Machine learning methods for inflation forecasting in Brazil: new contenders *versus* classical models*

Gustavo Silva Araujo †

Wagner Piazza Gaglianone ‡

February 19, 2020

Abstract

We conduct an extensive out-of-sample forecasting exercise, across a variety of machine learning techniques and traditional econometric models, with the objective of building accurate forecasts of the Brazilian consumer prices inflation at multiple horizons. A large database of macroeconomic and financial variables is employed as input to the competing methods. The results corroborate recent findings in favor of the nonlinear automated procedures, indicating that machine learning algorithms (in particular, random forest) can outperform traditional forecasting methods in terms of mean-squared error. The main reason is that some machine learning methods can yield a sizeable reduction in the forecast bias, while keeping the forecast variance under control. As result, forecast accuracy can be improved over traditional inflation forecasting models. These findings offer a valuable contribution to the field of macroeconomic forecasting, and provide alternative methods to the usual statistical models often based on linear statistical relationships.

Keywords: Machine Learning; Inflation; Forecast.

JEL Classification: C14; C15; C22; C53; C55; E17; E31.

*The views expressed in the paper are those of the authors and do not necessarily reflect those of the Banco Central do Brasil. We are especially grateful for the helpful comments and suggestions given by Marcelo Antonio T. de Aragão, Jorge Henrique de Frias Barbosa, Vicente da Gama Machado, Marcelo C. Medeiros, Euler Pereira G. de Mello and André Minella. We also benefited from comments given by the seminar participants at the 18th ESTE - Time Series and Econometrics Meeting (Gramado, September 2019) and the IV Workshop da Rede de Pesquisa do Banco Central do Brasil (Brasília, November 2019).

†Research Department, Banco Central do Brasil, and Ibmec-RJ. E-mail: gustavo.araujo@bcb.gov.br

‡Corresponding Author. Research Department, Banco Central do Brasil, Av. Presidente Vargas, 730, 14th floor, Centro, Rio de Janeiro, BRAZIL, CEP 20071-900. E-mail: wagner.gaglianone@bcb.gov.br

1 Introduction

Machine Learning (ML) is a branch of artificial intelligence, often described as the art and science of pattern recognition. It is essentially a data-driven approach, with mild assumptions about the underlying statistical relationships in the data, and entails a large variety of methods. Machine learning also usually comprises two core elements, a learning method and an algorithm, enabling one to automate as many of the modeling choices as possible in a manner that is not subject to the discretion of the forecaster (Hall, 2018).

Producing accurate forecasts is not an easy task, since it requires an approach complex enough to incorporate relevant variables but also focused on excluding irrelevant data. In this sense, machine learning methods, in general, are able to deal with nonlinear patterns in the data, often hidden to standard linear models, thus offering an alternative (and compelling) approach to traditional econometric models.

The objective of this paper is to forecast Brazilian inflation based on a large number of macroeconomic and financial variables. Our goal is also to assess whether machine learning approaches can indeed offer improvement to forecast accuracy in applied macroeconomics and make a contribution to the standard statistical toolkit used in macro forecasting.

Many emerging economies, including Brazil, have experienced periods of hyperinflation in past decades. Nowadays, inflation in those countries is much lower, in a historical perspective, but still greater and more volatile when compared to developed economies.[1] This empirical evidence adds uncertainty to investment decisions and shortens the investment horizon in emerging markets, making the construction of accurate inflation forecasts a relevant task in these economies.

It is well known in the literature that a good in-sample fit does not guarantee a good out-of-sample forecast performance (Greene, 2003). Moreover, machine learning algorithms generally deal with large amounts of data (*big data*). However, in macroeconomics, the usual indicators of interest are collected on an annual, quarterly or monthly basis and, therefore, lead to much less data accumulation compared, for instance, to a daily high-frequency database. In principle, this could undermine the reliability of machine learning results, especially since the data is split into training and test sets (i.e., in-sample and out-of-sample, respectively), reducing still further

[1] The IMF (2018) report projects a consumer prices inflation in 2019 (annual percent change) of 1.9% for the advanced economies and 5.2% for the emerging market and developing economies. Also, these figures are very heterogeneous among emerging countries. For example, the IMF forecasts for inflation in 2019 for Chile, Brazil and India are, respectively, 3.0%, 4.2% and 4.9%, whereas for Turkey and Argentina are 16.7% and 31.7%, respectively.

the actual data used for model estimation.

In order to check for actual predictive power, we construct an out-of-sample empirical exercise with sixteen inflation forecasting methods, and forecast horizon ranging from one month up to twelve months. The list of competing models includes some traditional econometric approaches (ARMA, VAR), reduced-form structural models (Phillips curves), factor models, regularization methods (lasso, ridge, elastic net), regression trees (random forest, quantile regression forest) and survey-based forecasts (Focus).

Real-time inflation forecasting has been studied extensively in the literature (Stock and Watson, 1999). Due to an ever-changing world, producing reliable inflation forecasts is a constant challenge for policymakers and of greatest importance to economic agents and their investment decisions. One of the key features of the inflation dynamics in emerging economies is the degree of persistence (or inertia).[2] Besides past inflation, other predictors usually suggested in the literature to forecast inflation include measures of economic slack (e.g., unemployment) in a traditional Phillips curve setup, variables related to production (Stock and Watson, 1999), financial variables (Forni et al., 2003), surveys of expectations (Ang et al., 2007; Faust and Wright, 2013), among many others.

Moreover, there is a variety of approaches to model the inflation dynamics. According to Ang et al. (2007), economists use four main methods to forecast inflation: time-series models (e.g., ARIMA), structural models (e.g., Phillips curve), asset price models (e.g., term-structure of interest rate), and methods that employ survey-based measures (e.g., survey of professional forecasters). This lack of consensus motivates the use of an automated method to find out what are the best variables to predict inflation at different horizons.

The literature on macroeconomic forecasting using machine learning methods is relatively new and far from extensive; see Medeiros et al. (2016) and Garcia et al. (2017) for applications with Brazilian data, and Cheng et al. (2019) for aggregating individual survey-based forecasts, using machine learning tools, to improve forecasting of the US inflation.

Our research contributes to this fast growing literature in three ways: The first original contribution of this paper is to propose a new quantile combination approach using quantile regression forest to build conditional mean forecasts. The second contribution is to help "open-

[2] In Brazil, the relevance of past inflation has been vastly documented. For instance, Kohlscheen (2012) suggests that models in which past inflation have greater weight in the expectations formation process are more accurate than others purely based on the rational expectations assumption. In turn, Gaglianone, Guillén and Figueiredo (2018) points out to the relevance of considering a time-varying inertia when building accurate forecasting models.

ing" the machine learning *black box*,[3] by constructing a set of auxiliary graphs: (i) *word cloud* and *variable importance* plots to reveal the most important variables for inflation forecasting, thus summarizing the more relevant predictors according to a given ML method of interest; (ii) decomposition of the mean-squared forecast error plots, which allows one to disentangle the effect of forecast bias from the variance of the forecast. This is particularly important in model selection and helps understanding why some methods display a better forecast accuracy compared to others; and (iii) time series plots of the differences between the cumulative squared prediction error, which complement the graphical analysis by presenting the evolution over time of the cumulative performance of a given forecasting method in respect to a selected benchmark. The third contribution is to build a set of high dimensional models to forecast Brazilian inflation, in the same spirit as in Medeiros et al. (2016). However, compared to the previous papers: (i) we broaden the range of forecasting methods under consideration to include different ML approaches, besides considering a set of traditional inflation forecasting methods; and (ii) we put together a larger database of financial and macroeconomic variables, to include additional predictors that might be of practical interest.

The outline of the paper is as follows. In Section 2, we present the methodology comprising machine learning methods and traditional econometric models to predict inflation. Section 3 presents an out-of-sample empirical exercise and Section 4 concludes.

2 Methodology

2.1 *Machine Learning* in a nutshell

Most traditional forecasting methods rely on fitting data to a pre-specified relationship between dependent and independent variables, thus assuming a specific functional and stochastic process. In contrast, a different approach to statistical analysis and forecasting, in particular, is offered by machine learning (ML), which is to a great extent a data-driven approach, since it makes almost no assumption about the underlying statistical relationship in the data.

According to Hansen (2019): *"The term 'machine learning' is a new and somewhat vague term, but typically is taken to mean procedures which are primarily used for point prediction in settings with unknown structure. Machine learning methods generally allow for large sample sizes, large number of variables, and unknown structural form."* In fact, machine learning

[3]The *black box* term applied to describe ML techniques has been around for years now. It is often employed to critisize neural networks' lack of explainability. We take a step towards transparency (turning the *black box* into a *gray box*) by providing complementary tools to analyze and further understand the ML outcomes.

encompasses a wide variety of models, nonetheless, it often comprises two core elements: a *learning method*, where data is used to determine the best fit for the input variables, and an *algorithm* which models the relationship between the input and output. In general, ML can be categorized into three types (see Jung et al., 2018):

(i) *supervised learning*, where the dependent variables are clearly identified, even if the specific relationships in the data are not known (e.g., linear regression, logistic regression);

(ii) *unsupervised learning*, where there is no specific output defined beforehand, and the goal is to recognize data patterns and determine output classification categories (e.g., cluster analysis, principal components); and

(iii) *reinforcement learning*, which iteratively search for an optimal location of the input variables that yield the highest reward, that is, optimize a given "reward" function using no training set (e.g., sarsa, Q-learning).

According to Varian (2014), the growing amounts of data and ever complex-growing relationships warrant the usage of machine learning approaches in economics. Here, we build inflation forecasts using five different machine learning (*supervised*) algorithms: ridge regression, lasso, elastic net, random forest and quantile regression forest.

The first three methods are regularization techniques that introduce penalties for *overfitting*[4] the data. For example, the elastic net approach mixes two different kinds of regularization, by penalizing both the number of variables in the model and the extent to which any given variable contributes to the model's forecast. By applying these penalties, the elastic net *learns* which variables are most important, eliminating the need for researchers to make discretionary choices about which variables to include.

The last two methods are nonparametric approaches, based on the recursive binary partitioning of the covariate space, which can deal with very large number of explanatory variables, thus producing highly nonlinear predicted models.

[4]In statistics, *overfitting* denotes the production of an analysis, which is assumed to be valid for the entire population (for instance, an estimated input-output relationship), that corresponds too closely to a particular set of data, but it may fail to fit additional data, or forecast future observations, reliably.

2.2 ML estimation and forecasting

Our main goal is to forecast the inflation rate y_{t+h}, at period $t+h$, using the information set available at period t. In this sense, inflation is modeled as a function of a set of predictors $\widetilde{x}_t$, measured at time t, as follows:

$$y_{t+h} = \Upsilon_h\left(\widetilde{x}_t\right) + \varepsilon_{t+h}, \tag{1}$$

where $\Upsilon_h(\cdot)$ is a possibly nonlinear mapping of a set of predictors, ε_{t+h} is the forecasting error and $\widetilde{x}_t$ may include weakly exogenous predictors, lagged values of inflation and a number of factors computed from a large number of potential covariates; see Garcia et al. (2017).

Here, we consider $\widetilde{x}_t' \equiv \{\mathbf{1}_t, x_t, x_{t-1}, \ldots, x_{t-s}, d_{1,t}, ..., d_{11,t}\}$, where $\mathbf{1}_t$ is a constant term, $x_t = \{x_{1,t}, \ldots, x_{n,t}\}$ is a set of n predictors, $d_{i,t}$ are dummies added to control for seasonality[5] and s is the maximum lag adopted for the set of variables x_t when forming the database $\widetilde{x}_t'$.

In order to build our forecasting exercise, we divide the sample into two sub-periods: the first one ($t = 1, ..., T_1$) is labeled as "training set", where observations of inflation (y_t) are confronted with forecasts provided by the ML. The out-of-sample forecasts are considered in the second sub-period, also known as the "test set", comprising the last P observations of our sample ($t = T_1 + 1, ..., T$). This way, $P = T - T_1$ observations are used to compare different forecasts, computing forecast-accuracy measures.

For the three regularization approaches considered in this paper (ridge regression, lasso and elastic net), the mapping $\Upsilon_h(\cdot)$ is linear, such that:

$$y_{t+h} = \widetilde{x}_t'\beta_h + \varepsilon_{t+h}, \tag{2}$$

where $\beta_h \in \mathbb{R}^{ns+12}$ is a vector of unknown parameters. The inflation forecast from the linear ML approach, $f^{ML}_{y_{T_1+h}}$, using a sample of $t = 1, ..., T_1$ observations, is given by:

$$f^{ML}_{y_{T_1+h}} = \widetilde{x}_{T_1}'\widehat{\beta_h}, \quad \text{for } h = 1, ..., H. \tag{3}$$

To evaluate forecast $f^{ML}_{y_{T_1+h}}$, we compute its respective mean-squared error as follows: $MSE_h = \frac{1}{P}\sum_{t=T_1+1}^{T}\left(y_t - f^{ML}_{y_{T_1+h}}\right)^2$.

Note that we adopt the *direct forecast* approach, where the inflation h periods ahead (y_{T_1+h}) is modeled as a function of a set of predictors $\widetilde{x}_{T_1}'$ measured at time T_1. In other words, for

[5] Although inflation seasonality could alternatively be captured by the candidate predictors with seasonal behavior.

each horizon h we estimate a different vector of unknown parameters β_h (in contrast to the iterated multistep approach; see Marcellino, Stock and Watson, 2006). This way, we avoid the necessity of estimating a model for the time-evolution of $\widetilde{x}_t$.

2.2.1 Ridge Regression

It is well known that OLS often does poorly in prediction on future data, for instance, due to overfitting. In this sense, penalization techniques have been proposed in the literature to improve OLS accuracy. For instance, the ridge regression (see Hoerl and Kennard, 1988) minimizes the squared sum of the residuals subject to a bound on the l_2-norm of the parameters, as follows:

$$\widehat{\beta} = \underset{\{\beta_1,\ldots,\beta_k\}}{\arg\min} \left(\frac{1}{T}\sum_{t=1}^{T} \left(y_t - \sum_{j=1}^{k} x'_{j,t}\beta_j \right)^2 + \lambda \sum_{j=1}^{k} \beta_j^2 \right), \tag{4}$$

where β is the $k \times 1$ vector of parameters, y_t is the dependent variable, $\left\{x_{1,t}, \ldots, x'_{k,t}\right\}$ is the $k \times 1$ vector of regressors and λ is the so-called shrinkage parameter.

Note that the extent of the shrinkage penalty is determined by the parameter λ, whose optimal value will in practice be determined by *cross-validation* (i.e., splitting the data into K folds and iteratively re-estimating the model for each fold). Choosing a higher λ will lead to a stronger shrinkage of the regression coefficients, whereas setting $\lambda = 0$ will produce the same results of a standard ordinary least squares (OLS) regression.

Also, because ridge regression is a continuous shrinkage method, it can achieve a better out-of-sample performance through a *bias-variance* trade-off (i.e., use regularization to balance the forecast errors due to bias and variance). In particular, the ridge regression is good at improving the OLS counterpart when multicollinearity is present. However, ridge cannot produce a parsimonious model, since it always keeps all the predictors in the model.

2.2.2 Lasso

The least absolute shrinkage and selection operator (lasso) was originally proposed by Tibshirani (1996). The core idea is to shrink to zero the irrelevant coefficients. The lasso is a penalized least squares method imposing an l_1-penalty on the regression coefficients, as follows:

$$\widehat{\beta} = \underset{\{\beta_1,\ldots,\beta_k\}}{\arg\min} \left(\frac{1}{T}\sum_{t=1}^{T} \left(y_t - \sum_{j=1}^{k} x'_{j,t}\beta_j \right)^2 + \lambda \sum_{j=1}^{k} \left|\beta_j\right| \right), \tag{5}$$

where (as in the ridge regression) β is the vector of parameters and λ is the shrinkage parameter. Due to the nature of the l_1-norm, lasso is able to do continuous shrinkage and automatic *variable selection* simultaneously, whereas the ridge regression only shrinks the coefficients close to zero (but does not exclude them from the model). Also, by setting $\lambda = 0$ leads to the OLS estimation.

According to Cheng et al. (2019), lasso is "*the most intensively studied statistical method in the past 15 years*". Indeed, it has shown success in many practical situations, since it can handle more variables than observations.

Nonetheless, it has some limitations and might even become an inappropriate variable selection method in some cases. Zou and Hastie (2005) list a few examples: (i) when the number of predictors k is greater than the number of observations T, the lasso selects at most T variables before it saturates, due to the nature of the convex optimization problem; (ii) in the case of *grouping effect*[6], the lasso tends to select only one variable from the group (and does not care which one is selected); (iii) in the case of $T > k$ and in the presence of high correlations between predictors, it has been empirically observed that ridge regression tends to perform better than lasso.

2.2.3 Adaptive Lasso

Zou (2006) shows that the lasso estimator is inconsistent for variable selection under certain circumstances. This way, the author proposes a new version of the lasso, called the adaptive lasso (or simply *adalasso*), where adaptive weights are used for penalizing different coefficients in the l_1-penalty. According to the author, the adaptive lasso enjoys the oracle properties (i.e., it performs as well as if the true underlying model were known) and not select useless variables (which may damage the forecasting accuracy). The core idea behind the model is to use some previously known information to select the variables more efficiently.

In practice, it consists of a two-step estimation that uses a first model to generate different weights w_j for each candidate variable $x_{j,t}$. These weights are used in the second-step in the lasso estimation as additional information. The *adalasso* estimator is thus defined as:

$$\widehat{\beta} = \underset{\{\beta_1,\ldots,\beta_k\}}{\arg\min} \left(\frac{1}{T} \sum_{t=1}^{T} \left(y_t - \sum_{j=1}^{k} x'_{j,t} \beta_j \right)^2 + \lambda \sum_{j=1}^{k} w_j \left| \beta_j \right| \right), \tag{6}$$

where $w_j = \left| \widehat{\beta}^*_j \right|^{-\tau}$ represents the weights; $\widehat{\beta}^*_j$is a parameter estimated in the first-step, and

[6]The grouping effect occurs if the regression coefficients of a group of highly correlated variables tend to be equal (up to a change of sign if negatively correlated).

$\tau > 0$ is an additional tuning parameter (which can be chosen by using the same criterion as λ) that determines how much one wants to emphasize the difference in the weights.

In general, τ is set to unity and $\widehat{\beta}_j^*$ is the respective lasso coefficient estimated in the first-step. According to Medeiros and Mendes (2016), the conditions required by the *adalasso* estimator are very general, and the model works even when the errors are non-Gaussian, heteroskedastic, and the number of variables increases faster than the number of observations.

2.2.4 Elastic Net

The elastic net is a regularization and variable selection method proposed by Zou and Hastie (2005), as a generalization of the lasso. Similarly to the lasso, the elastic net simultaneously does automatic variable selection and continuous shrinkage, and it can select groups of correlated variables. According to the authors: "*It is like a stretchable fishing net that retains 'all the big fish'.*"

Simulation studies show that the elastic net often outperforms the lasso, in terms of predictive power, while enjoying a similar sparsity representation. The elastic net encourages a grouping-effect, where highly correlated regressors tend to be jointly included (or excluded) from the model, and it can be particularly useful when the number of predictors k is high when compared to the number of observations T.

For a nonnegative shrinkage parameter λ, and a combination parameter α strictly between 0 and 1, the elastic net solves the following problem:

$$\widehat{\beta} = \underset{\{\beta_1,\ldots,\beta_k\}}{\arg\min} \left(\frac{1}{T}\sum_{t=1}^{T}\left(y_t - \sum_{j=1}^{k} x'_{j,t}\beta_j \right)^2 + \lambda P_\alpha\left(\beta\right) \right), \tag{7}$$

where

$$P_\alpha\left(\beta\right) = \sum_{j=1}^{k} \alpha\left|\beta_j\right| + \frac{(1-\alpha)}{2}\beta_j^2. \tag{8}$$

Note that the elastic net is the same as the lasso when $\alpha = 1$. As α shrinks toward 0, elastic net approaches the ridge regression. For other values of α, the penalty term $P_\alpha\left(\beta\right)$ interpolates between the l_1-norm of β and the squared l_2-norm of β. The tuning parameter λ controls the overall strength of the penalty. Note the objective function is convex and so can be minimized using any convex optimization method such as gradient or coordinate descent.

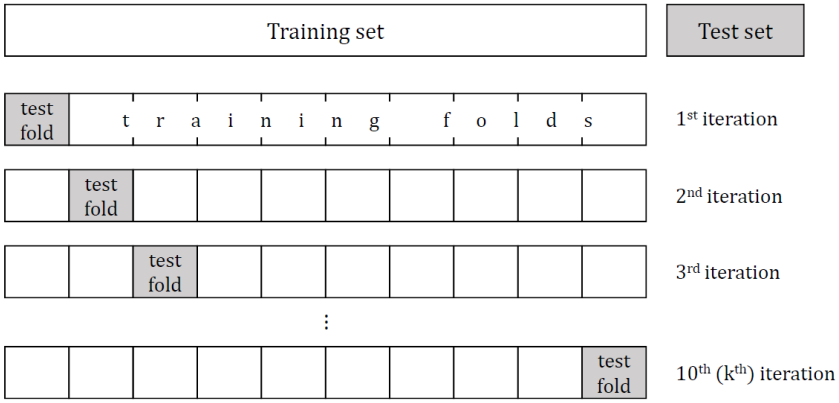
In addition, although we defined the elastic net by using (λ, α), this is not the only choice as the tuning parameters; see Zou and Hastie (2005). For example, one could use the l_1-norm

of the coefficients or the fraction of the l_1-norm to parameterize the elastic net.

Choice of the tuning parameters (λ, α) There are well-established methods for choosing tuning parameters. For instance, K-fold *cross-validation* is a popular method for computing the prediction error and comparing different models using training data. The loss often used for cross-validation is the mean squared-error (MSE). The goal is to produce the so-called "cross-validation curve", which is built by computing the MSE as a function of the tuning parameter λ chosen over a pre-selected grid.

To do so, for each selected fold, the algorithm splits the training set of observations in two parts: *training folds* (used for the estimation of parameters) and *test fold* (based on the remaining observations, used for model predictions); see Figure 1. Then, forecast errors are computed and used to calculate the MSE over the entire set of predictions using all K-folds.

Figure 1 - Example of a K-fold cross-validation ($K = 10$)



Source: Jung et al. (2018).

In the elastic net, there are two tuning parameters, so one needs to cross-validate the model on a two-dimensional surface. The minimum MSE, thus, provides the pair (λ, α) to be used in the final model estimation. Parameters can be estimated using the penalized maximum likelihood, in which the regularization path (i.e., the path of each coefficient β_j against, for instance, the l_1-norm of the whole coefficient vector as λ varies) can be computed.

On the other hand, Zou et al. (2007) show that one can consistently estimate the degrees of freedom of the lasso model using information criterion as an alternative to the cross-validation approach. An advantage of such procedure is that selecting the model using information criterion is faster than using cross-validation. More importantly, performing cross-validation in a time-series context may be complicated in cases where the data is not independent and identically distributed (i.i.d.). See Medeiros et al. (2016) for further details.

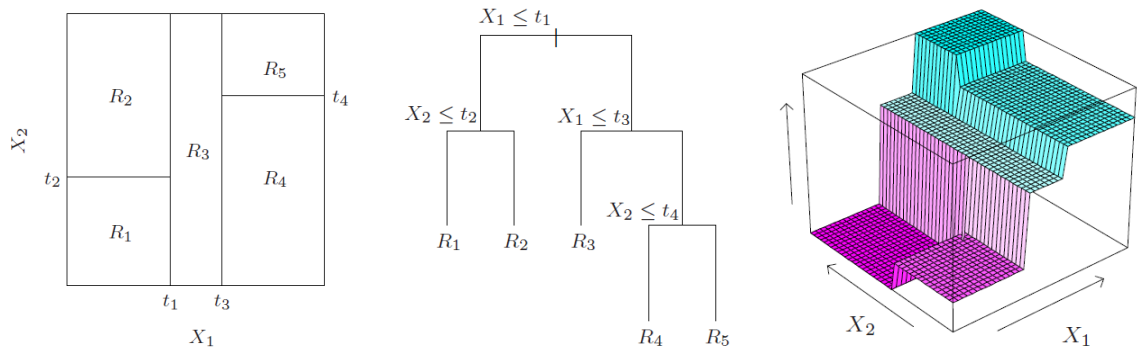
In this paper, we select the best lasso, adalasso and elastic net models using the Bayesian Information Criterion (BIC).

2.2.5 Random Forest

Random Forest (RF) was introduced as a machine learning tool in Breiman (2001) and have since proven to be very popular and powerful for high-dimensional regression and classification. A random forest is a collection of regression trees, designed to reduce the prediction variance by using bootstrap aggregation (*bagging*) of randomly constructed regression trees.[7]

A *regression tree* is a nonparametric model based on the recursive binary partitioning of the covariate space X. The main idea is that if a sufficiently large number of step functions are used, then a step function can be a good approximation to any functional form.[8] According to Garcia et al. (2017), the model is usually displayed in a graph, which has the format of a binary decision tree with P parent nodes (or split nodes) and L terminal nodes (also called leaves; which represent different partitions of X).[9] Figure 2 shows an example of a regression tree with two covariates.

Figure 2 - Example of a recursive binary splitting in a regression tree



Notes: The left panel shows an example of partition of a two-dimensional covariate space by recursive binary splitting. The center panel exhibits the corresponding tree and the right panel shows a perspective plot of the prediction surface. Source: Hastie et al. (2009, chapter 9).

[7] According to Hastie et al. (2009), tree learning is invariant under scaling and various other transformations (and it is robust to inclusion of irrelevant covariates), however it is seldom accurate. In particular, large trees tend to learn highly irregular patterns and overfit their training sets, thus producing low bias but very high prediction variance. In order to reduce such high variance, random forests average multiple decision trees, trained on different parts of the same training set. This often comes at the expense of a small increase in the bias, but generally improves the overall performance of the final model.

[8] According to Hansen (2019): *"The literature on regression trees has developed some colorful language to describe the tools, based on the metaphor of a living tree. 1. A split point is node. 2. A subsample is a branch. 3. Increasing the set of nodes is growing a tree. 4. Decreasing the set of nodes is pruning a tree."*

[9] According to the authors, the partitions are often defined by a set of hyperplanes, each of which is orthogonal to the axis of a given predictor variable (also called the split variable).

Note that we first split the covariate space into two regions ($X_1 \leq t_1$ and $X_1 > t_1$)[10] and model the dependent variable by the mean of Y in each region. The selected variable (X_1) and the corresponding split-point (t_1) are chosen in order to achieve the best fit. Then one (or both) of these regions are split into two more regions, and this process is continued, until some stopping rule is applied. In the example shown in Figure 2, the regression tree model predicts Y with a constant c_m in region R_m, $m = 1, ..., 5$, as follows:

$$E_{\text{regression tree}}\left(Y \mid (X_1, X_2)\right) = \sum_{m=1}^{5} c_m \mathbf{1}_{\{(X_1,X_2)\in R_m\}}. \tag{9}$$

We now turn to the question of how to properly grow a regression tree: the algorithm needs to automatically decide on both the splitting variables and split points. In the previous example, if one assumes a mean-squared error loss function, the optimal $\widehat{c_m}$ is simply the average of the response Y in the region R_m. However, finding the best partition in terms of overall MSE, according to Hastie et al. (2009), is usually computationally infeasible. In this sense, the authors propose the following approach, focused on the implementation of CART (classification and regression tree) models:

(i) consider a splitting variable j and split point s, and define the pair of half-planes:

$$R_1(j,s) = \{X \mid X_j \leq s\} \quad \text{and} \quad R_2(j,s) = \{X \mid X_j > s\}, \tag{10}$$

(ii) find the splitting variable j and split point s that solve the minimization problem:

$$\min_{j,s}\left[\min_{c_1}\sum_{x_i\in R_1(j,s)}(y_i - c_1)^2 + \min_{c_2}\sum_{x_i\in R_2(j,s)}(y_i - c_2)^2\right], \tag{11}$$

where the previous inner minimizations, for any choice j and s, can be solved by:

$$\widehat{c_1} = E\left(y_i \mid x_i \in R_1(j,s)\right) \quad \text{and} \quad \widehat{c_2} = E\left(y_i \mid x_i \in R_2(j,s)\right). \tag{12}$$

Note that for a given splitting variable, the calculation of the optimal split point s can be easily done. Thus, by searching through all covariates, the determination of the best pair (j, s) is feasible. Then, based on the best split, we divide the data into the two resulting regions R_1

[10] Rather than splitting each node into just two groups, one might consider multiple splits into more than two groups at each stage. However, according to Hastie et al. (2009, p.311), while this can sometimes be useful, it is not a good general strategy, since multiple splits fragment the data too quickly, leaving insufficient data at the next level down.

and R_2 and repeat the splitting process on each of the two regions. This process is repeated on all of the resulting regions. To sum it up, the regression tree can be estimated by repeating the three steps below, for each terminal node of the tree, until the minimum number of observations at each node is achieved:

(1) randomly select m out of p covariates as possible split variables;[11]

(2) select the best variable/split point among the m candidates;

(3) split the node into two child nodes.

In practice, one major problem with regression trees is their high prediction variance. Usually, a small change in the data lead to a very different sequences of splits. The main reason for such instability is the hierarchical nature of the algorithm: the effect of a big error in the top split is propagated down to all of the splits below it.

To overcome this issue, one can employ the *bagging* technique (i.e., bootstrap aggregation), which consists on fitting the same regression tree several times to bootstrap-sampled versions of the training data and average the result. This bootstrapping approach often leads to better model performance because it decreases the variance of the model, without increasing too much the bias.[12]

The *random forest* approach uses a modified bagging algorithm (called *random subspace projection*) that selects, at each candidate split in the learning process, a random subset of the covariates. The reason for doing this is the correlation of the trees in an ordinary bootstrap sample: if one or a few covariates are very strong predictors for the dependent variable, these covariates will be selected in many of the K bootstraped trees, causing them to become correlated. According to Hansen (2019), the modification proposed by RF is to *decorrelate* the bootstrap regression trees by introducing extra randomness. Besides, the reduction of the tuning parameter m will, in general, reduce the correlation between any pair of trees. The random forest algorithm can be summarized as follows:[13]

Given a training set (Y_i, X_i), for $i = 1, ..., n$, where Y is the dependent (response) variable

[11] The size of a tree is a tuning parameter governing the model's complexity, and the optimal size should be adaptively chosen from the data. The preferred strategy is to stop the splitting process when some minimum node size is reached. Typically, for regression problems with p predictors, the literature recommends to use $m = p/3$ (rounded down) in each split, with a minimum node size of 5 as the default; see Hastie et al. (2009, chapter 15.3) for more details.

[12] While the predictions of a single tree are highly sensitive to noise in its training set, the average of many trees might be not, as long as the trees are not correlated. Besides, training many trees on a single training set would give strongly correlated trees, whereas bootstrap sampling helps de-correlating the trees by showing them different training sets.

[13] The appendix B provides a short mathematical description of the random forest approach. See also Hastie et al. (2009, chapters 9 and 15) for further details.

and X represents a set of covariates, bagging repeatedly (K times) selects a random sample (with replacement) of the training set and fits regression trees to these bootstraped samples, that is, for $k = 1, ..., K$:

(i) sample with replacement n training observations from (X, Y); calling them (X_k, Y_k);

(ii) train a regression tree $T_k(\cdot)$ on (X_k, Y_k);

(iii) build the random forest prediction of Y conditioned on the test set (unseen samples x') by averaging the predictions from all the individual regression trees on x', as follows:

$$E_{\text{random forest}}(Y \mid X = x') = \frac{1}{K}\sum_{k=1}^{K} T_k(x'). \tag{13}$$

2.2.6 Quantile Regression Forest

Random forest approximates the conditional mean of Y by constructing a weighted average over the sample observations of Y. Nonetheless, random forests can also provide information about the full conditional distribution of the response variable, not only about the conditional mean. This information can be used, for instance, to build prediction intervals and account for outliers in the data. This way, conditional quantiles can be inferred with quantile regression forests (QRF), a generalization of random forests proposed by Meinshausen (2006).[14]

The idea is to provide a non-parametric way of estimating conditional quantiles for a high-dimensional set of predictor variables. According to the author, the QRF algorithm is shown to be consistent and competitive in terms of predictive power. First, recall that the conditional distribution function (CDF) of Y, given $X = x$, is given by:

$$F(y \mid X = x) = \Pr(Y \leq y \mid X = x) = E\left(I_{\{Y \leq y\}} \mid X = x\right). \tag{14}$$

Also, recall that the conditional quantile of Y, given $X = x$, at quantile level τ, is given by:

$$Q_\tau(Y \mid X = x) = \inf\{y : F(y \mid X = x) \geq \tau\}. \tag{15}$$

In other words, for a continuous distribution function of Y, conditional on $X = x$, the probability of Y being smaller than $Q_\tau(\cdot)$ is exactly equal to τ. Now, similarly to the random forest approximation of the conditional mean, define an approximation to $E\left(I_{\{Y \leq y\}} \mid X = x\right)$

[14] The main difference between QRF and RF is that for each node (in each tree), RF keeps only the mean of the observations that fall into this node (and neglects all other information). In contrast, QRF keeps the value of all observations in this node (not just their mean) and assesses the conditional distribution based on this full information.

by the weighted mean over the observations of $I_{\{Y \leq y\}}$, as follows:

$$\widehat{F}\left(y \mid X = x\right) = \sum_{i=1}^{n} w_i(x) I_{\{Y_i \leq y\}}, \tag{16}$$

using the same weights $w_i(x)$ for random forests, as defined in the appendix A. See the appendix B for a summary of the algorithm used to compute the previous CDF estimate. Estimates $\widehat{Q}_\tau\left(\cdot\right)$ of the conditional quantiles $Q_\tau\left(\cdot\right)$ can, thus, be obtained by simply plugging $\widehat{F}\left(y \mid X = x\right)$, instead of $F\left(y \mid X = x\right)$, into (15).

In this paper, we go one step further by relating the conditional quantiles with the conditional mean of Y. This could be accomplished by integrating the conditional quantile function of Y over the entire domain $\tau \in [0, 1]$, as follows (see Koenker, 2005, p.302):

$$E\left(Y \mid X = x\right) = \int_0^1 Q_\tau\left(Y \mid X = x\right) d\tau. \tag{17}$$

The conditional mean of Y, based on the QRF approach, can thus be approximated[15] by a sum of estimated conditional quantiles, as follows:[16]

$$\int_0^1 Q_\tau\left(Y \mid X = x\right) d\tau = \lim_{P \to \infty} \left(\sum_{p=1}^{P} \widehat{Q}_{\tau_p}\left(Y \mid X = x\right) \Delta\tau_p \right). \tag{18}$$

The approximation of the conditional mean by a combination of conditional quantiles is not a novel approach in the literature. Indeed, it has a long tradition in statistics (see Judge et al., 1988) and has been previously applied in the forecasting literature. Nonetheless, an original contribution of this paper is to propose a new quantile combination approach, based on quantile regression forest to build conditional mean forecasts, through equations (15), (16), (17) and (18).

The proposed quantile combination approach based on QRF follows the spirit of the averaging scheme applied to quantiles conditional on predictors selected by *lasso*, as proposed by Lima and Meng (2017). The advantage of these approaches relies on the fact that quantiles are robust to outliers (in our case, extreme unanticipated inflationary shocks), which potentially improves forecast-accuracy and likely impact the performance of standard models, which are usually designed to only account for average responses.

[15] By applying the second fundamental theorem of calculus (or the Newton-Leibniz axiom) on the sum of quantiles, the Riemann integral is obtained in the limit $P \to \infty$ (see Apostol, 1967) and the partitions $\Delta\tau_p = \frac{1}{P+1}$ get finer (i.e., $\Delta\tau_p \to 0$ as long as $P \to \infty$).

[16] We rely on the fact that the conditional quantiles are consistenly estimated using the QRF approach.

In other words, each considered predictor might be useful to forecast some, but not all, conditional quantiles of Y (being called as *partially weak predictor*). Moreover, if the predictor helps forecast all quantiles, it is then considered to be a *strong predictor*, whereas predictors that help predict no quantile at all are called *fully weak predictors*.

According to Lima and Meng, the quantile combination method usually results in a prediction model in which the coefficients of fully weak predictors are not statistically significant (in contrast to statistically significant strong predictors), while the coefficients of partially weak predictors are adjusted to reflect the magnitude of their contribution to the conditional mean forecast. This method potentially offers improvement in forecast accuracy compared to usual conditional mean models not designed to deal with partial and fully weak predictors across quantiles and over time.

2.3 Traditional Inflation Forecasting

We next build inflation forecasts using more traditional methods. This short suite of models, of course, is far from an exhaustive list and should not be interpreted as the best possible one, since more complex models (e.g., DSGE) could be included. Although we think that this extension would be valuable, the list next presented also seems to be a reasonable approximation to the spectrum of models often used by economic agents interested in producing inflation forecasts.

Random Walk: The standard random walk (RW) model assumes that the $h-$period inflation change is an unforecastable martingale difference sequence (MDS), that is $E\left(y_{t+h} - y_t \mid \mathcal{F}_t\right) = 0$, for all $t = 1, ..., T_1$ and $h = 1, ..., H$. The out-of-sample inflation forecast, $f^{rw}_{y_{T_1+h}}$, is given by:

$$f^{rw}_{y_{T_1+h}} = y_{T_1}. \tag{19}$$

RW-AO: This is the variant of the pure random walk model, considered by Atkeson and Ohanian (2001), which takes the average inflation over the previous four quarters as the forecast for y_{T_1+h}. Here, in order to build a low-variance forecast, we consider the moving average over the previous four years, as follows:

$$f^{rw-ao}_{y_{T_1+h}} = \frac{1}{48}\sum_{j=0}^{47} y_{T_1-j}. \tag{20}$$

ARMA: One of the most common statistical models used for time-series forecasting is the autoregressive moving average (ARMA) model, which assumes that future observations are

primarily driven by recent observations. Inflation, which often exhibits persistent behavior, is largely consistent with this assumption. The best model for the monthly inflation rate y_t, in our sample, according to the Schwarz information criterion, is the AR(1), described as follows:

$$y_t = \alpha + \beta y_{t-1} + \varepsilon_t, \tag{21}$$

where the estimates $\left[\widehat{\alpha}; \widehat{\beta}\right]'$ can be computed using a sample with $t = 1, ..., T_1$ observations. The respective h-step-ahead forecast ($f^{ar}_{y_{T_1+h}}$) is given by

$$f^{ar}_{y_{T_1+h}} = \widehat{\beta}^h y_{T_1} + \sum_{i=0}^{h-1} \widehat{\alpha}\widehat{\beta}^i. \tag{22}$$

VAR: The vector autoregression (VAR) is also a traditional forecasting method based on a backward-looking approach. In its basic form, a VAR consists of a set of k endogenous variables $x_t = \left[\begin{array}{ccc} x_{1,t} & \cdots & x_{k,t} \end{array} \right]'$. The VAR($p$) process is defined as $x_t = F_1 x_{t-1} + ... + F_p x_{t-p} + \phi + u_t$, where F_i are $k \times k$ coefficient matrices for $i = 1, .., p$, ϕ is a $k \times 1$ vector of intercepts and u_t is a $k-$dimensional white noise process with $E(u_t) = 0$. As well-known, the VAR(p) can be rewritten as a VAR(1), as follows:

$$\xi_t = F\xi_{t-1} + c + v_t, \tag{23}$$

where $\xi_t = \left[\begin{array}{cccc} x_t & x_{t-1} & \cdots & x_{t-p+1} \end{array} \right]'$ is a $kp \times 1$ vector stacking all variables (and lags), F is a $kp \times kp$ matrix and $c = \left[\begin{array}{cccc} \phi & 0 & \cdots & 0 \end{array} \right]'$ and $v_t = \left[\begin{array}{cccc} u_t & 0 & \cdots & 0 \end{array} \right]'$ are $kp \times 1$ vectors. Successive substitution for lagged ξ_t's gives (Lütkepohl, 2005): $\xi_{t+h} = F^h \xi_t + \sum_{i=0}^{h-1} F^i (c + v_{t+h-i})$. Taking the conditional expectation on both sides: $E\left(\xi_{t+h} \mid \mathcal{F}_t\right) = F^h \xi_t + \sum_{i=0}^{h-1} cF^i$. Now, selecting the $i-th$ element of ξ_{t+h} gives $E\left(x_{i,t+h} \mid \mathcal{F}_t\right) = J_i' E\left(\xi_{t+h} \mid \mathcal{F}_t\right)$, where J_i' is a $1 \times kp$ selection-vector filled with zeros, excepting the $i-th$ element, which is set to one.

In this paper, the VAR is estimated with four endogenous variables: market price inflation (approximately 75% of IPCA), administered price inflation (approximately 25% of IPCA), M4 and nominal exchange rate (R$/US$),[17] and 1 lag (by Schwarz information criterion and diagnostic testing). The exchange rate and M4 are first-differenced to avoid unit roots. The choice of variables recognizes the different time dynamics of the two main components of inflation[18],

[17] See variables n.2, 3, 38 and 53 in Table C1 (Appendix C).

[18] The administered price inflation is in some way regulated by a public agency or set by contracts (often including backward indexation clauses), rather than by the interaction between domestic demand and supply

and incorporates the possible pass-through of imported inflation to domestic inflation. Assuming that $(x_{1,t}, x_{2,t})'$ denotes, respectively, the market price inflation and administered price inflation, the VAR forecast for the headline inflation, $f^{var}_{y_{T_1+h}}$, can be constructed by aggregating the h-step ahead forecasts of the two main inflation components, using $\omega = 0.75$ as the weight of the IPCA market, in respect to the IPCA headline, as follows:

$$f^{var}_{y_{T_1+h}} = \widehat{E}\left(\omega x_{1,T_1+h} + (1-\omega)\, x_{2,T_1+h} \mid \mathcal{F}_{T_1}\right) \tag{24}$$

$$= \left(\omega J_1' + (1-\omega)\, J_2'\right) \widehat{F}^h \xi_{T_1} + \sum_{i=0}^{h-1} \widehat{c}\widehat{F}^i. \tag{25}$$

PC-backward: The Phillips curve model (PC) has a long tradition in forecasting inflation (Stock and Watson, 1999). We consider here a backward-looking version of the curve, only including past inflation (inertia), imported inflation (pass-through channel) and output gap (traditional monetary policy channel via aggregate demand). Following the VAR approach, we also disaggregate inflation in two components (market price inflation and administered price inflation), which are modeled separately. First, we estimate a Phillips curve for inflation of market prices, as follows:

$$\pi^{market}_{t+h} = \alpha_0 + \alpha_1 \pi^{market}_t + \alpha_2 \pi^{imp}_t + \alpha_3 g_t + \varepsilon_{t+h}, \tag{26}$$

where π^{market}_t is the inflation of market prices, π^{imp}_t is the imported inflation,[19] and g_t is the output gap.[20] Regarding the administered price inflation, we estimate an auxiliary ARMA(p, q) model. The Schwarz information criterion for lag selection indicate one lag only and no moving average term ($p = 1, q = 0$). Finally, we use each model to produce h-step ahead point forecasts ($h = 1, \ldots, 12$ months) which are, then, aggregated using corresponding weights to build the forecast for the IPCA inflation.

PC-hybrid: This approach considers a hybrid (New Keynesian) version of the Phillips curve, which includes backward and forward looking terms, imported inflation and output gap; see

conditions. According to Minella et al. (2003), the dynamics of such prices differ from the market prices in three ways: (i) dependence on international prices in the case of refined petroleum products; (ii) greater pass-through from the exchange rate; and (iii) stronger backward-looking behavior.

[19] Defined as the sum of the nominal exchange rate (R$/US$) monthly percentage variation and the U.S. inflation (assumed to be 2.0% per year or, equivalently, 0.165% per month).

[20] The output gap is based on the seasonally adjusted IBC-BR index of economic activity. The Hodrick-Prescott (HP) filter is employed to generate the output gap in a recursive estimation scheme, that is, we re-construct the entire output gap series for each new observation added to the estimation sample along the out-of-sample exercise (and, then, re-estimate the Phillips curve to construct new $h-$step ahead forecasts).

Arruda et al. (2011) and Gaglianone, Issler and Matos (2017). The hybrid-version of the Phillips curve for inflation of market prices is given by:

$$\pi_{t+h}^{market} = \alpha_0 + \alpha_1 \pi_t^{market} + \alpha_2 \pi_{t+h|t}^{\exp} + \alpha_3 \pi_t^{imp} + \alpha_4 g_t + \varepsilon_{t+h}, \tag{27}$$

where the additional term $\pi_{t+h|t}^{\exp}$ denotes the $h-$step ahead expected inflation (*Focus* survey).[21] We impose the coefficient restriction: $\alpha_1 + \alpha_2 + \alpha_3 = 1$, to guarantee a vertical long-run Phillips curve. The inflation forecasts for the administered price inflation and the IPCA follow the same procedures described in the previous approach.

Factor model 1 (direct forecast): The idea that time variations in a large number of variables can be summarized by a small number of factors is empirically attractive and it is employed in a large number of studies in economics and finance; see Forni et al. (2000) and Stock and Watson (2002). Let $x_{i,t}$ be the observed data for the $i-$th cross-section unit at time t, for $i = 1, ..., N$ and $t = 1, ..., T_1$, and consider the following factor representation of the data:

$$x_{i,t} = \lambda_i' F_t + e_{i,t}, \tag{28}$$

where F_t is a vector of common factors, λ_i is a vector of factor loadings associated with F_t and $e_{i,t}$ is the idiosyncratic component of $x_{i,t}$. Note that λ_i, F_t and $e_{i,t}$ are unknown since only $x_{i,t}$ is observable. Here, we estimate the factors and respective loadings using principal components analysis (PCA), which is a well-established technique for dimension reduction in time series. The number of components is determined by the Bai and Ng (2002) criterion. After the PCA estimation of the common factors F_t, we employ the *direct forecast* approach, to model the inflation rate at time $t + h$, as follows:

$$y_{t+h} = \beta_h F_t + \varepsilon_{t+h}. \tag{29}$$

Therefore, the inflation forecast from the *direct* factor model approach, $f_{y_{T_1+h}}^{fm-direct}$, using a sample of $t = 1, ..., T_1$ observations, is given by:

$$f_{y_{T_1+h}}^{fm-direct} = \widehat{\beta_h} \widehat{F_{T_1}}, \quad \text{for } h = 1, ..., H. \tag{30}$$

Factor model 2 (iterated forecast): This approach is a variant of the previous one, but

[21] Median of survey forecasts, collected every 15^{th} day (or the next available workday) of the month m. For $h = 1, ..., 12$, it is the median forecast for month $m + 1, m + 2, ..., m + 13$.

using an iterated forecast method instead of the direct forecast approach. The idea is again to employ common factors, but to model the inflation rate in a contemporaneous way in respect to the factors, that is:

$$y_t = \gamma F_t + v_t. \tag{31}$$

Following the literature (e.g., Bańbura et al., 2013), we specify the factors as following a VAR process, that is, $F_t = \Phi(L)F_t + u_t$. Thus, the inflation forecast from the *iterated* factor model approach, $f_{y_{T_1+h}}^{fm-iterated}$, using a sample of $t = 1, ..., T_1$ observations, is given by:

$$f_{y_{T_1+h}}^{fm-iterated} = \widehat{\gamma}\widehat{F_{T_1+h|T_1}}, \quad \text{for } h = 1, ..., H, \tag{32}$$

where $\widehat{F_{T_1+h|T_1}}$ are the $h-$step ahead (out-of-sample) forecasts of the common factors, using the VAR model estimated in a recursive scheme.

Factor models 3 and 4 (with targeted predictors, direct or iterated): These are the same previous factor model forecasts, but based on a subset of predictors that are selected by taking into account that our variable of interest is the inflation rate. Here, we follow the idea of Bai and Ng (2008), who showed that the factor model out-of-sample forecasting performance could be improved by previously selecting (or targeting) the predictors.

The core idea is that irrelevant predictors employed to build a factor model only add noise into the analysis, and thus produce factors with a poor predictive performance. In this sense, we use a pre-selection of variables to be included in the factor analysis, as follows:

(i) in the direct forecast case, we first regress the inflation rate y_{t+h} (or y_t in the iterated case) on the intercept and the candidate variable $\widetilde{x}'_{i,t} \in \widetilde{x}'_t$, for all $i = 1, ..., q$.

(ii) calculate the $t-$statistics for the coefficient associated to $\widetilde{x}'_{i,t}$.

(iii) include $\widetilde{x}'_{i,t}$ in the set of predictors (used to extract the factors) only if it is statistically significant at a 5% level.

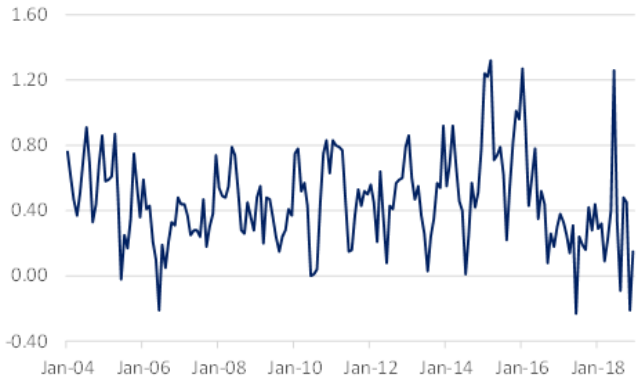
(iv) proceed as before, in the direct or iterated factor model cases, to build the respective inflation forecasts.

3 Empirical Exercise

3.1 Data

We focus the analysis on the Brazilian IPCA monthly inflation, which is a consumer price index, measured by the Brazilian Institute of Geography and Statistics (IBGE), used to compute the official inflation target. Also, the main inflation-linked federal government bond in Brazil (NTN-B) use the IPCA as their reference.[22] The sample period spans 15 years of data, from January 2004 to December 2018 (180 observations).[23] Figure 3 shows the IPCA inflation in our sample period, which starts a decade after the Brazilian monetary stabilization plan in mid-1994. Note that the inflation level is similar to the ones from other inflation-targeting emerging countries.

Figure 3 - IPCA inflation rate (% per month)



We also use a quite diverse set of macroeconomic and financial variables drawn from a number of categories.[24] Our dataset consists of $n = 120$ monthly variables, including: price indexes, interest rates, financial markets, economic activity, labor market, government debt, import and export of goods and services, and international variables that are potentially related to the Brazilian economy. The data sources are the Banco Central do Brasil, FGV, IBGE, IpeaData and Reuters. Appendix C presents the full list of variables used as potential predictors for inflation.

[22] NTN-B is the acronym for *Nota do Tesouro Nacional*, type B, which is the equivalent to the Treasury Inflation-Protected Securities (TIPS) in the U.S.

[23] According to Machado and Portugal (2014), the limited sample problem is a well-known constraint for inference in Brazilian studies, particularly in inflation dynamics where different policy regimes have been the case. In this sense, selecting the sample from 2004 to 2018 helps us avoid large structural regime breaks.

[24] In the search for models and variables to forecast and explain inflation, besides the usual macroeconomic variables, we included many financial variables, which are shown in the literature (Forni et al., 2003) to be significant predictors that can help forecasting inflation. For instance, financial market-based implied inflation (i.e., breakeven inflation rate), besides providing a closer monitoring of inflation expectations (since they can be updated on a continuously intra-day basis), are also competitive in terms of short-run predictive ability when compared to survey-based expectations (Araujo and Vicente, 2017).

All variables are automatically tested for unit-root using the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test and first-differentiated when necessary. In all five machine learning approaches, we use $s = 12$ lags of the n candidate regressors in equation (1), besides including an intercept and seasonal dummies as additional possible predictors of inflation. This way, $dim\,(\widetilde{x}_t') = 1,452$ variables.

We use data over the period from January 2004 to January 2011 ($T_1 = 85$ observations) for model estimation (*training set*) and reserve the remaining data (*test set*) for the forecast comparison using $P = T - T_1 = 95$ observations. The first part of the sample is used to estimate the econometric models and train the machine learning approaches (cross-validation and selection of the tuning parameters), whereas the remaining observations are used for out-of-sample forecast comparison for horizons $h = 1, ..., 12$ months.

All models are recursively estimated by using a growing window (increasing sample size), as we incorporate every new time-series observation, one at a time. In this context, each model is initially estimated using the first T_1 observations and the out-of-sample point forecasts are generated. We, then, add an additional observation at the end of the *training set*, re-estimate the models and generate again out-of-sample forecasts. This process is repeated along the remaining data (*test set*). See Morales-Arias and Moura (2013) for a detailed discussion about recursive versus rolling window forecasting.[25] The evaluation period for $h = 1$ ranges from February 2011 to December 2018 (95 forecasts), whereas for $h = 12$ ranges from January 2012 to December 2018 (84 forecasts).[26]

The empirical exercise is implemented using the R software (version 3.5.1, i386). The ridge regression, lasso and elastic net models are estimated using the R package *glmnet* (version 2.0-16), which fits a generalized linear model via penalized maximum likelihood. The adalasso model is implemented using the R package *HDeconometrics* (version of January 26, 2018), available at: https://github.com/gabrielrvsc/HDeconometrics. The same R package is used to compute the BIC information criterion and choose the tuning parameters λ and α. In turn, in order to implement the random forest and the quantile regression forest methods, we use the R package *ranger* (version 0.11.1).

[25] As a robustness check, we could employ a rolling window estimation (fixed sample size), which is usually more suitable in the presence of structural breaks. We leave this exercise for future extensions of this paper.

[26] To avoid extra (and unnecessary) complications in the implementation of the forecasting exercise, we refrain to do a real-time analysis. Thus, a note of caution regarding the interpretation of results applies, mainly due to two concerns: (i) not all useful predictors may be available to the forecaster in real time; and (ii) several predictors are subject to data revisions (e.g., the CPI data become available only with a one-month delay).

3.2 Results

Our empirical exercise entails the following sixteen inflation forecasting methods: random walk; random walk (Atkeson-Ohanian); ARMA; VAR; Phillips curve (backward-looking or hybrid); factor model[27] (direct or iterated forecast; with or without targeted predictors); ridge regression; lasso; elastic net; random forest; quantile regression forest;[28] and the Focus survey (included as a benchmark).[29]

Since the traditional models used to forecast inflation are vastly discussed and documented in the literature, we focus on the estimation of the machine learning approaches.

Figure 4 shows how the lasso and the elastic net works in practice: the value of $log(\lambda)$ is shown on the horizontal axis, and the vertical axis presents the regularization path, that is, the coefficient estimates β_j (each line represents a different variable).

The number of selected variables for each value of λ is presented on top of each graph in Figure 4. Note that the amount of selected variables, as well as the size of the coefficients, decrease towards zero as long as the shrinkage parameter λ augments. The vertical gray dotted line indicates the value of λ chosen from cross-validation. In the lasso and the elastic net final estimation ($h = 1$, full sample), there are only 29 and 48 non-zero coefficients, respectively, out of the total amount of $1,452$ potential predictors; thus confirming the parsimonious outcomes in both methods.[30]

Figure 5 shows how the proposed selection procedures works over time. The horizontal axis represents the end of the estimation sample, along the out-of-sample forecasting exercise, and the vertical axis denotes all the $1,452$ regressors. A blue dot indicates that variable i has a non-zero coefficient in the lasso estimation (a red dot, in the elastic net case) with sample ending at period t, used to build forecasts for y_{t+h}. In other words, Figure 5 shows that the statistical significance of the coefficients vary considerably over time for some variables, while staying relatively stable for others. Examining such coefficient estimates allows us to discover how the learned models change in response to different economic conditions over time. See the Appendix D for results with $h > 1$.[31]

[27] For $h = 1$, we extract seven factors, which jointly account for 55% of the total variance in the data.

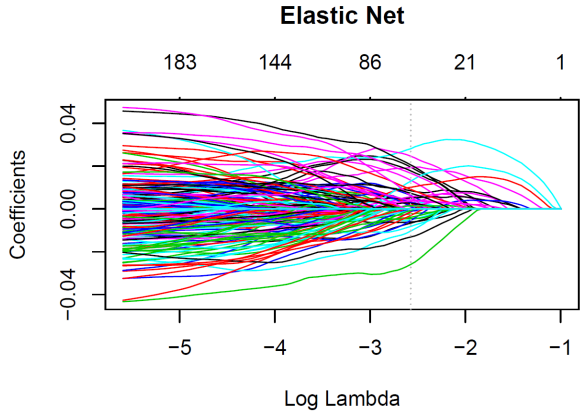
[28] We used an amount of $10,000$ trees in both the random forest and the quantile regression forest. In the latter method, we adopted the following grid of quantile levels: $\tau \in (0.05, 0.10, 0.15, ..., 0.95)$.

[29] The Focus survey is organized by the Banco Central do Brasil and started in 1999 with the implementation of the inflation-targeting regime. It contains daily forecasts from more than 100 institutions (financial or non-financial), for different horizons and a large number of economic variables. It also has a Top5 ranking contest built to improve forecasting expertise.

[30] By using regularization to control the size of the model, LASSO and elastic net can set many coefficients to zero and, this way, deliver a more parsimonious model compared to an unregularized linear model.

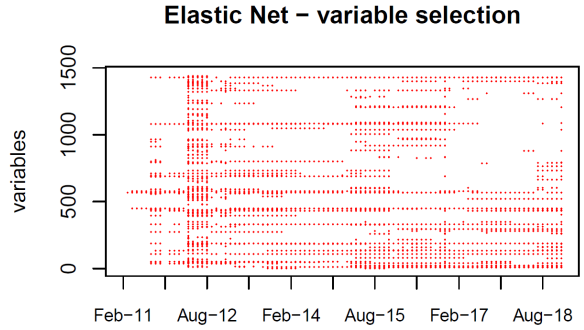
[31] Figure D2 in Appendix shows the average number of variables selected by lasso and elastic net. The elastic

Figure 4 - Elastic Net coefficients $(h = 1)$



Notes: Full sample estimation for h=1 (Jan2004-Nov2018). The vertical axis shows the estimates for each log(lambda). The number of selected variables is shown on top of each graph. The vertical dotted line indicates the choice from cross-validation.

Figure 5 - Elastic Net variable selection $(h = 1)$



Another analysis that we find interesting is the identification of which variables are chosen by the machine learning methods to predict inflation. Although we do not attempt to economically interpret the driving-forces behind the machine learning forecasts, further inspecting these models, to better understand how they are making forecasts, may reveal new statistical relationships in the data previously overlooked by standard linear models.

In this sense, Figure 6 shows the *word clouds* containing the most frequent variables selected by lasso and elastic net in all horizons. To do so, we compute the frequency of non-zero coefficients of a given variable, taking into account all out-of-sample observations and forecast horizons $h = 1, ..., 12$. Then, we rank these frequencies in order to create a list with the most frequent variables; shown in Figure 6 with a larger font size. Variables with the same frequency are depicted with the same size and color.

net selects more variables than the lasso does, probably owing to the grouping effect, as discussed in Zou and Hastie (2005). Also, the models for shorter horizons are, usually, more parsimonious.

Figure 6 - Elastic Net word clouds for all horizons

The variables identified by lasso and elastic net include mostly past inflation (probably due to the decades of inertial inflationary dynamics in Brazil); industrial production (in particular, consumer and intermediate goods, besides passenger cars and light vehicles); labor market (wholesale and retail trade); exterior sector (linked to commodity exports); and public sector (especially, public debt and primary result).

Appendix D provides additional word clouds built for selected horizons. Overall, variable selection seems to be quite different across distinct horizons, indicating that the best predictors for short-run forecasting are not very useful for medium or long-term forecasting (and vice versa). Besides, despite the usual suspects (e.g., past inflation), it is interesting to find novel predictors to help forecast the Brazilian inflation; such as the commercial electricity consumption, in the short-run forecast, and the initial public offers (IPOs), in the long-run forecast.

Figure 7 - Inflation and forecasts ($h = 1$)

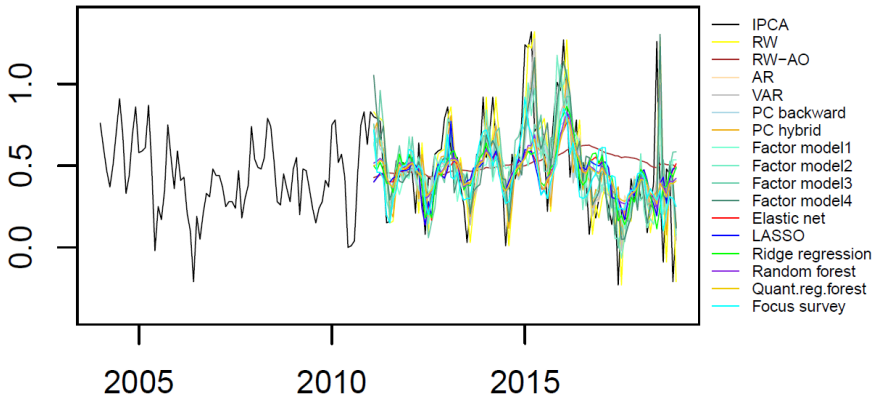
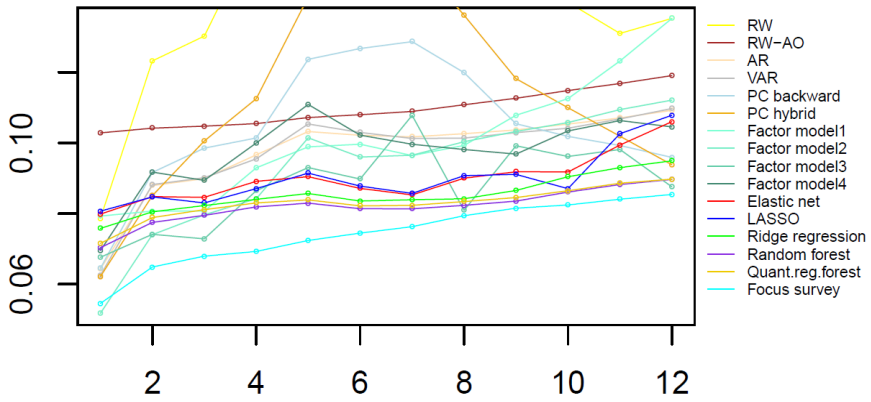


Figure 7 shows the observed inflation rate and the out-of-sample forecasts of the sixteen approaches covered in this paper for $h = 1$. The individual forecast errors for each horizon are squared and averaged to derive an overall MSE for the out-of-sample evaluation period. We also show the p-value of the Diebold and Mariano (1995) test,[32] using the Focus survey forecast

[32]The null hypothesis assumes equal forecasting accuracy of two competing forecasts. The variances entering the test statistics use here the Newey and West (1987) HAC covariance estimator.

as benchmark. Figure 8 and Table 1 present the results.

Figure 8 - Mean Squared Error (MSE)



Note: MSEs are shown in the vertical axis and the forecast horizon (in months) are in the horizontal axis.

The Focus survey is chosen as benchmark.[33] Regarding the shortest horizon ($h = 1$), the best model is the *factor model with iterated forecast*, closely followed by the Focus survey. The hybrid Phillips curve and the VAR[34] are, respectively, in the third and fourth places. According to the Diebold-Mariano test, both the factor model and the hybrid PC exhibits the same forecasting accuracy when compared to the Focus survey.

For longer horizons, the Focus survey dominates the competition, in terms of MSE, although quite often presents an equal predictive ability in comparison to the second and/or third places (factor model, random forest or quantile regression forest).

Among the five machine learning (ML) techniques, the *random forest* is the best one, in all horizons, closely followed by the quantile regression forest (QRF) and, later on, by the ridge regression. In addition, the ML forecasts showed superior predictive power and usually dominate the inflation forecasts of traditional approaches.

[33] In our exercise, we select the cross-section median (of the panel of survey forecasts) on the 15th calendar day of each month. The inflation rate of a given month is publicly released by IBGE around the 8th calendar day of the following month. This way, the Focus forecast for $h = 1$ month represents (in fact) an inflation expectation formed approximately 23 days before the realization of the target variable; which is clearly a shorter horizon when compared to the other forecasting methods considered in this paper.

[34] The VAR performs slightly better compared to the AR(1) in shorter horizons, possibly due to its enlarged information set and the choice of separately modeling the disaggregate inflation indexes (market price inflation and administered price inflation).

Table 1 - Mean Squared Error (MSE)

	forecast horizon (in months)											
Model	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
1 - Random walk	0.079**	0.123***	0.130***	0.160***	0.198***	0.196***	0.183***	0.169***	0.151***	0.140**	0.131**	0.135**
	(0.034)	(0.003)	(0.000)	(0.000)	(0.000)	(0.000)	(0.001)	(0.003)	(0.010)	(0.020)	(0.038)	(0.028)
2 - Random walk (Atkeson and Ohanian)	0.103***	0.104***	0.105***	0.105***	0.107***	0.108***	0.109***	0.111***	0.113**	0.115**	0.117**	0.119***
	0.000	(0.001)	(0.002)	(0.003)	(0.004)	(0.003)	(0.006)	(0.009)	(0.011)	(0.012)	(0.012)	(0.008)
3 - ARMA	0.064	0.088***	0.090*	0.097**	0.103**	0.102**	0.102**	0.103*	0.104*	0.105*	0.107*	0.109**
	(0.120)	(0.008)	(0.060)	(0.040)	(0.023)	(0.030)	(0.045)	(0.051)	(0.060)	(0.057)	(0.056)	(0.049)
4 - VAR	0.062*	0.088***	0.090*	0.096**	0.105**	0.103**	0.101*	0.101*	0.103*	0.104*	0.107*	0.110*
	(0.091)	(0.008)	(0.051)	(0.043)	(0.014)	(0.030)	(0.054)	(0.064)	(0.067)	(0.061)	(0.058)	(0.054)
5 - Phillips curve (backward)	0.064*	0.092***	0.099**	0.101**	0.124***	0.127***	0.129***	0.120**	0.105**	0.102***	0.099*	0.096
	(0.070)	(0.004)	(0.031)	(0.019)	(0.006)	(0.008)	(0.008)	(0.015)	(0.032)	(0.006)	(0.068)	(0.174)
6 - Phillips curve (hybrid)	0.062	0.085**	0.101**	0.113**	0.141**	0.150**	0.153***	0.136**	0.118***	0.110***	0.102*	0.094
	(0.123)	(0.023)	(0.029)	(0.014)	(0.014)	(0.015)	(0.008)	(0.011)	(0.008)	(0.008)	(0.067)	(0.252)
7 - Factor model (direct forecast)	0.079***	0.081**	0.081**	0.093*	0.099*	0.100*	0.096*	0.099**	0.108**	0.113**	0.123**	0.135***
	(0.007)	(0.045)	(0.040)	(0.071)	(0.100)	(0.087)	(0.075)	(0.034)	(0.026)	(0.036)	(0.040)	(0.005)
8 - Factor model (iterated forecast)	0.052	0.074	0.080*	0.084**	0.101***	0.096*	0.097	0.100*	0.103*	0.106*	0.109**	0.112**
	(0.645)	(0.189)	(0.084)	(0.014)	(0.003)	(0.079)	(0.119)	(0.095)	(0.076)	(0.054)	(0.045)	(0.050)
9 - Factor model (direct forecast, targeted)	0.068*	0.074	0.073	0.086***	0.093*	0.090**	0.108**	0.081	0.099**	0.096	0.098	0.088
	(0.085)	(0.155)	(0.250)	(0.009)	(0.083)	(0.031)	(0.022)	(0.401)	(0.048)	(0.115)	(0.168)	(0.379)
10 - Factor model (iterated forecast, targeted)	0.07	0.092*	0.089**	0.100***	0.111***	0.102**	0.100**	0.098*	0.097	0.103*	0.106**	0.105*
	(0.127)	(0.074)	(0.040)	(0.007)	(0.002)	(0.018)	(0.047)	(0.086)	(0.125)	(0.060)	(0.039)	(0.094)
11 - Elastic net	0.080***	0.085***	0.085***	0.089***	0.090***	0.087**	0.085*	0.090**	0.092*	0.092*	0.099**	0.106***
	(0.001)	(0.001)	(0.004)	(0.005)	(0.003)	(0.017)	(0.052)	(0.045)	(0.094)	(0.073)	(0.012)	(0.008)
12 - LASSO	0.081***	0.085***	0.083***	0.087***	0.091***	0.088**	0.086**	0.091*	0.091	0.087	0.103***	0.108***
	(0.001)	(0.001)	(0.005)	(0.003)	(0.002)	(0.014)	(0.050)	(0.059)	(0.143)	(0.209)	(0.007)	(0.005)
13 - Ridge regression	0.076**	0.080**	0.082**	0.084**	0.086**	0.084*	0.084	0.084	0.087	0.09	0.093*	0.095
	(0.014)	(0.033)	(0.044)	(0.045)	(0.050)	(0.078)	(0.113)	(0.190)	(0.197)	(0.140)	(0.092)	(0.106)
14 - Random forest	0.070**	0.078**	0.079*	0.082**	0.083*	0.081*	0.081	0.082	0.084	0.086	0.088	0.09
	(0.014)	(0.023)	(0.055)	(0.049)	(0.064)	(0.098)	(0.161)	(0.282)	(0.346)	(0.253)	(0.231)	(0.238)
15 - Quantile regression forest	0.072**	0.079**	0.081*	0.083**	0.084*	0.082*	0.082	0.083	0.085	0.086	0.089	0.09
	(0.015)	(0.030)	(0.053)	(0.047)	(0.062)	(0.099)	(0.143)	(0.238)	(0.306)	(0.261)	(0.238)	(0.259)
16 - Focus survey	0.055	0.065	0.068	0.069	0.072	0.074	0.076	0.079	0.082	0.082	0.084	0.085
	-	-	-	-	-	-	-	-	-	-	-	-
Number of observations	95	94	93	92	91	90	89	88	87	86	85	84

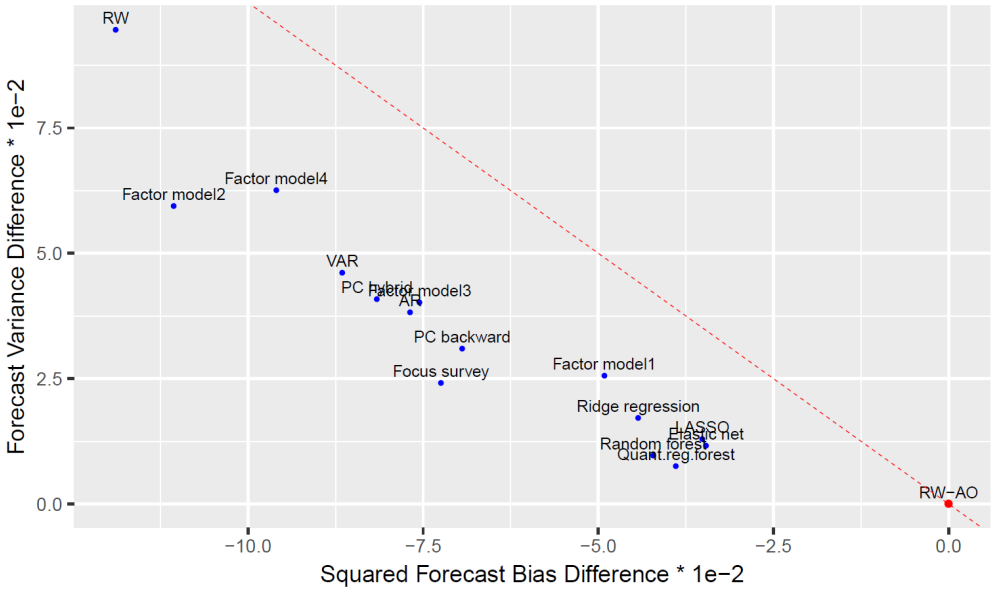
Notes: Full sample from January 2004 to December 2018 (180 observations). Shaded cells indicate the Top3 models (lowest MSEs) for each horizon. The p-value of the Diebold and Mariano (1995) test is shown in parentheses, considering the Focus survey as the benchmark. The null hypothesis assumes equal predictive ability. *, **, and *** indicate rejection of the null at 10%, 5% and 1% levels, respectively. Last line shows the number of out-of-sample observations used to compute the MSEs.

Next, we investigate the trade-off between variance and bias of each forecasting method. Following Lima and Meng (2017), we decompose the MSE into two parts: the forecast variance and the squared forecast bias. To do so, we calculate the MSE of any forecast $\widehat{f_{y_{t+h}}}$ as $\frac{1}{T^*}\sum_t \left(y_{t+h} - \widehat{f_{y_{t+h}}}\right)^2$ and the unconditional forecast variance as $\frac{1}{T^*}\sum_t \left(\widehat{f_{y_{t+h}}} - \frac{1}{T^*}\sum_t \widehat{f_{y_{t+h}}}\right)^2$, where T^* is the total number of out-of-sample forecasts. The squared forecast bias is, then, computed as the difference between MSE and forecast variance.

Figure 9 shows the relative forecast variance and squared forecast bias of all forecasting methods. The relative forecast variance (squared bias) is calculated as the difference between the forecast variance (squared bias) of the $i-th$ model and the forecast variance (squared

bias) of the moving-average approach RW-AO. Thus, the value of relative forecast variance (squared bias) for the RW-AO is necessarily equal to zero. Moreover, each point on the red dotted line represents a forecast with the same MSE as the RW-AO. Points to the right of the line are forecasts outperformed by the RW-AO, and points to the left represent forecasts that outperform the RW-AO. Since the RW-AO is a simple moving average of inflation, it will have a very low variance but will likely be biased.

Figure 9 - Scatterplot of relative forecast variance and squared forecast bias ($h = 1$)



Notes: The y-axis and x-axis represent relative forecast variance and squared forecast bias, computed as the difference between the forecast variance (squared bias) of the considered approach and the forecast variance (squared bias) of the RW-AO. Each point on the red dotted line represents a forecast with the same MSE as the RW-AO; points to the right are forecasts outperformed by the RW-AO, and points to the left represent forecasts that outperform the RW-AO.

Note that for $h = 1$ all forecasts outperformed the RW-AO. Combining this result with the empirical observation that the variances of forecasts are not lower than the variance of the RW-AO (i.e., all the blue dots fall above the horizontal zero line), we conclude that such performance relies almost exclusively on a predictor's ability to lower forecast bias relative to that of RW-AO.

Overall, the success of the factor model 2 in the short-run can be explained by its ability to substantially reduce the forecast bias at the expense of a moderate increase in forecast variance. This trade-off delivered the lowest MSE for $h = 1$. On the other hand, the relatively weak performance of lasso and elastic net, among the ML methods, seems to be mainly driven by a substantial squared bias.

The main message is that the forecasting methods that yield a sizeable reduction in the forecast bias, while keeping variance under control, are able to improve forecasting accuracy over the lowest-variance approach (RW-AO). This explains the superior performance of the one-month ahead forecasts of the factor model 2.

The MSE decomposition for other horizons are presented in the Appendix E. Note that the random forest forecasts achieve a middle ground in terms of variance versus bias whereas the other methods that reduce forecast variance significantly are unable to lower bias by a large extent. For this reason, excepting the Focus survey, the random forest forecast often outperforms the other existing forecasting methods by meaningful margins.

4 Conclusions

The purpose of this article is to study the inflation forecast accuracy of sixteen competing methods; including traditional econometric models (ARMA, VAR), reduced-form structural models (Phillips curve), factor models, survey-based forecasts, regularization procedures (ridge, lasso and elastic net) and more recent machine learning techniques (random forest and quantile regression forest). The variable of interest is the Brazilian inflation as measured by the IPCA. In order to evaluate the predictive power of each method, we conduct a truly out-of-sample empirical exercise, where each method produces point forecasts for horizons $h = 1, ..., 12$ months ahead.

The results indicate that some machine learning algorithms are able to consistently outperform traditional econometric models in terms of MSE, thereby offering a relevant addition to the field of economic forecasting. Thus, the non-linear machine learning algorithms, applied here to solve an economic forecasting problem, can offer a valuable contribution to usual statistical models, quite often based on a linear approach. According to Hall (2018), the key to this result is to control the model complexity by using an algorithm that yields a model complex enough to avoid underfitting the data but not so complex as to overfit it.

The findings documented in this paper represent a valuable input to policymakers, academics and practitioners interested in better forecasting inflation in Brazil and, more broadly, improving the ability of macroeconomic models to fit the Brazilian data.

Possible extensions of this paper include: (i) adding other machine learning and artificial intelligence methods in the set of forecasting methods (e.g., neural network or ensemble learning); (ii) forecast combination techniques, such as the OLS regression of Granger and Ramanathan

(1984), the consensus regression of Capistrán and Timmermann (2009), the bias-corrected average forecasts of Issler and Lima (2009) and Gaglianone and Issler (2015); and (iii) disaggregate forecasting, by separately modeling selected inflation components (e.g., administered price inflation, tradables and non-tradables).

References

[1] Altmann, A., Tolosi, L., Sander, O., Lengauer, T., 2010. Permutation importance: a corrected feature importance measure. *Bioinformatics* 26, 1340-1347.

[2] Ang, A., Bekaert, G., Wei, M., 2007. Do Macro Variables, Asset Markets or Surveys Forecast Inflation Better? *Journal of Monetary Economics* 54, 1163-1212.

[3] Apostol, T.M., 1967. *Calculus, Vol. 1: One-Variable Calculus with an Introduction to Linear Algebra* (2nd Ed.), New York: John Wiley & Sons.

[4] Araujo, G.S., Vicente, J.V.M., 2017. Estimação da Inflação Implícita de Curto Prazo. Working Paper n. 460, Central Bank of Brazil.

[5] 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.

[6] Atkeson, A., Ohanian, L., 2001. Are Phillips curves useful for forecasting inflation? *Federal Reserve Bank of Minneapolis Quarterly Review* 25, 2-11.

[7] Bai, J., Ng, S., 2002. Determining the number of factors in approximate factor models. *Econometrica* 70, 191-221.

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

[9] Bańbura, M., Giannone, D., Modugno, M., Reichlin, L., 2013. Now-casting and the real-time data flow. Working Paper Series n.1564, European Central Bank.

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

[11] Cheng, K., Huang, N., Shi, Z., 2019. Survey-Based Forecasting: To Average or Not To Average. Mimeo. Available at: https://www.researchgate.net/publication/336141912

[12] Clark, T.E., West, K.D., 2007. Approximately Normal Tests for Equal Predictive Accuracy in Nested Models. *Journal of Econometrics* 138, 291-311.

[13] Diebold, F.X., Mariano, R.S., 1995. Comparing Predictive Accuracy. *Journal of Business and Economic Statistics* 13, 253-263.

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

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

[16] Faust, J., Wright, J.H., 2013. Forecasting Inflation. Handbook of Economic Forecasting, vol. 2A, Chapter 1, 3-56. Ed. Elsevier B.V.

[17] Forni, M., Hallim, M., Lippi, M., Reichlin, L., 2000. The generalized dynamic factor model: Identification and estimation. *Review of Economics and Statistics* 82, 540-554.

[18] Forni, M., Hallin, M., Lippi, M., Reichlin, L., 2003. Do financial variables help forecasting inflation and real activity in the euro area? *Journal of Monetary Economics* 50, 1243-1255.

[19] Friedman, J., R. Tibshirani, and T. Hastie., 2010. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* 33(1), 1-22.

[20] Gaglianone, W.P., Guillén, O.T.C., Figueiredo, F.M.R., 2018. Estimating Inflation Persistence by Quantile Autoregression with Quantile-Specific Unit Roots. *Economic Modelling* 73(C), 407-430.

[21] Gaglianone, W.P., Issler, J.V., Matos, S.M., 2017. Applying a Microfounded-Forecasting Approach to Predict Brazilian Inflation. *Empirical Economics* 53(1), 137-163.

[22] Garcia, M.G.P., Medeiros, M.C., Vasconcelos, G.F.R., 2017. Real-time inflation forecasting with high-dimensional models: The case of Brazil. *International Journal of Forecasting* 33, 679-693.

[23] Granger, C.W.J., Ramanathan, R., 1984. Improved methods of combining forecasting. *Journal of Forecasting* 3, 197-204.

[24] Greene, W. H. 2003. Econometric Analysis. Fifth edition. Upper Saddle River, United States: Prentice Hall.

[25] Hall, A.S., 2018. Machine Learning Approaches to Macroeconomic Forecasting. *Federal Reserve Bank of Kansas City Economic Review*, 4th quarter of 2018, 63-81.

[26] Hansen, B.E., 2019. Econometrics. Current manuscript, https://www.ssc.wisc.edu/~bhansen/econometrics/Econometrics.pdf

[27] Hastie, T., Tibshirani, R., Friedman, J., 2009. The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd edition. Springer-Verlag, New York.

[28] Hoerl, A., Kennard, R., 1988. Ridge regression. In Encyclopedia of Statistical Sciences, vol. 8, 129-136. New York: Wiley.

[29] IMF - International Monetary Fund, 2018. World Economic Outlook: Challenges to Steady Growth. Statistical Appendix, October 2018.

[30] Janitza, S., Celik, E., Boulesteix, A.-L., 2018. A computationally fast variable importance test for random forests for high-dimensional data. *Advances in Data Analysis and Classification* 12(4), 885-915.

[31] Judge, G.G., Hill, R.C., Griffiths, W.E., Lütkepohl, H., Lee, T.-C., 1988. *Introduction to the Theory and Practice of Econometrics*. New York, Wiley.

[32] Jung, J.K., Patnam, M., Ter-Martirosyan, A., 2018. An Algorithmic Crystal Ball: Forecasts-based on Machine Learning. IMF Working Paper WP/18/230.

[33] Koenker, R., 2005. *Quantile Regression*. Cambridge University Press.

[34] Kohlscheen, E., 2012. Uma nota sobre erros de previsão da inflação de curto prazo. *Revista Brasileira de Economia* 66, 289-297.

[35] Lima, L.R., Meng, F., 2017. Out-of-sample return predictability: a quantile combination approach. *Journal of Applied Econometrics* 32(4), 877-895.

[36] Lütkepohl, H. 2005. New Introduction to Multiple Time Series Analysis. Springer-Verlag.

[37] Machado, V., Portugal, M., 2014. Measuring inflation persistence in Brazil using a multivariate model. *Revista Brasileira de Economia* 68 (2), 225-241.

[38] Marcellino, M., Stock, J., Watson, M., 2006. A comparison of direct and iterated multistep AR methods for forecasting macroeconomic time series. *Journal of Econometrics* 135, 499-526.

[39] Medeiros, M., Mendes, E., 2016. L1-regularization of high-dimensional time-series models with flexible innovations. *Journal of Econometrics* 191, 255-271.

[40] Medeiros, M., Vasconcelos, G.F.R., de Freitas, E.H., 2016. Forecasting Brazilian Inflation with High Dimensional Models. *Brazilian Review of Econometrics* 36(2), 223-254.

[41] Meinshausen, N., 2006. Quantile Regression Forests. *Journal of Machine Learning Research* 7, 983-999.

[42] Minella, A., de Freitas, P.S., Goldfajn, I., Muinhos, M.K., 2003. Inflation targeting in Brazil: constructing credibility under exchange rate volatility. *Journal of International Money and Finance* 22 (7), 1015-1040.

[43] Morales-Arias, L., Moura, G.V., 2013. Adaptive forecasting of exchange rates with panel data. *International Journal of Forecasting* 29, 493-509.

[44] Nembrini, S., Koenig, I.R., Wright, M.N., 2018. The revival of the Gini Importance? *Bioinformatics* 34(21), 3711-3718.

[45] Nowotarski, J., Raviv, E., Trück, S., Weron, R., 2014. An empirical comparison of alternative schemes for combining electricity spot price forecasts. *Energy Economics* 46, 395-412.

[46] Stock, J., Watson, M., 1999. Forecasting inflation. *Journal of Monetary Economics* 44, 293-335.

[47] Stock, J., Watson, M., 2002. Forecasting Using Principal Components from a Large Number of Predictors. *Journal of the American Statistical Association* 97(460), 1167-1179.

[48] Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society* 58(1), 267-288.

[49] Varian, H.R., 2014. Big data: New tricks for econometrics. *Journal of Economic Perspectives* 28(2), 3-28.

[50] Zou, H., 2006. The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association* 101 (476), 1418-1429.

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

[52] Zou, H., Hastie, T., Tibshirani, R., 2007. On the degrees of freedom of the lasso. *The Annals of Statistics* 35, 2173-2192.

Appendix A: Further details on *random forest*

In this section, we represent mathematically the random forest model, following the discussion in Meinshausen (2006): consider n independent observations (Y_i, X_i), for $i = 1, ..., n$, and let θ be the random parameter vector that determines how a tree $T(\theta)$ is grown, that is, characterizes the tree in terms of split variables, cut-points at each node, and terminal-node values. Also, let $\Im$ be the space in which X lives, that is $X : \Omega \to \Im$, where $\Im \subseteq \mathbb{R}^p$ and $p \in \mathbb{N}_+$ is the dimensionality of the set of covariates X.

Every leaf of a tree (terminal node) $l = 1, ..., L$ corresponds to a subspace of $\Im$, that is $R_l \subseteq \Im$. For every $x \in \Im$, there is one (and only one) leaf l such that $x \in R_l$ (corresponding to the leaf that is obtained when dropping x down the tree). Denote this leaf by $l(x, \theta)$ for tree $T(\theta)$. The prediction of a single tree $T(\theta)$ conditioned on $X = x$ is obtained by averaging over the observed values in leaf $l(x, \theta)$. Let the weight vector $w_i(x, \theta)$ be given by a positive constant if observation X_i is part of leaf $l(x, \theta)$ and 0 if it is not. The weights sum to one, such that:

$$w_i(x, \theta) = \frac{\mathbf{1}_{\{X_i \in R_{l(x,\theta)}\}}}{\sum_{j=1}^{n} \mathbf{1}_{\{X_j \in R_{l(x,\theta)}\}}}. \tag{33}$$

The forecasting model based on a single *regression tree*, conditioned on a covariate $X = x$, is then the weighted average of the original observations Y_i, for all $i = 1, ..., n$, that is:

$$E_{\text{regression tree}}(Y \mid X = x) = \sum_{i=1}^{n} w_i(x, \theta) Y_i. \tag{34}$$

Note that conditional on the knowledge of the subregions R_l, for $l = 1, ..., L$, the relationship between inflation Y and the set of covariates X in equation (1) is approximated here by a piecewise constant model, where each leaf represents a distinct regime (see Garcia et al., 2017).

Now, using *random forests*, the conditional mean above is approximated by the averaged prediction of K single trees, each constructed with a parameter vector θ_k, $k = 1, ..., K$. Let $w_i(x)$ be the average of $w_i(x, \theta_k)$ over this collection of trees, as follows:

$$w_i(x) = \frac{1}{K} \sum_{k=1}^{K} w_i(x, \theta_k). \tag{35}$$

The prediction of random forests is, thus, the averaged response of all trees, as follows:

$$E_{\text{random forest}}(Y \mid X = x) = \sum_{i=1}^{n} w_i(x) Y_i. \tag{36}$$

Note that the approximation of the conditional mean of Y given $X = x$ is given by a weighted sum over all observations. The weights vary with the covariate and tend to be large for those observations $i \in \{1, ..., n\}$ where the conditional distribution of Y, given $X = X_i$, is similar to the conditional distribution of Y given $X = x$.

Random Forest and Variable Importance

Random forests are among the most popular machine learning methods due to their relatively good forecasting accuracy, robustness and ease of use. In contrast to parametric methods, random forests are fully non-parametric and can deal with nonlinear effects, thus offering

a great model flexibility in practical applications. Furthermore, RF can even be applied in the statistically challenging setting in which the number of variables is higher than the number of observations. This makes random forests especially attractive for complex high-dimensional data applications; see Janitza et al. (2018).

Nonetheless, a suitable understanding of the *black box* mechanism behind the random forest method is of greatest importance. Nowadays, machine-learning models are often deployed to production without a proper understanding of why exactly the algorithms make the decisions they do. As these new tools become more relevant in everyday life, model interpretability becomes one of the most important problems in machine learning these days. In particular, regarding the use of RF as a forecasting device, it is critical to comprehend the key variable interactions that are providing the predictive accuracy.

One attempt to tackle this issue is to compute the so-called "variable importance measures", by attributing scores to the variables, which reflect their relative importance in the overall model accuracy. Such measures can be used to identify relevant features, perform variable selection and quantify the prediction strength of each variable, allowing one to rank the variables according to their predictive abilities. See Hastie et al. (2009, chapter 15) for further details.[35]

In this paper, we provide a global insight into the random forest's behavior by computing two variable importance measures, based on the "permutation" approach of Altmann et al. (2010) and on the "impurity-corrected" method of Nembrini et al. (2018). Moreover, we carry out the Janitza et al. (2018) hypothesis test of no association between the predictor and the dependent variable for both measures. As result, for each forecast horizon, we build variable importance graphs (showing the top 20 most important variables) as well as word clouds for random forests.

The permutation method, also known as the mean decrease in accuracy, is one of the most common variable importance measures, and it is computed from the change in prediction accuracy when removing any association between the dependent variable (response) and a given regressor (i.e., feature or predictor), with large changes indicating that the predictor is important.[36] One disadvantage of the permutation approach is to produce biased outcomes when predictors are highly correlated. In addition, adding a correlated variable to the RF model can decrease the importance of another variable. Furthermore, the permutation importance is very computationally intensive in the case of high dimensional data.

Alternative importance measures based on impurity (i.e., how well the regression trees split the variables) are popular because they are simple, fast to compute and can be more robust to data perturbations compared with those based on permutation.[37] However, the impurity importance is known to be biased towards variables with more categories or more possible split

[35] There are many other ways on the lookout for opening the ML black box. Just to mention a few examples: (i) Partial Dependence Plots (PDP), which show the marginal effect of a given predictor on the outcome of a ML model; and (ii) Surrogate Models (SM), which are auxiliary interpretable models (e.g., linear regression), built to approximate the predictions of a ML model in order to understand the black box outcomes by analyzing (and interpreting) the surrogate model's responses.

[36] According to Nembrini et al. (2018): *"To calculate the permutation importance of the variable xi, its original association with the response y is broken by randomly permuting the values of all individuals for xi. With this permuted data, the tree-wise out-of-bag (OOB) estimate of the prediction error is computed. The difference between this estimate and the OOB error without permutation, averaged over all trees, is the permutation importance of the variable xi. This procedures is repeated for all variables of interest x1,. . . ,xp. The larger the permutation importance of a variable, the more relevant the variable is for the overall prediction accuracy."*

[37] Recall that random forest consists of a number of decision trees. Every node in the trees is a condition on a given variable, and it is designed to optimally split the dataset into two parts so that overall model accuracy can be improved. The measure based on which the (locally) optimal condition is chosen is called impurity (or variance, in the case of the regression trees). This way, one can compute how much each variable reduces the weighted impurity in a tree. For a forest, the impurity reduction from each variable can be averaged and a ranking of variables can be constructed according to this importance measure.

points. Also, when the dataset has two (or more) correlated variables, any of them can be selected as predictor. Nevertheless, once one of these (correlated) variables is used as predictor, the importance of others is significantly reduced, since the impurity these other variables can decrease is already reduced by the first selected variable.[38] In this sense, Nembrini et al.(2018) propose the "corrected impurity" importance measure, which is unbiased in terms of the number of categories and category frequencies and is computationally efficient (i.e., almost as fast as the standard impurity importance and much faster than the permutation importance).

Besides building a ranking of importance, it is also crucial to statistically check whether a given predictor is important (or not) in respect to the depend variable of the RF model. According to Janitza et al. (2018), the variable importance depends on many different factors, including aspects related to the data (e.g., correlations, signal-to-noise ratio or the total number of variables) as well as on the random forest specific factors (such as the choice of the number of randomly drawn candidate predictor variables for each split node). Therefore, there is no universally applicable threshold that can be used to statistically discriminate between important and non-important variables. Nonetheless, several hypothesis-testing approaches have been developed. The permutation-based tests entail the repeated computation of random forests. While for low-dimensional settings those approaches might be computationally tractable, for high-dimensional models (e.g., including thousands of predictors), computing time might become enormous. In this sense, Janitza et al. (2018) propose a variable importance test that is appropriate for high-dimensional data where many variables do not carry any information related to the dependent variable. According to the authors, the testing approach, based on cross-validation procedures, shows at least comparable power at a substantially smaller computation time.

Appendix B: Further details on *quantile regression forest*

The QRF algorithm, proposed by Meinshausen (2006), for computing the estimate of the conditional distribution function, can be summarized as follows:

(a) grow trees $T(\theta_k)$, for $k = 1, ..., K$, as in random forests. However, for every leaf (on each tree) consider all observations in the leaf, not just their average.

(b) for a given $X = x$, drop x down all trees. Compute the weight $w_i(x, \theta_k)$ of observation $i \in \{1, ..., n\}$ for every tree as in (33). Compute weight $w_i(x)$ for every observation $i \in \{1, ..., n\}$ as an average over $w_i(x, \theta_k)$, for all $k = 1, ..., K$, as in (35).

(c) compute the estimate of the distribution function as in (16) for all $y \in \mathbb{R}$, using the weights from the previous step (b).

[38] This is not an issue in respect to model forecasting, but regarding model interpretation, it can lead to the incorrect conclusion that one of the variables is a strong predictor while the others (correlated variables) are not important, while, in reality, they are all close in respect to their statistical relationship with the dependent variable. This effect can be attenuated by using random variable selection at each node (instead of using all possible variables) when growing a tree within the random forest setup.

Appendix C: Data

Table C1 - List of macroeconomic and financial variables

Series	Category	Name	Source	Original
1	Inflation	IPCA (consumer price index)	IBGE	%p.m.
2	Inflation	IPCA (consumer price index, market prices)	IBGE	%p.m.
3	Inflation	IPCA (consumer price index, regulated and monitored prices)	IBGE	%p.m.
4	Inflation	IPCA (consumer price index, tradables)	BCB	%p.m.
5	Inflation	IPCA (consumer price index, nontradables)	BCB	%p.m.
6	Inflation	IPC-Fipe (consumer price index)	Fipe	%p.m.
7	Inflation	IPC-Br (consumer price index)	FGV	%p.m.
8	Inflation	IPA-DI (wholesale price index)	FGV	%p.m.
9	Inflation	IGP-DI (general price index)	FGV	%p.m.
10	Inflation	IGP-M (general price index)	FGV	%p.m.
11	Inflation	IGP-10 (general price index)	FGV	%p.m.
12	Inflation	INCC (national index of building costs)	FGV	%p.m.
13	Inflation	Core IPC-Br (core inflation)	FGV	%p.m.
14	Inflation	Core IPCA - Exclusion EX0 (core inflation)	BCB	%p.m.
15	Inflation	Core IPCA - Exclusion EX1 (core inflation)	BCB	%p.m.
16	Inflation	Core IPCA - Double Weight (core inflation)	BCB	%p.m.
17	Inflation	Core IPCA - Trimmed Means Smoothed (core inflation)	BCB	%p.m.
18	Inflation	Break Even Inflation (IPCA, 1 year)	Anbima	%p.a.
19	Inflation	Break Even Inflation (IPCA, 2 years)	Anbima	%p.a.
20	Inflation	Break Even Inflation (IPCA, 5 years)	Anbima	%p.a.
21	Interest rates	Nominal policy interest rate (Selic)	BCB	%p.a.
22	Interest rates	Nominal policy interest rate (long-term interest rate, TJLP)	BCB	%p.a.
23	Interest rates	Nominal market interest rate (prefixed, 1 year)	Anbima	%p.a.
24	Interest rates	Nominal market interest rate (prefixed, 2 years)	Anbima	%p.a.
25	Interest rates	Nominal market interest rate (prefixed, 5 years)	Anbima	%p.a.
26	Interest rates	Nominal market interest rate (Swap Pré-DI, 1 year)	Reuters	%p.a.
27	Interest rates	Real market interest rate (Swap Pré-DI, 1 year, deflator: Focus 12m infl.expect.)	Reuters, BCB	%p.a.
28	Interest rates	Real market interest rate (indexed IPCA, 1 year)	Anbima	%p.a.
29	Interest rates	Real market interest rate (indexed IPCA, 2 years)	Anbima	%p.a.
30	Interest rates	Real market interest rate (indexed IPCA, 5 years)	Anbima	%p.a.
31	Money	Monetary base	BCB	R$ thousand
32	Money	Money supply (currency outside banks)	BCB	R$ thousand
33	Money	Demand deposits	BCB	R$ thousand
34	Money	Savings deposits	BCB	R$ thousand
35	Money	M1	BCB	R$ thousand
36	Money	M2	BCB	R$ thousand
37	Money	M3	BCB	R$ thousand
38	Money	M4	BCB	R$ thousand
39	Banking sector	Credit spread (nonearmarked credit rate - Selic rate)	BCB	basis points
40	Banking sector	Non-Performing Loans (NPL) of total credit	BCB	%
41	Banking sector	Loan-to-Deposit ratio (LTD)	BCB	Units
42	Banking sector	Reserve requirements ratio (financial inst. reserve requirements / total deposits)	BCB	Units
43	Banking sector	Real growth of nonearmarked credit operations outstanding	BCB	R$ million
44	Capital markets	Initial Public Offers (IPOs) accumulated in 12 months (Brazil)	BCB	R$ million
45	Capital markets	Net equity of stock funds (Brazil)	BCB	R$ million
46	Capital markets	Net equity of financial investment funds (Brazil)	BCB	R$ million
47	Capital markets	Ibovespa (Brazil)	Reuters	Index
48	Capital markets	MSCI emerging countries (EM, US$)	Reuters	Index
49	Capital markets	MSCI developed countries (World, US$)	Reuters	Index
50	Risk premium	Embi+Br (Emerging Markets Bond Index Plus Brazil, spread)	Reuters	basis points
51	Risk premium	Embi+composite (average spread of 16 emerging countries)	Reuters	basis points
52	Risk premium	CDS (Credit Default Swap, Brazil 5 years)	Reuters	basis points
53	Exchange rates	FX-rate (nominal exchange rate, R$/US$)	BCB	Units
54	Exchange rates	REER (Real effective exchange rate, IPA-13 currencies)	Funcex	Index
55	Global Economy	U.S. dollar index (DXY, geometric average of 6 currencies in respect to US$)	Reuters	Index
56	Global Economy	U.S. Treasury 2 years (Treasury nominal interest rates)	Reuters	%p.a.
57	Global Economy	U.S. Treasury 10 years (Treasury nominal interest rates)	Reuters	%p.a.
58	Global Economy	U.S. Treasury 5 years TIPS (Treasury Inflation-Protected Securities)	Reuters	%p.a.
59	Global Economy	CRB all commodities index	Reuters	Index
60	Global Economy	Oil price (WTI, Oklahoma-USA)	Reuters	US$/barrel
61	Global Economy	VIX CBOE volatility index (30-day expected volatility of the S&P500)	Reuters	Index

Table C1 - List of macroeconomic and financial variables (cont.)

Series	Category	Name	Source	Original
62	Exterior	Import price index	Funcex	Index
63	Exterior	Import quantum index	Funcex	Index
64	Exterior	Export price index	Funcex	Index
65	Exterior	Export quantum index	Funcex	Index
66	Exterior	Imports (FOB, total)	MDIC/Secex	US$
67	Exterior	Exports (FOB, total)	MDIC/Secex	US$
68	Exterior	Exports (FOB, primary goods)	MDIC/Secex	US$
69	Exterior	International reserves (total)	BCB	US$ million
70	Exterior	Current account (monthly, net)	BCB	US$ million
71	Exterior	Current account (accumulated in 12 months, in relation to GDP)	BCB	%
72	Exterior	FDI (Foreign Direct Investment, accumulated in 12 months)	BCB	US$ million
73	Exterior	FPI (Foreign Portfolio Investment, accumulated in 12 months)	BCB	US$ million
74	Economic activity	IBC-BR (central bank economic activity index)	BCB	Index
75	Economic activity	GDP (accumulated in the last 12 months, current prices)	BCB	R$ million
76	Economic activity	Consumer confidence index	Fecomercio	Index
77	Labor	Unemployment rate (open)	IBGE	%
78	Labor	Registered employees index (wholesale and retail trade)	MTE	Index
79	Labor	Registered employees index (construction sector)	MTE	Index
80	Labor	Hours worked in production (São Paulo)	Fiesp	Index
81	Labor	Real overall wages (industry, São Paulo)	Fiesp	Index
82	Industry	Industrial production (total)	IBGE	Index
83	Industry	Industrial production (mineral extraction)	IBGE	Index
84	Industry	Industrial production (manufacturing industry)	IBGE	Index
85	Industry	Industrial production (capital goods)	IBGE	Index
86	Industry	Industrial production (intermediate goods)	IBGE	Index
87	Industry	Industrial production (consumer goods)	IBGE	Index
88	Industry	Industrial production (durable goods)	IBGE	Index
89	Industry	Industrial production (semidurable and nondurable goods)	IBGE	Index
90	Industry	Installed capacity utilization (São Paulo)	Fiesp	%
91	Industry	Capacity utilization (manufacturing industry, FGV)	FGV	%
92	Industry	Steel production	BCB	Index
93	Industry	Vehicles production (total)	Anfavea	Units
94	Industry	Passenger cars and light commercial vehicles production	Anfavea	Units
95	Industry	Truck production	Anfavea	Units
96	Industry	Bus production	Anfavea	Units
97	Industry	Production of agricultural machinery (total)	Anfavea	Units
98	Sales	Sales volume index in the retail sector (total)	IBGE	Index
99	Sales	Sales volume index in the retail sector (fuel and lubricants)	IBGE	Index
100	Sales	Sales volume index in the retail sector (hyperm., superm., food, bever. and tobacco)	IBGE	Index
101	Sales	Sales volume index in the retail sector (textiles, clothing and footwear)	IBGE	Index
102	Sales	Sales volume index in the retail sector (furniture and white goods)	IBGE	Index
103	Sales	Sales volume index in the retail sector (vehicles and motorcycles, spare parts)	IBGE	Index
104	Sales	Sales volume index in the retail sector (hyperm. and supermarkets)	IBGE	Index
105	Sales	Vehicle sales (total)	Anfavea	Units
106	Sales	Domestic vehicle sales	Anfavea	Units
107	Energy	Electric energy consumption (commercial)	Eletrobras	GWh
108	Energy	Electric energy consumption (residential)	Eletrobras	GWh
109	Energy	Electric energy consumption (industrial)	Eletrobras	GWh
110	Energy	Electric energy consumption (other)	Eletrobras	GWh
111	Energy	Electric energy consumption (total)	Eletrobras	GWh
112	Public sector	Primary result of consolidated public sector (current monthly flows)	BCB	R$ (million)
113	Public sector	Primary result of consolidated public sector (flows accum. in 12 months)	BCB	R$ (million)
114	Public sector	Primary result of consolidated public sector (flow accum. in 12 months, % GDP)	BCB	%
115	Public sector	Net public debt (total, federal government and central bank, % GDP)	BCB	%
116	Public sector	Net public debt (internal, federal government and central bank, % GDP)	BCB	%
117	Public sector	Net public debt (external, federal government and central bank, % GDP)	BCB	%
118	Public sector	Net public debt (total, consolidated public sector, balances in reais)	BCB	R$ (million)
119	Public sector	Net public debt (internal, consolidated public sector, balances in reais)	BCB	R$ (million)
120	Public sector	Net public debt (external, consolidated public sector, balances in reais)	BCB	R$ (million)

Appendix D: Results for Lasso and Elastic Net

Figure D1 - Lasso and Elastic Net variable selection $(h = 3, 12)$

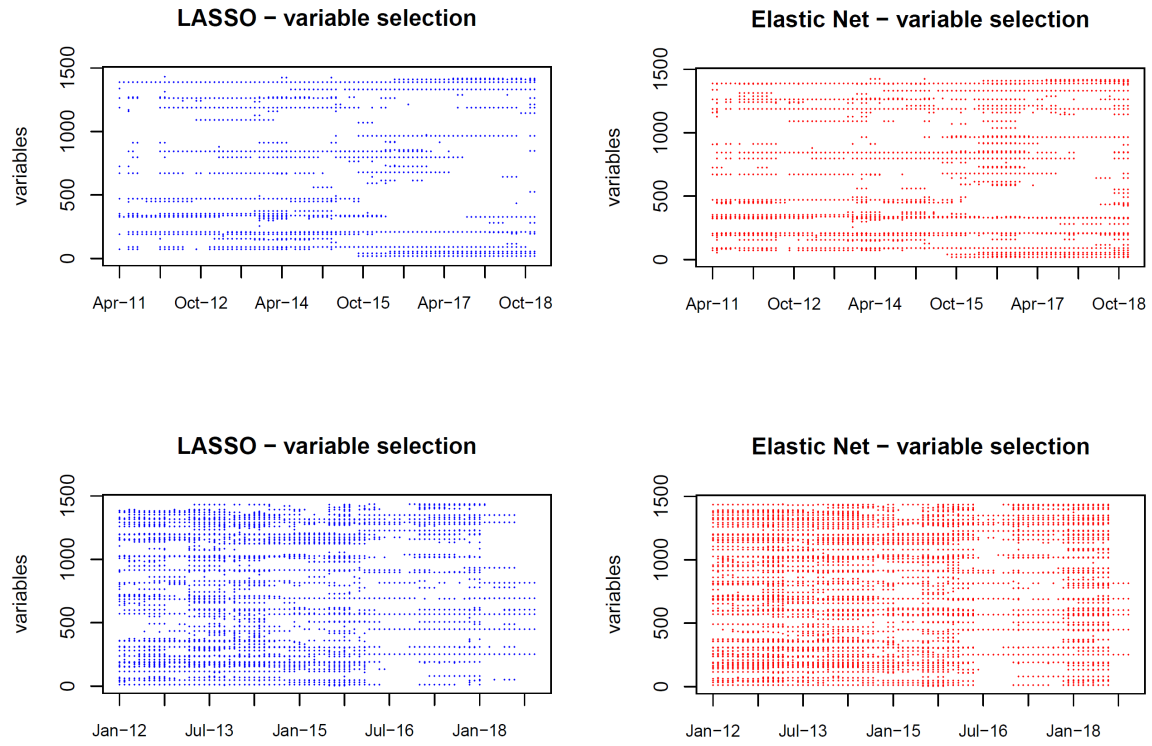


Figure D2 - Average number of variables selected by lasso and elastic net

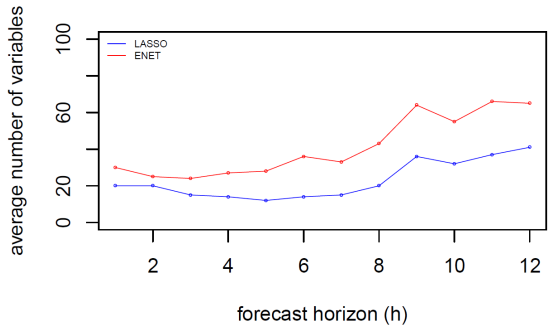


Figure D3 - Lasso (left) and Elastic Net (right) word clouds for $h = 1, 2, 3$

Appendix E: Results for selected horizons

Figure E1 - Inflation and forecasts for selected horizons ($h = 3, 12$)

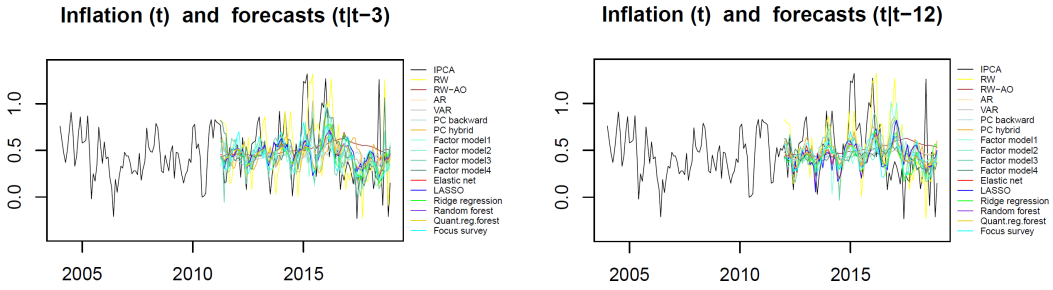
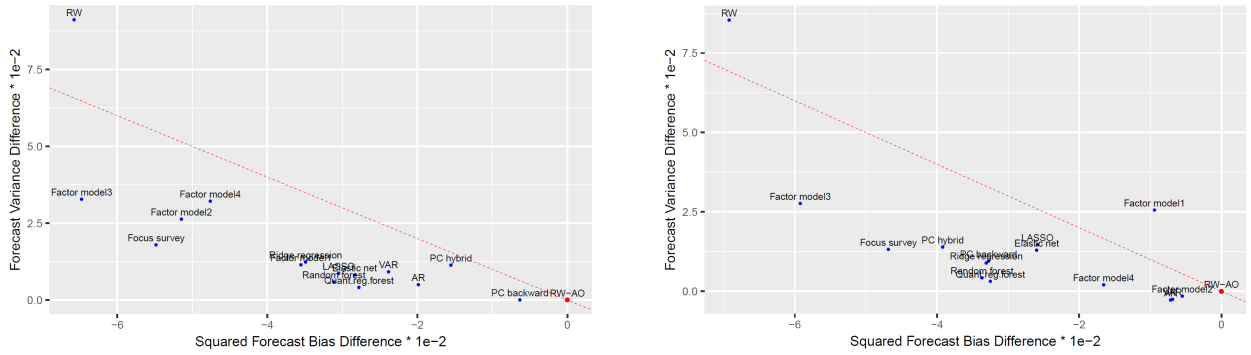


Figure E2 - Scatterplot of relative forecast variance and squared forecast bias ($h = 3, 12$)



Notes: The y-axis and x-axis represent relative forecast variance and squared forecast bias, computed as the difference between the forecast variance (squared bias) of the considered approach and the forecast variance (squared bias) of the RW-AO. Each point on the red dotted line represents a forecast with the same MSE as the RW-AO; points to the right are forecasts outperformed by the RW-AO, and points to the left represent forecasts that outperform the RW-AO.