Received: 5 July 2018 | Revised: 10 March 2019 | Accepted: 3 January 2020
DOI: 10.1002/for.2658

RESEARCH ARTICLE

WILEY

Identifying US business cycle regimes using dynamic factors and neural network models

Barıș Soybilgen

Center for Financial Studies, Istanbul Bilgi University, Istanbul, Turkey

Correspondence
Barıș Soybilgen, Center for Financial Studies, Istanbul Bilgi University, Santralistanbul, 34060, Istanbul, Turkey.
Email: baris.soybilgen@bilgi.edu.tr

Abstract

We use dynamic factors and neural network models to identify current and past states (instead of future) of the US business cycle. In the first step, we reduce noise in data by using a moving average filter. Dynamic factors are then extracted from a large-scale data set consisted of more than 100 variables. In the last step, these dynamic factors are fed into the neural network model for predicting business cycle regimes. We show that our proposed method follows US business cycle regimes quite accurately in-sample and out-of-sample without taking account of the historical data availability. Our results also indicate that noise reduction is an important step for business cycle prediction. Furthermore, using pseudo real time and vintage data, we show that our neural network model identifies turning points quite accurately and very quickly in real time.

KEYWORDS

business cycle, dynamic factor model, neural network, recession

1 | INTRODUCTION

Whether the USA is in a recession or an expansion at any given time is crucial information for all economic agents in the USA and around the globe. In particular, identifying the start of a recession as early as possible may help policymakers to take necessary precautions for the economy. However, the business cycle dating committee of the National Bureau of Economic Research (NBER), which currently maintains the chronology of the US business cycle, has historically announced business cycle turning points with a significant delay. Therefore, over the years, many business cycle dating methodologies have been proposed in the literature.[1] In this study, we use dynamic factor models (DFM) and neural network (NN) models to determine business cycle states of current and past periods in real time. We predict recessions and expansions in three steps. First, we filter noise in the data set using a moving average filter. Second, we use the DFM proposed by Giannone, Reichlin, and Small (2008) to extract a handful of dynamic factors from a large number of data series. Finally, we feed these dynamic factors into NNs to determine recession and expansion periods in real time.

Predicting economic variables by factors extracted from large/medium-scale data sets is a widespread approach in the literature,[2] but this is not still common for predicting business cycle regimes for future or current periods. In one of the notable studies, Fossati (2016) uses probit and Markov switching models with factors to determine current business conditions. For predicting future business cycle regimes, Bellégo and Ferrara (2009) extract static factors from 13 variables and feed them into a probit model to forecast euro area recessions. Chen, Iqbal, and Lai (2011) also follow a similar approach by extracting factors from a data set including 131 variables and inserting them into probit models to predict recessions in the US economy.

[1]See Hamilton (2011) for a survey of models that aim to identify turning points in real time.

[2]See Eickmeier and Ziegler (2008) for a meta-analysis of factor forecast applications for output and inflation and see Bańbura, Giannone, Modugno, and Reichlin (2013) for factor nowcasting applications for output.

Furthermore, Fossati (2015) forecasts US recessions using a probit model with factors, but he uses dynamic factors instead of static ones and a smaller data set. Finally, Christiansen, Eriksen, and Møller (2014) use factors with probit models to test the predictive ability of sentiment variables for US recessions.

Except for Giusto and Piger (2017), who use a simple machine learning algorithm known as learning vector quantization (LVQ) to identify turning points for the US business cycle, other studies that use factors to predict current or future recession and expansion periods utilize parametric models. Given that the true data-generating process is unknown, a nonparametric approach may be more appropriate for predicting US business cycle regimes. Our interest lies in nonparametric NN algorithms. NNs have been successfully applied to problems in computer science, engineering, medical, and financial applications. However, NNs are rarely used for predicting business cycle regimes in real time. One notable exception is the study of Qi (2001), which uses a two-layered NN model for 1- to 8-quarter ahead out-of-sample business cycle state predictions. Qi uses NNs with one or two variables to obtain predictions for the US business cycle regimes. Compared to Qi, we use a large-scale data set consisting of more than 100 variables. We focus on identifying business cycle regimes of current and past periods instead of forecasting whether there will be a recession or expansion in coming periods because our model is based on the nowcasting methodology of Giannone et al. (2008), which is best suited to obtaining predictions of the target variable for the present or the very recent past.

In this study, we use dynamic factors and NN models to identify business cycle regimes in real time. First, we show that the NN model follows the NBER's business cycle chronology quite accurately in-sample and out-of-sample without taking account of the historical data availability between 1960:07 and 2016:12. In that exercise, our results also indicate that noise reduction increases the prediction performance of NN models. We then test the identification performance of our model by replicating the historical data availability in each estimation period for 1979:01–1983:12 and 1990:01–2016:12. We also adopt a two-step turning-point detection strategy similar to Chauvet and Piger (2008) to make sure that our results are comparable with those of the LVQ presented in Giusto and Piger (2017) and the results of dynamic factor Markov switching (DFMS) models shown in Chauvet and Piger (2008) and Piger (2018). For the period 1990:01–2016:12, we document that NN models determine turning points quite accurately and very quickly both in expansion and recession periods. For the period 1979:01–1983:12, the accuracy of NN models is a slightly off compared to the period 1990:01–2016:12, but NN models can still identify turning points very quickly during this period. Given that the NBER announces turning points of the US business cycle with a significant lag and most dating methodologies fail to determine US business cycle regimes in a timely fashion as shown by Hamilton (2011), our proposed methodology can be helpful for both policymakers and market participants to infer the current state of the economy without much delay. We also compare NN models against LVQ and DFMS models and show that NN models identify turning points much faster than those models.

The remainder of this paper is as follows. Section 2 introduces the methodology. Section 3 describes the data set. Section 4 presents the empirical results, and Section 5 concludes.

2 | METHODOLOGY

In this study, we use DFMs and NNs to determine business cycle regimes. Before extracting factors, we reduce noise in data by taking 5-month averages of monthly data, as noisy data can reduce the prediction performance of our models.[3] We then perform dimensionality reduction by employing a DFM, because using a full data set can cause overfitting of NNs and lead to a poor prediction performance due to irrelevant and noisy variables. A DFM is appropriate for reducing the dimension of a macroeconomic data set because a small number of factors is enough to capture most of the dynamics among macroeconomic data series.[4]

2.1 | The dynamic factor model

Let us assume that standardized and filtered n monthly series $x_t = (x_{1,t}, x_{2,t}, \dots, x_{n,t})', t = 1, 2, \dots, T$ as in Giannone et al. (2008) have the following approximate dynamic factor model:

$$x_t = \mu + \Lambda f_t + \xi_t; \quad \xi_t \sim \mathbb{N}(0, \Sigma_\xi), \tag{1}$$

$$f_t = \Phi(L) f_{t-1} + B\eta_t; \quad \eta_t \sim \mathbb{N}(0, I_q), \tag{2}$$

where μ is a constant, Λ is an nxr matrix of factor loadings, $f_t = (f_{1,t}, f_{2,t}, \dots, f_{r,t})'$ are unobserved common factors that satisfy $r \ll n$, and ξ_t is an idiosyncratic component assumed to be multivariate white noise with diagonal covariance matrix Σ_ξ. As shown in Equation (2), f_t is assumed to follow a vector autoregression process driven by a q-dimensional vector of common shocks, η_t, that follows a white-noise process. B is an $r \times q$ matrix of full rank q with $q \leqslant r$ and $\varphi(L)$ is an $r \times r$ lag polynomial matrix.

[3]In Section 4.1, we also present results with 3 months' moving averages and no filter to show the importance of noise filtering.
[4]See Sargent and Sims (1977) and Giannone, Reichlin, and Sala (2005).

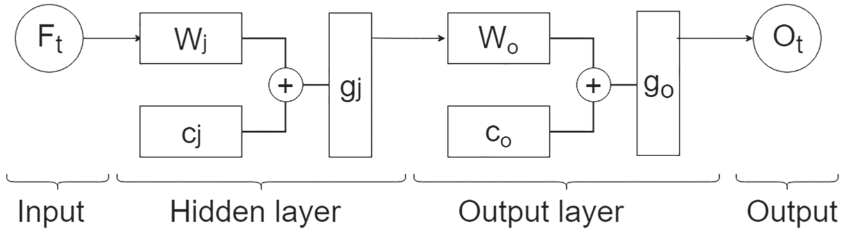

FIGURE 1 A two-layered feedforward NN

Following Giannone et al. (2008), we use a two-step estimation approach to obtain common factors. In the first step, the initial estimate of common factors are obtained by the principal component analysis, and then parameters of the model are estimated via OLS using only the balanced part of the data set. In the second step, estimates of common factors are obtained via Kalman smoother for both the balanced part and the unbalanced part of the data set. [5]

2.2 | The neural network model

After obtaining common factors, we feed those into NNs to identify US business cycle regimes. Let y_t be a categorical variable that shows NBER recession periods as 1 and NBER expansion periods as 0. We then use the following two-layered feedforward NN model, which is also represented in Figure 1:

$$O_t = g_o[W_o g_j(W_j \hat{F}_t + c_j) + c_o], \tag{3}$$

where $O_t = (1 - y_t, y_t)$ is the output, $\hat{F}_t$ are estimated dynamic factors that are standardized to zero mean and unit variance, W_j is an $s \times r$ matrix of weights in the hidden layer, W_o is a $2 \times s$ matrix of weights in the output layer, c_j is an $s \times 1$ vector of ones in the hidden layer, c_o is a 2×1 vector of ones in the output layer, g_j is a tan-sigmoid transfer function, and g_o is a soft-max transfer function. Finally, s is the number of neurons in a hidden layer.

There are various backpropagation algorithms to train an NN model. In general, a backpropagation algorithm first assigns initial values to weights, then the initial output is calculated using initial weights. Afterwards, the initial output is compared with actual values using a loss function, and the error values are propagated backwards via gradients to neurons in previous layers. The backprogation algorithm uses these error values to update the weights. Another set of outputs is calculated using new weights and this process continues until the error threshold, the minimum performance gradient, or the maximum number of iterations is reached.[6]

Assuming that the number of patterns is proportional to the number of weights and biases (N), the computational complexity of the simple gradient backpropagation algorithm is $O(3N^2)$ (Møller, 1993). Even though the complexity of the simple gradient backpropagation is low, it is often slow in converging due to the fact that weights are only adjusted according to steepest descent direction. An alternative approach to the simple gradient backpropagation is conjugate gradient backpropagation with line search (CGL), which performs a line search along conjugate directions for faster convergence. As shown by Møller (1993), CGL's computational complexity is around $O(6 - 20N^2)$, which is higher than the simple gradient backpropagation's complexity, but the CGL updates weights by combining the steepest descent direction and the previous search direction, which increases the convergence speed on average. We use one of the popular variants of conjugate gradient backpropagation algorithms with Polak–Ribière updates (CGPR)[7] to train the NN model.[8]

After obtaining estimated weights, the prediction for the current period obtained at time t, $\hat{O}_{t,t} = [\text{Prob}(\hat{y}_{t,t} = 0|\hat{F}_t), \text{Prob}(\hat{y}_{t,t} = 1|\hat{F}_t)]$, are computed as follows:

$$\hat{O}_{t,t} = g_o[\hat{W}_o g_j(\hat{W}_j \hat{F}_t + \hat{c}_j) + \hat{c}_o]. \tag{4}$$

As the loss function, we use the mean squared error (MSE). For stopping criteria, we set the error term goal as 10^{-5}, the minimum performance gradient as 10^{-7}, and the maximum number of iterations as 1,000. Furthermore, we use an early stopping technique with six maximum cross-validation failures to prevent overfitting.[9]

3 | THE DATA SET

Our data set is based on the large-scale data set of McCracken and Ng (2016) (FRED-MD), including data for output, income, labor market, housing, consumption, orders, inventories, money, credit, interest rates, exchange rates, prices, and the stock market. We choose this data set because it is publicly available to all researchers, it is updated monthly, and revisions are handled by data

[5]See Doz, Giannone, and Reichlin (2011) for the properties of the two-step estimator.

[6]We use the Matlab 2017b Neural Network Toolbox to train the NN model. Unless otherwise stated, default parameters of the Neural Network Toolbox are used for the backpropagation algorithm.

[7]For a detailed description of the CGPR algorithm see Demuth, Beale, De Jess, and Hagan (2014).

[8]Other variants of conjugate gradient backpropagation algorithms also yield similar results.

[9]For various values of stopping criteria, NNs produce very similar results. Implementation of the backpropagation algorithms in Matlab 2017b is quite robust in that sense.

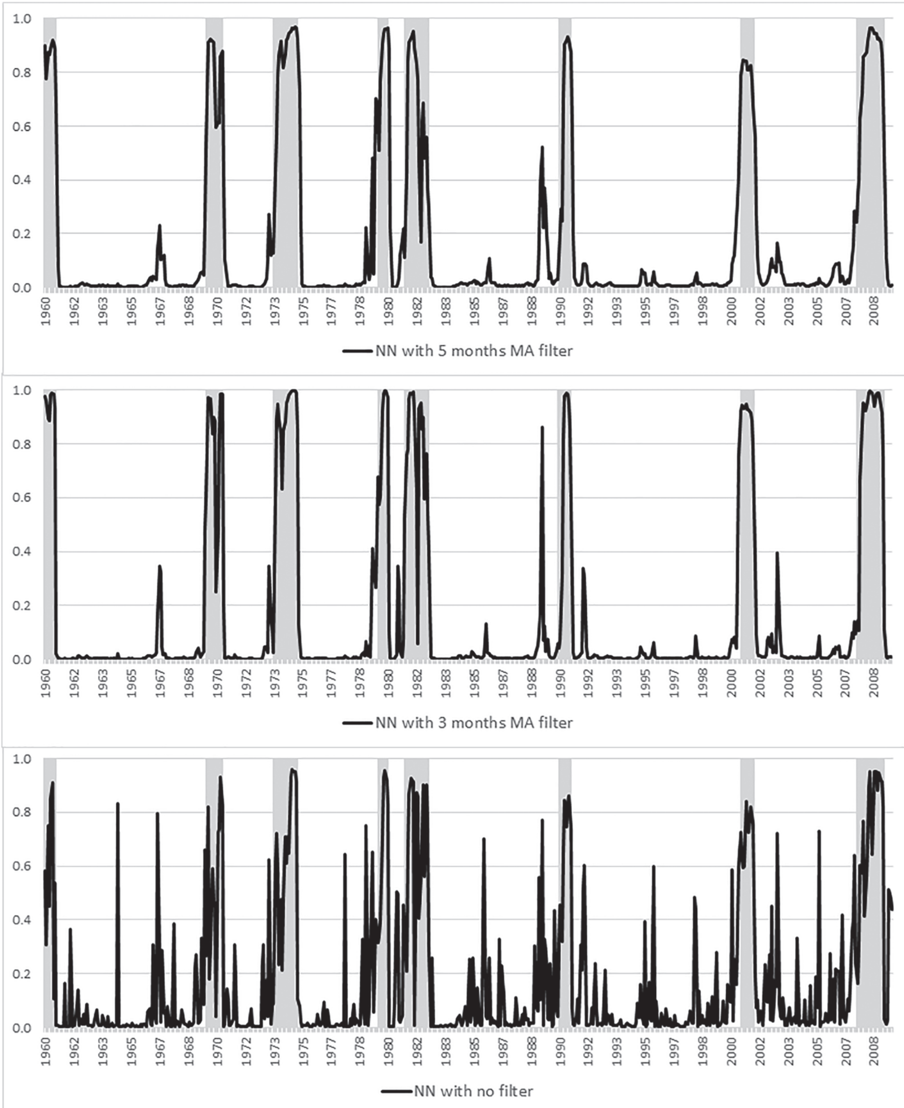

FIGURE 2 Predictions of NN models with different filters (1960:07–2016:12). In the upper and middle panels, data are filtered by the 5-month moving average filter and the 3-month moving average filter, respectively. In the lower panel, data are not filtered. Factors are obtained by using a DFM with $r = 2$, $q = 2$, $p = 1$

specialists. We have vintage data from 1999:08 onwards. During this period, new variables are sometimes added to the FRED-MD data set and some variables are discarded. Variables and their period of use are listed in the Appendix. Furthermore, all variables are transformed appropriately to ensure stationarity. Their applied transformations are also shown in the Appendix.

In this study, our aim is to predict recession and expansion periods in the US economy. As a result, our dependent variable is a binary categorical variable that shows recession periods as 1 and other periods as 0. We determine recession and expansion periods according to the Business Cycle Dating Committee of the NBER, which currently maintains a chronology of the US business cycle.[10] According to NBER's dating of trough and peak points of the US economy, we define an expansion as a period following a trough until a peak is announced. Remaining periods are defined as recession.

4 | EMPIRICAL RESULTS

In this section, we first show how well NN models fit data in-sample and out-of-sample without taking account of the historical data availability and we perform robustness checks. We then determine the NBER turning points recursively using a pseudo real-time and vintage data set between 1990:01 and 2016:12. Furthermore, we analyze the identification performance of NN models during the

[10]Instead of a regular definition of an economic recession in terms of two consecutive quarters of decline in real gross domestic product (GDP), the committee does not have a fixed definition of a recession. They analyze a broad range of economic indicators including real manufacturing and trade sales, industrial production index, real personal income less transfers, aggregate hours of work in the total economy, payroll survey employment, household survey employment, as well as monthly and quarterly GDP to assess contraction and expansion dates.

TABLE 1 QPSs of NNs with different filters

	No MA	3 MA	5 MA
Estimation period	0.064	0.025	0.034
Test period	0.038	0.019	0.018

Note. "No MA" indicates that data are not filtered; "3 MA" and "5 MA" indicate that data are filtered by the 3-month moving average filter and the 5-month moving average filter, respectively. The estimation period is between 1960:07 and 1989:12. The test period is between 1990:01 and 2016:12.

1980s recessions using only a pseudo real-time data set. We also compare NN models with LVQ and DFMS models.

To evaluate the prediction performance of models in Section 4.1, the quadratic probability score (QPS), which is equivalent to the MSE for probability predictions, is used. The QPS is defined as follows:

$$\mathrm{QPS}_t = 2/T\sum_{t=1}^{T}[\mathrm{Prob}(\hat{y}_t = 1|\hat{F}_t) - y_t]^2. \tag{5}$$

The range of QPS is between 0 and 2, and smaller values indicate better forecasting performance.

To extract factors, we use a DFM with $r = 2, q = 2, p = 1$ as in Giannone et al. (2008).[11] For each NN model, the neuron structure in the hidden layer is determined according to the performance of the NN model in the initial estimation period. We test the number of neurons up to 10 and choose the neuron structure that minimizes the QPS in the initial estimation period.[12]

NN models are sensitive to initial weights. Therefore, we run NNs 100 times in each estimation window to ensure robustness of the results. We then use equal weights to combine outputs of all 100 NN models.

4.1 | The fit of models and robustness checks

To evaluate the goodness of fit, we present predictions of NNs for the whole sample. We use the 2017:01 FRED-MD data set, which contains the period between 1960:07 and 2016:12.[13] To be in line with the out-of-sample identification exercise performed in the next section, the estimation period is restricted to the period covering 1960:07–1989:12 and the rest is used for the test period.

TABLE 2 QPSs of NNs with different DFM specifications

DFM			Estimation period		Test period	
r	q	p	QPS	Rank	QPS	Rank
1	1	1	0.071	31	0.043	23
1	1	2	0.068	29	0.042	21
1	1	3	0.071	32	0.043	25
1	1	4	0.078	39	0.047	30
2	1	1	0.063	25	0.032	12
2	1	2	0.065	27	0.032	13
2	1	3	0.063	26	0.033	15
2	1	4	0.062	24	0.033	14
2	2	1	0.034	3	0.018	2
2	2	2	0.039	13	0.019	3
2	2	3	0.041	15	0.020	4
2	2	4	0.040	14	0.022	7
3	1	1	0.066	28	0.062	39
3	1	2	0.074	34	0.053	35
3	1	3	0.078	37	0.054	36
3	1	4	0.078	38	0.055	37
3	2	1	0.041	16	0.045	28
3	2	2	0.045	18	0.038	18
3	2	3	0.048	21	0.043	24
3	2	4	0.046	19	0.044	27
3	3	1	0.036	7	0.017	1
3	3	2	0.037	10	0.020	5
3	3	3	0.031	2	0.023	8
3	3	4	0.027	1	0.025	10
4	1	1	0.080	40	0.065	40
4	1	2	0.078	36	0.052	33
4	1	3	0.076	35	0.053	34
4	1	4	0.072	33	0.050	32
4	2	1	0.069	30	0.041	20
4	2	2	0.059	23	0.042	22
4	2	3	0.052	22	0.044	26
4	2	4	0.048	20	0.045	29
4	3	1	0.036	8	0.055	38
4	3	2	0.035	6	0.034	17
4	3	3	0.039	12	0.040	19
4	3	4	0.041	17	0.049	31
4	4	1	0.035	5	0.021	6
4	4	2	0.034	4	0.025	9
4	4	3	0.037	9	0.028	11
4	4	4	0.037	11	0.033	16

Note. r, q, and p show the number of static factors, dynamic factors, and lags of the DFM, respectively. The estimation period is between 1960:07 and 1989:12. The test period is between 1990:01 and 2016:12. The Rank column shows the rank of an NN according to the QPS.

To utilize the early stopping technique, the first 70% of the estimation period is used for training and the rest of the estimation period is reserved for cross-validation. Therefore, we use weights calculated in the training period (1960:07–1980:11) to calculate outputs in the cross-validation period (1980:12–1989:12) and the test period (1990:01–2016:12).

In the upper part of Figure 2, predictions from the baseline NN model are shown. The model seems to capture

[11] Determining the specification for a DFM is a difficult job. Alternatively, one can also use information criteria such as Bai and Ng (2002, 2007) to determine the specification of the DFM. However, as stated by Bańbura and Rünstler (2011), these criteria usually indicate a large number of factors, leading to volatile forecasts. Therefore, we follow a simple approach and use the specification of Giannone et al. (2008), which is quite a good specification in these kinds of forecasting exercises for the US economy. In Section 4.1, we also present results of other specifications.

[12] Results show that the number of neurons that yields the lowest QPS is 10 for the baseline NN model.

[13] We lose some data at the beginning of the sample due to transformations.

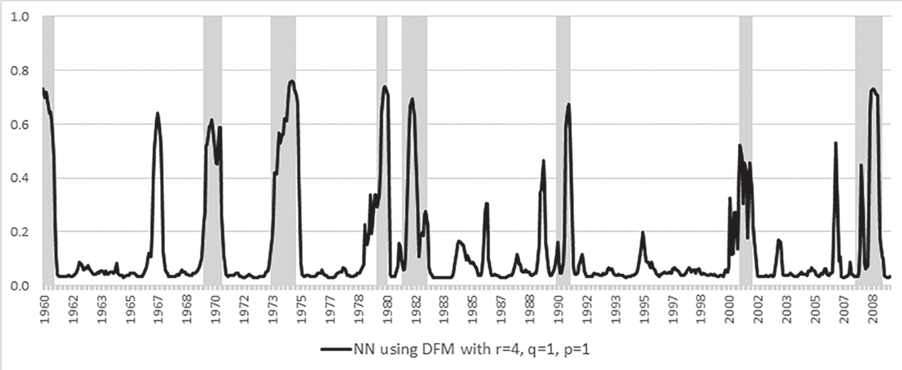

FIGURE 3 Predictions of the NN model using a DFM with $r = 4$, $q = 1$, $p = 1$ (1960:07–2016:12). Data are filtered by the 5-month moving average filter

recession periods quite well, especially in the test period. In the estimation period, the NN model cannot capture the second half of the 1981–82 recession. Furthermore, the NN model shows a spike in July 1989. The model shows that the probability of recession is about 50% in July 1989. As a robustness check, we also present results using a no-moving-average filter and a 3-month moving average filter in Figure 2. When noise reduction technique is not used, predictions from the NN model is highly noisy. There are many misclassified periods both in the estimation period and the test period. When 3 months moving average is used for smoothing data, noise in data mostly disappears. However, the problem in July 1989 becomes more apparent, and the NN model indicates more than 80% recession probability even though it is not a recession period. We also present the QPS of NNs with a 5-month moving average filter, a 3-month moving average filter, and no filter in Table 1. Interestingly, the 3-month moving average filter yields the lowest QPS in the estimation period and the 5-month moving average filter has the lowest QPS in the test period. As expected, noisy data reduce the prediction performance of NNs, and even slight noise reduction greatly improves the forecasting performance of NNs.

In our base model, we extract factors using a DFM with $r = 2$, $q = 2$, $p = 1$. Compared to other possible specifications, we show how this specification performs by presenting QPS results of NNs in Table 2 using the following DFM specifications: $1 \leq r \leq 4$, $1 \leq q \leq 4$, $q \leq r$ and $1 \leq p \leq 4$. Compared to other specifications, the baseline specification performs quite well even though it is not the best model. In general, models with r & $q = 2$, r & $q = 3$, and r & $q = 4$ perform quite well in many cases. To give an idea of how much worst-performing models predict, in Figure 3 we show predictions derived from an NN using a DFM with $r = 4$, $q = 1$, $p = 1$, which is the worst-performing specification. This model produces many false signals. For example, it shows more than 50% of recession probability in the first 5 months of 1967. Furthermore, it misses quite a few recession periods compared to the base model. However, it still roughly captures business cycle regimes.

4.2 | Real-time performance of models

In the previous section, we ignore historical data availability to assess the fit of models over the whole data sample and perform some robustness checks. In this section, we analyze the real-time performance of models. Unfortunately, we do not have vintage data before 1999:08,[14] so for detecting turning points between 1990:01 and 1999:07 we ignore historical data revisions and use the data set of 1999:08 while replicating historical data availability using a stylized calendar. For the period 1999:08–2016:12, we use vintage data for predicting business cycle regimes. We assume that predictions are produced twice per month. The first one is at the beginning of the month and the second one is in the middle of the month after industrial production is released. Furthermore, the FRED-MD vintage data set in each period does not have the same number of variables. There are two reasons for this: first, discontinuation of old series and introduction of new series; second, Haver Analytics did not collect some series at all times.

The publication lag of the NBER business cycle chronology is not as straightforward as other data series because the NBER historically announced turning points of the business cycle with a delay of between 4 and 21 months and did not release any official announcements that help us to update the information set between turning point announcements. To replicate historical data availability and update our information set continuously, despite the lack of any official NBER announcements during long expansion periods, we implement the following set of assumptions similar to Giusto and Piger (2017): (1) the date of a turning point is known once it is announced by the NBER; (2) a peak will be announced by the NBER with a maximum publication lag of 12 months; and (3) after a

[14]McCracken and Ng (2016) construct FRED-MD vintage data using historical Haver Analytics data, and the St. Louis FED has been backing up the Haver databases since August 1999.

TABLE 3 Business cycle peak and trough dates: NBER, DFMS, and LVQ (1990:01–2016:12)

NBER turning point		DFMS turning point		LVQ turning point	
Date	Announcement day	Date	Detection day	Date	Detection day
Peak points					
Jul-90	25-Apr-91	Jul-90	28-Feb-91	Jun-90	18-Oct-90
Mar-01	26-Nov-01	Jan-01	31-Jan-02	Mar-01	3-Nov-01
Dec-07	1-Dec-08	Jan-08	1-Jan-09	Feb-08	7-Jun-08
Trough points					
Mar-91	22-Dec-92	Mar-91	30-Sep-91	Apr-91	17-Jun-92
Nov-01	17-Jul-03	Nov-01	31-Aug-02	Jan-02	5-Oct-02
Jun-09	20-Sep-10	Jun-09	1-Jan-10	Jun-09	5-Dec-09

Note. First, third, and fifth columns show the peak and trough months established by the NBER, the DFMS, and the LVQ, respectively. Second, fourth, and sixth columns show the day, the peak and trough months are first identified by the NBER, the DFMS, and the LVQ, respectively. Results for the LVQ are obtained from Giusto and Piger (2017) and results for the DFMS are obtained from Chauvet and Piger (2008) and Piger (2018).

TABLE 4 Business cycle peak and trough dates: NN (1990:01–2016:12)

NN turning point		Lead/lag discrepancy	Days ahead of best competing model
Date	Detection day		
Peak points			
Sep-90	16-Jan-91	2	-90
Jan-01	1-May-01	-2	186
Jan-08	1-Apr-08	1	67
Trough point			
Apr-91	1-Aug-91	1	60
Dec-01	1-Apr-02	1	152
Jul-09	1-Oct-09	1	35

Note. The first column shows the peak and trough months established by the NN model. The second column presents the day; the peak and trough months are first identified by the NN model. The third column shows the difference between turning-point dates established by the NBER and the NN model. The fourth column shows the difference between detection lags of the NN model and the best competing model.

peak is announced by the NBER, the recession will last at least 6 months starting from the announced peak.

In this exercise, we determine the NBER turning points in each period recursively from 1990:01 to 2016:12. The initial estimation period is between 1960:07 and 1989:12. In each iteration, the first 70% of the estimation period is used for training and the rest of the estimation period is reserved for cross-validation. To show the performance of the NN, we compare our models against the LVQ presented in Giusto and Piger (2017) and the popular DFMS model proposed by Chauvet and Piger (2008). To be comparable with those models, we use a two-step turning-point detection strategy similar to Chauvet and Piger. We declare a recession when the probability of recession goes over 80% and remains there for three consecutive periods[15] and the peak month is identified as the first month prior to the month for which the probability of recession crosses 50%. In a similar manner, 80% threshold is switched with 20% when identifying the trough month.

Tables 3 and 4 present turning-point dates established by the NBER, the LVQ, the DFMS, and the NN, and when the turning point is detected by the NBER and competing models. First, it is clearly seen from Tables 3 and 4 that the NN establishes turning points more quickly than any other model and the NBER in nearly all cases. Only for the peak in 1990 does the LVQ establish turning points faster than the NN.

One of the reasons why our model is faster than competing models is that we also fill the missing data at the end of the sample using the DFM, unlike other competing models. Figure 4 shows predictions of the NN for $t, t-1$, and $t-2$. It can be seen from Figure 4 that predictions for $t-2$ capture business cycle regimes quite accurately. However, predictions for t and $t-1$ are quite noisy and miss many recession periods. These results show that the DFM is having a hard time while filling missing observations at the end of the sample. Nevertheless, when information from both nowcasting and backcasting is combined, the NN with factors identifies turning points very quickly without producing any false cycles.

[15]At each time, we predict both the current period and all previous periods where we lack information about the business cycle state. If three consecutive periods that have the recession probability of over 80% appear in that prediction time, we declare a recession.

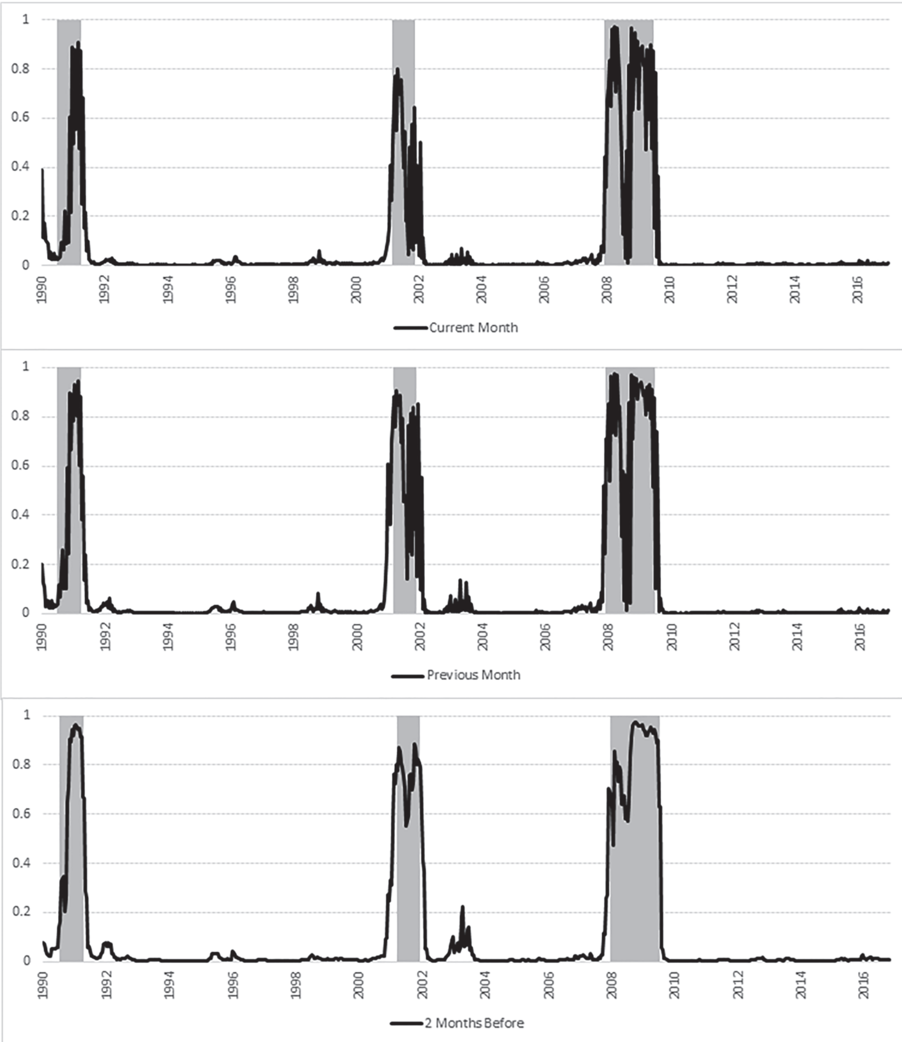

FIGURE 4 Predictions of the NN model for the current month (t), the previous month ($t-1$), and the 2 months before ($t-2$) (1990:01–2016:12). In the upper and middle panels, predictions for the current and previous months are displayed, respectively. In the lower panel, predictions for the 2 months before are shown. Data are filtered by the 5-month moving average filter. Factors are obtained by using a DFM with $r=2$, $q=2$, $p=1$

Furthermore, in some cases the NN is so much quicker than competing models in establishing the turning points that nowcasting the current period is irrelevant. For example, the NN establishes the peak and the trough in 2001 186 days and 152 days faster than the LVQ, respectively. On the other hand, the NN is slightly more inaccurate compared to the LVQ or the DFMS when establishing turning-point dates. However, slight inaccuracy is a small price compared to huge gains in detection time of the turning points.

Even though, we use an 80% threshold in the previous table to be comparable with Chauvet and Piger (2008) and Giusto and Piger (2017), the NN mostly produces recession probabilities close to 0 during expansion periods and above 50% probability during recession periods, especially when predicting $t-2$, as seen in Figure 4. Therefore, we can use lower thresholds without worrying about false signals, unlike Giusto and Piger.[16] In Table 5 we show turning point dates established by NN models with 70%, 60%, and 50% thresholds, and in Table 6 we present the identification lag of NN models with various thresholds.

Table 5 shows that turning point dates established by NN models do not change much with various thresholds. The only difference between models is that the peak of December 2007 is identified as January 2008 by the NN model with the 80% threshold, but NN models with lower thresholds identify the peak month as November 2007.

On the other hand, Table 6 shows that the identification lag of NN models declines sharply when thresholds are lowered. For peak periods, the average identification lag of the NN model with the 80% threshold is 96 days. When we decrease the threshold to 70% or lower, it reduces to 55 days. By reducing the threshold 10 percentage points, NN models identify peak points more than 40 days faster, on average. For trough periods, the average identification lag of the NN model with the 80% threshold is 122 days. When the threshold is reduced to 60%, 70%, and 50%, the average identification lag for trough periods becomes 111 days, 106 days, and 101 days, respectively. By reducing the threshold

[16]When Giusto and Piger (2017) decrease the threshold to 50%, their model produces three false recession periods.

TABLE 5 Business cycle peak and trough dates: NNs with 70%, 60%, and 50% threshold levels (1990:01–2016:12)

70% threshold	60% threshold	50% threshold
Peak points		
Sep-90	Sep-90	Sep-90
Jan-01	Jan-01	Jan-01
Nov-07	Nov-07	Nov-07
Trough points		
April-91	April-91	Apr-91
Dec-01	Dec-01	Dec-01
Jul-09	Jul-09	Jul-09

Note. Columns show the peak and trough months established by NN models with 70%, 60%, and 50% threshold levels, respectively.

TABLE 6 Identification lags of NNs with 80%, 70%, 60%, and 50% threshold levels (1990:01–2016:12)

80% threshold	70% threshold	60% threshold	50% threshold
Peak points			
168	135	135	135
30	0	0	0
91	31	31	31
Trough points			
122	91	91	91
121	121	104	90
122	122	122	122

Note. The identification lag for each turning point is measured as the number of days between the last day of the NBER turning point month and the day that turning point is first identified by the NN model.

TABLE 7 Business cycle peak and trough dates: NN (1979:01–1983:12)

NBER turning point		NN turning point		Lead/lag	Days ahead
Date	Announcement day	Date	Detection day	discrepancy	of LVQ
Peak points					
Jan-80	3-Jun-80	Sep-79	17-Dec-79	-4	137
Jul-81	6-Jan-82	Aug-81	16-Dec-81	1	-11
Trough points					
Jul-80	8-Jul-81	Jul-80	14-Nov-80	0	21
Nov-82	8-Jul-83	Jul-82	15-Oct-82	-4	182

Note. First and third columns show the peak and trough months established by the NBER and the NN, respectively. Second and fourth columns show the day; the peak and trough months are first identified by the NBER and the NN, respectively. The fifth column shows the difference between turning-point dates established by the NBER and the NN model. The last column shows the difference between detection lags of the NN model and the LVQ, which is the best competing model.

to 50%, there are 21 days of gain. Results also show that NN models can identify peaks much faster than troughs. This is important because the identification of peaks is much more important than the identification of troughs for both policymakers and market participants.

4.3 | Dating the early 1980s recessions

In the previous exercise, we start predicting business cycle states from the start of 1990 because NNs usually require a long training period to achieve a good prediction performance. However, business cycle dating studies also include the early 1980s recessions while testing business cycle identification performance of their models (e.g., Chauvet & Piger, 2008; Giusto & Piger, 2017; Hamilton, 2011). Therefore, our aim in this exercise is to test the performance of our models during early 1980s.

For this exercise, our initial estimation period is between 1960:07 and 1978:12, and we determine the NBER turning points in each period recursively from 1979:01 to 1983:12. To increase our training period, we do not use the early stopping technique[17] for this exercise. We assume that predictions are produced in the middle of the month after industrial production is released. All other assumptions are the same as in the previous exercise.

Table 7 presents the real-time performance of the NN model. Our results show that the NN model dates the turning points much faster than the NBER for both peaks and troughs. We also compare the identification speed of our model with other models in the last column of Table 7 and, on average, the NN model dates turning points faster than

[17]In general, the early stopping technique prevents overfitting and increases the speed of the estimation process.

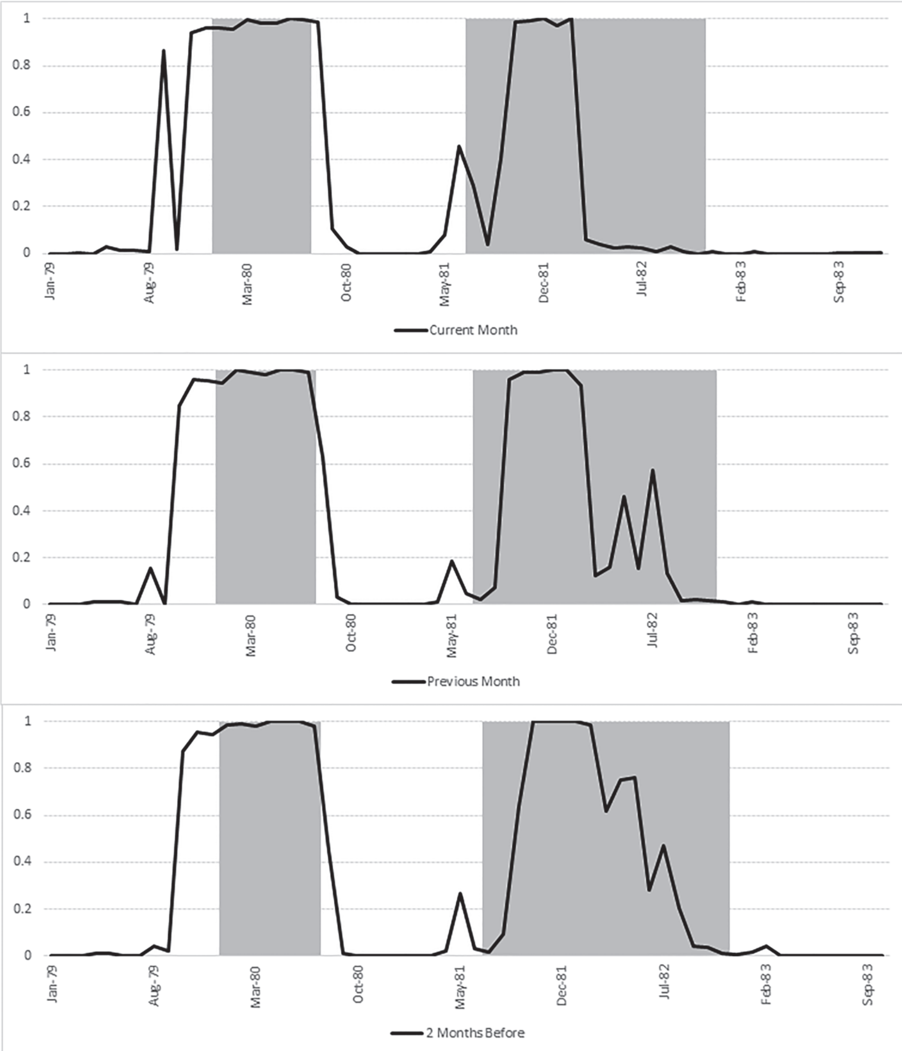

FIGURE 5 Predictions of the NN model for the current month (t), the previous month ($t-1$), and the 2 months before ($t-2$) (1979:01-1983:12). In the upper and middle panels, predictions for the current and previous months are displayed, respectively. In the lower panel, predictions for the 2 months before are shown. Data are filtered by the 5-month moving average filter. Factors are obtained by using a DFM with $r=2$, $q=2$, $p=1$

the competing models. However, the accuracy of the NN model is slightly off for the peak of the 1980 recession and the trough of the 1981–82 recession. The NN model dates those turning points 4 months earlier than the NBER. On the other hand, the NN model identifies the trough of the 1980 recession and the peak of the 1981–82 recession quite accurately. We also present predictions of the NN model for t, $t-1$, and $t-2$ in Figure 5. For the 1980 recession, we have quite smooth predictions, and the DFM seems to fill missing observations at the end of the sample quite accurately. For the 1981–82 recession, predictions for t and $t-1$ fail to identify the recession starting from April 1982, whereas predictions for $t-2$ are doing a much better job of capturing the recession. It is also seen from Figure 5 that the NN model identifies the the peak of the 1980 recession and the trough of the 1981–82 recession earlier than the NBER. For the period 1979:01–1983:12, the NN model manages to identify business cycle turning points quite fast, but compared to the period 1990:01–2016:12 the NN model identifies business cycle states less accurately during the early 1980s.

5 | CONCLUSION

In this study, we propose an NN model to determine the current state of the US business cycle. We estimate the NN model in three steps. In the first step, we filter noise in the data set using a moving average filter. In the second step, we use a DFM to extract two common factors from a large-scale data set. In the third step, we feed these factors into NN models to obtain the current state of the US business cycle.

First, we evaluate the fit of models over the whole data sample by ignoring historical data availability and perform robustness checks. In that exercise, we show that the NN model follows the NBER's business cycle chronology quite accurately in-sample and out-of-sample, and results indicate that noise reduction is important in obtaining smooth and accurate prediction probabilities.

We then assess the turning-point identification performance of NN models between 1990:01 and 2016:12 by taking account of historical data availability and compare them to LVQ and DFMS models. We document that NN models determine turning points quite accurately and very quickly in real time. Results also show that NN models identify turning points much faster than the competing models. Furthermore, NN models identify peaks much faster than troughs. Finally, we analyze NN models' business cycle state identification performance during the early 1980s recessions to have a complete performance analysis of NN models. Even though NN models still manage to identify business cycle turning points very quickly during this period, the identification accuracy of NN models is slightly off during early 1980s. Given that most dating methodologies identify turning points with a significant delay, NN models can be used to obtain timely information on the current state of the business cycle in real time.

ORCID

Barış Soybilgen https://orcid.org/0000-0003-4906-1878

REFERENCES

Bai, J., & Ng, S. (2002). Determining the number of factors in approximate factor models. *Econometrica*, *70*(1), 191–221.

Bai, J., & Ng, S. (2007). Determining the number of primitive shocks in factor models. *Journal of Business and Economic Statistics*, *25*(1), 52–60.

Bańbura, M., Giannone, D., Modugno, M., & Reichlin, L. (2013). Now-Casting and the Real-Time Data Flow. In G. Elliott, & A. Timmermann (Eds.), *Handbook of Economic Forecasting*, Vol. *2*. Amsterdam, Netherlands: North-Holland, pp. 195–237.

Bańbura, M., & Rünstler, G. (2011). A look into the factor model black box: Publication lags and the role of hard and soft data in forecasting GDP. *International Journal of Forecasting*, *27*(2), 333–346.

Bellégo, C., & Ferrara, L. (2009). Forecasting Euro Area Recessions Using Time-Varying Binary Response Models for Financial Variables. (*Working Paper No.259*). Paris, France: Banque de France.

Chauvet, M., & Piger, J. (2008). A comparison of the real-time performance of business cycle dating methods. *Journal of Business and Economic Statistics*, *26*(1), 42–49.

Chen, Z., Iqbal, A., & Lai, H. (2011). Forecasting the probability of US recessions: A probit and dynamic factor modelling approach. *Canadian Journal of Economics*, *44*(2), 651–672.

Christiansen, C., Eriksen, J. N., & Møller, S. V. (2014). Forecasting US recessions: The role of sentiment. *Journal of Banking and Finance*, *49*, 459–468.

Demuth, H. B., Beale, M. H., De Jess, O., & Hagan, M. T. (2014). *Neural network design*. Stillwater, OK: Martin Hagan.

Doz, C., Giannone, D., & Reichlin, L. (2011). A two-step estimator for large approximate dynamic factor models based on Kalman filtering. *Journal of Econometrics*, *164*(1), 188–205.

Eickmeier, S., & Ziegler, C. (2008). How successful are dynamic factor models at forecasting output and inflation? A meta-analytic approach. *Journal of Forecasting*, *27*(3), 237–265.

Fossati, S. (2015). Forecasting US recessions with macro factors. *Applied Economics*, *47*(53), 5726–5738.

Fossati, S. (2016). Dating US business cycles with macro factors. *Studies in Nonlinear Dynamics and Econometrics*, *20*(5), 529–547.

Giannone, D., Reichlin, L., & Sala, L. (2005). *Monetary Policy in Real Time*, pp. 161–200.

Giannone, D., Reichlin, L., & Small, D. (2008). Nowcasting: The real-time informational content of macroeconomic data. *Journal of Monetary Economics*, *55*(4), 665–676.

Giusto, A., & Piger, J. (2017). Identifying business cycle turning points in real time with vector quantization. *International Journal of Forecasting*, *33*(1), 174–184.

Hamilton, J. D. (2011). Calling recessions in real time. *International Journal of Forecasting*, *27*(4), 1006–1026.

McCracken, M. W., & Ng, S. (2016). FRED-MD: A monthly database for macroeconomic research. *Journal of Business and Economic Statistics*, *34*(4), 574–589.

Møller, M. F. (1993). A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, *6*(4), 525–533.

Piger, J. (2018). History of real time recessions. Retrieved from http://pages.uoregon.edu/jpiger/us_recession_probs.htm/history_of_real_time_recess.xls

Qi, M. (2001). Predicting US recessions with leading indicators via neural network models. *International Journal of Forecasting*, *17*(3), 383–401.

Sargent, T. J., & Sims, C. A. (1977). Business Cycle Modeling without Pretending to have Too Much a Priori Economic Theory. (*Working Paper No.55*). Minneapolis, MN: Federal Reserve Bank of Minneapolis.

AUTHOR BIOGRAPHY

Barış Soybilgen is an assistant professor at Istanbul Bilgi University and assistant director of Istanbul Bilgi University's Center for Financial Studies

How to cite this article: Soybilgen B. Identifying US Business Cycle Regimes Using Dynamic Factors and Neural Network Models. *Journal of Forecasting*. 2020;39:827–840. https://doi.org/10.1002/for.2658

APPENDIX : DESCRIPTION OF THE DATA SET

TABLE A1

Code	Description	Transformation	Period
RPI	Real personal income	1	All
W875RX1	Real personal income excluding current transfer receipts	1	All
DPCERA3M086SBEA	Real personal consumption expenditures (chain-type quantity index)	1	2003:12–
CMRMTSPLx	Real manufacturing and trade industries sales	1	All
RETAILx	Real retail and food services sales	1	All
INDPRO	Industrial production index	1	All
IPFPNSS	Industrial production: Final products and nonindustrial supplies	1	All
IPFINAL	Industrial production: Final products (market group)	1	All
IPCONGD	Industrial production: Consumer goods	1	All
IPDCONGD	Industrial production: Durable consumer goods	1	2002:11–
IPNCONGD	Industrial production: Nondurable consumer goods	1	2002:11–
IPBUSEQ	Industrial production: Business equipment	1	2002:11–
IPMAT	Industrial production: Materials	1	All
IPDMAT	Industrial production: Durable materials	1	2002:11–
IPNMAT	Industrial production: Nondurable materials	1	2002:11–
IPMANSICS	Industrial production: Manufacturing (SIC)	1	All
IPB51222S	Industrial production: Residential utilities	1	2002:11–
IPFUELS	Industrial production: Fuels	1	2002:11–
CUMFNS	Capacity utilization: Manufacturing (SIC)	2	All
CLF16OV	Civilian labor force	1	All
CE16OV	Civilian employment level	1	All
UNRATE	Civilian unemployment rate	2	All
UEMPMEAN	Average (mean) duration of unemployment	2	All
UEMPLT5	Number of civilians unemployed for less than 5 weeks	1	All
UEMP5TO14	Number of civilians unemployed for 5–14 Weeks	1	All
UEMP15OV	Number of civilians unemployed for 15 weeks and over	1	All
UEMP15T26	Number of civilians unemployed for 15–26 Weeks	1	All
UEMP27OV	Number of civilians unemployed for 27 weeks and over	1	All
CLAIMSx	Initial claims	1	All
PAYEMS	All employees: Total nonfarm payrolls	1	All
USGOOD	All employees: Goods-producing industries	1	All
CES1021000001	All employees: Mining and logging: Mining	1	All
USCONS	All employees: Construction	1	All
MANEMP	All employees: Manufacturing	1	All
DMANEMP	All employees: Durable goods	1	All
NDMANEMP	All employees: Nondurable goods	1	All
SRVPRD	All employees: Service-providing industries	1	All
USTPU	All employees: Trade. Transportation and utilities	1	2003:05–
USWTRADE	All employees: Wholesale trade	1	All
USTRADE	All employees: Retail trade	1	All
USFIRE	All employees: Financial activities	1	All
USGOVT	All employees: Government	1	All
CES0600000007	Average weekly hours of production and nonsupervisory employees: Goods-producing	2	All
AWOTMAN	Average weekly overtime hours of production and nonsupervisory employees: Manufacturing	2	All
AWHMAN	Average weekly hours of production and nonsupervisory employees: Manufacturing		
HOUST	Housing starts: Total: New privately owned housing units started	1	All
HOUSTNE	Housing starts in northeast census region	1	All
HOUSTMW	Housing starts in midwest census region	1	All
HOUSTS	Housing starts in south census region	1	All
HOUSTW	Housing starts in west census region	1	All
PERMIT	New private housing units authorized by building permits	1	All

Continues

TABLE A1 Continued

Code	Description	Transformation	Period
PERMITNE	New private housing units authorized by building permits in the northeast census region	1	All
PERMITMW	New private housing units authorized by building permits in the midwest census region	1	All
PERMITNE	New private housing units authorized by building permits in the northeast census region	1	All
PERMITMW	New private housing units authorized by building permits in the midwest census region	1	All
PERMITS	New private housing units authorized by building permits in the south census region	1	All
PERMITW	New private housing units authorized by building permits in the west census region	1	All
AMDMNOx	Manufacturers' new orders: Durable goods	1	All
ANDENOx	New orders for nondefense capital goods	1	All
AMDMUOx	Value of manufacturers' unfilled orders for durable goods industries	1	All
BUSINVx	Total business inventories	1	All
ISRATIOx	Total business: Inventories to sales ratio	2	All
M1SL	M1 Money stock	1	All
M2SL	M2 Money stock	1	All
M2REAL	Real M2 money stock	1	All
AMBSL	St. Louis adjusted monetary base	1	All
TOTRESNS	Total reserves of depository institutions	1	All
NONBORRES	Reserves of depository institutions. Nonborrowed	1	All
BUSLOANS	Commercial and industrial loans. All commercial banks	1	All
REALLN	Real estate loans. All commercial banks	1	All
NONREVSL	Total nonrevolving credit owned and securitized. Outstanding	1	All
CONSPI	Nonrevolving consumer credit to personal income	1	All
S&P 500	S&P 500	1	All
S&P: indust.	S&P 500 industries	1	All
S&P div. yield	S&P dividend yield	1	All
S&P PE ratio	S&P PE ratio	1	All
FEDFUNDS	Effective federal funds rate	2	All
CP3Mx	3-month AA financial commercial paper rate	2	All
TB3MS	3-month Treasury bill: Secondary market rate	2	All
TB6MS	6-month Treasury bill: Secondary market rate	2	All
GS1	1-year Treasury constant maturity rate	2	All
GS5	5-year Treasury constant maturity rate	2	All
GS10	10-year Treasury constant maturity rate	2	All
AAA	Moody's seasoned Aaa corporate bond yield	2	All
BAA	Moody's seasoned Baa corporate bond yield	2	All
COMPAPFFx	3-month commercial paper minus federal funds rate	0	All
TB3SMFFM	3-month Treasury bill minus federal funds rate	0	All
TB6SMFFM	6-month Treasury bill minus federal funds rate	0	All

TABLE A2

Code	Description	Transformation	Period
T1YFFM	1-year Treasury constant maturity minus federal funds rate	0	All
T5YFFM	5-year Treasury constant maturity minus federal funds rate	0	All
T10YFFM	10-year Treasury constant maturity minus federal funds rate	0	All
AAAFFM	Moody's seasoned Aaa corporate bond minus federal funds rate	0	All
BAAFFM	Moody's seasoned Baa corporate bond minus federal funds rate	0	All
EXSZUSx	Switzerland/US foreign exchange rate	1	All
EXJPUSx	Japan/US foreign exchange rate	1	All
EXUSUKx	US/UK foreign exchange rate	1	All
EXCAUSx	Canada/US foreign exchange rate	1	All
PPIFGS	Producer price index by commodity for finished goods	3	–2016:01
PPIFCG	Producer price index by commodity for finished consumer goods	3	–2016:01
PPIITM	Producer price index by commodity intermediate materials: Supplies and components	3	–2016:01
PPICRM	Producer price index by commodity for crude materials for further processing	3	–2016:01
OILPRICEx	Crude oil prices: West Texas intermediate (WTI)—Cushing. Oklahoma	3	All
PPICMM	Producer price index by commodity metal s and metal products: Primary nonferrous metals	3	All
CPIAUCSL	Consumer price index for all urban consumers: All items	3	All
CPIAPPSL	Consumer price index for all urban consumers: Apparel	3	All
CPITRNSL	Consumer price index for all urban consumers: Transportation	3	All
CPIMEDSL	Consumer price index for all urban consumers: Medical care	3	All
CUSR0000SAC	Consumer price index for all urban consumers: Commodities	3	All
CUSR0000SAD	Consumer price index for all urban consumers: Durables	3	–2014:11
CUSR0000SAS	Consumer price index for all urban consumers: Services	3	All
CPIULFSL	Consumer price index for all urban consumers: All items less food	3	All
CUUR0000SA0L2	Consumer price index for all urban consumers: All items less shelter	3	All
CUSR0000SA0L5	Consumer price index for all urban consumers: All items less medical care	3	All
PCEPI	Personal consumption expenditures: Chain-type price index	3	2000:07–
DDURRG3M086SBEA	Personal consumption expenditures: Durable goods (chain-type price index)	3	2000:07–
DNDGRG3M086SBEA	Personal consumption expenditures: Nondurable goods (chain-type price index)	3	2000:07–
DSERRG3M086SBEA	Personal consumption expenditures: Services (chain-type price index)	3	2000:07–
CES0600000008	Average hourly earnings of production and nonsupervisory employees: Goods-producing	3	All
CES2000000008	Average hourly earnings of production and nonsupervisory employees: Construction	3	All
CES3000000008	Average hourly earnings of production and nonsupervisory employees: Manufacturing	3	All
MZMSL	MZM Money stock	1	All
DTCOLNVHFNM	Consumer motor vehicle loans owned by finance companies. Outstanding	1	All
DTCTHFNM	Total consumer loans and leases owned and securitized by finance companies. Outstanding	1	All
INVEST	Securities in bank credit at all commercial banks	1	All
VXOCLSx	CBOE S&P 100 volatility index: VXO	1	All[a]
WPSFD49207	Producer price index by commodity for finished goods	3	2016:02–
WPSFD49502	Producer price index by commodity for finished consumer goods	3	2016:02–
WPSID61	Producer price index by commodity intermediate materials: Supplies and components	3	2016:02–
WPSID62	Producer price index by commodity for crude materials for further processing	3	2016:02–
CUUR0000SAD	Consumer price index for all urban consumers: Durables	3	2014:12–

Note. The column "Code" shows the code of the variable in the FRED-MD database. The column "Transformation" denotes the following data transformation for a series: (0) no transformation; (1) monthly growth rate; (2) monthly differences; (3) monthly differences of the yearly growth rate.
Except the period between 2004:12 and 2005:07.