2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI)

Machine Learning For Classification Of Economic Recessions

Bruce Jackson
Graduate Programs in Software
University of St. Thomas
St. Paul, MN
jack3604@stthomas.edu

Manjeet Rege
Graduate Programs in Software
University of St. Thomas
St. Paul, MN
rege@stthomas.edu

Abstract

The ability to quickly and accurately classify economic activity into periods of recession and expansion is of great interest to economists and policy makers. Machine Learning methods can potentially be applied to the classification of business cycles. This paper describes two machine learning methods, K-Nearest Neighbor and Neural Networks, and compares them to a Dynamic Factor Markov Switching model for determining business cycle turning points. We conclude that machine learning techniques can offer more accurate classifiers that are worthy of additional study.

1. Introduction

The introduction of time-series economic data and business cycle classification models can be traced back to the National Bureau of Economic Research (NBER), which was founded in 1920 and published its first classification of business cycle dates in 1929. The Business Cycle Dating Committee (BCDC) of the NBER broadly defines a recession as the number of months between a peak and a trough in economic activity, and conversely, an expansion as the time period between a trough and a peak. A recession is not simply two consecutive quarters of decline in real Gross Domestic Product (GDP). "Rather, a recession is a significant decline in economic activity spread across the economy, lasting more than a few months, normally visible in real GDP, real income, employment, industrial production, and wholesale-retail sales."[1] As there is no fixed definition of economic activity, the BCDC is given wide discretion in its selection and interpretation of data in order to classify a period of time as belonging to a recession or expansion.

This paper describes one of the prevailing analytical methods used by economists for business cycle classification, the Dyanmic Factor Markov Switching (DFMS) model, and compares it to two machine learning methods, K-Nearest Neighbor (KNN) and Neural Networks (NN). Neural networks would appear to be well-suited to the task of business cycle classification because they can handle multivariate inputs with non-linear behavior, thus addressing the comovement and asymmetry seen with business cycle indicators. The remainder of the paper is organized as follows. Section 2 introduces the data that are typically used for quantitative business cycle classification, discusses some of the challenges associated with the use of that data, and how quantitative business cycle models evolved to address those challenges. Section 3 provides a brief overview of the extant analytical methods and introduces the KNN and NN models. Section 4 describes the methods used to compare and contrast the different models using a single data set and a receiver operator characteristic analysis, then proceeds to evaluate the three models. Section 5 concludes.

2. Data For Business Cycle Analysis

The development of data sources for business cycle analysis can be traced back to the 1930's and the work of NBER economists Arthur Burns and Wesley Mitchell, who developed 1,277 time series measurements of the economy and introduced the first statistical tools to study business cycles. Subsequent work by the NBER established the foundation for the creation of three weighted indices: The Composite Index of Leading Indicators, the Composite Index of Coincident Indicators and the Composite Index of Lagging Indicators. The coincident index was constructed to have cyclical turning points that mirrored those in the overall economy, making it ideal for business cycle analysis.

2.0.1 The Composite Index of Coincident Indicators

Among the data inputs used by the BCDC to inform their decision-making process, the four Coincident Economic Indicators (CEI) are considered to be the most important.[2]

[1] https://www.nber.org/cycles.html

[2] http://www.nber.org/cycles/r/recessions.html

978-1-7281-1337-1/19/$31.00 ©2019 IEEE
DOI 10.1109/IRI.2019.00019

The Composite Index of Coincident Indicators is derived from four time-series data sets: Nonfarm payroll employment, industrial production, real personal income excluding transfer receipts, and real manufacturing and trade sales. The focus on the coincident index and its four inputs was important for the development of quantitative business cycle models because it served as a convenient form of dimension reduction that was backed up by economic research. Not surprisingly, many of the quantitative business cycle turning point models we examined use some form of the Coincident Index components as their primary data input.

2.1 Analytical Challenges Presented By The Data

The collection of data and the selection of the coincident index components was an important step forward for business cycle analysis, but the use of this data set presented challenges for model development that included timeliness, asymmetry, co-movement and dimensionality.

2.1.1 Timeliness

An analysis of NBER business cycle announcements since 1980 shows that the official BCDC pronouncement date of a business cycle peak or trough can lag the actual turning point date by 5 to 22 months. The last pronouncement of the committee occurred in September 2010 when they identified June 2009 as the trough of the last recession, a 16-month lag. In their defense, the mission of the BCDC is to maintain a chronology of the business cycle, which places a higher value on accuracy rather than speed. Hence, the committee waits until it is confident it can assign an accurate peak or trough after conducting a thorough analysis of all the available data.

A major rate-limiting factor for improved speed is the time it takes to collect and assemble the source data used to construct the economic indicators. Government agencies try to balance the need for speed and accuracy by making a series of data releases for each economic indicator with an increasing level of accuracy for each subsequent release. The final data may be directionally different than the preliminary data, providing a ready example of the trade-off between speed and accuracy. The use of real-time data sets with the most recently available monthly data has been shown to produce more rapid determinations with acceptable accuracy [5], [7], [12].

2.1.2 Asymmetry

Zarnowitz [21] commented that the term ‘business cycle’ is a misnomer as the observed fluctuations vary in amplitude, scope, and duration, with no unique periodicity, which collectively leads to asymmetric behavior. Another conceptual framework for thinking about the asymmetry of business cycles refers to the periods of expansion and contraction as different phases or regimes, with the inflection points labeled as phase shifts or regime changes. The asymmetric behaviors seen during expansions and contractions are indicative of non-linear relationships in the data, which implies that non-linear analytical techniques are preferred when designing quantitative business cycle classification models.

2.1.3 Comovement

Burns & Mitchell [3] conceived of business cycles as roughly synchronous movements in many activities. The term comovement is often used to describe this tendency of individual measures of economic activity to move together. Comovement often occurs when shocks to the aggregate economy affect many sectors simultaneously [15], making the behavior of interest to the BCDC in their efforts to date the shift between phases. In terms of quantitative business cycle modeling, comovement is a multivariate phenomenon that implies a multivariate modeling response is required.

2.1.4 Dimensionality

As more variables are added to multivariate models to address comovement, another issue arises: Dimensionality. Specifically, as more variables are added to a multivariate model, the number of parameters to be estimated increases with the square of the number of variables, a phenomenon known as the curse of dimensionality. The challenge for quantitative business cycle classification model design is how to utilize hundreds of possible variables without the need for unreasonable computational overhead to calculate the model parameters. Early Dynamic Factor models addressed this issue by limiting their inputs to the four factors of the composite coincident index, a convenient form of dimension reduction supported by studies that verified the relevance of these factors [17], [18]. These real economic factors provide a limited view of the economy, however, by excluding financial factors. Recessions with financial market origins differ from those brought on by supply shocks or monetary policy shocks [16].

3. Quantitative Business Cycle Classification Models

Early modeling efforts were dominated by linear time-series models with origins in the 1930s, based on the linear difference model [20]. Broadly speaking, the first autoregression models assumed future values of GDP were a function of previous, lagged values of the data and that business cycles fluctuated predictably around a long-term trend line. As previously discussed, univariate linear models are not well-suited to deal with the comovement of multiple variables or the asymmetric behavior of business cycles, which led to the development of new modeling techniques that included the regime switching model and the dynamic factor model. These models were later combined into the dynamic factor Markov Switching (DFMS) model. Machine learing alternatives to the DFMS model include K-Nearest Neighbor (KNN) and Neural Networks.

3.1 Regime-Switching Model

The first regime switching business cycle classification model was detailed by Hamilton [11], who addressed the asymmetry issue by introducing a variable that modeled the state of the economy using a non-linear process. This state variable is similar to a dummy variable, but as the state of the economy is not directly observable, Hamilton devised a technique where the value of the state variable was the result of a probability estimation process governed by a first-order Markov chain. Notably, the Markov-switching process was non-linear and permitted the influence of factors outside of (i.e. exogenous to) the model, thus addressing the issue of asymmetry seen in economic time-series data. Being univariate, the model was still limited in its ability to address comovement.

The model employs an iterative process for estimation of the model parameters and the values of the state variable called the Hamilton Filter. This algorithm uses Maximum Likelihood Estimation (MLE) to construct a conditional density function followed by estimation of the conditional probability of the latent Markov state, which then permits evaluation of the conditional likelihood of the observable variables. The filter is also used to extract the values of the state variable, which takes the form of a posterior probability estimated from the transition probabilities that govern the Markov chain of the unobserved states. Alternatives to the Hamilton filter proposed in subsequent studies include Bayesian estimation [1], the Kalman filter [13] and the Bayesian Gibbs sampler [14], to name a few.

3.2 Dynamic Factor Model

As mentioned earlier, the Composite Indices derived from the NBER Business Cycle Indicators drew criticism because they were constructed subjectively without a mathematical approach or a clear connection to economic theory, which led to the development of composite indices for the coincident and leading indicators using a single index, dynamic factor model [17, 18]. This model assumes that the behavior of the time-series inputs is driven by two unobserved components, a common factor representing the overall state of the economy, which drives the comovement of all the variables, and an idiosyncratic factor, which accounts for measurement error and movements specific to an individual time series input. These latent variables are assumed to follow a stochastic autoregression process.

The unobserved factors for the model are estimated by combining the Kalman filter algorithm with Maximum Likelihood Estimation (MLE). After constructing an index from the values of the unobserved common factor, the model was found to closely match the previous version of the CEI while providing a formal mathematical structure for its construction. The direct estimation of the vector of unobserved factors also meant that future values could be forecasted or incorporated into other econometric forecasting models.

3.3 Dynamic Factor Markov Switching Model

While the dynamic factor model addressed the issue of co-movement, the latent factors were estimated with linear autoregression techniques, which made it difficult to address the asymmetric behavior of business cycles. By combining the dynamic factor model with a regime switching model, the Dynamic Factor Markov-Switching (DFMS) model was able to address comovement with the multivariate dynamic factor model and the asymmetry of regime shifts with a discrete-state Markov process. Diebold & Rudebusch [9] were the first to suggest combining the two models in a single index format, but Chauvet [5] was the first to build and estimate a dynamic factor regime- switching model using the components of the coincident index.

A Bayesian Gibbs sampling approach as described in [14] was used by Chauvet to estimate the model parameters and state variable probabilities. Chauvet & Piger [6], [7] found that combining a DFMS model with real-time data was able to significantly improve the speed with which business cycle determinations were made compared to the BCDC, thus addressing the timeliness issue.

3.4 K-Nearest Neighbor

Where DFMS methods use probability density functions to classify data as belonging to a particular regime, neighbors-based classification methods use distance functions. Training data are set in a feature space and a machine learning algorithm calculates the distance of a new data point to the training data points according to a prespecified distance function, Euclidean distance being the standard method. Classification is accomplished by assigning the data point to a class by majority vote of the nearest data points. The number of data points forming the group of nearest neighbors who participate in the vote is pre-specified as an integer, K, hence the name K-Nearest Neighbor. KNN can be computationally expensive on larger datasets because all distances between all points are calculated for comparison.

3.5 The Neural Network Model

The basic building block of neural network computing is the perceptron, which can be thought of as an artificial neuron that receives, processes and transmits information. In the human neuron, the processing and modulation of electrical and chemical signals is performed by synapses. The perceptron algorithmically mimics the synapse by multiplying each input value by a value called the weight. The signals received by a human neuron accumulate until they exceed a particular threshold. Once the threshold is exceeded, the neuron is triggered to send an impulse, which is referred to as the action potential. The perceptron models this process by taking a weighted sum of the inputs and applying a series of functions to generate the output. An activation function (e.g. rectified linear unit) is used to train the neural net using a loss function (e.g. mean square error) that is optimized with an algorithm (e.g. stochastic gradient descent) to adjust the input weights. The aforementioned functions are specified prior to training the model and are collectively referred to as the hyperparameters. The process of adjusting the input weights involves a series of passes forward and backward through the neural net, with each round trip of the full data set referred to as an epoch. The combination of the node, the weights, the hyperparameters and a constant called the bias, forms a basic, single-neuron perceptron, which is essentially a linear classifier similar to a support vector machine that can separate the input data into two categories. This single-neuron perceptron model can be extended by adding more neurons, creating a single layer perceptron, and by adding more layers to create a multi-layer perceptron (MLP).

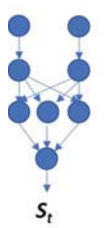

Figure 1. Neural Network Architecture.

The arrangement of the perceptrons and the layers is referred to as the architecture of the neural network, which maps the flow of data through the input layer, the hidden layer (or layers), and the output layer. By using multiple layers of non-linear processing units, neural network models can extract multiple levels of features or representations of the data, as well as combinations of features. Given these capabilities, neural networks would seem to be well suited for the task of quantitative business cycle classification because they are multivariate and non-linear, and can thus address the issues of asymmetry and comovement.

4. Model Comparison

The Dynamic Factor Markov Switching model is established in the literature for quantitative business cycle classification, making it well-suited to be the comparator model for our machine learning methods. To perform the comparison, we run each model using a uniform data set and then compare the the results to the official NBER business cycle designations. Model performance is evaluated using a Receiver Operator Characteristic (ROC) analysis, which is a standard method to evaluate the quality of a binomial classifier.

4.1 Model Comparison Dataset

As discussed earlier, the coincident index was important for the development of quantitative business cycle classification models because it served as a convenient form of dimension reduction validated by a body of econometric research. In order to maintain a link to past research while simplifying the current task, our comparison dataset builds upon an existing real-time dataset composed of the four components of the Composite Index of Coincident Indicators. In addition to updating the dataset for the most recent vintages, we provide details on the source data, discuss the definition of a data vintage, and address the issue of asynchronous datasets. We also construct a dataset containing the NBER business cycle designations, with a discussion of coding alternatives for the recessions and how to address the extended periods of time between BCDC pronouncements during expansions.

4.1.1 Data Sourcing

The data used to compare the models begins with the the real-time dataset from Camacho et al [4], which is composed of the components of the Composite Index of Coincident Indicators: Nonfarm payroll employment, industrial production, real personal income excluding transfer receipts, and real manufacturing and trade sales. We take the original data spanning January 1967 through May 2017 and add the most recent values from an archival versionof the St. Louis Federal Reserve Economic Research Divisions Federal Reserve Economic Data (FRED) database known as ALFRED (ArchivaL FRED), which contains archived real-time data vintages from the FRED database. These new data bring the dataset up to August 2018, with 619 months of data captured in 501 vintages.

4.1.2 Defining A Vintage

Our models make use of real-time datasets, which organize the data for each economic indicator into vintages that reflect the unrevised data that were available at a given point in time. With each set of monthly data releases, a new data vintage is created to reflect the best available information for a particular time period. The release of new data for the individual indicators is not synchronized, however, which means a cut-off point must be specified for when a particular data release is included in a particular vintage. Note that the vintage refers to the time period covered by the data, time t, not when it is assembled at time t+1. Previous work assumes the data for a vintage is assembled around the middle of the month, which results in a vintage that includes non-farm payroll employment and industrial production for time t, real personal income excluding transfer receipts at time t-1, and real manufacturing and trade sales for time t-2. The model comparison data set uses that same convention with a cutoff point on the 15^{th} of the month. This data is thought to provide a more realistic assessment of model performance under real-world conditions and has been shown to produce more rapid determinations with acceptable accuracy.

4.1.3 Missing Data

Because of the asynchronous data production schedule, the last entry for personal income is blank and the last two entries for real manufacturing and trade sales are blank when using real-time datasets, a condition referred to as ragged edges or ragged ends. Most analytical methods cannot process datasets with ragged edges, so a strategy for dealing with the missing values is required. One option is to trim back the dataset two periods using only the time periods where there is data for all four indicators, but at the risk of excluding newer data that could be informative. Several alternatives exist to estimate the missing values prior to analysis. For example, the DFMS model of Camacho et al [4] randomly assigns new values to the blank data items prior to analysis. Our machine learning model opts for an autoregressive method that takes the last known value for a data series and feeds it forward to complete the dataset.

4.1.4 Coding NBER Peak/Trough Data

For machine learning models like KNN that train the classifier using a true classification label, a dataset with the ground truth values is required. We construct an NBER dataset for that purpose, which codes business cycle data as a dummy variable with periods of expansion = 0 and recessions = 1. An issue arises when coding the peak month of an expansion and the trough month of a recession, since the BCDC only identifies the month and year of the turning point without specifying whether the turning point was closer to the start or the end of the month. There are a few possible strategies for coding the data. The trough method assigns the start of the recession to the period following the declared peak month and assumes the recession runs through the end of the trough month. The peak method includes the month declared as the start of the recession and excludes the trough month. Both of these methods produce equal recession durations. It is also possible to include both the peak and the trough month or exclude the peak and trough months. The default method for FRED is the trough method, which we adopt for our dataset as well.

4.1.5 Lag Time Between NBER Pronouncements

When training a classifer, it is desirable to use only the values where a definitive designation of the business cycle state is available. A challenge in making this designation is that the BCDC determinations are intermittent, which results in extended periods of time with no official designation of a recession or expansion. We employ a series of decision rules to re-classify historic data as being in-sample when it becomes likely that there has been no change in the state of the economy since the last pronouncement. Following a peak, we assume that a recession runs at least 6 months and allow the following months to run as out-of-sample until an official trough designation is made. Following a trough announcement and the start of a new expansion, we assume a new expansion runs at least 6 months, then after the expansion passes 12 months we allow the window to roll forward and change the designation of the year-ago period from unknown to expansion.

4.2 Model Comparison Metrics

This paper evaluates the quality of a business cycle classification with the Receiver Operator Characteristic (ROC), an established quality metric for binomial classifiers. The analysis begins by comparing the model output to the ground truth, in our case the official NBER business cycle classifications.

The ROC analysis shows the ability of the classifier to correctly identify recessions as the True Positive Rate (TPR), compared to the cost of an incorrect classification, shown by the False Positive Rate (FPR). The ROC analysis can be shown graphically by computing distributions of the TPR and FPR for the complete range of decision thresholds, then plotting the TPR on the y-axis and the FPR on the x-axis, with each point representing the overlap between the two distributions at a particular decision threshold.

A classifier with perfect discrimination would show no overlap between the two distributions, resulting in curve that passed through the upper left-hand corner with 100 percent true positives and no false positives. Conversely, classifiers with performance no better than random chance would

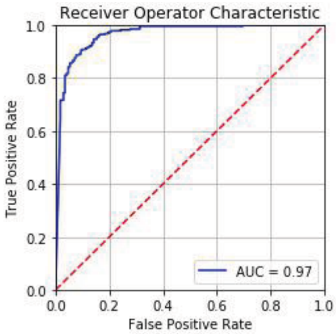

Figure 2. ROC Curve Example

fall along a straight diagonal line from the lower left-hand corner to the upper right-hand corner. The ROC curve can be used to assess the overall accuracy of a binary classifier by looking at the Area Under the Curve (AUC), which can be captured as a single number known as the AUC Score. The scores range from 0.5 (no better than random chance, which represents the area under the diagonal line) to 1.0 (perfect discrimination).

4.3 DFMS Model Implementation

The DFMS model created by Camacho et al [4] serves as the baseline quantitative business cycle classification model. Unlike early DFMS models, the approach of Camacho et al permits the use of unbalanced data sets and variables with different measurement frequencies. This flexibility enhances the ability of the DFMS model to utilize real-time data sets where the availability of the most recent data can vary depending on the production schedule for a particular economic indicator, resulting in ragged end datasets. The program computes the sequential growth rates and standardizes the inputs prior to estimation of the parameters. Model estimation is performed recursively using the latest vintage for the time period under consideration. The computer code using the GAUSS Matrix Programming Language and the original data are available on Camacho's web site.[3]

4.3.1 DFMS Results

Charts summarizing the classification results are shown in Figure 3. A comparison of the recession probabilities generated by the DFMS model and the NBER defined recessions show congruent results. With the NBER business cycle classifications serving as ground truth, we calculate a confusion matrix with the true positives, true negatives, false positives and false negatives. After calculating the TPR and

[3]http://www.um.es/econometria/Maximo/publications.html

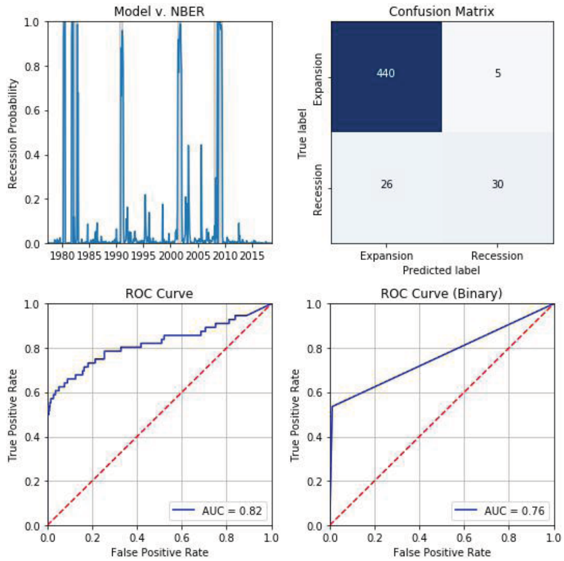

Figure 3. DFMS results. The baseline model generated a 0.82 AUC score. The AUC score for binary version of the classifier was 0.76 using a 50% cutoff.

the FPR using the last vintage, the ROC curve is then plotted. The Area Under the Curve for the DFMS model is 0.82 using the probabilities output by the program. We also create a binary classifier by setting a threshold value of greater than or equal to 50 percent for the state variable to signify a recession and less than 50 percent to represent an expansionary business cycle and show the ROC curve. The AUC of 0.76 generated by the binary classifer is slightly lower than original ROC curve analysis.

4.4 KNN Model Implementation

Our K-Nearest Neighbor model is written in Python 3.6.5 using the sci-kit learn classification package. Our model allows the KNN algorithm to define the feature space decision boundaries directly from a train-test split of the data before cross-validating the results. Unlike DFMS, KNN does not naturally lend itself to autoregressive parameters, which are desirable when dealing with time series data that exhibit trend and cycle characteristics. To address this issue, a second data set is lagged one month and combined with the original data into a single data set (eight variables in total), a methodology suggested by [10]. Training is done using the in-sample NBER business cycle data as the classification variable. For each data vintage, the model performs 100 iterations of KNN classification, a procedure is referred to as bootstrap aggregation (a/k/a bagging). Bagging is a meta-algorithm that generates multiple versions of a predictor or classifier and then takes a majority vote to arrive at the final result [2]. Bagging has been found to improve the performance for unstable classifiers which vary significantly with small changes in the data set, especially classification and regression trees (CART). The results of the iteration are averaged to produce the final probability of recession. The classifier can be used to predict the labels of the out-of-sample data set.

4.4.1 KNN Results

The summary charts for the KNN model are shown in Figure 4. The results show good agreement with the official NBER business cycle designations. To measure the quality of the model we capture the AUC score for all vintages of the model over all of the iterations, then average the results, resulting in a 0.78 Area Under the Curve for the entire data set. We also ran a train/test split of the last vintage in order to generate a cross-validated ROC curve. The Area Under the Curve for the final vintage of the KNN model is 0.94 using the binary classifier. KNN does not generate probabilities.

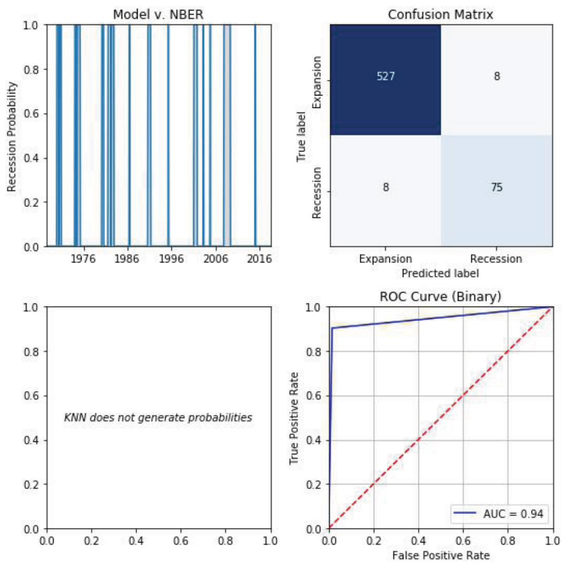

Figure 4. KNN Results. This machine learning classifier generated a 0.94 AUC score.

4.5 Neural Network Model Implementation

Our binary classification neural network model is written in Python 3.6.5 using the Keras and TensorFlow packages, structured as a multi-layer neural network with an input layer, a hidden layer and an output layer. We use the same dataset treatment as the KNN model, which produces eight input vectors (4 indicators at time t, 4 lagged indicators at t-1). All eight of the input vectors are normalized and fed to the input layer of the model. The hidden layer is fully-connected, (a/k/a dense) meaning that all input vectors are fed to all nodes. The output layer takes the hidden layer and reduces it to a single value between zero and one, which represents our state variable. Dropout layers inserted between the input, hidden and output layers randomly set a fraction rate of input units to 0 at each update during training to help prevent overfitting. The data are split into a training set (consisting of randomly selected observations representing 33 percent of the total dataset) and a test set, which is used to validate the model after it is fitted with the training data. This model deals with missing values using the feed-forward approach, where the last known value of an indicator is carried forward to fill out the dataset.

Certain model hyperparameters must be specified in advance, including the activation function, the optimizer, and the loss function. A sigmoid activation function (similar to logistic regression) was selected, which is generally used for binary classification problems. The model uses an RMSProp optimizer, an unpublished, adaptive learning rate method proposed by [19], which divides the learning rate for a weight by a running average of the root mean squared (RMS) gradients for that weight. An adaptive learning rate optimizer was selected because it performs well with sparse data and with deep neural networks. Since our classification model outputs a probability value between 0 and 1, we selected a cross-entropy loss function (a/k/a log loss) to measure performance. The log loss function is commonly used to fit logistic regression models and is a natural choice for our binary classification model.

4.5.1 Neural Network Results

The summary charts for our neural network model are shown in Figure 5. To measure the quality of the overall model we captured the AUC score for all vintages of the model over all of the iterations and averaged the results, which yielded a 0.96 Area Under the Curve. We also ran the model using the t-1 vintage as the training set and the t vintage as the validation set so that we could generate a ROC curve. The Area Under the Curve for the final vintage of the NN model is 0.97 using the raw output, and 0.81 for a binary classifier. While the overall fit is quite good, the increased sensitivity of the neural network classification appears noisy and resulted in a lower AUC when evaluated as a binary classifier. A slightly lower cutoff point for recessions could potentially increase the overall performance.

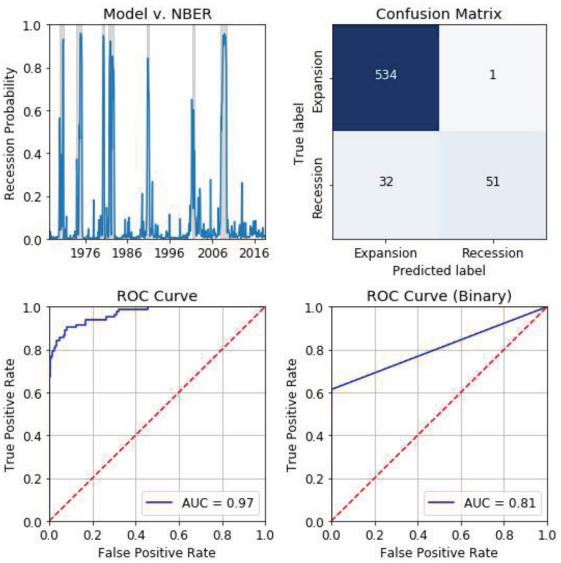

Figure 5. Neural Network Results. The AUC score of 0.97 is the highest of the three models. A cutoff below 50% could potentially improve the binary AUC score.

5. Conclusions

This paper set out to describe two machine learning methods, K-Nearest Neighbor and Neural Networks, and compare them to an extant Dynamic Factor Markov Switching model for determining business cycle turning points. We found both machine learning models produced higher AUC scores than the DFMS model, leading us to conclude that machine learning techniques are worthy of additional study for addition to the econometricians armamentarium.

Asymetry and comovement could be further addressed with the addition of more perceptrons and layers to our neural network model, which further refine the quantitative business cycle classification task [8]. Perceptron features can be further enhanced to provide various latency, lag and memory characteristics that could further improve accuracy, and provide ample opportunities for future research.

References

[1] J. H. Albert and S. Chib. Bayes Inference via Gibbs Sampling of Autoregressive Time Series Subject to Markov Mean and Variance Shifts. *Journal of Business and Economic Statistics*, 11(1):1–15, 1993.

[2] L. Breiman. Bagging Predictors. *Machine Learning*, 24(2):123–140, 1996.

[3] A. Burns and W. Mitchell. *Measuring Business Cycles*. National Bureau of Economic Research, Boston, 1946.

[4] M. Camacho, G. Perez-Quieros, and P. Poncela. Markov-switching dynamic factor models in real time. *International Journal of Forecasting*, 34(4):598–611, 2018.

[5] M. Chauvet. An Econometric Characterization of Business Cycle Dynamics with Factor Structure and Regime Switching. *International Economic Review*, 39(4):969–996, 1998.

[6] M. Chauvet and J. Piger. A Comparison of the Real Time Performance of Business Cycle Dating Methods. *Federal Reserve Bank of St. Louis Review*, 85(2):47–61, 2003.

[7] M. Chauvet and J. Piger. A Comparison of the Real Time Performance of Business Cycle Dating Methods. *Journal of Business and Economic Statistics*, 26(1):42–49, 2008.

[8] T. R. Cook and A. S. Hall. Macroeconomic indicator forecasting with deep neural networks. *Federal Reserve Bank of Kansas City, Research Working Paper 17-11, September*, 2017.

[9] X. F. Diebold and G. Rudebusch. Measuring business cycles: A modern perspective. *Review of Economics and Statistics*, 78:67–77, 1996.

[10] A. Giusto and J. Piger. Identifying Business Cycle Turning Points in Real Time with Vector Quantization. *International Journal of Forecasting*, 33(1):174–184, 2017.

[11] J. Hamilton. A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle. *Econometrica*, 57(2):357–384, 1989.

[12] J. Hamilton. Calling Recessions in Real Time. *Econometrica*, 27(4):1006–1026, 2011.

[13] C. J. Kim. Dynamic linear models with markov-switching. *Journal of Econometrrics*, 6-(1-2):1–22, 1994.

[14] C. J. Kim and C. R. Nelson. Business Cycle Turning Points, a New Coincident Index, and Tests of Duration Dependence Based on a Dynamic Factor Model with Regime Switching. *The Review of Economics and Statistics*, 80(2):188–201, 1998.

[15] R. Lucas. Econometric Policy Evaluation: A Critique. *Carnegie-Rochester Conference Series on Public Policy*, 1(1):19–46, 1976.

[16] S. Ng and J. H. Wright. Facts and challenges from the great recession for forecasting and macroeconomic modeling. *Journal of Economic Literature*, 51(4):1120–1154, 2013.

[17] J. W. Stock and M. W. Watson. New Indexes of Coincident and Leading Economic Indicators. *NBER Macroeconomics Annual*, 4:351–409, 1989.

[18] J. W. Stock and M. W. Watson. *The Leading Economic Indicators: New Approaches and Forecasting Records*, chapter A Probability Model of the Coincident Economic Indicators, pages 63–90. Cambridge University Press, 1991.

[19] T. Thielman and G. Hinton. Rmsprop: Divide the gradient by a running average of its recent magnitude. coursera: Neural networks for machine learning, 2014.

[20] J. Tinbergen. Sur la dtermination statistique de la position d'quilibre cyclique. *Review of the International Statistical Institute*, 4(2):173–188, 1936.

[21] V. Zarnowitz. Recent Work on Business Cycles in Historical Perspective: A Review of Theories and Evidence. *Journal of Economic Literature*, 23(6):523–580, 1985.