Machine Learning vs Traditional Forecasting Methods: An Application to South African GDP

LISA-CHEREE MARTIN

Stellenbosch Economic Working Papers: WP12/2019

www.ekon.sun.ac.za/wpapers/2019/wp122019

August 2019

KEYWORDS: Machine learning, Forecasting, Elastic-net, Random Forests, Support Vector Machines, Recurrent Neural Networks
JEL: C32, C45, C53, C88

DEPARTMENT OF ECONOMICS
UNIVERSITY OF STELLENBOSCH
SOUTH AFRICA

UNIVERSITEIT
STELLENBOSCH
UNIVERSITY

BER
BUREAU FOR ECONOMIC RESEARCH

A WORKING PAPER OF THE DEPARTMENT OF ECONOMICS AND THE
BUREAU FOR ECONOMIC RESEARCH AT THE UNIVERSITY OF STELLENBOSCH

www.ekon.sun.ac.za/wpapers

Machine Learning vs Traditional Forecasting Methods: An Application to South African GDP

Lisa-Cheree Martin[1]

This study employs traditional autoregressive and vector autoregressive forecasting models, as well as machine learning methods of forecasting, in order to compare the performance of each of these techniques. Each technique is used to forecast the percentage change of quarterly South African Gross Domestic Product, quarter-on-quarter. It is found that machine learning methods outperform traditional methods according to the chosen criteria of minimising root mean squared error and maximising correlation with the actual trend of the data. Overall, the outcomes suggest that machine learning methods are a viable option for policy-makers to use, in order to aid their decision-making process regarding trends in macroeconomic data. As this study is limited by data availability, it is recommended that policy-makers consider further exploration of these techniques.

[1] lisamartin@sun.ac.za, Department of Economics, Stellenbosch University

Table of Contents

1 Introduction

Advances in technology and statistical learning have created an opportunity that should be taken advantage of. The improved forecasting and data analytics provided by machine learning techniques has become a useful tool, and one of great interest in various industries, and for good reason. These enhancements have the ability to bring about improved decision making for policy-makers, through the improvement and analysis of available information. Furthermore, with the rapid growth of the scale of data and information available, improvements in the speed and accuracy of decisions are vital.

Various applications of machine learning techniques have been considered for prediction in macroeconomics. Comparisons of the forecasting abilities of machine learning and traditional methods, have been undertaken for a number of variables. These include CPI, GDP, the price of oil, as well as the occurrence of a recession or currency crisis. Additionally, many macroeconomic time series are usually published with a time lag, a problem which could be overcome by machine learning techniques. For example, a machine learning algorithm may be capable of accurately predicting an upcoming data release, which eliminates the problem created by a lag. Moreover, these techniques do not have to be limited to domestic variables. Forecasting methods can be applied to data on foreign trade partners, among others.

The aim of the study is to discover how selected machine learning forecasting methods compare to traditional methods of prediction, when applied to South African time series. The study will forecast South African Gross Domestic Product (GDP), using both traditional and machine learning methods, in order to observe the resulting forecasting performance of each. The traditional techniques used to forecast the time series are Autoregression and Vector Autoregression, which will be compared to the machine learning techniques Elastic-net Regression, Support Vector Machines, Random Forests and Recurrent Neural Networks (RNN). The results will be compared to come to a conclusion about the relative performance of machine learning.

A review of available literature and a background to the study is presented in Section 2. This is followed by a description of the data and methodology of the study in Sections 3 and 4, respectively. This will entail an overview of the relevant South African data, which will be used in the study, as well as a description of each of the chosen traditional and machine learning techniques to be used. After the appropriate techniques are applied to the data, results will be presented and deliberated in section 5. A discussion of the outcome of the results and its implications for South African policy will be presented in Section 6. Finally, conclusions will be drawn in Section 7.

2 Literature Review

Machine learning is a branch of statistics that involves the use of a broad range of tools that can be used to understand data, usually for the purposes of prediction or forecasting (James et al. 2013: 1). Machine learning techniques have become increasingly popular in many fields of study in recent years, including Economics. A steady increase in the volume of data available, in addition to improvements in data processing and storage, has created the need to make use of more data-driven modelling techniques. The ability of machine learning to use large data sets to determine a generalizable pattern is what makes it an appealing technique. By fitting more flexible and complex functional forms to the data, machine learning techniques are often able to produce successful results in out-of-sample tests (Chakraborty and Joseph 2017: 1).

It is important to note, however, that no single technique will be useful for every task. Many econometric problems can be categorized by one of two goals: inference or prediction. In the case of inference, the estimation of parameters that underlie a relationship between the response and predictor variables is required. This parametric estimation approach is useful in understanding the expected change in a response variable, given a change in a predictor variable. The focus is on the estimation of the values of the parameters or coefficients of a model for which the functional form has been assumed (James et al. 2013: 21).

Machine learning algorithms are not meant to be used for this purpose (Mullainathan and Spiess 2017: 88). These algorithms do not make an assumption about the functional form and parameters of a model i.e. they are non-parametric estimation approaches. These often produce coefficient estimates that are inconsistent. Instead, the purpose of machine learning techniques is prediction of an output variable, or finding a structure within the data, based on certain input variables. In this case, the focus is not on interpreting the values of the specific parameters of the model, but rather on a specific estimated outcome (James et al. 2013: 23).

For the purposes of this paper, an emphasis is put on modelling supervised learning problems. This entails fitting a model with a defined response variable, for the purposes of either inference or prediction (James et al. 2013: 26). Many economic variables, for example prices, stocks, GDP, interest and foreign exchange rates, are observed over time. Of interest is not only the interrelationships between them, but also their relationships and trends over time. The analysis of these real-world economic time series is fundamental to a range of economic applications, including business cycle measurements, policy analysis and forecasting (Diebold 2017: 1).

While machine learning techniques potentially provide improved forecasts, there is a clear trade-off between predictive power and model interpretability (James et al. 2013: 25). Therefore, it can only provide an optimal solution when applied to a suitable problem. Econometricians, therefore, need to decide which is more important for the task at hand.

Policy makers require a comprehensive valuation of global or country-specific economic conditions. However, lags in the availability of important data mean that decision makers are forced to make assessments of economic conditions by looking at historical data. In some cases, it is possible to use traditional forecasting methods to estimate future conditions for decision-making. These methods include linear regression and vector autoregressive modelling, but these parametric methods still require assumptions about the model functional form, which becomes problematic as models become more complex (James et al. 2013 :104).

The increasing complexity of the world, and the rapidly growing volume of data, has resulted in models becoming more complex and requiring more accurate predictions, while dealing with the changing scale of "big data". Machine learning techniques are, thus, better suited for use in those scenarios in which prediction accuracy is more important than model interpretability. These techniques give emphasis to prediction accuracy through model validation and performance testing. This involves randomly splitting the data into a percentage of observations for model estimation, model calibration and model evaluation, to ensure consistent predictions based on the test sample (Chakraborty and Joseph 2017: 4).

Various machine learning techniques are available, each with varying strengths, weaknesses and suitability for different applications. With regards to forecasting macroeconomic data, several machine learning applications have been considered and continue to be studied across the globe. This stresses the importance of accurate forecasts of such data for policy makers, such as those at treasuries and central banks.

Lebanon is a prominent example of a country with major challenges to macroeconomic analysis. Policy makers face long lags in data releases, and without timely economic statistics, decisions are usually based on proxy measures. Specifically, GDP data are only gathered on an annual basis, with a publication delay of up to 2 years (Tiffin 2016: 3). While Lebanon is an extreme case of delayed data, this is still a key issue faced across countries.

Tiffin (2016) focusses on two popular machine learning techniques in an attempt to implement them in the Lebanese GDP estimation context, and provides acceptable out-of-sample forecasting results. The two methods used are an Elastic-net regression and the Random Forest algorithm, both of which are described as intuitive to most economists, as well as relatively easy to implement. The Elastic-net approach is a hybrid of the LASSO and Ridge Regression methods, while the Random Forest algorithm entails modifications to a type of decision tree approach.

Each method was used individually, using quarterly GDP data compiled for the years 1996 to 2010, and out-of-sample testing for the years 2011 to 2014. While the Elastic-net approach produced more accurate predictions, both machine learning procedures were discovered to provide a suitable method of GDP estimation that overcomes the challenges of the Lebanese data.

Chakraborty and Joseph (2017) use a framework for CPI inflation in the UK to forecast quarterly data over a medium term horizon. Publicly available data for the period 1998 to 2015 were used, and forecasts for inflation were made for two years in advance. The study uses the K-Nearest Neighbours (KNN), Regression Tree, Random Forest, Feed-Forward Artificial Neural Network (FFANN) and Support Vector Machine (SVM) approaches, which are then compared to more traditional approaches, used as references. These reference methods are the Ridge regression, Vector Autoregressive model and Autoregressive model with either 1 or p lags. The AR(1) results are used as a benchmark, with errors given relative to these results, for easier comparison.

In general, all reference methods were found to perform poorly in comparison to machine learning methods. The FFANN and SVM methods performed well when compared to the overall results, however, all methods’ performance deteriorated after the effects of the 2007/8 Global Financial Crisis. It is also worth noting that Random Forests and Neural Networks are more technically taxing and computationally challenging, especially as they increase in size (Chakraborty and Joseph 2017: 52-66).

However, Neural Networks (NN) have been successfully used by Choudhary and Haider (2012), who extend on earlier work by Nakamura (2006), McNelis and McAdam (2005), Binner et al. (2005), Binner and Gazely (2000) and Cameron et al. (1999). Each of these studies underline the forecasting potential of different approaches to Neural Networks.

The study builds on previous literature by evaluating the performance of multiple Artificial Neural Networks (ANN) techniques against the performance of a baseline Autoregressive model (AR(1)). The techniques are used to forecast the monthly inflation rates of 28 OECD countries. Performance is assessed by comparing the ability of the techniques to minimize Root Mean Squared Error (RMSE) or the Mean Absolute Percentage Error (MAPE).

The results indicate that Neural Network techniques are superior in terms of short-run forecasting. However, both Neural Networks and the AR(1) technique perform similarly for a large share of the countries. Finally, Choudhary and Haider (2012) find no conclusive evidence that either technique is dominant in long-run forecasting.

Given the strides made in machine learning, the potential for machine learning forecasting techniques should be considered by policy-makers. Currently, the South African Reserve Bank (SARB) makes use of two primary forecasting models (De Jager 2017). The first is the Reserve Bank’s core econometric model, which is a stylized error correction model. The second is the Quarterly Projection Model (QPM), which incorporates characteristics from dynamic stochastic general equilibrium (DSGE) modelling. However, in addition to these two models, the SARB makes use of various other models. This allows the bank to take advantage of the strength of multiple models, and make improved policy decisions.

The aim of this study is to explore the forecasting prospects of select machine-learning techniques. Should the results of the study suggest that machine learning methods have the potential to aid in the forecasting process, these techniques would naturally become candidates for additional models to be implemented by the SARB.

3 Data

The study aims to forecast the percentage change in quarterly South African GDP, one quarter into the future. The GDP and other data are all obtained from EasyData by Quantec. A total of 99 observations of each of the 15 variables were obtained, dating from Q2 1992 to Q4 2016, with the dependent variable *GDPPred* being the quarterly GDP observations from Q3 1992 to Q1 2017.

All variables used, with variable names in brackets, are the quarterly percentage change in Gross Domestic Product (*GDP*, *GDPPred*), Household Real Disposable Income (*HRDI*), Total Headline Consumer Price Index (*CPI*), Gold price in US Dollars (*gold*), Oil price in US Dollars (*oil*), Average Wage Rate (*wage*), M3 Money Supply (m3), South African Repo Rate (*repo*) and the Production Price Index (*ppi*), sourced from the Quantec Quarterly Forecast. Furthermore, the Primary Commodity Price Index (*comm*) and Net Realized Manufacturing Production (*man*) are sourced from International Monetary Fund and Bureau for Economic Research, respectively. Finally, the Terms of Trade (*tot*), Gross Value Added by the Tertiary Sector (*tertiary*) and National Government Expenditure (*govt*) are sourced from the SARB Quarterly Bulletin.

Table 6 displays the summary statistics of each of the variables used, and a correlation table of the variables used is presented below in Table 7 (see Appendix).

4 Methodology

The study will use programming in R to make a comparison between traditional and machine learning methods of forecasting South African GDP data. R is suitable for statistical programming, and has useful packages available online to help the process, and lessen the amount of preliminary coding that will need to be done. Where applicable, the data will be split into training and test data sets. The test sets will include sets for the periods prior to, after, and including the global financial crisis (GFC), as the crisis could impact the forecasting ability of the technique negatively.

Similarly to Chakraborty and Joseph (2017), the macroeconomic series will initially be forecasted using the traditional autoregressive and vector autoregressive approaches. Based on the case studies of Chakraborty and Joseph (2017), Tiffin (2016) and Choudhary and Haider (2012), four possible machine learning approaches that will be considered are the Elastic-net regression, Random Forests, Support Vector Machines (SVMs) and Recurrent Neural Networks (RNN). These popular techniques each provide some evidence of improved forecasting in the literature.

4.1 Traditional Forecasting: Autoregressive and Vector Autoregressive

Firstly, an Autoregressive model will be applied to the series. This is basically the process of regressing a variable on the lagged observation from that specific time series. An autoregressive model for p lags can be defined as:

$$Y_t = c + \sum_{i=1}^{p} \gamma_i Y_{t-i} + \varepsilon_t$$

where Y_{t-1}, $Y_{t-2}, \ldots,$ Y_{t-p} are the series' lagged values, γ_i to γ_p are parameters of the model, c represents a constant and ε_t represents white-noise.

A Dickey-Fuller test can be used to test for the presence of non-stationarity within the data. The outcomes of this test will aid in choosing the number of lags, i.e. the value of p in the above equation. The choice of the number of lags is decided based on the Akaike Information Criterion, with indicates that an order of two if required. We therefore use an AR(2) model.

Second, a Vector Autoregressive (VAR) model will be used. The VAR model generalizes an AR (1) model by allowing multiple evolving variables. The VAR model will be a multivariate model applied to the data, allowing for the determination of interdependencies among the data. A basic VAR(p) model can be defined as:

$$y_t = (y_{1t}, \ldots, y_{kt}, \ldots, y_{Kt})$$

$$y_t = A_1 y_{1t-1} + \cdots + A_p y_{t-p} + u_t$$

where y_t is a set of K endogenous variables, A_i are ($K \times K$) coefficient matrices for i = 1,..,p and u_t is a K-dimensional white-noise process with an expected value of zero. The number of lags of the VAR model will be limited to 1, due to the large number of parameters and the length of the time-series used.

4.2 Elastic-net Regressions

The ordinary least squares coefficient estimates are obtained by using a (multiple) linear regression model, which minimizes

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_{ij}\right)^2$$

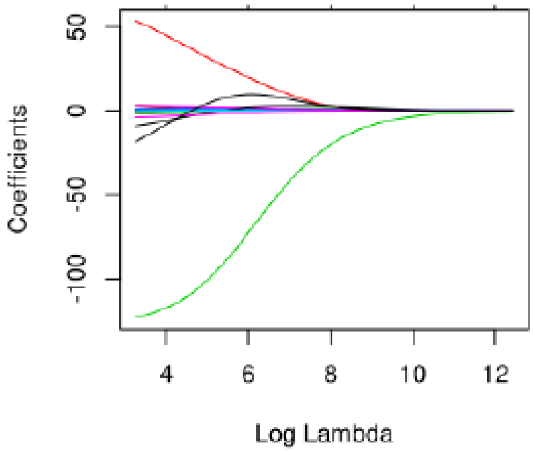
where $y_1, \dots, y_n$ are the dependent variable, $x_{11}, \dots, x_{np}$ are the independent variables and $\beta_0, \beta_{11}, \dots, \beta_{np}$ are the resulting beta coefficient estimates.

The Elastic-net regression is made up of a hybrid of the LASSO and Ridge regressions. These are both known as shrinkage methods. Each of these methods involves using least squares to fit a linear model, but with the addition of constraining or regularizing the coefficient estimates in such a way that *shrinks* them towards zero. This can significantly reduce the coefficient variance; however, it will result in a slight bias of the coefficient (James et al., 2013). This is a reasonable trade-off when the response and predictor variables have a relatively linear relationship, as the model will be less sensitive to changes in the data.

The Ridge regression differs from the least squares in that it aims to minimize the residual sum of squares, but it also minimizes $\sum_{j-1}^{p} \beta_j^2$, called the shrinkage penalty. This penalty is small when the coefficients are close to zero, which is why it pulls the estimates toward zero. A Ridge regression will produce a set of coefficients for each value of λ, the tuning parameter, which controls the relative impact of the two terms to be minimized on the coefficient estimates. Therefore, selecting a good λ is crucial, and done using cross-validation.

Figure 1 illustrates the coefficient estimates, obtained by a ridge regression, for a hypothetical data set (Dalpiaz, 2017). Prior to any regularization taking place, λ is equal to zero, and the corresponding regression coefficient estimates are the same as the OLS estimates. However, as λ is increased, the regression coefficient estimates shrink towards zero, until the model becomes null as λ tends to an extremely large value and the model has no predictors.

Figure 1: Ridge Regression Coefficient Estimates for Changing Values of Lambda

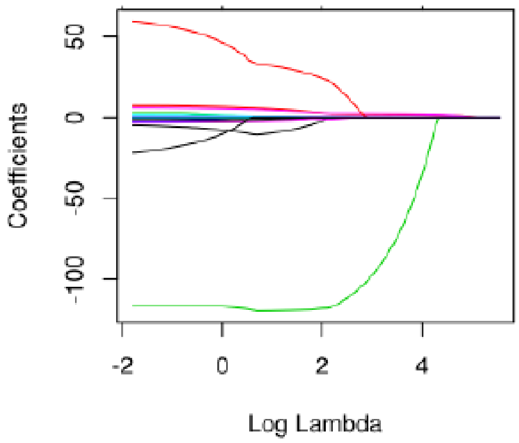


Source: Dalpiaz, 2017

The LASSO (Least Absolute Shrinkage and Selection Operator) has a different penalty. The shrinkage penalty is equal to $\lambda \sum_{j-1}^{p} |\beta_j|$ in this case, and has the added advantage of being able to force certain coefficients to equal zero. This advantage over the Ridge regression allow the LASSO to have variable selection properties, making the model more interpretable, but not necessarily more accurate.

Figure 2 illustrates the coefficient estimates, obtained by a LASSO regression, for a hypothetical data set (Dalpiaz, 2017). Prior to any regularization taking place, and λ is equal to zero, and the corresponding regression coefficient estimates are the same as the OLS estimates. However, as λ is increased, the regression coefficient estimates shrink towards zero. Unlike the Ridge regression coefficients, notice that certain coefficients are forced to equal zero.

Figure 2: LASSO Coefficient Estimates for Changing Values of Lambda



Source: Dalpiaz, 2017

The Elastic-net regression uses both approaches by minimising a combined penalty in the following way:

$$\sum_{t-1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}x_{ij}\beta_{ij}\right)^2+\lambda\sum_{j-1}^{p}[(1-\alpha)(\beta_j^2)+\ \alpha|\beta_j|]$$

where $y_1,\dots,y_n$ are the dependent variable, $x_{11},\dots,x_{np}$ are the independent variables, $\beta_0,\beta_{11},\dots,\beta_{np}$ are the resulting beta coefficient estimates, λ is the regularization tuning parameter, which shrinks the coefficients and α is a weighting tuning parameter, indicating the weight of each approach used within the Elastic-net.

In this manner, it selects the best predictors, while still maintaining the Ridge regression's ability to identify closely correlated predictors. Note that α is another tuning parameter that requires careful selection through a validation process. The validation process is intended to choose a combination of tuning parameters that minimize prediction error, based on the in-sample data.

The cross-validation process used to select the optimal α and λ values for the elastic-net regression is conducted using functions from the *caret* package in R, which uses the minimization of the Root Mean Squared Error to select the optimal model values. The resulting tuning parameter values are as follows:

Table 1: Elastic-net Tuning Parameters

Alpha	Lambda
0.00	0.87

Source: Author’s own calculations, using R package *caret*

4.3 Random Forests

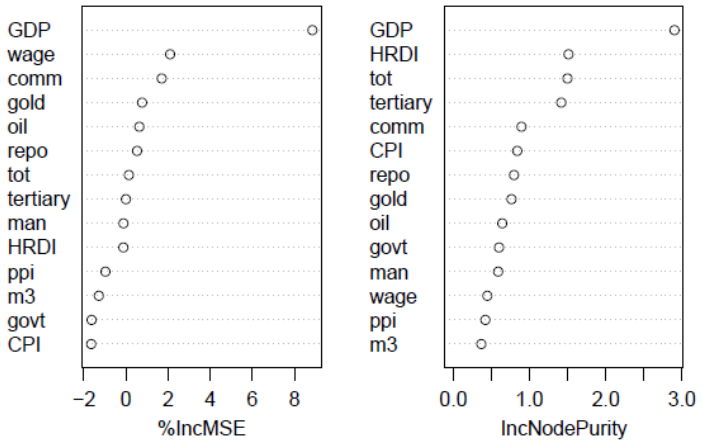
Basic tree-based methods are simple, but do not produce high prediction accuracy. A basic decision tree starts at an initial node. The tree will continue to branch out according to the number of predictor variables. The possible outcomes, or predictions, are found at the end of each branch. Depending on the observation data, and the branches of the tree that have their conditions satisfied, a predicted outcome will eventually be reached. This can be thought of as similar to a flow diagram.

By allowing for multiple trees, and producing multiple outcomes, a Random Forest can improve the accuracy of the predicted outcome (James et al. 2013: 319). First, multiple trees need to be created by bootstrapping the data. This is done by taking repeated samples from the initial single sample, thus producing multiple samples to train multiple trees with. However, if there exists one strong predictor within the sample, each tree will inevitably have the same top split. This will result in the trees being highly correlated, which adds to the variation within the model, instead of reducing it (James et al. 2013: 320).

An important requirement for a successful Random Forest is that the trees not be correlated, thus not allowing one single strong predictor to weigh too heavily on the outcome (James et al. 2013: 320). In order to achieve this, each tree is trained to be unique. For each tree, whenever a split into a new branch is considered, a random sample of *m* out of the *p* available predictor variables is chosen, and the split is only allowed to use one of the predictors in the subset (the value of *m* is usually approximately$\sqrt{p}$). In this way, the predicted outcome of each tree is uncorrelated with any other tree in the forest. We can then average across all of our predictions for a given observation to obtain a more accurate prediction than if only one tree was used. Our resulting average, obtained from an aggregation of the forest’s results, will be our predicted value for the problem.

Again, R packages are available to assist in the execution of the model. A Random Forest consisting of a total of 500 trees, with an *m* value of 4 predictors per subset is used for the training data set ($\sqrt{14}$ = 3.7). Figure 3 reports on two measures of variable importance of the given data and model. The former is a measure of the mean decrease of predictive accuracy of the model, when a given variable is excluded from the model. The latter measures the total decrease in node impurity as a result of a split over that specific variable, as an average over the total number of trees in the forest.

Figure 3: Variable Importance plot of the data



Source: Author's own calculation, using R package *randomForest*

4.4 Support Vector Machines

An SVM is an algorithm which can be used for either classification or regression. Since the study will be trying to predict an actual number value, an SVM regression technique should be used. The basic idea behind an SVM regression is to find a function that deviates from each given training observation by a value no larger than given error, ε.

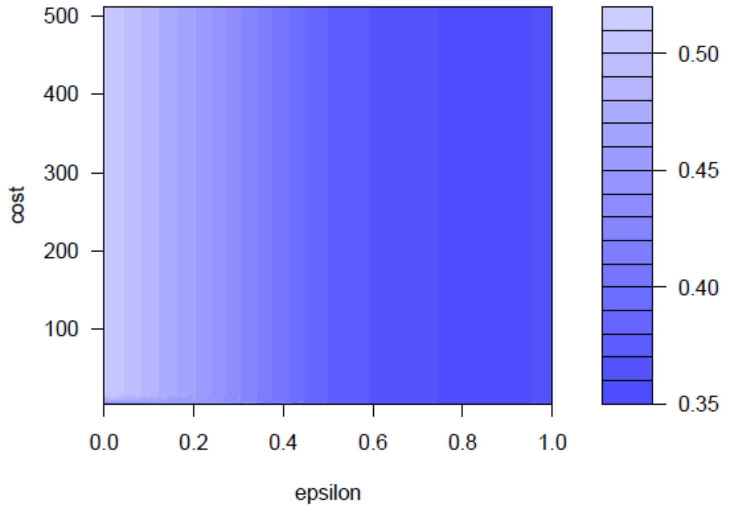
This is essentially a minimization problem, with the SVM regression finding the regression coefficients that minimize the error. However, it should be noted that part of the error is tolerated by the minimization constraints. This is done by adding values to the objective function of the problem that the regression errors are allowed to reach.

What makes SVM regression more flexible than other methods, is that the linear solution to the minimization problem is represented in dual form, not primal. In this manner, the model is represented by combinations of the points of the training data. These points, which are used within the model solution, are the support vectors (James et al., 2013).

Furthermore, SVM regression allows models that cannot be adequately described using a linear function, by extending the dual form with the use of a non-linear kernel function. This is a function which converts the given model data to a higher dimensional feature space. This would allow typically linear observations to be transformed into non-linear dimensions for use by the SVM. However, in either case, the result is a function that is used to predict new values.

In order to ensure optimum performance of the model, tuning parameters need to be selected. Tuning the model selects parameters for ε and the *cost* parameter. The *cost* parameter is used to reduce overfitting of the model. The R package available for SVM regression has a function that tests the performance of a number of combinations of ε and the *cost* parameter, and selects the best one. Figure 4 illustrates the performance of the model for these different combinations of ε and the *cost* parameter. The RMSE is closer to zero in the darker regions of the figure, and therefore represents better performance by the model.

Figure 4: Performance of SVM for combinations of epsilon and cost parameters



Source: Author's own calculations, using R package *e1071*

The chosen tuning parameters, which result in the best performance of the model, are as follows:

Table 2: SVM Tuning Parameters

Epsilon	Cost
0.80	4.00

Source: Author's own calculations, using R package *e1071*

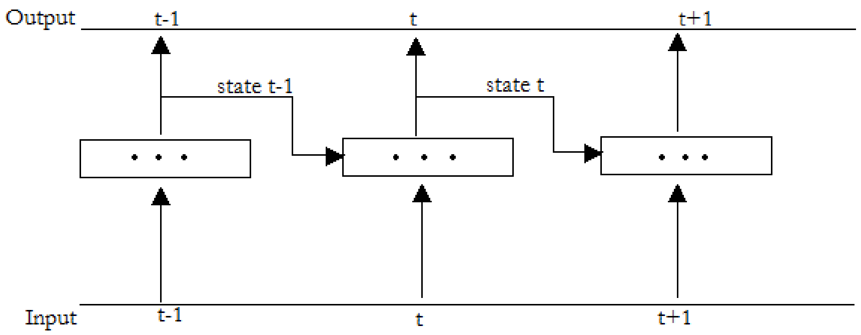
4.5 (Recurrent) Neural Networks

Neural Networks generally have no memory (Chollet and Allaire, 2018). What this means is that the algorithm processes each input that it receives, independently of the information received in any other state. For sequence processing, this would require the sequence being analysed to be given as a single vector input at every point in time. This single set of information will then be processed at once.

Unlike these *feedforward networks*, the *Recurrent Neural Network* (RNN) iterates through the given sequence, and maintains a *state* in which it stores relevant information that it has collected thus far in its analysis (Chollet and Allaire, 2018).This type of algorithm is especially applicable to time-series forecasting, as it internally loops through each sequence it receives. This allows the RNN to reuse the outputs, which it has previously produced, as an input to the algorithm.

Figure 5 illustrates this process. The algorithm produces an output at time *t-1*, which is stored in *state t-1*. This *state t-1* variable is then used as one of the inputs for time *t*, which produces another output. This output at time *t* is then stored in *state t* and used as an input for the algorithm at time *t+1*.

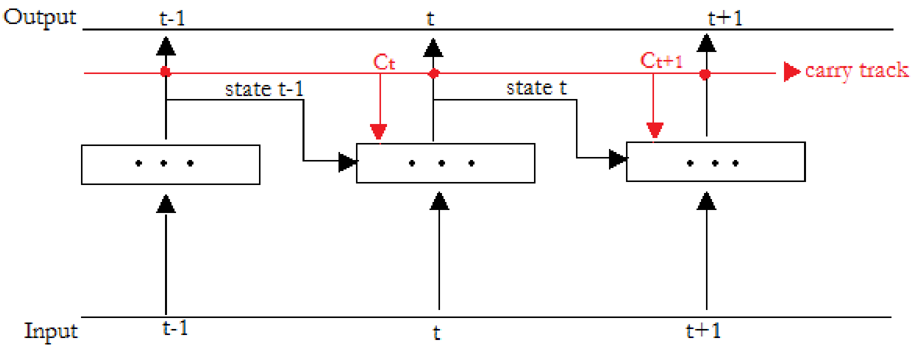
Figure 5: Recurrent Neural Network Process



Source: Author's own illustration, based on Chollet and Allaire, 2018

The above process represents what is called a Keras layer, which takes inputs about the shape of the data, and produces the necessary output per step in the given sequence. While the above representation is simple to understand, and could theoretically retain the information produced multiple periods before, it cannot actually achieve this in practice. To counter this we can make use of a Long Short-term Layer (LSTM) layer. Unlike the simple process presented in Figure 5, which stores the *t-1* period's output in *state t* i.e. only one period into the future, the LSTM layer can transport an output produced at any time along the sequence, more than 1 period into the future (Chollet and Allaire, 2018). This is illustrated by the *carry track* in Figure 6. *Ct* can be transferred along the *carry track* to both time *t*, as well as time *t+1*.

Figure 6: Recurrent Neural Network Process with an LSTM Layer



Source: Author's own illustration, based on Chollet and Allaire, 2018

5 Results

This section discusses the results of the different forecasting techniques applied to the GDP data. The performance metrics of each of the applied methods will be discussed. These metrics will be analysed for each of the pre-crisis, post-crisis and full test periods. In addition to this, the forecasted values of each of the techniques will be graphed and compared to the actual GDP values over the forecast period.

Each of the tables provide a comparison between each of the forecasting techniques, based on Root Mean Squared Error (RMSE), standard deviations (SD) and correlations with actual values (Corr). RMSE is chosen since it would give a relatively high weight to larger deviations than an alternative such as Mean Absolute Error. This is because it squares the errors before averaging and taking the root of the total error. The autoregressive and vector autoregressive models represent the traditional reference models, while the rest of the models are compared to these baseline results.

The autoregressive model is performed with two lags (AR(2)), as specified according to the Akaike Information Criterion (AIC). The vector autoregressive model is restricted to one lag, due to the large number of parameters. The Elastic-net model is implemented with an α of zero and a λ of 0.87. This means that the Elastic-net model uses only a Ridge regression to build the model, but is not necessarily the weight that would be applied to a different dataset. The Random Forest consists of 500 trees, with an *m* value of 4. The SVM is implemented with an ε of 0.8 and a cost parameter of 4. Finally, the RNN is implemented with two LSTM layers. As most of the parameter selection is done using automated processes from R packages, and model tuning and parameter selection is not the focus of the paper, the selection processes will not be further discussed.

Figures 7 to 12 graphically represent the predicted and actual GDP values over the entire forecasting period. Each graph consists of a black line illustrating the actual quarterly to changes in GDP, and a line of a different colour, which represents the forecasts of the model. The grey vertical lines on each graph indicate the beginning and end of the GFC, with the bankruptcy of Lehman Brothers chosen as the beginning of the crisis.

Table 3 displays the performance metrics of each of the forecasting techniques for the pre-crisis period from Q2 2002 to Q2 2008. Pre-crisis, the best performing model is the Elastic-net, which produces the lowest RMSE and the highest correlation with the GDP for the period. The Random Forest performs second best, while the VAR(1), SVM and AR(2) each produce a relatively close RMSE value. However, no other model is able to produce a correlation coefficient to match the Elastic-net.

Table 3: Pre-crisis Model Performance Metrics

	RMSE	SD	Corr
AR(2)	0.61	0.04	0.03
VAR(1)	0.59	0.04	0.03
Elastic-net	0.45	0.21	0.22
Random Forest	0.53	0.19	-0.02
SVM	0.61	0.15	-0.01
RNN	1.54	1.43	-0.1

Source: Author’s own calculations, using multiple R packages

The pre-crisis forecasts of the AR(2) and VAR(1) models are presented in Figures 7 and 8. Both indicate an initial forecast that appears to move in the direction of actual GDP, before reverting to a flat trend, which continues for the remainder of the period. The forecast line of the Elastic-net model follows the actual GDP line more closely, while the Random Forest, SVM and RNN forecasts fail to capture the trend of GDP between 2001 and 2007 (see Figures 9-12) .

During the actual crisis period, between Q3 2008 and Q2 2009, each of the machine-learning models successfully predict a negative percentage change in GDP. However, these predictions differ in magnitude, and the models all either under- or over-predict the change in GDP during this time. The Elastic-net and RNN produce the best predictions for this period, while the baseline AR(2) and VAR(1) model predictions remain flat.

Table 4 displays the performance metrics of each of the forecasting techniques for the post-crisis period from Q3 2009 to Q4 2016. Interestingly, all of the models except the Elastic-net have an improved RMSE and correlation coefficients for the post-crisis period. However, the Elastic-net continues to produce the highest correlation coefficient, while still competing in terms of RMSE. All models, including the Elastic-net, also experience an improvement in their standard deviations. These results are unanticipated, since the impact of the crisis was expected to weaken the forecasting performance of the models, as it did in Chakraborty and Joseph (2017).

Table 4: Post-crisis Model Performance Metrics

	RMSE	SD	Corr
AR(2)	0.49	0.01	0.11
VAR(1)	0.51	0	0.11
Elastic-net	0.51	0.17	0.49
Random Forest	0.49	0.16	0.23
SVM	0.46	0.1	0.21
RNN	0.87	0.72	-0.06

Source: Author’s own calculations, using multiple R packages

With regards to the graphed forecasts, none of the predicted GDP series appear to have improved, despite the changes observed in the performance metrics of the models. Most concerning are the improved correlation and RMSE of the baseline models, despite the fact that they follow a flat trend. Graphically, the Elastic-net still appears superior to the competing models, as it manages to predict the peaks of the actual GDP trend relatively well. However, the RNN model forecasts appear most closely matched to the actual post-crisis GDP trend from 2014 onwards. These observations emphasize the fact that the performance metrics cannot be considered sufficient evidence of the actual performance of the models, and must be accompanied by additional analysis, such as the graphical representations of the forecasts against the actual data.

Table 5 displays the performance metrics of each of the forecasting techniques for the entire testing period from Q2 2002 to Q4 2016. Over the entire testing period, the machine-learning models outperform the traditional AR(2) and VAR(1) models. The Elastic-net model is the best performing model, according to the performance metrics, with the lowest RMSE and highest correlation. The Random Forest and SVM models perform equally, to a certain degree, while the RNN model appears to make predictions that are excessively volatile. This could be due to the nature of South Africa’s GDP itself, as it is influenced by a number of international commodity and asset markets.

Table 5: Full Testing Period Performance Metrics

	RMSE	SD	Corr
AR(2)	0.64	0	0.01
VAR(1)	0.65	0.02	0.01
Elastic-net	0.55	0.24	0.66
Random	0.58	0.21	0.48
SVM	0.6	0.15	0.42
RNN	1.21	1.1	0.11

Source: Author's own calculations, using multiple R packages

Overall, the machine learning techniques perform well when compared to the baseline models. The Elastic-net model exhibited the best results in terms of performance metrics and observations of forecast graphs. The other machine learning methods also demonstrate potential for use in forecasting activities. It should be noted that the study outcomes are limited by the availability of data and some machine learning techniques require longer time-series to provide improved results.

6 Policy Recommendations

The South African Reserve Bank currently makes use of two primary forecasting models (De Jager, 2017). The first is the Reserve Bank's core econometric model, which is a stylized error correction model. The second is the Quarterly Projection Model (QPM), which incorporates characteristics from dynamic stochastic general equilibrium (DSGE) modelling. However, in addition to these two models, the SARB makes use of various other models. This allows the bank to take advantage of the strength of multiple models, and make improved policy decisions.

Similar to reserve banks in general, the SARB has to make policy decisions while dealing with data release lags. In addition to this, traditional forecasting models face three challenges (Kabundi et al., 2016). First, policy data tend to be irregular in their release patterns, which can sometimes lead to gaps in the data, especially towards the end of the dataset. Second, models require data with a frequency which matches the model, for example, a quarterly model would require quarterly data. Lastly, present day econometric models may not be able to process the larger sets of data, which are becoming more common with the introduction of big data.

Specifically, VAR models are meant to improve on the forecasting power of autoregressive models. However, VAR and other structural models can easily be misspecified, and are prone to forecasting failure caused by omitted variables or structural breaks (Aron and Muellbauer, 2002). Furthermore, these models are currently or have previously been implemented by the SARB to aid in the forecasting process of both GDP growth and inflation. Unless recognised, the weaknesses of these models can negatively impact the quality of policy decisions.

Since machine learning techniques are more flexible than traditional techniques, the former will allow policy-makers to act regardless of the first two challenges faced by traditional forecasting. Furthermore, machine learning methodologies have been developed and interest in the field has grown with the introduction of big data. It is therefore better suited to deal with larger, more complex datasets.

The literature on machine learning applications done on South African GDP data appears limited. However, Kabundi et al. (2016) performs a study which has implications for South African policy makers. The 2016 study which compares a nowcasting technique and other traditionally used forecasting techniques by using each method to forecast South African GDP. The models used include Autoregressive and Vector Autoregressive models. The study finds that the nowcasting method performs relatively well against the alternative, traditional methods. The study outcomes suggest that newer techniques, such as machine learning methods for forecasting, can be useful to reserve banks and should be considered for forecasting purposes.

Similarly, the results produced by this study have indicated that machine learning methods have the potential to aid in the forecasting process, and these techniques should be considered as additional models to be implemented by the SARB. Since the SARB makes use of multiple models, in addition to its two core models, it is a reasonable suggestion that the Bank should investigate the advantages and possible benefits it would gain from the machine learning techniques. While the study only demonstrates the forecasting ability of these models on a small scale, it nevertheless identifies a growing field and literature, which warrant further investigation of these methods.

7 Conclusion

Advances in technology and statistical learning have created an opportunity that should be taken advantage of. The improved forecasting and data analytics provided by machine learning techniques has become a useful tool, and one of great interest in various industries. The enhancements to data modelling brought forth by machine learning have the ability to bring about improved decision making for policy-makers. Furthermore, with the rapid growth of the scale of data and information available, improvements in the speed and accuracy of decisions are vital.

Various applications of machine learning techniques have been considered for prediction in macroeconomics. Comparisons of the forecasting abilities of machine learning, as opposed to traditional methods, has been undertaken for a number of variables. The aim of this study was to discover how selected machine learning forecasting methods compare to traditional methods of prediction when applied to a South African GDP forecasting problem. The study made use of both traditional and machine learning methods to observe the resulting forecasting performance of each, given the South African dataset.

The study first presented a review of the available literature as a background to the field of machine learning, as applied to these forecasting problems. Following this, the data and methodology were outlined. This included a discussion on each of the chosen forecasting techniques, both traditional and machine learning. These forecasting techniques are Autoregressive, Vector Autoregressive, Elastic-net regression, Random Forests, Support Vector Machines (SVMs) and Recurrent Neural Networks (RNN). Following the implementation of each of these techniques, the results have been presented, and the policy implications of these outcomes are that there is potential for the use of machine learning techniques for forecasting.

Over the entire testing period from 2002 to 2016, the machine-learning models outperformed the traditional models. The Elastic-net model produced the best performance, in terms of performance metrics, with the lowest RMSE and highest correlation. The Random Forest and SVM models performed equally well, compared to each other, but relatively well compared to traditional methods. The RNN model was the only machine learning method that did not produce performance metrics which outperformed the traditional methods.

Overall, the machine learning techniques demonstrate the potential to be used for forecasting purposes that may improve policy decision-making and the forecasting process surrounding it. Therefore, these techniques should be considered as additional models to be implemented by the SARB. It should be noted, however, that the study outcomes are limited by the availability of data and some machine learning techniques require longer time series to provide improved results. While the study only demonstrates the forecasting ability of these models on a small scale, it nevertheless identifies a growing field and literature, which warrant further investigation.

8 References

Aron, Janine, and John Muellbauer (2002). 'Interest Rate Effects on Output: Evidence from a GDP Forecasting Model for South Africa.' *IMF Staff Papers* 49(1): 185–213.

Chakraborty, Chiranjit, and Andreas Joseph (2017). 'Machine Learning at Central Banks.' *Bank of England, Working Paper 674*.

Chollet, François, and Joseph J Allaire (2018). *Deep Learning with R*. Manning Publications Company.

Choudhary, M Ali, and Adnan Haider. (2012). 'Neural Network Models for Inflation Forecasting: An Appraisal.' *Applied Economics* 44(20): 2631–5.

Dalpiaz, David. (2017). 'R for Statistical Learning.' Available at: https://daviddalpiaz.github.io/r4sl/index.html. (accessed 15 November 2018).

De Jager, Shaun. (2017). 'Comparing the SARB's Quarterly Projection Model to the 'Core' Macro-Econometric Model.' *Macro Models Unit, Policy Development Wing, South African Reserve Bank*.

Diebold, Francis X. (2017). 'Time-Series Econometrics: A Concise Course.' *University of Pennsylvania, Pennsylvania*.

James, Gareth, Daniela Witten, Trevor Hastie, and Robert Tibshirani. (2013). *An Introduction to Statistical Learning*. Springer.

Kabundi, Alain, Elmarie Nel, Franz Ruch, and others. (2016). 'Nowcasting Real GDP Growth in South Africa.' *Economic Research Southern Africa, Working Paper 581. Cape Town: ERSA*.

Mullainathan, Sendhil, and Jann Spiess. (2017). 'Machine Learning: An Applied Econometric Approach.' *Journal of Economic Perspectives* 31(2): 87–106.

Quantec (2018). *Easydata Research Domain*. Software and database. Pretoria.

Tiffin, Andrew. (2016). 'Seeing in the Dark: A Machine-Learning Approach to Nowcasting in Lebanon.' *IMF Working Paper, WP/16/56*.

9 Appendix

Table 6: Summary statistics of each of the variables included in the study

	GDPPred	GDP	HRDI	CPI	gold	oil	wage	comm	ppi	repo	m3	man	tot	tertiary	govt
Observations	99	99	99	99	99	99	99	99	99	99	99	99	99	99	99
NAs	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Minimum	-1.56	-1.56	-4.62	-2.06	-13.29	-51.66	-3.15	-40.52	-4.41	5	-1.09	-2100	-5.17	-0.36	-16.72
Quartile 1	0.29	0.29	0.07	0.87	-2.43	-6.04	1.09	-2.88	0.43	6.92	1.61	-71.01	-1.22	0.43	-7.97
Median	0.74	0.74	0.72	1.49	0.45	2.22	2.18	1.27	1.59	10.5	2.67	-22.5	0.25	0.73	0.02
Arithmetic Mean	0.68	0.67	0.74	1.5	1.43	2.08	2.15	1.12	1.63	10.34	2.92	-33.58	0.27	1.04	3.43
Quartile 3	1.11	1.11	1.46	2.05	4.2	9.67	2.96	6.19	2.66	13.08	4.01	52.7	1.91	1.64	12.2
Maximum	1.87	1.87	6.14	4.24	17.53	38.2	9.52	18.09	10.43	21.69	9.26	500	6.74	3.57	38.35
Variance	0.4	0.41	2.96	0.97	35.3	199.11	3.89	70.9	3.84	16.73	4.13	79761	6.07	0.7	186.18
Standard dev	0.64	0.64	1.72	0.98	5.94	14.11	1.97	8.42	1.96	4.09	2.03	282.42	2.46	0.84	13.64
Skewness	-0.74	-0.75	0.07	0.02	0.65	-0.4	0.93	-1.33	0.65	0.43	0.66	-4.36	0.13	0.73	0.58
Kurtosis	0.96	0.88	2.56	1.36	0.27	1.57	3.48	4.95	3.69	-0.75	0.42	28.56	0.02	-0.23	-0.57

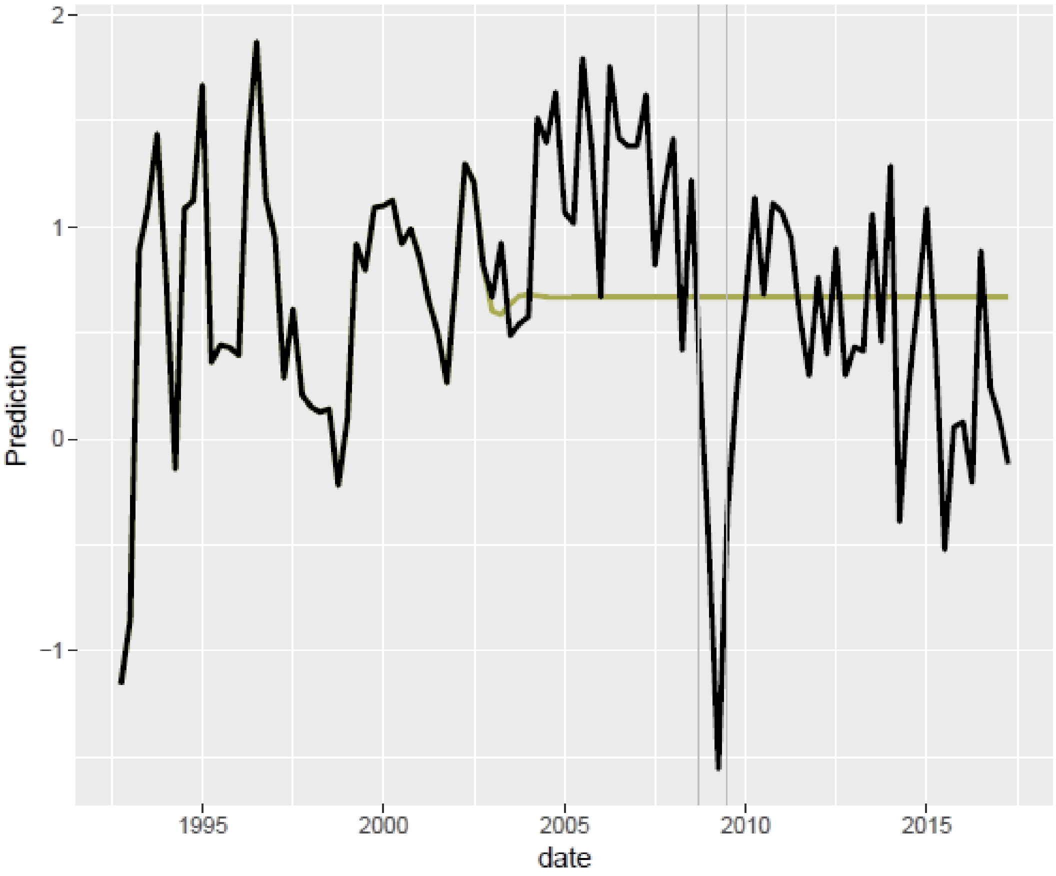
Source: Author’s own calculations, using data from Quantec (2018)

Table 7: Correlation table (matrix) of each of the variables included in the study

	GDPPred	GDP	HDRI	CPI	gold	oil	wage	comm	ppi	repo	m3	man	tot	tertiary	govt
GDPPred	1														
GDP	0.57	1													
HDRI	0.31	0.27	1												
CPI	-0.16	-0.17	-0.23	1											
gold	0.29	0.11	-0.03	-0.04	1										
oil	0.31	0.33	-0.01	0.09	0.23	1									
wage	0	0.04	-0.02	0.24	-0.07	0.05	1								
comm	0.39	0.38	0.04	0.11	0.32	0.94	0.1	1							
ppi	0.02	0.16	0.01	0.46	-0.12	0.27	0.05	0.26	1						
repo	-0.12	-0.13	-0.09	0.36	-0.19	-0.05	0.16	-0.09	0.12	1					
m3	0.17	0.32	0.12	0.03	0.09	0.04	0	0.06	0.08	0.19	1				
man	0.16	0.03	0.09	0.03	-0.05	0.1	0.14	0.09	0.01	-0.15	0.01	1			
tot	0.02	0.01	0.02	-0.22	0.25	-0.15	-0.1	-0.06	-0.4	-0.09	0.07	-0.19	1		
tertiary	0.38	0.52	0.25	-0.16	0.19	0.17	-0.05	0.17	-0.03	0.01	0.41	-0.01	0.1	1	
govt	0.11	0	-0.02	-0.03	0.04	-0.02	0.09	0	-0.41	0.01	0.16	0.02	0.23	0.19	1

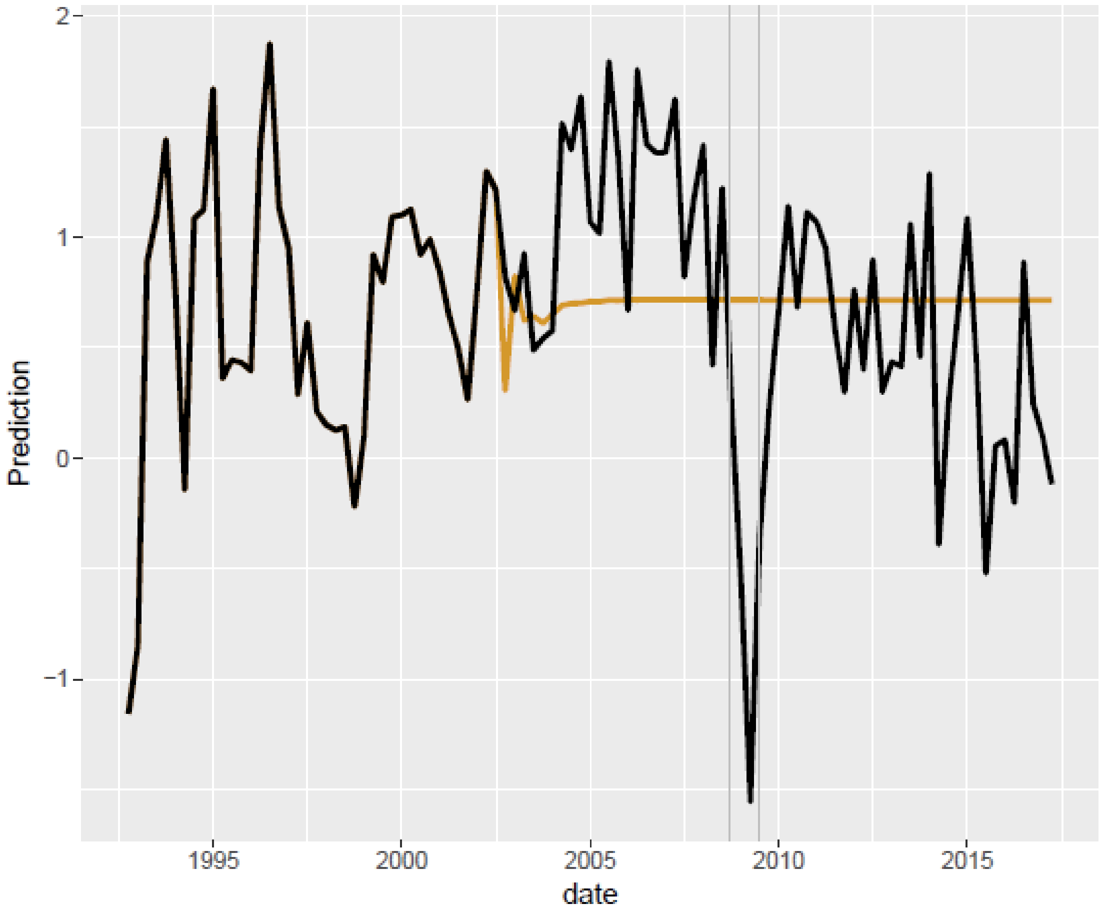
Source: Author’s own calculation, using data from Quantec (2018)

Figure 7: Autoregressive (2) Model Actual vs Predicted GDP



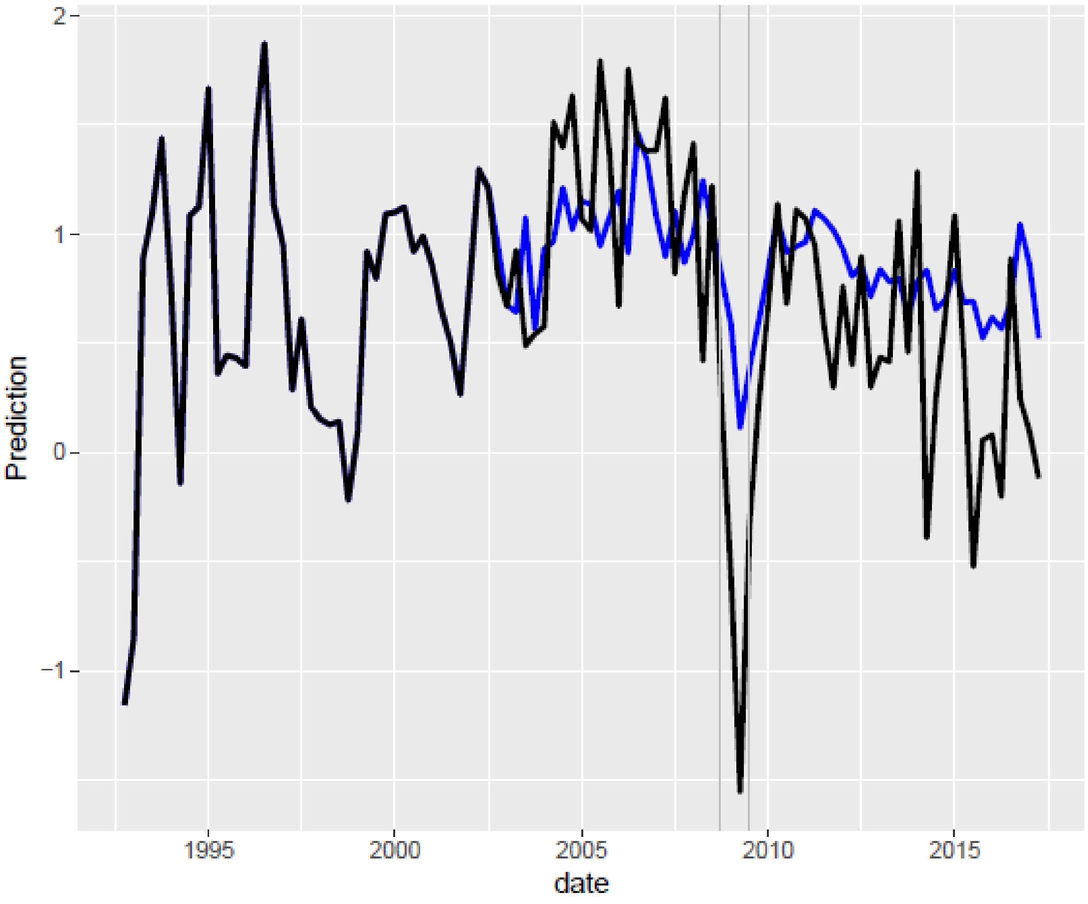
Source: Author’s own calculations, using multiple R packages

Figure 8: Vector Autoregressive (1) Model Actual vs Predicted GDP



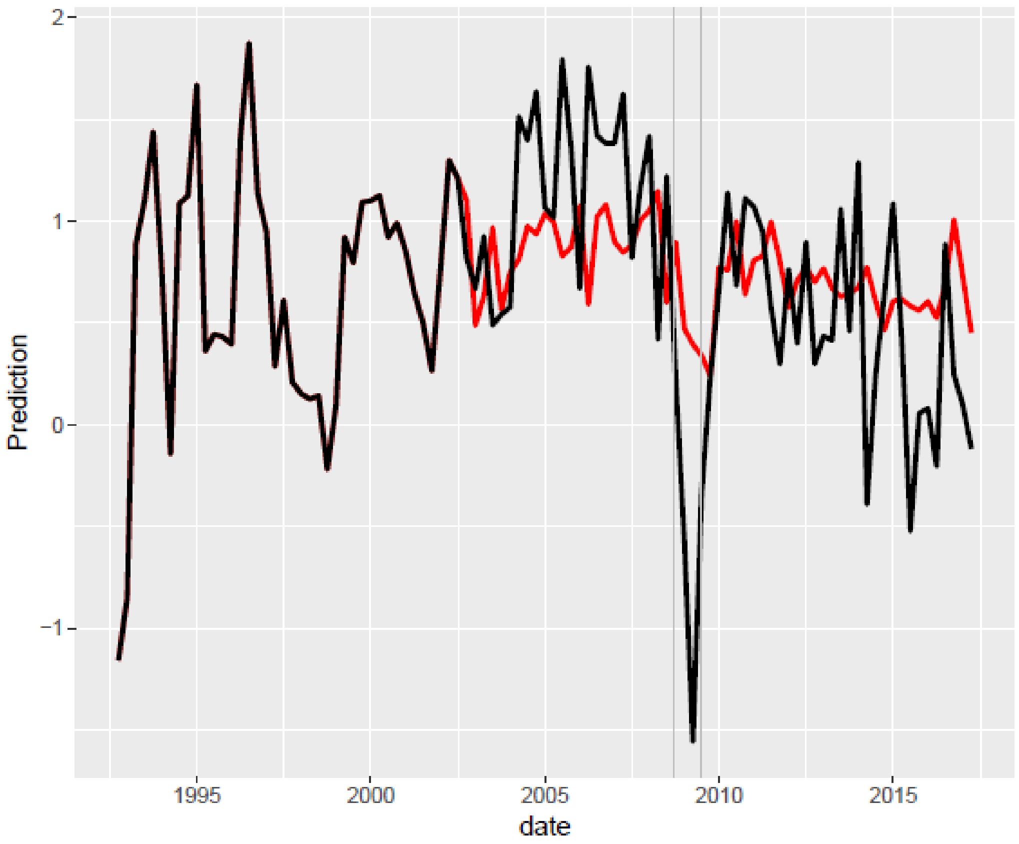
Source: Author’s own calculations, using multiple R packages

Figure 9: Elastic-net Model Actual vs Predicted GDP



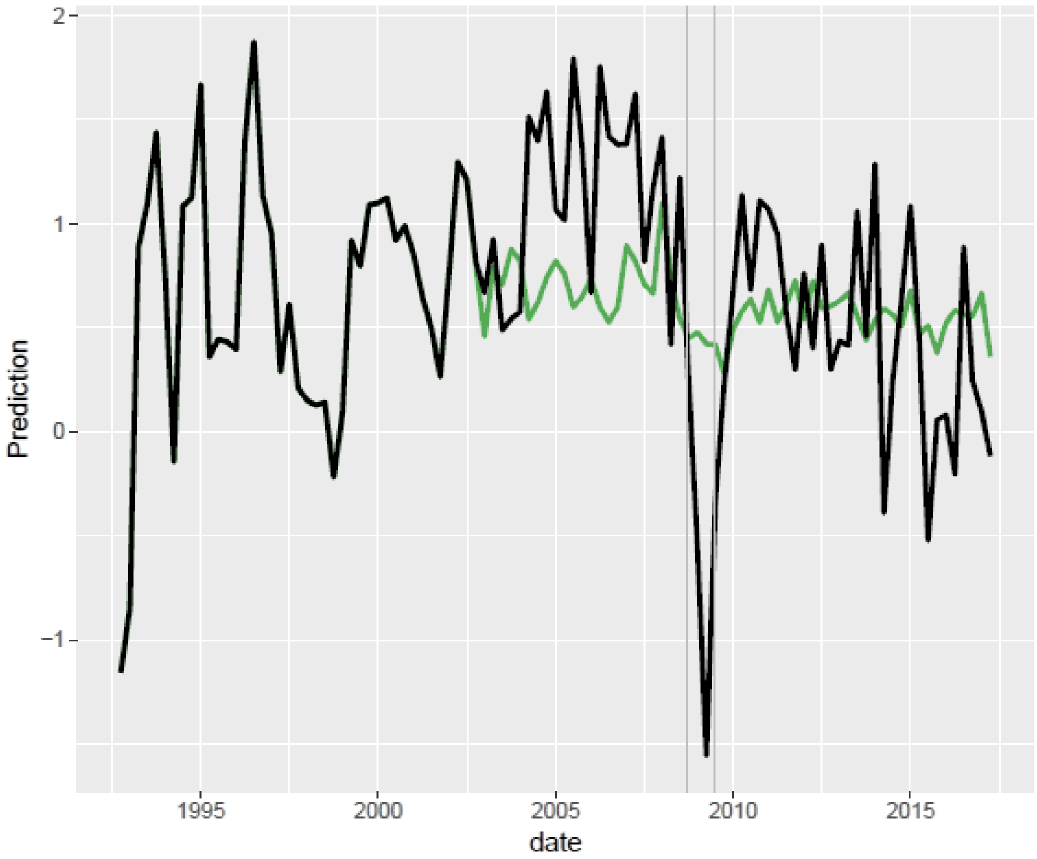
Source: Author’s own calculations, using multiple R packages

Figure 10: Random Forest Model Actual vs Predicted GDP



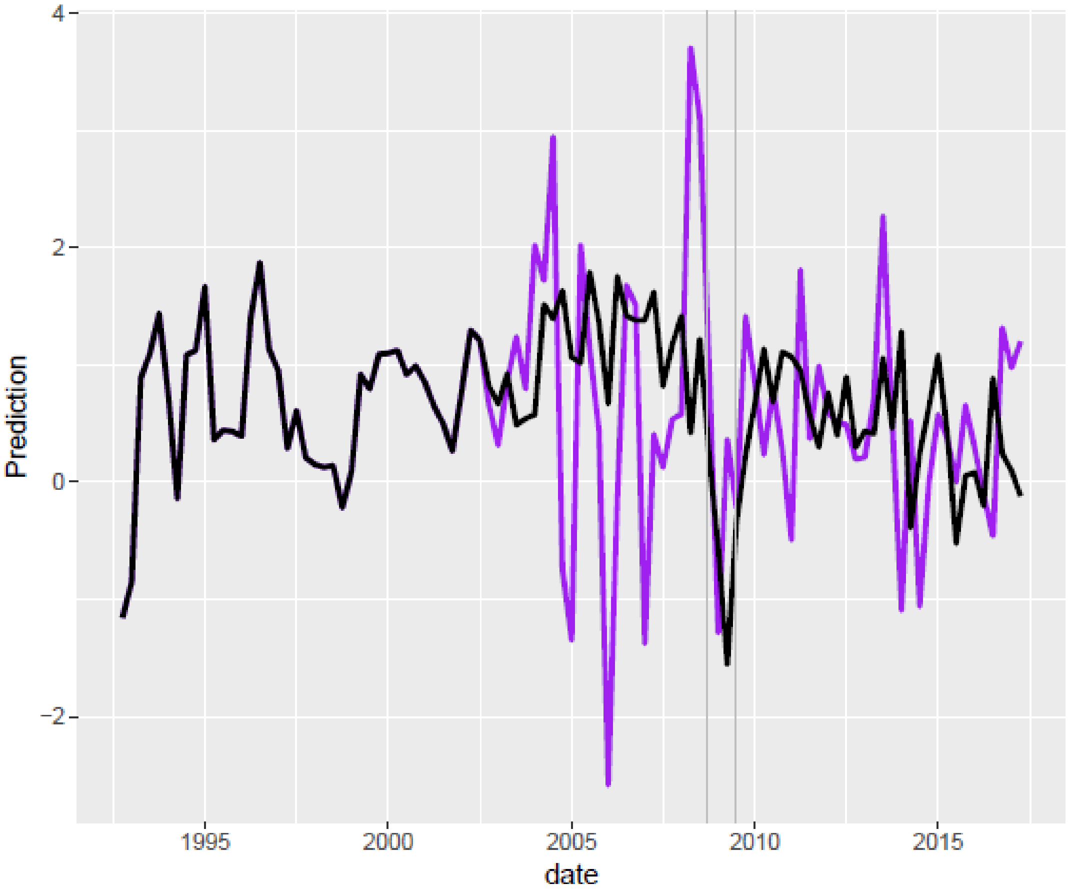
Source: Author’s own calculations, using multiple R packages

Figure 11: SVM Model Actual vs Predicted GDP



Source: Author’s own calculations, using multiple R packages

Figure 12: RNN Model Actual vs Predicted GDP



Source: Author’s own calculations, using multiple R packages