. Baggging does not have the worst performance among monthly models, but it performs worse than benchmark in daily nowcasting, so its benefit is the lowest, 24%. Factor Models are all beneficial for daily nowcasting.

These benefits of nowcasting the inflation daily occur because, during the month, the information set becomes larger every day due to new data that is being released on many macroeconomic series. Updating the forecast of target variable with the use of this new information reduces the uncertainty about the target during the month, i.e., reduces forecasting error. On the other hand, when the prediction is made once a month, it is as if every day the same prediction was issued by the economist without updating it during the month, without

using new information. The empirical evidence on benefits of nowcasting with high-frequency data *vis-à-vis* forecasting with low-frequency data is widely provided by the literature on economic nowcasting, e.g. Giannone et al. (2008), Banbura et al. (2013), Modugno (2013) and others.

To obtain the results illustrated by Tables 2 and 3, formulas (1) and (2) were used to calculate each model's errors in daily nowcasting exercise. Note that, from (1) and (2), it can be seen that each model m's nowcasting error that appears in the first column of Table 3 is the average of the 20 nowcasting errors corresponding to each nowcasting horizon $h = 1, 2, ..., 20$ (in days). As explained in Section 3, for each model m there are 20 models being computed; then, their predictions are being averaged over the 47 months of the testing sample and over the 20 models. For example, working with CSR (or any other) technique, if the true IPCA rate relative to August was released by IBGE on 08/09/2015, so on 09/09/2015 the prediction of the $CSR_{h=20}$ model, which is made for 20 days ahead, is calculated and compared with the true inflation released on 08/10/2015. In the same manner, on 10/09/2015, the prediction of the $CSR_{h=19}$ model, which is made for 19 days ahead, is calculated and compared with the true inflation released in 08/10/2015. And on 07/10/2015, the prediction of the $CSR_{h=1}$ model, which is made for one day ahead, is calculated and compared with the true inflation released in 08/10/2015. Table 4 illustrates the nowcasting error for each technique and for each nowcasting horizon. For example, for each month of the Testing Sample, the model $CSR_{h=20}$ yields one nowcasting error. The result in Table 4 for the model $CSR_{h=20}$ is the average of these errors over the 47 months of the Testing Sample. The same holds for any technique and any horizon h. Thus, the formulas to calculate RMSE and MAE of each cell of Table 4 are $\text{RMSE}\left(\hat{y}_t, y_t\right) = \sqrt{\frac{1}{T}\sum_{t=1}^{T}\left(\hat{y}_t - y_t\right)^2}$ and $\text{MAE}\left(\hat{y}_t, y_t\right) = \frac{1}{T}\sum_{t=1}^{T}\left|\hat{y}_t - y_t\right|$, where $T = 47$. Note that Table 4 is the detailed version of Table 3, that is, Table 4 shows explicitly which errors are being averaged by (1) and (2) over horizons to yield the results of Table 3. Moreover, in each cell of Table 4, the nowcasting error is compared with the corresponding forecasting error of the second column of Table 3 to quantify the benefit of daily nowcasting *vis-à-vis* monthly forecasting. For example, for CSR technique, the errors of $CSR_{h=20}, CSR_{h=19}, ..., CSR_{h=1}$ models were all compared with $RMSE = 0,320$ and $MAE = 0,254$, to calculate percentage benefit of daily nowcasting given by $\left(1 - \frac{\text{RMSE of } CSR_h}{0,320}\right) \times 100$ for RMSE and $\left(1 - \frac{\text{MAE of } CSR_h}{0,254}\right) \times 100$ for MAE, where $h = 1, 2, ..., 20$. The same holds for all other techniques: LASSO, RF, etc.

The detailed results in Table 4 show that, generally, when h decreases from 20 to one, that is, while the date of new release of true IPCA is approaching, the nowcasting error decreases, and the benefit of daily nowcasting increases. Figures 8-9 were plotted to illustrate this. In fact, during the month, new information is entering the information set, and this helps the economist to issue a more accurate prediction. It can be seen that the benefits of daily nowcasting can be as high as 70% for some horizons. On average, they are in order of 50%-60%, as evidenced by Table 3. One possible reason of why the gains are so high is that monthly forecasting does not take into account the daily-frequency features, such as the inflation Monitors. As discussed in the next Subsection, the most informative features are the daily ones, thus not considering them can lead to bigger prediction errors.

Table 4: Daily nowcasting errors for each technique and for each horizon h.

Random Walk:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,251	0,231	0,213	0,199	0,183	0,163	0,149	0,141	0,127	0,112	0,102	0,096	0,086	0,088	0,101	0,121	0,129	0,136	0,132	0,131
(MAE)	(0,188)	(0,174)	(0,160)	(0,152)	(0,144)	(0,130)	(0,118)	(0,109)	(0,099)	(0,088)	(0,083)	(0,079)	(0,071)	(0,074)	(0,084)	(0,100)	(0,105)	(0,109)	(0,104)	(0,106)
Benefit	20%	26%	32%	36%	42%	48%	52%	55%	59%	64%	67%	69%	72%	72%	68%	61%	59%	59%	58%	58%
	(21%)	(27%)	(33%)	(36%)	(40%)	(45%)	(50%)	(54%)	(58%)	(63%)	(65%)	(67%)	(70%)	(69%)	(65%)	(58%)	(56%)	(56%)	(56%)	(55%)
AR:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,184	0,169	0,154	0,147	0,132	0,131	0,132	0,137	0,127	0,125	0,124	0,135	0,131	0,150	0,181	0,173	0,186	0,185	0,185	0,185
(MAE)	(0,140)	(0,127)	(0,127)	(0,124)	(0,111)	(0,112)	(0,101)	(0,099)	(0,099)	(0,101)	(0,105)	(0,105)	(0,124)	(0,149)	(0,135)	(0,139)	(0,136)	(0,132)	(0,123)	(0,127)
Benefit	40%	45%	49%	52%	57%	57%	57%	55%	59%	59%	56%	57%	51%	41%	43%	39%	40%	40%	43%	40%
	(41%)	(47%)	(47%)	(48%)	(54%)	(53%)	(58%)	(58%)	(58%)	(58%)	(56%)	(56%)	(48%)	(38%)	(44%)	(42%)	(43%)	(45%)	(48%)	(47%)
LASSO:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,145	0,156	0,196	0,146	0,144	0,141	0,136	0,145	0,147	0,169	0,170	0,164	0,122	0,117	0,113	0,126	0,131	0,144	0,134	0,135
(MAE)	(0,121)	(0,124)	(0,136)	(0,125)	(0,124)	(0,122)	(0,124)	(0,108)	(0,099)	(0,109)	(0,106)	(0,108)	(0,093)	(0,087)	(0,084)	(0,096)	(0,102)	(0,107)	(0,106)	(0,108)
Benefit	52%	49%	36%	52%	53%	54%	55%	53%	52%	45%	44%	46%	60%	62%	63%	59%	57%	53%	56%	56%
	(49%)	(48%)	(42%)	(47%)	(48%)	(48%)	(48%)	(54%)	(58%)	(54%)	(55%)	(54%)	(61%)	(63%)	(65%)	(59%)	(57%)	(55%)	(55%)	(54%)
adaLASSO:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,144	0,133	0,147	0,188	0,166	0,146	0,143	0,119	0,119	0,127	0,157	0,173	0,162	0,122	0,157	0,152	0,164	0,155	0,129	0,097
(MAE)	(0,140)	(0,127)	(0,136)	(0,166)	(0,157)	(0,142)	(0,101)	(0,088)	(0,078)	(0,081)	(0,133)	(0,141)	(0,116)	(0,076)	(0,092)	(0,095)	(0,111)	(0,091)	(0,081)	(0,049)
Benefit	51%	55%	50%	36%	44%	50%	52%	60%	60%	57%	47%	41%	45%	59%	47%	48%	44%	48%	58%	67%
	(42%)	(47%)	(44%)	(31%)	(35%)	(41%)	(58%)	(64%)	(68%)	(67%)	(47%)	(42%)	(52%)	(69%)	(67%)	(61%)	(54%)	(63%)	(67%)	(80%)
ElNet:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,198	0,193	0,169	0,168	0,160	0,157	0,152	0,150	0,150	0,149	0,145	0,142	0,140	0,137	0,129	0,119	0,110	0,109	0,109	0,100
(MAE)	(0,179)	(0,164)	(0,159)	(0,158)	(0,140)	(0,137)	(0,127)	(0,122)	(0,105)	(0,099)	(0,093)	(0,092)	(0,086)	(0,086)	(0,085)	(0,084)	(0,080)	(0,080)	(0,075)	(0,051)
Benefit	38%	40%	47%	47%	50%	51%	52%	53%	53%	53%	55%	56%	56%	57%	59%	63%	66%	66%	66%	69%
	(28%)	(34%)	(36%)	(37%)	(44%)	(45%)	(49%)	(51%)	(58%)	(60%)	(63%)	(63%)	(65%)	(65%)	(66%)	(66%)	(68%)	(68%)	(70%)	(80%)
adaElNet:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,197	0,192	0,178	0,157	0,164	0,155	0,155	0,154	0,154	0,152	0,151	0,122	0,140	0,139	0,134	0,129	0,124	0,110	0,097	0,085
(MAE)	(0,141)	(0,101)	(0,156)	(0,163)	(0,154)	(0,153)	(0,134)	(0,128)	(0,125)	(0,109)	(0,103)	(0,09)	(0,097)	(0,093)	(0,09)	(0,089)	(0,079)	(0,074)	(0,074)	(0,047)
Benefit	33%	35%	40%	47%	45%	48%	48%	48%	48%	49%	49%	59%	53%	53%	55%	56%	58%	63%	67%	77%
	(41%)	(58%)	(35%)	(32%)	(35%)	(36%)	(44%)	(46%)	(48%)	(54%)	(57%)	(62%)	(60%)	(61%)	(63%)	(63%)	(67%)	(69%)	(69%)	(80%)
Ridge Regression:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,192	0,180	0,170	0,169	0,166	0,156	0,148	0,148	0,146	0,139	0,133	0,132	0,130	0,127	0,126	0,125	0,125	0,125	0,123	0,115
(MAE)	(0,186)	(0,179)	(0,176)	(0,173)	(0,139)	(0,126)	(0,119)	(0,117)	(0,114)	(0,104)	(0,097)	(0,092)	(0,086)	(0,074)	(0,074)	(0,070)	(0,070)	(0,068)	(0,060)	(0,057)
Benefit	39%	43%	46%	46%	47%	50%	53%	53%	53%	56%	58%	58%	59%	59%	60%	60%	60%	60%	61%	63%
	(20%)	(23%)	(24%)	(26%)	(40%)	(46%)	(49%)	(50%)	(51%)	(55%)	(58%)	(61%)	(63%)	(68%)	(68%)	(70%)	(70%)	(71%)	(74%)	(75%)
Bagging:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,340	0,332	0,329	0,332	0,303	0,296	0,289	0,277	0,242	0,220	0,220	0,216	0,211	0,191	0,169	0,163	0,148	0,131	0,116	0,070
(MAE)	(0,200)	(0,196)	(0,193)	(0,191)	(0,190)	(0,188)	(0,183)	(0,166)	(0,164)	(0,161)	(0,158)	(0,156)	(0,131)	(0,125)	(0,122)	(0,121)	(0,120)	(0,111)	(0,107)	(0,092)
Benefit	-11%	-8%	-7%	-5%	1%	3%	6%	10%	21%	28%	28%	30%	31%	38%	45%	47%	52%	57%	62%	77%
	(15%)	(17%)	(18%)	(19%)	(19%)	(20%)	(22%)	(30%)	(30%)	(32%)	(33%)	(34%)	(44%)	(47%)	(48%)	(49%)	(49%)	(53%)	(55%)	(61%)
CSR:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE	0,192	0,191	0,175	0,170	0,154	0,152	0,152	0,146	0,140	0,136	0,121	0,113	0,112	0,100	0,095	0,094	0,093	0,091	0,091	0,062
(MAE)	(0,190)	(0,168)	(0,165)	(0,152)	(0,151)	(0,114)	(0,113)	(0,110)	(0,104)	(0,102)	(0,088)	(0,086)	(0,083)	(0,080)	(0,073)	(0,069)	(0,064)	(0,057)	(0,056)	(0,055)

Benefit	40% (25%)	40% (34%)	45% (35%)	47% (40%)	52% (40%)	52% (55%)	53% (56%)	54% (57%)	56% (59%)	58% (60%)	62% (65%)	65% (66%)	65% (67%)	69% (68%)	70% (71%)	71% (73%)	71% (75%)	72% (77%)	72% (78%)	81% (78%)
JMA:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE (MAE)	0,198 (0,147)	0,192 (0,143)	0,186 (0,133)	0,183 (0,130)	0,179 (0,129)	0,165 (0,125)	0,159 (0,122)	0,153 (0,120)	0,148 (0,119)	0,148 (0,117)	0,145 (0,116)	0,143 (0,111)	0,140 (0,109)	0,136 (0,104)	0,112 (0,092)	0,109 (0,091)	0,107 (0,088)	0,106 (0,087)	0,098 (0,085)	0,091 (0,081)
Benefit	36% (39%)	38% (41%)	40% (45%)	41% (46%)	42% (46%)	47% (48%)	49% (49%)	51% (50%)	52% (51%)	52% (51%)	53% (52%)	54% (54%)	55% (55%)	56% (57%)	64% (62%)	65% (62%)	65% (63%)	66% (64%)	68% (65%)	71% (66%)
DFM/PCA:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE (MAE)	0,164 (0,110)	0,164 (0,113)	0,170 (0,117)	0,162 (0,124)	0,159 (0,132)	0,152 (0,132)	0,150 (0,104)	0,157 (0,109)	0,145 (0,117)	0,143 (0,127)	0,133 (0,102)	0,131 (0,100)	0,121 (0,093)	0,116 (0,087)	0,103 (0,081)	0,121 (0,081)	0,119 (0,093)	0,109 (0,082)	0,106 (0,080)	0,099 (0,074)
Benefit	42% (50%)	42% (49%)	40% (47%)	43% (44%)	44% (40%)	47% (40%)	47% (53%)	45% (51%)	49% (47%)	50% (42%)	53% (54%)	54% (55%)	57% (58%)	59% (60%)	64% (64%)	57% (63%)	58% (58%)	62% (63%)	63% (64%)	65% (67%)
Target Factors:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE (MAE)	0,197 (0,149)	0,190 (0,139)	0,157 (0,122)	0,150 (0,113)	0,136 (0,102)	0,117 (0,089)	0,089 (0,083)	0,091 (0,082)	0,128 (0,099)	0,130 (0,098)	0,132 (0,102)	0,094 (0,080)	0,095 (0,085)	0,125 (0,109)	0,121 (0,089)	0,138 (0,097)	0,135 (0,098)	0,144 (0,105)	0,131 (0,097)	0,145 (0,103)
Benefit	32% (33%)	35% (38%)	46% (45%)	48% (49%)	53% (54%)	60% (60%)	69% (63%)	69% (64%)	56% (56%)	55% (56%)	55% (55%)	68% (64%)	67% (62%)	57% (51%)	58% (60%)	53% (57%)	54% (56%)	50% (53%)	55% (57%)	50% (54%)
Boosting Factors:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE (MAE)	0,181 (0,130)	0,176 (0,128)	0,174 (0,125)	0,173 (0,121)	0,158 (0,117)	0,153 (0,114)	0,150 (0,113)	0,145 (0,111)	0,143 (0,108)	0,139 (0,107)	0,129 (0,104)	0,125 (0,101)	0,123 (0,100)	0,118 (0,099)	0,109 (0,094)	0,106 (0,092)	0,100 (0,084)	0,098 (0,080)	0,091 (0,071)	0,087 (0,069)
Benefit	39% (42%)	41% (43%)	42% (44%)	42% (46%)	47% (48%)	49% (49%)	50% (50%)	51% (50%)	52% (52%)	53% (52%)	57% (54%)	58% (55%)	59% (55%)	60% (56%)	63% (58%)	64% (59%)	66% (62%)	67% (64%)	69% (68%)	71% (69%)
RF:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE (MAE)	0,197 (0,157)	0,181 (0,143)	0,164 (0,132)	0,159 (0,128)	0,144 (0,120)	0,141 (0,119)	0,140 (0,118)	0,130 (0,116)	0,129 (109)	0,122 (0,096)	0,113 (0,084)	0,111 (0,077)	0,106 (0,071)	0,092 (0,064)	0,079 (0,060)	0,074 (0.053)	0,073 (0,047)	0,066 (0,044)	0,059 (0,036)	0,051 (0,034)
Benefit	32% (28%)	38% (35%)	43% (40%)	45% (42%)	50% (45%)	51% (46%)	52% (46%)	55% (47%)	56% (50%)	58% (56%)	61% (62%)	62% (65%)	63% (68%)	68% (71%)	73% (78%)	74% (56%)	75% (79%)	77% (80%)	80% (84%)	82% (84%)
RF/OLS:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE (MAE)	0,238 (0,191)	0,214 (0,188)	0,182 (0,179)	0,175 (0,177)	0,167 (0,171)	0,158 (0,166)	0,152 (0,164)	0,146 (0,162)	0,143 (0,154)	0,139 (0,108)	0,139 (0,093)	0,129 (0,087)	0,127 (0,081)	0,125 (0,081)	0,121 (0,052)	0,112 (0,046)	0,110 (0,023)	0,038 (0,013)	0,027 (0,008)	0,018 (0,004)
Benefit	15% (9%)	24% (10%)	35% (14%)	38% (15%)	41% (18%)	44% (21%)	46% (22%)	48% (22%)	49% (26%)	51% (48%)	51% (56%)	54% (58%)	55% (61%)	56% (61%)	57% (75%)	60% (78%)	61% (89%)	87% (94%)	90% (96%)	93% (98%)
adaLASSO/RF:																				
h	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
RMSE (MAE)	0,202 (0,161)	0,175 (0,129)	0,179 (0,143)	0,150 (0,116)	0,168 (0,127)	0,166 (0,129)	0,142 (0,107)	0,128 (0,100)	0,130 (0,092)	0,137 (0,105)	0,132 (0,111)	0,138 (0,102)	0,124 (0,090)	0,136 (0,100)	0,131 (0,106)	0,106 (0,080)	0,125 (0,100)	0,102 (0,077)	0,123 (0,097)	0,095 (0,069)
Benefit	35% (33%)	44% (46%)	42% (41%)	52% (52%)	46% (47%)	47% (46%)	54% (55%)	59% (58%)	58% (62%)	56% (56%)	58% (54%)	56% (58%)	60% (62%)	56% (58%)	58% (56%)	66% (67%)	60% (58%)	67% (68%)	60% (60%)	69% (71%)

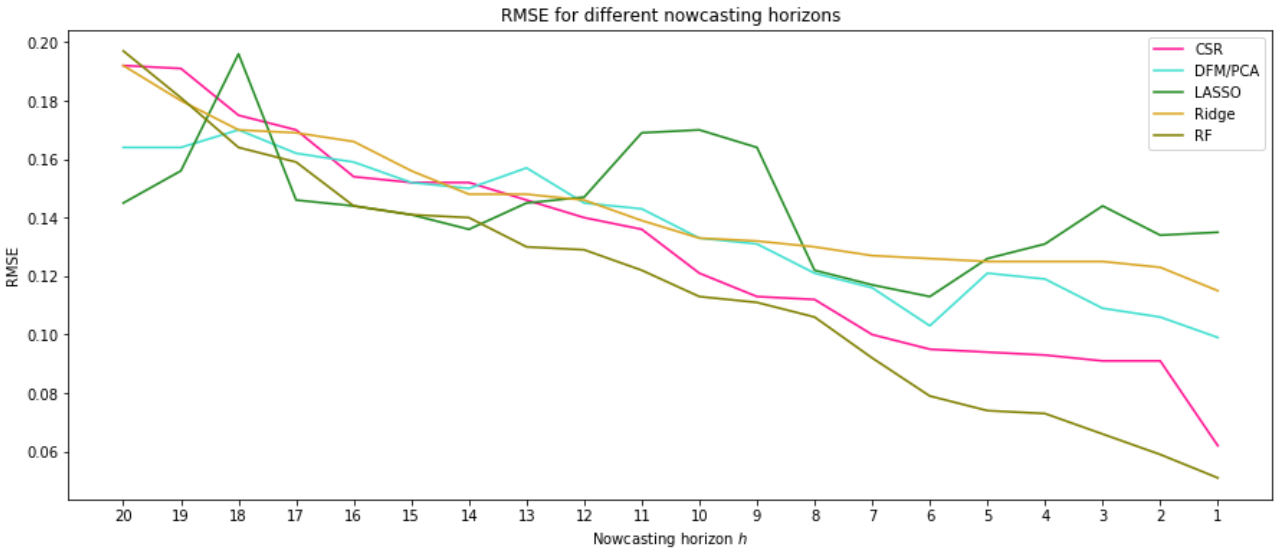


Figure 8: RMSE for different nowcasting horizons.

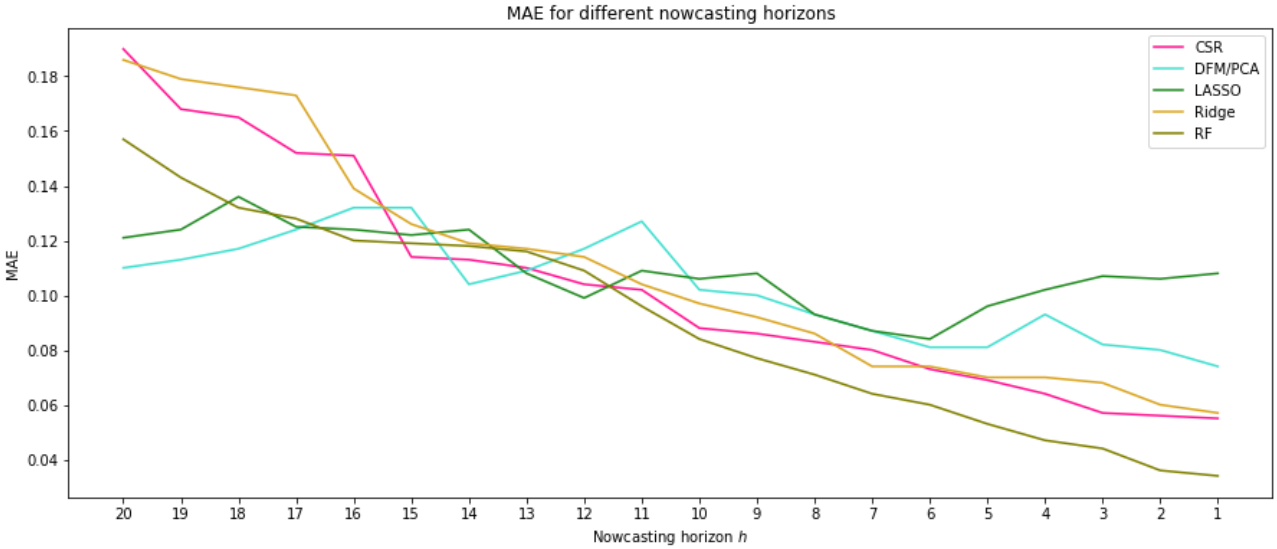


Figure 9: MAE for different nowcasting horizons.

6.3 Feature informativeness

For real-time nowcasting with high-dimensional data, it is crucial to know which features are the most informative so that to keep track of their releases more vigilantly. This subsection shows which variables were considered the most informative by three models: LASSO, Ridge Regression and Random Forest. The choice of these models is due to their different approaches: LASSO is linear and sparsity-inducing, RR is linear and nonsparsity-inducing, and RF is highly nonlinear and nonsparsity-inducing. It was verified that the sets of variables selected by these models for different horizons h are quite similar, so let us consider only models with $h = 1$.

The LASSO model selected only 7 variables from the dataset; all other features were set to zero. The selected features are: Monitor IPCA Ponta, Monitor IPCA-15 Ponta, Brazil ANBIMA IPCA Inflation Assumption for NTN-B M+0, Brazil CPI IPCA IBGE Transportation Inflation, Brazil CPI INPC, FGV Brazil CPI IPC-DI, and FGV Brazil Construction Prices INCC-10. Two groups of variables are represented here: (I) Prices and (II) Money and Finance; another groups were considered uninformative by the model. From these 7 features, 3 of them have daily frequency and 4 have monthly frequency (and thus were interpolated to daily frequency via Kalman Filter). It is worth noting that, from four Monitors, the Ponta versions were selected. As stated in Section 2, the Ponta, which is 7-days-index, anticipates the movements of the 30-days-index. Being selected by LASSO algorithm indicates that it is more useful than 30-days Monitors for the purpose of daily nowcasting of inflation. As can be consulted in Table A in Appendix, the ANBIMA projection for inflation is used for NTN-B yield calculator to adjust par value until the IPCA release date. Also, as pointed out in Table A, Transportation is the second most important item in composition of IPCA index, corresponding to 21,9527% of IPCA, according to POF 2008/2009 (Research of Household Budget). Being selected by LASSO confirms its informativeness about IPCA.

The Ridge Regression model did not set any of the 173 features to zero; the smallest coefficients are very close to zero but are not exactly zero as in LASSO (the reason for that was illustrated in Figure 5). To show which features were considered the most informative by the RR model, the features are ranked according to the absolute value of their weights (recall that all the features have been previously standardized so this comparison is valid). Below, the first 20 most informative features are listed (in decreasing order of informativeness):

1) Brazil Reserve Requirements of Financial Institutions - Savings Deposits;
2) Monitor IPCA;
3) Monitor IPCA-15;
4) Brazil ANBIMA IPCA Inflation Assumption for NTN-B M+0;
5) Monitor IPCA Ponta;
6) Brazil Current Account Balance on Goods and Services;
7) Brazil IPCA-15 CPI Extended National;
8) FGV Brazil CPI IPC-10;
9) Brazil CPI INPC;
10) Brazil CPI IPCS Weekly;

11) Brazil ANP Sales of Ethanol by State - National Total;
12) Brazilian States Debt;
13) Monitor IPCA-15 Ponta;
14) Brazil CPI IPCA IBGE Transportation Inflation;
15) Brazil Financial Account Loans Net Incurrence of Liabilities;
16) Brazil Money Supply M3;
17) Brazil Federal Income Agency Tax Collection Nominal;
18) Brazil Business Loans 15 to 90 Days Late;
19) Brazil Commercial Banks Foreign Exchange Position;
20) Nominal Exchange Rate BRL USD.

All the groups of features are represented in this selection, except for two groups: (VI) Labor and Employment and (VII) Expectations: neither feature from these two groups appears among the first 53 most informative variables in this model. Analyzing the 20 most informative features, it comes out that the most represented group is (I) Prices (as was expected to be), followed by (II) Money and Finance, then by (V) Public Sector, (IV) External Sector and (III) Production and Sales. All the four Monitors entered this list, and the most informative monitors are the 30-days ones. Five of the seven features selected by LASSO entered this list, which corroborates their importance. The list above also evidences the importance of daily- and weekly-frequency features: they stand for 35% of this list, whereas they represent 20% of the original dataset.

To assess the informativeness of the features in the Random Forest model, two metrics are used: the *% Increase in MSE* and the *Increase in Node Purity*.

To calculate the *% Increase in MSE* for each feature $n = 1, ..., N$, the following algorithm is executed:

1) For each tree $m = 1, ..., M$, compute the MSE on the OOB (out-of-bag) portion[17] of the data.

2) For each feature $n = 1, ..., N$:

   a) Randomly permute its values in OOB samples. Then, repeat step (1);

   b) For each tree, calculate the (normalized) difference between the two MSE's: the MSE of step (1) and the MSE of step (2a);

   c) Calculate the average, over the trees, of the decrease in accuracy due to the permutation. Express this result in percentage. This is the *% Increase in MSE* measure for the feature n.

[17] Out-of-Bag portion of the data are observations that were left out-of-sample during bootstrap sampling. This portion is usually about one-third of the original sample's number of observations. Indeed, when sampling T instances with replacement from the original sample of size T, the probability that an observation t is never sampled is $\left(1 - \frac{1}{T}\right)^T$. Now, take $\lim_{T\to\infty}\left(1 - \frac{1}{T}\right)^T = \frac{1}{e} \approx 0,368$.

The higher the %IncMSE for a feature n, the more important is this feature, because using the same model with the data that is the same except for this feature deteriorates the predictive power of the model.

The *Increase in Node Purity* is the total decrease in node impurities (measured by the Residual Sum of Squares) from splitting the feature n, averaged over all trees. The higher the IncNodePurity, the more informative is the feature.

Table 5 shows the 20 most informative features according to %IncMSE and Table 6 does the same according to IncNodePurity.

Table 5: The 20 most informative features in the Random Forest model according to % Increase in MSE.

Feature	**%IncMSE**
1) Monitor IPCA	19,35968551
2) Monitor IPCA Ponta	13,21532067
3) Monitor IPCA-15 Ponta	12,26039434
4) Brazil CPI IPCA IBGE Food Inflation	7,92423957
5) Monitor IPCA-15	7,49133694
6) Brazil Total Electricity Consumption	6,03749282
7) Brazil CPI IPCA IBGE Hous Inflation	6,00058733
8) Brazil Money Supply M1	5,50997775
9) Brazil Financial Account Direct Investment Intercompany Assets	5,24058485
10) Brazil Fed Govt credit provided to Official Financial institutions in % of GDP	5,22827035
11) Brazil Industrial Production Activity Extractive Industry	5,22219543
12) Brazil Monetary Base Bank Reserves	5,0766168
13) Brazil International Reserves Liquidity Concept Total US$	5,07587178
14) FGV Brazil CPI IPC-DI	4,96540274
15) Brazil CNI Consumer Confidence Household Debt Situation	4,95498586
16) Brazil CPI IPCA Coefficient of Variation Market Expectation Next 12 Months	4,94316227
17) CNI Brazil Manufacture Industry Real Wages	4,90652866
18) Brazil Public Net Fiscal Debt % of GDP	4,85433722
19) Brazil Fin Acct Portfolio Investment Acquisition of Financial Assets Credit	4,84533886
20) Secovi Brazil Real Estate Units Average Sale Time Period	4,83955007

Table 6: The 20 most informative features in the Random Forest model according to Increase in Node Purity.

Feature	**Inc. Node Purity**
1) Monitor IPCA	614
2) Monitor IPCA-15	95,3
3) Brazil ANBIMA IPCA Inflation Assumption for NTN-B M+0	3,72
4) Monitor IPCA Ponta	1,37
5) Monitor IPCA-15 Ponta	0,978
6) Brazil CPI IPCS Weekly	0,202
7) Brazil Auto Sales Total	0,152
8) Brazil CPI IPCA IBGE Food Inflation	0,149
9) Brazil Business Loans 15 to 90 Days Late	0,136
10) Bloomberg Barclays EMGILB Ex-Brazil Govt Inflation-Linked 1-10yrs CPI	0,120
11) Brazil Amplified Retail Sales Volume	0,102
12) Brazil Real Minimum Wage	0,0958
13) Brazil Manufactured Products Tax Income Nominal	0,0952
14) Brazil Financial Index	0,0951
15) Brazilian States Debt to Foreigners in % of GDP	0,0909
16) Brazil ANBIMA Estimated Index Assumption IGP-M	0,0902
17) Ibovespa Index	0,0890
18) Brazil Income Tax Collection Nominal	0,0871
19) CNI Brazil Manufacture Industry Capacity Utilization	0,0850
20) Brazil Total Electricity Consumption	0,0848

Analyzing Tables 5 and 6, the following conclusions can be made:

- The four Monitors are on the top of the list, and the most informative one is the Monitor IPCA;
- The Food component of the IPCA inflation is one of the most informative features. In fact, according to according to POF 2008/2009 (Research of Household Budget), this is the most representative component in the IPCA index, having the weight 22,0828% in the IPCA.
- Brazil ANBIMA IPCA Inflation Assumption for NTN-B M+0 is confirmed, by Table 6, to be one of the most informative features, as was already indicated by LASSO and Ridge variable selection.
- All the groups of features are represented in these Tables, i.e., these lists are more diversified. In Table 5, for example, the Prices Group is the most represented: it has 7 features among the most informative. The second most represented group in Table 5 is Production and Sales (4 features), followed by Money and Finance (3 features), External Sector (2 features), Public Sector (2 features) and Labor and Employment (1 feature) and Expectations (1 feature). It is worth noting that, differently from the Shrinkage models, here the Production and Sales features are more important.

– In Table 5, Money Supply M1 appears in the top 10 features, which can indicate the importance of the Quantitative Theory of Money to analyze the inflation process.

The figures 10-13 illustrate the word clouds for LASSO, Ridge Regression and Random Forest (%IncMSE and IncNodePurity) variable selection. The names of the features displayed in these Figures are their Bloomberg tickers (consult Table A in appendix).

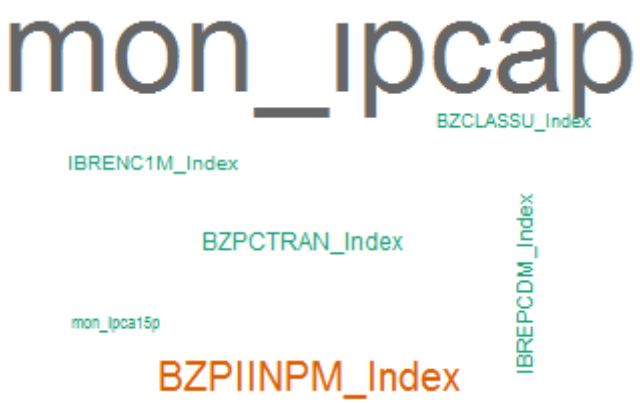


Figure 10: Word cloud for LASSO variable selection.

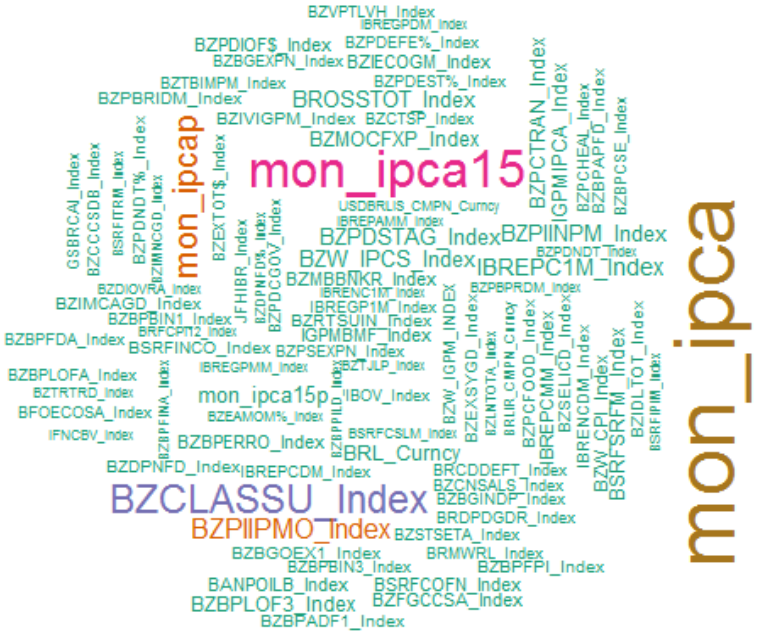


Figure 11: Word cloud for Ridge Regression variable selection.

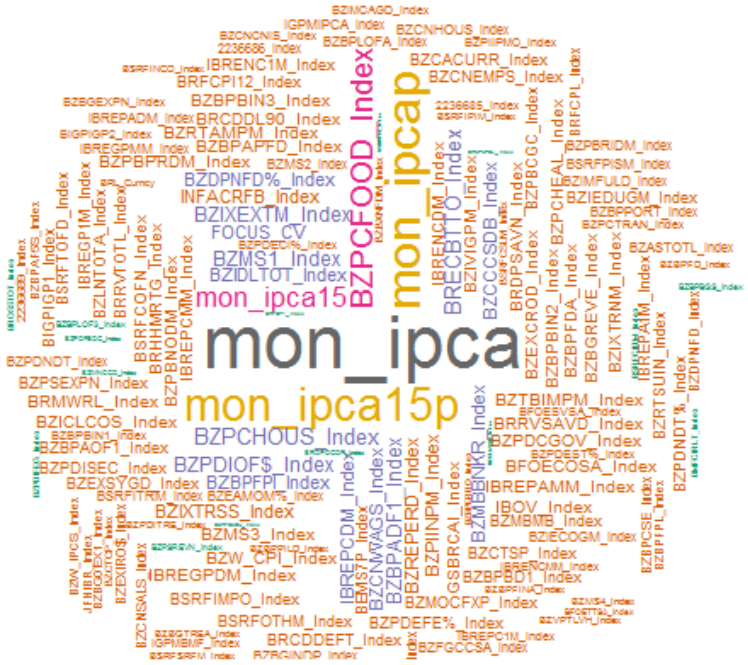


Figure 12: Word cloud for Random Forest variable selection according to % Increase in MSE.

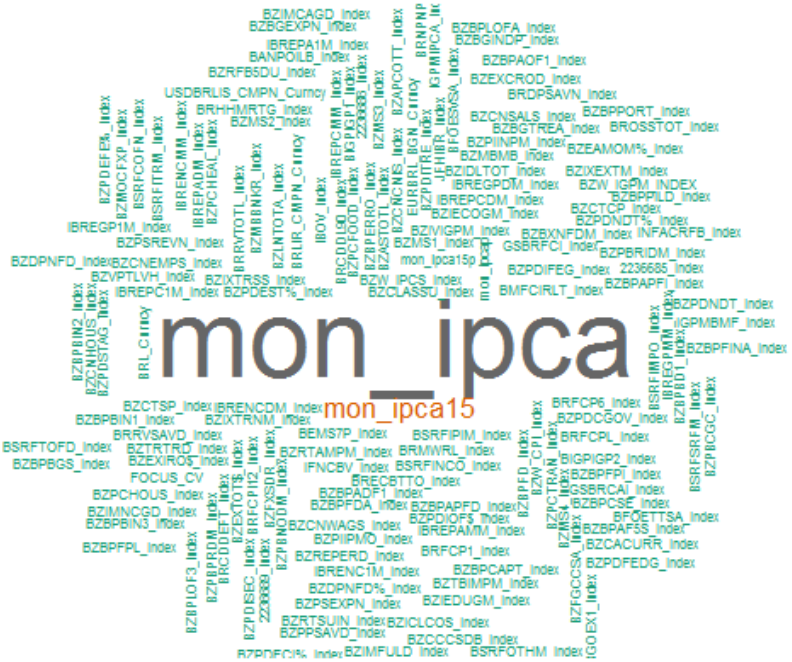


Figure 13: Word cloud for Random Forest variable selection according to Increase in Node Purity.

7 Conclusion

This paper presented original empirical results of nowcasting Brazilian IPCA inflation rate daily, using Machine Learning techniques. Firstly, a large mixed-frequency dataset was collected, and Kalman Filter was used to balance the panel, interpolating all the features to daily frequency. The interpolation used daily covariates to increase the accuracy of the filter. Then, two univariate models and 14 Machine Learning models were learned on the sample from 01/12/2006 till 08/10/2014 and tested on Testing Sample from 09/10/2014 till 03/10/2018. Their predictive quality was assessed by the RMSE and MAE functionals. The 3 tasks that were stated in the Introduction as the aims of this work were performed, and their results can be summarized as follows: (1) There is a benefit of using data-intensive Machine Learning techniques relatively to univariate benchmarks that varies from 4% to 20%; (2) There is a large benefit of nowcasting inflation daily instead of forecasting it one a month, and this benefit varies from 50% to 60%, on average; (3) The most informative features are the daily-frequency ones, especially the Monitors of inflation, produced by FGV.

This work is all about measuring the benefits of daily nowcasting; however, the *costs* should also be considered and "put on the other plate of the scale". Complex ML models, such as RF, have high computational cost, and, the larger the database, the more time it takes to be learned. Hence, feature proliferation should be avoided, and the efforts should be concentrated on the few most informative features. It seems that the most important path for future research in this topic is optimization of the computer codes that perform nowcasting, because inflation nowcasting is a task that imposes time constraints on the velocity of the codes. If a code has a cost of completing this task in some days, the nowcasting exercise becomes meaningless because until then the true inflation rate becomes known.

Given this velocity requirement to codes for nowcasting, there is a wide range of further research topics that can develop this research program. Alternative Machine Learning methods can be implemented, such as deep Neural Networks, Bayesian VAR etc. Cutting-edge research in this area can lead to creation of new, more efficient, Machine Learning techniques that perform even better the task of inflation nowcasting. Creation of new, more efficient models is a product of trial-and-error, which is the essence of the art of pattern recognition.

Another suggestion for future research is to implement a combination of nowcasts. This can be made, for example, by dividing the sample in three parts (for estimation of the models, estimation of the weights for each model and computation of prediction errors) and using RF or adaLASSO for model selection (instead of feature selection).

8 References

[1] Andreou, E., Ghysels, E., Kourtellos, A. (2013), Should macroeconomic forecasters use daily financial data and how?, *Journal of Business and Economic Statistics*, 31(2), 240–251.

[2] Arruda, E., Ferreira, R., Castelar, I. (2011), Modelos lineares e não lineares da Curva de Phillips para previsão da taxa de inflação no Brasil, *Revista Brasileira de Economia*, 65, 237–252.

[3] Bai, J. and Ng, S. (2008), Forecasting economic time series using targeted predictors, *Journal of Econometrics*, 146, 304-317.

[4] Bai, J. and Ng, S. (2009), Boosting diffusion indexes, *Journal of Applied Econometrics*, 24, 607-629.

[5] Banbura, M., Giannone, D. and Reichlin, L. (2010), Large Bayesian vector autoregressions, *Journal of Applied Econometrics* 25, 71-92.

[6] Banbura, M., Giannone, D., Modugno, M., and Reichlin, L. (2013), Now-casting and the real-time data flow, *Handbook of economic forecasting* 2, 193–224.

[7] Bernanke, B. S., Gertler, M., & Watson, M. (1997), Systematic monetary policy and the effects of oil price shocks, *Brookings Papers in Economic Activity No. 1*, Washington, DC: Brookings Institution.

[8] Bishop, C.M. (2006), Pattern Recognition and Machine Learning, *Springer*.

[9] Breiman, L. (1996), Bagging predictors, *Machine Learning* 24(2), 123-140.

[10] Breiman, L. (2001), Random forests, *Machine Learning* 45, 5-32.

[11] Chakraborty, C. and Joseph, A. (2017), Machine learning at central banks, *Staff Working Paper*, 674, Bank of England.

[12] Chamberlain, G., and Rothschild, M. (1983), Arbitrage Factor Structure, and Mean-Variance Analysis of Large Asset Markets, *Econometrica*, 51, 1281-1304.

[13] Colombo, E., Pelagatti, M. (2019), Statistical learning and exchange rate forecasting, *Quaderni del Dipartimento di Economia internazionale, delle istituzioni e dello sviluppo*, dis1901.

[14] Cortes, C. and Vapnik, V. (1995), Support vector networks, *Machine Learning*, 20, 273–297.

[15] Dhrymes, P. (1974), Econometrics: Statistical Foundations and Applications, *Springer*, Ch.2, pp. 53-65.

[16] Doz, C., Giannone, D. and Reichlin, L. (2011), A two-step estimator for large approximate dynamic factor models based on kalman filtering, *Journal of Econometrics*, 164(1), 188–205.

[17] Elliott, G., Gargano, A. and Timmermann, A. (2013), Complete subset regressions, *Journal of Econometrics* 177(2), 357-373.

[18] Elliott, G., Gargano, A. and Timmermann, A. (2015), Complete subset regressions with large-dimensional sets of predictors, *Journal of Economic Dynamics and Control* 54, 86-110.

[19] Engle, R.F. and Watson, M.W. (1981), A One-Factor Multivariate Time Series Model of Metropolitan Wage Rates, *Journal of the American Statistical Association*, 76, 774-781.

[20] Engle, R.F. and Watson, M.W. (1983), Alternative Algorithms for Estimation of Dynamic MIMIC, Factor, and Time Varying Coefficient Regression Models, *Journal of Econometrics*, 23, pp. 385-400.

[21] Flannery, M. J. and Protopapadakis, A. (2002), Macroeconomic Factors do Influence Aggregate Stock Returns, *Review of Financial Studies*, 15, 751-782.

[22] Forni, M., Giannone, D., Lippi, M., and Reichlin, L. (2009), Opening the black box: Structural factor models with large cross sections, *Econometric Theory*, 25(5), 1319–1347.

[23] Garcia, M., Medeiros, M., Vasconcelos, G. (2017), Real-time inflation forecasting with high-dimensional models: The case of Brazil, *International Journal of Forecasting*, 33(3), 679-693.

[24] Giannone, D., Reichlin, L., Sala, L. (2004), Monetary Policy in Real Time, *NBER Macroeconomics Annual*, pp. 161–200.

[25] Giannone, D., Reichlin, L., Small, D. (2008), Nowcasting: The real-time informational content of macroeconomic data, *Journal of Monetary Economics*, 55(4), 665–676.

[26] Hamilton, J.D. (1994), Time Series Analysis, *Princeton University Press*, Princeton.

[27] Hansen, B. E. & Racine, J. S. (2012), Jackknife model averaging, *Journal of Econometrics* 167(1), 38-46.

[28] Harvey, A.C. (1989), Forecasting, Structural Time Series Models and the Kalman Filter, *Cambridge University Press*, Cambridge.

[29] Hastie, T., Tibshirani, R., Friedman, J. (2001), The Elements of Statistical Learning, *Springer*.

[30] Hoerl, A. E., Kennard, R. W. (1970), Ridge regression: Biased estimation for nonorthogonal problems, *Technometrics* 12(1), 55-67.

[31] Hotelling, H. (1933), Analysis of a complex of statistical variables into principal components, *Journal of Educational Psychology*, 24(6), 417-441.

[32] Huang, W., Nakamori, Y., Wang, S.-Y. (2005), Forecasting stock market movement direction with support vector machine, *Computers and Operations Research*, 32, 2513–2522.

[33] Issler, J. A. V., Notini, H. H., (2016), Estimating Brazilian Monthly GDP: a State-Space Approach, *Revista Brasileira de Economia* 70, 41 – 59.

[34] Medeiros, M. and Vasconcelos, G. (2016), Forecasting macroeconomic variables in data-rich environments, *Economics Letters* 138, 50-52.

[35] Medeiros, M., Zilberman, E., Vasconcelos, G., Veiga, A. (2019), Forecasting Inflation in a Data-Rich Environment: The Benefits of Machine Learning Methods, *Journal of Business and Economic Statistics*, 1-45.

[36] Modugno, M. (2013), Nowcasting Inflation using High-Frequency Data, *International Journal of Forecasting*, 29, 664-675.

[37] Mönch, E., & Uhlig, H. (2005), Towards a monthly business cycle chronology for the Euro Area, *Journal of Business Cycle Measurement and Analysis*, 2005(1), 43–69.

[38] Mullainathan, S., Spiess, J. (2017), Machine learning: An applied econometric approach, *Journal of Economic Perspectives*, 31, 87-106.

[39] Pearson, K. (1901), On Lines and Planes of Closest Fit to Systems of Points in Space, *Philosophical Magazine*, 2 (6), 559–572.

[40] Quah, D., and Sargent, T.J. (1993), A Dynamic Index Model for Large Cross Sections, *Business Cycles, Indicators, and Forecasting*, 285-310.

[41] Richardson, A., Mulder, T.v.F., Vehbi, T. (2018), Nowcasting New Zealand GDP Using Machine Learning Algorithms, *Centre for Applied Macroeconomic Analysis* Working Paper.

[42] Sargent, T.J. (1989), Two Models of Measurements and the Investment Accelerator, *Journal of Political Economy*, 97, 251–287.

[43] Sargent, T. J., Sims C.A. (1977), Business Cycle Modeling Without Pretending to Have Too Much A-Priori Economic Theory, in *New Methods in Business Cycle Research*, ed. by C. Sims. Federal Reserve Bank of Minneapolis.

[44] Stock, J.H., and Watson, M.W. (1989), New Indexes of Coincident and Leading Economic Indicators, *NBER Macroeconomics Annual*, 351-393.

[45] Stock, J. H. and Watson, M. W. (2002a), Forecasting using principal components from a large number of predictors, *Journal of the American statistical association*, 97(460), 1167–1179.

[46] Stock, J. H. and Watson, M. W. (2002b), Macroeconomic forecasting using diffusion indexes, *Journal of Business and Economic Statistics*, 20(2), 147–162.

[47] Stock, J.H. and Watson, M.W. (2011), Dynamic Factor Models, In: *Oxford Handbook on Economic Forecasting*, eds. Michael P. Clements and David F. Hendry. Oxford: Oxford University Press.

[48] Stock, J. H. and Watson, M. W. (2016), Dynamic Factor Models, Factor-Augmented Vector Autoregressions, and Structural Vector Autoregressions in Macroeconomics. In: *Handbook of Macroeconomics*, Volume 2A, 415-525.

[49] Stock, J. H., Watson, M. W. (2017), Twenty Years of Time Series Econometrics in Ten Pictures, *Journal of Economic Perspectives*, 31 (2), 59-86.

[50] Tibshirani, R. (1996), Regression shrinkage and selection via the LASSO, *Journal of the Royal Statistical Society*, Series B (Methodological) 58, 267-288.

[51] Varian, Hal R. (2014), Big Data: New Tricks for Econometrics, *Journal of Economic Perspectives*, 28(2), 3–28.

[52] Watson, M.W., (2004), Comment on Giannone, Reichlin, and Sala, *NBER Macroeconomics Annual*, 216-221.

[53] Xie, W., Yu, L., Xu, S., Wang, S. (2006), A New Method for Crude Oil Price Forecasting Based on Support Vector Machines, *International Conference on Computational Science*, 2006.

[54] Zhang, X., Wan, A. T. and Zou, G. (2013), Model averaging by jackknife criterion in models with dependent data, *Journal of Econometrics* 174(2), 82-94.

[55] Zou, H. (2006), The adaptive lasso and its oracle properties, *Journal of the American Statistical Association*, 101(476), 1418-1429.

[56] Zou, H., Hastie, T. (2005), Regularization and variable selection via the elastic net, *Journal of the Royal Statistical Society*: Series B (Statistical Methodology) 67(2), 301-320.

9 Appendix

Table A — Dataset and transformations of the data

	Name of series	Bloomberg Ticker	Format*	Fr. **	Source	Tr. ***	Description
Group I: Prices							
0	Brazil CPI IPCA MoM	BZPIIPCM Index	NSA MoM%	M	IBGE	(0)	See Section 1
1	Monitor IPCA	mon_ipca****	NSA MoM%	D	FGV	(0)	See Section 2
2	Monitor IPCA Ponta	mon_ipcap	NSA MoM%	D	FGV	(0)	See Section 2
3	Monitor IPCA-15	mon_ipca15	NSA MoM%	D	FGV	(0)	See Section 2
4	Monitor IPCA-15 Ponta	mon_ipca15p	NSA MoM%	D	FGV	(0)	See Section 2
5	Brazil CPI IPCA IBGE Food Inflation MoM	BZPCFOOD Index	NSA MoM%	M	IBGE	(0)	According to POF 2008/2009 (Research of Household Budget), foods and beverages have weight of 22,0828% in the IPCA.
6	Brazil CPI IPCA IBGE Transportation Inflation MoM	BZPCTRAN Index	NSA MoM%	M	IBGE	(0)	According to POF 2008/2009 (Research of Household Budget), transportation has weight of 21,9527% in the IPCA.
7	Brazil CPI IPCA IBGE Hous Inflation MoM	BZPCHOUS Index	NSA MoM%	M	IBGE	(0)	According to POF 2008/2009 (Research of Household Budget), housing has weight of 14,2752% in the IPCA.
8	Brazil CPI IPCA IBGE Health Inflation MoM	BZPCHEAL Index	NSA MoM%	M	IBGE	(0)	According to POF 2008/2009 (Research of Household Budget), health and personal care have weight of 11,0797% in the IPCA.
9	Brazil IPCA-15 CPI Extended National MoM	BZPIIPMO Index	NSA MoM%	M	IBGE	(0)	The collection of prices occurs between 16th day of the previous month and 15th day of the reference month t. Is published approximately on the 23th day of the month t.
10	Brazil CPI INPC MoM	BZPIINPM Index	NSA MoM%	M	IBGE	(0)	
11	FGV Brazil General Prices IGP-10 MoM	IBREGP1M Index	NSA MoM%	M	FGV	(0)	The collection of prices occurs between 11th day of the previous month and 10th day of the reference month t. Is published approximately on the 20th day of the month t.
12	FGV Brazil General Prices IGP-M MoM	IBREGPMM Index	NSA MoM%	M	FGV	(0)	The collection of prices occurs between 21th day of the previous month and 20th day of the reference month t. Is published approximately on the 29th day of the month t.

13	FGV Brazil General Prices IGP-DI MoM	IBREGPDM Index	NSA MoM%	M	FGV	(0)	The collection of prices occurs between 1st day of the previous month and 30th day of the reference month t. Is published approximately on the 10th day of the month $t+1$. The only difference between IGP-10, IGP-M and IGP-DI indexes is the calculation period.
14	FGV Brazil Wholesale Prices IPA-10 MoM	IBREPA1M Index	NSA MoM%	M	FGV	(0)	
15	FGV Brazil IGP-M Wholesale Prices IPA-M MoM	IBREPAMM Index	NSA MoM%	M	FGV	(0)	
16	FGV Brazil Wholesale Prices IPA-DI MoM	IBREPADM Index	NSA MoM%	M	FGV	(0)	
17	FGV Brazil CPI IPC-10 MoM	IBREPC1M Index	NSA MoM%	M	FGV	(0)	
18	FGV Brazil IGP-M CPI IPC-M MoM	IBREPCMM Index	NSA MoM%	M	FGV	(0)	The general price index measures a broader range of Brazilian inflation than the consumer price index. It is constructed from 3 price indexes: wholesale price index IPA (60%), consumer price index IPC (30%) and an index of national construction costs INCC (10%). This weighting scheme aims to reproduce the value added of each sector of the economy (wholesale, retail and construction) at the time it was introduced in the 1940s.
19	FGV Brazil CPI IPC-DI MoM	IBREPCDM Index	NSA MoM%	M	FGV	(0)	Measures retail prices inflation. The collection of prices occurs in the cities of Rio de Janeiro and São Paulo within households with income from 1 to 33 minimum wages. Represents 33% of IGP-DI.
20	FGV Brazil Construction Prices INCC-10 MoM	IBRENC1M Index	NSA MoM%	M	FGV	(0)	
21	FGV Brazil IGP-M Construction Prices INCC-M MoM	IBRENCMM Index	NSA MoM%	M	FGV	(0)	
22	FGV Brazil Construction Prices INCC-DI MoM	IBRENCDM Index	NSA MoM%	M	FGV	(0)	
23	Brazil CPI Fipe Weekly	BZW CPI Index	NSA MoM%	W	FIPE	(0)	CPI-Fipe - Consumer Price Index from Fundação Instituto de Pesquisas Econômicas/University of São Paulo. The index reflects the cost of living of families with income of 1 to 20 minimum wages in the city of São Paulo. Index with a base year of 7/94=100.

24	Brazil CPI IPCS Weekly	BZW IPCS Index	NSA MoM%	W	FGV	(0)	IPC-S is a consumer price index reported weekly; it compares prices of a recent 30-day period to a similar 30-day period from a month ago.
Group II: Money and Finance							
25	Brazil Monetary Base	BZMBMB Index	NSA Value BRL Billions	M	BCB	(1)	This concept tracks notes and coins in circulation plus minimum reserves that credit institutions hold with the central bank. It is sometimes also referred to as base money. The format Value refers to any economic concept that is reported in currency (e.g., gross domestic product, financial account, exports etc).
26	Brazil Money Supply M1	BZMS1 Index	NSA Value BRL Billions	M	BCB	(1)	The money supply measures the total amount of money in circulation in a country or group of countries in a monetary union.
27	Brazil Money Supply M2	BZMS2 Index	NSA Value BRL Billions	M	BCB	(1)	
28	Brazil Money Supply M3	BZMS3 Index	NSA Value BRL Billions	M	BCB	(1)	
29	Brazil Money Supply M4	BZMS4 Index	NSA Value BRL Billions	M	BCB	(1)	
30	Brazil Monetary Base Bank Reserves	BZMBBNKR Index	NSA Value BRL Billions	M	BCB	(1)	
31	Brazil Reserve Requirements of Financial Institutions - Savings Deposits	BRRVSAVD Index	NSA Value BRL Millions	M	BCB	(1)	Reserve requirements are the minimum reserves required for depository institutions. They are set by the central bank within limits specified by laws for depository institutions. A change in the minimum reserve ratio affects the amount of its deposit base a financial institution can lend out. Reserve requirements are an instrument of monetary policy.
32	Brazil Total Reserve Requirements of Financial Institutions	BRRVTOTL Index	NSA Value BRL Millions	M	BCB	(1)	

33	Brazil International Reserves - Liquidity Concept - Total - US$ MM	BZIDLTOT Index	NSA Value USD Millions	M	BCB	(1)	International reserves are liquid assets held by a country's central bank or other monetary authority in order to implement monetary policies effecting the country's currency exchange rate and ensuring the payment of its imports. The assets include foreign currency and foreign denominated bonds, gold reserves, SDRs (special drawing rights) and the IMF reserve position.
34	Brazil Financial System Loans	BZLNTOTA Index	NSA Value BRL Billions	M	BCB	(1)	
35	Brazil Personal Loans More Than 90 Days Late	BRCDDEFT Index	NSA %	M	BCB	(1)	This concept tracks loans that are in default or close to being in default. A loan is nonperforming when payments of interest and principal are past due for a specified period of time (e.g. 90 days or more).
36	Brazil Nonperforming Loans of Public Financial Institutions	BRNPNPBL Index	NSA %	M	BCB	(4)	Idem.
37	Brazil Business Loans 15 to 90 Days Late	BRCDDL90 Index	NSA %	M	BCB	(1)	Idem.
38	Brazil Household Debt without Mortgage Payments as % of Disposable Income	BRHHMRTG Index	NSA %	M	BCB	(5)	Consumer or Household Credit tracks the outstanding amount of credit (or loans) used by consumers to finance purchases of goods or services. At its broadest, this concept can include everything from credit card lending, to auto loans, to lines of credit and mortgages.
39	Brazil BNDES Long Term Interest Rate TJLP	BZTJLP Index	NSA %	D	BNDES	(4)	The long-term interest rate target (Taxa de Juros de Longo Prazo or TJLP) is set quarterly by the National Monetary Council. The rate is used as the benchmark rate for loans from the Brazilian Development Bank to companies.
40	Brazil Commercial Banks Foreign Exchange Position	BZMOCFXP	Value	M	BCB	(1)	
41	Brazil Financial Index	IFNCBV Index	Index	D	B3	(4)	The purpose of the Financial Index is to measure the behavior of stocks that are representative of the financial sector, which includes financial intermediaries, miscellaneous financial services, pension funds and insurance companies.

42	Brazil Reference Interest Rate TR	BZTRTRD Index	NSA %	D	BCB	(1)	The Brazilian Daily Reference Rate (Taxa Referencial Diária) is the daily apportionment of the reference rate (TR), which is a monthly index calculated by the Brazilian Central Bank using an adjusted weighted average of the daily total retail CDB (certificates of deposit) operations made by the 30 largest financial institutions of the country in terms of volume of those operations, disconsidering the two largest and two smallest averages. The rate is then multiplied by a reducing factor (redutor), which aims to extract the portion related to the real interest rate and taxes of the CDBs. The apportionment is valid for 30 actual days and is made taking into account the number of settlement days in the period. e.g. The rate on 06/25/2014 (0.1083% per month) is valid through 07/25/2014, in a period comprised of 22 settlement days. The rate on 04/30/2014 (0.0673% per month) is valid through 05/30/2014, in a period comprised of 21 settlement days.
43	BRL Interest Rate Return Curncy	BRLIR CMPN Curncy	Index	D	CMPN Composite (NY)	(5)	The return gained from collecting interest on a currency. Interest rate return only uses the accumulated interest and ignores the spot movement. Base date is 1/1/1999.
44	USD-BRL Int Rate Spread Curncy	USDBRLIS CMPN Curncy	Index	D	CMPN Composite (NY)	(2)	The interest rate spread is calculated by taking the difference between the long interest rate return and the short interest rate return. This is effectively the carry that an investor would be earning or paying assuming there was no spot movement. Base date is 1/1/1999.
45	Brazil BM&F Interest Rates Future Number of Long Contracts	BMFCIRLT Index	Units	D	B3	(4)	Open contracts on the BM&FBOVESPA by all kinds of investors.
46	GS Brazil Financial Conditions Index	GSBRFCI Index	Index	D	Goldman Sachs	(4)	
47	Brazil Savings Accounts Deposit - 1 Day Yield Rate	BZPPSAVD Index	NSA %	D	BCB	(1)	

48	Brazil Cetip DI Interbank Deposit Rate	BZDIOVRA Index	NSA %	D	CETIP	(4)	Brazil's Interbank Deposit Rate Over (known in Brazil as Taxa DI-Over) is the daily average annualized rate calculated by the number of business days in the month, of the one-day interbank deposit rates. This rate is calculated by the Central of Custody and Settlement of Private Bonds (CETIP), based on the period from the transaction date to the last trading day.
49	Brazil Selic Target Rate	BZSTSETA Index	NSA %	D	BCB	(4)	A target interest rate set by the central bank in its efforts to influence short-term interest rates as part of its monetary policy strategy.
50	Brazil Total Savings Deposits	BRDPSAVN Index	NSA Value BRL Millions	M	BCB	(6)	The Brazilian Money Market Deposits (CDB) data series includes rates, inflows, returns and balances, for Fixed, TR Floating, Di Floating and Other Floating securities. The data is calculated in a daily basis and released with a maximum lag of 6 days.
51	EURBRL Spot Exchange Rate - Price of 1 EUR in BRL	EURBRL BGN Curncy	NSA Value	D	BCB	(7)	
52	Brazil Central Bank SDR Avg Rate	BZFXSDR Index	NSA Value	D	BCB	(7)	These are the average exchange rates calculated and informed by the Central Bank of Brazil. SDR stands for Special Drawing Rights. The SDR unit of currency is valued against a composite od the major world currencies.
53	Nominal Exchange Rate BRL USD	BRL Curncy	NSA Value	D	BCB	(7)	
54	Ibovespa Index	IBOV Index	NSA Index	D	B3	(7)	
55	Brazil Money Market CDB DI Floating - Daily Return	BRDPDGDR Index	NSA %	D	BCB	(1)	Daily return of Brazilian Money Market CDB DI Floating.
56	Brazil Selic Average Overnight Daily Rate	BZSELICD Index	NSA %	D	BCB	(1)	An interest rate set in one-day operations, backed by Brazilian government bonds, between financial institutions, registered in the Special System for Settlement and Custody (SELIC). This rate floats near the SELIC Target rate, which is established by the BCB in order to conduct its monetary policy using Open Market Operations as the instrument.

57	Anbima Brazil IPCA Inflation Linked Bonds IMA-B 5 Index Duration	BZRFB5DU Index	NSA Units Business Days	D	ANBIMA	(0)	index of duration (weighted average of times until the cash flow is received) of ANBIMA Brazil IPCA Inflation Linked Bonds IMA-B 5. IMA-B is ANBIMA Market Index of the portfolio composed by NTN-B Bonds, which are Brazilian government bonds linked to IPCA inflation. Duration shows the present value's sensitiveness to changes in interest rate.
58	Brazil ANBIMA IPCA Inflation Assumption for NTN-B M+0	BZCLASSU Index	NSA %	D	ANBIMA	(0)	IPCA inflation assumption is used on NTN-B yield calculator to adjust par value. The assumption is the ANBIMA projection until the IPCA release date, when the actual data replaces the forecast.
59	Brazil BM&F Future Contracts Referential Inflation Rate IPCA	IGPMIPCA Index	NSA %	D	B3	(7)	IPCA inflation reference rate on the B3 (Brazilian Stock Exchange) futures contract exchange.
60	Brazil BM&F Future Contracts Referential Inflation Rate IGPM	IGPMBMF Index	NSA %	D	B3	(7)	IGP-M inflation reference rate on the B3 (Brazilian Stock Exchange) futures contract exchange.
61	Bloomberg Barclays EMGILB Ex-Brazil Govt Inflation-Linked 1-10yrs CPI	BEMS7P Index	NSA Value BRL	D	Bloomberg	(7)	Bloomberg Barclays Emerging Market Government Inflation-Linked Bond Index for Brazil.
62	Crude Oil Futures Front Contract in Brazilian Real	INFACRFB Index	NSA Value BRL	D	Bloomberg	(7)	
63	Brazil ANBIMA Estimated Index Assumption IGP-M	BZIVIGPM Index	NSA %	D	ANBIMA	(0)	Monthly projection of the FGV Inflation indicator IGP-M, released by ANBIMA's Clearing House, to update the par value of Brazilian Inflation-linked Notes NTN-C.
Group III: Production and Sales							
64	Brazil Economic Activity GDP MOM% IBC-BR	BZEAMOM% Index	SA MoM%	M	BCB	(0)	This concept provides an estimate of overall economic activity in an economy. It is typically a monthly measure of value added by industry (sometimes referred to as monthly GDP) that is released in advance of quarterly GDP estimates.
65	GS Brazil Current Activity Indicator	GSBRCAI Index	Index	M	Goldman Sachs	(1)	Goldman Sachs Current economic Activity Indicator.
66	Brazil Industrial Production Activity Manufacturing Industry MoM2012	BZIXTRSS Index	SA MoM%	M	IBGE	(4)	Industrial production measures the output of industrial establishments in the following industries: mining and quarrying, manufacturing and public utilities (electricity, gas and water supply). Production is based on the volume of the output.

67	Brazil Industrial Production Activity Other Chemicals MoM2012	BZIXTRNM Index	SA MoM%	M	IBGE	(0)	Idem
68	Brazil Industrial Production Activity Extractive Industry MoM2012	BZIXEXTM Index	SA MoM%	M	IBGE	(0)	Idem
69	Anfavea Brazil Vehicle Production	BZVPTLVH Index	NSA Units	M	Anfavea	(2)	This concept tracks the number of motor vehicles produced during the reference period.
70	Brazil ANP Oil Production (Barrels)	BANPOILB Index	NSA Volume Barrels	M	ANP	(2)	This ticker shows the total amount of oil produced in Brazil, expressed in barrels of oil (BBL). This sector has the fuel and gasoline sales monthly fundamental data for Brazil by state, which are calculated by the National Agency of Petroleum (ANP). ANP is Brazilian National Agency of Petroleum, Natural Gas and Biofuels. ANP is the federal government agency responsible for regulation of the oil sector. The date is a monthly calculation of different fundamental data. Although it's a monthly report source updates are often delayed. Data is taken directly from the source without any manipulations. Calculation methodologies are regulated by ANP Ordinances and technical notes (Prices include freight and VAT).
71	Brazil Conab Soybean Production Monthly Estimate/Thousand Metric Tons	BZCTSP Index	NSA Quantity 1000 Metric Tonnes	M	CONAB	(4)	This sector consolidates the data released by CONAB (Companhia Nacional de Abastecimento do Brasil) in a monthly basis related to Soybean, Corn, Cotton, Sugarcane, Coffee Crop in Brazil on its report 'Levantamentos de Safra' or 'Crop Estimates'. CONAB publishes an assessment of Brazilian Grains, Oilseeds and Softs surveys for Planted Area, Yield and Production, and also a crop assessment. The latest information is subject to change up to the end of the crop year.
72	Brazil Conab Corn Production Monthly Estimate/Thousand Metric Tons	BZCTCP Index	NSA Quantity 1000 Metric Tonnes	M	CONAB	(1)	Idem

73	Brazil IBGE Cotton Monthly Production Estimated	BZAPCOTT Index	NSA Quantity 1000 Metric Tonnes	M	IBGE	(4)	This sector consolidates Brazilian Agriculture Production estimatives for many different commodities from IBGE/SIDRA program.
74	Brazil Total Electricity Consumption	BRECBTTO Index	NSA Quantity Gigawatthours	M	EPE	(4)	EPE - Empresa de Pesquisa Energética. This data is updated by the source with a two-to-three month lag.
75	Brazil Consumer Goods SA MoM	BZIECOGM Index	SA MoM%	M	IBGE	(0)	
76	Brazil Durable Consumer Goods SA MoM	BZIEDUGM Index	SA MoM%	M	IBGE	(0)	
77	CNI Brazil Manufacture Industry Capacity Utilization SA	BZCNCNIS Index	SA %	M	CNI	(1)	Capacity utilization tracks the extent to which the installed productive capacity of a country is being used in the production of goods and services. For some countries this concept is reported as the percent of capacity being used for production (as opposed to sitting idle). For other countries, this concept is measured through business surveys (tracking business leaders' opinions on their use of productive capacity).
78	CNI Brazil Manufacture Industry Real Sales SA 2006=100	BZCNSALS Index	SA Index	M	CNI	(4)	Wholesale sales (also referred to as wholesale trade) is a form of trade in which goods are purchased and stored in large quantities and sold to resellers, professional users or groups, but not to final consumers. This concept is based on the volume of goods sold.
79	Brazil Retail Sales Revenue Supermarkets Index NSA	BZRTSUIN Index	NSA Index	M	IBGE	(2)	Retail sales (also referred to as retail trade) tracks the resale of new and used goods to the general public, for personal or household consumption. This concept is based on the value of goods sold.
80	Brazil Amplified Retail Sales Volume MoM SA	BZRTAMPM Index	SA MoM%	M	IBGE	(0)	Idem
81	Brazil Auto Sales Total	BZASTOTL Index	NSA Units	M	Fenabrave	(7)	This concept tracks the number of motor vehicles newly registered with a government authority.
82	Brazil ANP Sales of Ethanol by State - National Total	BROSSTOT Index	Volume m3	M	ANP	(5)	This sector has the ethanol monthly sales by Brazilian state or region, which are calculated by the Agency of Petroleum (ANP).

83	Secovi Brazil Real Estate Units Average Sale Time Period	BZREPERD Index	NSA Units	M	Secovi	(1)	This concept tracks sales of newly constructed homes during the reference period.
84	Brazil FGV Consumer Confidence Index SA September 2005=100	BZFGCCSA Index	SA Index	M	FGV	(4)	The target sample size factors in the ranges of income across the different municipalities in Brazil to allow for an even distribution of responses. Target Audience: households, survey preferably to be taken by head of household who is >18yrs old. Sample Size: 2045 households. Date of Survey: first 2 weeks of the month.
85	Brazil CNI Industrial Confidence General Large Co(s)	BZICLCOS Index	NSA % Balance/Diffusion Index	M	CNI	(7)	Target Audience: Companies with 10 or more employees. Sample Size: approximately 2,000 companies. Date of Survey: first and second week of each month.
86	Brazil CNI Consumer Confidence Household Debt Situation 2001=100	BZCCCSDB Index	NSA Index	M	CNI	(7)	Consumer confidence tracks sentiment among households or consumers. The results are based on surveys conducted among a random sample of households.
Group IV: External Sector							
87	Brazil Exports USD FOB	BZEXTOT$ Index	NSA Value USD Millions	M	MDIC	(7)	Total Exports = Sugar + Cocoa + Coffee + Foot/Leath + Meat + Oil Deriv + Tobacco + Machine/Mechanical + Transportation + Electrical Products + Paper + Iron + Chemical Products + Metallurgical Products + Textile + Soybean + Orange Juice + Metals + Milk + Fruits + Fish + Furniture + Others.
88	Brazil Exports of Soybeans Including Grinded - US$ FOB	BZEXSYGD Index	NSA Value USD Millions	M	MDIC	(1)	
89	Brazil Exports By Sector Iron Ore USD FOB	BZEXIRO$ Index	NSA Value USD Millions	M	MDIC	(7)	
90	Brazil Exports of Crude Oil - US$ FOB	BZEXCROD Index	NSA Value USD Millions	M	MDIC	(3)	
91	Brazil Trade Balance FOB Imports NSA	BZTBIMPM Index	NSA Value USD Millions	M	MDIC	(7)	The total imports = Food + Beverage + Tobacco + Fuel + Fat + Oil + Chemical Product + Manufacture Products + Transportation + Other Manufacture Products + Others.
92	Brazil Imports of Capital Goods - US$ FOB	BZIMCAGD Index	NSA Value USD Millions	M	MDIC	(7)	
93	Brazil Imports of Nondurable Consumer Goods - US$ FOB	BZIMNCGD Index	NSA Value USD Millions	M	MDIC	(7)	

94	Brazil Imports of Fuels and Lubricants - US$ FOB	BZIMFULD Index	NSA Value USD Millions	M	MDIC	(7)	
95	Brazil Current Account Monthly	BZCACURR Index	NSA Value USD Millions	M	BCB	(7)	The current account is part of the balance of payments. It tracks all transactions, excluding financial transactions, that involve economic values and occur between residents of a country and nonresidents. Major components include trade in goods, trade in services, income and current transfers.The balance of payments is a record of a country's overall international transactions with the rest of the world (i.e. transactions between residents of a country and nonresidents). The balance of payments is divided into current, capital and financial accounts.
96	Brazil Current Account Balance on Goods and Services	BZBPBGS Index	NSA Value USD Millions	M	BCB	(7)	
97	Brazil Current Account Construction Credit	BZBPCSE Index	NSA Value USD Millions	M	BCB	(1)	
98	Brazil BOP Portfolio Investment Foreign in Fixed Income	BZBPPILD Index	NSA Value USD Millions	M	BCB	(0)	This series represents cross-border transactions between residents and non-residents in debt securities as recorded under the portfolio investment account of the Balance of Payments. Debt securities include bills.
99	Brazil BOP Portfolio Investment Net	BZBPPORT Index	NSA Value USD Millions	M	BCB	(0)	The financial & capital accounts are part of the balance of payments. The financial account tracks all of a country's external financial assets and liabilities. Major components of the financial account include direct investment, portfolio investment, other investments and reserve assets. The capital account tracks transfers of ownership across borders of fixed assets and acquisition or disposal of nonproduced, nonfinancial assets.The balance of payments is a record of a country's overall international transactions with the rest of the world (i.e. transactions between residents of a country and nonresidents). The balance of payments is divided into current, capital and financial accounts.

100	Brazil Financial Account Portfolio Investment Net Incurrence of Liabilities	BZBPFPL Index	NSA Value USD Millions	M	BCB	(0)	
101	Brazil Financial Account Direct Investment Balance	BZBPFD Index	NSA Value USD Millions	M	BCB	(0)	
102	Brazil BOP Financial Derivatives Assets	BZBPFDA Index	NSA Value USD Millions	M	BCB	(7)	
103	Brazil Fin Acct Portfolio Investment Equity and Inv Fund Shares Assets Bought	BZBPAPFD Index	NSA Value USD Millions	M	BCB	(0)	
104	Brazil Fin Acct Portfolio Investment Equity and Inv Fund Shares Assets Sold	BZBPAPFI Index	NSA Value USD Millions	M	BCB	(0)	
105	Brazil BOP Capital Account Net	BZBPCAPT Index	NSA Value USD Millions	M	BCB	(0)	
106	Brazil Primary Income	BZBPBIN1 Index	NSA Value USD Millions	M	BCB	(7)	
107	Brazil Primary Income Compensation of Employees	BZBPBD1 Index	NSA Value USD Millions	M	BCB	(1)	
108	Brazil Secondary Income	BZBPBIN2 Index	NSA Value USD Millions	M	BCB	(7)	
109	Brazil Secondary Income from General Government	BZBPBIN3 Index	NSA Value USD Millions	M	BCB	(0)	
110	Brazil BOP Financial Account Net	BZBPFINA Index	NSA Value USD Millions	M	BCB	(1)	
111	Brazil BOP Errors and Omissions	BZBPERRO Index	NSA Value USD Millions	M	BCB	(0)	
112	Brazil Fin Acct Portfolio Investment Acquisition of Financial Assets Credit	BZBPFPI Index	NSA Value USD Millions	M	BCB	(0)	
113	Brazil Financial Account Direct Investment Intercompany Assets	BZBPADF1 Index	NSA Value USD Millions	M	BCB	(1)	

114	Brazil Financial Account Portfolio Investment Debt Securities Long-Term Assets	BZBPAF5S Index	NSA Value USD Millions	M	BCB	(0)	
115	Brazil Financial Account Government Currency and Deposits Assets	BZBPAOF1 Index	NSA Value USD Millions	M	BCB	(0)	
116	Brazil Financial Account Loans Net Incurrence of Liabilities	BZBPLOF3 Index	NSA Value USD Millions	M	BCB	(0)	
117	Brazil Financial Account Central Banks Currency and Deposits Liabilities	BZBPLOFA Index	NSA Value USD Millions	M	BCB	(1)	
	Group V: Public Sector						
118	Brazil Public Entreprises Debt	BZPDCGOV Index	NSA Value BRL Billions	M	BCB	(4)	This concept includes all financial liabilities of a government (either central or central + local governments). These liabilities are typically in the form of government bills and bonds.
119	Brazil Total Net Debt in % of GDP	BZPDNDT% Index	NSA %	M	BCB	(1)	
120	Brazilian Cities Debt to Foreigners in % of GDP	BZPDECI% Index	NSA %	M	BCB	(1)	
121	Brazilian States Debt to Foreigners in % of GDP	BZPDEST% Index	NSA %	M	BCB	(1)	
122	Brazilian Federal Government Debt to Foreigners in % of GDP	BZPDEFE% Index	NSA %	M	BCB	(1)	
123	Brazil Fed Govt credit provided to Official Financial institutions in % of GDP	BZPDIOF$ Index	NSA %	M	BCB	(7)	
124	Brazil Net Consolidated Public Sector Debt	BZPDNDT Index	NSA Value BRL Billions	M	BCB	(4)	
125	Brazilian Federal Government Debt	BZPDFEDG Index	NSA Value BRL Billions	M	BCB	(4)	
126	Brazilian Federal Government Domestic Debt	BZPDIFEG Index	NSA Value BRL Billions	M	BCB	(4)	

127	Brazilian Enterprises Securitized Domestic Debt	BZPDISEC Index	NSA Value BRL Billions	M	BCB	(1)	
128	Brazilian Treasury Securitized Domestic Debt	BZPDITRE Index	NSA Value BRL Billions	M	BCB	(1)	
129	Brazilian States Debt	BZPDSTAG Index	NSA Value BRL Billions	M	BCB	(7)	
130	Brazil Public Primary Budget Result	BZPBPRDM Index	NSA Value BRL Billions	M	BCB	(1)	The government budget balance is the difference between government revenues and government expenditures. A budget surplus would mean revenues are higher than expenditures. A deficit means expenditures are greater than revenues.
131	Brazil Public Nominal Budget Result	BZPBNODM Index	NSA Value BRL Billions	M	BCB	(1)	
132	Brazil Public Nominal Interest Payments	BZPBRIDM Index	NSA Value BRL Billions	M	BCB	(1)	
133	Brazil Federal Govt and Central Bank Primary Balance	BZPBCGC Index	NSA Value BRL Billions	M	BCB	(1)	
134	Brazil Social Security Values Expenditures	BZPSEXPN Index	NSA Value BRL Millions	M	BCB	(7)	
135	Brazil Social Security Values Revenues	BZPSREVN Index	NSA Value BRL Millions	M	BCB	(7)	
136	Brazil Total Federal Revenue	BSRFTOFD Index	NSA Value BRL Millions	M	MinEcon	(7)	This concept tracks government revenues, including both tax and nontax receipts.
137	Brazil Importing Tax Income Nominal	BSRFIMPO Index	NSA Value BRL Millions	M	MinEcon	(7)	
138	Brazil Income Tax Collection Nominal	BSRFINCO Index	NSA Value BRL Millions	M	MinEcon	(7)	
139	Brazil CIDE Fuels Tax Income Nominal MoM	BSRFCIDM Index	NSA MoM%	M	MinEcon	(0)	
140	Brazil Social Security Contribution Tax Income Nominal MoM	BSRFCOFN Index	NSA MoM%	M	MinEcon	(0)	

141	Brazil Social Contribution over Net Profit Tax Income Nominal MoM	BSRFCSLM Index	NSA MoM%	M	MinEcon	(1)	
142	Brazil Manufactured Products Tax Income Nominal MoM	BSRFIPIM Index	NSA MoM%	M	MinEcon	(1)	
143	Brazil Rural Territory Tax Income Nominal MoM	BSRFITRM Index	NSA MoM%	M	MinEcon	(1)	
144	Brazil Other Federal Agencies Tax Income Nominal MoM	BSRFOTHM Index	NSA MoM%	M	MinEcon	(0)	
145	Brazil PIS PASEP Tax Income Nominal MoM	BSRFPISM Index	NSA MoM%	M	MinEcon	(0)	
146	Brazil Federal Income Agency Tax Collection Nominal MoM	BSRFSRFM Index	NSA MoM%	M	MinEcon	(0)	
147	Brazil Central Government Net Revenue	BZBGREVE Index	NSA Value BRL Billions	M	STN	(7)	
148	Brazil Central Government Revenue from the National Treasury	BZBGTREA Index	NSA Value BRL Billions	M	STN	(7)	
149	Brazil Central Government Total Expenditures	BZBGEXPN Index	NSA Value BRL Billions	M	STN	(7)	This concept tracks government expenditures.
150	Brazil Public Net Fiscal Debt	BZDPNFD Index	NSA Value BRL Billions	M	BCB	(7)	
151	Brazil Public Net Fiscal Debt % of GDP	BZDPNFD% Index	NSA %	M	BCB	(2)	
152	Brazil National Treasury Revenue from Industrialized Products Tax	BZBGINDP Index	NSA Value BRL Billions	M	STN	(7)	
153	Brazil National Treasury Other Current Expenditures	BZBGOEX1 Index	NSA Value BRL Billions	M	STN	(7)	

Group VI: Labor, Employment and Income

154	IMF Brazil Labor Force	2236685 Index	NSA Value Thousands	M	IMF	(7)	
155	IMF Brazil Employment	2236686 Index	NSA Value Thousands	M	IMF	(7)	

156	IMF Brazil Unemployment Rate	2236689 Index	NSA %	M	IMF	(2)	
157	Brazil Real Minimum Wage	BRMWRL Index	NSA Value BRL	M	IPEA	(7)	
158	CNI Brazil Manufacture Industry Real Wages SA 2006=100	BZCNWAGS Index	SA Index	M	CNI	(7)	This concept generally tracks total remuneration (in cash or in kind) paid to employees in return for work done (or paid leave).
159	CNI Brazil Manufacture Industry Employment SA 2006=100	BZCNEMPS Index	SA Index	M	CNI	(7)	This concept measures the number of employed people as tracked by a household labor force survey.
160	CNI Brazil Manufacture Industry Working Hours SA 2006=100	BZCNHOUS Index	SA Index	M	CNI	(7)	This concept measures the aggregate number of hours worked during the reference period.
161	Brazil Formal Employment Index Total SA 2001=100	BFOETTSA Index	SA Index	M	BCB	(7)	
162	Brazil Formal Employment Index Commerce SA 2001=100	BFOECOSA Index	SA Index	M	BCB	(7)	
163	Brazil Formal Employment Index Services SA 2001=100	BFOESVSA Index	SA Index	M	BCB	(7)	

Group VII: Expectations

164	Brazil CPI IGPM Weekly Preview	BZW IGPM INDEX	NSA MoM%	W	FGV	(0)	
165	Brazil FGV Consumer General Price Index Market First 10 Day Period Preview	BIGPIGP1 Index	NSA %	M	FGV	(0)	
166	Brazil FGV Consumer General Price Index Market Second 10 Day Period Preview	BIGPIGP2 Index	NSA %	M	FGV	(0)	
167	Brazil CPI IPCA Median Market Expectation Next 12 Months YoY	BRFCP1 Index	NSA YoY%	D	BCB	(4)	FOCUS Research
168	Brazil CPI IPCA Coefficient of Variation Market Expectation Next 12 Months YoY	-	NSA YoY%	D	BCB	(0)	FOCUS Research
169	Brazil CPI IGPM Median Market Expectation Next 12 Months YoY	BRFCPL Index	NSA YoY%	D	BCB	(0)	FOCUS Research

170	Brazil CPI IPCA-15 Median Market Expectation Next 12 Months YoY	BRFCP6 Index	NSA YoY%	D	BCB	(4)	FOCUS Research
171	Brazil CPI IPA-DI Smooth Median Market Expectation Next 12 Months YoY	BRFCPI12 Index	NSA YoY%	D	BCB	(0)	FOCUS Research
172	Brazil Business Expectations Future Delivery NSA MoM	BZBXNFDM Index	NSA MoM%	M	FECAP	(0)	This concept tracks business sentiment within the retail trade industry. The results are based on a survey conducted among a representative sample of businesses in the retail sector.
173	JPMorgan Forecast History Index Brazil	JFHIBR Index	SA %	W	JP Morgan	(1)	The J.P.Morgan Forecast History Index (JFHI) is a rolling average of projected real GDP growth annualized over the t-1, t, t+1, and t+2 quarters, where t is the quarter in which the forecast is made. For example, the JHFI for January 23, 2015 is the average of projected annualized sequential real GPD growth (%q/q, saar) for 4Q14, 1Q15, 2Q15, 3Q15. Note, the index will jump when the date rolls into a new quarter. All forecasts come from J.P. Morgan economic research. Indexes are available for each country in the J.P. Morgan economic research universe as well as for global and regional aggregates. For further documentation, see 'Know thyself: Evaluating and using J.P. Morgan economic forecasts', Joseph Lupton et al, J.P. Morgan, Global Issues, 23 September 2014.

Source: Bloomberg. Access in December 2019 and January 2020.

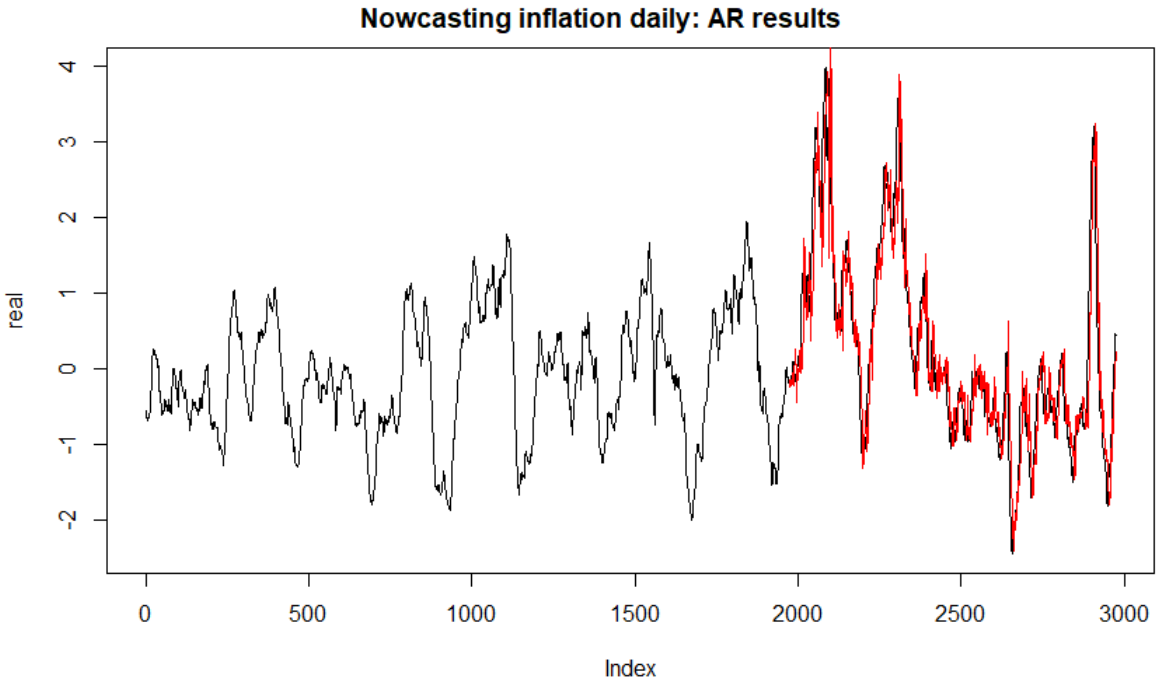
* NSA: Not seasonally adjusted (NSA) data do not contain any adjustments for seasonal or calendar effects (i.e. no effort has been made to remove the effects of seasonal patterns or calendar variations from the underlying data). NSA data are sometimes referred to as raw or original data. SA: Seasonally adjusted (SA) data have been adjusted for the effects of seasonal patterns. These seasonal effects can include things such as increased retail spending around Christmas or decreased construction activity during winter months in colder climates. Seasonal adjustment often also removes the effects of calendar variations (e.g. differing number of working days per month). Analysts generally prefer to use seasonally adjusted data, as it is easier to observe the underlying trend in the data series. MoM%: Data with the transformation MoM% are period-to-period growth rates on monthly data. They can be calculated by taking the underlying data of the reference month divided by the previous month as per the following formula: $(\frac{M_t}{M_{t-1}} - 1) \times 100$, where M_t is underlying level in reference month, t. YoY% is analogous yearly concept. "%" means percentage change over the previous period. "% Balance/Diffusion Index": The "% Balance/Diffusion Index" transformation is primarily used to summarize answers to qualitative, multiple choice questions from business or consumer surveys (i.e. questions such as What are your hiring intentions over the next 3 months? or How do you

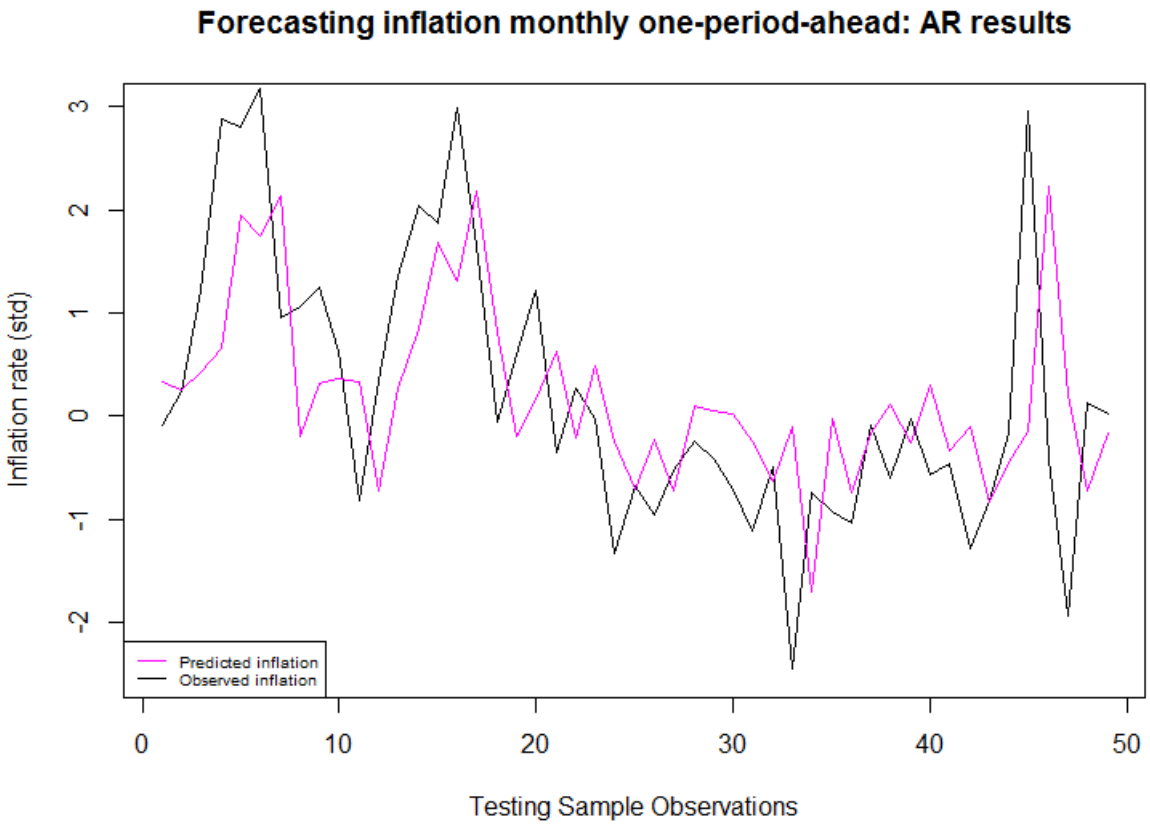
expect prices to move over the next 12 months?, etc.). A % Balance (also referred to as net balance) is calculated by taking the difference between the percentage of respondents giving favorable and unfavorable responses (e.g. 30% of respondents expect an increase in prices over the next 12 months, 20% expect a decrease in prices, therefore the percent balance is 10. No change responses are typically disregarded). A Diffusion Index is normally calculated by taking the percentage of favorable responses plus half the percentage of no change responses (e.g. 30% of respondents expect prices to increase over the next 12 months, 50% of respondents expect no change, therefore the diffusion index would be 55).

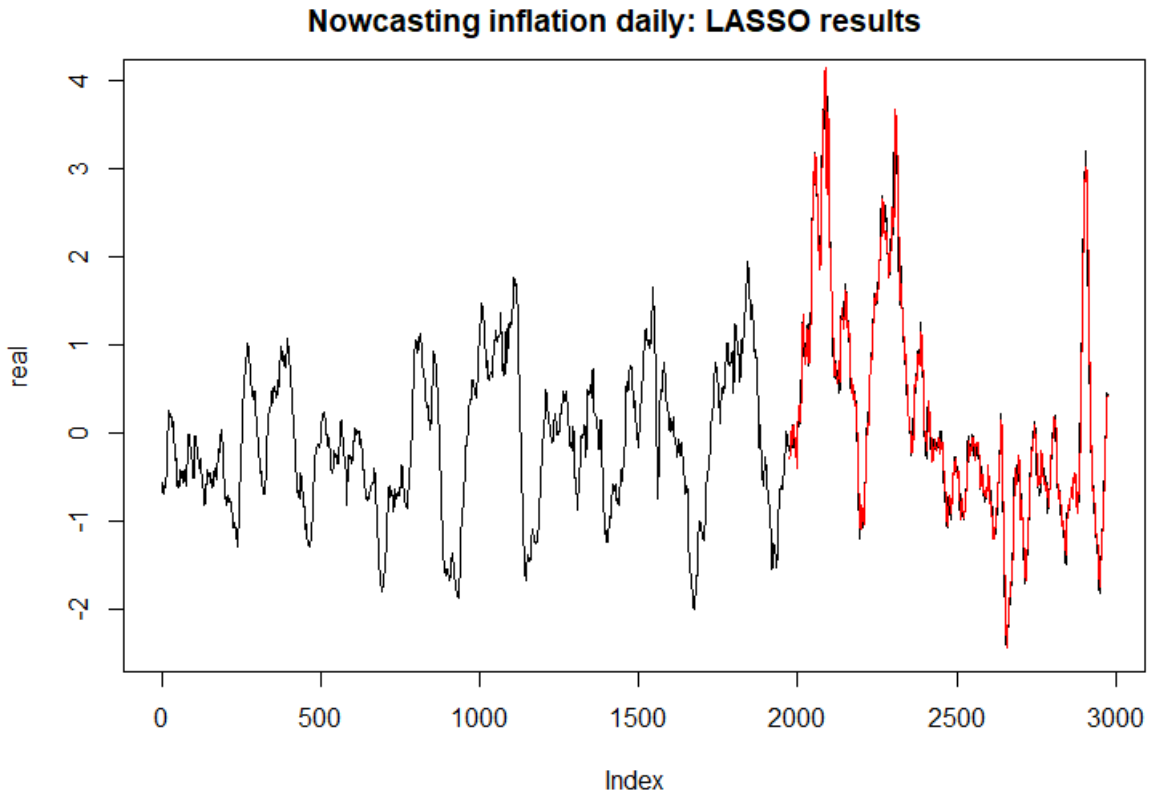
** Frequency of the data: (D) Daily; (W) Weekly; (M) Monthly.

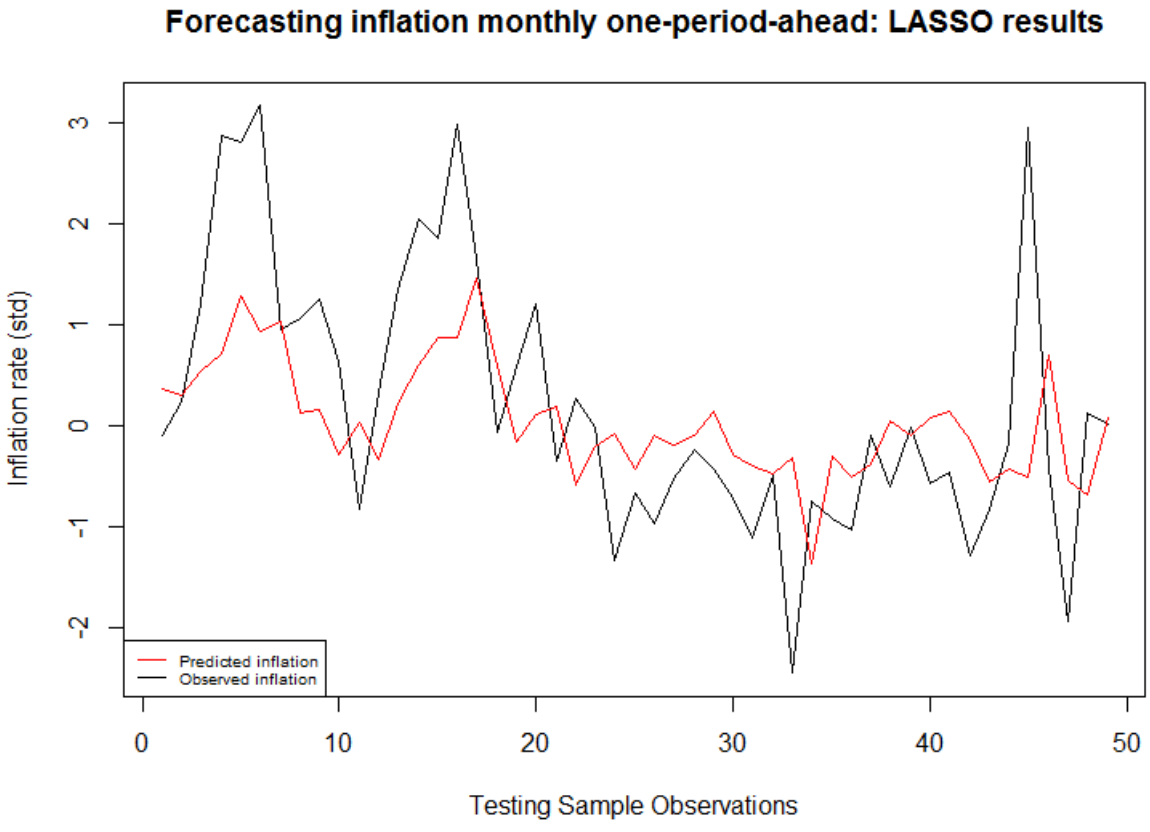
*** Transformation applied to the series to make it stationary. To test stationarity, the augmented Dickey–Fuller test (ADF) is used. The possible transformations are: (0) no transformation; (1) Δx_t; (2) $\Delta^2 x_t$; (3) $\log x_t$; (4) $\Delta \log x_t$; (5) $\Delta^2 \log x_t$; (6) $\Delta^3 \log x_t$; (7) $\Delta\Big(\frac{x_t}{x_{t-1}} - 1\Big)$.

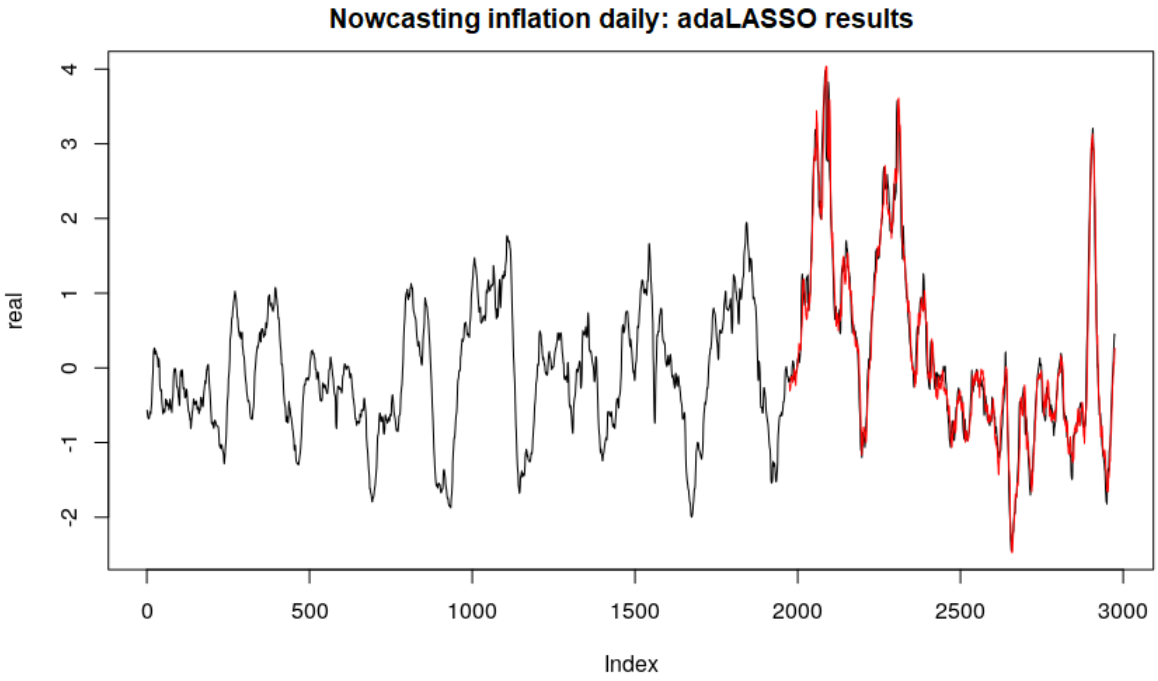
**** The four inflation Monitors do not have Bloomberg tickers, so these names – mon_ipca, mon_ipcap, mon_ipca15, mon_ipca15p – were created here *ad hoc*.

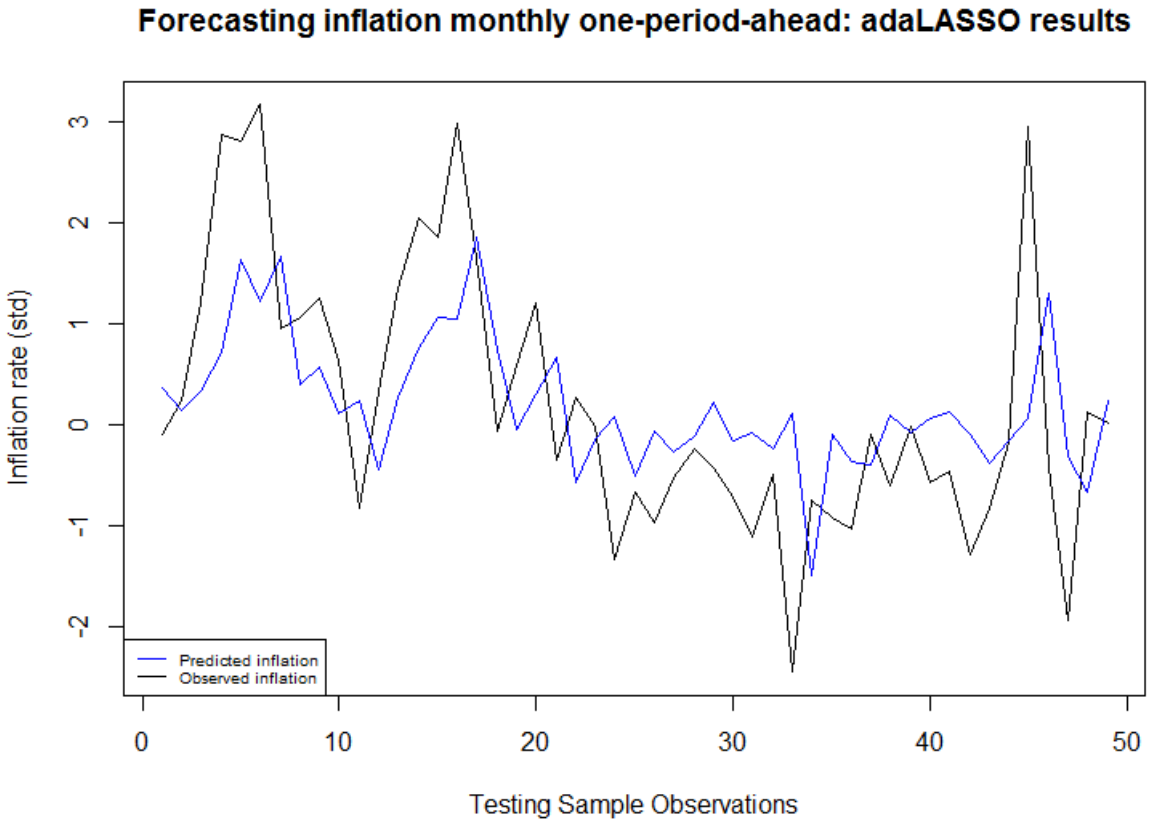

Nowcasting inflation daily: AR results
real
Index
0
500
1000
1500
2000
2500
3000
-2
-1
0
1
2
3
4

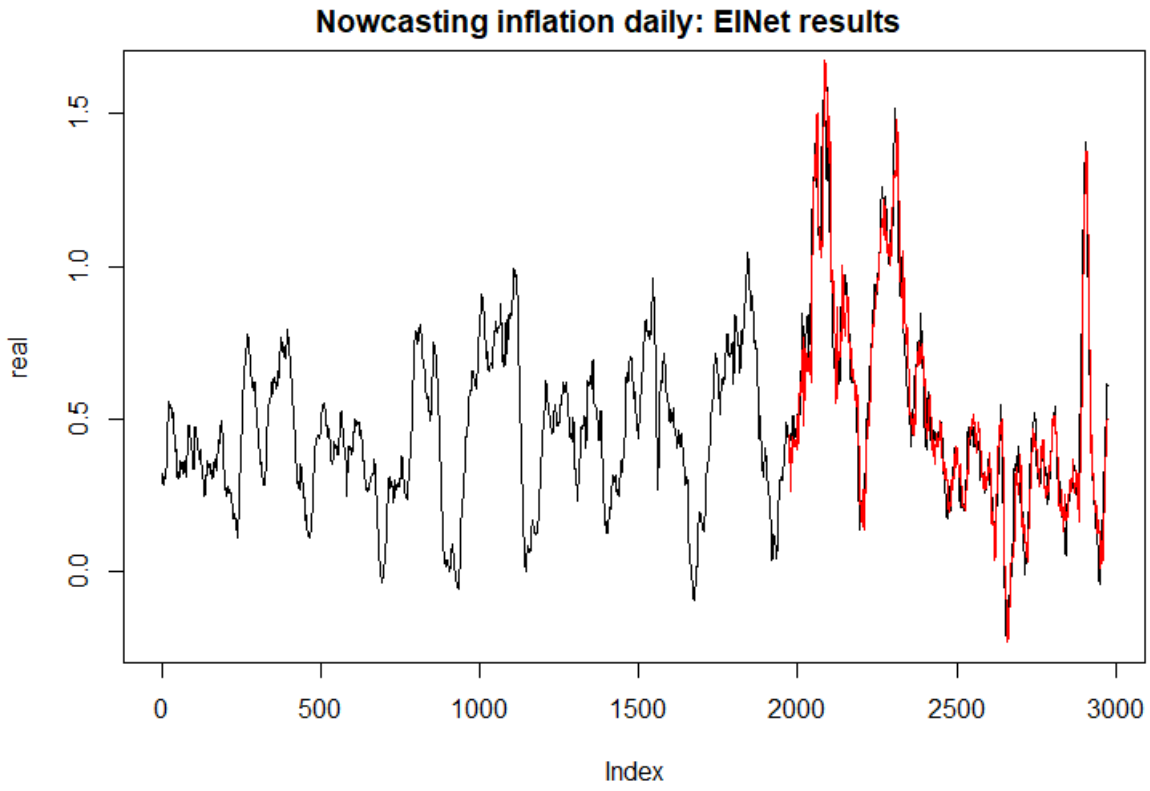

Forecasting inflation monthly one-period-ahead: AR results
Inflation rate (std)
Testing Sample Observations
Predicted inflation
Observed inflation
0
10
20
30
40
50
-2
-1
0
1
2
3

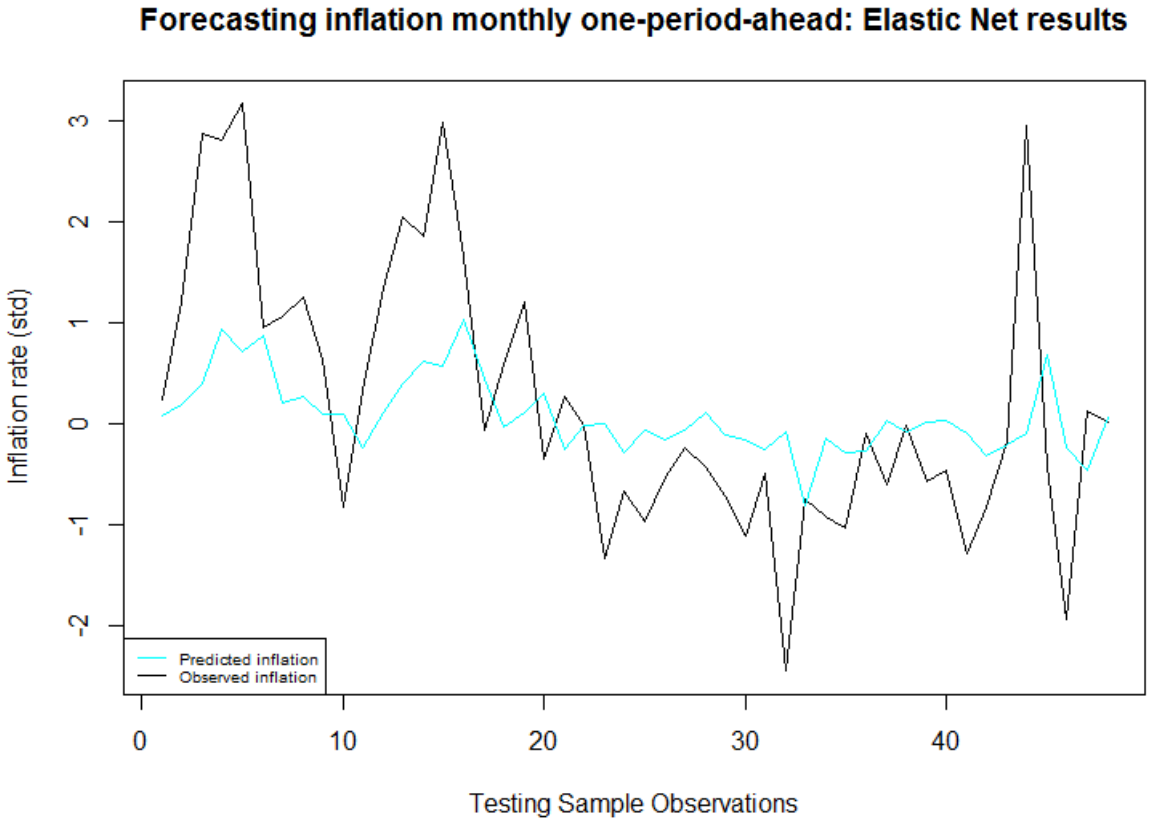

Nowcasting inflation daily: LASSO results
real
Index
-2
-1
0
1
2
3
4
0
500
1000
1500
2000
2500
3000

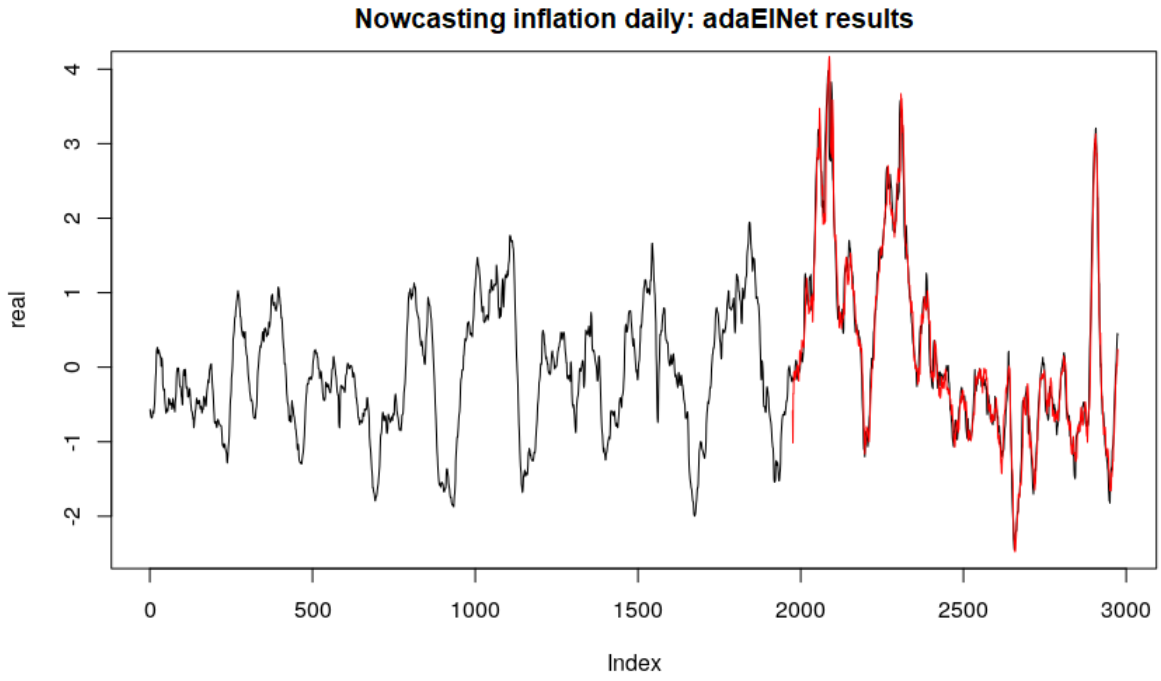

Forecasting inflation monthly one-period-ahead: LASSO results
Inflation rate (std)
Testing Sample Observations
Predicted inflation
Observed inflation
-2
-1
0
1
2
3
0
10
20
30
40
50

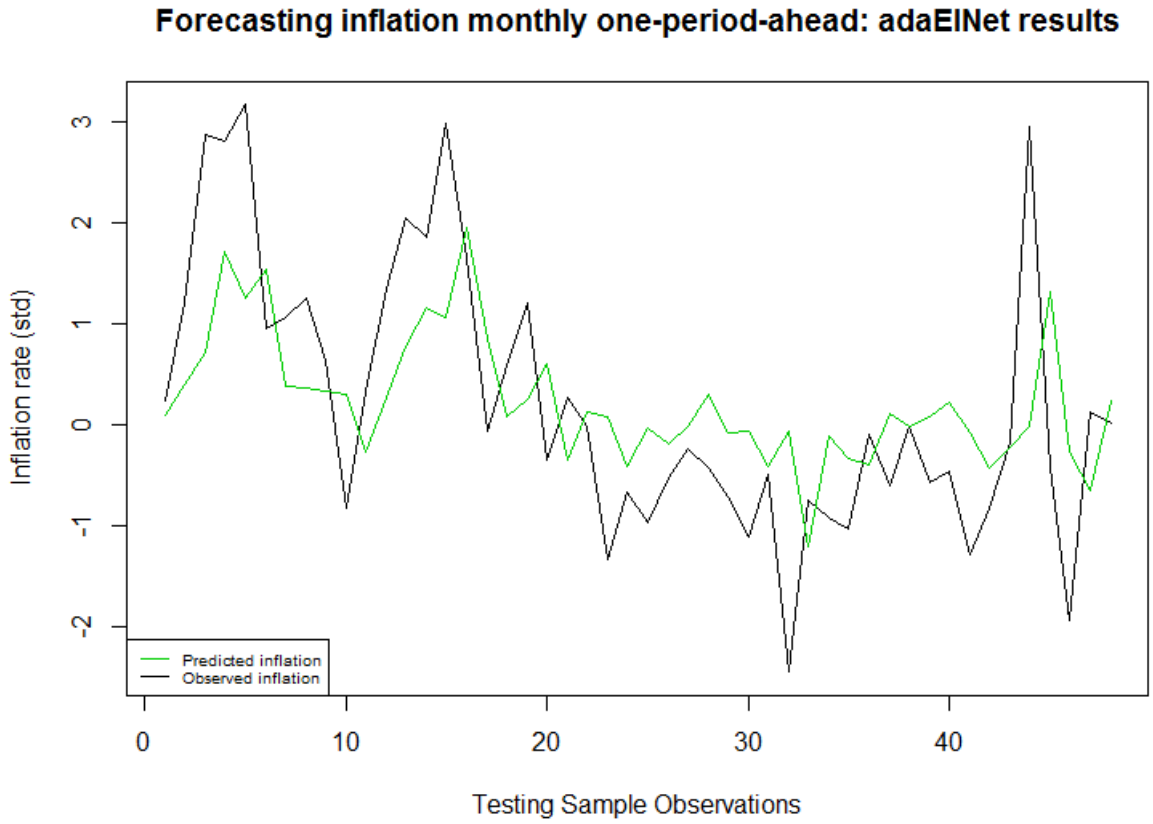

Nowcasting inflation daily: adaLASSO results
real
Index
-2
-1
0
1
2
3
4
0
500
1000
1500
2000
2500
3000

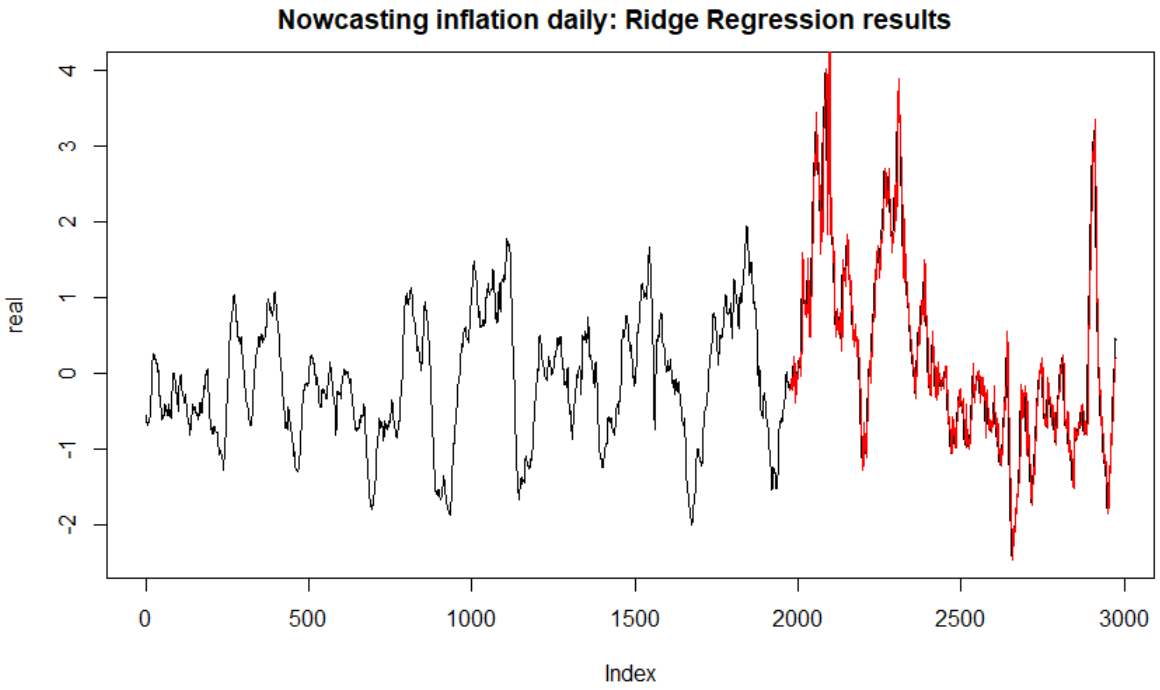

Forecasting inflation monthly one-period-ahead: adaLASSO results
Inflation rate (std)
Testing Sample Observations
Predicted inflation
Observed inflation
-2
-1
0
1
2
3
0
10
20
30
40
50

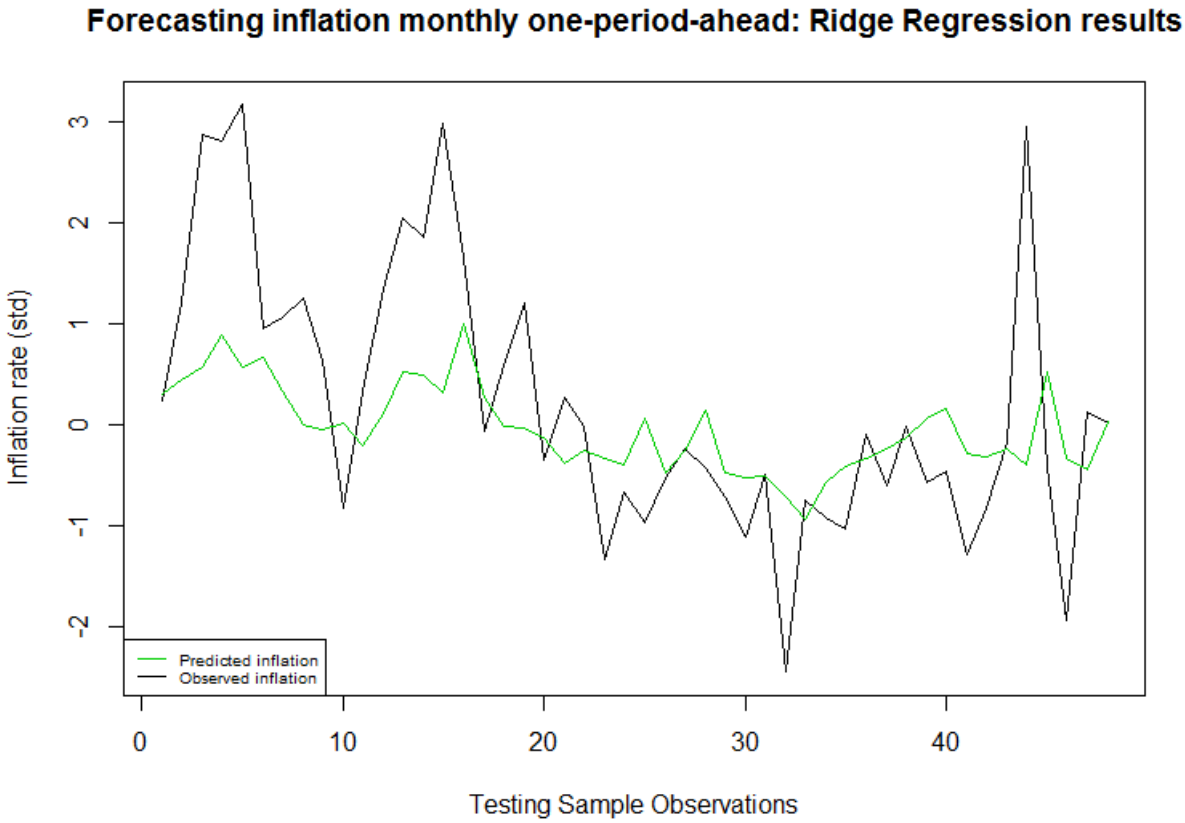

Nowcasting inflation daily: ElNet results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index

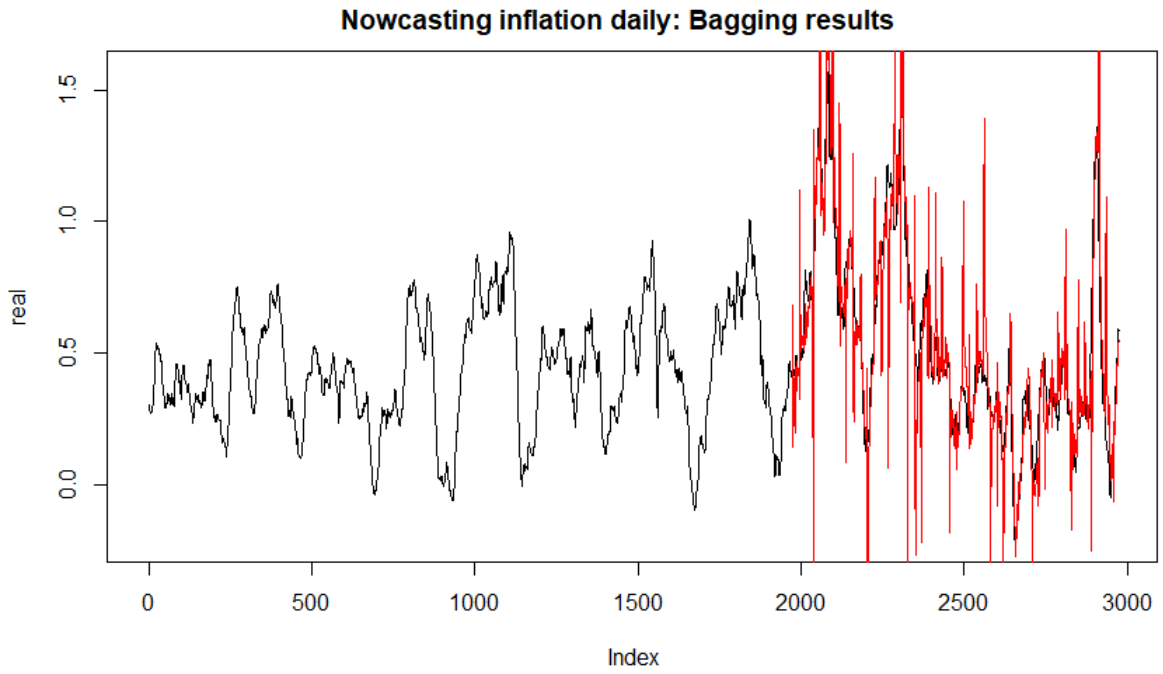

Forecasting inflation monthly one-period-ahead: Elastic Net results
Inflation rate (std)
-2
-1
0
1
2
3
Predicted inflation
Observed inflation
0
10
20
30
40
Testing Sample Observations

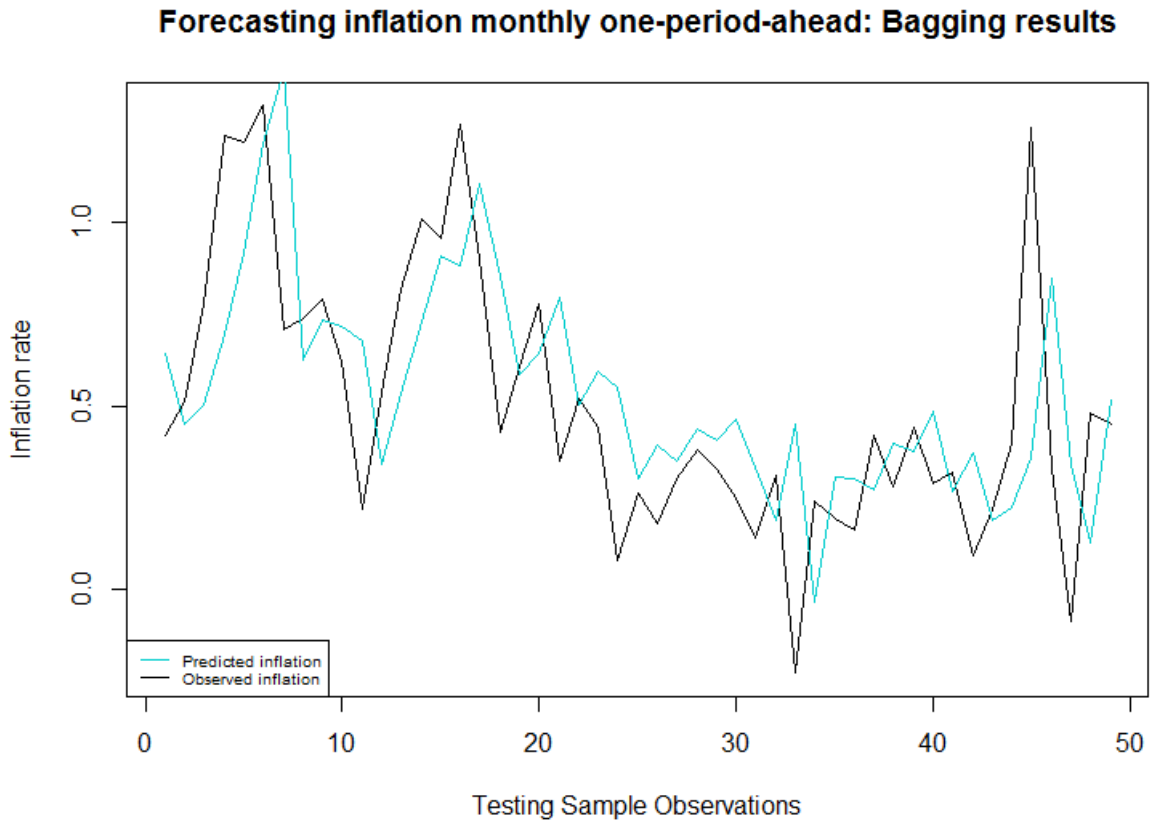

Nowcasting inflation daily: adaElNet results
real
Index
-2
-1
0
1
2
3
4
0
500
1000
1500
2000
2500
3000

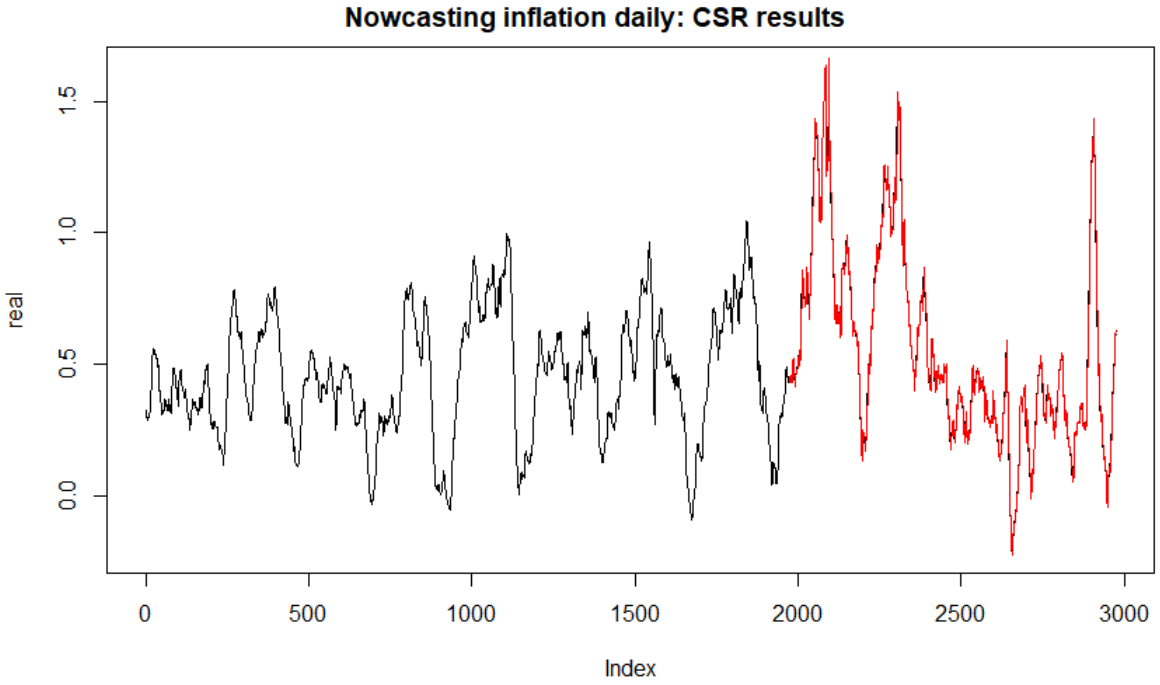

Forecasting inflation monthly one-period-ahead: adaElNet results
Inflation rate (std)
Testing Sample Observations
Predicted inflation
Observed inflation
-2
-1
0
1
2
3
0
10
20
30
40

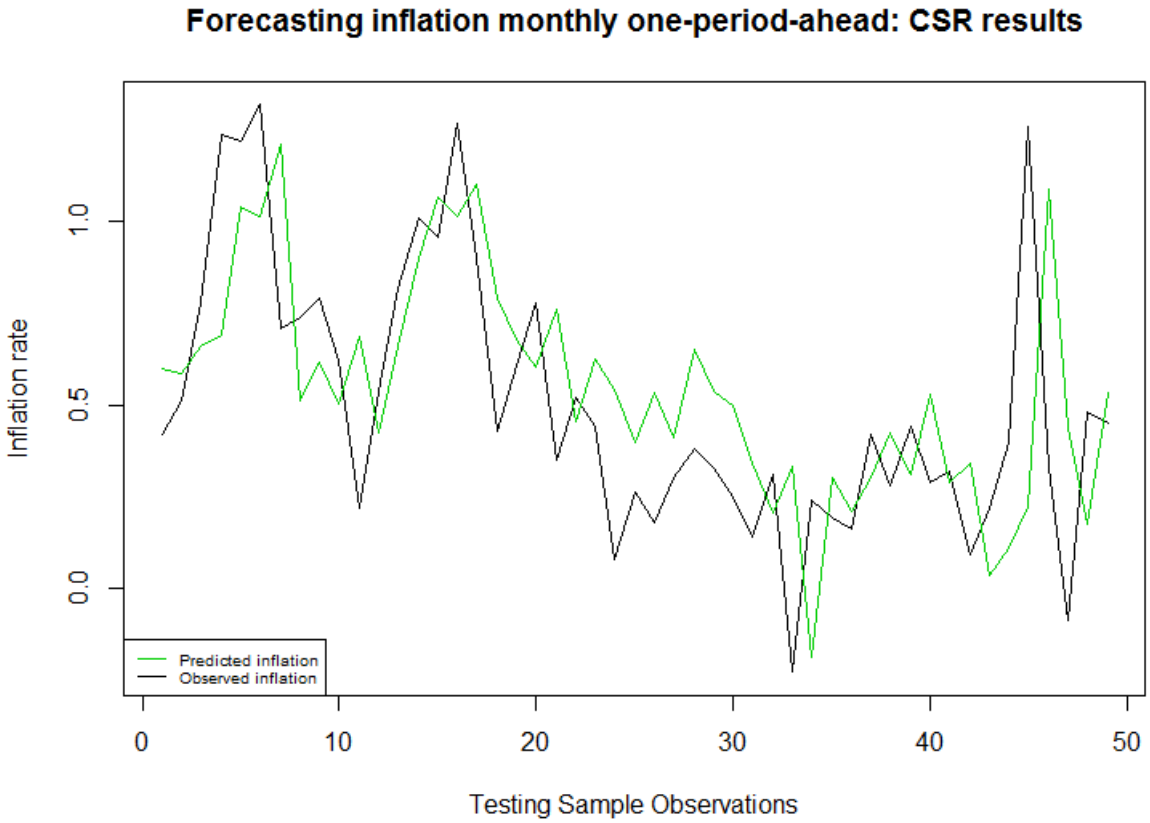

Nowcasting inflation daily: Ridge Regression results
real
Index
-2
-1
0
1
2
3
4
0
500
1000
1500
2000
2500
3000

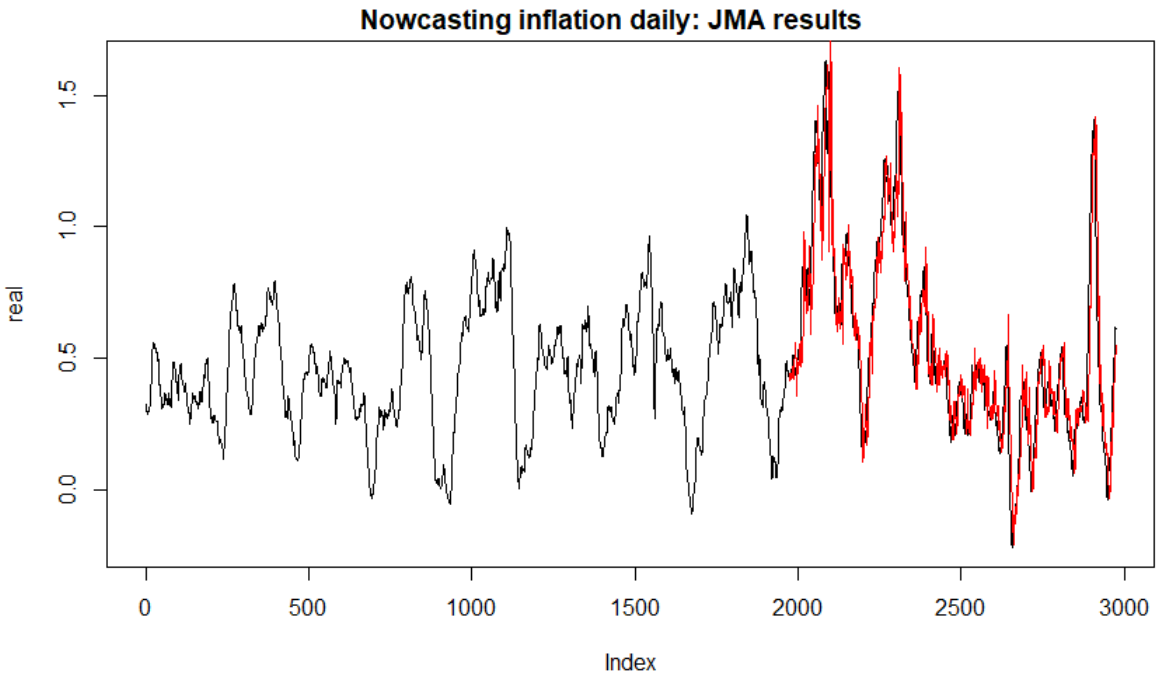

Forecasting inflation monthly one-period-ahead: Ridge Regression results
Inflation rate (std)
Testing Sample Observations
Predicted inflation
Observed inflation
-2
-1
0
1
2
3
0
10
20
30
40

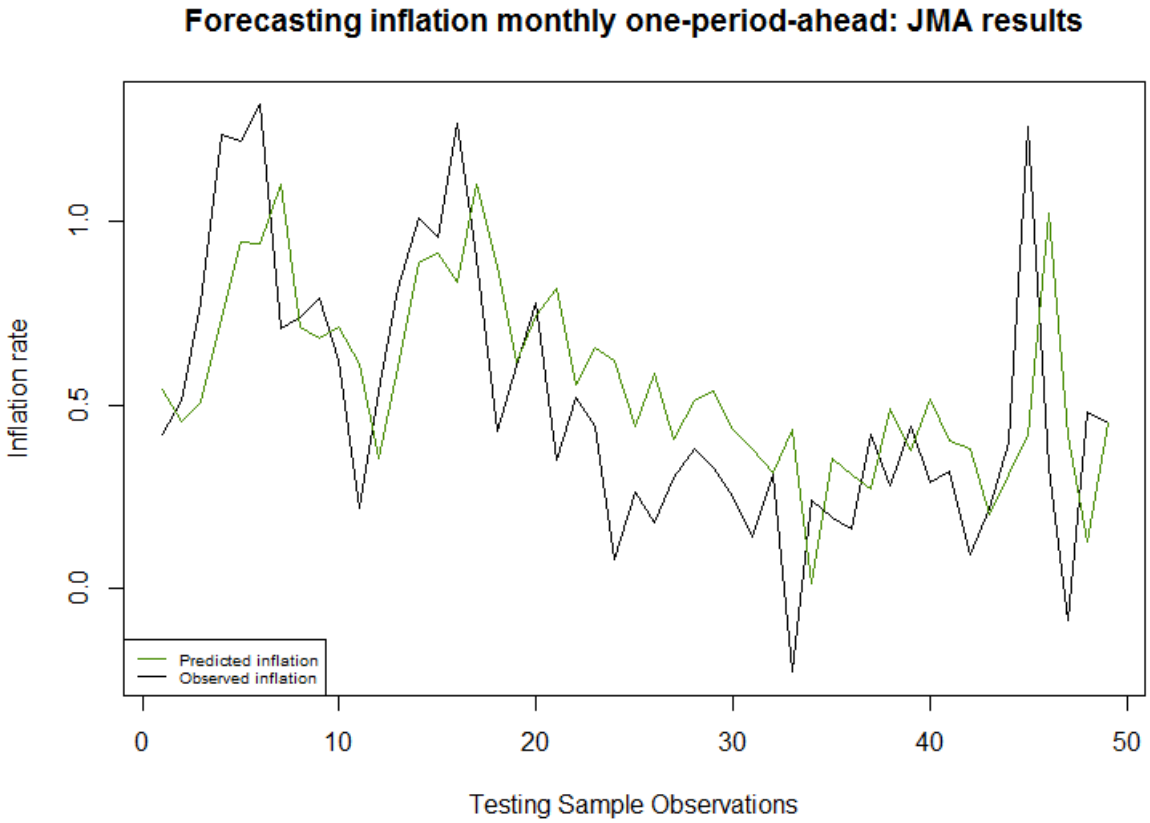

Nowcasting inflation daily: Bagging results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index

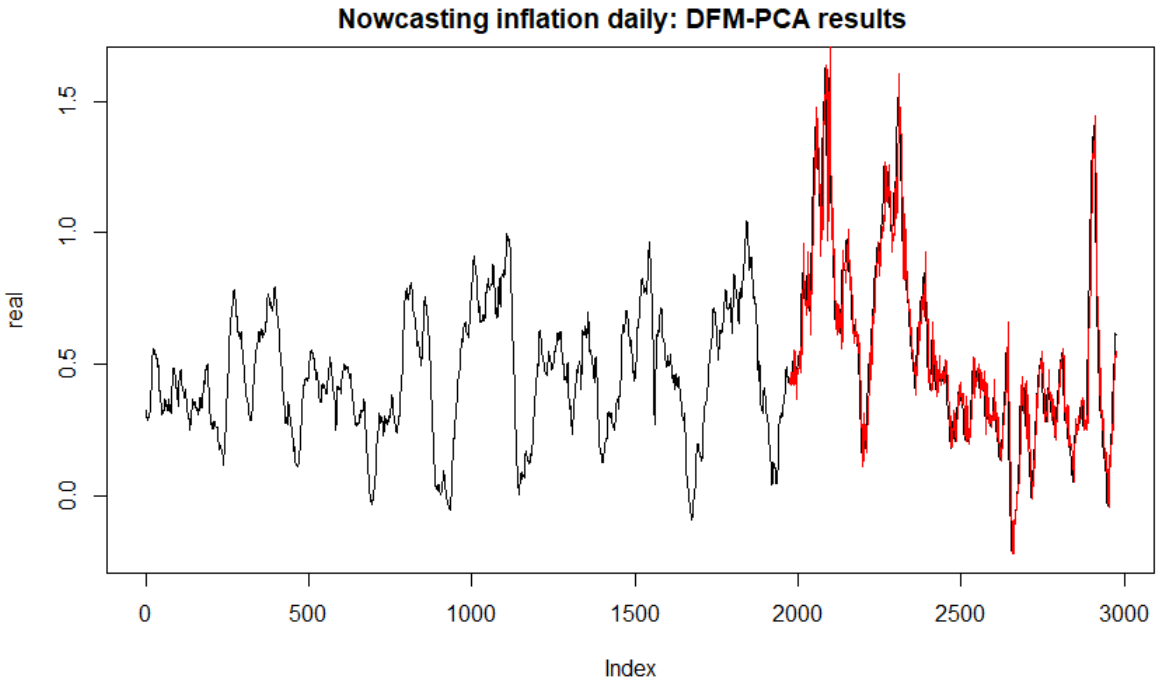

Forecasting inflation monthly one-period-ahead: Bagging results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations

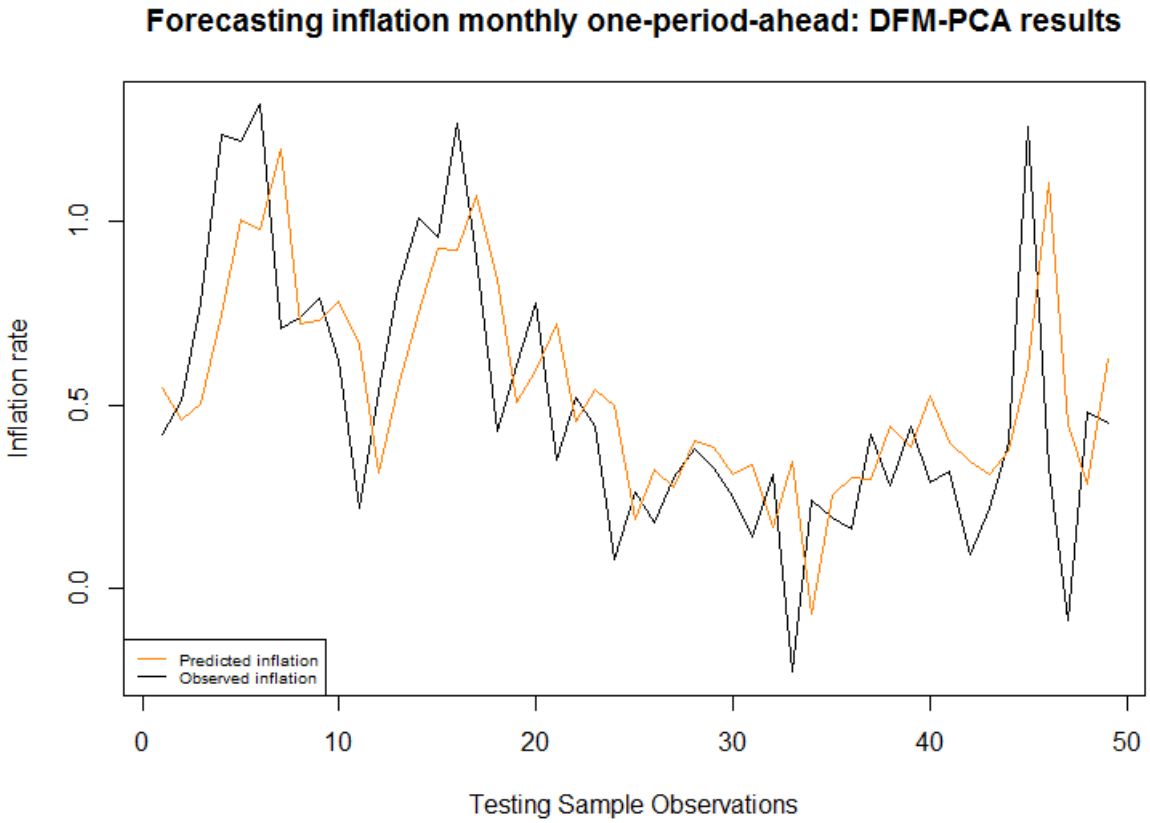

Nowcasting inflation daily: CSR results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index

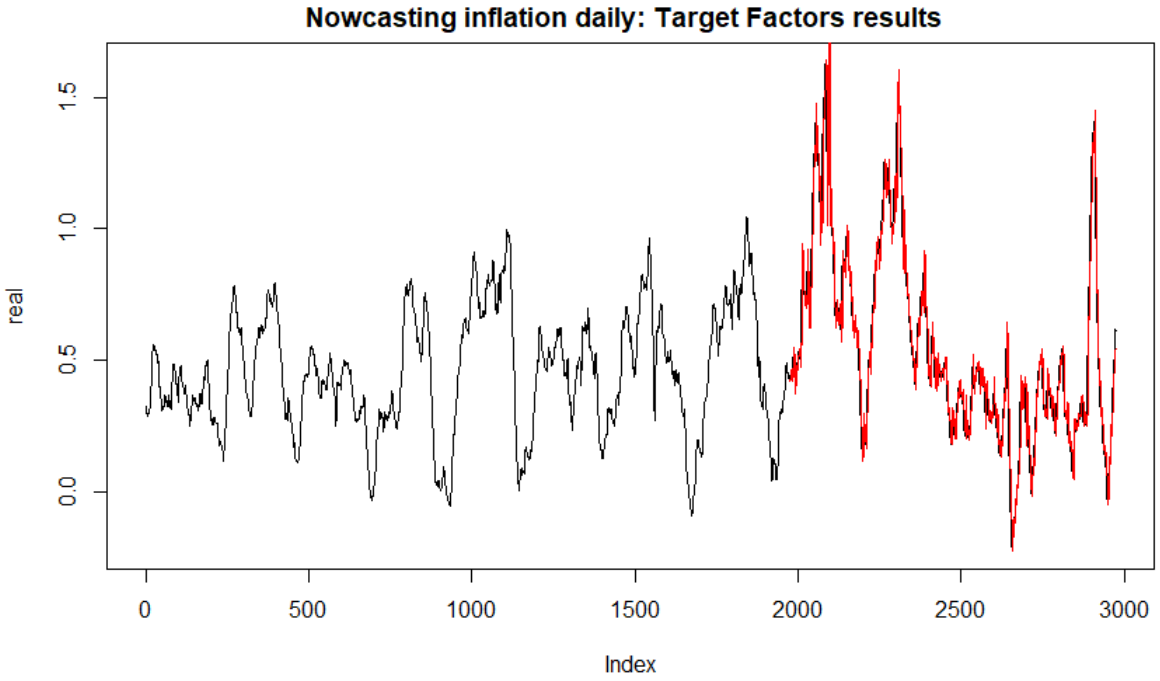

Forecasting inflation monthly one-period-ahead: CSR results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations

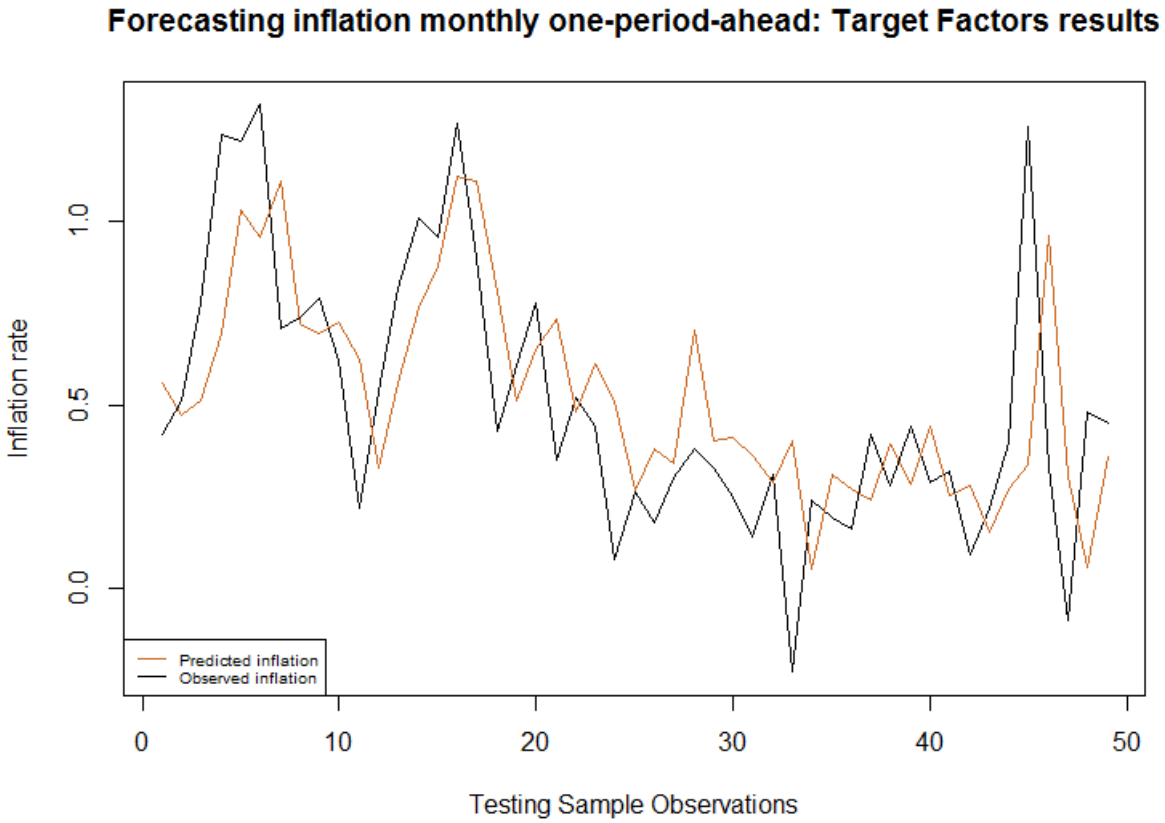

Nowcasting inflation daily: JMA results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index

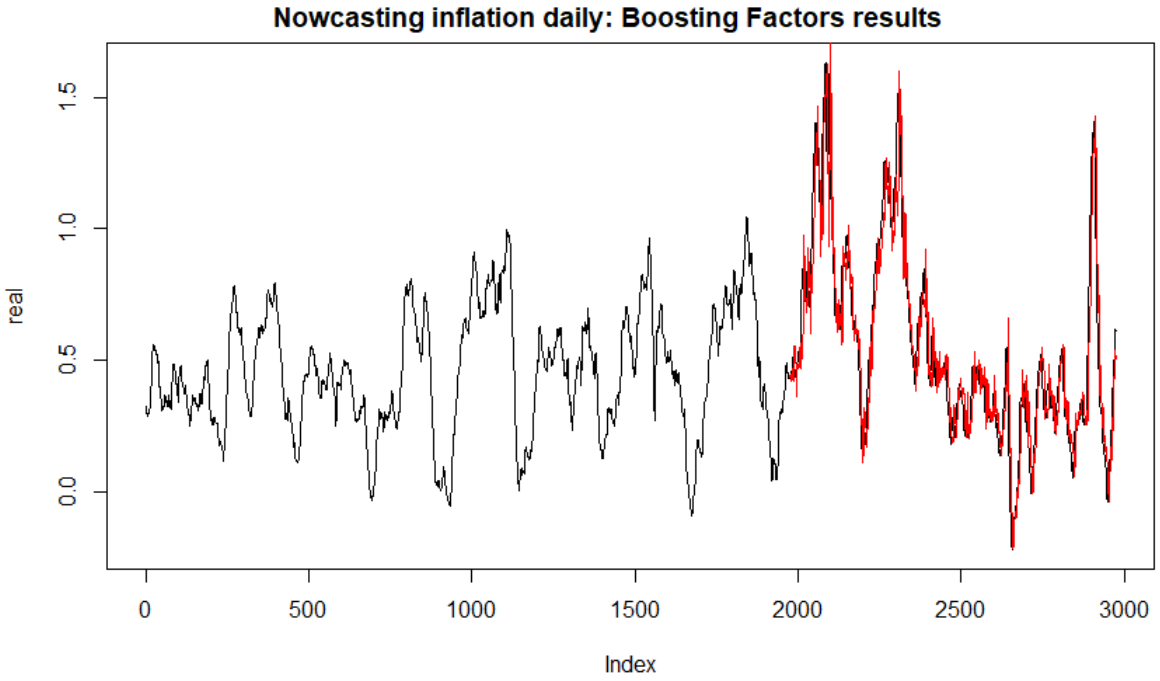

Forecasting inflation monthly one-period-ahead: JMA results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations

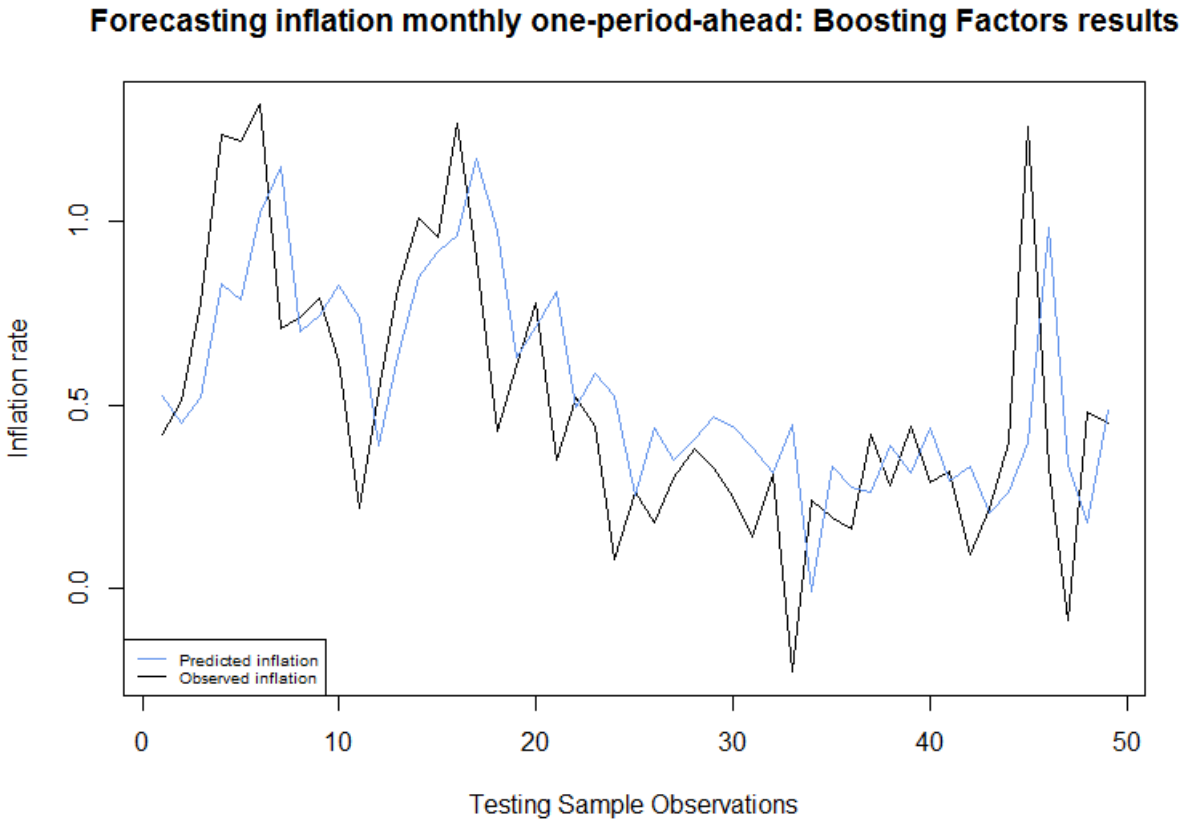

Nowcasting inflation daily: DFM-PCA results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index

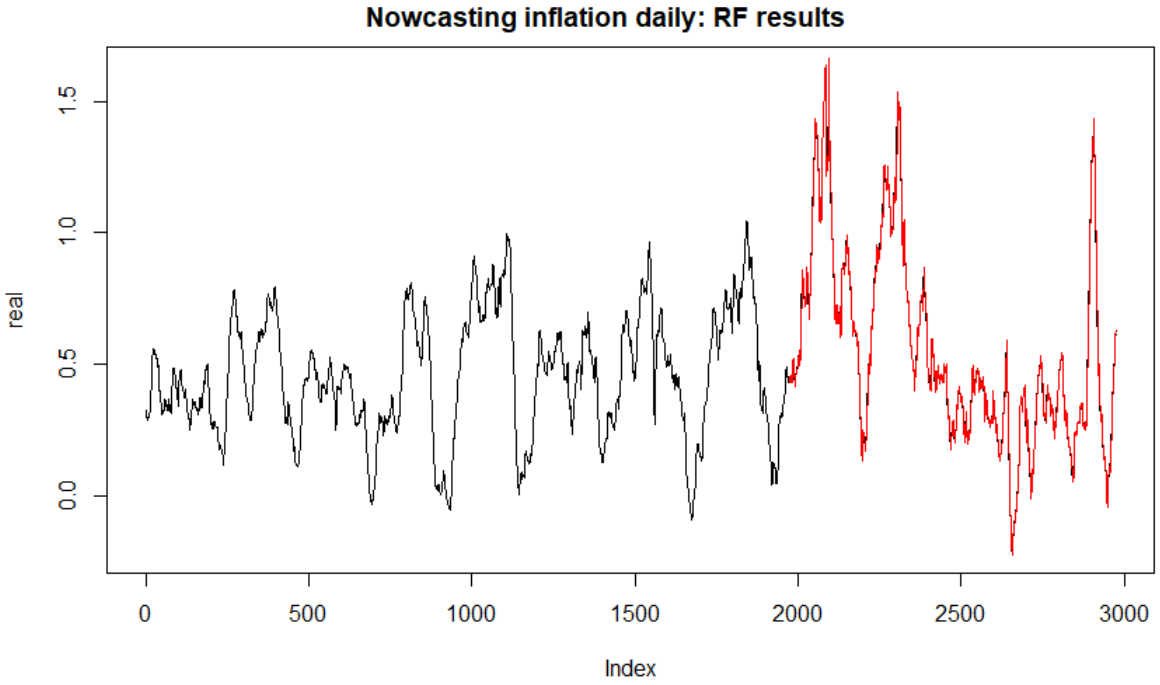

Forecasting inflation monthly one-period-ahead: DFM-PCA results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations

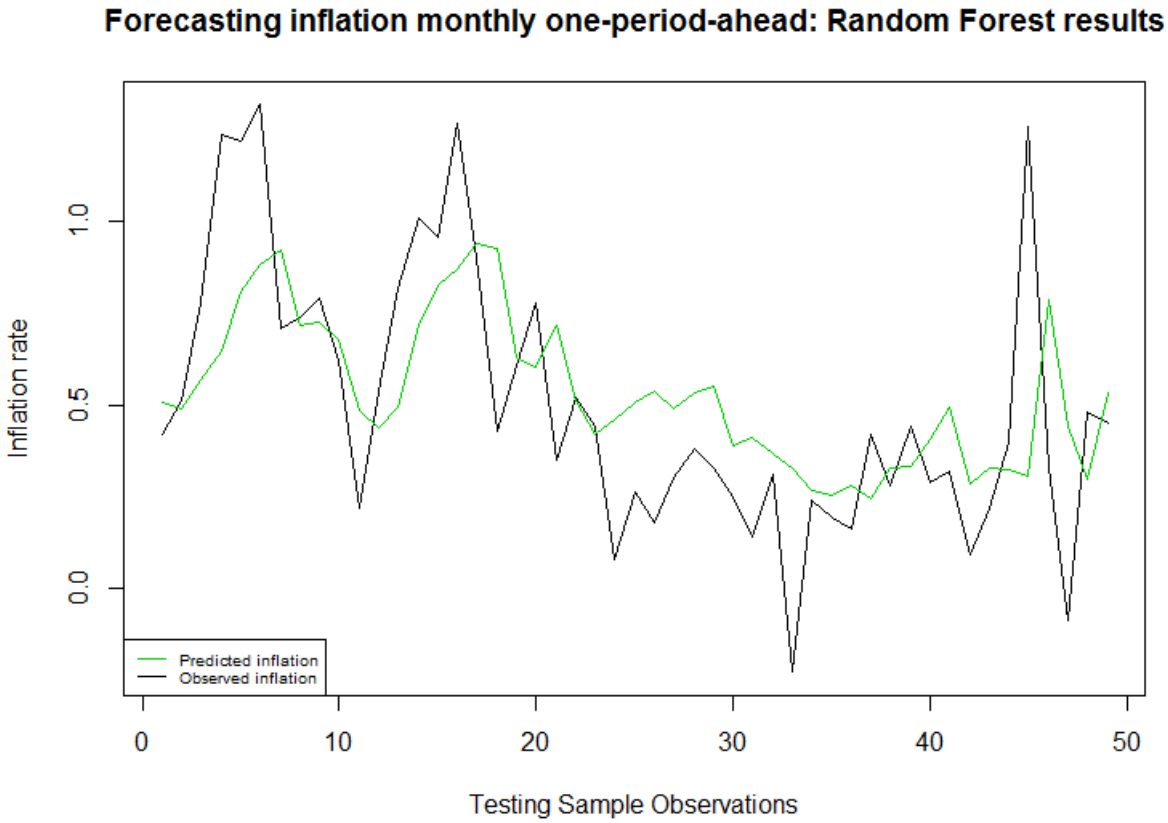

Nowcasting inflation daily: Target Factors results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index

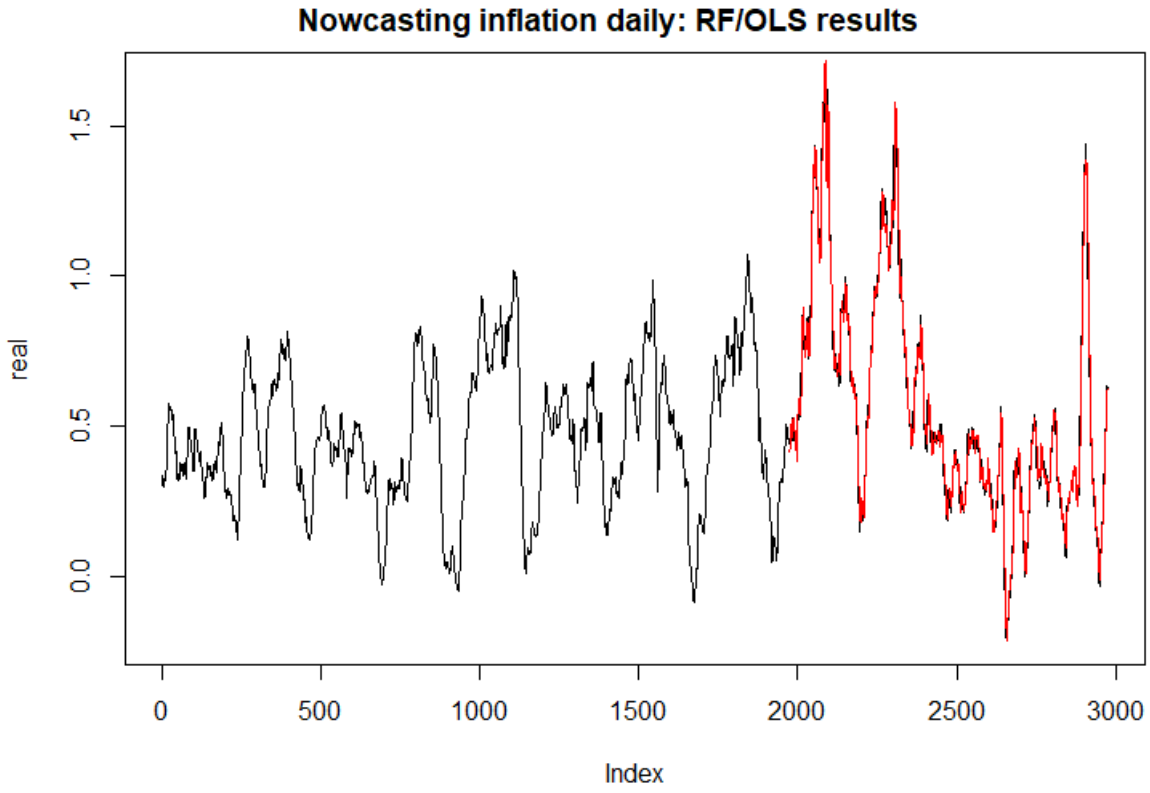

Forecasting inflation monthly one-period-ahead: Target Factors results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations

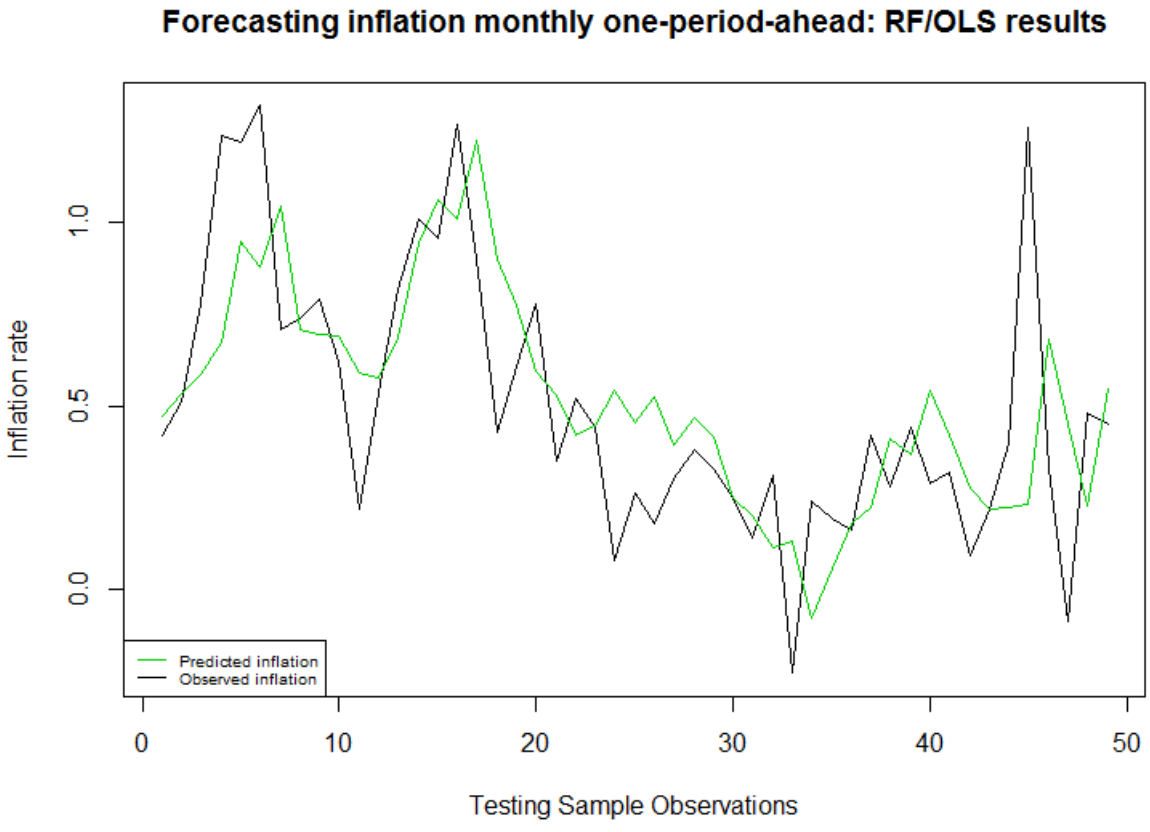

Nowcasting inflation daily: Boosting Factors results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index

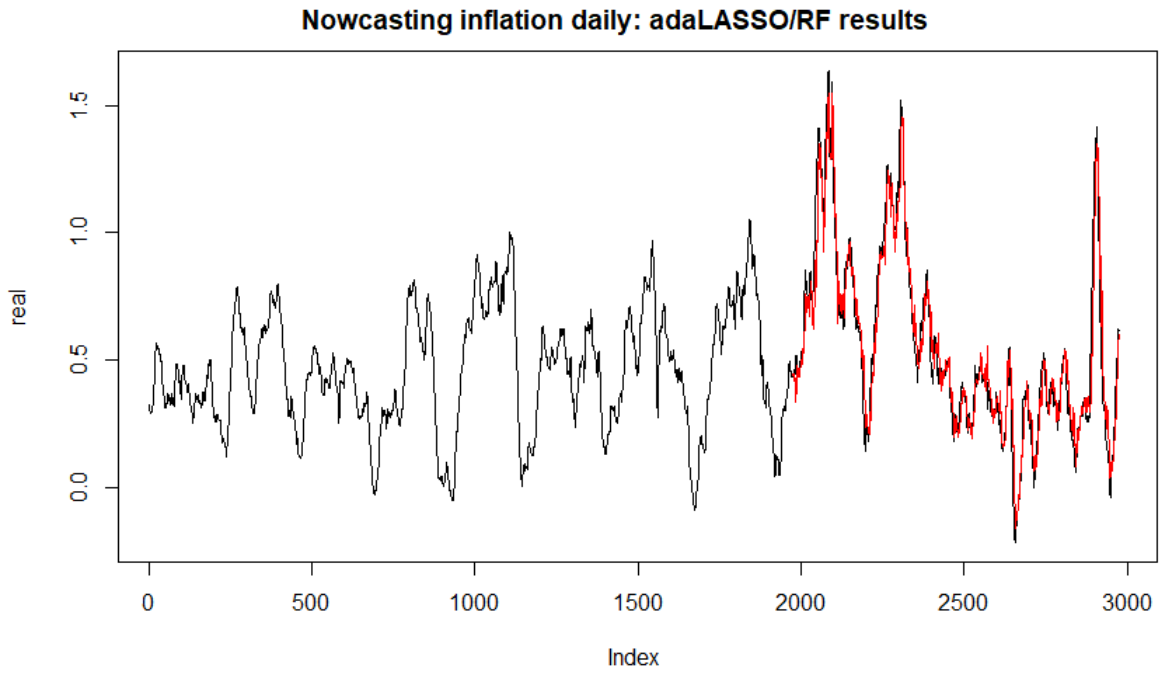

Forecasting inflation monthly one-period-ahead: Boosting Factors results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations

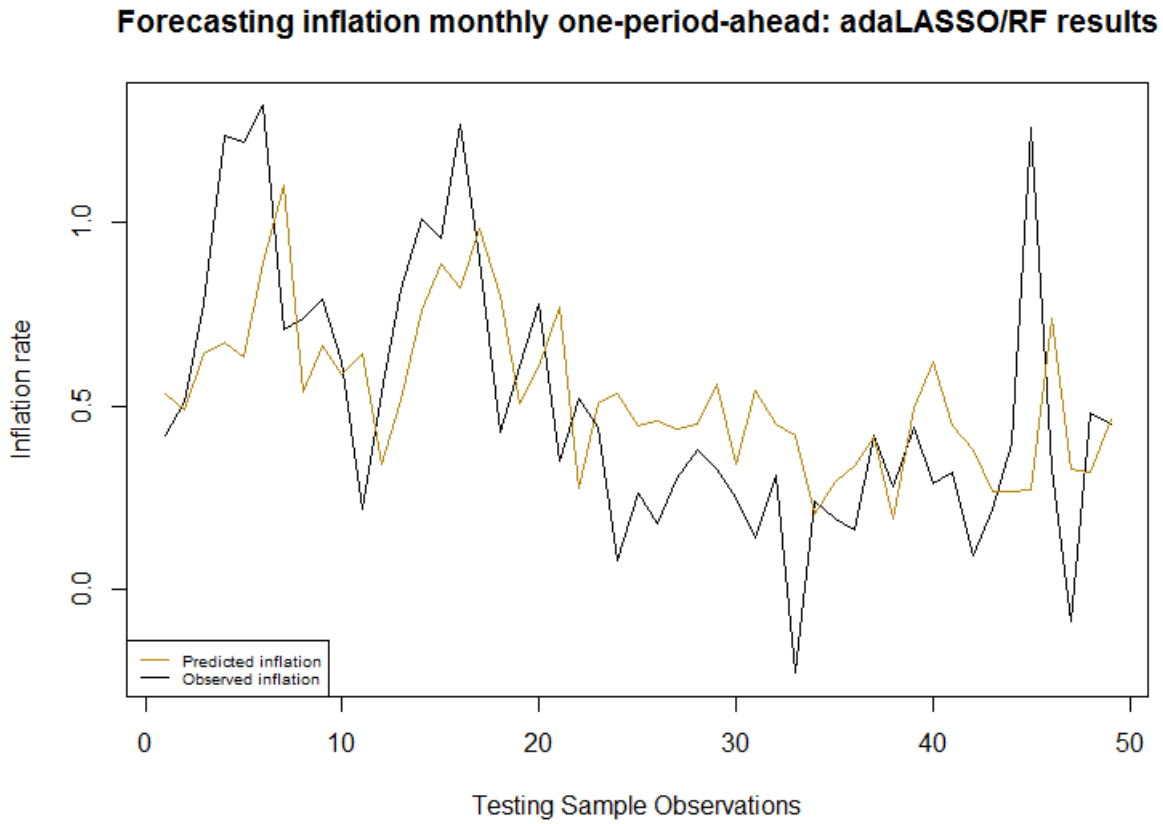

Nowcasting inflation daily: RF results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index


Forecasting inflation monthly one-period-ahead: Random Forest results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations


Nowcasting inflation daily: RF/OLS results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index


Forecasting inflation monthly one-period-ahead: RF/OLS results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations


Nowcasting inflation daily: adaLASSO/RF results
real
0.0
0.5
1.0
1.5
0
500
1000
1500
2000
2500
3000
Index


Forecasting inflation monthly one-period-ahead: adaLASSO/RF results
Inflation rate
0.0
0.5
1.0
Predicted inflation
Observed inflation
0
10
20
30
40
50
Testing Sample Observations