CENTRE FOR CENTRAL BANKING STUDIES

ECONOMIC MODELLING AND FORECASTING

Measuring the effects of monetary policy: the factor-augmented VAR (FAVAR) model

by

Andrew Blake, Haroon Mumtaz and Ole Rummel
Centre for Central Banking Studies
Bank of England

13 February 2015

ole.rummel@bankofengland.co.uk
1

1   The factor-augmented vector autoregression (FAVAR) model

This exercise provides an application of a factor-augmented VAR (FAVAR) in EViews. The aim of the exercise is to measure the effects of monetary policy in a large dataset of 79 US macroeconomic variables in the spirit of Bernanke *et al.* (2005). In particular, it replicates the (semi-parametric) two-stage estimation procedure in their paper, in which factors are estimated by principal components prior to the estimation of the FAVAR.[1]

In particular, the first stage consists of four steps:

- principal component analysis is used to estimate common factors (F_t) using **all** the variables in a large dataset, X_t;
- the series in X_t are divided into slow- and fast-moving variables;
- principal component analysis is applied to the slow-moving variables to get a matrix of slow-moving factors ($F_{slow,t}$); and
- the following regression is estimated:

$$F_t = \alpha + D \times F_{slow,t} + B \times Y_t + \varepsilon_t \tag{1}$$

where Y_t is a vector of observable variables.

In a process called factor rotation, the estimated factors, $F_{new,t}$, are then calculated as the difference $F_{new,t} = F_t - B \times Y_t$. The second stage of the Bernanke *et al.* (2005) procedure consists of estimating a VAR in $F_{new,t}$ and Y_t.

The exercise elaborates on all of these points, especially the fact that the estimated factors can be used as data in second-stage regressions, even though they are generated regressors.

2   Data and EViews program

All the underlying files can be found in a folder called **FAVAR** in the current course data folder.

The FAVAR model used in this exercise is based on a quarterly dataset of 79 US variables, spanning the period from 1979 Q3 to 2007 Q4 (a total of 114 observations for each series). A list of the specific variables can be found in the file **names.xls**.

The underlying data in (approximately) stationary form are provided in the file called **data.xls**.[2] The top column of the data divides the variables into slow- and fast-moving variables, where the latter generally refer to asset prices. They are called fast-moving variables as they adjust quickly (contemporaneously) to changes in the monetary policy instrument, in this case the federal funds rate (`ffr`). To import this

---

[1] Their (fully parametric) one-step approach involves Bayesian likelihood methods and Gibbs sampling to estimate the factors and the dynamics simultaneously in a state-space model and will not be discussed here. The two methods produce qualitatively similar results, although the two-step approach tends to produce more 'plausible' results.

[2] Approximate stationarity has been achieved by either taking the first difference or the first log difference.

data into EViews in an easy way, we transferred the data into the file **dataforeviews.xls**, where the 63 slow-moving variables are denoted by `x1` to `x63`, while the 16 fast-moving variables are denoted by `z1` to `z16`. The last column of this file (`ffr`) refers to the federal funds rate.[3]

The FAVAR is estimated using the EViews program called **favar.prg**. The program is divided into several steps, which will be described below.

```
'Step 1: read data into EViews workfile
create q 1979:3 2007:4
read(a2,s=sheet1) dataforeviews.xls 80
```

The first step loads the 80 variables that are contained in **dataforeviews.xls** into EViews and creates a new (unnamed) workfile consisting of quarterly data with a sample range from 1979 Q3 to 2007 Q4.

```
'Step 2: make a group out of all the variables
group gall x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 x12 x13 x14 x15
x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 x29 x30
x31 x32 x33 x34 x35 x36 x37 x38 x39 x40 x41 x42 x43 x44 x45
x46 x47 x48 x49 x50 x51 x52 x53 x54 x55 x56 x57 x58 x59 x60
x61 x62 x63 z1 z2 z3 z4 z5 z6 z7 z8 z9 z10 z11 z12 z13 z14 z15
z16

matrix data = @convert(gall)
```

The second step creates a group (called `gall`) out of all the variables, with the exception of the federal funds rate (`ffr`). It also converts the group consisting of individual series into a matrix, called `data`, for use in Step 9 further down below.

```
'Step 3: extract the first three principal components from the
entire dataset
gall.pcomp(cor,eigval=v1,eigvec=m1) pc1 pc2 pc3
```

The third step extracts three principal components (called `pc1`, `pc2` and `pc3`) from the **entire** dataset of 79 US macroeconomic variables. Note that we use the option 'cor' when calculating principal components. This forces EViews to use the correlation matrix to extract the principal components and implies that the data does not need to be standardised (why?). The eigenvectors (eigenvalues) are saved in a matrix called `m1` (`v1`) (which will be used later on). Looking at the output, we can see that the first three principal components explain slightly more than 52 per cent of the variation in the entire dataset.

The next two steps first group the 63 **slow-moving** variables into an EViews group called `gslow` and then extract their first three principal components (called `pc1s`, `pc2s` and `pc3s`), as they will be needed in subsequent steps.

```
'Step 4: make a group out of the slow-moving variables
```

---

[3] As in Bernanke *et al.* (2005), we could assume that this is the only variable that is known to the central bank with certainty. All other variables are 'noisy' macroeconomic indicators.

```
group gslow  x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11 x12 x13 x14
x15 x16 x17 x18 x19 x20 x21 x22 x23 x24 x25 x26 x27 x28 x29
x30 x31 x32 x33 x34 x35 x36 x37 x38 x39 x40 x41 x42 x43 x44
x45 x46 x47 x48 x49 x50 x51 x52 x53 x54 x55 x56 x57 x58 x59
x60 x61 x62 x63

'Step 5: extract the first three principal components from the
slow-moving variables
gslow.pcomp(cor,eigval=v1s,eigvec=m1s) pc1s pc2s pc3s
```

Having extracted three principal components from the entire dataset in Step 3, we are confronted with the following econometric issue. The 79 information variables can be thought of as being composed of two types of variables: **slow-moving** and **fast-moving** variables. Slow-moving variables, such as real activity measures, consumer and producer prices, deflators of GDP and its components, wages or spending are assumed not to respond contemporaneously to (unanticipated) changes in monetary policy. In contrast, fast-moving variables (e.g., financial variables such as asset prices, interest rates, exchange rates and commodity prices) will respond contemporaneously to policy shocks. The three principal components (`pc1`, `pc2` and `pc3`) are extracted from a dataset that contains both types of variables, i.e., financial variables as well. It is therefore likely that these financial variables are contemporaneously correlated with the federal funds rate, `ffr`. This makes it infeasible to use the original three principal components in a VAR with `ffr` ordered last in a Cholesky – or recursive – decomposition, as the ordering of the variables matters: the variable ordered last does not affect the variables before it in the current period, while the variable ordered first affects the others immediately. This assumption obviously becomes untenable if the principal components are derived from both slow- and fast-moving variables.

We use a statistical technique called **factor rotation** to remove this correlation, such that the VAR can be identified in the standard recursive way. Practically, this amounts to estimating the following OLS regressions:

$$\text{pc1} = \gamma_1 + \beta_1 \text{pc1s} + \beta_2 \text{pc2s} + \beta_3 \text{pc3s} + \zeta \text{ffr}_t + u_t \qquad (2)$$
$$\text{pc2} = \gamma_2 + \beta_4 \text{pc1s} + \beta_5 \text{pc2s} + \beta_6 \text{pc3s} + \theta \text{ffr}_t + v_t \qquad (3)$$
$$\text{pc3} = \gamma_3 + \beta_7 \text{pc1s} + \beta_8 \text{pc2s} + \beta_9 \text{pc3s} + \omega \text{ffr}_t + z_t \qquad (4)$$

The 'new' (or rotated) principal components (`pc1n`, `pc2n` and `pc3n`) are then:

$$\text{pc1n} = \text{pc1} - \zeta \text{ffr}_t \qquad (5)$$
$$\text{pc2n} = \text{pc2} - \theta \text{ffr}_t \qquad (6)$$
$$\text{pc3n} = \text{pc3} - \omega \text{ffr}_t \qquad (7)$$

where ζ, θ and ω in (5), (6) and (7) are the estimated coefficients on `ffr` in (2), (3) and (4) respectively.

Note that the residuals from (2), (3) and (4) are not used directly. Rather, the approach employs only the difference between the old principal components and the (scaled) `ffr`. The intuition behind this set-up is that the right-hand side of the regressions in (2), (3) and (4) represents a decomposition of the common factors into

a part not affected by `ffr` (the slow-moving factors) and the impact of `ffr` (given by ζffr_t, θffr_t and ωffr_t respectively). In the EViews program, this is accomplished in Steps 6 and 7, which create the three new series `pc1n`, `pc2n` and `pc3n`.

```
'Step 6: OLS regressions of pc1, pc2 respectively on pc1s,
pc2s and pc3s and ffr
equation eq1.ls pc1 c pc1s pc2s pc3s ffr
equation eq2.ls pc2 c pc1s pc2s pc3s ffr
equation eq3.ls pc3 c pc1s pc2s pc3s ffr

'Step 7: remove the impact of ffr from pc1, pc2 and pc3
series pc1n = pc1 - (eq1.@coefs(5)*ffr)
series pc2n = pc2 - (eq2.@coefs(5)*ffr)
series pc3n = pc3 - (eq3.@coefs(5)*ffr)
```

The next step estimates a VAR with six lags using the rotated factors `pc1n`, `pc2n` and `pc3n` as well as the federal funds rate, `ffr`. Note that no attempt has been made to test for the appropriate number of lags in the VAR. The chosen lag length of six is therefore for illustrative purposes only. We could, of course, spend some time and effort selecting the optimal lag length of the VAR using either likelihood-ratio tests or information criteria (making sure that the residuals from the FAVAR have the appropriate statistical properties).

```
'Step 8: estimate a (FA)VAR with pc1n, pc2n, pc3n and ffr
var favar.ls 1 6 pc1n pc2n pc3n ffr
favar.impulse(100,g,matbys=irf) pc1n pc2n pc3n ffr @ ffr
```

Note that the monetary policy shock in the VAR is identified in the standard recursive manner, that is, by ordering the federal funds rate **last** and treating its innovations as the policy shock. Conditional on the true lag length of the FAVAR being six, do the factors enter the FAVAR with significant coefficients?

Step 8 also calculates impulse response functions over 100 periods (saved in a matrix called `irf`) for the three rotated factors and the federal funds rate in response to a shock in the federal funds rate.

Recall that we have to ensure that fast-moving variables are contemporaneously correlated with the federal funds rate, `ffr`. Step 9 ensures that this will be the case. It does this by manipulating the weights with which all the variables are reconstituted from the three principal components (and, when required, `ffr` as well). In other words, the next step builds the weights correctly, taking into account the fact that some variables (i.e., the fast-moving ones) have a contemporaneous relationship with `ffr` (while the slow-moving ones do not). The loop in the code goes first through the slow-moving variables and estimates the appropriate factor loadings between y and x using OLS (recall that the OLS estimate of the coefficient vector β in $y = X\beta + \varepsilon$ in matrix notation is given by $\beta = (X'X)^{-1}X'y$), where y is the j-th slow-moving data series ($j = 1, 2, ... , 63$) and X = (`pc1n pc2n pc3n`). Then the loop goes through the fast-moving variables ($j = 64, 65, ... , 79$) and does the same thing, but with X = (`pc1n pc2n pc3n ffr`), as these variables are related to `ffr`.

```
'Step 9: estimate the coefficients relating the data and pc1n,
pc2n, pc3n and ffr via OLS
matrix(79,4) weights 'creates a matrix that will hold the
proper weights
group pcn pc1n pc2n pc3n
matrix xx=@convert(pcn) 'new matrix containing the three
factors
scalar j
for j=1 to 63 'loop over the 63 slow-moving variables
'Left-hand side of the regression
matrix yy=@subextract(data,1,j,@rows(data),j) 'this sets up
the jth column of data for estimation
matrix btemp=@inverse(@transpose(xx)*xx)*(@transpose(xx)*yy)
weights(j,1)=btemp(1,1)
weights(j,2)=btemp(2,1)
weights(j,3)=btemp(3,1) 'the previous three lines save the
weights
next
group pcn1 pc1n pc2n pc3n ffr
matrix xx1=@convert(pcn1) 'matrix with three factors and ffr
for j=64 to 79 'loop over the 16 fast-moving variables
'Left-hand side of the regression
yy=@subextract(data,1,j,@rows(data),j) 'jth column of data
matrix
btemp=@inverse(@transpose(xx1)*xx1)*(@transpose(xx1)*yy)
weights(j,1)=btemp(1,1)
weights(j,2)=btemp(2,1)
weights(j,3)=btemp(3,1)
weights(j,4)=btemp(4,1) 'note coefficients relating ffr and
the data
next
```

We now have impulse response functions over 100 periods for the three rotated factors and the federal funds rate in response to a shock in the federal funds rate. But is this what we are interested in? What we would like to have are the impulse response functions of **all** of our underlying 79 variables with respect to the federal funds rate, rather than the impulse response functions of just the three rotated principal components. The 'beauty' of the FAVAR approach is that all impulse response functions can be easily recovered from the impulse response functions of just the three rotated principal components. This is what the final step does.

```
'Step 10: construct the impulse responses of all the
underlying variables
matrix irf1=@subextract(irf,1,13,100,16) 'irf of pc1n, pc2n
and pc3n to ffr shock
matrix irfmat=irf1*@transpose(weights)
irfmat.write(t=xls) irfmat.xls
```

3   Running the program

The EViews program **favar.prg** is run by opening it from within EViews. Go to **File/Open/Program…** and select **favar.prg** from its current location. Then click on **Run**, making sure that you also click the box at the bottom of the window (Make this the default execution mode). This ensures that EViews looks for associated programmes, such as data files, within the same location as the program you have just run.

EViews will generate all the output by itself. As instructed in Step 9, it will also create an Excel file called **irfmat.xls**, which contains the impulse response functions to a shock in `ffr` for all 79 macroeconomic variables over 100 periods.

4   Forecasting

Once the factors have been estimated, the forecast of a variable of interest, y_t, over a horizon h, can be derived from either an autoregressive-distributed-lag (ARDL) type univariate regression equation or a VAR-type multivariate model. In forecast exercises more than one step ahead, two approaches exist: the **iterated** (or recursive) approach, which, for a given forecast horizon, uses the forecasts made for the previous horizons, and the **direct** approach, which seeks to predict the value at horizon h directly, that is, without forecasting the variable of interest in the intermediate steps. In other words, **iterated** multi-period-ahead time-series forecasts are made using a one-period-ahead model, iterated forward for the desired number of periods.

Theoretically, it is *a priori* unclear whether direct or iterated forecasts are to be preferred.   While direct forecasts are better because they avoid potential misspecification error in the VAR for y_t and F_t (Boivin and Ng (2005)), iterated forecasts are better because they are more efficient if the VAR model is correctly specified (Marcellino *et al.* (2006)). In general, direct forecasting of a variable h-steps ahead ($h > 1$) reduces the forecast bias resulting from the use of estimated parameters that may appear in the case of a multi-step iterated forecast (Chevillion (2007)).

In the particular context of factor models, simulations done by Boivin and Ng (2005) do not indicate a significant difference between a direct and an iterated forecast if estimated factors are used. This fact notwithstanding, many empirical applications seem to prefer the direct approach ([Barhoumi et al. (2013)](#)).

As explained in Stock and Watson (2011), one-step-ahead forecasts of a variable, y_t (which may or may not be an element of X_t used to estimate the factors) can be computed by regressing y_{t+1} on $F_{new,t}$, y_t, lags of y_t and – optionally – additional lags of $F_{new,t}$. The ARDL(m,p)-model for direct forecasting at forecast horizon h is then of the form:

$$\hat{y}_{t+h|t} = \hat{\alpha}_h + \sum_{j=1}^{m} \hat{\beta}_{hj} \hat{F}_{new,t-j+1} + \sum_{j=1}^{p} \hat{\theta}_{hj} y_{t-j+1} \tag{8}$$

where $\hat{F}_{new,t}$ is the ($K\times1$) vector of estimated (rotated) factors, m and p are the autoregressive orders and $\hat{\beta}_{hj}$ is a vector of estimated coefficients of dimension K.[4] In direct forecast exercises, the parameters α_h, β_h and θ_h will in general depend on the particular forecast horizon h.

In an empirical analysis, Stock and Watson (2005) find that methods that include the first few (and most significant) principal components in addition to own-variable autoregressive dynamics generally work best for forecasting and that a few principal components are responsible for most of the improvement in forecasting performance. This is because the forecaster gets the benefit of using all N variables in X_t by using only K factors, where K is typically much smaller than N. In other words, the dimension of the efficient population forecasting regression does not increase as one adds variables to the system, as we consistency use the K factors only.

We conduct forecasting exercises for a set of key variables, predicting variables representing real activity, inflation and a number of financial and monetary variables. In this part of the exercise, we use a FAVAR model to forecast individual components using US data. Two EViews programs are set up to be used for this purpose. The first, called `factor_model.prg`, does the estimation, which can be altered by the researcher to build a model for the four series we try to forecast that they are happy with. In going through the program, we find that it provides – using a few key EViews tricks and short-cuts – a lot more flexibility for use than the original piece of code. We will highlight the enhanced flexibility as we go through the code.

We suggest three 'final dates' from which we try and forecast: 2002 Q4, 2004 Q4 and 2006 Q4 – which is what the first part of the EViews code below does. Selecting the appropriate starting point for the forecast now simply translates into selecting one of the three dates by appropriately removing the apostrophe in front of the selected time period and commenting out the other two dates.

```
' Last period of data
%end  = "2002q4"
' %end  = "2004q4"
' %end  = "2006q4"
```

The next two lines define the forecast horizon, i.e., h (`!nfore`) is now set to $h = 8$ quarters.

```
' Number of periods to forecast
!nfore = 8
```

Rather than hardcode both the number and the names of the (original) principal components into the EViews program, we can do both more elegantly in one go by defining a 'set' of principal components and using a placeholder (`%pc`) that, in this instance, states both the number (3) and the names of the principal components we are interested in (`pc1, pc2, pc3`). This can be easily adjusted: putting in `%pc = "pc1`

---

[4] The optimal autoregressive orders m and p can be obtained by minimising an Akaike-, Schwarz- or Hannan-Quinn-type information criterion.

pc2 pc3 pc4 pc5″, for example, defines five principal components and their names in the program.

```
' Number/names of principal components
%pc  = "pc1 pc2 pc3"
```

Just as above, we then proceed with the FAVAR analysis, making use of some of the placeholders we have created above, such as the `%end` period. In this way, we can easily define three different sample periods for later use. The first one, `sfore`, defines the period for the forecast, which runs from the previously-specified `%end` period plus one more period to the previously defined `%end` period plus the forecast horizon. The second, `sest`, is the estimation sample, which runs from the first period in the sample to the previously-specified `%end` period. The final sample called `plotfore` is the sample period for plotting the forecast. It includes both the forecast period as well as some historical data to put the forecast period into perspective.

```
' Do the FAVAR analysis

'Step 1: read data into EViews workfile
create q 1979q3 2008q4
read(a2,s=sheet1) dataforeviews.xls 80

sample sfore    {%end}+1 {%end}+!nfore
sample sest     1979q3 {%end}
sample plotfore 2000q1 {%end}+!nfore
```

In order to compare the forecast with actual outturns, both in- and out-of-sample, we save the actual data values of the variables we are going to forecast by giving the series a different name. As you can see, the new series is the old series label replaced by a more meaningful label and the suffix `_actual`, indicating that we are dealing with the actual underlying data. In other words, the five variables we will be forecasting later on are the federal funds rate (`ffr`), inflation (`x49`), real personal disposable income (`x9`), unemployment (`x37`) and wages (`x28`). This can, of course, be easily adjusted.

```
' Save real values of data for comparison
series ffr_actual   = ffr
series infl_actual  = x49
series rpdy_actual  = x9
series unemp_actual = x37
series wages_actual = x28
```

The next part of the code first loops through the 63 slow-moving variables `x1` to `x63`, setting up equation names for each of these variables, and then does the same for the remaining 16 fast-moving variables, `z1` to `z16`. In this way, we have an

equation name ready for forecasting when we change the variables that we are forecasting.

```
' Create names for equations
%xall = "x1"
for !i = 2 to 63
    %xall = %xall + " x"+@str(!i)
next
%zall = "z1"
for !i = 2 to 16
    %zall = %zall + " z"+@str(!i)
next
%pcs = ""
for %v {%pc}
    %pcs = %pcs + " " + %v + "s"
next
```

We then follow the same steps as above, i.e., we make a group out of all the variables. Using the built-in flexibility of EViews programming, note how this can now be done in one line without having to list all the 79 variables.

```
'Step 2: make a group out of all variables
group gall {%xall} {%zall}
matrix data = @convert(gall)
```

We then set the sample for analysis, taking recourse to the previously-defined sample periods. In our case, we are using the estimation sample, so rather than having to write out the beginning and end period of the sample, we use the sest placeholder that we defined above.

```
' Set chosen sample for analysis
smpl sest
```

Steps 3 through 7 are exactly ads above, except that the – admittedly clunky – programming in the earlier example has been streamlined considerably. For example, in Step 3, there is no need to write out all the principal components and their names, as we can use the %pc placeholder for that. In Step 4, we can dispense with writing out the 63 slow-moving variables as we have replaced them by %xall from above. In Step 5, the principal components based on the slow-moving variable will automatically be labelled with an s suffix using the %pcs placeholder. Finally, in Steps 6 and 7, we can do the rotation of the original factors without having to write out each of the required OLS equations.

```
'Step 3: extract the first three principal components
from the entire dataset
gall.pcomp(cor,eigval=v1,eigvec=m1) {%pc}

'Step 4: make a group out of the slow-moving variables
```

```
group gslow {%xall}

'Step 5: extract the first three principal components
from the slow-moving variables
gslow.pcomp(cor,eigval=v1s,eigvec=m1s) {%pcs}

'Steps 6 & 7: OLS regressions of pc1-3 on pc{!i}s and
ffr, remove impact in pc{!i}n
%pcn = ""
for !i = 1 to @wcount(%pc)
  equation eq{!i}.ls pc{!i} c ffr {%pcs}
  series pc{!i}N = pc{!i} - eq{!i}.@coefs(2)*ffr
  %pcn = %pcn + " pc" + @str(!i) +"N"
next
```

The final Step is the same as above, i.e., having created the rotated factors, we estimate the FAVAR in the specified number of new factors and the monetary policy instrument.

```
'Step 8: estimate the (FA)VAR with pc{!i}n and ffr
var favar.ls 1 6 {%pcn} ffr


' ***************************************************
```

The forecasting part of the program starts after the starred line. To begin with, we turn the estimated FAVAR into an EViews **model** object, which will allow us to use the (FA)VAR for forecasting. We therefore take the VAR from Step 8, which we have called `favar` and turn it into a `model` object called `favarmod`.

```
' FAVAR forecasts: make a model
favar.makemodel(favarmod)
```

Using the estimated model called `favarmod`, which for all intents and purposes is a VAR(6) in the four variables `pc1n`, `pc2n`, `pc3n` and `ffr`, we 'solve' the model over the `sfore` sample period. Recall that the `sfore` period refers to the forecasting period, i.e., having estimated the FAVAR VAR object over the `sest` sample period, we now solve the FAVAR model object into the future. The forecast values for the three principal components will automatically be labelled by the EViews suffix _0, which indicates, among other things, a forecast (this is what the line `pc{!i}n =  pc{!i}n_0` does). Similarly, the forecasted values of `ffr` will be labelled `ffr_0`. These four new variables (`pc1n_0`, `pc2n_0`, `pc3n_0`, `ffr_0`) should now have appeared in the list of variables in the workfile. They constitute forecasts of the underlying four variables entering into the FAVAR model object. We should note at this stage that the FAVAR model is not the only way of generating forecasts of the factors, although it is probably the most convenient in this context.

```
' Forecast factors
smpl sfore
favarmod.solve

for !i = 1 to @wcount(%pc)
  pc{!i}n = pc{!i}n_0
next
ffr  = ffr_0
```

At this point, we need to select the variables to forecast. Having generated forecasts for the (rotated) factors (and `ffr`), we need to do the same for the set of variables that we are interested in forecasting. We will again do this by building regression models that, after estimation, will be used for forecasting. As we will be building forecasting models, we need to set the sample period to the estimation sample, `sest`.

```
' Choose series to forecast
smpl sest
series infl  = x49    ' CPI inflation
series rpdy  = x9     ' Real personal disposable income
series unemp = x37    ' Unemployment
series wages = x28    ' Nonfarm payrolls
```

Probably the easiest forecasting model at this stage is another VAR. The following two lines therefore set up an illustrative VAR(2) in inflation (`infl`), real personal disposable income (`rpdy`), unemployment (`unemp`) and wages (`wages`). On account of the fact that it contains four variables, it is called `var4`. A more conscientious modelling approach would spend more time deciding on the appropriate lag length of the VAR, starting with the minimum lag indicated by the information criteria and making sure that the estimated VAR model displayed at least no residual autocorrelation.

```
' Default VAR of 4 variables
var var4.ls 1 2 infl rpdy unemp wages
```

With all of the above preliminary work, we can build **prototype** factor-based linear regression forecasting models. In fact, we will build a regression model each for `infl`, `rpdy`, `unemp` and `wages`, which is what the last part of the code does. Note that each EViews equation object is also turned into an EViews model object.

```
' Factor-based linear regression forecasting models
' One forecasting model for each of the variables to be
forecast

equation eqninfl.ls infl c {%pcn} ffr
eqninfl.makemodel(inflmod)
```

```
equation eqnrpdy.ls rpdy c {%pcn} ffr
eqnrpdy.makemodel(rpdymod)

equation eqnunemp.ls unemp c {%pcn} ffr
eqnunemp.makemodel(unempmod)

equation eqnwages.ls wages C {%pcn} ffr
eqnwages.makemodel(wagesmod)
```

We should feel free at this point to experiment with different forecasting models. The prototype models outlined above are deliberately kept very simple and are meant to be the starting point for a more elaborate modelling exercise. We note that all variables enter contemporaneously into the model, such that there are no dynamics (e.g., lags). We could be interested in differences across models, i.e., are the best models broadly the same? In addition, you may want to consider issues such as what the role of lagged dependent variables is and what role the factors play in each of the equations.

Once we have found satisfactory forecasting models for the four variables, a second EViews program, called `factor_forecasts.prg`, takes the estimated models from `factor_model.prg` and forecasts with them. It does so by applying a VAR in the four variables. In particular, as the first part of the EViews program shows, we take the existing `var4` EViews VAR object and make it into an EViews model object. We then change the sample period to the forecast period (`sfore`) and forecast with the VAR over that period.

```
' Forecast with VAR model
var4.makemodel(var4mod)
smpl sfore
var4mod.solve
```

The next part of the code does a bit of housekeeping. In particular, it stores all the forecasts from the VAR model by taking the individual model forecasts (which, you will remember, have a `_0` suffix) and relabeling them by the suffix `_var`, indicating that the forecasts were generated by a VAR.

```
' Store forecasts
smpl @all
series infl_var   = infl_0
series rpdy_var   = rpdy_0
series unemp_var  = unemp_0
series wages_var  = wages_0
```

In order to compare our VAR forecasts with the individual linear regression forecasting models that we arrived at before, the next part of the code forecasts over the `sfore` sample period with the four factor-based forecasting models. The four forecasts with the suffix `_0` are therefore relabelled with the suffix `_fac`, indicating that they are derived from the factor-based forecasting models.

```
' Forecast with the factor models
smpl sfore
inflmod.solve
rpdymod.solve
unempmod.solve
wagesmod.solve

' Store values
smpl plotfore
series infl_fac   = infl_0
series rpdy_fac   = rpdy_0
series unemp_fac  = unemp_0
series wages_fac  = wages_0
```

In the final part of `factor_forecasts.prg`, we group different variables and plot them. In particular, for each variable that we forecast, we show the actual values, the forecast based on the VAR and the forecast based on the individual fact-based model. In the case of inflation, that would be `infl_actual`, `infl_var` and `infl_act` respectively.

```
' Plot
group infl_g infl_actual infl_var infl_fac
group rpdy_g rpdy_actual rpdy_var rpdy_fac
group unemp_g unemp_actual unemp_var unemp_fac
group wages_g wages_actual wages_var wages_fac

infl_g.line
rpdy_g.line
unemp_g.line
wages_g.line
```

5   Structural factor-augmented VARs (SFAVARs)

To improve upon the interpretability of the FAVAR model, a more structural factor model approach could be applied. While the underlying motivation of this structural FAVAR (SFAVAR) approach remains the identification and assessment of a monetary policy shock in a data-rich environment, Belviso and Milani (2006) have suggested extracting **structural** factors from groups of variables of the same economic category rather than from a large cross-section of macroeconomic time series. In this way, factors have a clearer economic meaning and structural interpretation.[5]

In particular, Belviso and Milani (2006) partition the vector of economic variables, X_t, in such a way that each variable is explained by one of the following seven structural factors:

---

[5] Other papers to propose such an approach are the papers Ludvigson and Ng (2007, 2009).

- **real activity factor**: this factor is related to the theoretical macroeconomic concept of the output gap, providing a summary of the real economic activity situation, and is determined by variables such as industrial production, capacity utilisation, employment/unemployment indicators, inventory stocks, new and unfilled orders, consumer expenditures, etc.;
- **inflation (or price pressures) factor**: this factor provides a more comprehensive concept of inflation or price pressures in the economy and incorporates data from the evolution of consumer prices, retail prices, producer prices, wages, oil prices, etc.;
- **interest rates factor**: this factor is composed of a number of interest rates at different medium- and long-term maturities on both public and private fixed-income instruments;
- **financial market factor**: the factor is composed of various asset prices and is introduced in light of recent research that links monetary policy responses to movements in asset prices, as described in, *inter alia*, Bernanke and Gertler (2001), which allows us to verify the existence and relevance of a financial market channel of the monetary policy transmission channel;
- **money factor**: this factor consists of data on money stock variables, deposits, bank reserves, etc.;
- **credit (or credit conditions) factor**: this factor allows us to verify the empirical importance of the credit channel of the monetary policy transmission mechanism and explains many private credit and loan variables, etc.;[6] and
- **expectations factor**: this factor contains expectations regarding production, employment, inventories, new orders, inflation, short-term interest rates

We should note, however, that the assignment of variables to different sub-categories, which allows for the interpretation of the factors, still necessarily implies some arbitrary choices. Moreover, assuming that a single factor is sufficient to explain all the variables in each category may also be overly optimistic. The advantage of SFAVAR's over conventional FAVAR's obviously hinges on the plausibility of these restrictions.

These concerns notwithstanding, we have assigned our 80 variables to the eight sub-categories in the spirit of Belviso and Milani (2006). The final classification is shown in Table 1. We stress the purely **illustrative** nature of this exercise, as some of the categories are extremely small: the money, credit and expectations factors contain only three, two and one variables respectively.[7] The low number N_i of variables in particular sub-categories X_t^i can be alleviated by estimating the SFAVAR using Bayesian techniques. In other words, just like Bernanke et al. (2005), the model of Belviso and Milani (2006) can also be estimated using either a two-step (semi-

---

[6] The credit channel of the monetary policy transmission mechanism represents a potentially important channel that is frequently disregarded in structural VAR analyses.

[7] The expectations factor will therefore be equal to the one variable that is contained within the category, which is the PMI composite index.

parametric) method, in which the factors, F_t, are estimated by principal components prior to the estimation of the factor-augmented VAR; or a (fully parametric) one-step method, which makes use of Bayesian likelihood methods and Gibbs sampling to estimate the factors, F_t, and the dynamics simultaneously in a state-space model.

Some quick evidence on the plausibility of the single factors may be gauged in a principal component framework by looking at the percentages of variation explained by the first factor. In our case, for four of the seven cases, the first component explains from 59 to 97 percent of the total variability; in two cases it explains less than 50 percent.[8]

Table 1: The SFAVAR data set

Real activity factor		
1	X1	Industrial production: business equipment
2	X2	Industrial production: consumer goods
3	X3	Industrial production: durable consumer goods
4	X4	Industrial production: durable materials
5	X5	Industrial production: final products
6	X6	Industrial production: index
7	X7	Industrial production: materials
8	X8	Industrial production: nondurable consumer goods
9	X9	Real disposable personal income
10	X11	Employment: construction
11	X12	Employment: durable goods manufacturing
12	X13	Employment: education and health services
13	X14	Employment: financial activities
14	X15	Employment: good-producing industries
15	X16	Employment: government
16	X17	Employment: information services
17	X18	Employment: leisure and hospitality
18	X19	Employment: natural resources and mining
19	X20	Employment: non-durable goods manufacturing
20	X21	Employment: other services
21	X22	Employment: professional and business services
22	X23	Employment: retail trade
23	X24	Employment: service-providing industries
24	X25	Employment: total private industries
25	X26	Employment: trade, transportation and utilities
26	X27	Employment: wholesale trade
27	X28	Total non-farm payrolls: all employees
28	X29	Civilian employment
29	X30	Average duration of unemployment
30	X31	Civilian participation rate
31	X32	Civilians unemployed: 15 weeks and over
32	X33	Civilians unemployed for 15-26 weeks
33	X34	Civilians unemployed for 27 weeks and over
34	X35	Civilians unemployed for less than 5 weeks

---

[8] In the one remaining case, the factor is equal to the single underlying variable that enters into the expectations factor category, meaning that the percentage of variation explained by the first (and only) principal component is 100 per cent.

35	X36	Civilians unemployed for 5-14 weeks
36	X37	Unemployed
37	X38	Average weekly hours: manufacturing
38	X39	Real personal consumption expenditures
39	X40	Consumption: durable goods
40	X41	Consumption: non-durable goods
41	X42	Consumption: services
42	X43	Housing starts

Inflation factor

43	X49	CPI: all-items
44	X50	CPI for all urban consumers: all-items less energy
45	X51	CPI for all urban consumers: all-items less food
46	X52	CPI for all urban consumers: all-items less food and energy
47	X53	CPI for all urban consumers: energy
48	X54	CPI: food
49	X55	CPI: energy
50	X56	CPI: apparel
51	X57	CPI: commodities
52	X58	CPI: durables
53	X59	CPI: medical care
54	X60	CPI: non-durables
55	X61	CPI: services
56	X62	Earnings: construction
57	X63	Earnings: manufacturing

Interest rate factor

58	Z11	Ten-year US government bond yield index
59	Z12	Five-year US government bond yield index
60	Z13	Corporate bond yield
61	Z14	Prime lending rate
62	Z15	Moody's corporate AAA yield
63	Z16	Moody's corporate BAA yield

Financial market factor

64	Z1	US dollar trade-weighted index
65	Z2	US dollar per euro
66	Z3	US dollar per Japanese yen
67	Z4	US dollar per Canadian dollar
68	Z5	Dow Jones Industrials total return index
69	Z6	S&P 500 Energy total return index
70	Z7	S&P 500 Finance total return index
71	Z8	S&P 500 total return index
72	Z9	S&P Transportation total return index
73	Z10	S&P Utilities total return index

Money factor

74	X44	Money stock: M1
75	X45	Money stock: M2
76	X46	Money stock: MZM

Credit factor

77	X47	Total consumer credit outstanding
17800	X48	Total non-revolving credit outstanding

Expectations factor

79	X10	PMI composite index
Federal funds rate		
80	FFR	Federal Funds rate

Figure 1 shows the seven estimated factors: `pc1r` refers to the real activity factor, `pc1p` refers to the inflation factor, `pc1i` refers to the interest rate factor, `pc1f` refers to the financial market factor, `pc1m` refers to the money factor, `pc1c` refers to the credit factor and `x10`, by dint of being the only variable in the category, refers to the expectations factor. It should be noted that not all estimated factors appear (approximately) stationary, which would suggest that the model is well estimated.

This is borne out by the fact that, for a given normalisation, not all of the loadings may have their sign consistent with the structural interpretations of the factors. For instance, on the plus side, unemployed workers disaggregated by duration (series `X32` to `X36`) and the total number of unemployed all have **negative** loadings in the real activity factor. It is also possible to use the estimates to evaluate which are the most relevant series for the aggregate dynamics. We observe some counterintuitive results in the money factor. The model ought to select markedly in favour of the narrower descriptions of the money stock. What we observe is that the aggregate M1 money stock (series `X44`) has loading 0.48, the aggregate M2 money stock (series `X45`) has loading 0.60 and the MZM money stock (series `X46`) has loading 0.64.[9]

Figure 1: Estimated structural factors

---
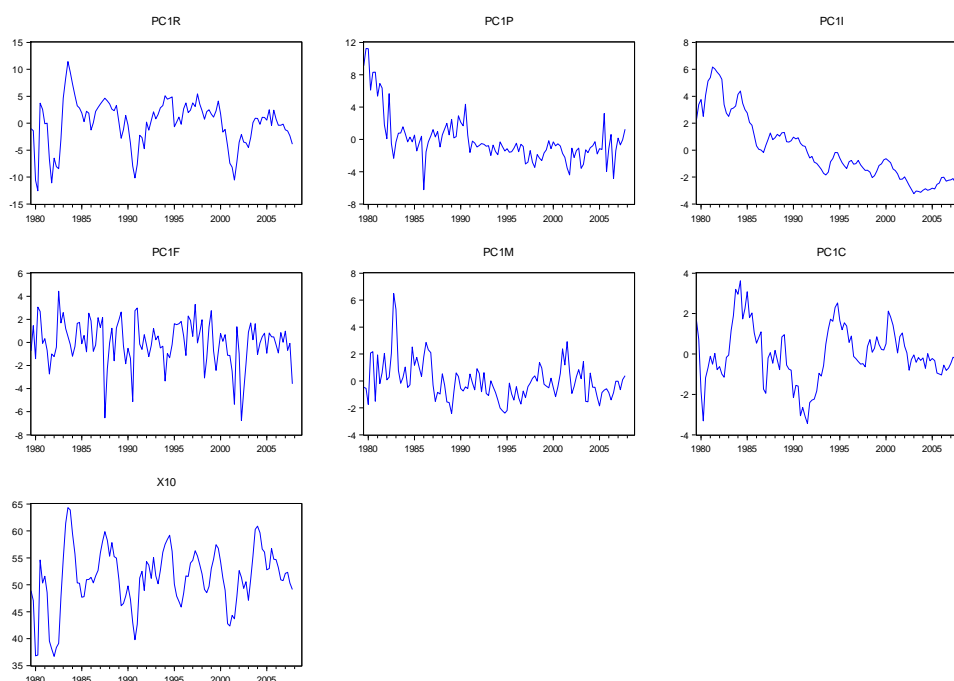
[9] MZM is a measure of the liquid money supply within an economy and represents all money in M2 less the time deposits, plus all money market funds.

This is an appropriate point to discuss one of the main drawbacks of the SFAVAR approach. The main identifying assumption requires that the errors are uncorrelated both within factor subgroups and across different subgroups, i.e., $E[X^1_{mt}X^2_{nt}|F_t] = 0$.[10] In other words, the authors assumed that the relation (and so correlation) of the variables is completely explained by the value of the factors. If this assumption is not met empirically, not only will the model be statistically mis-specified, but the economic interpretation of the factors may be questionable.

It is highly likely that this assumption will not be met empirically. An obvious counter-example involves stock-market variables, in the sense that their zero contemporaneous correlation with interest rates, inflation and output may be unrealistic. On the other hand, the assumption on the residuals is testable.[11]

Now that we have derived factors that are economically interpretable, we can examine their reaction – and the reaction of the variables used in their construction – to a monetary policy shock.

While there is now no need to differentiate between slow- and fast-moving variables and rotate the factors, we do still have to worry about the implications of the Cholesky decomposition and a recursive ordering of the variables in the FAVAR. One problem arising from the system is the presence of the **interest rate factor**, which includes data on several longer-term interest rates.

Allowing the monetary policy rate, `ffr`, to respond to the interest rate factor leads to an identification problem: we would run the risk of confusing an arbitrage condition with the policy rule (Leeper *et al.* (1996)). In the Cholesky ordering, the interest rate factor as well as the expectations factor are therefore ordered **after** the monetary policy instrument, `ffr`. In other words, we allow for a contemporaneous response of `ffr` to the other factors: inflation, real activity, credit, money and financial market (which react to monetary policy only with a lag).[12]

We retain one of the advantages of the FAVAR framework, namely that we can derive impulse-response functions not only for the (structural) factors but also for all the variables explained by the factors.

The **illustrative** SFAVAR can be estimated using the EViews program called **favar_bm.prg**.

6   Extensions of the (S)FAVAR approach

In closing, I would like to highlight three recent extensions of the (S)FAVAR approach. To begin with, identification of the structural shocks can be achieved by methods other than short-run restrictions using the Cholesky decomposition. Ahmadi and Uhlig (2012), for example, employ sign restrictions for this purpose.

Moreover, Banerjee and Marcellino (2009) have extended the FAVAR model to the factor-augmented vector error-correction model, which makes it possible to

---

[10] This means that, conditional on the factors, the contemporaneous covariances among variables should all be zero.

[11] Indeed, Belviso and Milani (2006, p. 9) explicitly mention some large correlations that exist between estimated errors for variables that are not in the same factor group, which contradicts the assumption that errors are uncorrelated not only within, but also across different factor groups.

[12] Good economic practice would call for an assessment to the robustness of the main results to different orderings of the variables in the FAVAR.

integrate variables that are non-stationary, doing away with the need to render (approximately) stationary before they enter into the FAVAR.

Finally, some recent papers have started to incorporate dynamics into the FAVAR approach, in particular in terms of:

- the factor loadings;
- the (autoregressive) coefficients of the factor VAR;
- the contemporaneous relationships between the factors; and
- the variance-covariance matrix of the innovations

A possible specification of the two equations of a time-varying-FAVAR (TV-FAVAR) encapsulating time-variation in the above three areas is therefore:

$$\begin{bmatrix} F_t \\ Y_t \end{bmatrix} = \Phi_t(L) \begin{bmatrix} F_{t-1} \\ Y_{t-1} \end{bmatrix} + v_t \tag{9}$$

$$X_t = \Lambda(F_t, Y_t) + e_t = \Lambda^f_t F_t + \Lambda^y_t Y_t + e_t \tag{10}$$

where $E(e_t) = 0$ and $E(e_t e_t') = \Sigma_{et}$. We note the time subscripts on all the coefficients (or parameters) entering equations (9) and (10), i.e., $\Phi_t(L)$, Λ^f_t, Λ^y_t and Σ_{et}.

While the TV-FAVAR given in equations (9) and (10) is generally estimated using Bayesian likelihood methods and Gibbs sampling (with some Matlab code available from, amongst many other sources, Gary Koop's website at http://personal.strath.ac.uk/gary.koop/bayes_matlab_code_by_koop_and_korobilis.html), Eickmeier *et al.* (2011) propose an alternative – classical – approach that involves a standard two-step estimation model using maximum likelihood.

Just as in the Bernanke *et al.* (2005) or the Belviso and Milani (2006) models, the first step consists of estimating the factors in a static context using principal components analysis. Then, in the second step, the model with time-varying parameters is estimated equation-by-equation using the state-space representation and the Kalman filter, meaning that the time-varying parameters evolve as (independent) random walks. As should be obvious, this approach requires the (reasonable?) assumption that the equations of the FAVAR model are conditionally independent. The latter is achieved *via* a lower-triangular matrix in the FAVAR representation, which renders the FAVAR equations conditionally independent. In terms of the volatility specification, the authors deviate from the common assumption in the literature that volatility is driven by an additional latent factor, and instead specify volatility as an (exponentially affine) function of lagged factors, which makes the FAVAR equations conditionally linear.[13]

7   Conclusions

The FAVAR methodology developed by Bernanke *et al.* (2005) avoids the degrees of freedom (curse of dimensionality) problem associated with standard VAR models. Moreover, FAVARs enjoy a number of advantages over VARs:

---

[13] Coding this up as an EViews program is currently work in progress.

- they allow for a better identification of the monetary policy shock;
- they can avoid the use of a single variable to proxy theoretical constructs, such as the output gap; and
- they allow researchers to compute impulse response functions for all variables included in the underlying large dataset

Overall, this literature has permitted researchers to incorporate larger and more realistic information sets.

At the same time, though, their shortcoming is that the factors are not identified and therefore lack any economic interpretation. This drawback has been addressed by proposing structural FAVARs (SFAVARs) that – by explaining different sub-categories of the data – provide a structural interpretation of the factors.

References and further reading

Ahmadi, P A and Uhlig, H (2012), 'Measuring the dynamic effects of monetary policy shocks: A Bayesian FAVAR approach with sign restriction', manuscript, University of Chicago.

Banerjee, A and Marcellino, M (2009), 'Factor-augmented error correction models', Chapter 9 in Castle, J L and Shephard, N (eds), *The methodology and practice of econometrics: a festschrift in honour of David F Hendry*, Oxford University Press, Oxford, pages 227-54.

Barhoumi, K, Darné, O and Ferrara, L (2013), 'Dynamic factor models: a review of the literature', *Journal of Business Cycle Measurement and Analysis*, Vol. 2013, No. 2, pages 73-107. http://www.oecd-ilibrary.org/docserver/download/3313021ec005.pdf?expires=1417937114&id=id&accname=ocid42009749&checksum=23FE99C0977BBB2AD58407517FCE0387.

Belviso, F and Milani, F (2006), 'Structural factor-augmented VARs (SFAVARs) and the effects of monetary policy', *Topics in Macroeconomics*, Vol. 6, No. 3, Article 3. http://www.bepress.com/bejm/topics/vol6/iss3/art2.

Bernanke, B S, Boivin, J and Eliasz, P (2005), 'Measuring the effects of monetary policy: a factor augmented vector autoregressive (FAVAR) approach', *Quarterly Journal of Economics*, Vol. 120, No. 1, pages 387-422.

Bernanke, B S and Gertler, M L (2001), 'Should central banks respond to movements in asset prices?', *American Economic Review*, Vol. 91, No. 2, pages 253-57.

Boivin, J and Ng, S (2005), 'Understanding and comparing factor-based forecasts', *International Journal of Central Banking*, Vol. 1, No. 3, pages 117-51.

Breitung, J and Eickmeier, S (2005), 'Dynamic factor models', *Deutsche Bundesbank Discussion Paper No 38/2005, Series 1: Economic Studies*. http://econstor.eu/bitstream/10419/19623/1/200538dkp.pdf .

Chevillon, G (2007), 'Direct multi-step estimation and forecasting', *Journal of Economic Surveys*, Vol. 21, No. 4, pages 746-85.

Eklund, J and Kapetanios, G (2008), 'A review of forecasting techniques for large data sets', *National Institute Economic Review*, Vol. 203, No. 1, pages 109-15.

Eickmeier, S, Lemke, W and Marcellino, M (2011), 'Classical time-varying FAVAR models – estimation, forecasting and structural analysis', *Deutsche Bundesbank Discussion Paper No 04/2011, Series 1: Economic Studies*. http://www.bundesbank.de/Redaktion/EN/Downloads/Publications/Discussion_Paper_1/2011/2011_04_08_dkp_04.pdf?__blob=publicationFile.

Koop, G and Korobilis, D (2009), 'Bayesian multivariate time series methods for empirical macroeconomics', *Foundations and Trends in Econometrics*, Vol. 3, No. 4, pages 267-358. http://personal.strath.ac.uk/gary.koop/koop_korobilis_Foundations_and_Trends_2010.pdf.

Leeper, E M, Sims, C A and Zha, T (1996), 'What does monetary policy do?', in Brainard, W C and Perry, G L (eds), *Brookings Papers on Economic Activity*, Vol. 1996, No. 2, pages 1-78 (with comments and discussion). http://www.brookings.edu/~/media/Projects/BPEA/1996%202/1996b_bpea_leeper_sims_zha_hall_bernanke.PDF.

Ludvigson, S C and Ng, S (2007), 'The empirical risk-return relation: a factor analysis approach', *Journal of Financial Economics*, Vol. 83, No. 1, pages 171-222.

Ludvigson, S C and Ng, S (2009), 'Macro factors in bond risk premia', *Review of Financial Studies*, Vol. 22, No. 12, pages 5027-67.

Marcellino, M, Stock, J H and Watson, M W (2006), 'A comparison of direct and iterated multistep AR methods for forecasting macroeconomic time series', *Journal of Econometrics*, Vol. 135, Nos. 1-2, pages 499-526.

Stock, J H and Watson, M W (2011), 'Dynamic factor models', Chapter 2 in Clements, M P and Hendry, D F (eds), *The Oxford handbook of economic forecasting*, Oxford, Oxford University Press, pages 35-60.

Stock, J H and Watson, M W (2012), 'Generalised shrinkage methods for forecasting using many predictors', *Journal of Business & Economic Statistics*, Vol. 30, No. 4, pages 481-93.