Some Notes on the Kalman Filter

Ed Herbst

March 14, 2025

State space models form a very general class of models that encompass many of the specifications that we encountered earlier. VARMA models, linearized DSGE models, and more can be written in state space form. State space models are particularly popular at the FRB. For example, the models in the r^* suite can all be written in state space form. (setq line-spacing 0) A state space model can be described by two different equations: a measurement equation that relates an *unobservable* state vector s_t to the *observables* y_t, and a transition equation that describes the evolution of the state vector s_t. For now, we'll restrict attention to the case in which both of these equations are linear.

Measurement. The measurement equation is of the form

$$y_t = D_{t|t-1} + Z_{t|t-1} s_t + \eta_t, \quad t = 1, \dots, T \tag{1}$$

where y_t is an $n_y \times 1$ vector of observables, s_t is an $n_s \times 1$ vector of state variables, $Z_{t|t-1}$ is an $n_y \times n_s$ vector, $D_{t|t-1}$ is a $n_y \times 1$ vector, and η_t are innovations (or often "measurement errors") with mean zero and $\mathbb{E}_{t-1}[\eta_t \eta_t'] = H_{t|t-1}$.

- The matrices $Z_{t|t-1}$, $D_{t|t-1}$, and $H_{t|t-1}$ are in many applications constant ("time-invariant.")
- However, it is sufficient that they are predetermined at $t-1$. They could be functions of $y_{t-1}, y_{t-2}, \dots$.
- To simplify the notation, we will denote them by Z_t, D_t, and H_t, respectively.

Transition. The transition equation is of the form

$$s_t = C_{t|t-1} + T_{t|t-1} s_{t-1} + R_{t|t-1} \epsilon_t \tag{2}$$

where R_t is $n_s \times n_\epsilon$, and ϵ is a $n_\epsilon \times 1$ vector of innovations with mean zero and variance $\mathbb{E}_{t|t-1}[\epsilon_t \epsilon_t'] = Q_{t|t-1}$.

- The assumption that s_t evolves according to an VAR(1) process is not very restrictive, since it could be the companion form to a higher order VAR process.
- It is furthermore assumed that (i) expectation and variance of the initial state vector are given by $\mathbb{E}[s_0] = A_0$ and $\mathbb{V}[s_0] = P_0$;
- η_t and ϵ_t are uncorrelated with each other in all time periods , and uncorrelated with the initial state. This assumption is not really necessary, but it simplies things considerable.

The collection of matrices in () and () define the state space system. For that reason, they are often referred to as the "system matrices."
Example. Consider the ARMA(1,1) model of the form

$$y_t = \phi y_{t-1} + \epsilon_t + \theta\epsilon_{t-1} \quad \epsilon_t \sim iid\mathcal{N}(0, \sigma^2) \tag{3}$$

The model can be rewritten in state space form

$$\begin{aligned} y_t &= [1\ \theta]\begin{bmatrix} \epsilon_t \\ \epsilon_{t-1} \end{bmatrix} + \phi y_{t-1} && (4) \\ \begin{bmatrix} \epsilon_t \\ \epsilon_{t-1} \end{bmatrix} &= \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} \epsilon_{t-1} \\ \epsilon_{t-2} \end{bmatrix} + \begin{bmatrix} \eta_t \\ 0 \end{bmatrix} && (5) \end{aligned}$$

where $\epsilon_t \sim iid\mathcal{N}(0, \sigma^2)$. Thus, the state vector is composed of $s_t = [\epsilon_t, \epsilon_{t-1}]'$ and $D_t = \rho y_{t-1}$. This construction is not unique. We could also write the model as:

$$\begin{aligned} y_t &= [1\ 0]\begin{bmatrix} y_t \\ \epsilon_t \end{bmatrix} && (6) \\ \begin{bmatrix} y_t \\ \epsilon_t \end{bmatrix} &= \begin{bmatrix} \phi & \theta \\ 0 & 0 \end{bmatrix}\begin{bmatrix} y_{t-1} \\ \epsilon_{t-1} \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \end{bmatrix}\epsilon_t. && (7) \end{aligned}$$

Notice in this formulation the state vector $s_t = [y_t, \epsilon_t]$ is partially observed. So it's not true, strictly speaking, that the entire s_t vector must be unobserved. $\square$.

If the system matrices $D_t, Z_t, H_t, C_t, T_t, R_t, Q_t$ are non-stochastic and predetermined, then the system is linear and y_t can be expressed as a function of present and past η_t's and ϵ_t's. We've done some work on linear systems previously (VARs), so the natural next step is to expand our toolkit to do the kinds of things we liked to do with VARs:

- Calculate predictions $y_t|Y^{t-1}$, where $Y^{t-1} = [y_{t-1}, \dots, y_1]$,
- Obtain a likelihood function

$$p(Y^T|\{Z_t, D_t, H_t, T_t, C_t, R_t, Q_t\}), \tag{8}$$

- and, *something we didn't do with VARs*, back out a sequence

$$\left\{p(s_t|Y^t, \{Z_t, D_t, H_t, T_t, C_t, R_t, Q_t\})\right\}_{t=1}^T.$$

Now, if the state vector was observed, it would be easy to combine equation () and () to obtain a VAR jointly in $[y_t, s_t]$. Thus, it would be straightforward to obtain the (perhaps conditional) likelihood:

$$p(Y^T, S^T|\{Z_t, D_t, H_t, T_t, C_t, R_t, Q_t\}).$$

But life is hard, and we don't get to observe S^T. We need to compute the likelihood for the data we have, i.e., the likelihood in (). We have to marginalize out S^T. It turns out that there is an algorithm that does this, and fulfills the three desiderata above. The algorithm is called the *Kalman Filter* and was originally adopted from the engineering literature.

The Kalman Filter

For this presentation of the Kalman filter, we're going to assume that the system matrices are time invariant, that is, they do not depend on t. So we drop these subscripts from our notation. Furthermore, we're going to collect them in the vector $\theta = [C, T, R, Q, D, Z, H]$, where the *vec* operator is being implicitly applied to each matrix.

We're also going to assume that the innovations η_t and ϵ_t are normally distributed. We need to this to obtain an exact likelihood, although the Kalman filter can be used to obtain an optimal—in terms of MSE—predictor y_{t+h} given Y^T for $h \geq 1$ using linear projections, regardless of the parametric distributions for η_t and ϵ_t. The chapter on state space models in cite:Hamilton derives this. In this case the likelihood calculation delivers a quasi-likelihood.

With our normality assumption, the derivation of the Kalman filter has a natural Bayesian interpretation. Before we proceed, we're going to state some results about multivariate normal distributions, which will help later on.

Lemma. Let $(x', y')'$ be jointly normal with

$$\mu = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix} \quad \text{and} \quad \Sigma = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}$$

Then the *pdf* $(x|y)$ is multivariate normal with

$$\begin{aligned} \mu_{x|y} &= \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y) \qquad (9) \\ \Sigma_{xx|y} &= \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx} \qquad (10) \end{aligned}$$

Note that the converse is not necessarily true. □

In both theory and practice, the Kalman filter proceeds recursively, using the natural prior-posterior sequencing, after an initialization.

Initialization. We're going to start at period $t = 0$, that is, the period before we first observe y. We assume that s_0 is normally distributed:

$$s_0|\theta \sim \mathcal{N}(A_0, P_0). \qquad (11)$$

Importantly, we conceptualize this distribution as prior distribution. We'll discuss possible ways to select A_0 and P_0 in a bit.

Prediction. We can combine our prior distribution for s_0 with the state transition equation (). Since s_0 is normally distributed and ϵ_1 is also normally distributed (and independent of s_0), s_1 is also normally distributed,

$$s_1|\theta \sim \mathcal{N}\left(A_{1|0}, P_{1|0}\right)$$

where

$$A_{1|0} = C + TA_0 \text{ and } P_{1|0} = TP_0T' + RQR'.$$

Note that this is the unconditional distribution of s_1, a prior distribution for s_1 before seeing y_1. We write the mean $A_{1|0}$ and $P_{1|0}$ as conditional on time $t = 0$.

Next consider the prediction of y_1. The conditional distribution of y_1 is of the form

$$y_1 | s_1, \theta \sim \mathcal{N}(D + Z s_1, H) \tag{12}$$

Since $s_1 \sim \mathcal{N}(A_{1|0}, P_{1|0})$, we can deduce that the marginal distribution of y_1 is of the form

$$y_1 | \theta \sim \mathcal{N}(\hat{y}_{1|0}, F_{1|0}) \tag{13}$$

where

$$\hat{y}_{1|0} = D + Z A_{1|0} \text{ and } F_{1|0} = Z P_{1|0} Z' + H.$$

Here we've been explicit in going $s_0 \to s_1 \to y_1$.

Updating. Another way to see this is to rewrite the observation equation () in terms of s_{t-1} and ϵ_t. If s_0 is normally distributed as above it's easy to see that s_1 and y_1 are jointly normally distributed with the marginal and conditional distributions mentioned above. We have:

$$s_1 = C + T s_0 + R\epsilon_t \tag{14}$$
$$y_1 = D + ZT s_0 + Z\epsilon_t + \eta_t. \tag{15}$$

Direct calculation yields:

$$\begin{bmatrix} s_1 \\ y_1 \end{bmatrix} \Bigg| \theta \sim \mathcal{N}\left(\begin{bmatrix} A_{1|0} \\ \hat{y}_{1|0} \end{bmatrix}, \begin{bmatrix} P_{1|0} & P_{1|0} Z' \\ Z P_{1|0} & F_{1|0} \end{bmatrix} \right). \tag{16}$$

Consider the third goal of toolbox: delivering $p(s_1|y_1, \theta)$. Well, we can get that easily using the formula for the conditional normal distribution:

$$s_1 | y_1 \sim N\left(A_{1|0} + P_{1|0} Z' F_{1|0}^{-1} \left(y_1 - \hat{y}_{1|0} \right), P_{1|0} - P_{1|0} Z' F_{1|0}^{-1} Z P_{1|0} \right). \tag{17}$$

Note that we could have instead obtained this using:

$$p(s_1 | y_1, \theta) \propto p(y_1 | s_1, \theta) p(s_1 | \theta), \tag{18}$$

i.e., our good friend Bayes rule! Note the conjugacy (normal-normal) likelihood-prior relationship yields a normally distributed posterior. Finally, let's call give our updated state mean and variance:

$$A_1 = A_{1|0} + P_{1|0} Z' F_{1|0}^{-1} \left(y_1 - \hat{y}_{1|0} \right) \text{ and } P_1 = P_{1|0} - P_{1|0} Z' F_{1|0}^{-1} Z P_{1|0}. \tag{19}$$

Generalization. Now, with the distribution form $s_1|y_1, \theta$, we're back where we started! So all we have to do is construct $s_2|y_1, \theta$ and $y_2|s_2, y_1, \theta$ in an identical fashion as above, and so on for $t = 2, \ldots, T$. We can summarize the recursions:

1. **Initialization.** Set $s_0 \sim N(A_0, P_0)$.

2. **Recursions.** For $t = 1, \ldots, T$:

$$\text{state prediction} : A_{t|t-1} = C + TA_{t-1} \text{ and } P_{t|t-1} = TP_{t-1}T' + RQR'. \tag{20}$$

$$\text{observation prediction} : \hat{y}_{t|t-1} = D + ZA_{t|t-1} \text{ and } F_{t|t-1} = ZP_{t|t-1}Z' + H. \tag{21}$$

$$\begin{aligned} \text{state update} : A_t &= A_{t|t-1} + P_{t|t-1}Z'F_{t|t-1}^{-1}\left(y_t - \hat{y}_{t|t-1}\right) \text{ and} \\ P_t &= P_{t|t-1} - P_{t|t-1}Z'F_{t|t-1}^{-1}ZP_{t|t-1}. \end{aligned} \tag{22}$$

Likelihood function. We can define the one-step ahead forecast error

$$\nu_t = y_t - \hat{y}_{t|t-1} = Z(s_t - A_{t|t-1}) + \eta_t. \tag{23}$$

The likelihood function is given by

$$\begin{aligned} p(Y^T|\theta) &= \prod_{t=1}^{T} p(y_t|Y^{t-1}, \theta) \\ &= (2\pi)^{-n_y T/2} \left(\prod_{t=1}^{T} |F_{t|t-1}|\right)^{-1/2} \times \exp\left\{-\frac{1}{2}\sum_{t=1}^{T} \nu_t F_{t|t-1}^{-1}\nu_t'\right\} \end{aligned} \tag{24}$$

This representation of the likelihood function is often called prediction error form, because it is based on the recursive prediction one-step ahead prediction errors ν_t. $\square$

Discussion

Initialization. First, on the initialization step, if the system-matrices are time-invariant and the process for s_t is stationary (i.e., all the eigenvalues of T are less than one in magnitude), it might make sense to initialize the Kalman filter from the invariant distribution, i.e., we have A_0 and P_0 such that

$$A_0 = (I_{n_s} - T)^{-1}C \text{ and } P_0 = TP_0T' + RQR'.$$

If the system is not too big, you can solve for P_0 directly using the *vec* operator:

$$vec(P_0) = \left(I_{n_s^2} - (T' \otimes T)\right)^{-1} RQR'.$$

Otherwise, there are algorithms available for computing P_0 reliably and quickly.

If the system is not stationary, it's common practice to set the variance of P_0 be extremely large, like $1000 \times I_{n_s}$.

Kalman Gain. In (), the matrix that maps the prediction errors, ν_t, into the state revision is important enough to warrant it's own name: the Kalman Gain. The Kalman Gain,

$$K_t = P_{t|t-1}ZF_{t|t-1}^{-1},$$

is an $n_s \times n_y$ matrix that maps the "surprises" (forecast errors) in the observed data to changes in our beliefs about the underlying unobserved states. Essentially, the gain tells us how we learn about the states from the data.

Time-varying system matrices and missing data. The Kalman filter recursions in (), (), and () are valid if the system matrices are time-varying (but pre-determined.) In practice, it is simply a matter of adding the relevant subscripts onto the system matrices. An important case of time-varying system matrices is when they are constant except for the fact that some of the observations are missing; i.e, for some t, at least one element of y_t is missing. In this case, we simply modify the observation equation ()—and hence, () and ()—in order to account for the fact that we observe fewer series at some periods. Suppose in period t we observe n_{y_t}, which is less than or equal to n_y. Define the $n_{y_t} \times n_y$ select matrix M_t, to be the matrix whose columns are comprised of $\{e_i : i\text{th series is observed}\}$, where e_i is the $n_y \times 1$ vector with a one in the ith position and zeros elsewhere. Then,

$$D_t = M_t D, \quad Z_t = M_t Z, \text{ and } H_t = M_t H M_t'. \tag{25}$$

The ability to handle missing data is an extremely powerful feature of the Kalman filter, as it allows us to both handle estimating models with missing data, and make inference about the missing data itself. More on this later. Most programmed Kalman filter routines can handle missing data without an modification of the system matrices on the part of the user. Simply code your missing data as `nan`. Finally, note that the likelihood calculation in () needs to be modified (i.e., n_y needs to be replaced by n_{y_t}.) Again, preprogrammed routines should handle this without user intervention.
"Steady-state" Kalman filter. Suppose the system matrices are constant. If we combine (), () () for the state variance, we obtain

$$P_{t+1|t} = TP_{t|t-1}T' + RQR - TP_{t|t-1}Z'(ZP_{t|t-1}Z' + H)^{-1}ZP_{t|t-1}T' \tag{26}$$

with $P_{0|-1} = P_0$. This equation is known as the *matrix Riccati recursion*, a discrete time analogue to the popular set of ODEs. Under some regularity conditions, as t gets sufficiently large, $P_{t+1|t} \to \bar{P}$, i.e., there is an invariant solution to the Riccati equation. Some people refer to this as the "steady-state" prediction variance (and correspondingly, the "steady-state" Kalman gain.) It can be useful in computation as well: after a sufficiently amount of time, one does not need to continue to update $P_{t|t-1}$, which is the typically the costliest part of evaluating the Kalman filter. Note this also makes clear that the variances in the Kalman filter to not depend on the observed data.
Caution. Some authors adopt a slightly different timing convention with the Kalman Filter; specifically, cite:DurbinKoopman2001. The initialization of the filter changes slightly. It's all very tedious.

Kalman Smoothing

Note that the Kalman filter is a *filter*: it delivers the sequence of smoothed distribution $\{s_t|Y^t\}_{t=1}^T$, which since they are normal, are simply described by the sequence $\{A_t, P_t\}_{t=1}^T$. Sometimes, we interested in the *smoothed* distributions, $\{s_t|Y^T\}_{t=1}^T$, that is distributions of the unobserved states conditional on all of the data. These distributions are also normally distributed, and can be found another recursive algorithm known as the Kalman smoother.

The Kalman smoother is more or less the Kalman filter in reverse. Let's define

$$A_{t|T} = \mathbb{E}[s_t|Y^T] \text{ and } P_{t|T} = \mathbb{V}[S_t|Y^T].$$

The Kalman smoother delivers to the sequence $\{A_{t|T}, P_{t|T}\}_{t=1}^T$. Clearly, $A_{T|T} = A_T$ and $P_{T|T} = P_T$. Consider next computing the smoothed distribution at time $T-1$. Consider the joint distribution of the form

$$\begin{bmatrix} s_{T-1} \\ s_T \\ y_T \end{bmatrix} \Bigg| Y^{T-1}, \theta \sim \mathcal{N}\left(\begin{bmatrix} A_{T-1} \\ A_{T|T-1} \\ \hat{y}_{T|T-1} \end{bmatrix}, \begin{bmatrix} P_{T-1} & P_{T-1}T' & P_{T-1}T'Z' \\ TP_{T-1} & P_{T|T-1} & P_{T|T-1}Z' \\ ZTP_{T-1} & ZP_{T|T-1} & F_{T|T-1} \end{bmatrix} \right). \tag{27}$$

Thus, the mean of $s_{T-1}|Y^T, \theta$ is given by:

$$\begin{aligned} A_{T-1|T} &= A_{T-1} + P_{T-1}T'Z'F_{T|T-1}^{-1}(y_T - \hat{y}_{T|T-1}) \\ &= A_{T-1} + \underbrace{P_{T-1}T'P_{T|T-1}^{-1}}_{J_{T-1}}(A_T - A_{T|T-1}) \qquad \text{using (22).} \end{aligned} \tag{28}$$

The variance is similarly calculated as:

$$\begin{aligned} P_{T-1|T} &= P_{T-1} - P_{T-1}T'Z'F_{T|T-1}^{-1}ZTP_{T-1} \\ &= P_{T-1} - J_{T-1}(P_T - P_{T|T-1})J_{T-1}' \qquad \text{using (22).} \end{aligned} \tag{29}$$

To extend this to $T-2$ and so on simply modify (27). Note that the procedure sketched here can be numerically unstable, most packaged software will take care of this.

Drawing from the Smoothed Distribution

Often times one wants to simulate from the smoothed distribution. Conceptually this is straightforward, but note that our sequence of smoothed distributions we derived above does not include the joint distribution of the s_ts. Drawing from the joint distribution $S^T|Y^T$ is known as *simulation smoothing*. Doing this quickly and accurately has been a topic of research of the past few decades. cite:fruhwirth1994data and cite:CaKohn94 independently developed methods of drawing samples of $S^T|Y^T$ using a recursive technique consisting of first sampling $s_T|Y^T$ and then sampling $S_{T-1}|Y^T, s^T$ and so on. Importance computational improvements were made first by cite:Jong1995 and then cite:Durbin2002.

In what follows, I'll discuss the simulation smoothing method of . This algorithm simulates from $S^T|Y^T$ by simulating from the structural shocks, ϵ_t, measurement errors, η_t, and initial condition s_0. With these simulations in hand, one can construct a draw from $S^T|Y^T$ through recursive substitution. In a nutshell

- Let $w_t = [\eta_t', \epsilon_t']'$. Thus, our disturbances are collected in the vector $[s_0, W^{T\prime}]'$. Unconditionally, this vector is distributed normally with mean $[A_0, \mathbf{0}_{1xn_yn_\epsilon}]'$ and variance $\text{diag}([P_0, H_1, Q_1, \ldots, H_T, Q_T])$. Simulate from this distribution, and call it w^+. Using this simulation, construct a counterfactual observations series Y^{+T}

- Next, use the Kalman smoother to compute $[\hat{A}, \hat{W}^{+T}] = \mathbb{E}[w^+|Y^+]$

$\mathbb{E}[W^T|Y^T]$

- Note, if x and y are normally distributed, then drawing from x and y and computing $x - \Sigma_{x,y}\Sigma_y^{-1}(y-\mu_y)$ is a draw from $x|y$.

Consider first $\epsilon_t|Y^T$. It's easy to see that this is normally distributed with

$$\mu = Q_T R_T' Z_T' F_T^{-1} \nu_T \text{ and } \sigma^2 = Q_T - Q_T R_T' Z_T' F_T^{-1} Z_T R_T Q_T$$

Next consider $(\epsilon_{t-1}, y_T)|Y^{T-1}$ again this normally distributed with:

$$\mu = \begin{bmatrix} Q_{T-1}R_{T-1}'Z_{T-1}'F_{T-1}^{-1}\nu_{T-1} \\ \hat{y}_T \end{bmatrix} \text{ and } \Sigma = \begin{bmatrix} Q_{T-1} - Q_{T-1}R_{T-1}'Z_{T-1}'F_{T-1}^{-1}Z_{T-1}R_{T-1}Q_{T-1} & \\ Z & F_T \end{bmatrix}$$

Let's also recall that $s_{T-2}, \epsilon_{T-1}, y_{t-1}|Y^{T-2}$ is normally distributed with

$$\mu = (a_{T-2|T-2}, 0, \hat{y}_{T-1|T-2}) \tag{30}$$

$$V = \begin{bmatrix} P_{T-2|T-2} & 0 & P_{T-2|T-2}T_{T-1}'Z_{T-1}' \\ 0 & Q_{T-1} & Q_{T-1}R_{T--1}'Z_{T-1}' \\ Z_{T-1}T_{T-1}P_{T-2|T-2} & Z_{T-1}R_{T-1}Q_{T-1} & F_{T-1}1 \end{bmatrix} \tag{31}$$

Thus $s_{T-2}, \epsilon_{T-1}|Y^{T-1}$

$$\mu = (Q_{T-1}R_{T--1}'Z_{T-1}'F_{T-1}^{-1}\nu_{T-1}, a_{T-2|T-2}) \tag{32}$$

$$V = \begin{bmatrix} P_{T-2|T-2} - P_{T-2|T-2}T_{T-1}'Z_{T-1}'F_{T-1}^{-1}Z_{T-1}T_{T-1}P_{T-2|T-2} & -P_{T-2|T-2}T_{T-1}'Z_{T-1}'F_{T-1}^{-1}Z_{T-1}R_{T-1}Q_{T-1} \\ -Q_{T-1}R_{T--1}'Z_{T-1}'F_{T-1}^{-1}Z_{T-1}T_{T-1}P_{T-2|T-2} & Q_{T-1} - Q_{T-1}R_{T--1}'Z_{T-1}'F_{T-1}^{-1}Z_{T-1}R_{T-1}Q_{T-1} \end{bmatrix} \tag{33}$$

Note that we can write: $y_t - \hat{y}_{t|t-1} = Z_T T_T (T_{T-1}s_{T-2} + R_{t-1}\epsilon_{t-1}) - Z_T T_T (T_{T-1}s_{T-2} + R_{t-1}\epsilon_{t-1})$

An Example: GDP+

Here I'm going to through a simple state space model described in cite:Aruoba$_{2016}$. Since GDP data is inherently noisy, the authors use both income-side (GDP_{It}) and expenditure-side (GDP_{Et}) data on GDP growth to infer the true (unobserved) growth rate, GDP_t. The authors posit that the true growth rate follows an AR(1):

$$GDP_t = \mu(1-\rho) + \rho GDP_{t-1} + \epsilon_t, \quad \epsilon_t \sim IIDN(0, \sigma^2). \tag{34}$$

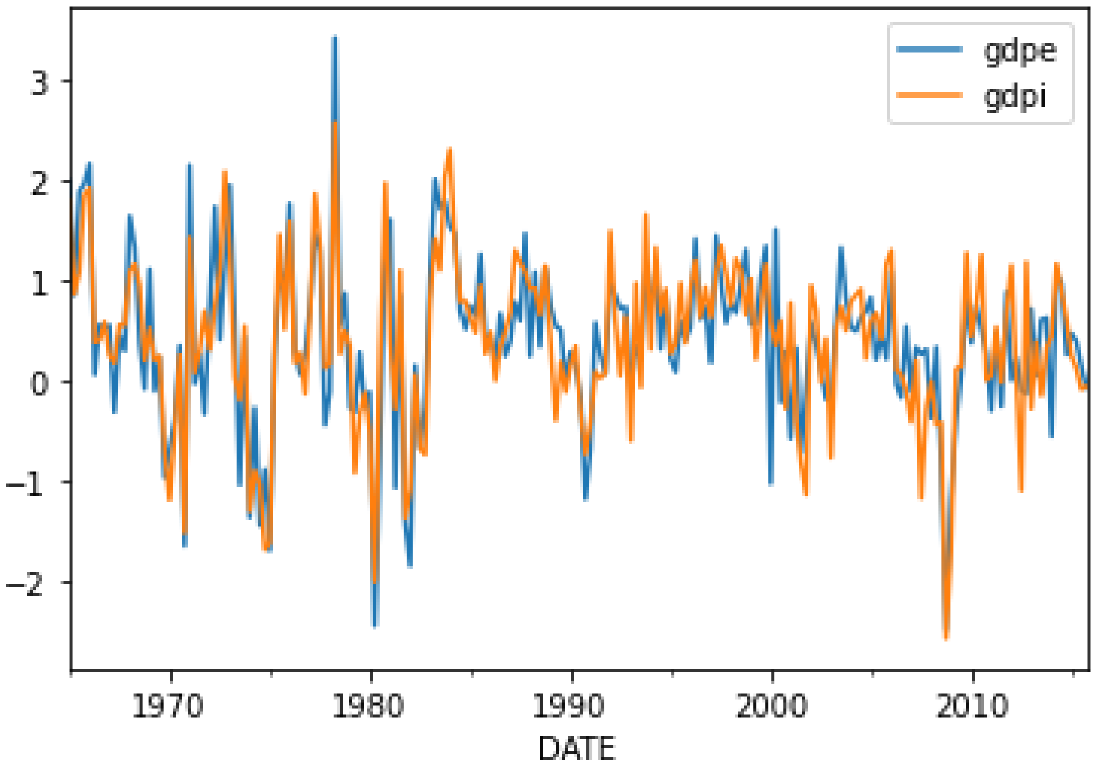
An that both income- and expenditure-side estimates are mismeasured versions of this:

$$\begin{matrix} GDP_{Et} \\ GDP_{It} \end{matrix} \Bigg| GDP_t \sim IIDN\left(\begin{bmatrix} GDP_t \\ GDP_t \end{bmatrix}, \begin{bmatrix} \sigma_E^2 & 0 \\ 0 & \sigma_I^2 \end{bmatrix} \right) \tag{35}$$

We can cast this into state space form with $n_y = 2$ and $n_s = n_\epsilon = 1$. We have

$$C = \mu(1-\rho), \quad T = \rho, \quad R = 1, \text{ and } Q = \sigma^2,$$
$$D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad Z = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \text{ and } H = \begin{bmatrix} \sigma_E^2 & 0 \\ 0 & \sigma_I^2 \end{bmatrix}. \tag{36}$$

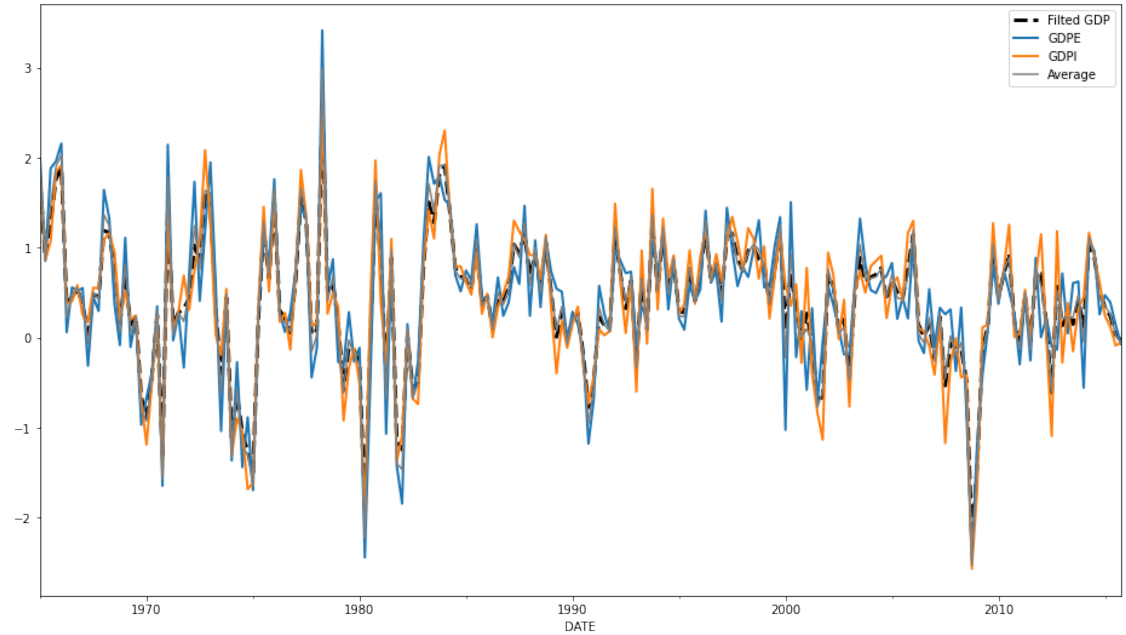
Here's a look at the data:



We can use the Kalman filter to maximize the likelihood function, since we haven't quite worked out how to elicit the posterior of this model just yet.

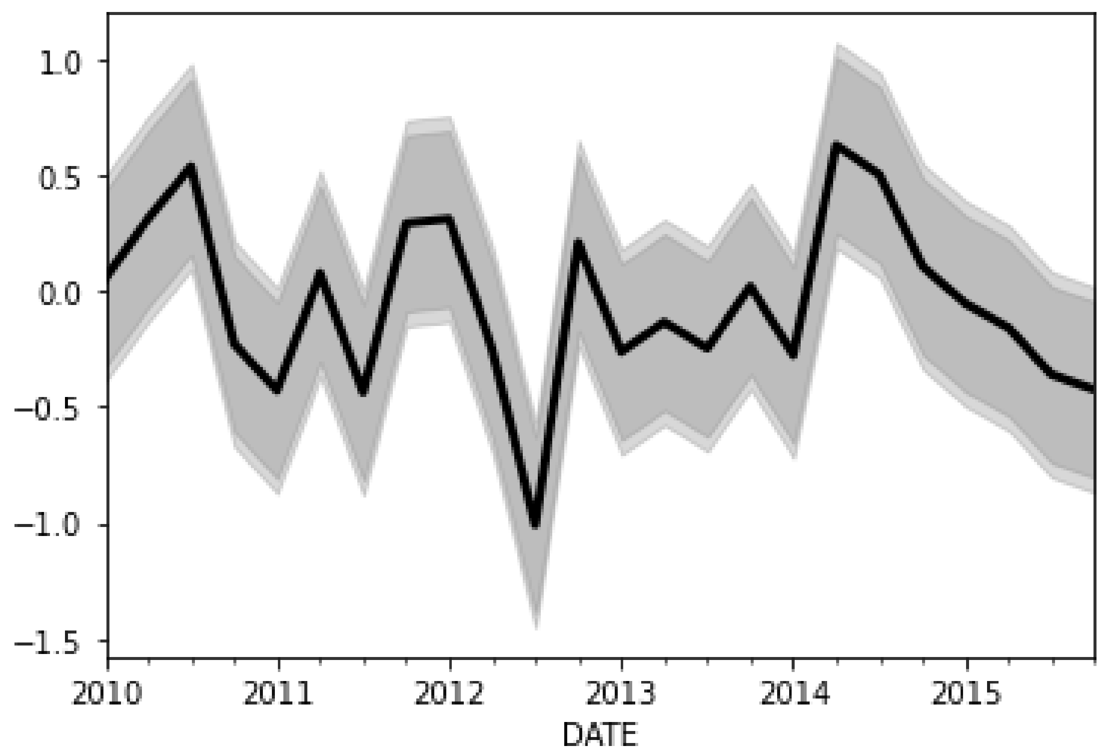
```
2025-03-14 09:49:21,192 - dsge.parser - INFO - Reading YAML from file: /home/eherbst/Dropbox/tea
2025-03-14 09:49:21,201 - dsge.parser - INFO - Detected model type: dsge
2025-03-14 09:49:21,251 - dsge.dsge.core - INFO - Validating model leads and lags
2025-03-14 09:49:21,255 - dsge.dsge.core - INFO - DSGE model 'gdp_plus' creation complete with 1
2025-03-14 09:49:21,256 - dsge.dsge.core - INFO - Initializing DSGE model
2025-03-14 09:49:21,257 - dsge.dsge.core - INFO - DSGE model initialized with 1 variables and 1
Initial likelihood:  -4358.774970922921
Maximized Likelihood:  -364.1097251748766
{'rho': 0.5009065968115182, 'mu': 0.3961715677832375, 'sige': 0.3025985769461368, 'sigi': 0.3991
```

Next, let's compute the standard errors for the estimates, computed using the hessian of the log likelihood. We these point estimates, we can use the kalman filter to extract $\{A_t\}_{t=1}^T$, the filtered means of the "true" GDP series. We'll plot them along with the observables, and the simple average of expenditure-side and income-side GDP estimates.

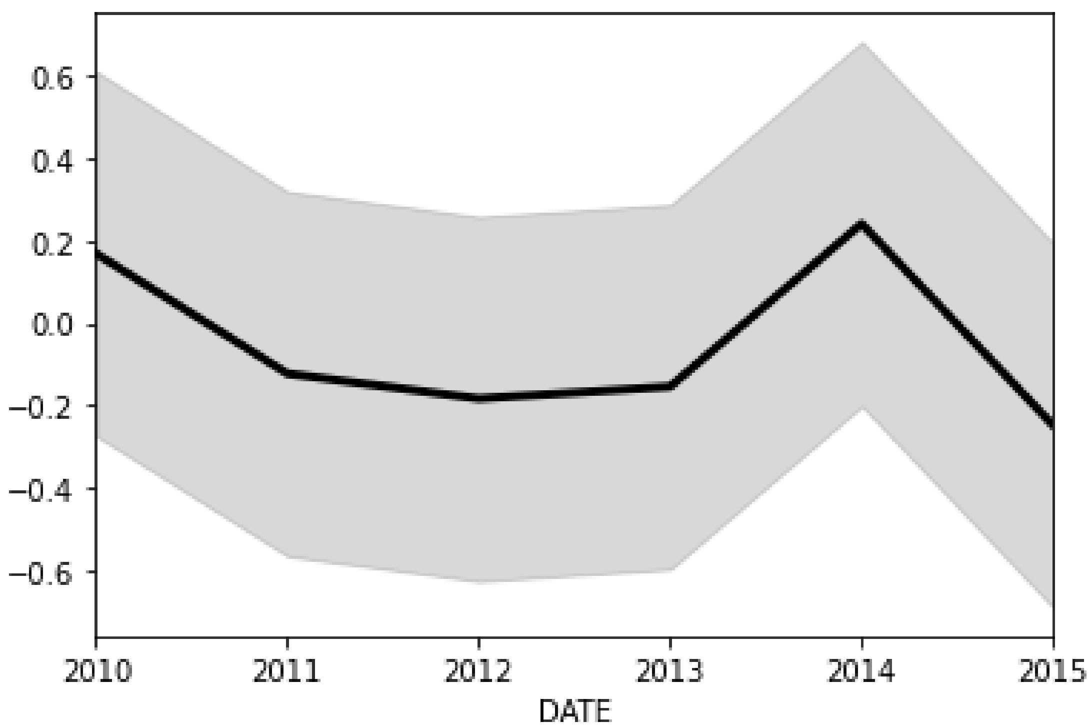


:RESULTS:

We can also compute the filtered states of the other variables, and plot them.



Let's do the same thing, but for the annual averages of GDP.


0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
2010
2011
2012
2013
2014
2015
DATE