Estimate VAR with Parameter Constraints

estimate_VAR_with_constraints.m

by Jaromir Benes

15 March 2014

Summary

VARs can be estimated with various types of linear parameter constraints. Use two basic ways how to impose such constraints, and compare the results with the unrestricted VAR estimated previously in `estimate_simple_VAR`.

Contents

1 Clear Workspace

```
clear;
close all;
clc;
%#ok<*NOPTS>
```

2 Read Data, Dates and Previously Estimated VAR

Load historical data prepared in `read_data`, and the dates defining the start and end of the historical sample. These are the same input as in `estimate_simple_VAR`. Load also the VAR estimated previously in `estimate_simple_VAR` as a point of reference.

```
load read_data.mat g2 startHist endHist;
load estimate_simple_VAR.mat v;
```

3 Impose Parameter Constraints

Parameter constraints (i.e. constraints on the elements of the transition matrix, the constant vector, or the cointegration vector multiplier) can be imposed in two ways:

1. If you need to simply fix one or more parameters to certain numbers, use the options `'A='`, `'K='` or `'G='`. Create an NaN matrix or vector of an appropriate size (see below), assign the desired numbers to the coefficients you want to fix, and leave those coefficients that are to be estimated freely as `NaN`.

- the option `'A='` (for constraints on the transition matrix) needs to be an Ny-by-Ny-by-P matrix;
- the option `'K='` (for constraints on the constant vector) needs to be an Ny-by-1 vector;
- the option `'G='` (for constraints on the cointegration vector multipliers) needs to be an Ny-by-Ng matrix;

where Ny is the number of variables, P is the order of the VAR, and Ng is the number of cointegration vectors specificed through the option `'cointeg='`.

2. If you need to impose general linear constraints, use the option `'constraints='`, and specify a string or cell array of strings with individual constraints, refering to the individual elements of `A`, `K`, or `G`.

First, use three different ways to impose the same constraints on the transition matrix. The constraint is the the effect of the 1st and 2nd variables on the 3rd variable is zero across all lags; in other words, all the following elements of matrix `A` will be fixed to zero: `A(3,1,1)` (the effect of the 1st variable on the 3rd variable in the 1st lag), `A(3,2,1)` (2nd variable on 3rd variable, 1st lag), `A(3,1,2)` (1st variable on 3rd variable, 2nd lag), and `A(3,2,2)` (2nd variable on 3rd variable, 2nd lag).

Impose these constraints using the option `'A='` [1], the option `'constraints='` with four individual constraints specified as a cellstr, and finally using the same option `'constraints='` with the constraints specified in a more compact form.

```
yList = get(v,'yList');
p = get(v,'order');

v1 = VAR(yList);
constrA = nan(4,4,2);
constrA(3,1:2,:) = 0;
[v1,vd1] = estimate(v1,g2,startHist:endHist, ...
    'order=',p,'const=',false, ...
    'A=',constrA); [1]

v2 = VAR(yList);
[v2,vd2] = estimate(v2,g2,startHist:endHist, ...
    'order=',p,'const=',false, ...
    'constraints=',{'A(3,1,1)=0','A(3,2,1)=0','A(3,1,2)=0','A(3,2,2)=0'});

v3 = VAR(yList);
[v3,vd3] = estimate(v3,g2,startHist:endHist, ...
    'order=',p,'const=',false, ...
    'constraints=','A(3,1:2,:)=0');
```

Next, impose a more general constraint on the transition matrix: The sum of the first-lag effects of the 1st and 2nd variable on the 3rd variable is imposed to be 1.

```
v4 = VAR(yList);
[v4,vd4] = estimate(v4,g2,startHist:endHist, ...
    'order=',p,'const=',false, ...
    'constraints=','A(3,1,1)+A(3,2,1)=-1');
```

4 Compare Estimated Transition Matrices

Use the function `get` with the query `'A*'` to retrieve the transition matrix from each estimated VAR. The returned matrices are all Ny-by-Ny-by-P where Ny is the number of variables, and P is the order of the VAR. `A0` is the transition matrix of the original unconstrained VAR [2], `A1`, `A2` and `A3` are the transition matrices of the three VARs with zero constraints, [3] [4] [5] (these transition matrices are obviously identical), and `A4` is the transition matrix of the VAR with a general linear constraint [6].

```
A0 = get(v, 'A*'); [2]

A1 = get(v1,'A*'); [3]
```

```
A2 = get(v2,'A*'); 4
A3 = get(v3,'A*'); 5

A4 = get(v4,'A*'); 6

size(A0)
size(A1)
size(A2)
size(A3)
size(A4)
```

```
ans =
     4     4     2
ans =
     4     4     2
ans =
     4     4     2
ans =
     4     4     2
ans =
     4     4     2
```

The transition matrices in v1, v2, and v3 are identical.

```
maxabs(A1 - A2)
maxabs(A1 - A3)
```

```
ans =
   8.2157e-15
ans =
   8.2157e-15
```

Print the transition matrices on the 1st lag, including the original unconstrained VAR. The zeros are the result of the parameter constraints imposed in estimation.

```
A0(:,:,1)
A1(:,:,1)
A2(:,:,1)
A3(:,:,1)
```

```
ans =
    1.5382   -0.0583    0.0310   -0.0028
    0.2041    0.2333    0.0381    0.0012
   -0.4046   -0.8197    0.2376   -0.0543
   -3.9568    2.1695   -0.1406    0.2290
```

```
ans =
    1.5548   -0.0247    0.0356   -0.0031
    0.1912    0.2072    0.0345    0.0014
         0         0    0.3494   -0.0630
   -4.1181    1.8426   -0.1852    0.2324
ans =
    1.5548   -0.0247    0.0356   -0.0031
    0.1912    0.2072    0.0345    0.0014
         0         0    0.3494   -0.0630
   -4.1181    1.8426   -0.1852    0.2324
ans =
    1.5548   -0.0247    0.0356   -0.0031
    0.1912    0.2072    0.0345    0.0014
         0         0    0.3494   -0.0630
   -4.1181    1.8426   -0.1852    0.2324
```

Print the transition matrices on 2nd lag.

```
A0(:,:,2)
A1(:,:,2)
A2(:,:,2)
A3(:,:,2)
```

```
ans =
   -0.6314    0.0965   -0.0126    0.0004
   -0.0262    0.2896    0.0215    0.0235
    0.6397   -0.9185    0.1502    0.1049
    2.2617   -0.6669   -0.1176    0.0344
ans =
   -0.6576    0.1342   -0.0115    0.0017
   -0.0058    0.2604    0.0207    0.0225
         0         0    0.1755    0.1355
    2.5168   -1.0331   -0.1277    0.0222
ans =
   -0.6576    0.1342   -0.0115    0.0017
   -0.0058    0.2604    0.0207    0.0225
         0         0    0.1755    0.1355
    2.5168   -1.0331   -0.1277    0.0222
ans =
   -0.6576    0.1342   -0.0115    0.0017
   -0.0058    0.2604    0.0207    0.0225
         0         0    0.1755    0.1355
    2.5168   -1.0331   -0.1277    0.0222
```

Verify the constraint in the 4th case, `A(3,1,1)+A(3,2,1)=-1`.

```
A4(3,1,1)
A4(3,2,1)
A4(3,1,1) + A4(3,2,1)
```

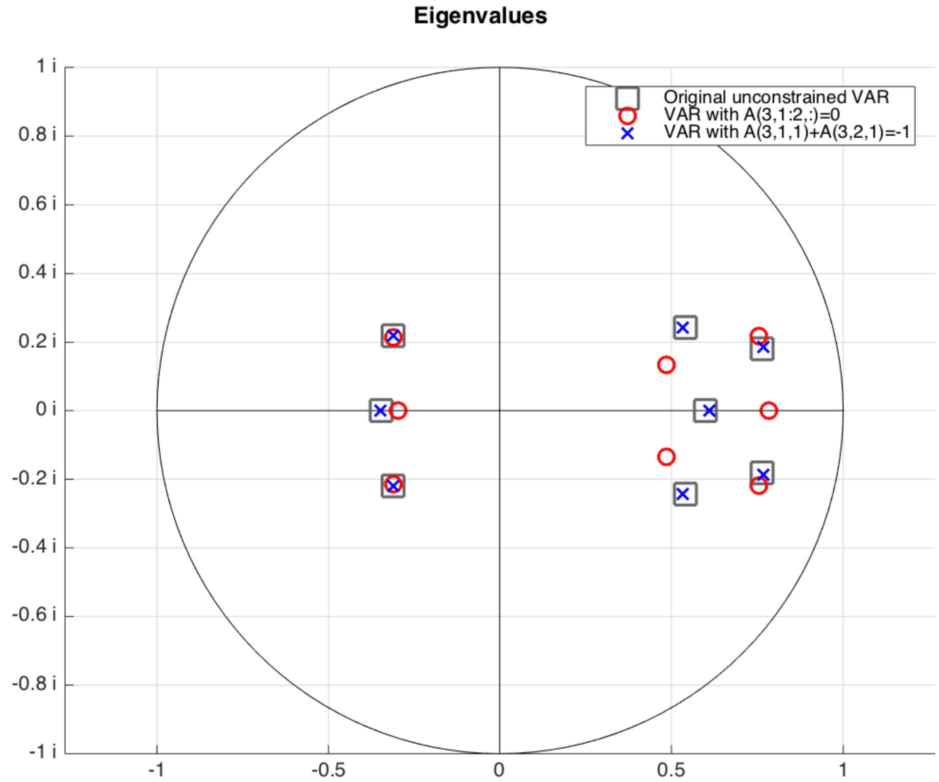
```
ans =
   -0.2583
ans =
   -0.7417
ans =
   -1.0000
```

5 Compare Eigenvalues

Plot and compare the eigenvalues for the three types of VARs: the original unconstrained one (`v`) $\boxed{7}$, the one with `A(3,1:2,:)=0` (the VAR objects `v1`, `v2`, `v3` are identical) $\boxed{8}$, and the ones with `A(3,1,1)+A(3,2,1)=-1` (`v4`) $\boxed{9}$.

```
figure();
hold on;
ploteig(v,'color=',0.4*[1,1,1],'marker=','s','markerSize=',14); 7
ploteig(v1,'color=','red','marker=','o','markerSize=',8); 8
ploteig(v4,'color=','blue'); 9
grid on;

grfun.ftitle('Eigenvalues');
legend('Original unconstrained VAR', ...
    'VAR with A(3,1:2,:)=0', ...
    'VAR with A(3,1,1)+A(3,2,1)=-1');
```



6 Help on IRIS Functions Used in This File

Use either `help` to display help in the command window, or `idoc` to display help in an HTML browser window.

```
help VAR
help VAR/VAR
help VAR/estimate
help VAR/get
help grfun/ploteig
help maxabs
```