Kalman Filtering and Historical Simulations

filter_hist_data.m

by Jaromir Benes

May 27, 2015

Summary

Run the Kalman filter on the historical data to back out unobservable variables (such as the productivity process) and shocks, and perform a number of analytical exercises that help understand the inner workings of the model.

Contents

1 Clear Workspace

Clear workspace, close all graphics figures, clear command window, and check the IRIS version.

```
clear;
close all;
clc;
irisrequired 20140315;
%#ok<*EVLC>
```

2 Load Estimated Model Object and Historical Database

Load the model object estimated in `estimate_params.m`, and the historical database created in `read_data`. Run `estimate_params` at least once before running this m-file.

```
load estimate_params.mat mest;
load read_data.mat d startHist endHist;
```

3 Run Kalman Filter

The output data struct returned from the Kalman filter, `f`, consist by default of three sub-databases:

- `mean` with point estimates of all model variables as tseries objects,
- `std` with std dev of those estimates as tseries objects,
- `mse` with the MSE matrix for backward-looking transition variables.

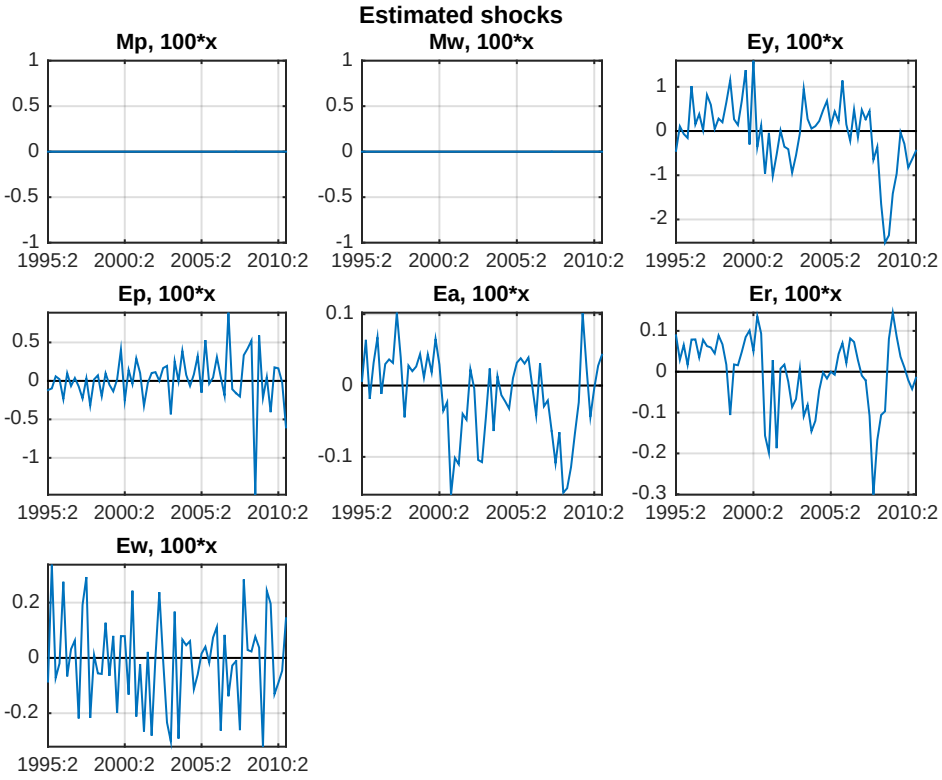
Use the options `'output='`, `'meanOnly='`, `'returnStd='` and `'returnMse='` to control what is reported in the output data struct.

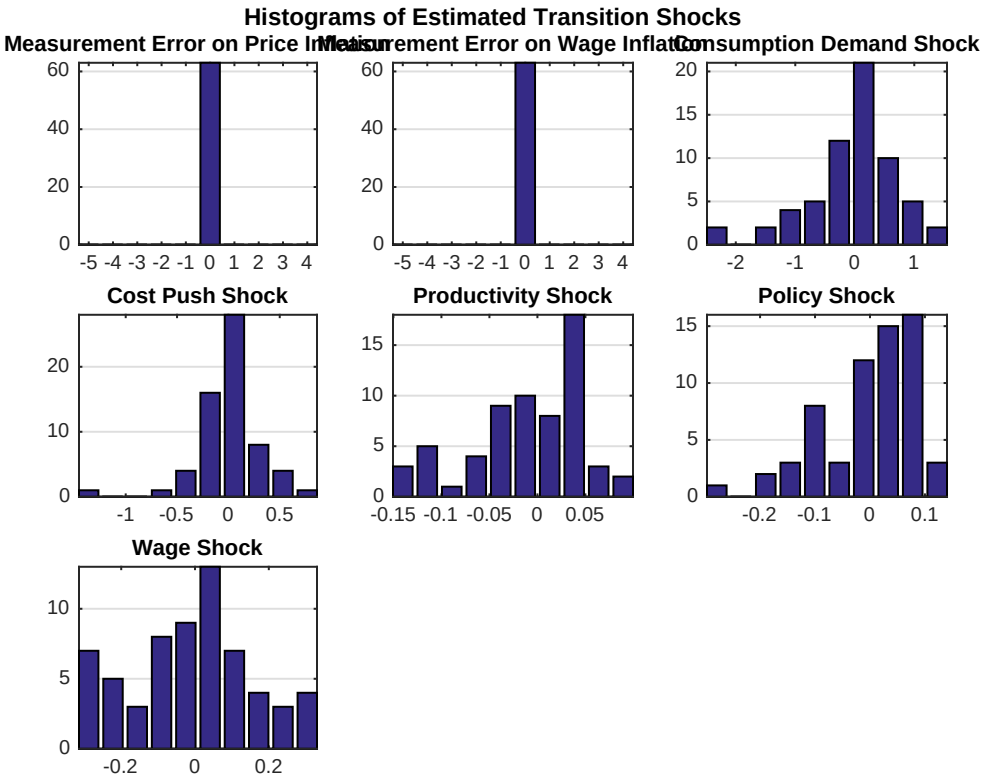
```
[~,f,v,~,pe,co] = filter(mest,d,startHist:endHist+10);
```

4 Plot Estimated Shocks

The measurement shocks are kept turned off in our exercises (i.e. their standard errors are zero), and hence their estimates are zero throughout the historical sample.

```
49 list = get(mest,'elist');
50
51 dbplot(f.mean,startHist:endHist,list, ...
52     'tight=',true,'zeroline=',true,'transform=',@(x) 100*x);
53 ftitle('Estimated shocks');
54
55 dbplot(f.mean,startHist:endHist,list, ...
56     'tight=',true,'zeroLine=',true,'plotfunc=',@hist, ...
57     'title',get(mest,'eDescript'),'transform=',@(x) 100*x);
58 ftitle('Histograms of Estimated Transition Shocks');
```





5 K-Step-Ahead Kalman Predictions

Re-run the Kalman filter requesting now also prediction step data (see the option `'output='`) extended to 5 quarters ahead (see the option `'ahead='`). Each row of the time series returned in the `.pred` sub-database contains t|t-1, t|t-2, ..., t|t-k predictions.

Because of the option `'meanOnly=' true` [1], the filter output struct, `g`, only containes mean databases directly under `.pred` and `.smooth`, and no subdatabases `.mean` are created [2].

Use the function `plotpred` [3] to organise and plot the data in a user-convenient way.

```
74 k = 8;
75
76 [~,g] = filter(mest,d,startHist:endHist, ...
77     'output=','pred,smooth','meanOnly=',true,'ahead=',k); [1]
78
79 g %#ok<NOPTS>
80 g.pred [2]
```

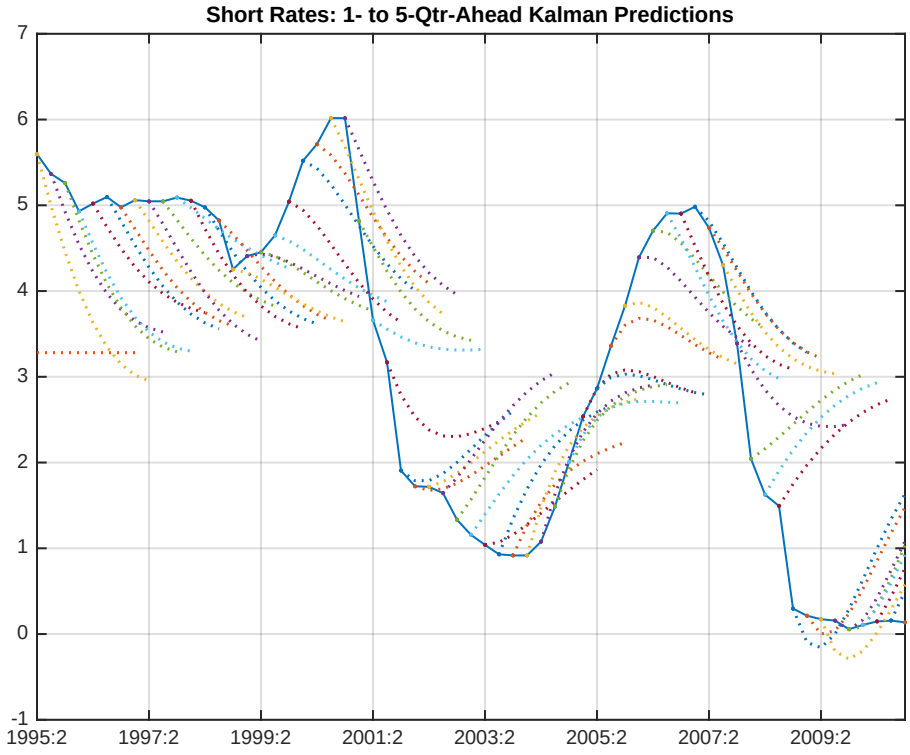
```
g.smooth

figure();
[h1,h2] = plotpred(startHist:endHist,[d.Short,g.pred.Short]); 3
set(h1,'marker','.');
set(h2,'linestyle',':','linewidth',1.5);
grid on;
title('Short Rates: 1- to 5-Qtr-Ahead Kalman Predictions');
```

```
g =
      pred: [1x1 struct]
    smooth: [1x1 struct]
ans =
      Short: [63x8 tseries]
       Infl: [63x8 tseries]
     Growth: [63x8 tseries]
       Wage: [63x8 tseries]
          Y: [63x8 tseries]
          N: [63x8 tseries]
          W: [63x8 tseries]
          Q: [63x8 tseries]
          H: [63x8 tseries]
          A: [63x8 tseries]
          P: [63x8 tseries]
          R: [63x8 tseries]
         Pk: [63x8 tseries]
         Rk: [63x8 tseries]
     Lambda: [63x8 tseries]
         dP: [63x8 tseries]
        d4P: [63x8 tseries]
         dW: [63x8 tseries]
        RMC: [63x8 tseries]
         Mp: [63x8 tseries]
         Mw: [63x8 tseries]
         Ey: [63x8 tseries]
         Ep: [63x8 tseries]
         Ea: [63x8 tseries]
         Er: [63x8 tseries]
         Ew: [63x8 tseries]
      alpha: 1.0074
       beta: 0.9962
      gamma: 0.6000
      delta: 0.0300
          k: 10
         pi: 1.0062
        eta: 6
```

```
        psi: 0.2500
        chi: 0.9138
        xiw: 133.8447
        xip: 264.6905
       rhoa: 0.9000
       rhor: 0.8587
     kappap: 2.9459
     kappan: 0.3419
     Short_: -3.9012
      Infl_: -0.3539
    Growth_: 0.0078
      Wage_: -1.9244
ans =
      Short: [63x1 tseries]
       Infl: [63x1 tseries]
     Growth: [63x1 tseries]
       Wage: [63x1 tseries]
          Y: [65x1 tseries]
          N: [63x1 tseries]
          W: [65x1 tseries]
          Q: [63x1 tseries]
          H: [63x1 tseries]
          A: [65x1 tseries]
          P: [67x1 tseries]
          R: [64x1 tseries]
         Pk: [63x1 tseries]
         Rk: [63x1 tseries]
     Lambda: [63x1 tseries]
         dP: [64x1 tseries]
        d4P: [63x1 tseries]
         dW: [64x1 tseries]
        RMC: [63x1 tseries]
         Mp: [63x1 tseries]
         Mw: [63x1 tseries]
         Ey: [63x1 tseries]
         Ep: [63x1 tseries]
         Ea: [63x1 tseries]
         Er: [63x1 tseries]
         Ew: [63x1 tseries]
      alpha: 1.0074
       beta: 0.9962
      gamma: 0.6000
      delta: 0.0300
          k: 10
         pi: 1.0062
        eta: 6
```

```
      psi: 0.2500
      chi: 0.9138
      xiw: 133.8447
      xip: 264.6905
     rhoa: 0.9000
     rhor: 0.8587
   kappap: 2.9459
   kappan: 0.3419
   Short_: -3.9012
    Infl_: -0.3539
  Growth_: 0.0078
    Wage_: -1.9244
```



6 Resimulate Filtered Data

This is to illustrate that running a simulation with the initial conditions and shocks estimated by the Kalman filter exactly reproduces the historical paths of the observables.

```
96 s = simulate(mest,f.mean,startHist:endHist,'anticipate=',false);
97
98 dbfun(@(x,y) max(abs(x-y)),f.mean,s)
```

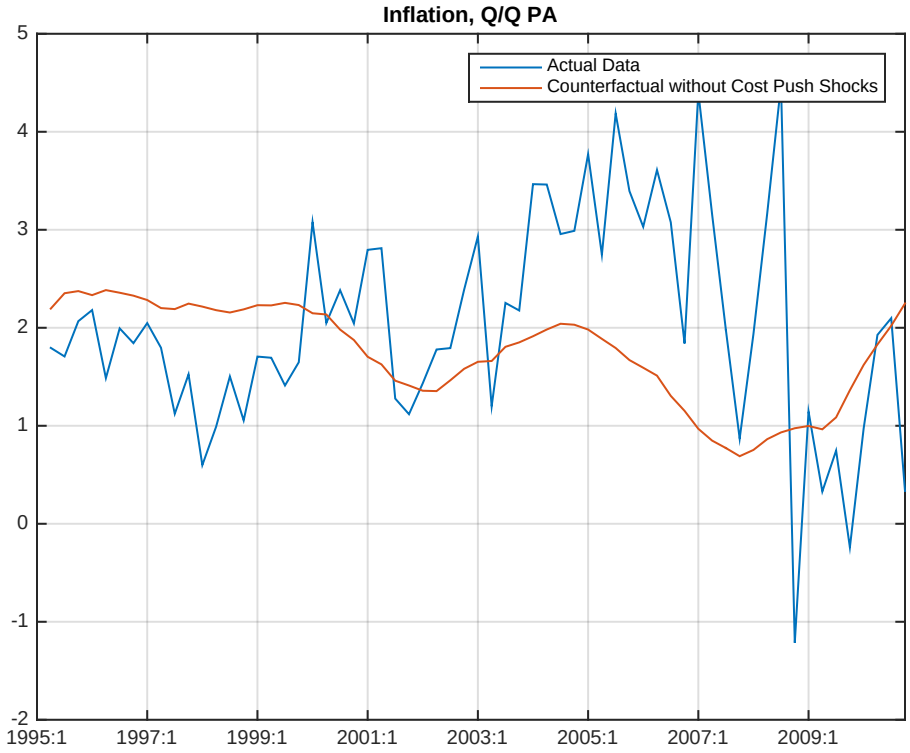
```
ans =
      Short: 5.4179e-14
       Infl: 7.2831e-14
     Growth: 4.2633e-14
       Wage: 1.0392e-13
          Y: 1.5543e-15
          N: 2.2204e-16
          W: 7.1054e-15
          Q: 1.9984e-15
          H: 1.5543e-15
          A: 8.8818e-16
          P: 2.6645e-15
          R: 2.2204e-16
         Pk: 5.7732e-15
         Rk: 2.2204e-16
     Lambda: 1.5543e-15
         dP: 2.2204e-16
        d4P: 6.6613e-16
         dW: 4.4409e-16
        RMC: 5.5511e-16
         Mp: 0
         Mw: 0
         Ey: 0
         Ep: 0
         Ea: 0
         Er: 0
         Ew: 0
      alpha: 0
       beta: 0
      gamma: 0
      delta: 0
          k: 0
         pi: 0
        eta: 0
        psi: 0
        chi: 0
        xiw: 0
        xip: 0
       rhoa: 0
       rhor: 0
     kappap: 0
```

```
    kappan: 0
    Short_: 0
     Infl_: 0
   Growth_: 0
     Wage_: 0
```

7 Run Counterfactual

Remove the cost-push shocks from the filtered database, and re-simulate the historical data. This experiment shows what the data would have looked like if inflation had been determeined exactly by the Phillips curve without any cost-push shocks.

```
107 f1 = f.mean;
108 f1.Ep(:) = 0;
109
110 s1 = simulate(mest,f1,startHist:endHist,'anticipate=',false);
111
112 figure();
113 plot([s.Infl,s1.Infl]);
114 grid on;
115 title('Inflation, Q/Q PA');
116 legend('Actual Data','Counterfactual without Cost Push Shocks');
```



8 Simulate Contributions of Shocks

Re-simulate the filtered data with the 'contributions=' option set to true. This returns each variable as a multivariate time series with $n+1$ columns, where n is the number of model shocks. The first n columns are contributions of individual shocks (in order of their appearance in the !transition_shocks declaration block in the model file), the last, $n+1$-th column is the contribution of the initial condition and/or the deterministic drift.

```
128 c = simulate(mest,s,startHist:endHist+8, ...
129     'anticipate=',false,'contributions=',true,'dboverlay=',true);
130
131 c %#ok<NOPTS>
132 c.Infl
```

```
c =
      Short: [71x9 tseries]
       Infl: [71x9 tseries]
```

```
     Growth: [71x9 tseries]
       Wage: [71x9 tseries]
          Y: [73x9 tseries]
          N: [71x9 tseries]
          W: [73x9 tseries]
          Q: [71x9 tseries]
          H: [71x9 tseries]
          A: [73x9 tseries]
          P: [75x9 tseries]
          R: [72x9 tseries]
         Pk: [71x9 tseries]
         Rk: [71x9 tseries]
     Lambda: [71x9 tseries]
         dP: [72x9 tseries]
        d4P: [71x9 tseries]
         dW: [72x9 tseries]
        RMC: [71x9 tseries]
         Mp: [71x9 tseries]
         Mw: [71x9 tseries]
         Ey: [71x9 tseries]
         Ep: [71x9 tseries]
         Ea: [71x9 tseries]
         Er: [71x9 tseries]
         Ew: [71x9 tseries]
      alpha: 1.0074
       beta: 0.9962
      gamma: 0.6000
      delta: 0.0300
          k: 10
         pi: 1.0062
        eta: 6
        psi: 0.2500
        chi: 0.9138
        xiw: 133.8447
        xip: 264.6905
       rhoa: 0.9000
       rhor: 0.8587
     kappap: 2.9459
     kappan: 0.3419
     Short_: -3.9012
      Infl_: -0.3539
    Growth_: 0.0078
      Wage_: -1.9244
ans =
        tseries object: 71-by-9
        1995Q2:  0     0   -0.017867    -0.40054   0.0011837   -0.024855   -0.023286     2.2604     0
```

1995Q3:	0	0	-0.013561	-0.64523	0.014462	-0.055547	0.049371	2.3578	0
1995Q4:	0	0	-0.0038854	-0.30689	0.027567	-0.099555	0.092359	2.3579	0
1996Q1:	0	0	0.0061424	-0.15265	0.049125	-0.13973	0.11555	2.3021	0
1996Q2:	0	0	0.06543	-0.89838	0.085689	-0.1904	0.1995	2.2242	0
1996Q3:	0	0	0.093521	-0.36362	0.12422	-0.24838	0.23976	2.1485	0
1996Q4:	0	0	0.10235	-0.48483	0.17007	-0.29775	0.2626	2.0904	0
1997Q1:	0	0	0.073449	-0.23543	0.2225	-0.34823	0.27838	2.0571	0
1997Q2:	0	0	0.053225	-0.40489	0.27921	-0.39345	0.21203	2.05	0
1997Q3:	0	0	0.013825	-1.0681	0.35174	-0.43104	0.19044	2.0656	0
1997Q4:	0	0	-0.069015	-0.72185	0.43043	-0.45637	0.24316	2.0985	0
1998Q1:	0	0	-0.16255	-1.6172	0.49911	-0.48213	0.21994	2.1418	0
1998Q2:	0	0	-0.26051	-1.1894	0.56514	-0.50255	0.1885	2.1891	0
1998Q3:	0	0	-0.33429	-0.6522	0.6234	-0.50415	0.1359	2.2349	0
1998Q4:	0	0	-0.37467	-1.1331	0.67296	-0.45433	0.068986	2.2749	0
1999Q1:	0	0	-0.44527	-0.52441	0.71666	-0.39378	0.046153	2.3067	0
1999Q2:	0	0	-0.53533	-0.5343	0.75002	-0.32851	0.013327	2.3289	0
1999Q3:	0	0	-0.60461	-0.84296	0.77982	-0.27364	0.011141	2.3416	0
1999Q4:	0	0	-0.63389	-0.5835	0.80353	-0.24312	-0.04019	2.3456	0
2000Q1:	0	0	-0.72458	0.92997	0.83096	-0.24214	-0.058403	2.3424	0
2000Q2:	0	0	-0.75746	-0.087065	0.8583	-0.2542	-0.044425	2.3336	0
2000Q3:	0	0	-0.85349	0.40185	0.87476	-0.29932	-0.061581	2.3212	0
2000Q4:	0	0	-0.94581	0.16937	0.87955	-0.35958	-0.0055448	2.3066	0
2001Q1:	0	0	-1.0622	1.0911	0.84544	-0.3586	-0.011655	2.2912	0
2001Q2:	0	0	-1.1132	1.1855	0.77357	-0.28473	-0.025117	2.2761	0
2001Q3:	0	0	-1.1416	-0.18253	0.65741	-0.20883	-0.10888	2.262	0
2001Q4:	0	0	-1.0989	-0.29281	0.50911	-0.078662	-0.17	2.2494	0
2002Q1:	0	0	-0.96937	0.07641	0.33371	0.039636	-0.28478	2.2384	0
2002Q2:	0	0	-0.80186	0.42458	0.15409	0.13392	-0.36123	2.229	0
2002Q3:	0	0	-0.61463	0.3284	-0.020679	0.21038	-0.33184	2.221	0
2002Q4:	0	0	-0.44158	0.80668	-0.19679	0.28388	-0.27881	2.2144	0
2003Q1:	0	0	-0.25646	1.2789	-0.37068	0.3474	-0.27619	2.2087	0
2003Q2:	0	0	-0.048933	-0.46221	-0.53129	0.3797	-0.34226	2.2039	0
2003Q3:	0	0	0.19301	0.44805	-0.66537	0.41536	-0.33689	2.1997	0
2003Q4:	0	0	0.38157	0.32554	-0.78513	0.44798	-0.38927	2.1958	0
2004Q1:	0	0	0.49569	1.552	-0.87659	0.49644	-0.39421	2.1923	0
2004Q2:	0	0	0.54439	1.4794	-0.94347	0.5524	-0.36005	2.1889	0
2004Q3:	0	0	0.54206	0.91527	-0.98807	0.59246	-0.29115	2.1857	0
2004Q4:	0	0	0.50675	0.95808	-1.0149	0.60333	-0.24655	2.1826	0
2005Q1:	0	0	0.44777	1.7897	-1.0204	0.59058	-0.21561	2.1797	0
2005Q2:	0	0	0.33884	0.85836	-1.0034	0.55303	-0.17856	2.1768	0
2005Q3:	0	0	0.21627	2.3935	-0.96394	0.49801	-0.13086	2.1741	0
2005Q4:	0	0	0.078117	1.7241	-0.90445	0.41746	-0.092206	2.1716	0
2006Q1:	0	0	-0.023557	1.4383	-0.82518	0.31136	-0.039709	2.1693	0
2006Q2:	0	0	-0.15391	2.0984	-0.73609	0.20198	0.032604	2.1671	0
2006Q3:	0	0	-0.30604	1.7711	-0.64957	0.080195	0.015437	2.1651	0
2006Q4:	0	0	-0.42012	0.68732	-0.55978	-0.044005	0.01307	2.1633	0

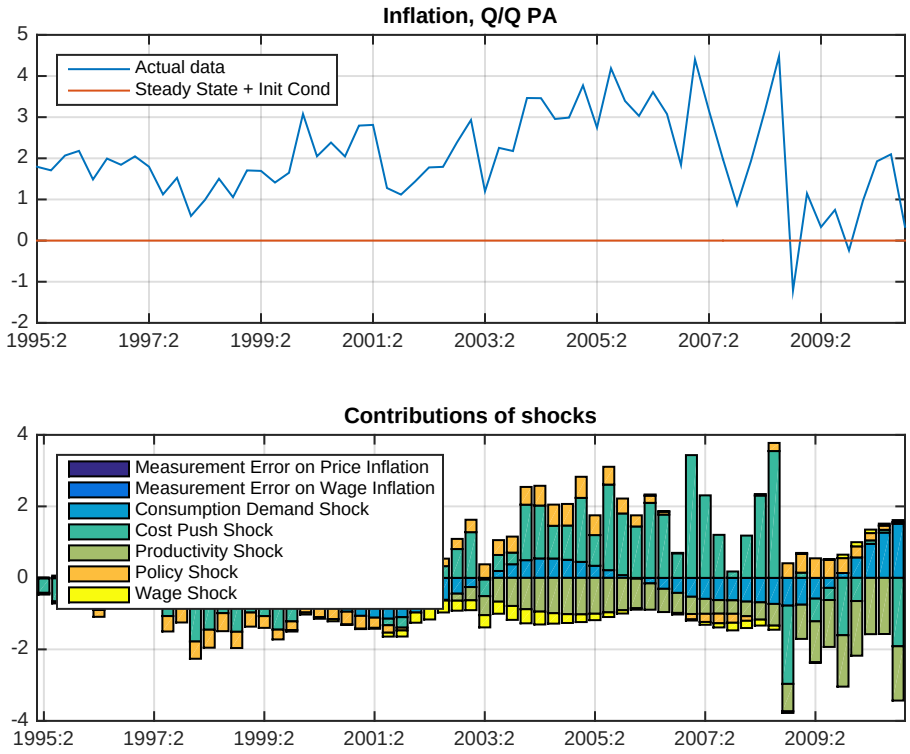
```
        2007Q1:  0     0    -0.52804      3.4344    -0.48163    -0.15151   -0.031534     2.1617     0
        2007Q2:  0     0     -0.5874      2.3086    -0.41786     -0.2275    -0.07877     2.1602     0
        2007Q3:  0     0    -0.62155      1.2042    -0.37917    -0.26781    -0.11746     2.1589     0
        2007Q4:  0     0    -0.62805     0.17916    -0.37691    -0.25087    -0.21231     2.1577     0
        2008Q1:  0     0      -0.664      1.1829    -0.40664    -0.13021    -0.20221     2.1566     0
        2008Q2:  0     0    -0.68399      2.3041    -0.48358    0.041157    -0.16579     2.1557     0
        2008Q3:  0     0    -0.72845      3.5475    -0.60691      0.2275    -0.11461     2.1548     0
        2008Q4:  0     0    -0.77585     -2.1895    -0.76942     0.40925   -0.043092      2.154     0
        2009Q1:  0     0    -0.75002     0.14627    -0.95662     0.52344    0.028221     2.1533     0
        2009Q2:  0     0      -0.581     -0.6347     -1.1498     0.54757  -0.0053699     2.1526     0
        2009Q3:  0     0    -0.28001    -0.33898     -1.3121     0.50334    0.022627      2.152     0
        2009Q4:  0     0     0.13662     -1.6033     -1.4384     0.41663    0.096681     2.1515     0
        2010Q1:  0     0     0.57258    -0.64764     -1.5289     0.30925     0.11731      2.151     0
        2010Q2:  0     0      0.9517    0.097829     -1.5739     0.20379    0.097263     2.1505     0
        2010Q3:  0     0      1.2668    0.077389     -1.5698     0.11723    0.056372     2.1501     0
        2010Q4:  0     0      1.5141      -1.913     -1.5186    0.048387    0.054827     2.1497     0
        2011Q1:  0     0      1.7026      -1.487     -1.4341  -0.0039307    0.052329     2.1493     0
        2011Q2:  0     0      1.8158     -1.1223     -1.3252   -0.039407    0.048314      2.149     0
        2011Q3:  0     0      1.8501    -0.82291     -1.2023    -0.05957    0.042806     2.1488     0
        2011Q4:  0     0      1.8114    -0.58644      -1.075   -0.067078     0.03619     2.1485     0
        2012Q1:  0     0      1.7121    -0.40675    -0.95139   -0.065097    0.029021     2.1483     0
        2012Q2:  0     0       1.568     -0.2755    -0.83728   -0.056812     0.02188     2.1481     0
        2012Q3:  0     0      1.3956    -0.18355    -0.73638   -0.045089    0.015269     2.1479     0
        2012Q4:  0     0      1.2106    -0.12201     -0.6503   -0.032264   0.0095568     2.1477     0
  Columns 1 through 3
    'Infl <--[+] Mp'    'Infl <--[+] Mw'    'Infl <--[+] Ey'
  Columns 4 through 6
    'Infl <--[+] Ep'    'Infl <--[+] Ea'    'Infl <--[+] Er'
  Columns 7 through 9
    'Infl <--[+] Ew'    'Infl <--[+] Init...'    'Infl <--[+] Nonl...'
        user data: empty
        export files: [0]
```

To plot the shock contributions, use the function `conbar`. Plot first the actual data and the effect of the initial condition and deterministic constant (i.e. the last, $n+1$-th column in the database `c`) in the upper panel, and then the contributions of individual shocks, i.e. the first n columns.

```
figure();

subplot(2,1,1);
plot(startHist:endHist,[s.Infl,c.Infl{:,end}]);
grid on;
title('Inflation, Q/Q PA');
legend('Actual data','Steady State + Init Cond', ...
    'location','northWest');
```

```
150
151 subplot(2,1,2);
152 barcon(startHist:endHist,c.Infl{:,1:end-2});
153 grid on;
154 title('Contributions of shocks');
155
156 edescript = get(mest,'eDescript');
157 legend(edescript{:},'location','northWest');
```

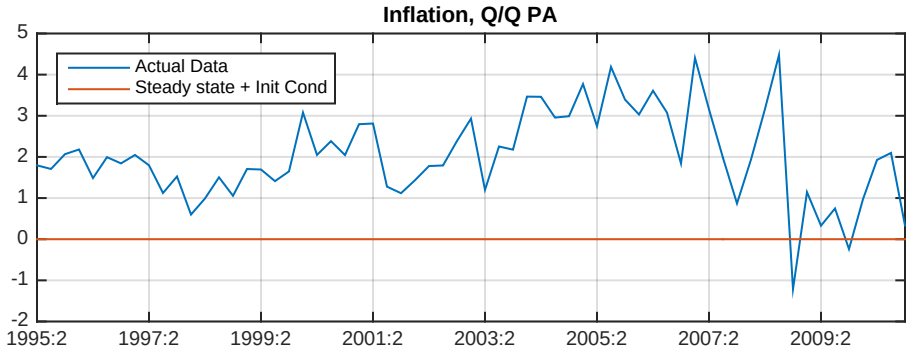


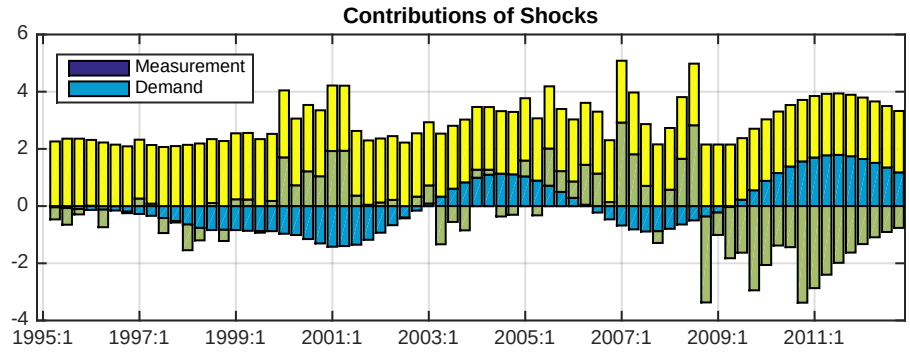
9 Plot Grouped Contributions

Use a `grouping` object to define groups of shocks whose contributions will be added together and plotted as one category. Run `eval` to create a new database with the contributions grouped accordingly [4]. Otherwise, the information content of this figure window is the same as the previous one.

```
167 g = grouping(mest,'shock');
```

```
168 g = addgroup(g,'Measurement','M.*');
169 g = addgroup(g,'Demand','Ey,Er');
170 g = addgroup(g,'Supply','Ep,Ea,Ew');
171
172 [cg,leg] = eval(g,c); [4]
173
174 figure();
175
176 subplot(2,1,1);
177 plot(startHist:endHist,[s.Infl,c.Infl{:,end}]);
178 grid on;
179 title('Inflation, Q/Q PA');
180 legend('Actual Data','Steady state + Init Cond','location','northWest');
181
182 subplot(2,1,2);
183 conbar(cg.Infl{:,1:end-1});
184 grid on;
185 title('Contributions of Shocks');
186 legend(leg{1:end-1},'location','northWest');
```





10 Save Output Data for Future Use

Save the output database `f` from the basic run of the filter in a mat-file (binary file) for future use.

```
193 save filter_hist_data.mat f;
```

11 Help on IRIS Functions Used in This Files

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

```
help model/filter
help model/simulate
help tseries/conbar
help tseries/plotpred
help grfun/movetosubplot
help data/dbfun
```