

# SOLUTIONS TO EXERCISES ON: CONDITIONAL FORECASTING

PAWEL ZABCZYK

*Centre for Central Banking Studies  
Bank of England*

The Matlab solutions to all the exercises can be found in the CondForecasting.m file, while the Eviews solution is in CondForecasting.wf1 (additionally, the final section of this note provides step by step instructions on what buttons to press in Eviews to generate the results required). The handout below should be used in conjunction with those files and contains answers to the questions posed.

**Exercise 1.** Inspecting the VAR coefficients reveals that government spending  $g_{t-1}$  affects  $y_t$  negatively, though the coefficient is very small (-0.0048).

Computing the error term covariances then shows that the implied residuals  $\varepsilon^y$  and  $\varepsilon^g$  are positively correlated, with the covariance between the two equal to 0.113 (0.115 in Eviews). Notably, the Kalman filter will make use of the covariance matrix when generating the conditional forecast of  $g_t$  (and we won't when mechanically setting  $\varepsilon^y$  to zero in Exercise 3).

**Exercise 2.** The VAR implied unconditional forecasts are as follows:

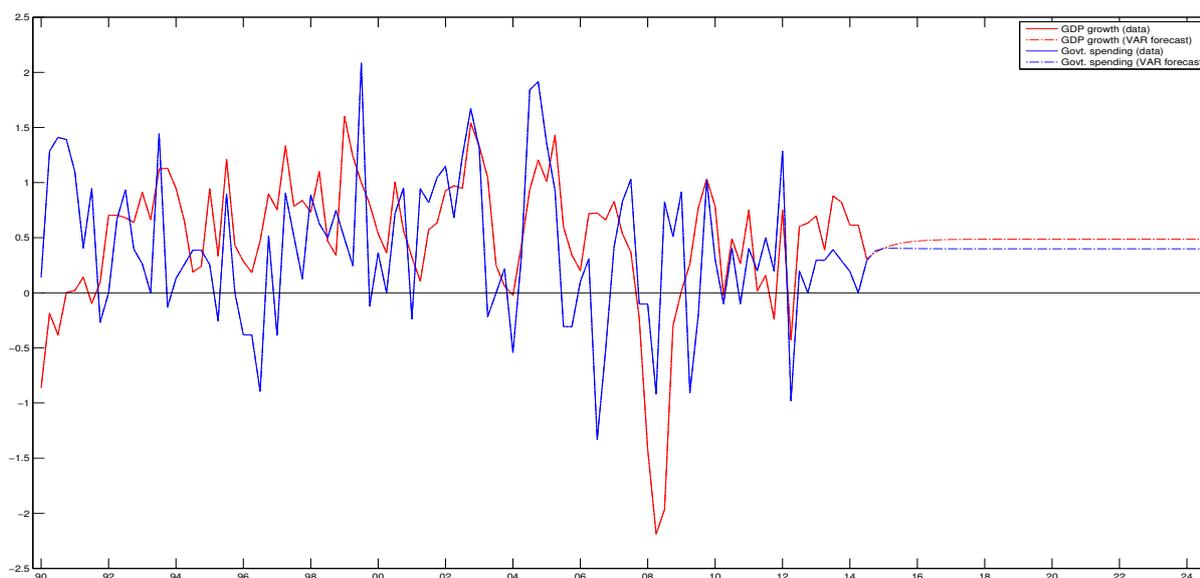


FIGURE 1. Data and unconditional forecasts from the VAR(1) model

**Exercise 3.** To compute the impact of government spending growth being higher by 1 percent over three years, we subtract the unconditional forecast from the conditional one (doing so for  $g$  will give us the profile of the ‘shock’ to  $g$  which we’re conditioning on).

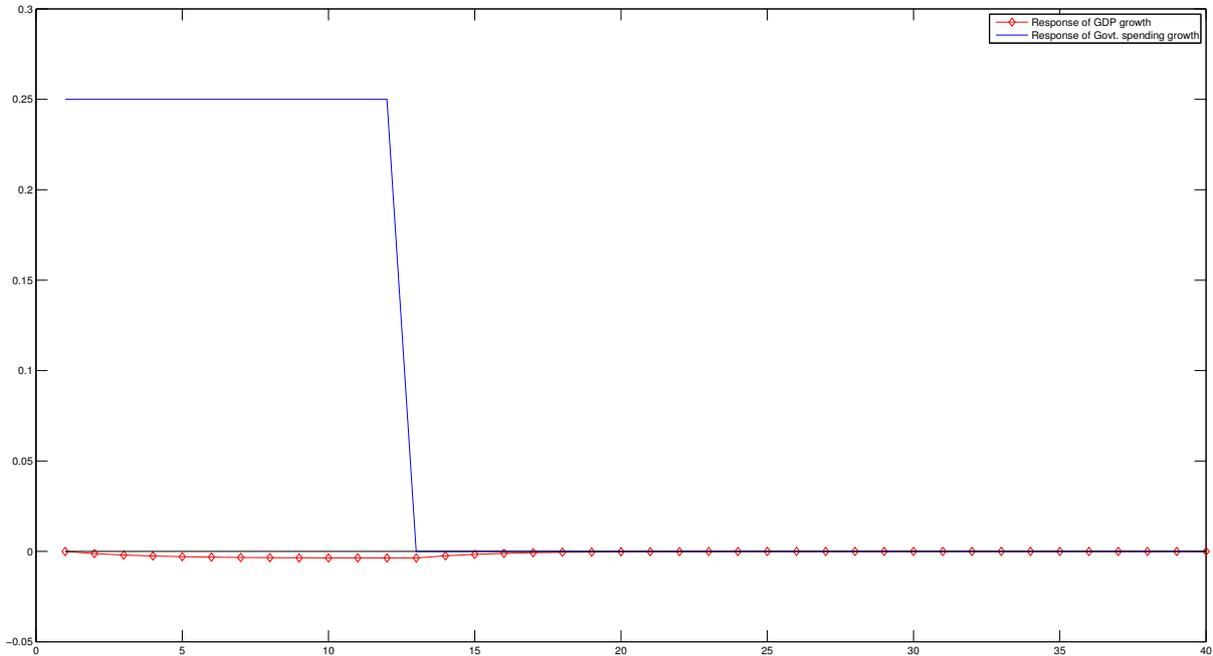


FIGURE 2. VAR implied impact of increase in government spending

Inspecting the response of GDP growth to the change in government spending implied by the VAR, we see that the response is negative and small in magnitude. Basically, as implied by the coefficients of the VAR (specifically the -0.0048 that we have already alluded to), higher values of  $g$  are associated with slightly lower values of  $y$ , which is why the fiscal expansion ends up pushing down on output.

**Exercise 4.** The Kalman-filter implied estimate of the impact of higher government spending on GDP growth is as follows:

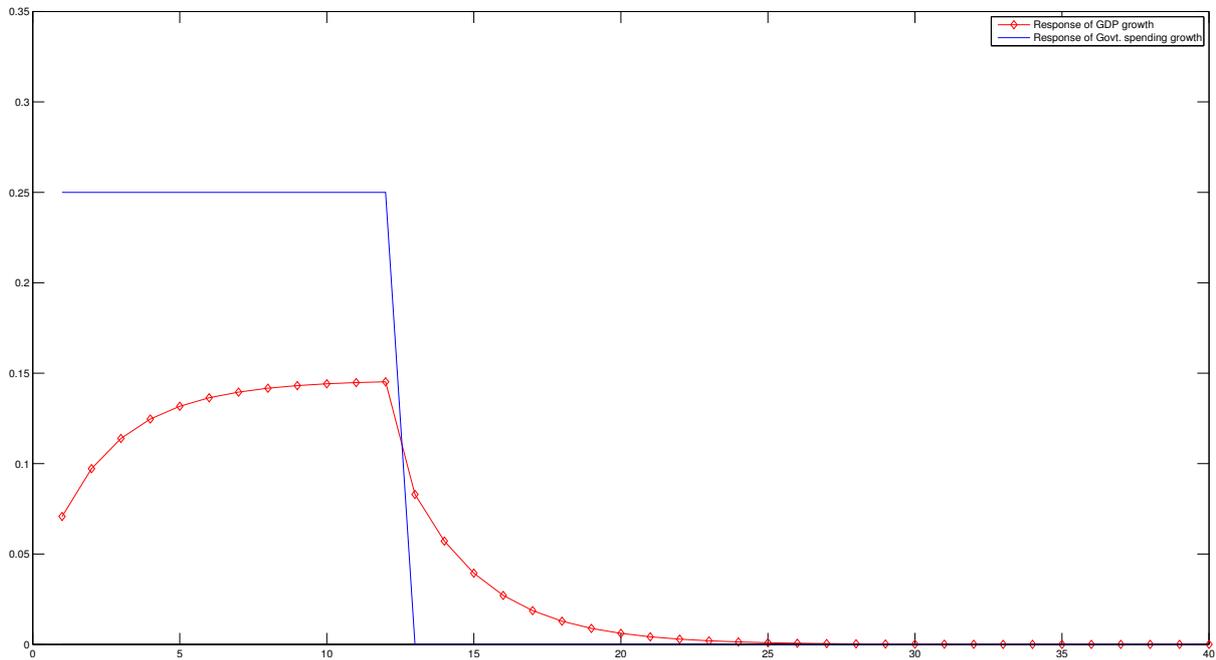


FIGURE 3. Kalman-filter implied impact of increase in government spending

We see that the paths for GDP growth implied by the Kalman filter and the VAR differ markedly - with the KF predicting a significant increase in output. As discussed in Exercise 5, the difference will be entirely driven by the different underlying profile of the residuals  $\varepsilon^y$  and  $\varepsilon^g$ .

**Exercise 5.** The implied residuals look as follows:

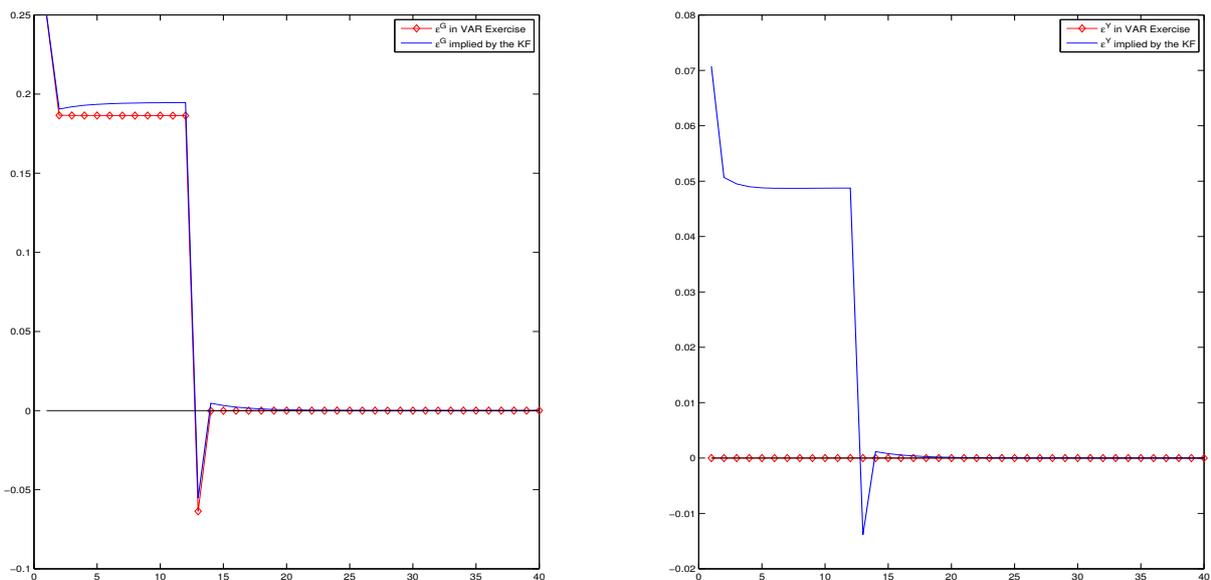


FIGURE 4. Transition equation residuals implied by the VAR and the Kalman filter

As can be readily seen, the residuals on the equation for government spending growth, i.e.  $\varepsilon^g$  implied by the VAR and the Kalman filter are relatively similar, in stark contrast to the  $\varepsilon^y$ .

Since  $\varepsilon^g$  does not affect  $y$  directly, but only indirectly via  $g$ , and since, as discussed above, the coefficient on  $g$  is very small, it is safe to conclude that the differences in the responses of output to changes in government spending are not caused by these small differences in  $\varepsilon^g$ .

Clearly, this leaves the differences in  $\varepsilon^y$  as the only possible explanation for the markedly different response of output. What is going on?

When running the VAR we mechanically set the  $\varepsilon^y$  residual to zero, meaning that higher government spending could only affect  $y$  directly through  $g$ . But the Kalman filter will not make such assumptions. Instead, based on the variance covariance matrix of the transition equation, which we provided, it will note that residuals on the two VAR equations have historically tended to be positively correlated. As a consequence, the positive values of  $\varepsilon^g$  will be associated with positive values of  $\varepsilon^y$ . This is what the Kalman filter ‘perceives’ to be the most likely pattern of residuals generating the implied path for  $g$  (the charts do suggest a strong positive correlation of residuals). These positive  $\varepsilon^y$  residuals then significantly and positively affect output  $y$  and drive the differences in responses documented in Exercise 4.

Since  $y$  itself affects  $g$  negatively (the VAR coefficient is -0.056), therefore, to ensure that  $g$  equals its conditioning path  $\tilde{g} + 25bp$ , the residuals  $\varepsilon^g$  implied by the Kalman filter actually need to be slightly higher than those implied by the VAR. This accounts for the small differences in  $\varepsilon^g$  generated by the two methods and visible in the chart above.

Hopefully this drives home the point that different combinations of residuals  $\varepsilon$  (constrained to generate the same conditioning path for  $g$ ) can actually imply very different paths for the unconstrained variables (like  $y$ ). Shocks matter, and whatever we do, we need to take a stand on their most likely combination (and even if we may not be doing that explicitly, e.g. when using the Kalman filter, the method effectively makes these choices for us)!

Note: these instructions work in Eviews 7. Slight tweaks may be necessary to make this work in other versions of Eviews.

**Exercise 1.** As usual we start by creating a workfile and loading the data.

To create a new workfile either press Ctrl+N or go to File → New → Workfile. We set the frequency to Quarterly (to match that of the data) and we fix the start date to 1990Q2 and the End date to 2025Q1 (we want space for 10 years of forecasts) and press Enter.

To import the data go to File → Import → Import from File → Choose Excel 97 - 2003 → navigate to the folder which contains the .xls file and confirm. In step 1 choose custom range and set it to DATA!\$C\$2:\$D\$101 (as we only need the  $y$  and  $g$  series) and click on Next. In step 2 set the name of the first series to  $y$  (click on the column and overwrite the entry in the name field located top right) and that of the second to  $g$  and click on next. In step 3 set the start date to 1990Q2 (retaining the quarterly frequency) and then click on Finish. We now have the two required series in the workfile.

To estimate a VAR(1) model go to Quick → Estimate VAR → Change the Lag Intervals to 1 (we only want one lag), set Endogenous variables to  $Y$  and  $G$  and make sure the Exogenous Variables contains a  $c$  (we want to have a constant in the specification) and click on OK. This should take you to a screen which has the VAR coefficients discussed in the Exercise 1 solution. To inspect the error term covariance matrix click on View (in the VAR Menu) → Residuals → Covariance matrix, which again displays the coefficients discussed in Exercise 1 (by choosing Correlation matrix we can see that the correlation coefficient equals 0.39 – i.e. the residuals are reasonably strongly positively correlated). When you close the VAR output you will be asked if you want to Delete it or Name or Store. Click on Name and call the output VAR, which will make the estimation results accessible from the workfile.

**Exercise 2.** We now want to use the VAR to generate an unconditional forecast for  $y$  and  $g$ . To forecast from a VAR we need to make it into a model. To do that, we click on Proc (in the VAR menu) → Make Model (we can label it VARmodel to avoid confusion). To then forecast using the model, click on the Solve button, and then in the Solution sample box, enter the period over which you wish to forecast (for us that will be the ten year period 2015Q2 - 2025Q1). Then hit OK (keep all the other selections at their default).

You will then find two new series in your workfile:  $y_0$  and  $g_0$ . These are composed of the data (where we have them) and the VAR forecast where we don't. By selecting the series and

clicking on **Graph** you can then display charts similar to Figure 1 reported in the solution to Exercise 2.

**Exercise 3.** We start by creating our conditioning series for  $g$  by adding 0.25 to  $g\_cond$  for twelve quarters starting in 2015Q2. One way of doing this is to create a new series called *shock* (Object → Generate Series). In the Enter equation window type  $shock = 0$ . Then click on the series (it should consist of zeros) and click on the **Genr** button and in the Enter equation type  $shock = shock + 0.25$ . Change the **Sample** to 2015Q2 2018Q1 and click on **OK** (this will add 0.25 to  $shock$  for 12 quarters starting in 2015Q2 and finishing in 2018Q1). Click on **Genr** again and this time in the Enter equation window type  $g\_cond = g\_0 + shock$  (keep the sample equal to 1990Q2 2025Q1). After clicking on **OK** you should see a new series called  $g\_cond$  in the window. It will be equal to the unconditional forecast  $g\_0 + 0.25bp$  for twelve quarters starting in 2015Q2 - i.e. it will be our conditioning path for  $g$ .

Inspecting the VAR transition equation we see that knowledge of the conditioning path for  $g$ , which we've just computed, along with the assumption that  $\varepsilon^y \equiv 0$  should suffice to back out the conditional forecast for  $y$ . The main issue being that the formula for  $y$  which is

$$y_t = \mu_y + [f_{y1} \ f_{y2}] \cdot [y_{t-1} \ g_{t-1}]'$$

is recursive in nature which creates a technical complication in Eviews. To get around this, without resorting to programming, and to back out the VAR-implied conditional forecast for  $y$  we can proceed as follows:

- In the Eviews command window, at the top of the screen type  $g = g\_cond$  and press **Enter** (this ensures that  $g$  contains future data and equals the conditioning path)
- Now go back to the VARmodel and click on **Solve**. In the **Solution scenarios & output** select **Scenario 1** and click on **Edit Scenario Options**. Click on the **Excludes** tab and tick the box saying **For all other endogenous - Exclude for periods where Actuals exist**. Click on **OK** twice.<sup>1</sup>

This will add two series to your workspace called  $g\_1$  and  $y\_1$  where  $g\_1$  is simply our conditioning path for government spending and  $y\_1$  is our conditional GDP forecast (implied by the equation above).

<sup>1</sup>This works well because we have no data for  $y$  over the forecast period. If we did, we would need to enter  $g$  in the **Variable specific excludes** box (and un-tick the box suggested above). Either way, for this to work, we would need to ensure we have values for  $g$  over the entire forecast horizon.

We can easily back out  $\epsilon_t^g$  using standard EViews commands from

$$g_t = \mu_g + f_{g1}y_{t-1} + f_{g2}g_{t-1} + \epsilon_t^g$$

which implies

$$\epsilon_t^g = g_t - (\mu_g + f_{g1}y_{t-1} + f_{g2}g_{t-1})$$

where the  $\mu_g$  and  $f_{g1}, f_{g2}$  are simply the VAR coefficients. To do so click on the VAR object and select View  $\rightarrow$  Representations. Copy the equation for  $g$  which should be

$$G = -0.0555982713592 * Y(-1) + 0.253520447232 * G(-1) + 0.324888841138$$

Then go to Object  $\rightarrow$  Generate Series and paste the equation in and rearrange to obtain

$$varepsi\_g\_VAR = g\_cond - (0.324888841138 - 0.0555982713592 * y\_1(-1) + 0.253520447232 * g\_cond(-1))$$

change the sample to 2015Q2 2025Q1 (we only care about the implied residuals over the forecast horizon) and confirm. This will create the required *varepsi\_g\_VAR* series.

Finally we can back out the impact of the shock on output. To do that, click on Object  $\rightarrow$  Generate Series and type

$$y\_g\_shock\_VAR = y\_1 - y\_0$$

which will create the response line plotted in Figure 2 (the reason it looks different is because of differences in scale).

**Exercise 4.** To use the Kalman filter to back out the conditional forecast for  $y$  we need to set up a state space model. To do that we start by setting up objects for *beta00* and *p00*

- Object (in the workfile window)  $\rightarrow$  New Object  $\rightarrow$  Matrix - Vector - Coefficient  $\rightarrow$  Type in Name beta00  $\rightarrow$  click OK
- Select Vector in Type and in Dimensions choose 2 rows and 1 column. Click on OK
- Since we want the initial value of the state to equal our last observations for  $y$  and  $g$  we then click on the Edit +/- button and set R1 equal to 0.3064805468533383 (the final observation for  $y$ ) and R2 equal to 0.2912621359223211 (the final observation for  $g$ ) and exit
- To set up *p00* go to Object (in the workfile window)  $\rightarrow$  New Object  $\rightarrow$  Matrix - Vector - Coefficient  $\rightarrow$  Type in Name p00  $\rightarrow$  click OK
- Select Symmetric Matrix in Type and in Dimensions choose 2 rows and 2 columns. Click on OK

- The matrix is full of zeros which is almost what we want (after all the initial observation is equal to data, so it makes sense to assume we observe it exactly). However, since Eviews refuses to work with a zero matrix we set the elements on the diagonal equal to 0.00000000001
- Again, it is important that you select **Symmetric Matrix** rather than just **Matrix** in order for the Kalman Filter not to crash later

We now declare the state space model

- Go to Object (in the workfile window) → New Object → SSpace → Set Name for object as ss01 → and click on OK
- This opens the blank state space object window, in which we need to type the following

```
@mprior beta00
```

```
@vprior p00
```

```
@signal g_cond = G_KF
```

```
@state Y_KF = 0.160251879731 + 0.674816838951 *Y_KF(-1) - 0.00483000017266*G_KF(-1) + ey
```

```
@state G_KF = 0.324888841138 - 0.0555982713592*Y_KF(-1) + 0.253520447232 *G_KF(-1) + eg
```

```
@ename ey
```

```
@ename eg
```

```
@evar var(ey) = 0.2142514012257589
```

```
@evar var(eg) = 0.407517726875192
```

```
@evar cov(ey,eg) = 0.1153145001272934
```

- Note that to avoid confusion, we renamed the state variables to  $G_{KF}$  and  $Y_{KF}$ , accordingly, after we run filter these variables will hold the conditional forecasts for  $g$  and  $y$  implied by the KF
- By construction, we want  $G_{KF}$  to equal the path we're conditioning on i.e.  $g_{cond}$ . This is the signal / observation equation
- The transition equation coefficients as well as variance-covariance matrix come straight from the VAR we've estimated

- Since the variance-covariance matrix is not diagonal we use a combination of the `ename` and `evar` commands to enter it, as indicated above

All we need to do now is to run the Kalman Filter. To do that go to Proc (in the state space object window) → Estimate. Since we only care about the conditional forecast over the future we change the sample period to 2015Q1 2025Q1 and click on OK. We then go to Proc (in the State Space window) → Make State Series → select Filtered State Estimates and leave only the star in the Series names window (we already have the KF suffix).

Finally, and as in the VAR exercise, we can back out the impact of the shock on output. To do that, click on Object → Generate Series and type

$$y\_g\_shock\_KF = y\_KF - y\_0$$

over 2015Q2 2025Q1 which will create the response line plotted in Figure 3 (again, the only reason it looks slightly different is the scale).

**Exercise 5.** To back out the sequences of shocks implied by the Kalman filter we proceed exactly as in Exercise 3. Specifically, we go to Object → Generate Series and type in (where the coefficients can be copied from the VAR specification)

$$varepsi\_y\_KF = y\_kf - (0.160251879731 + 0.674816838951 * y\_kf(-1) - 0.00483000017266 * g\_kf(-1))$$

and then repeat but type

$$varepsi\_g\_KF = g\_kf - (0.324888841138 - 0.0555982713592 * y\_kf(-1) + 0.253520447232 * g\_kf(-1))$$

again with the sample set to 2015Q2 2025Q2 in both cases. This generates the series reported in Figure 4.