Transfer Learning for Business Cycle Identification

Marcelle Chauvet* Rafael Guimaraes†
chauvet@ucr.edu rrsguim@gmail.com

This Version: July 28, 2020

PRELIMINARY AND INCOMPLETE - CHECK WITH AUTHORS FOR LATEST DRAFT.

Abstract

A deep learning model is proposed as a strategy for identifying business cycles. The approach integrates the idea of storing knowledge gained from one region's economics experts and applying it to other geographic areas. The first is captured with a supervised deep neural network model and the second by applying it to another dataset, a transfer learning procedure. The hyperparameters were tuned using a grid search process. The empirical results indicate that the strategy proposed leads to successful business cycle identification.

*University of California, Riverside (UCR).
†Central Bank of Brazil and Federal University of Rio Grande do Sul (UFRGS). The views expressed in this work are those of the authors and do not necessarily reflect those of Central Bank of Brazil, UFRGS and UCR or its members.

Contents

1 Introduction

Monitoring business cycle phases is a traditional task in applied macroeconomics. Progressive market integration has induced a worldwide interest in the analysis of cyclical fluctuations through the use of economic indicators (Chauvet, 2001, p.21). Changes in exchange rates, outputs, consumption, inflation, and interest rates in different parts of the world can influence the effectiveness of government policies and the competitive position of businesses, even those not directly related to international operations (Chauvet and Yu, 2006, p.43). As a result, a wide range of techniques has been developed since the seminal work by Burns and Mitchell (1946). Recently, new approaches have emerged due to the progress in machine learning (ML) research, centering on building models that achieve better forecasting performance than the non-ML models or that identify turning points more timely (Piger, 2020).

In this work, we contribute to the literature by exploring the transfer learning (Pratt et al., 1991) capability of artificial neural networks, a characteristic that, to the best of our knowledge, has not yet been evaluated to monitor business cycle phases. Our goal follows Chauvet and Yu (2006) in provide additional tools for governments and the private sector to conduct their activities in light of both national and international economic conditions. For that, we adopt a combined strategy of deep neural network and transfer learning to address the practical research problem of identifying the business cycle phases. As in the dynamic factor model with Markov-switching introduced by Chauvet (1998), the deep neural network approach accounts for the idea of business cycles as the simultaneous, asymmetrical, and nonlinear movement of economic activity in various sectors. Additionally, transfer learning improves learning in a new task through the transfer of knowledge from a related task that has already been learned (Torrey and Shavlik, 2009). Therefore this additional strategy over the standard approaches has transferability accounting for the main advantage, which is relevant in situations where the data is often limited, like identifying business cycles worldwide.

The rest of the paper is organized as follows. Section 2 presents a literature review and in Section 3 we discuss the methodology. Section 4 presents our empirical findings, and, in Section 5, the final remarks.

2 Literature

Business cycles are recurrent sequences of alternating phases of expansion and contraction among many economic activities (Burns and Mitchell, 1946). According to Harding and Pagan (2005), there are three ways in the literature to describe what we mean by a cycle, depending on whether the main focus is on the fluctuation of the level of economic activity, the level of economic activity less a permanent component, or the growth rate of economic activity.

In the United States, the National Bureau of Economic Research (NBER)[1] Business Cycle Dating Committee provides a chronology of business cycle expansion and recession dates. According to Piger (2020), because the NBER methodology is not explicitly formalized, literature has worked to develop and evaluate formal statistical methods for establishing the historical dates of economic recessions and expansions in both the U.S. and international data. Estrella and Mishkin (1998), Estrella et al. (2000), Kauppi and Saikkonen (2008), Rudebusch and Williams (2009) and Sebastian (2016) use an available historical indicator of the class, as the NBER dates, to estimate the parameters of the models such as logit or probit. This strategy is

[1]The NBER does not define a recession in terms of two consecutive quarters of decline in real 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. Source: https://www.nber.org/cycles.html.

called a supervised classifier in the statistical classification literature, in contrast to unsupervised classifiers, which endogenously determine the classes. Unsupervised classifiers have also been used, with the primary example being the Markov-switching (MS) framework of Hamilton (1989), which become a relevant tool for applied work in economics. Chauvet (1998) proposes a dynamic factor model with Markov-switching (DFMS) to identify expansion and recession phases from a group of coincident indicators and Chauvet and Hamilton (2005), Chauvet and Piger (2008) and Camacho et al. (2018) evaluate the performance of variants of this DFMS model to identify NBER turning points in real-time. See Piger (2020) for a comprehensive review.

Recently, artificial intelligence (AI) has gained considerable prominence due to performances in autonomous vehicles, intelligent robots, image and speech recognition, automatic translations, medical and law usage (Makridakis, 2017). In Economics, the application of machine learning (ML) methods, an AI technique, is not new, and in a way, it has followed the phases of use in other areas. This has extended from the earliest attempts in the 1940s, followed by the rising expectations and the results in the 1960s, through the period of frustration in the 1970s, to the continuity of its use by a small group of researchers in the 1980s, and the resurgence in the 1990s (Stergiou and Siganos, 2011). Finally, from the beginning of the 21st century, significant progress has been observed in many areas, attracting attention and funding for research.

Applied ML papers related to business cycles can be separated depending on whether the main focus is *predicting* or *identifying* turning points and phases. For example, Hoptroff et al. (1991), Qi (2001), Klinkenberg (2003), Nasr et al. (2007), Berge (2013), Ma (2015), Garbellano (2016), Nyman and Ormerod (2017), and James et al. (2019) have applied machine learning techniques such as artificial neural networks, support vector machines, boosting, k-nearest neighbor, and random forest to *forecasting* turning points, recessions, or business cycles phases mainly in the US, but also other countries[2]. These studies have generally reported some improvements over non-ML strategies. The other set of papers is concerned about *identifying* the turning points for real-time classification. Morik and Rüping (2002), Giusto and Piger (2017), Soybilgen (2018), Raffinot and Benoit (2019) and Jackson and Rege (2019) have applied inductive logic programming, learning vector quantization, random forest, boosting, k-nearest neighbor and artificial neural networks fed with dynamic factors. Piger (2020), in a comprehensive analysis, compares five ML techniques with DFMS. These studies have reported quickly and accurately turning points identification.

Lastly, some literature is dedicated to the study of business cycles worldwide, as in Chauvet and Yu (2006), Cuba-Borda et al. (2018), Abberger et al. (2020), and the reference turning points of the OECD Composite Leading Indicators[3]. As mentioned in the introduction, in this paper, we explore a transfer learning approach to this subject.

3 Methodology

Due to the availability of various ML methods and that we are, especially in economics, in the explanatory era of its applications, works often apply several ML approaches to a specific dataset to compare their performances, as in Tiffin (2016), Cook and Hall (2017), Garcia et al. (2017) and Gu et al. (2018). Makridakis et al. (2018) go further to compare various non-ML and ML forecasting methods. Another possible approach is to select in advance one that is suitable for a specific task, which, in our case, is to evaluate the transfer learning capability of a supervised deep neural network to address the practical research problem of identifying the

[2]United Kingdom, Japan, West German and Lebanon.

[3]Available at https://www.oecd.org/sdd/leading-indicators.

business cycle phases.

A deep neural network, also known as deep learning (DL), is an artificial neural network with multiple layers hidden between the input and output layers (Bengio, 2009). In essence, almost all DL algorithms can be described as a combined specification of a data set, a cost function, an optimization procedure, and a model (Goodfellow et al., 2016). This data-driven framework is flexible enough to be training with different features, the independent variables, and targets, the dependent variables. See Appendix A.1 for details. A supervised DL model learns from the features-targets without the need for assumptions about its relation, which prevents from chose, in our case, an underlying economics school of thought to set up a model, although the variable selection might represent it. The algorithm maps the input-output relation variables according to the training and validation data. For instance, once we choose NBER[4] turning points classification data as output label, the DL algorithm will learn from them how to classify the business cycle, implicit following the same school of thought.

Transfer learning (TL) refers to the situation where what has been learned in one setting (e.g., distribution P1) is exploited to improve generalization in another setting (say, distribution P2). The learner must perform two or more different tasks, but it is assumed that many of the factors that explain the variations in P1 are relevant to the variations that need to be captured for learning P2 (Goodfellow et al., 2016, 534). In the conventional transfer learning approach, we first train a base network on a base dataset and task, and then we repurpose the learned features or transfer them to a second target network to be trained on a target dataset and task. This process will tend to work if the features are general, meaning suitable to both base and target tasks, instead of specific to the base task (Yosinski et al., 2014). Figure 1 illustrates these dynamics. The base networks (top two rows) are trained using standard deep learning procedures on datasets A and B. The labeled rectangles (e.g. WA1) represent the weight vector learned for that layer, with the color indicating which dataset the layer was originally trained on. The vertical, ellipsoidal bars between weight vectors represent the activations of the network at each layer. The target networks (bottom two rows) represents transfer learning strategies. The first n weight layers of the network (in this example, n = 3) are copied from a network trained on one dataset (e.g. A), and then the entire network is trained on the other dataset (e.g. B). Usually, the first n layers are either locked during training or allowed to learn. In this work, we chose to keep all layers locked.

3.1 Data

A deep learning model is capable of handling a large number of explanatory variables (features). However, there is a relevant trade-off for our strategy: the more features we use during learning, the less likely we will be able to apply the transfer learning. For example: if we train for the United States business cycles classification using dozens of features, we will need the same quantity of features, or additional treatment for each missing one, to carry out transfer learning to other places. So, in our case, and conditionally to the results quality, the fewer features, the better. Also, we chose to use quarterly data because this is the frequency at which the most relevant variables for the classification of the business cycle are available worldwide.

Thus, our feature selection comprises the coincident variables indicated by NBER as the fundamental, i.e., measures of gross domestic product (GDP), income, employment, industrial production, and wholesale-retail sales. The target values are the business cycles chronology provided by the datings committee. To have a common starting point for each dataset, we restrict the series beginning to eliminate missing values. We adopted a U.S. dataset for deep learning and two datasets, with data from Brazil and Europe, for transfer learning. We computed the first difference of the logarithm of the input features, capturing the growth rate (Harding and

[4]The most followed classification for U.S. business cycle.

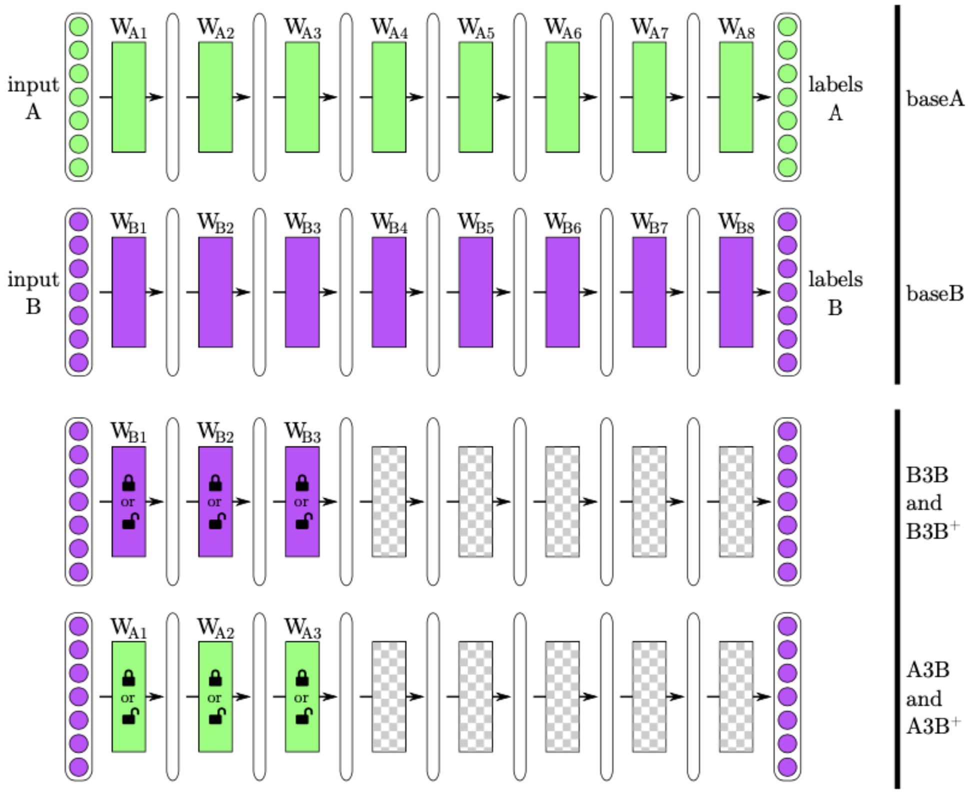

Figure 1: Transfer learning overview - reproduced from Yosinski et al. (2014)

Pagan, 2005, 152-154). The option for a feedforward network, which represents memory-less models, derives from the focus on contemporary movement between the selected variables and the business cycle. This implies disregarding the time dependence observed on the variables and shuffling the data to break it. The resulting model accounts just for coincident relations. Introducing time dependency is left for future work, as mentioned in Section 5. The Appendix A.2 summarizes the information about all series, mostly from the FRED-MD[5] dataset, provided by the Federal Reserve Bank of St. Louis.

3.2 Implementation Details

The models were implemented using TensorFlow[6], an interface for expressing machine learning algorithms, and an implementation for executing such algorithms. TensorFlow is flexible and can be used to express a wide variety of algorithms, including training and inference algorithms for deep neural network models. It has been used to conduct research and deploy machine learning systems into production across more than a dozen areas of computer science and other fields (Abadi et al., 2015).

Beginning with the deep learning step, we split the U.S. dataset into train, validation, and test sets. Then, we define a function that creates a neural network with a densely connected hidden layer, ReLU as activation function, a dropout layer to reduce overfitting, and a sigmoid output layer that returns the probability of recession. The optimal set of hyperparameters in order to maximize the area under the ROC curve (AUROC) was obtained with Keras Tuner[7]. Next, we retrain the model with the optimal hyperparameters to evaluate the results. The last

[5]https://fred.stlouisfed.org/.
[6]https://www.tensorflow.org/.
[7]https://www.tensorflow.org/tutorials/keras/keras_tuner.

step is the transfer learning strategy. The weight layers of the network for Euro and Brazillian data were copied from the network trained on the U.S. data. It is similar to the last row of Figure 1, except that we do not retrain the parameters. It is as if these two datasets function as out-of-sample.

4 Results

Our combined cross-sectional and shuffle approach implies slightly different results each time we run the model due to different compositions in the selected data sets for training and validation. However, the well-known excellent neural network generalization ability resulted in minimum differences, generating similar performances in the various runs. The datasets we deal with and the code we wrote, available at https://github.com/rrsguim/PhD_Economics, allow anyone to evaluate additional runs.

Some conclusions can be drawn from the results. First, the deep learning model learned from NBER how to classify business cycles (Figure 2). The complete sample contains 211 observations. In this run, there are 44 observations in the test set, an AUROC at 0.9099 and a confusion-matrix analysis with 36 expansions points detected, or true negatives (TN), one expansion point incorrectly detected, a false positive (FP), one recession point missed, a false negative (FN), and five recessions point detected, or true positives (TP), considering a threshold of 0.5. Second, the transfer learning strategy to other datasets, the Euro Area and Brazil, confirms the excellent performance of the proposed classifier (Figures 3 and 4). There are 60 observations for the Euro Area, with 49 TN, zero FP, two FN, and nine TP, and, for Brazil, there are 80 observations, with 61 TN, one FP, six FN, and twelve TP. Finally, it is essential to note that sometimes a misclassification like a false negative refers to a quarter surrounded by true negatives. For example, the six false negatives referring to Brazil's data do not represent six missed recessions, but six quarters not captured and dispersed throughout the sample - Figure 4 shows that the four Brazilian recessions have been identified. In Section 5, we comment on possible strategies to address this point.

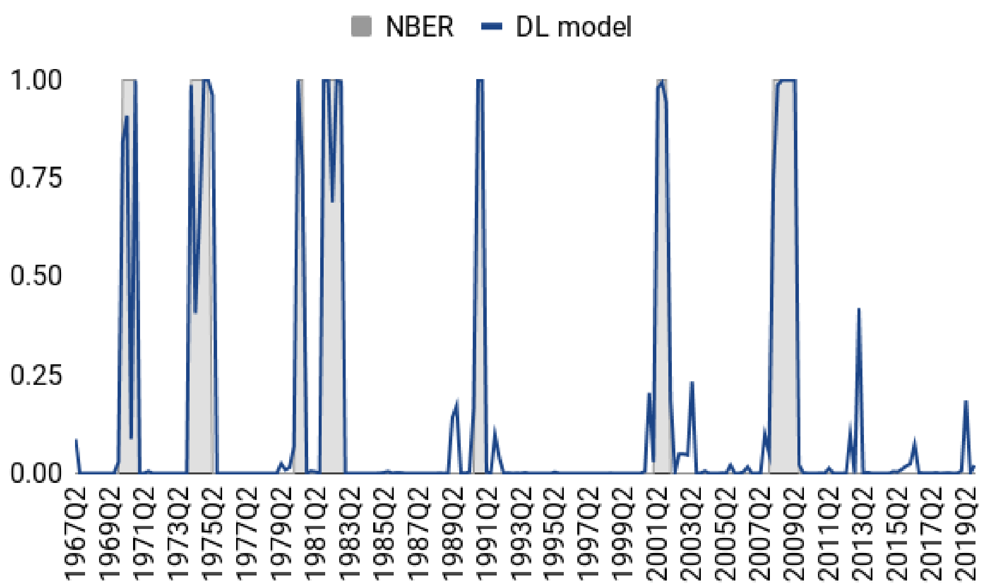

Figure 2: Deep Learning with the U.S. data

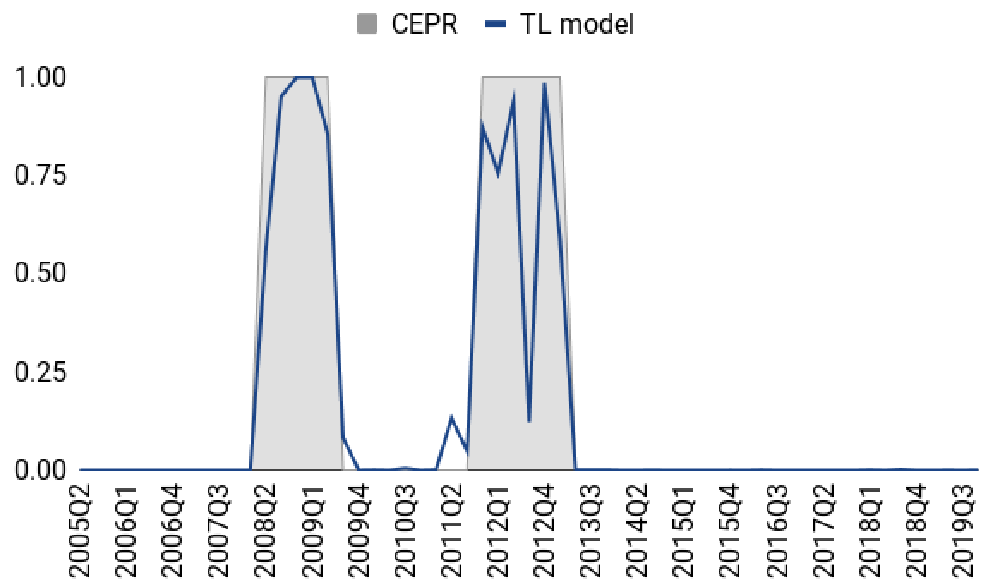

Figure 3: Transfer Learning with the Euro Area data

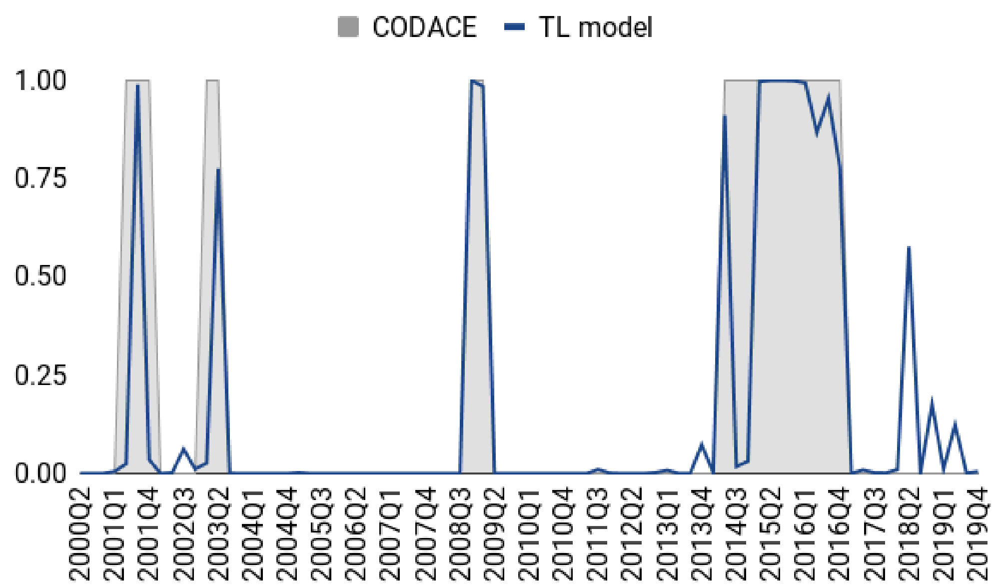

Figure 4: Transfer Learning with Brazilian data

5 Discussion

In this paper, we have constructed a deep neural network model that learned from NBER how to identify the United States business cycle phases and then applied the knowledge gained, through transfer learning, to identify the Euro area and the Brazilian business cycles. This combined approach has demonstrated excellent performance compared to CEPR and CODACE dating committees, emerging as a potential supplementary tool for governments and the private sector to conduct their activities in light of national and international economic conditions, since the model can also be applied to a country or region that do not have a business cycle dating committee. We kept the setup model as simple as possible since we wanted to verify the suitability of a transfer learning strategy for business cycle identification and avoid overfitting, a well-know issue when dealing with machine learning. Depending on the main focus, future work might include time dependency, monthly data, additional features, and fine-tuning strategies as retraining the model through unlocking weights when transfer learning.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: https://www.tensorflow.org/. software available from tensorflow.org.

Abberger, K., Graff, M., Campelo, A.J., Gouveia, A.C.L., Müller, O., Sturm, J.E., 2020. The Global Economic Barometers: Composite indicators for the world economy. Research Working Paper KOF WP 471-20. KOF Swiss Economic Institute.

Bengio, Y., 2009. Learning Deep Architectures for AI. Foundations and Trends in Machine Learning 2, 1–127.

Berge, T.J., 2013. Predicting recessions with leading indicators: model averaging and selection over the business cycle. Research Working Paper RWP 13-05. Federal Reserve Bank of Kansas City.

Bishop, C.M., 1994. Neural Networks for Pattern Recognition. Oxford University Press.

Burns, A.F., Mitchell, W.C., 1946. Measuring Business Cycles. National Bureau of Economic Research, Inc.

Camacho, M., Perez-Quiros, G., Poncela, P., 2018. Markov-switching dynamic factor models in real time. International Journal of Forecasting 34, 598–611.

Chauvet, M., 1998. An econometric characterization of business cycle dynamics with factor structure and regime switching. International Economic Review 39, 969–96.

Chauvet, M., 2001. A monthly indicator of brazilian gdp. Brazilian Review of Econometrics 21, 1–47.

Chauvet, M., Hamilton, J.D., 2005. Dating Business Cycle Turning Points. NBER Working Papers. National Bureau of Economic Research, Inc.

Chauvet, M., Piger, J., 2008. A comparison of the real-time performance of business cycle dating methods. Journal of Business Economic Statistics 26, 42–49.

Chauvet, M., Yu, C., 2006. International business cycles: G7 and OECD countries. Economic Review 91, 43–54.

Cook, T.R., Hall, A., 2017. Macroeconomic Indicator Forecasting with Deep Neural Networks. Research Working Paper RWP 17-11. Federal Reserve Bank of Kansas City.

Cuba-Borda, P., Mechanick, A., Raffo, A., 2018. Monitoring the World Economy : A Global Conditions Index. IFDP Notes. Board of Governors of the Federal Reserve System (U.S.).

Estrella, A., Mishkin, F., 1998. Predicting u.s. recessions: Financial variables as leading indicators. The Review of Economics and Statistics 80, 45–61.

Estrella, A., Rodrigues, A.P., Schich, S., 2000. How stable is the predictive power of the yield curve? evidence from Germany and the United States. Staff Reports 113. Federal Reserve Bank of New York.

Garbellano, J., 2016. Nowcasting recessions with machine learning: New tools for predicting the business cycle, in: Thesis.

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

Giusto, A., Piger, J., 2017. Identifying business cycle turning points in real time with vector quantization. International Journal of Forecasting 33, 174–184.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

Gu, S., Kelly, B., Xiu, D., 2018. Empirical asset pricing via machine learning. Chicago Booth Research Paper 18.

Hamilton, J.D., 1989. A New Approach to the Economic Analysis of Nonstationary Time Series and the Business Cycle. Econometrica 57, 357–384.

Harding, D., Pagan, A., 2005. A suggested framework for classifying the modes of cycle research. Journal of Applied Econometrics 20, 151–159.

Hoptroff, R.G., Bramson, M.J., Hall, T.J., 1991. Forecasting economic turning points with neural nets, in: IJCNN-91-Seattle International Joint Conference on Neural Networks, pp. 347–352.

Jackson, B., Rege, M., 2019. Machine learning for classification of economic recessions, in: 2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI), pp. 31–38.

James, A., Abu-Mostafa, Y.S., Qiao, X., 2019. Nowcasting recessions using the svm machine learning algorithm. arXiv:1903.03202.

Kauppi, H., Saikkonen, P., 2008. Predicting u.s. recessions with dynamic binary response models. The Review of Economics and Statistics 90, 777–791.

Klinkenberg, R., 2003. Predicting phases in business cycles under concept drift.

Ma, J.B., 2015. Applications of machine learning in forecasting recessions: boosting united states and japan.

Makridakis, S., 2017. The forthcoming artificial intelligence (ai) revolution: Its impact on society and firms. Futures 90, 46–60.

Makridakis, S., Spiliotis, E., Assimakopoulos, V., 2018. Statistical and machine learning forecasting methods: Concerns and ways forward. PLoS ONE 13.

Morik, K., Rüping, S., 2002. A multistrategy approach to the classification of phases in business cycles, in: Elomaa, T., Mannila, H., Toivonen, H. (Eds.), Machine Learning: ECML 2002, Springer Berlin Heidelberg. pp. 307–318.

Nasr, G.E., Dibeh, G., Achkar, A., 2007. Predicting business cycle turning points with neural networks in an information-poor economy, in: SCSC.

Nyman, R., Ormerod, P., 2017. Predicting economic recessions using machine learning algorithms. arXiv:1701.01428.

Piger, J., 2020. Turning points and classification, in: Fuleky, P. (Ed.), Macroeconomic Forecasting in the Era of Big Data. Springer International Publishing, pp. 585–624.

Pratt, L.Y., Mostow, J., Kamm, C.A., Kamm, A.A., 1991. Direct transfer of learned information among neural networks, in: Proceedings of AAAI-91, pp. 584–589.

Qi, M.H., 2001. Predicting us recessions with leading indicators via neural network models, in: Princeton University Senior Theses.

Raffinot, T., Benoit, S., 2019. Investing Through Economic Cycles with Ensemble Machine Learning Algorithms. Working Papers. HAL.

Rudebusch, G., Williams, J., 2009. Forecasting recessions: The puzzle of the enduring power of the yield curve. Journal of Business Economic Statistics 27, 492–503.

Sebastian, F., 2016. Dating US business cycles with macro factors. Studies in Nonlinear Dynamics & Econometrics 20, 529–547.

Soybilgen, B., 2018. Identifying US business cycle regimes using dynamic factors and neural network models. MPRA Paper. University Library of Munich, Germany.

Stergiou, C., Siganos, D., 2011. Neural networks. Technical Report. Imperial College London.

Tiffin, A., 2016. Seeing in the dark: a machine-learning approach to nowcasting in lebanon. IMF Working Paper .

Torrey, L., Shavlik, J., 2009. Transfer learning, in: Olivas, E.S., Guerrero, J.D.M., Sober, M.M., Benedito, J.R.M., López, A. (Eds.), Handbook Of Research On Machine Learning Applications and Trends: Algorithms, Methods and Techniques (2 Volumes). chapter 11.

Yosinski, J., Clune, J., Bengio, Y., Lipson, H., 2014. How transferable are features in deep neural networks?, in: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems 27. Curran Associates, Inc., pp. 3320–3328. URL: http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks.pdf.

Appendix A Appendix

A.1 Feedforward neural networks

A deep neural network is an artificial neural network with multiple layers hidden between the input and output layers (Bengio, 2009). There are variations of deep neural networks, such as convolutional, recurrent or recursive nets. We adopted *deep feedforward networks* (DFN), also called *feedforward neural networks* or *multilayer perceptrons* (Goodfellow et al., 2016, 164). These models are called feedforward because information flows through the function being evaluated from inputs, through the intermediate computations used to define the function, and finally to the output target. There are no feedback connections in which the outputs of the model are fed back into itself. A feedforward network with a single layer is sufficient to represent any function, but the layer might be infeasibly large and might fail to learn and generalize correctly. In many circumstances, using more in-depth models can reduce the number of units required to represent the desired function and can reduce the amount of generalization error.

For instance, in a feedforward network having two layers there are d inputs, M hidden units and c output units. We can write down the correspondent analytic function as follows (Bishop, 1994, 118-9). The output of the jth hidden unit is obtained by first forming a weighted linear combination of the d input values, and adding a bias, to give

$$a_j = \sum_{i=1}^{d} w_{ji}^{(1)} x_i + w_{j0}^{(1)} \ . \tag{1}$$

Here $w_{ji}^{(1)}$ denotes a weight in the first layer, going *from* input i *to* hidden unit j, and $w_{j0}^{(1)}$ denotes the bias for hidden unit j. The bias term for the hidden units is made explicit by the inclusion of an extra input variable x_0 whose value is permanently set at $x_0 = 1$. This can be represented analytically by rewriting (1) in the form

$$a_j = \sum_{i=0}^{d} w_{ji}^{(1)} x_i \ . \tag{2}$$

The activation of hidden unit j is then obtained by transforming the linear sum in (2) using an activation function $g(\cdot)$ to give

$$z_j = g(a_j) \ . \tag{3}$$

The outputs of the network are obtained by transforming the activations of the hidden units using a second layer of processing elements. Thus, for each output unit k, we construct a linear combination of the outputs of the hidden units of the form

$$a_k = \sum_{j=1}^{M} w_{kj}^{(2)} z_j + w_{k0}^{(2)} \ . \tag{4}$$

Again, we can absorb the bias into the weights to give

$$a_k = \sum_{j=0}^{M} w_{kj}^{(2)} z_j \ . \tag{5}$$

which can be represented by including an extra hidden unit with activation $z_0 = 1$. The activation of the kth output unit is then obtained by transforming this linear combination using a non-linear activation function, to give

$$y_k = \tilde{g}(a_k) \ . \tag{6}$$

Here we have used the notation $\tilde{g}(\cdot)$ for the activation function of the output units to emphasize that this need not be the same function as used for the hidden units. If we combine (2), (3), (5) and (6) we obtain an explicit expression for the complete function in the form

$$y_k = \tilde{g}\left(\sum_{j=0}^{M} w_{kj}^{(2)} g\left(\sum_{i=0}^{d} w_{ji}^{(1)} x_i \right) \right) \ . \tag{7}$$

According to (Goodfellow et al., 2016, 173-4), the cost functions for neural networks are more or less the same as those for other parametric models, such as linear models. The total cost function used to train a neural network will often combine cost functions with a regularization term in order to prevent overfitting[8]. Regularization is any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error. Regularization is one of the central concerns of the field of machine learning, rivaled in its importance only by optimization (Goodfellow et al., 2016, 117).

Back-propagation is a method to calculate a gradient that is needed in the calculation of the weights to be used when training the network. Back-propagation is a special case of an older and more general technique called automatic differentiation. In the context of learning, back-propagation is commonly used by the optimization algorithm to adjust the weight of neurons by calculating the gradient of the loss function. This technique is also sometimes called backward propagation of errors, because the error is calculated at the output and distributed back through the network layers. (Goodfellow et al., 2016, 213) describe the general back-propagation procedure.

The primary architectural considerations are choosing the depth of the network and the width of each layer. Deeper networks are often able to use far fewer units per layer and far fewer parameters, as well as frequently generalizing to the test set, but they also tend to be harder to optimize. The ideal network architecture for a task must be found via experimentation guided by monitoring the validation set error (Goodfellow et al., 2016, 194).

Among potential choices for the nonlinear activation function, such as sigmoid, hyperbolic, softmax, there is a popular functional form in recent literature known as the rectified linear unit (ReLU), defined as

$$ReLU(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{otherwise.} \end{cases} \tag{8}$$

A.2 Data description

[8]See Goodfellow et al. (2016), Chapters 6 and 7 for further discussion.

Table 1: Dataset

FRED Code	Series	Start	Characteristics and transformations	Source
	UNITED STATES			
USRECQ	NBER based Recession Indicator	1967:Q1	Recession: 1 = true ; 0 = false	NBER
GDPC1	Real Gross Domestic Product	1967:Q1	Billions of Chained 2012 Dollars, s.a., first log-diff.	FRED
PIECTR	Real personal income ex current transfers	1967:Q1	Billions of Chained 2012 Dollars, s.a., first log-diff.	FRED
PRS85006013	Nonfarm Business Sector employment index	1967:Q1	Index 2012 = 100, s.a., first log-diff.	FRED
IPB50001SQ	Industrial production index	1967:Q1	Index 2012 = 100, s.a., first log-diff.	FRED
CQRMTSPL	Real manufacturing and trade ind. sales	1967:Q1	Millions of Chained 2012 Dollars, s.a., first log-diff.	FRED
	EURO AREA			
N/A	CEPR based Recession Indicator	2005:Q1	Recession: 1 = true — 0 = false	CEPR
CLVMNACSCAB1GQEA19	Real Gross Domestic Product (19 countries)	2005:Q1	Millions of Chained 2010 Euros, s.a., first log-diff.	FRED
NAEXKP02EZQ189S	Private Final Consumption Expenditure	2005:Q1	Billions of Chained 2012 Dollars, s.a., first log-diff.	FRED
LFESEETTEZQ647S	Employees	2005:Q1	Persons, s.a., first log-diff.	FRED
PRMNTO01EZQ657S	Total Manufacturing Production	2005:Q1	Growth Rate Previous Period, s.a., log-diff.	FRED
SLRTTO01EZQ657S	Volume of Total Retail Trade sales	2005:Q1	Growth Rate Previous Period, s.a., log-diff.	FRED
	BRAZIL			
N/A	CODACE based Recession Indicator	2000:Q1	Recession: 1 = true — 0 = false	CODACE
NAEXKP01BRQ652S	Total Gross Domestic Product	2000:Q1	Chained 2000 Real, s.a., first log-diff.	FRED
NAEXKP02BRQ189S	Private Final Consumption Expenditure	2000:Q1	Chained 2000 Real, s.a., first log-diff.	FRED
N/A	Registered Employees Index	2000:Q1	Index Dez-2001 = 100, s.a., first log-diff.	BCB
BRAPROINDQISMEI	Production of Total Industry	2000:Q1	Index 2015 = 100, s.a., first log-diff.	FRED
BRASARTQISMEI	Total Retail Trade	2000:Q1	Index 2015 =100, s.a., first log-diff.	FRED