A Macroeconomic Nowcasting Model

March 2021

1 Motivation

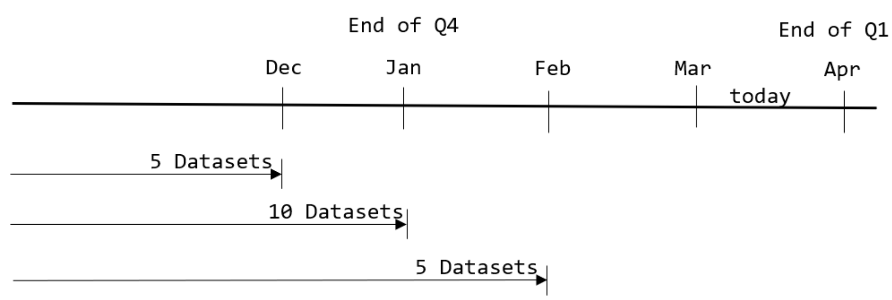
Nowcasting is the prediction of the present, the near future, and the near past. Nowcasting is important in economics because many important macroeconomic statistics are released with a lengthy delay. For example, the Bureau of Economic Analysis releases quarterly GDP typically two months after the quarter has already ended - a significant delay for any companies or individuals who need the data for planning and forecast models.

This delay is particularly salient during times of high volatility. During the first few months of the COVID-19 pandemic in the United States, many companies attempted to use high-frequency indicators to attempt nowcasts of the state of the macroeconomy. For example, JP Morgan forecasts in March 13 predicted Q2 GDP growth of -3%; the estimate was revised down to -14% by March 21, -25% by March 25, and -40% by April 10. Yet many of the models were ad-hoc and only able to use a small number of predictive indicators, such as jobless claims or traffic data.

Nowcasting is about deciphering key information about the state of the economy before official data is released. Because of the fundamentally urgent nature of nowcasting, it is important that nowcast models exploit any latest, high-frequency data available. Nowcasts should be able to generate constantly rolling forecasts, updating these numbers in response to any new data releases.

For example, suppose the date is early March, and the variable we want to predict is Q1 GDP. The simplest way to predict Q1 GDP would be to use historical quarterly data from various economic variables. But this data would only be up to Q4 of last year, and would fail to capture the critically important predictive power that could be provided by monthly and daily data released throughout January and February.

Suppose instead, we used monthly data as our predictors of Q1 GDP. Again, we will soon run in to a problem. As an example, imagine that we had imported 20 monthly data series. Suppose 5 of these series ended in December, 10 ended in January, and 5 ended in March.



Traditional modeling methods would require us to either throw out variables or throw out months - for example, we could truncate all our data series at January and lose out the information provided by the 5 February data points. Alternatively, we could completely remove the 15 variables with data releasing before March. Both methods are unappealing.

In this paper, we will utilize a methodology that will allow us to use the information from all variables at any dates. This model will give an updated forecast in response to any new data releases. Additionally, the model can be generalized to nowcast any time series variable, not just GDP. The methodology for the model will be described in the next section.

2 Methodology

2.1 Data

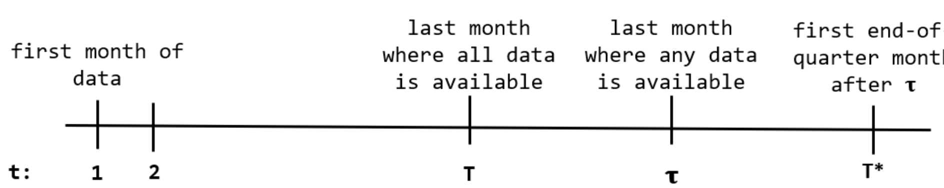
We begin by importing monthly data from the St. Louis Federal Reserve Database (FRED). Let N refer to the number of different monthly variables.

2.2 Time Periods

The imported monthly data will have ragged edges - i.e., some monthly data will be available for later months than others.

We will let T denote the number of dates for which data is available for all data series. τ will denote the number of dates for which data is available for at least one data series. $T*$ will denote number of dates up to the end-of-quarter month of the τ date. For example, suppose date τ occurs on February. The end-of-quarter month, T^*, will be March (since Q1 runs through the end of March).

In other words, data will be indexed by $t = 1, 2, \ldots, T, T+1, \ldots, \tau, \ldots, T^*$, where dates $T+1$ through τ are the dates for which only some data are available, and dates $\tau + 1$ through T^* are the dates for which no data is available up to the next quarter-ending month.



2.3 Factor VAR

We model the factors as following a vector-autoregressive (VAR) process, i.e.,

$$\underbrace{\begin{bmatrix} \mathbf{f1}_t \\ \mathbf{f2}_t \\ \vdots \\ \mathbf{fR}_t \end{bmatrix}}_{z_t} = B \underbrace{\begin{bmatrix} \mathbf{f1}_{t-1} \\ \mathbf{f2}_{t-1} \\ \vdots \\ \mathbf{fR}_{t-1} \end{bmatrix}}_{z_{t-1}} + C + \underbrace{\begin{bmatrix} v1_t \\ v2_t \\ \vdots \\ vR_t \end{bmatrix}}_{v_t},$$

$$\begin{aligned} \text{where } z_t \text{ is the } R \times 1 \text{ matrix of time } t \text{ factors,} \\ B \text{ is the } R \times R \text{ coefficient matrix,} \\ C \text{ is the } R \times 1 \text{ constant matrix,} \\ \text{and } v_t \text{ is the } R \times 1 \text{ matrix of errors for time } t. \end{aligned}$$

We wish to estimate the coefficient matrices B and C. This can be done via OLS estimation. We first rewrite the data as the standard linear equation,

$$\underbrace{\begin{bmatrix} \mathbf{f1}_2 & \mathbf{f2}_2 & \dots & \mathbf{f6}_2 \\ \mathbf{f1}_3 & \mathbf{f2}_3 & \dots & \mathbf{f6}_3 \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{f1}_T & \mathbf{f2}_T & \dots & \mathbf{f6}_T \end{bmatrix}}_{\Gamma} = \underbrace{\begin{bmatrix} 1 & \mathbf{f1}_1 & \mathbf{f2}_1 & \dots & \mathbf{f6}_1 \\ 1 & \mathbf{f1}_2 & \mathbf{f2}_2 & \dots & \mathbf{f6}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \mathbf{f1}_{T-1} & \mathbf{f2}_{T-1} & \dots & \mathbf{f6}_{T-1} \end{bmatrix}}_{\Psi} \underbrace{\begin{bmatrix} C' \\ B' \end{bmatrix}}_{\Lambda} + \underbrace{\begin{bmatrix} v1_2 & v2_2 & \dots & vR_2 \\ v1_3 & v2_3 & \dots & vR_3 \\ \vdots & & & \\ v1_T & v2_T & \dots & vR_T \end{bmatrix}}_{V},$$

$$\begin{aligned} \text{where } \Gamma \text{ is the } T-1 \times R \text{ dependent data matrix,} \\ \Psi \text{ is the } T-1 \times R+1 \text{ independent data matrix,} \\ \Lambda \text{ is the } R+1 \times R \text{ matrix of coefficient weightings,} \\ \text{and } V \text{ is the } T-1 \times R \text{ matrix of residuals.} \end{aligned}$$

The coefficient matrix Λ can be estimated by the standard OLS estimator.

$$\widehat{\Lambda} = (\Psi'\Psi)^{-1}(\Psi'\Gamma)$$

It can then be partitioned to calculate $\widehat{B}'$ and $\widehat{C}'$, which can then be transposed to derive our estimates of the original coefficient matrices B and C, $\widehat{B}$ and $\widehat{C}$.

2.4 Dynamic Factor Models

Now let us consider again the monthly covariates which were include in the principal components analysis. We will model these as dynamic factor models (DFMs), i.e. - they are regressed on the factor variables derived from earlier.

The factor models take the following form:

$$\underbrace{\begin{bmatrix} \mathbf{ue}_t \\ \mathbf{pce}_t \\ \vdots \\ \mathbf{claims}_t \end{bmatrix}}_{y_t} = A \underbrace{\begin{bmatrix} \mathbf{f1}_t \\ \mathbf{f2}_t \\ \vdots \\ \mathbf{fR}_t \end{bmatrix}}_{z_t} + \underbrace{\begin{bmatrix} w1_t \\ w2_t \\ \vdots \\ wN_t \end{bmatrix}}_{w_t},$$

$$\begin{aligned} \text{where } y_t \text{ is the } N \times 1 \text{ vector of monthly variables at time } t, \\ A \text{ is the } N \times R \text{ coefficient matrix,} \\ z_t \text{ is the } R \times 1 \text{ vector of factors at time } t, \\ \text{and } w_t \text{ is the } N \times 1 \text{ vector of errors at time } t. \end{aligned}$$

We wish to estimate the coefficient matrix A. As before, we can do this by estimating this as an OLS equation, writing the data matrices as follows

$$\underbrace{\begin{bmatrix} \mathbf{ue}_2 & \mathbf{pce}_2 & \dots & \mathbf{claims}_2 \\ \mathbf{ue}_3 & \mathbf{pce}_3 & \dots & \mathbf{claims}_3 \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{ue}_T & \mathbf{pce}_T & \dots & \mathbf{claims}_T \end{bmatrix}}_{\Phi} = \underbrace{\begin{bmatrix} \mathbf{f1}_2 & \mathbf{f2}_2 & \dots & \mathbf{f6}_2 \\ \mathbf{f1}_3 & \mathbf{f2}_3 & \dots & \mathbf{f6}_3 \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{f1}_T & \mathbf{f2}_T & \dots & \mathbf{f6}_T \end{bmatrix}}_{\Omega} A' + \underbrace{\begin{bmatrix} w1_2 & w2_2 & \dots & wR_2 \\ w1_3 & w2_3 & \dots & wR_3 \\ \vdots & & & \\ w1_T & w2_T & \dots & wR_T \end{bmatrix}}_{W} \tag{1}$$

As before we can estimate A with the standard OLS estimator.

$$\widehat{A}' = (\Omega'\Omega)^{-1}(\Omega'\Phi)$$

2.5 State-Space Setup

Now, combining our equations for the DFM and the VAR, we have the below system.

$$\begin{aligned} z_t &= Bz_{t-1} + Cx + v_t \\ y_t &= Az_t + w_t \end{aligned}$$

This system is now fully specified and in state-space form. The first equation is our state (or transition) equation. The second equation is our measurement equation.

We use our estimated values B, C, and A calculated in our previous two sections. To run the Kalman Filter, we will want to create the actual data matrices for z_t and y_t. z_t can be constructed as before, using data for factors from time 1 through T. However, unlike in the previous two sections, we will want to create y_t matrices not for just time periods 1 through T, but now for time periods 1 through τ. Elements in y_t may be set to any value for missing observations; the process of Kalmam filtration will render this choice irrelevant.

Specifically, we construct the matrices below.

$$z_t = \begin{bmatrix} \mathbf{f1}_t \\ \mathbf{f2}_t \\ \vdots \\ \mathbf{fR}_t \end{bmatrix}, \forall t \in 1, \dots, T$$

$$y_t = \begin{bmatrix} \mathbf{ue}_t \text{ if available, otherwise } 0 \\ \mathbf{pce}_t \text{ if available, otherwise } 0 \\ \vdots \\ \mathbf{claims}_t \text{ if available, otherwise } 0 \end{bmatrix}, \forall t \in 1, \dots, \tau$$

For Kalman filtration, we also require an assumed distribution on v_t and w_t. We assume that v_t is distributed normally with mean 0 and constant diagonal covariance matrix denoted Q, with diagonal entries calculated by taking the average squared values of the residuals of the VAR.

We also assume w_t is distributed normally with mean 0. However, we no longer specify the covariance matrix as constant, but as the time-dependent matrices R_t. For $t \in 1, \dots, T+1$, we let R_t be a diagonal covariance matrix with diagonal entries calcualted by taking the average squared values of the residuals of the DFM. For $t \in T+1, \dots, \tau$, we let the diagonal elements of R_t be equal to infinity if the corresponding element of y_t is missing for that time period; and equal to the average squared value of the residual if otherwise.

$$\begin{aligned} v_t &\sim \mathcal{N}(0, Q) \\ w_t &\sim \mathcal{N}(0, R_t) \end{aligned}$$

We begin with the unconditional mean of $\widehat{Z}_{0|-1} = 0$ and unconditional variance of VAR $\Sigma_{0|-1} = 0$.

3 Kalman Filtration

Now that our state-space model is fully specified, we can begin the Kalman filter recursions.

$$\begin{aligned} z_t &= Bz_{t-1} + Cx + v_t \\ y_t &= Az_t + w_t \\ v_t &\sim \mathcal{N}(0, Q) \\ w_t &\sim \mathcal{N}(0, R_t) \end{aligned}$$

To solve this programmatically, we will need the previously estimated matrices A, B, and C; the matrices z_t from 1 through T; the matrices y_t from 1 through τ; the covariance matrix Q; and finally, the covariance matrices R_t from 1 through τ.

We initialize the Kalman filter with the following standard assumptions.

$$\begin{aligned} \mathbf{z}_{0|0} &= 0 \\ \mathbf{CovZ} &= 0 \end{aligned}$$

Now for $t = 1, \ldots, \tau$, we iterate through the Kalman filter recursions and iteratively calculate the values below.

$$\begin{aligned}
\mathbf{z}_{t|t-1} &= B\mathbf{z}_{t-1|t-1} + C \\
\mathbf{CovZ}_{t|t-1} &= B\mathbf{CovZ}_{t-1|t-1} + Q \\
\mathbf{y}_{t|t-1} &= A\mathbf{z}_{t|t-1} \\
\mathbf{CovY}_{t|t-1} &= A\mathbf{CovZ}_{t|t-1}A' + R_t \\
P_t &= \mathbf{CovZ}_{t|t-1}A'\mathbf{CovY}^{-1}_{t|t-1} \\
\mathbf{z}_{t|t} &= \mathbf{z}_{t|t-1} + P_t(\mathbf{y}_t - \mathbf{y}_{t|t-1}) \\
\mathbf{CovZ}_{t|t} &= \mathbf{CovZ}_{t|t-1} - P_t(\mathbf{CovY}_{t|t-1})P'_t
\end{aligned}$$

Note that the during recursions $T+1, \ldots \tau$, the infinite values in the R_t matrix will cause infinite values in the $\mathbf{CovY}_{t|t-1}$ matrix. This may prevent standard computational methods from computing the inverse of the matrix needed in the step for calculation of $\mathbf{CovZ}_{t|t}$. Alternative methods, such as a Cholesky decomposition before inversion, can be used to subvert this problem.

The Kalman filter allows us to recover all the time t conditional state matrices $z_{t|t}$ that have been adjusted for information from the monthly datasets. However, of more interest to us is the value of the state matrices when conditioned on all data available at time τ, $z_{t|\tau}$. This can be recovered by using the Kalman smoother.

Recursively iterating over $t = \tau - 1, \ldots, 1$, we calculate the following values.

$$\begin{aligned}
S_t &= \mathbf{CovZ}_{t|t}B'\mathbf{CovZ}^{-1}_{t+1|t} \\
\mathbf{z}_{t|\tau} &= \mathbf{z}_{t|t} + S_t(\mathbf{z}_{t+1|\tau} - \mathbf{z}_{t+1|t}) \\
\mathbf{CovZ}_{t|\tau} &= \mathbf{CovZ}_{t|t} - S_t(\mathbf{CovZ}_{t+1|t} - \mathbf{CovZ}_{t+1|\tau})S'_t
\end{aligned}$$

These values $\mathbf{z}_{t|\tau}$ will serve as our estimates of the state variables (i.e., the PCA factors) from time 1 through τ.

Finally, we want to forecast the the state vector $z_{t|\tau}$ for $t = \tau + 1, \ldots, T^*$. This can be done through the typical Kalman filter forecasting step.

Recursively iterating over $t = \tau + 1, \ldots, T^*$, we calculate the following values.

$$\begin{aligned}
\mathbf{z}_{t|\tau} &= B\mathbf{z}_{t-1|\tau} + C \\
\mathbf{CovZ}_{t|\tau} &= B\mathbf{CovZ}_{t-1|\tau}B' + Q \\
\mathbf{y}_{t|\tau} &= A\mathbf{z}_{t|\tau} \\
\mathbf{CovY}_{t|\tau} &= A\mathbf{CovZ}_{t|\tau}A' + R_0
\end{aligned}$$

Combining the calculations for $\mathbf{z}_{t|\tau}$ with the ones derived from the Kalman smoother, we will now be able to obtain the full time series for the factors from time 1 through time T^*.