WP/18/230

IMF Working Paper

# An Algorithmic Crystal Ball: Forecasts-based on Machine Learning

by Jin-Kyu Jung, Manasa Patnam, and Anna Ter-Martirosyan

***IMF Working Papers* describe research in progress by the author(s) and are published to elicit comments and to encourage debate.** The views expressed in IMF Working Papers are those of the author(s) and do not necessarily represent the views of the IMF, its Executive Board, or IMF management.

INTERNATIONAL MONETARY FUND

© 2018 International Monetary Fund WP/18/230

**IMF Working Paper**

Strategy, Policy and Review Department

# An Algorithmic Crystal Ball: Forecasts-based on Machine Learning[1]

**Prepared by Jin-Kyu Jung, Manasa Patnam, and Anna Ter-Martirosyan**

Authorized for distribution by Vikram Haksar

November 2018

***IMF Working Papers* describe research in progress by the author(s) and are published to elicit comments and to encourage debate.** The views expressed in IMF Working Papers are those of the author(s) and do not necessarily represent the views of the IMF, its Executive Board, or IMF management.

## Abstract

Forecasting macroeconomic variables is key to developing a view on a country's economic outlook. Most traditional forecasting models rely on fitting data to a pre-specified relationship between input and output variables, thereby assuming a specific functional and stochastic process underlying that process. We pursue a new approach to forecasting by employing a number of machine learning algorithms, a method that is data driven, and imposing limited restrictions on the nature of the true relationship between input and output variables. We apply the Elastic Net, SuperLearner, and Recurring Neural Network algorithms on macro data of seven, broadly representative, advanced and emerging economies and find that these algorithms can outperform traditional statistical models, thereby offering a relevant addition to the field of economic forecasting.

JEL Classification Numbers: C53, C45.

Keywords: Machine learning, forecasts, forecast errors.

Author’s E-Mail Addresses: jin-kyu.jung@whu.edu; mpatnam@imf.org; atermartirosyan@imf.org

[1] The authors would like to thank Tamim Bayoumi, Aquiles Farias, Vikram Haksar, Federico Kalan, Jungjin Lee, Sanjaya Panth, Helene Poirson Ward, Murtaza Husain Syed, Andrew Tiffin, and participants of the IMF research seminar for suggestions. All errors and omissions are our own.

Contents Page

I. Introduction . . . . . 4

II. Forecasting with Machine Learning . . . . . 7
A. Type of Learning . . . . . 8
B. Learning Method - Testing and Validation . . . . . 9
C. ML Algorithms . . . . . 10
1. Elastic Net . . . . . 10
2. Recurrent Neural Network . . . . . 13
3. Super Learner . . . . . 16

III. Data . . . . . 18

IV. Results . . . . . 21
A. Quarterly and Annual Forecast Performance . . . . . 21
B. Crisis-period forecasting and additional model benchmarks . . . . . 24

V. Conclusion . . . . . 26

VI. Appendix
Further Details on the Methodology . . . . . 28

References . . . . . 30

Tables

1. Selected learners for the Super Learner library . . . . . 17
2. Data Overview . . . . . 20
3. RMSE benchmarks for 1-step ahead real GDP growth (quarterly) . . . . . 22
4. RMSE benchmarks for 1-step ahead real GDP growth (annual) . . . . . 23
5. RMSE benchmarks for 1-step ahead real GDP growth (quarterly), forecast period including crisis episodes . . . . . 25
6. Super Learner Algorithm Library . . . . . 29

Figures

1. Overview of Machine Learning . . . . . 8
2. 10-fold cross-validation . . . . . 10
3. Simple perceptron . . . . . 13
4. Typical neural network . . . . . 14
5. Recurrent neural network . . . . . 15
6. USA Elastic Net Out-of-Sample Predictions Fit, including crisis period . . . . . 26
7. Neural Network Example . . . . . 28

# Glossary

## Acronyms

| | |
|---|---|
| **ANN** | Artificial Neural Network |
| **ARIMA** | Auto-Regressive Integrated Moving Average |
| **GDP** | Gross Domestic Product |
| **GRU** | Gated Recurrent Unit |
| **ICRG** | International Country Risk Guide |
| **IMF** | International Monetary Fund |
| **LASSO** | Least absolute shrinkage selection operator |
| **LSTM** | Long Short-Term Memory |
| **ML** | Machine Learning |
| **OLS** | Ordinary Least Squares |
| **PMI** | Purchasing Managers Index |
| **RMSE** | Root Mean Squared Error |
| **RNN** | Recurrent Neural Network |
| **RSS** | Residual Sum of Squares |
| **WEO** | World Economic Outlook (by the IMF) |

## Terms

| | |
|---|---|
| **cross-validation** | splitting dataset into $k$-folds and iteratively estimating model for each fold |
| **ensemble** | weighted combination of different algorithms into one model |
| **learner** | algorithm to be trained |
| **nowcasting** | forecasting values for the present, the very near future or the very recent past |
| **over-fitting** | over-generalization of an assumed input-output relationship |
| **perceptron** | computer model devised to represent or simulate the ability of the brain to recognize and discriminate |
| **training** | estimation of model parameters by feeding in the existing dataset |

## I. Introduction

Forecasting macroeconomic variables is key to developing a view on a country's economic outlook. Understanding the current and future state of the economy enables timely responses and policy measures to maintain economic and financial stability and boost resilience to episodes of crises and recessions.

Many international institutions and economic research bodies regularly produce and publish forecasts on economic variables. For instance, the International Monetary Fund (IMF) publishes macroeconomic forecasts bi-annually, as well as two interim updates, as part of its World Economic Outlook publications as does the World Bank with its Global Economic Prospects report in January and June each year. Many other institutions both private (Survey of Professional Forecasters, Consensus Economics, etc.) and public (central banks, government agencies) go through an intricate process to generate economic forecasts, underlining their importance to both researchers and policy-makers.

As with any task involving predictions of the future, however, forecasting is afflicted with error. For instance, Timmermann (2007) finds that WEO forecasts display a tendency for systematic over-prediction and that its performance is similar to that of the consensus forecasts. Similarly Genberg and Martinez (2014) also find that WEO forecasts tend to be consistently over-optimistic in times of country-specific, regional, and global recessions.

Most traditional forecasting models for economic variables rely on fitting data to a pre-specified relationship between input variables and the output (to be forecasted) variable. These models thereby assume a stochastic process underlying the true relationship between the variables in question (Breiman, 2001). In such cases, therefore, the model can only be as good as its specification, regardless of what the data might suggest. In contrast, a different approach to statistical analysis in general and forecasting in particular is offered by machine learning algorithms, which make next to no assumption[2] about the underlying relationship between the variables at hand and instead rely on an algorithmic approach to finding a function which best represents the relationship between input and output data. This latter approach to statistical modeling has long been largely ignored by economic research, however a recently growing awareness of its large applicability has led to the application of machine learning algorithms to economic research questions (Bang, Sen, and Basuchoudhary, 2015). A potential drawback to this approach is that, given the lack of a pre-defined model for analysis, machine-learning

[2] Exceptions exist, such as for support vector machines which make assumptions about the separability of data, etc.

algorithms hold limited explanatory power and cannot,as a result, readily explain what factors drive the forecasts. Yet, despite this limitation, Varian (2014) argues that growing amounts of data and ever complex-growing relationships warrant the usage of machine-learning approaches in the context of economics and gives a comprehensive overview of the literature required for economists to learn these techniques.

In light of this approach to economic forecasting and the fact that still only few applications of these methods are available in the context of macroeconomic now-casting and forecasting (Baldacci and others, 2016), we present three different machine learning algorithms to compute short-term GDP growth forecasts for seven countries across geographies and levels of economic development. Our goal is to assess whether machine learning techniques can offer a potential improvement to forecast accuracy, and thereby make a useful addition to the standard statistical toolbox for economic forecasting. We base our assessment on the application of the machine learning technique to a widely available economic data (IMF-WEO, for example), making our results comparable, in principle, to benchmark forecasts made in the WEO[3]. In other words, we work with traditional data sources to build our forecasts, without adding data from more granular or novel data sources, so that our results can reflect only the benefits of using the machine learning technique. Additionally, since the goal of our exercise is to obtain accuracy improvements for forecasting, we do not attempt to economically interpret, causally or otherwise, what factors drive the forecasts. We will therefore assess the contribution of the variables used for our analysis only in terms of their explanatory power for the overall forecast.

Our paper contributes to the small, but burgeoning literature, that seeks to apply machine learning (ML) techniques to improve economic forecasting. Chakraborty and Joseph (2017) explore areas of application for machine learning models in the context of central banking and see a large number of possibilities where these could be employed for the work of policy-makers. Accordingly, some research has already taken upon applying these new tools for economic forecasting and to come up with an alternative way to compute economic forecasts. For instance, Biau and D'Elia (2010) employ a Random Forest algorithm to forecast euro area GDP and find that some versions of this machine learning based approach are able to outperform benchmark forecasts produced by a standard AR model. Tiffin (2016) tries the Elastic Net and Random Forest algorithms to nowcast GDP growth in Lebanon, a country which only officially releases growth figures with a two-year delay, and thereby presents a viable alternative to determining timely estimates of economic growth in an emerging market economy. Tkacz and Hu (1999) introduce an approach to forecasting GDP growth using artificial Neural Networks (aNN) and

[3] In some specifications, when forecasting over an annual horizon, we supplement the annual data with current quarterly/monthly data to add a layer of now-casting.

obtain 15 to 19 percent more accurate forecasts than corresponding linear benchmark models. Chuku, Oduor, and Simpasa (2017) similarly employ artificial Neural Networks to forecast economic time series in African countries and find that these perform at least somewhat better than traditional, structural econometric and ARIMA models.

In our paper, we use a variety of machine learning algorithms – specifically, the Elastic Net, Recurrent Neural Network (RNN) and Super Learner – that encompass a range of complex linear and non-linear model specifications as well as non-parametric approaches to modeling. The Elastic Net algorithm provides a regression-based approach to forecasting that has the advantage of classic OLS, in terms of interpreting the contribution of each variable, but additionally offers a flexible way to conduct variable selection and prevent forecast over-fitting.

The RNN is a nonlinear dynamical model used to represent complex dynamical or sequential relationships between variables. This algorithm recognizes the time-series dimension of the data and makes use of sequential information to capture long-term temporal dependencies between the various input variables and the desired output (for example, GDP growth). To account for the additional cross-sectional interdependence of various economic variables, we employ a multivariate extension to the traditional RNN, which to the best of our knowledge, has not been applied to the field of economic forecasting before. Finally, the Super Learner algorithm combines output from a variety of machine learning algorithms by assigning weights on each algorithm's predictions based on their accuracy.

We apply the suite of machine learning algorithms to determine one-step[4] ahead GDP growth forecasts for Germany, Mexico, Philippines, Spain, United Kingdom, United States, and Vietnam and are able to achieve enhanced forecast performances as compared to standard benchmarks such as the forecasts produced by the IMF-WEO. Our choice of countries covers three advanced/G-7 economies (United States, United Kingdom and Germany), together with a diverse set of emerging and developing economies (Mexico, Philippines and Vietnam). We also included Spain in our analysis, as an example of a Euro-area, recession hit country, with a faster-than expected recovery, to test the strength of the ML approach on crisis induced volatility episodes[5].

[4] For quarterly data, "one-step ahead" designates the forecast for next quarter, for annual data the next year.

[5] Our exercise does not cover low-income economies as data for these economies are only available for shorter time periods and suffer from missing data, resulting in a prohibitively low number of observations. In future research, we plan to extend our coverage of economies by using recent advances in the application of ML techniques to missing data as presented for example by Che and others (2018).

We find the accuracy improvement from ML forecasts, over the WEO, to range between 49%-82% (depending on the country) for quarterly forecasts and between 4%-38% for annual forecasts. We find varied performance across the different ML algorithms, depending on the frequency of the data, with the ensemble-based approach (Super Learner) consistently outperforming other approaches when dealing with quarterly data. The RNN offers some advantages for a select set of advanced countries, when forecasting on an annual horizon.

The remainder of this paper is organized as follows: Section II provides a brief overview of the fundamental difference in traditional statistical analysis and the algorithm-based machine learning models. We also discuss the methodology of the three algorithms we employ in our forecasting exercise in this section. Section III presents the original data used and the different additions that were made to the dataset over the course of the estimation procedure. Section IV discusses the forecast performances achieved by the machine learning algorithms and compares them to conventional benchmarks. Section V concludes.

## II. Forecasting with Machine Learning

When it comes to statistical modeling, we can observe two large schools or "cultures" (Breiman, 2001). One assumes a specific stochastic data model that underlies the data generation process while the other rather tries to find a function that best predicts outputs given certain inputs. The former is the approach that so far has mostly dominated economic research and is applicable to the larger part of known econometric models. Typically, various different assumptions about the data distribution and type of relationship (e.g. linear/non-linear) between dependent and independent variables are made when setting up such a traditional statistical data model. In contrast to this, the algorithmic modeling culture considers the nature of the relationship between input and output variables as unknown and instead finds a function that pragmatically operates on the inputs to most accurately produce the given, observed outputs. This approach is commonly referred to as *machine learning* and comprises two 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(s) and/or the output. Figure 1 presents a general overview of the two elements which are discussed in the following sub-sections in detail.

**Figure 1. Overview of Machine Learning**

***Machine Learning Technique***

| **Learning** | **Algorithm** |
|---|---|
| *Type:* | |
| • Supervised<br>e.g., forecasting growth, credit default. | • Ridge regression (e.g., LASSO), Neural Networks, Decision trees. |
| • Unsupervised<br>e.g., risk rating, cluster analysis, anomoly detection. | • Principal components, k-mean/ hierarchical clustering. |
| *General Method:* | |
| • Train and Test | • Algorithm applied on a subset of data (*training set*) and performance evaluated on another subset (*test set*). |
| • Validate | • Fine tune algorithm parameters. |

## A. Type of Learning

While there is no one machine learning algorithm but rather a whole universe of different techniques, they can broadly be categorized into (1) supervised and (2) unsupervised learners (Hastie, Tibshirani, and Friedman, 2004)[6].

(1) Supervised learning pertains to problems where the output needed from a given model, i.e. the dependent variables, are clearly identified, even if the specific relationships in the data are not known. Traditional econometric models (regression-based such as the OLS) typically fall under the category of supervised learning as they commonly try to quantify the effect of a series of independent, explanatory variables on one or more known dependent variables.

[6] A third type of technique called "Reinforcement Learning" seeks to optimize an unknown "reward" function through repeated retro-feedback (Barto and Dietterich, 2004). This type of algorithm is neither given the reward function nor any training examples but instead aims at iteratively determining an optimal location of the input variables that yield the highest reward. Due to the lack of a training example or training set, reinforcement learning algorithms distinguish themselves from supervised learning. They are also different from unsupervised learning since the target variable is given in the form of a reward.

(2) Unsupervised learning in contrast has no specific output defined beforehand. The goal of unsupervised learning is to detect a pattern or latent variables based on a range of observed input variables. Given a certain dataset, algorithms falling under this category are tasked with recognizing patterns in the data and determining output classification categories.

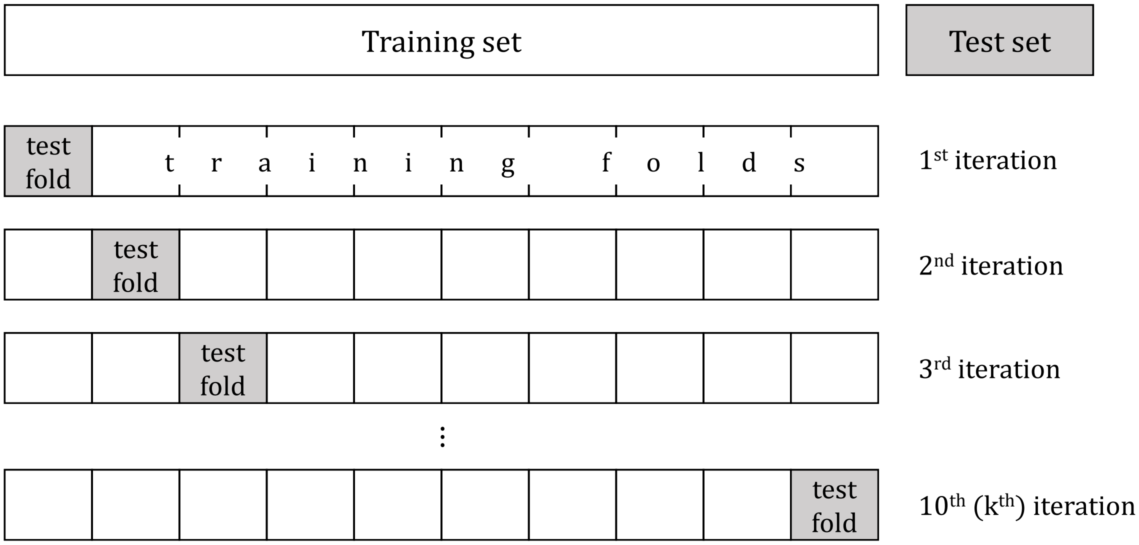
### B. Learning Method - Testing and Validation

Common to most machine learning algorithms, regardless of which of the above categories they belong to, are a number of practical measures to avoid the so-called phenomenon of *over-fitting*. Over-fitting denotes the over-generalization of an estimated input-output relationship to the extent that a specified model fits the data it was trained with very well, might however perform poorly if presented with new data. In such cases, a specific relationship between input and output variables for a given data sample was overly generalized and assumed to be valid for the entire population. Two measures try to mitigate this issue: *Testing* and *validation*.

Testing refers to the common practice of splitting the sample data into two parts: A first (usually larger) part called *training set* used to train a given learner algorithm and a second (usually smaller) part representing the *test data set* used to measure the predictive performance of the trained learner on previously "unseen" data. The test data is therefore put aside or "out of the sample". Any generalization stemming from the trained learner can thereby immediately be evaluated by applying the model on this new data. The predictive power and performance across different learning algorithms or learner specifications should also be assessed by comparing the error measures on this out-of-sample data to avoid the misleading comparison of over-fitted models.

Validation builds upon testing and pertains to the calibration of algorithms and is sometimes also called *tuning*. Depending on the algorithm at hand, different model parameters can be tuned such as the number of trees to grow in a decision tree algorithm, the penalty strength $\lambda$ in an Elastic Net model (section II.C.1), the number of nodes and layers in a neural network (section II.C.2), etc. While these parameters can be calibrated manually, it is most often more effective and efficient to apply a more automated way to find optimal parameter values through so-called *k-fold* cross-validation. With cross-validation, the existing training data set is again split into *k* folds and iteratively re-estimated. Figure 2 schematically shows an example of a cross-validation with 10 folds.

**Figure 2. 10-fold cross-validation**



The model is trained a total of *k* times and every time the parameters of the model are calibrated depending on what performance could be achieved. The higher *k* is, the better should the resulting calibration be. This, however, comes at the expense of computational efficiency with an increasing number of folds. It is common to apply 10 folds for learner parameter calibration within the scope of cross-validation.

## C. ML Algorithms

In this paper, we perform forecast estimations based on three machine learning algorithms which are all part of the supervised learning category: The Elastic Net, Recurrent Neural Network and Super Learner which are discussed in more detail over the following sections.

### 1. Elastic Net

The Elastic Net algorithm was originally proposed by Zou and Hastie (2005) and is a combination of the ridge and least absolute shrinkage and selection operator (LASSO) regressions. Both approaches are forms of penalized regressions, a method to improve Ordinary Least Squares (OLS) regressions by performing dimension reduction and/or variable selection when dealing with large datasets with multiple, possibly correlated regressors.

A ridge regression alone is very similar to an OLS regression in that the objective to minimize the residual sum of squares (RSS) prevails. An additional feature distinguishes this approach from a standard OLS regression, however, by trying to minimize an additional so-called *shrinkage penalty* term which decreases when the estimated coefficients of the regression become close to zero (Hoerl and Kennard, 1988). When both the residual sum of squares and this shrinkage penalty term are subject to a summarized minimization problem, the optimal result will be achieved by shrinking only those regressors that are correlated. Following the notation used by Tiffin (2016), the overall minimization problem is then given as follows:

$$\hat{\beta} = arg \min_{\hat{\beta}_j} \Bigg[ \underbrace{\sum_{i=1}^{n} (Y - X\hat{\beta})^2}_{\text{RSS}} + \underbrace{\lambda \sum_{j=1}^{p} (\hat{\beta}_j)^2}_{\text{ridge penalty}} \Bigg] \tag{1}$$

where $n$ is the number of observations and $p$ the number of explanatory variables. Clearly, the extent of the shrinkage penalty is determined by the parameter $\lambda$, whose optimal value will in practice be determined by iterative cross-validation. Generally speaking, a higher $\lambda$ will lead to a stronger shrinkage of the regression coefficients whereas a $\lambda$ of zero will simply produce the same results as a standard OLS regression.

Similarly, the least absolute shrinkage and selection operator (LASSO), originally proposed by Tibshirani (1996), shrinks the coefficients of an OLS regression, however, operates on a slightly different penalty term. The minimization problem is here given by

$$\hat{\beta} = arg \min_{\hat{\beta}_j} \Bigg[ \underbrace{\sum_{i=1}^{n} (Y - X\hat{\beta})^2}_{\text{RSS}} + \underbrace{\lambda \sum_{j=1}^{p} \left|\hat{\beta}_j\right|}_{\text{LASSO penalty}} \Bigg] \tag{2}$$

and implies that coefficient values of zero are now possible as long as the $\lambda$ parameter is large enough. The LASSO is thereby able to conduct an actual variable *selection* whereas the ridge regression only shrinks the coefficients close to zero but does not exclude them from the model altogether.

The Elastic Net algorithm now combines the penalty elements of both the ridge regression and the LASSO and regulates the size of the penalty via the previously known parameter $\lambda$ from equations (1) and (2):

$$\hat{\beta} = arg\min_{\hat{\beta}_j}\left[\underbrace{\sum_{i=1}^{n}(Y - X\hat{\beta})^2}_{\text{RSS}} + \lambda\sum_{j=1}^{p}\left[\underbrace{(1-\alpha)(\hat{\beta}_j)^2}_{\text{ridge}} + \underbrace{\alpha\left|\hat{\beta}_j\right|}_{\text{LASSO}}\right]\right] \tag{3}$$

The added term $\alpha$ determines the relative weights of the two penalties and is determined via cross-validation, much like the $\lambda$ parameter. In case $\alpha$ equals 0, the entire model becomes the original ridge regression specification and the LASSO specification when $\alpha$ equals 1. Zou and Hastie (2005) consider this joint penalty term of the Elastic Net a generalization of the LASSO which proves to be especially superior in situations where (1) $p > n$, i.e. the number of regressors exceeds the number of observations (also known as "fat data"), (2) a group of variables show high pairwise correlations, and (3) $n > p$, i.e. the number of observations largely exceeds the number of regressors (known as "tall data"). In the first case (1), a conventional LASSO penalty would only select a maximum of $n$ variables in the resulting regression due to the convex nature of the optimization problem. In the second case (2), high pairwise correlations among regressors causes the LASSO to select only one of the correlated regressors with little emphasis on which one is eventually kept in the regression. Finally, for tall data (3) and strong correlations between regressors, Tibshirani (1996) has found that the LASSO is outperformed by a ridge regression. The combination of the LASSO and ridge regression models through the alpha parameter into a new Elastic Net model now makes its predictive performance per definition as least as powerful as both approaches in a standalone setting (Tiffin, 2016).

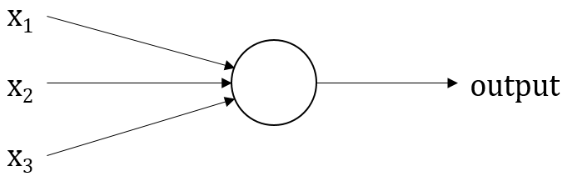
Overall, the advantages of the Elastic Net approach are its intuitiveness and high computational efficiency, its resilience against potential multicollinearity among regressors, and the incorporation of both dimension reduction and variable selection in one model. Smeekes and Wijler (2018) also confirm in an extensive simulation study that penalized regression methods are more robust to mis-specification than the well-known dynamic factor approach. Moreover, it is possible for the Elastic Net to produce an output indicating the selected variables and their respective weights in the final model specification, thereby enabling a basic understanding of the algorithmically determined input-output relationship, a feature often missing from the capabilities of machine learning models which are rather criticized for their "black-box nature" (Chakraborty and Joseph, 2017).

The data used in our attempt to predict GDP growth rates will be discussed in more detail in section III, however resembles a "fat" dataset, making the Elastic Net a particularly suitable approach for our problem.

## 2. Recurrent Neural Network

Artificial Neural Networks (ANN) are among the first machine learning algorithms that were developed and generally mimic the structure of the human brain by running one or more input variables through so-called "learning nodes" to produce an output of interest (Nielsen, 2015). One of the earliest kinds of learning nodes are called "perceptrons" and were first introduced by Frank Rosenblatt in 1958. While the original perceptron was only able to absorb binary inputs to produce a single binary output, the nowadays more commonly used "sigmoid neuron" is capable of processing both discrete and continuous inputs and outputs. The fundamental principle of how a learning node operates, however, prevails in that the provided inputs are run through a linear or (more commonly) non-linear model to produce a desired output variable. The simplest representation of a perceptron is given in figure 3, where three hypothetical inputs, $x_1$, $x_2$, and $x_3$ are considered.

**Figure 3. Simple perceptron**

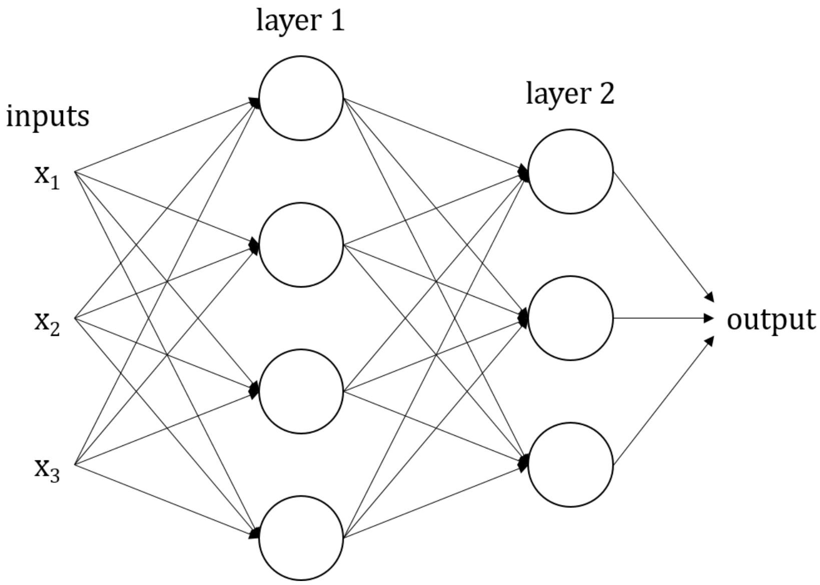


In order to derive the output, Rosenblatt (1958) introduced weights $(w_1, w_2, w_3)$ which represent the importance of the input variables in the output determination process. The overall output of the perceptron is then dependent on whether the weighted sum of the inputs exceeds or falls below a threshold value, which is a parameter of the perceptron (Nielsen, 2015). Algebraically put, this mechanism can be represented as follows:

$$\text{output} = \begin{cases} 0 & \text{if } \sum_j w_j x_j \leq \text{threshold} \\ 1 & \text{if } \sum_j w_j x_j > \text{threshold} \end{cases} \tag{4}$$

In reality, decision-making or any kind of input and output relationship is much more complex than what a single perceptron could model. Instead, an entire network of perceptrons is a more realistic representation of real-life decision-making processes, resulting in so-called neural *networks*. In such networks, different layers of perceptrons are linked to each other in a whole system of neurons and determine an output of interest in a more complex manner. A typical representation of a neural network is shown in figure 4 with a total of seven neurons spread out across two layers[7].

**Figure 4. Typical neural network**



The characteristic trait of information being passed from one neuron to the others in one direction has earned these types of neural networks the name of "feed-forward neural networks".

Such feed-forward neural networks can be powerful models for prediction applications and deal with both classification (discrete output) and regression (continuous output) problems. The specification of the neural network and especially the number of neurons and layers to be included in the final network can be regarded as an arbitrarily set number. There is currently no universally accepted analytical way to determine the optimal number of neurons and layers for a given classification or regression application, adding large "degrees of freedom" to the estimation of neural networks. Instead, the existing academic literature rather suggests a number of rules of thumb, ranging from "somewhere between the input layer size and the output layer size" (Blum, 1992) to "as many hidden nodes as dimensions needed to capture 70-90% of the variance [in] the input data" (Boger and Guterman, 1997). In practice, an initial specification of a neural network would be trained with the training data split and tested for predictive performance with the test data. If the performance is unsatisfactory, adjustments in the network's architecture (i.e. number of layers and respective neurons) can be made and the entire network

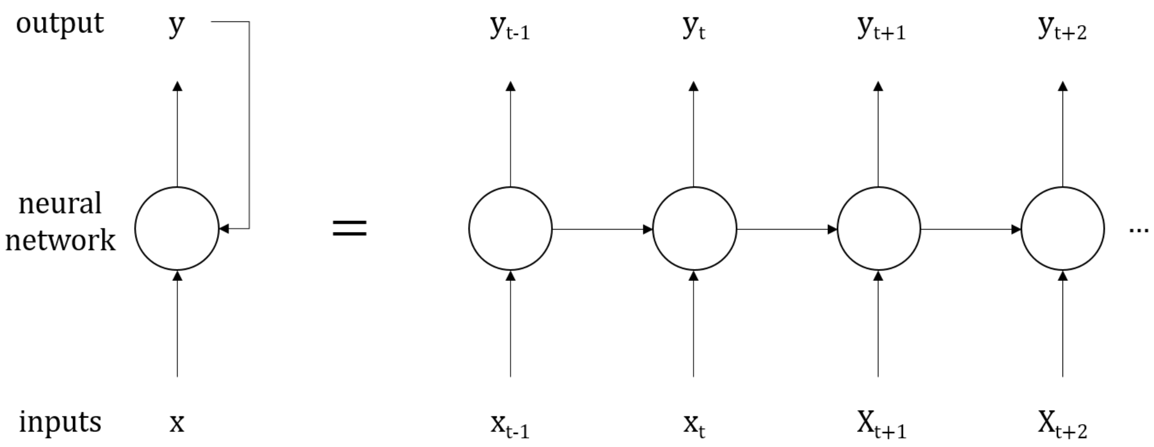
[7] For an example describing the intuitive workings of neural networks see the appendix section

re-trained and re-tested (Tkacz and Hu, 1999). This process may be repeated until the neural network achieves a satisfactory predictive performance. A more automated process is called *hyperparameter tuning*, which is basically an estimation of a series of different neural network specifications for the same classification or regression application. The network producing the best error measure (RMSE or accuracy) is then selected as optimal within the hyperparameter tuning exercise. This approach, however, requires large computational capacities and the issue of determining the right number of neurons and layers clearly represents a challenge to specifying the best-performing neural network.

Another potential issue with plain feed-forward neural networks is the quasi-treatment of input data as cross-sectional. Especially when it comes to time-series data, this feature of feed-forward neural networks might not be suitable for a given application due to the omission of the temporal component in the data and warrant the search for other, more appropriate models. One of these more appropriate models can be found in an extension of the feed-forward neural network, the so-called *recurrent neural networks* (Elman, 1990).

Recurrent neural networks (RNN) consist of the same perceptrons and layers that make up a plain feed-forward neural network with one important addition: starting at the first observation, the estimated output value is "passed on" to the estimation of the next observation's output value. In a time series context, the output corresponding to the $t+1$ observation is thus dependent on whatever output was computed for the observation in $t$. A schematic representation of a recurrent neural network is shown in figure 5, illustrating the workings of an RNN.

**Figure 5. Recurrent neural network**



The recurrent aspect of this type of neural network can be thought of as multiple copies of the same network, successively ordered and each passing a message to the successor. RNNs accordingly consume more computational power than plain feed-forward neural networks, may however produce better-performing models for a given time-series prediction problem. By incorporating more than one layer, RNNs are part of a field in machine learning called *deep learning* (LeCun, Bengio, and Hinton, 2015). A number of different extensions and enhanced models on the basis of RNNs exist, such as Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) or Gated Recurrent Units (GRU) (Cho and others 2014; Chung and others 2014). Even within the "standard" RNN space, a myriad of different specifications have emerged which each might be more or less suitable for a specific prediction application (Dorffner, 1996). While a considerable amount of research on RNNs and their applications exists (e.g. Debar and Dorizzi 1992, Connor, Martin, and Atlas 1994, Mikolov and others 2010, Gregor and others 2015, and many more), only a few deal with multivariate RNNs (e.g. Broomhead and Lowe 1988, Chakraborty and others 1992, Goel and others 2016, Che and Purushotham 2017) and even less deal with multivariate RNNs for an economics-related application (e.g. White and Diego 1988, Kamijo and Tanigawa 1990, Garcia Torres and Qiu 2018 for the prediction of financial market data and Kuan and Liu 1994 for exchange rate forecasts). For the purposes of this paper, we will be employing an *Elman* network that features an additional input layer called the *state layer* besides the multi-layer perceptron that underlies all neural networks. This state-layer incorporates a state space model approach to time series analysis in a neural network setting and is appropriate for the GDP growth prediction exercise in section IV. The Elman network we employ is specified with two layers of nodes containing 10 and 7 nodes, besides the input and output layers. This specification was chosen with model parsimony and adequate forecast performance in mind. We thereby present a multivariate recurrent neural network for macroeconomic forecasting purposes, an application that to the best of our knowledge has not yet been covered by the academic research literature.

### 3. Super Learner

Super Learner is an algorithm that creates an *ensemble* of different machine learning algorithms. Ensembles build a set of different learner algorithms and then classify or predict new data points by processing a weighted vote of said learners (Dietterich, 2000). The original ensemble was based on Bayesian averaging, however a great many different approaches to building ensembles have emerged ever since.

The Super Learner was first proposed by van der Laan, Polley, and Hubbard (2007) and uses cross-validation to identify a combination of learners from a library of algorithms that performs best on a given prediction problem. Different weights are assigned to the selected learners and adjusted iteratively in the process of cross-validation to minimize RMSE (for regression applications) or maximize accuracy (for classification problems). Table 6 lists the different learners that the R Super Learner package contains. It is possible to specify a subset of these learners and to exclude some algorithms from consideration for a Super Learner ensemble. For our study, we have specified a total of 10 algorithms in the Super Learner library as shown in table 1 which were selected based on their suitability to the forecasting problem at hand and overall parsimony.

**Table 1. Selected learners for the Super Learner library**

| Learner | Description |
|---|---|
| `bayesglm` | Bayesian generalized linear model |
| `gam` | generalized additive models |
| `glm` | generalized linear model |
| `glmnet` | Elastic Net |
| `mean` | arithmetic mean |
| `nnet` | neural network |
| `polymars` | polynomial spline regression |
| `randomForest` | random Forest |
| `rpart` | recursive partitioning |
| `svm` | support vector machine |

Polley and van der Laan (2010) conduct a study applying the Super Learner for a number of practical prediction problems and find that it can also be a robust selector of algorithms for small data sample sizes. The built-in cross-validated risk assessment of the assigned weights to each learner controls for over-fitting even when a large number of learners are included in the final ensemble. While it is possible to build any ensemble of different algorithms manually, the advantages of the Super Learner clearly lie in the cross-validated learner selection and the convenient automated weight assignment to each selected learner. A large number of different learner libraries and resulting ensembles can be tested and evaluated with only very small computational requirements. Especially its applicability to small sample sizes is a suitable attribute for our forecasting problem at hand.

## III. Data

Before turning to the results the three chosen learner models produced, it is crucial to discuss the data and particularly the challenges macroeconomic variables and their low frequencies present. Machine learning algorithms generally deal with large amounts of data, often called "big data". The high frequency nature of behavioral, geo-spatial, telemetric, transactional, and other data that are collected by personal devices and digital products allow for the compilation of vast databases, containing a myriad of different variables.

In macroeconomics, the traditional economic indicators of interest mostly pertain to national accounts or fiscal, labour, monetary, and trade statistics which are commonly collected on an annual or, at most, quarterly basis and therefore lead to much less data accumulation than the aforementioned high frequency variables. While main indicators such as GDP growth or inflation are fairly easily gathered and in some countries even available beyond a century, most other macroeconomic variables are much less easily obtained. Reliably and consistently, for most countries measures such as trade statistics, balance of payments data, and fiscal statistics etc. are mostly available from the 1980s. For emerging and especially developing economies, data availability is even poorer and sometimes close to not existent. This poses serious challenges to the applicability and reliability of results produced by machine learning algorithms, especially since an existing dataset is commonly split into a training and test data set, reducing the actual data used to train a learner even further. The seven countries in this study were mainly selected by assessing the following criteria:

1. Sufficient data availability in terms of observations $(n)$ and variables $(p)$
2. Availability of benchmark forecast performances (e.g. by central banks, IMF World Economic Outlook, etc.)
3. Preferably achieve balanced set of advanced, emerging and developing economies

Specifically, the selection of countries in our study and the available data are given in table 2. Germany, Mexico, Philippines, Spain, United Kingdom, United States and Vietnam represent countries with reasonable data availability across at least 29 (11) macroeconomic variables on a quarterly (annual) basis and cover advanced and emerging economies. Our choice of countries covers three advanced/G-7 economies (United States, United Kingdom and Germany), together with a diverse set of emerging economies (Mexico, Philippines and Vietnam). We also included Spain in our analysis, as an example of a Euro-area, recession hit country, with a faster-than expected recovery, to test the strength of the ML approach on crisis induced volatility episodes. Data for other developing and low-income economies suffer from missing data or are only available for shorter time periods, resulting in a prohibitively low number of observations.

For our study, we employ macroeconomic data as provided by the International Monetary Fund's World Economic Outlook (WEO) database as of April 2017. The IMF's WEO offers publicly accessible records of national accounts, monetary, trade and labor statistics as well as fiscal data and balance of payments accounts. In addition to WEO data, we have added survey data such as purchasing managers indexes (PMI), business and consumer confidence indexes and financial market data in form of stock market indexes, world energy prices, etc. as provided by Bloomberg for each of the countries. The exact data series available for each country slightly vary individually, however consistently represent indicators or proxies for the overall business sentiment that prevails in that country in that year. To further enlarge our dataset, we have also included data from the International Country Risk Guide (ICRG), a ratings-based publication assessing political, economic, and financial risk for 140 advanced, emerging, and developing economies. These risk types are evaluated across 30 specific risk metrics and mostly date as far back as 1984. Accordingly, we have added ICRG data for all countries where other macroeconomic data is available on or after 1984. Due to the above mentioned traditional use-cases of machine learning algorithms in large data environments, we have attempted to gather a somewhat sizable dataset with sufficient variables for our purposes in order to appropriately leverage the advantages of machine learning algorithms. All three machine learning models were fed all possible variables and no previous variable selection has been done by the authors. For each country, only data for the country in question was used (besides world energy prices) and all data was potentially available for WEO forecasters in April[8] of each respective year. Our focus on achieving close comparability, therefore, precludes the use of some mixed frequency lead-time indicators (useful for now-casting relevant outcome variables) or granular proxies (such as satellite data)[9].

Since we are interested in making one-year ahead growth forecasts, annual real GDP growth (our dependent variable $Y$) in time $t$ is associated with the explanatory variables ($X$) in time

[8] WEO benchmarks were taken from the April edition of each year's World Economic Outlook. The data used by us was available before April in each year and could have equally been used by WEO forecasters.

[9] A potential comparability issue due to ex-post revisions to the World Economic Outlook database theoretically prevails mostly for the forecast year 2016, which is the last year for which we conducted the below forecasting exercise. To address this concern, later in the results section we assess the sensitivity of our results to dropping the year 2016.

**Table 2. Data Overview**

| Country | Frequency | Obs. | Variables | Start |
|---|---|---|---|---|
| Germany | annual | 44 | 26 | 1972 |
| | quarterly | 91 | 39 | 1992Q1 |
| Mexico | annual | 28 | 47 | 1988 |
| | quarterly | 123 | 29 | 1984Q1 |
| Philippines | annual | 31 | 43 | 1985 |
| | quarterly | 67 | 37 | 1998Q1 |
| Spain | annual | 31 | 28 | 1980 |
| | quarterly | 79 | 46 | 1995Q1 |
| United Kingdom | annual | 25 | 31 | 1991 |
| | quarterly | 98 | 48 | 1991Q1 |
| United States | annual | 32 | 55 | 1984 |
| | quarterly | 118 | 45 | 1987Q2 |
| Vietnam | annual | 46 | 11 | 1970 |
| | quarterly | 54 | 44 | 2001Q2 |

$t-1$ to train the learning algorithms. Once the different algorithms are trained, the last actual data from 2016 ($X_{t+n}$) can be used to produce a real GDP growth forecast that represents the 2017 annual growth figure ($Y_{t+n}$). Analogously, quarterly y-o-y growth figures $Y_t$ are associated with the explanatory variables from a quarter ago, i.e. $X_{t-1}$, leaving e.g. the $X$ data of 2016 Q4 for prediction of growth in 2017 Q1.

To judge the accuracy of machine-learning based forecasts, we benchmark them against the forecast performance of the IMF's World Economic Outlook (WEO). The WEO annual forecast errors are obtained by comparing the annual next-year forecasts reported in the fall WEO vintages (October) to the actual outruns. Similarly, the WEO quarterly forecast errors are obtained by comparing the next-quarter forecasts reported in the respective spring (April) and fall vintages (October) to the actual outruns.

# IV. Results

## A. Quarterly and Annual Forecast Performance

Using the algorithms and data described above, we produced one-quarter and one-year ahead real GDP growth forecasts for the seven countries in our dataset and calculated the RMSEs as measures of forecast accuracy for each country and each learning algorithm[10].

In order to ensure comparability of RMSEs to the benchmark values, we have used a "recursive out-of-sample" calculation of the RMSEs in the style of Clark and McCracken (2001). We calculate forecast performance by allowing for the stepwise enlargement of the training data. More specifically, on any date *t*, we include all data from the beginning of the data series up to *t* in the training set. E.g. when in *t* the number of previous observations was 100, in *t+1* the number of observations will be 101, etc. RMSEs are calculated for the single point forecast in *t+h*. This is the best representation of how "online" machine learning algorithms usually work in a practical context. With further accumulation of data when moving ahead in time, more training data becomes available for learners which ideally should result in enhanced forecast performances. In addition, it also accurately represents the situation for a benchmark forecaster at any point in time, i.e. for a given year, only the information available at that point in time is used to compute the forecast without any benefit of hindsight or additionally available data, making this a "fair" comparison. The risk of over-fitting discussed in section II is also mitigated by the fact that for each forecast, a virtually new training set is used through the addition of a new observation in each instance. A misleading generalization of once estimated parameters can therefore not occur by design. While the Elastic Net and Super Learner algorithms each produce an output specifying the final selected variables and ensemble model, the nature of our estimation procedure as described above leads to the estimation of a new model specification for each year (or quarter). For lucidity purposes we do not report the individual model specifications due to the large number thereof[11].

The individual forecast errors for each year were squared, averaged and then square-rooted to derive an overall RMSE for a test period from 2010 to 2016. The common practice when it

[10] In addition to forecasting per country, we also considered pooling a set of comparable countries and performing the estimation. However, we found consistently that pooling produced inferior results compared to the single time-series estimation. Possible explanations for the under-performance of pooled estimators relate to the heterogeneity of country experiences (different crisis and boom periods) and model parameters that generate noisy training estimates.

[11] These are available upon request.

comes to the training and test data split is to randomly draw data points from the entire dataset, however for the purposes of our forecasting exercise, extreme events such as the global financial crisis 2008-2009, Asian financial crisis 1997, new economy bubble burst in the early 2000s etc. warrant the inclusion of these events in the training rather than the test set to help the model train well over the occurrence of such events. However, later in section IV.B, we evaluate the robustness of the ML methods to forecast extreme events and crisis-episodes by including the period of the GFC (2007 onwards) to the test set. A fully randomized test sample selection could compromise the comparability of forecast performances if data points from further in the past were randomly included in the test set. Additionally, the time series character of economic growth i.e. elements of autocorrelation and typical forecasting situations risk being ignored in a random test set. Selecting a fixed-period test data set should be able to set these concerns aside. While Rossi and Inoue (2012) find that forecast performances are typically not robust against specific choices of test periods, we also attempt to address this issue by contemplating two different test periods as shown in table 5.

The results for the forecasts based on this approach are shown in table 3, alongside a benchmark RMSE from the IMF's own World Economic Outlook forecasts. The gray-shaded cells indicate the forecasts outperforming (lower RMSE) the benchmark WEO RMSE in each row.

**Table 3. RMSE benchmarks for 1-step ahead real GDP growth (quarterly)**

| | obs. | WEO | RNN | Elastic Net | Super Learner | Accuracy Increase |
|---|---|---|---|---|---|---|
| United States | 118 | 0.72 | 0.77 | 0.46 | 0.33 | 55% |
| United Kingdom | 98 | 0.63 | 0.71 | 0.43 | 0.25 | 60% |
| Germany | 91 | 1.18 | 1.55 | 0.73 | 0.37 | 69% |
| Spain | 79 | 1.05 | 3.21 | 0.27 | 0.17 | 83% |
| Mexico | 123 | 0.95 | 1.46 | 0.87 | 0.62 | 36% |
| Philippines | 67 | 2.33 | 1.83 | 1.09 | 0.54 | 77% |
| Vietnam | 54 | 1.08 | 1.15 | 1.08 | 0.55 | 49% |

Note: Table 3 reports the RMSEs calculated for quarterly real GDP growth forecasts for a period from 2010-2016. The column *WEO* reports RMSEs from IMF forecasts. Gray-shaded cells indicate better fit than the respective benchmark WEO RMSE. The column *Accuracy Increase* indicates the percentage difference in the RMSE between the ML (best-model) and the WEO forecasts.

We find that in all cases, machine learning algorithms outperform the WEO forecasts in the test period between 2010 and 2016. The Elastic Net and Super Learner models consistently outperform the benchmark whereas the Recurrent Neural Network outperforms WEO forecasts only once in the case of the Philippines. We also calculate the accuracy improvement of the Super Learner model (the best performing ML technique) over the benchmark WEO performance and find a high performance increase. The accuracy improvements range from 49% (Vietnam) - 83% (Spain) with the average increase being approximately 61% across all countries. An examination of the produced ensembles by the Super Learner however yields no discernible pattern with regard to the question if there is an algorithm that particularly stands out with a disproportionate weight or otherwise is included frequently. As discussed in section II however, it is possible that better performing models exist within the bounds of the respective algorithms which have not been explored in this study. Especially the Recurrent Neural Network offers a theoretically infinite number of different specifications by altering the number of nodes and layers. The computationally highly intensive hyperparameter tuning discussed in section II.C.2 may offer an RNN specification that produces better performing forecast errors and leaves room for future research in this area.

To evaluate the forecasting performance of machine-learning models over an annual horizon,, we also repeat the forecasting exercise by using annual real GDP growth data[12], as shown in table 4.

**Table 4. RMSE benchmarks for 1-step ahead real GDP growth (annual)**

| | obs. | WEO | RNN | Elastic Net | Super Learner | Accuracy Increase |
|---|---|---|---|---|---|---|
| United States | 32 | 0.73 | 0.65 | 1.49 | 1.01 | 11% |
| United Kingdom | 25 | 0.74 | 0.55 | 1.55 | 7.65 | 26% |
| Germany | 44 | 1.53 | 1.29 | 1.26 | 4.49 | 18% |
| Spain | 31 | 1.87 | 2.73 | 2.51 | 2.03 | – |
| Mexico | 28 | 1.17 | 3.34 | 1.28 | 1.80 | – |
| Philippines | 31 | 2.04 | 1.96 | 2.34 | 3.35 | 4% |
| Vietnam | 46 | 0.79 | 1.17 | 0.49 | 0.91 | 38% |

Note: Table 4 reports the RMSEs calculated for annual real GDP growth data on an out-of-sample period from 2010-2016. The column *WEO* reports RMSEs from IMF forecasts. Gray-shaded cells indicate better fit than the respective benchmark WEO RMSE.The column *Accuracy Increase* indicates the percentage difference in the RMSE between the ML (best-model) and the WEO forecasts.

[12] The results shown are calculated by using annual real GDP growth data as opposed to annualized quarterly growth data. Incorporating annualized 4-step ahead quarterly growth data has proven to add more noise to the forecasting exercise and has therefore been disregarded.

Forecast performance declines in absolute as well as relative terms when measured against the WEO benchmark. For Spain and Mexico for instance, none of the ML models are able to outperform the RMSE produced by the WEO forecasts[13]. Accuracy improvements are not as high as those obtained using the quarterly models but range from 4% (Philippines) – 38% (Vietnam) with the average increase being approximately 19% across all countries registering an improvement. The low number of observations at the annual horizon greatly hampers the performance of ML forecasts as the algorithms have fewer sample observations to train over. We find, in general, that ML methods tend to perform poorly in small samples with volatile data-series, and resulting estimates from these cases should be treated with caution.

Finally to address the potential comparability concern between WEO and ML forecasts due to ex-post revisions to the World Economic Outlook, we assess the sensitivity of our results to dropping the year 2016 which is the latest year subject to considerable revisions at the time of forecasting. We find our results hold up remarkably well with accuracy improvements ranging from 34-84 percent for the quarterly horizon and 8-38 percent for the annual horizon. The pattern of our results is also fairly similar with the ML forecasts outperforming the WEO for all countries over the quarterly horizon and for 5 out of 7 countries for the annual horizon (with no improvements for Mexico and Spain). A caveat to this exercise is that, in principle, historical data series may be revised over longer time-horizons (for example, from periodic revisions of the national accounts) affecting therefore the ex-post quality and bias of the forecasts. Nevertheless, we believe that our analysis is not undermined significantly by this concern, since the direction of the bias in unknown ex-ante and especially given the strong robustness of our results to dropping the year where large revisions may have taken place.

### B. Crisis-period forecasting and additional model benchmarks

To test for the robustness of machine-learning forecasts to crisis episodes, we expand the RMSE test period to 2007 to 2016 and thereby incorporate the years of the global financial crisis. The benchmark values provided by the WEO forecasts changes accordingly and we expect an overall deterioration of forecast performances due to the increased volatility in economic growth between 2007 and 2010. The forecast performance results for next-quarter growth predictions are shown in table 5.

[13] One possible explanation for the low forecast accuracy of ML models Spain and Mexico at the annual horizon compared to the quarterly horizon (where forecast accuracy is high relative to the benchmark) is the large volatility of annual growth rates in the out-of-sample period. At the quarterly level, there is less inherent volatility due to the high time-series dependence of quarterly outruns.

**Table 5. RMSE benchmarks for 1-step ahead real GDP growth (quarterly), forecast period including crisis episodes**

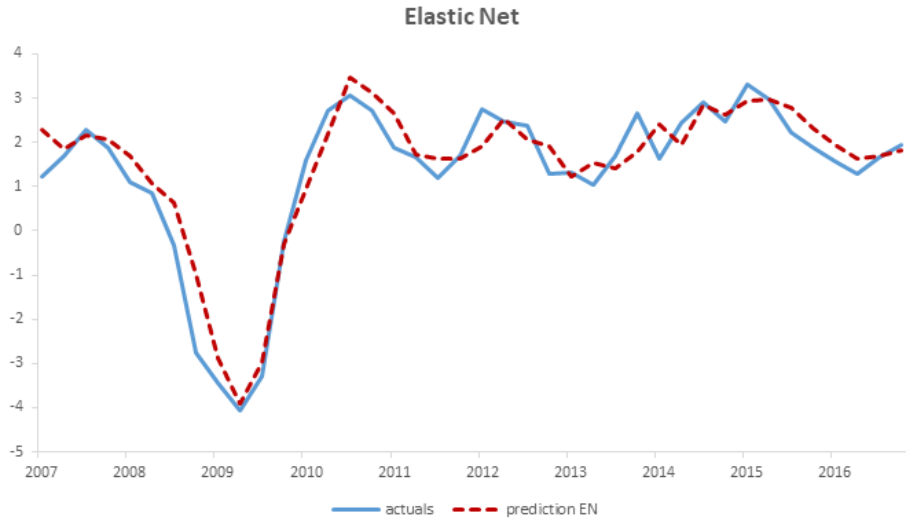
| | obs. | WEO | AR | BMA | VAR | ML | Accuracy Increase |
|---|---|---|---|---|---|---|---|
| United States | 118 | 1.11 | 0.94 | 0.51 | 1.65 | 0.35 | 68% |
| United Kingdom | 98 | 1.39 | 1.03 | 0.62 | 1.35 | 0.55 | 60% |
| Germany | 91 | 1.56 | 1.49 | 1.17 | 1.54 | 0.41 | 74% |
| Spain | 79 | 1.12 | 0.80 | 0.36 | 1.09 | 0.20 | 82% |
| Mexico | 123 | 1.60 | 1.52 | 2.09 | 1.15 | 0.81 | 49% |
| Philippines | 67 | 2.20 | 1.37 | 1.51 | 2.20 | 0.56 | 74% |
| Vietnam | 54 | 1.27 | 1.18 | 0.88 | 1.34 | 0.49 | 61% |

Note: Table 5 reports the RMSEs calculated for next-quarter real GDP growth forecasts for quarterly data from 2007-2016. The column *WEO* reports RMSEs from IMF forecasts, *AR* from an Autoregressive Model, *BMA* from a Bayesian Averaging Model and *VAR* from a Vector Autoregressive Model. The column *ML* reports the RMSEs produced by the best-performing machine learning algorithm out of the three considered in this paper for each country. Gray-shaded cells indicate the best RMSE for each country. The column *Accuracy Increase* indicates the percentage difference in the RMSE between the ML (best-model) and the WEO forecasts.

While in general RMSEs deteriorate as expected, the ML based forecasting models still consistently outperform WEO forecasts. As before we calculate the accuracy improvement of the Super Learner model (the best performing ML technique) over the benchmark WEO performance and find, again, a high performance increase. The accuracy improvements are high, ranging from 49% (Mexico) – 82% (Spain) with the average increase being approximately 67% across all countries (higher than those obtained using a post-crisis forecast window). For an additional comparison, table 5 also reports the RMSEs of a standard autoregressive, Bayesian model averaging and vector autoregressive model which were estimated using the same data as described in section III. While some of these more traditional econometric models are able to outperform the WEO forecasts, they all succumb to the performances of the best-performing machine learning algorithm in each case. As an example to visualize the predictive power of the machine learning techniques, figure 6 below plots the out-of-sample fit for the United States over the period 2007-2016, i.e. including the crisis period. As can be seen from the figure, the model delivers a prediction close to the actual out-runs, tracking well both the downturn and the up-ticks in real GDP growth.

Further limitations in terms of causal inference prevail for most advanced machine learning algorithms where the significance of individual variables cannot be evaluated by employing standard statistical tests (Chakraborty and Joseph, 2017). The presented forecast errors and the underlying forecasts themselves are therefore to be considered as stand-alone point predictions

**Figure 6. USA Elastic Net Out-of-Sample Predictions Fit, including crisis period**



and the explanatory power of the estimated machine learning models are not yet assessed in the traditional econometric sense in this paper. Nonetheless, the algorithmic approach to prediction evidently presents a valuable addition to the field of economic forecasting and offers a myriad of different fields of application.

## V. Conclusion

We have applied three different machine learning algorithms to a common economic forecasting problem and show that this approach to statistical analysis may offer a valuable addition to traditional forecasting models. While the three machine learning algorithms were able to consistently outperform a benchmark forecast performance by the IMF's World Economic Outlook forecasts, a number of caveats and questions for further research remain. The low number of observations is a challenge that presents itself to most traditional macroeconomic variables in a machine learning context and future research should address questions in how high frequency data can be employed as proxies for more traditional macroeconomic indicators in order to fully benefit from the advantages that machine learning algorithms offer. Furthermore forecast performances are sensitive to training and test split specifications and generally machine learning algorithms are strongly driven by the underlying data. Moreover, depending on which algorithm is chosen, a number of degrees of freedom exist in form of algorithm parameters the calibration of which may be highly costly in terms of computational capacity. Finding effective ways to efficiently determine optimal parameter values is a question that the data science research community is currently working on and whose resolution may greatly facilitate the specification of optimally calibrated models.

Even so, the potential of machine learning and the algorithmic approach to statistical analysis for economic research questions is evident, especially given the continuous accumulation of data and the ubiquitous access thereof. In the specific context of our study, we believe that forecasts produced by machine learning algorithms can not entirely replace expert forecasting altogether, however might serve as a valuable additional benchmark when making discretionary decisions about a final economic forecast. Future research will seek to extend the coverage of economies (including low-income and developing economies) by using recent advances in the application of ML techniques to missing data (Che and others, 2018) and the use of lead-time indicators and auxiliary granular data sources ("big data") as proxies[14]. Adding policy variables, such as the enactment of monetary and fiscal policy measures during crisis period, to forecast a variety of policy scenarios, in addition to the baseline, could also be considered. Alternative specifications for the presented learner algorithms and their respective strengths in terms of computational complexity, depth and types of layers needed or regularization parameters are also worth exploring and a growing body of research is studying this interesting area within machine learning, such as e.g. Cook, Smalter Hall and others (2017). Methods that are able to unbox and interpret machine learning models to provide explanation for their outputs , and help understand the differences in forecast performance across a wide-range of model and expert-based forecasts, will also be explored (see for example, Ribeiro, Singh, and Guestrin (2016), Goldstein and others (2015) and Doshi-Velez and Kim (2017)).

[14] See for example, Henderson, Storeygard, and Weil (2012) for the use of satellite data on night lights to augment official income growth measures.

# VI. Appendix
# Further Details on the Methodology

### Example: Purchasing decisions modeled in a feed-forward neural network

As a practical example for a neural network, the following situation may be considered: When a purchase decision for a product represents our output variable, a number of relevant inputs are conceivable. For instance, factors such as the price of the product, the remaining stock of the product at home, the consumption experience this product offers in comparison to other, similar products etc. are most certainly relevant factors for the purchase decision at hand (see figure 7. For more price sensitive consumers, the weight (or importance) of the input variable "price" will be different than for a price indifferent consumer. Accordingly, different preferences might prevail concerning the other factors relevant in the purchase decision, leading to different weights (or importances) of the respective factors and different threshold values. In reality, decision-making or any kind of input and output relationship is much more complex than what a single perceptron could model. Instead, an entire network of perceptrons is a more realistic representation of real-life decision-making processes, resulting in so-called neural *networks*. In such networks, different layers of perceptrons are linked to each other in a whole system of neurons and determine an output of interest in a more complex manner.

**Figure 7. Neural Network Example**

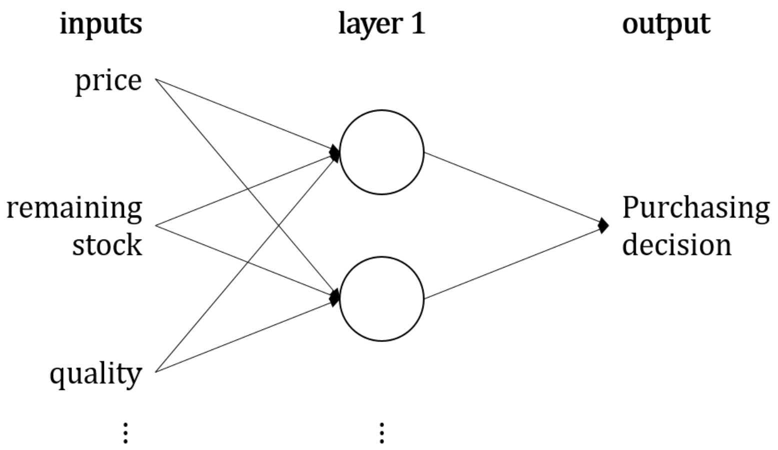


**Table 6. Super Learner Algorithm Library**

| Learner | Description |
|---|---|
| `bartMachine` | Bayesian additive regression trees |
| `bayesglm` | Bayesian generalized linear model |
| `biglasso` | extending Lasso model fitting to Big Data |
| `caret` | classification and regression training |
| `cforest` | conditional inference trees |
| `dbarts` | discrete Bayesian additive regression trees sampler |
| `earth` | multivariate adaptive regression splines |
| `extraTrees` | extremely randomized trees |
| `gam` | generalized additive models |
| `gbm` | gradient boosting |
| `glm` | generalized linear model |
| `glmnet` | Elastic Net |
| `ipredbagg` | improved bagging predictor |
| `interaction` | polynomial linear model |
| `kernelKnn` | kernelized k-nearest neighbor |
| `knn` | k-nearest neighbor |
| `ksvm` | kernel support vector machine |
| `lda` | latent Dirichlet allocation |
| `leekasso` | Lasso modification using only the top 10 predictors |
| `lm` | linear model |
| `loess` | local polynomial regression |
| `logreg` | logistic regression |
| `mean` | arithmetic mean |
| `nnet` | neural network |
| `nnls` | non-negative least squares |
| `polymars` | polynomial spline regression |
| `qda` | quadratic discriminant analysis |
| `randomForest` | random Forest |
| `ranger` | fast implementation of random Forests |
| `ridge` | ridge regression |
| `rpart` | recursive partitioning |
| `speedglm` | linear and generalized linear models for large data sets |
| `speedlm` | linear models for large data sets |
| `step` | stepwise regression |
| `stepAIC` | stepwise model selection by AIC |
| `svm` | support vector machine |
| `xgboost` | eXtreme gradient boost |

## References

Baldacci, Emanuele, Dario Buono, George Kapetanios, Stephan Krische, Massimiliano Marcellino, Gian Luigi Mazzi, and Fotis Papailias, 2016, "Big Data and Macroeconomic Nowcasting," Techn. rep.

Bang, James Thomas, Tinni Sen, and Atin Basuchoudhary, 2015, "New Tools for Predicting Economic Growth Using Machine Learning : A Guide for Theory and Policy," *Southern Economic Association Conference*.

Barto, Andrew G., and Thomas G. Dietterich, 2004, "Reinforcement learning and its relationship to supervised learning," *Handbook of Learning and Approximate Dynamic Programming*, pp. 47–64.

Biau, Olivier, and Angela D'Elia, 2010, "Euro Area GDP Forecasting Using Large Survey Datasets: A Random Forest Approach," *Euroindicators working papers*.

Blum, Adam, 1992, *Neural Networks in C++* (NY: John Wiley & Sons), 1st ed.

Boger, Zvi, and Hugo Guterman, 1997, "Knowledge Extraction from Artificial Neural Networks Models," in *IEEE Systems, Man, and Cybernetics Conference, Orland, FL, USA*.

Breiman, Leo, 2001, "Statistical Modeling: The Two Cultures," *Statistical Science*, Vol. 16, No. 3, pp. 199–231.

Broomhead, D. S., and D. Lowe, 1988, "Multivariable Functional Interpolation and Adaptive Networks," *Complex Systems*, Vol. 2, pp. 321– 355.

Chakraborty, Chiranjit, and Andreas Joseph, 2017, "Machine learning at central banks," *Bank of England Staff Working Paper No. 674*.

Chakraborty, Kanad, Kishan Mehrotra, Chilukuri K. Mohan, and Sanjay Ranka, 1992, "Forecasting the behavior of multivariate time series using neural networks," *Neural Networks*, Vol. 5, No. 6, pp. 961–970.

Che, Zhengping, and Sanjay Purushotham, 2017, "Recurrent Neural Networks for Multivariate Time Series With Missing Values," *International Conference on Learning Representations 2017*, pp. 1–15.

Che, Zhengping, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu, 2018, "Recurrent neural networks for multivariate time series with missing values," *Scientific reports*, Vol. 8, No. 1, p. 6085.

Cho, Kyunghyun, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, 2014, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*.

Chuku, Chuku, Jacob Oduor, and Anthony Simpasa, 2017, "Intelligent forecasting of economic growth for African economies: Artificial neural networks versus time series and structural econometric models," *IMF Workshop on Forecasting for Developing Economies*.

Chung, Junyoung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio, 2014, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,” *arXiv preprint*, pp. 1–9.

Clark, Todd E., and Michael W. McCracken, 2001, “Tests of equal forecast accuracy and encompassing for nested models,” *Journal of Econometrics*, Vol. 105, No. 1, pp. 85–110.

Connor, Jt, Rd Martin, and Le Atlas, 1994, “Recurrent Neural Networks and Robust Time Series Prediction,” *IEEE Transactions on Neural Networks*, Vol. 5, No. 2, pp. 240–254.

Cook, Thomas R, Aaron Smalter Hall, and others, 2017, “Macroeconomic Indicator Forecasting with Deep Neural Networks,” Techn. rep.

Debar, H, and B Dorizzi, 1992, “An application of a recurrent network to an intrusion detection system,” *IJCNN International Joint Conference on Neural Networks*, pp. 478–483.

Dietterich, Thomas G., 2000, “Ensemble Methods in Machine Learning,” *Multiple Classifier Systems*, Vol. 1857, pp. 1–15.

Dorffner, G., 1996, “Neural Networks for Time Series Processing.” *Neural Network World*, Vol. 6, pp. 447–468.

Doshi-Velez, Finale, and Been Kim, 2017, “Towards a rigorous science of interpretable machine learning,” *ArXiv*.

Elman, Jeffrey L., 1990, “Finding structure in time,” *Cognitive Science*, Vol. 14, No. 2, pp. 179–211.

Garcia Torres, Douglas, and Hongliang Qiu, 2018, *Applying Recurrent Neural Networks for Multivariate Time Series Forecasting of Volatile Financial Data*.

Genberg, Hans, and Andrew Martinez, 2014, “On the accuracy and efficiency of IMF forecasts: A survey and some extensions,” *IEO Background Paper No. BP/14/04 (Washington: Independent Evaluation Office of the IMF)*.

Goel, Hardik, Igor Melnyk, Nikunj Oza, Bryan Matthews, and Arindam Banerjee, 2016, *Multivariate Aviation Time Series Modeling: VARs vs. LSTMs*.

Goldstein, Alex, Adam Kapelner, Justin Bleich, and Emil Pitkin, 2015, “Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation,” *Journal of Computational and Graphical Statistics*, Vol. 24, No. 1, pp. 44–65.

Gregor, Karol, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra, 2015, “DRAW: A Recurrent Neural Network For Image Generation,” *arXiv preprint arXiv:1502.04623*.

Hastie, Trevor, Robert Tibshirani, and Jerome H. Friedman, 2004, *The Elements of Statistical Learning*.

Henderson, J Vernon, Adam Storeygard, and David Weil, 2012, “Measuring Economic Growth from Outer Space,” *American Economic Review*, Vol. 102, No. 2, pp. 944–1028.

Hochreiter, Sepp, and Jürgen Schmidhuber, 1997, "Long Short-Term Memory," *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780.

Hoerl, A., and R. Kennard, 1988, "Ridge regression," in *Encyclopedia of Statistical Sciences*, pp. 129–136.

Kamijo, Ken-ichi, and Tetsuji Tanigawa, 1990, "Stock price pattern recognition-a recurrent neural network approach," *IEEE Proc. of IJCNN*, pp. 215–221.

Kuan, C. M., and T. Liu, 1994, "Forecasting exchange rates using feedforward and recurrent neural networks," *Journal of Applied Econometrics*, Vol. 10, pp. 347–364.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton, 2015, "Deep learning," *Nature*, Vol. 521, No. 7553, pp. 436–444.

Mikolov, T, M Karafiat, L Burget, J Cernocky, and S Khudanpur, 2010, "Recurrent Neural Network based Language Model," *Interspeech*, pp. 1045–1048.

Nielsen, Michael, 2015, *Neural Networks and Deep Learning*.

Polley, Eric C., and Mark J. van der Laan, 2010, "Super Learner in Prediction," *U.C. Berkeley Division of Biostatistics Working Paper Series*, Vol. 266.

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin, 2016, "Why should i trust you?: Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135–1144 (ACM).

Rosenblatt, Frank, 1958, "The Perceptron : A Probablistic Model for Information Storage and Organization in the Brain," *Psychological Review*, Vol. 65, No. 6, pp. 386–408.

Rossi, Barbara, and Atsushi Inoue, 2012, "Out-of-Sample Forecast Tests Robust to the Choice of Window Size," *Journal of Business and Economic Statistics*, Vol. 30, No. 3, pp. 432–453.

Smeekes, Stephan, and Etiënne Wijler, 2018, *Macroeconomic Forecasting Using Penalized Regression*.

Tibshirani, Robert, 1996, "Regression Selection and Shrinkage via the Lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 58, No. 1, pp. 267–288.

Tiffin, Andrew, 2016, "Seeing in the Dark : A Machine-Learning Approach to Nowcasting in Lebanon," *IMF Working Paper*.

Timmermann, Allan, 2007, "An evaluation of the World Economic Outlook forecasts," *IMF Staff Papers*, Vol. 54, No. 1, pp. 1–33.

Tkacz, Greg, and Sarah Hu, 1999, "Forecasting GDP Growth Using Artificial Neural Networks," *Bank of Canada Working Paper No. 99-3*.

van der Laan, Mark J., Eric C Polley, and Alan E. Hubbard, 2007, "Super Learner," *Statistical Applications in Genetics and Molecular Biology*, Vol. 6, No. 1.

Varian, Hal R., 2014, "Big data: New tricks for econometrics," *The Journal of Economic Perspectives*, , No. June 2013, pp. 1–36.

White, Halbert, and San Diego, 1988, "Economic prediction using neural networks: the case of IBM daily stock returns," *IEEE International Conference on Neural Networks*, Vol. 2, pp. 451–458.

Zou, Hui, and Trevor Hastie, 2005, "Regularization and variable selection via the elastic net," *Journal of the Royal Statistical Society*, Vol. 67, No. 2, pp. 301–320.