LOCPROJ:
Stata Module to Estimate and Draw Local Projections

Alfonso Ugarte-Ruiz

Stata Conference, Stanford, 2023

- You can access the new commands here: https://ideas.repec.org/c/boc/bocode/s459204.html
- Or you can install by entering **ssc install locproj**

Local Projection is one of the fastest growing topics in the economic literature

- transmission of monetary policy/interventions
- impact of the introduction of banking regulation on credit markets
- impact of labour and product market reforms on economic growth
- transmission of monetary policy shocks on daily consumption, corporate sales and employment
- impact of financial crises on macroeconomic dynamics
- impact of universal banking on macroeconomic dynamics
- impact of climate change on inflation and growth dynamics
- non-linear exchange rate pass-through in emerging markets
- effects of carbon taxes on economic activity and CO2 emissions
- effects of fiscal consolidation on activity
- effects of inflation on public finances
- effects US monetary policy shocks on EMs spreads and capital flows
- effects of global shocks on EMs financial conditions
- response of real GDP per capita growth to global and idiosyncratic temperature shocks
- ... and so forth

https://sites.google.com/site/oscarjorda/home/local-projections

local projections

STATA code

- simple time series example: lp_example.do
- LPIV example: lpiv_example.do; .dta files needed: lpiv_15Mar2022.dta and RR_monetary_shock_quarterly.dta
- panel data example: lp_example_panel.do
- code for **inverse propensity score weighted - regression augmented LPs** used in:

  The Time for Austerity: Estimating the Average Treatment Effect of Fiscal Policy

  The Economic Journal, 2016, 126(590): 219-255
- code for **stratified LPs** used in:

  Sovereigns versus Banks: Credit, Crises, and Consequences

  Journal of the European Economic Association, 2016, 14(1): 45-79

R code: package *lpirfs*

- Code prepared by Philipp Adämmer. Here is a description of its capabilities

Silvia Miranda Agrippino's MATLAB code for "The Transmission of Monetary Policy Shocks" (with G. Ricco).

- typical and Bayesian local projections available here

replication code for "Impulse responses by smooth local projections" by Barnichon and Brownlees

- code from the Review and Economic Statistics publication available here

Bayesian local projections

- Amaze Lusompa has written a great paper on how to think about Bayesian local projections. Check his research here

inference

- Mikkel Plagborg-Møller is doing great research on local projections. I particularly like his paper with José Luis Montiel Olea. The paper is available here

references

main reference:

- Estimation and Inference of Impulse Responses by Local Projections

  American Economic Review 95(1), March 2005, 161-182

  list of citations from scholar.google.com with some interesting applications

how to use IV methods for identification

- The effects of quasi-random monetary experiments

  Journal of Monetary Economics, 112: 22-40

Usual steps for estimating a LP (most frequently using Stata)

- Define the horizon (h) and the type of transformation of the dependent variable, usually:
  - Levels or logarithms
  - Differences
  - Cumulative differences
  - Differences in logarithm or percentage differences
  - Cumulative differences in logs or cumulative percentage differences

- Construct a loop that runs the h-steps regressions using the desired estimation method and specification

- Extract and save the *h-step* estimated coefficient and standard error of the "shock/impulse" variable (or possibly more than one coefficient in nonlinear cases)

- Construct confidence intervals

- Graph the IRF

The new command LOCPROJ:

- Generates **temporary variables with the necessary transformations** of the response variable in order to estimate the IRF in the desired transformation option

- Allows choosing **different estimation methods** for both time series and panel data, including some **instrumental variables** methods

- The **shock** could be composed of **more than one variable** or variables, allowing to estimate responses to **linear combinations** of variables, including interactions with categorical or continuous variables.

- Allows the ***use of marginal effects*** instead of regression coefficients, which is highly convenient when the ***response variable is binary*** and we want to estimate the response as a probability.

- Allows different options regarding the horizon and the response starting period. For instance, it automatically adjusts the horizon in the case that the shock variable(s) is included with a lag and no contemporaneous term, which is convenient when playing with the ordering in a Cholesky decomposition

The new command LOCPROJ:

- The **options allow defining the desired specification in a fully automatic or in a more explicit way**, with many alternatives in between.

- If the user chooses the automatic specification, the syntax is very close to a typical regression command in Stata, with the only restriction that it interprets the variable that corresponds to the shock as the one just after the dependent variable or its lagged terms, and only that one variable represents the shock.

Automatic Specification (Shock and Lags)

```
locproj depvar shock [depvar lagged-terms] [shock lagged-terms] [controls] [if] [in] , [ hor(numlist integer) lcs(string) lcopt(string) fcontrols(varlist)
      instr(string) transf(string) met(string) model_options hopt(string) conf(numlist integer) noisily saveirf irfname(string) fact(real) margins mrfvar(varlist)
      mrpredict(string) mropt(string) nograph title(string) label(string) zero lcolor(string) ttitle(string) grname(string) grsave(string) as(string) gropt(string) ]
```

- Alternatively, the user can choose to explicitly define the shock variable (or variables), the number of lags of the shock, the number of lags of the dependent variable, and the control variables.

```
Explicit Specification (Shock and Lags)

    locproj depvar [if] [in] , [ hor(numlist integer) shock(varlist) controls(varlist) ylags(integer) slags(integer) lcs(string) lcopt(string) fcontrols(varlist)
        instr(string) transf(string) met(string) model_options hopt(string) conf(numlist integer) noisily saveirf irfname(string) fact(real) margins mrfvar(varlist)
        mrpredict(string) mropt(string) nograph title(string) label(string) zero lcolor(string) ttitle(string) grname(string) grsave(string) as(string) gropt(string) ]
```

- The explicit option is recommended when the shock should include more than one variable, for instance, an additional non-linear term, or an interaction with another variable.

- REMARK: If the shock includes an interaction with binary/categorical variables, then we must use the option **lcs()** or **margins()**

The new command LPGRAPH:

- **lpgraph** is a new post-estimation command that plots together the results of previously estimated IRFs of more than one model into one graph. The graph can include up until 3 IRFs.

- It can also generate 3 separate IRF graphs and combine the 3 of them in the same fashion as the command grah combine.

- The first option is convenient when we want to have a graph in which we compare the magnitude of the different IRFs, since they are going to share the same axis.

- The second option is more convenient when you want to create new IRF graphs of previously estimated models.

Syntax

```
lpgraph irfname1 irfname2 irfname3 , [ hor(numlist integer) zero title(string) ttitle(string) ytitle(string) separate ti1(string) ti2(string) ti3(string)
    lab1(string) lab2(string) lab3(string) lc1(string) lc2(string) lc3(string) nolegend grname(string) grsave(string) as(string) other_options]
```

Example 1. Use of locproj to replicate the IRFs in the "simple time series example”:
(lp-example do-file in Jordà website)

Example 1.1 Defining the basic specific options

```
. use AED_INTERESTRATES.dta
(Data for A. Colin Cameron (2022), ANALYSIS OF ECONOMIC DATA, Amazon)

. locproj gs10 gs1
```

"Automatic"

```
Impulse Response Function
```

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.83588	0.01328	0.80977	0.86199
1	0.82558	0.01368	0.79868	0.85248
2	0.81363	0.01434	0.78545	0.84182
3	0.80232	0.01503	0.77276	0.83188
4	0.79088	0.01569	0.76004	0.82172
5	0.78001	0.01628	0.74799	0.81202

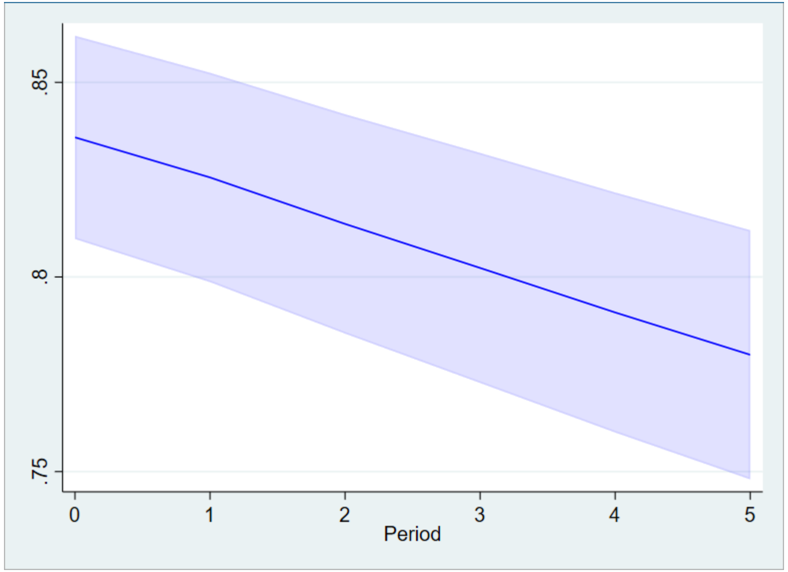
```
. locproj gs10, shock(gs1)
```

"Explicit"

```
Impulse Response Function
```

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.83588	0.01328	0.80977	0.86199
1	0.82558	0.01368	0.79868	0.85248
2	0.81363	0.01434	0.78545	0.84182
3	0.80232	0.01503	0.77276	0.83188
4	0.79088	0.01569	0.76004	0.82172
5	0.78001	0.01628	0.74799	0.81202

Dependent variable: gs10
Shock variable: gs1



Example 1.1 Defining the basic specific options

```
locproj gs10 l(0/4).gs1 l(1/3).gs10, hor(12)
```

"Automatic"

Impulse Response Function

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.74857	0.03402	0.68167	0.81546
1	1.03737	0.06900	0.90170	1.17304
2	1.09785	0.09559	0.90991	1.28579
3	1.11157	0.11778	0.87999	1.34315
4	1.04859	0.13755	0.77813	1.31905
5	1.06629	0.15214	0.76715	1.36542
6	0.99839	0.16348	0.67695	1.31983
7	0.98291	0.17251	0.64370	1.32211
8	1.02195	0.18117	0.66573	1.37818
9	1.02482	0.18916	0.65287	1.39677
10	1.01505	0.19740	0.62691	1.40320
11	1.05432	0.20508	0.65105	1.45759
12	0.99168	0.21280	0.57324	1.41012

.

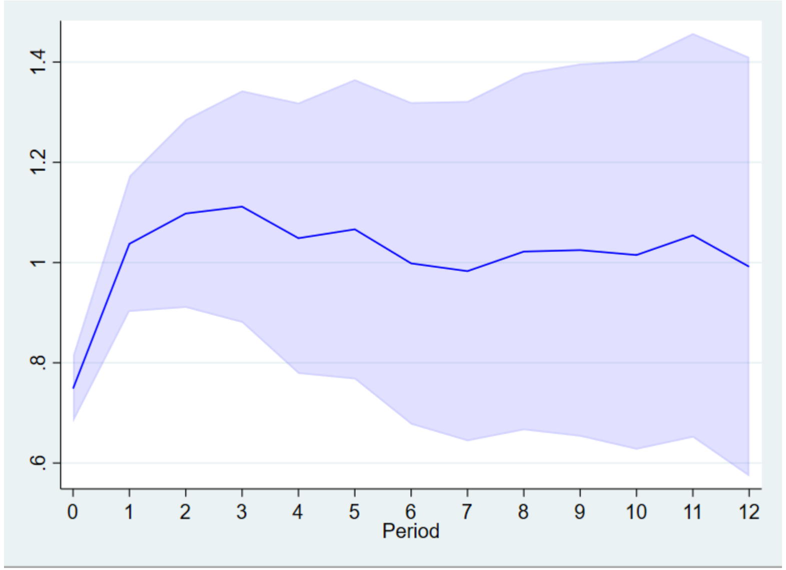
```
locproj gs10, shock(gs1) ylags(3) slags(3) hor(12)
```

"Explicit"

Impulse Response Function

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.74857	0.03402	0.68167	0.81546
1	1.03737	0.06900	0.90170	1.17304
2	1.09785	0.09559	0.90991	1.28579
3	1.11157	0.11778	0.87999	1.34315
4	1.04859	0.13755	0.77813	1.31905
5	1.06629	0.15214	0.76715	1.36542
6	0.99839	0.16348	0.67695	1.31983
7	0.98291	0.17251	0.64370	1.32211
8	1.02195	0.18117	0.66573	1.37818
9	1.02482	0.18916	0.65287	1.39677
10	1.01505	0.19740	0.62691	1.40320
11	1.05432	0.20508	0.65105	1.45759
12	0.99168	0.21280	0.57324	1.41012

Dependent variable: gs10
Shock variable: gs1
Lags of the dependent variable: 3
Lags of the shock: 4
Horizon: 12



Example 1.2 A simple nonlinear example

```
gen gs1_2 = gs1^2
```

Generate a quadratic term

```
locproj gs10, shock(gs1 gs1_2) ylags(3) slags(4) hor(12)
locproj gs10, shock(gs1 c.gs1#c.gs1) ylags(3) slags(4) hor(12)
```

Impulse Response Function

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.95404	0.07826	0.80015	1.10792
1	1.27418	0.16042	0.95875	1.58961
2	1.18387	0.22234	0.74669	1.62105
3	1.07331	0.27405	0.53445	1.61217
4	1.19139	0.31836	0.56540	1.81739
5	1.19293	0.35138	0.50200	1.88385
6	1.05068	0.37846	0.30650	1.79485
7	0.94413	0.40053	0.15653	1.73172
8	0.91296	0.42188	0.08338	1.74254
9	1.09478	0.44139	0.22683	1.96273
10	1.20998	0.46128	0.30293	2.11704
11	1.44678	0.48019	0.50252	2.39103
12	1.46988	0.49942	0.48780	2.45197

Example 1.3 Estimation method options

```
. locproj gs10 l(0/4).gs1 l(1/3).gs10, h(12) met(newey) hopt(lag)
```

Impulse Response Function

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.74857	0.04285	0.66432	0.83281
1	1.03737	0.07975	0.88056	1.19418
2	1.09785	0.11543	0.87090	1.32481
3	1.11157	0.14244	0.83151	1.39163
4	1.04859	0.15516	0.74350	1.35368
5	1.06629	0.16179	0.74818	1.38440
6	0.99839	0.15904	0.68567	1.31110
7	0.98291	0.15742	0.67338	1.29243
8	1.02195	0.16191	0.70359	1.34032
9	1.02482	0.16020	0.70982	1.33982
10	1.01505	0.16452	0.69156	1.33855
11	1.05432	0.20472	0.65176	1.45687
12	0.99168	0.20742	0.58381	1.39955

The Jordà example requires using Newey-West as the estimation method, which consequently requires specifying that the option "**lag**" in the Newey-West command should depend on the horizon of the IRF, using the option "**hopt()**"

Example 1.4 Displaying all the regression outputs

```
locproj gs10 l(0/4).gs1 l(1/3).gs10, h(12) met(newey) hopt(lag) noisily
```

gs10_h0

	Coefficient	Std.Err.	t-stat	p-value	min95	max95
gs1	.7485668	.0428475	17.47	0.000	.6643218	.8328119
L.gs1	-.9936843	.0832385	-11.94	0.000	-1.157345	-.830024
L2.gs1	.3507535	.1007337	3.48	0.001	.1526949	.5488121
L3.gs1	-.1703881	.0884316	-1.93	0.055	-.3442588	.0034827
L4.gs1	.0880813	.041456	2.12	0.034	.0065722	.1695905
L.gs10	1.258968	.0547217	23.01	0.000	1.151376	1.366559
L2.gs10	-.3845461	.086571	-4.44	0.000	-.5547585	-.2143337
L3.gs10	.0981334	.0563079	1.74	0.082	-.0125771	.2088438
_cons	.0520799	.0257988	2.02	0.044	.0013553	.1028045

gs10_h1

	Coefficient	Std.Err.	t-stat	p-value	min95	max95
gs1	1.037371	.0797519	13.01	0.000	.880565	1.194178
L.gs1	-1.165679	.2008635	-5.80	0.000	-1.560612	-.7707462
L2.gs1	.1143121	.1821842	0.63	0.531	-.2438944	.4725186
L3.gs1	-.0742497	.2134807	-0.35	0.728	-.4939906	.3454911
L4.gs1	.134118	.1065439	1.26	0.209	-.0753661	.3436021
L.gs10	1.033223	.1055747	9.79	0.000	.8256441	1.240801
L2.gs10	-.0484837	.1362856	-0.36	0.722	-.3164454	.219478
L3.gs10	-.0480161	.1072194	-0.45	0.655	-.2588284	.1627963
_cons	.1438161	.0624141	2.30	0.022	.0210989	.2665333

gs10_h2

	Coefficient	Std.Err.	t-stat	p-value	min95	max95
gs1	1.097853	.1154298	9.51	0.000	.8708955	1.32481
L.gs1	-1.232321	.2375826	-5.19	0.000	-1.699454	-.7651875
L2.gs1	.0256929	.2459752	0.10	0.917	-.4579421	.5093278
L3.gs1	.0575156	.2207032	0.26	0.795	-.3764295	.4914608
L4.gs1	.108708	.1222379	0.89	0.374	-.1316355	.3490514
L.gs10	1.029004	.1554225	6.62	0.000	.7234132	1.334594
L2.gs10	-.0096063	.181521	-0.05	0.958	-.3665116	.3472991
L3.gs10	-.1117394	.1435774	-0.78	0.437	-.3940403	.1705615
_cons	.2395626	.1078543	2.22	0.027	.0275002	.451625

If we want to **take a look at the regression output** for each one of the horizons of the IRF we can use the option noisily. The regression outputs displayed are not the direct outputs from whatever estimation method we are using, but a simplified output table.

The reason for this is that locproj uses temporary variables whose given names do not have any meaning and would be difficult to understand. locproj generates a new output table with variable names related to the variable list defined by the user.

Example 1.5 Transformation options

`tr(level)` **Levels**: forecast h periods ahead for each horizon of the IRF, i.e. Y_{t+h} with $h = 1 \dots hor$. It is the default option in case no transformation is specified. When the option ylags() is specified, it includes lags of the variable in levels, i.e. Y_{t-l} with $l = 1 \dots ylags$

`tr(diff)` **Differences**: forecasts in simple differences, i.e. $Y_{t+h} - Y_{t+h-1}$ with $h = 1 \dots hor$. When the option ylags() is specified, it includes lags of the variable in differences, i.e. $Y_t - Y_{t-l}$ with $l = 1 \dots ylags$

`tr(cmlt)` **Cumulative Differences**: forecasts in cumulative differences, i.e. $Y_{t+h} - Y_{t-1}$ with $h = 1 \dots hor$. When the option ylags() is specified, it includes lags of the variable in differences, i.e. $Y_t - Y_{t-l}$ with $l = 1 \dots ylags$

`tr(logs)` **Logs**: forecasts of the logarithm, i.e. $\ln(Y_{t+h})$ with $h = 1 \dots hor$. When the option ylags() is specified, it includes lags of the logarithm of the variable, i.e. $\ln(Y_{t-l})$ with $l = 1 \dots ylags$

`tr(logs diff)` **Log-differences**: forecasts in differences of its natural logarithm, i.e. $\ln(Y_{t+h}) - \ln(Y_{t+h-1})$ with $h = 1 \dots hor$. When the option ylags() is specified, it includes lags of the variable in differences, i.e. $\ln(Y_t) - \ln(Y_{t-l})$ with $l = 1 \dots ylags$

`tr(logs cmlt)` **Cumulative Log-differences**: forecasts in cumulative differences of natural logarithm, i.e. $\ln(Y_{t+h}) - \ln(Y_{t-1})$ with $h = 1 \dots hor$. When the option ylags() is specified, it includes lags of the variable in differences, i.e. $\ln(Y_t) - \ln(Y_{t-l})$ with $l = 1 \dots ylags$

Example 1.6 Saving the IRF results into new variables

```
save irfname(newirf)
```

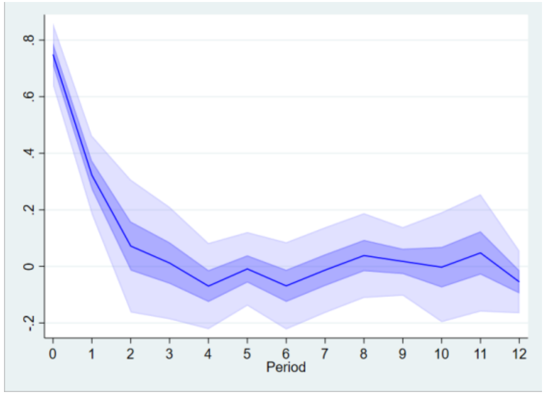
	newirf	newirf_se	newirf_up	newirf_lo
1	.7494722	.0426586	.8333465	.6655979
2	.3226883	.0540223	.4289065	.21647
3	.0721858	.0907106	.2505419	-.1061703
4	.0121001	.0767404	.162989	-.1387889
5	-.069358	.0588169	.0462903	-.1850063
6	-.008624	.0502905	.0902601	-.1075081
7	-.0685408	.0595396	.0485305	-.1856121
8	-.0128754	.0584877	.1021287	-.1278794
9	.0385645	.0578421	.1522999	-.075171
10	.0179519	.046774	.1099249	-.0740211
11	-.0025995	.0749954	.1448673	-.1500662
12	.0479875	.0800473	.2053894	-.1094144
13	-.0547293	.0427494	.0293322	-.1387908
14	.	.	.	.
15	.	.	.	.
16	.	.	.	.
17	.	.	.	.
18	.	.	.	.
19	.	.	.	.
20	.	.	.	.

Example 1.7 Confidence levels

```
conf(66 99)
```

Impulse Response Function

	IRF	Std.Err.	IRF LO~66	IRF UP 66	IRF LO~99	IRF UP 99
0	0.74947	0.04266	0.70872	0.79023	0.63904	0.85990
1	0.32269	0.05402	0.27108	0.37430	0.18284	0.46254
2	0.07219	0.09071	-0.01448	0.15885	-0.16265	0.30702
3	0.01210	0.07674	-0.06122	0.08542	-0.18657	0.21077
4	-0.06936	0.05882	-0.12555	-0.01317	-0.22163	0.08291
5	-0.00862	0.05029	-0.05667	0.03942	-0.13882	0.12157
6	-0.06854	0.05954	-0.12542	-0.01166	-0.22268	0.08560
7	-0.01288	0.05849	-0.06875	0.04300	-0.16430	0.13855
8	0.03856	0.05784	-0.01670	0.09383	-0.11119	0.18832
9	0.01795	0.04677	-0.02674	0.06264	-0.10315	0.13905
10	-0.00260	0.07500	-0.07425	0.06905	-0.19677	0.19157
11	0.04799	0.08005	-0.02849	0.12446	-0.15926	0.25524
12	-0.05473	0.04275	-0.09557	-0.01389	-0.16541	0.05596



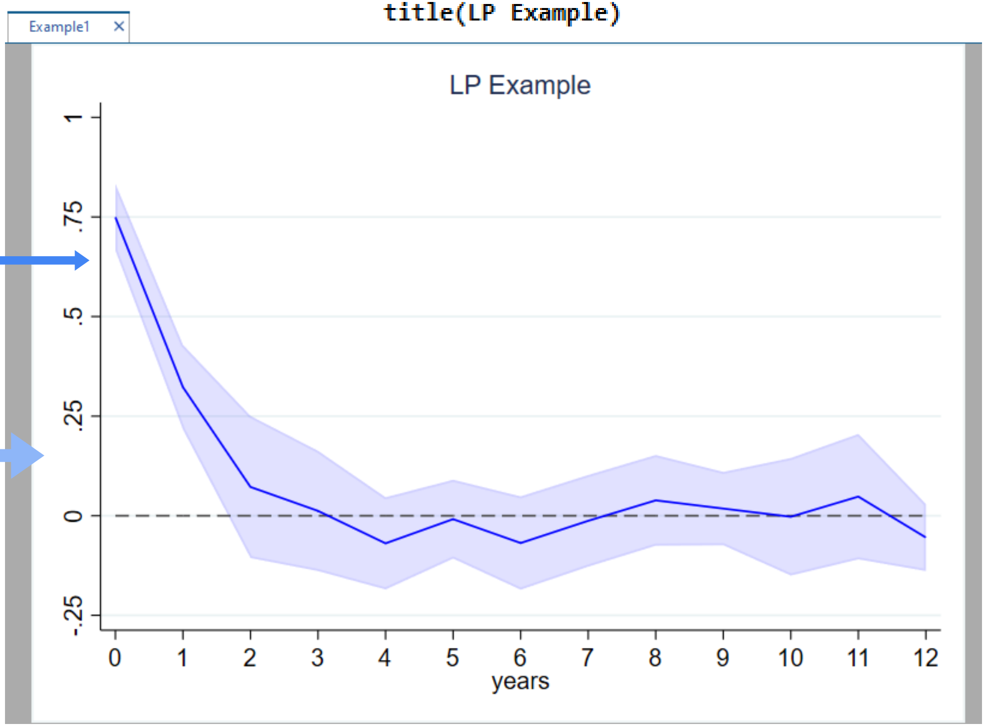
Example 1.8 Graph options

`grname(Example1)`

`title(LP Example)`

`gropt(graphregion(fcolor(white)) ylabel(-0.25(0.25)1))`

`zero`



`ttitle(years)`

`grsave(C:\Users\e017837\example1) as(png)`

```
     11 |   0.04799    0.08005   -0.10941     0.20539
     12 |  -0.05473    0.04275   -0.13879     0.02933
file C:\Users\e017837\example1.png saved as PNG format
```

Example 2. Replication of the IRFs in the "panel data example: lp_example_panel" do-file in Jordà website:

Short-run Cholesky identification

3-variable system:

$$y_t = (x_t, \pi_t, i_t)' \longrightarrow u_t = (u_{x,t}, u_{\pi,t}, u_{i,t})'$$

Residuals:

x_t: GDP growth
π_t: Inflation
i_t: FED funds interest rate

Exogenous shocks: Cholesky:

$$\begin{bmatrix} \varepsilon_{x,t} \\ \varepsilon_{\pi,t} \\ \varepsilon_{i,t} \end{bmatrix} = \begin{bmatrix} R_{x,x} & 0 & 0 \\ R_{\pi,x} & R_{\pi,\pi} & 0 \\ R_{i,x} & R_{i,\pi} & R_{i,i} \end{bmatrix} \begin{bmatrix} u_{x,t} \\ u_{\pi,t} \\ u_{i,t} \end{bmatrix}$$

- GDP growth is slow to respond, therefore, it responds contemporaneously only to GDP shocks, and with a lag to inflation and interest rate shocks
- Inflation responds contemporaneously to GDP shocks, but with a lag to interest rates
- Policy rate responds contemporaneously to all other shocks (all information)

We first estimate the response of real GDP to a 1pp shock to the real short-term interest rate (with one period lag). The estimation method is **xtreg**, using the "fixed-effect" estimator and a cluster-robust covariance matrix. The confidence level is 90%. We define the values of the y-axis, they y-axis title, the x-axis label and the title of the graph:

$$\Delta GDPR_{i,t+h} = \beta_h \Delta STIR_{i,t-1} + \cdots + \alpha_h \Delta CPIR_{i,t-1} + \cdots + \rho_h \Delta GDPR_{i,t-1} + \cdots$$

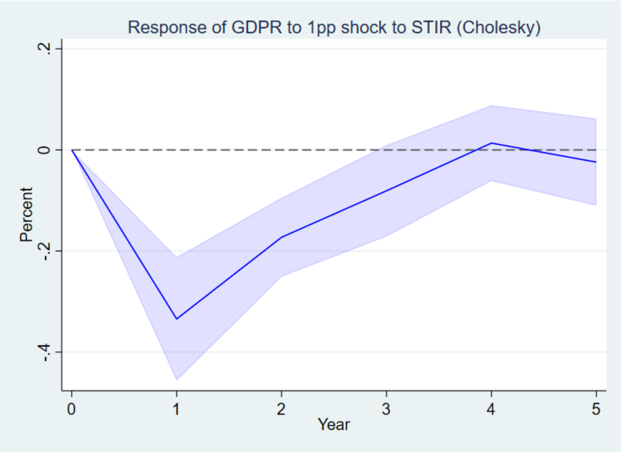
Using the transformation option transf() and the fully explicit options:

```
. locproj lgdpr, s(l.d.stir) c(l(1/3).d.lcpi) tr(diff) h(4) yl(3) sl(3) fe cluster(iso) z conf(90) gropt(ylabel(-0.4(0.2)0.2) ytitle(Percent)) tti(Year)
    title(Response of GDPR to 1pp shock to STIR (Cholesky))
```

The shock variable is included with a lag and no contemporaneous term (**L.D.STIR**), and thus, the response at $hor = 0$ is equal to 0 and an extra period is added to the final horizon period. (**GDPR** responds with a lag to all shocks)

Impulse Response Function

	IRF	Std.Err.	IRF LOW	IRF UP
0	**0.00000**	**0.00000**	**0.00000**	**0.00000**
1	**-0.33453**	**0.07037**	**-0.45694**	**-0.21212**
2	**-0.17288**	**0.04494**	**-0.25106**	**-0.09469**
3	**-0.08085**	**0.05201**	**-0.17133**	**0.00963**
4	**0.01355**	**0.04310**	**-0.06143**	**0.08852**
5	**-0.02416**	**0.04971**	**-0.11064**	**0.06233**



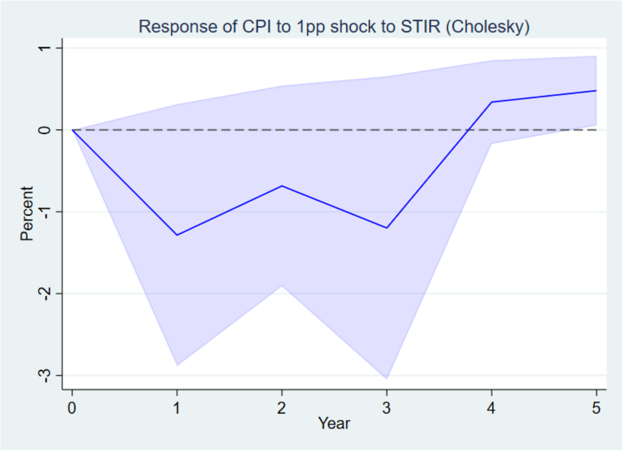
Alternatively, in the next example, when estimating the response of CPI to the short-term interest rate, we just need to change the dependent variable to CPI instead of GDPR. We also need to change the lags of the control variable, which now is the GDPR, from (0/3) to (1/3) in order to specify a Cholesky decomposition (CPI respond contemporaneously to GDPR, with a lag to STIR)

$$\Delta CPI_{i,t+h} = \beta_h \Delta STIR_{i,t-1} + \cdots + \alpha_h \Delta GDPR_{i,t} + \cdots + \rho_h \Delta CPI_{i,t-1} + \cdots$$

```
. locproj lcpi, s(l.d.stir) c(l(0/3).d.lgdpr) tr(diff) h(4) yl(3) sl(3) fe cluster(iso) z conf(90) gropt(ylabel(-0.4(0.2)0.2) ytitle(Percent)) tti(Year)
    title(Response of CPI to 1pp shock to STIR (Cholesky))
```

Impulse Response Function

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.00000	0.00000	0.00000	0.00000
1	-1.28506	0.91926	-2.88421	0.31409
2	-0.68441	0.70459	-1.91012	0.54131
3	-1.19804	1.06514	-3.05096	0.65488
4	0.33997	0.29382	-0.17116	0.85109
5	0.47954	0.24467	0.05392	0.90517



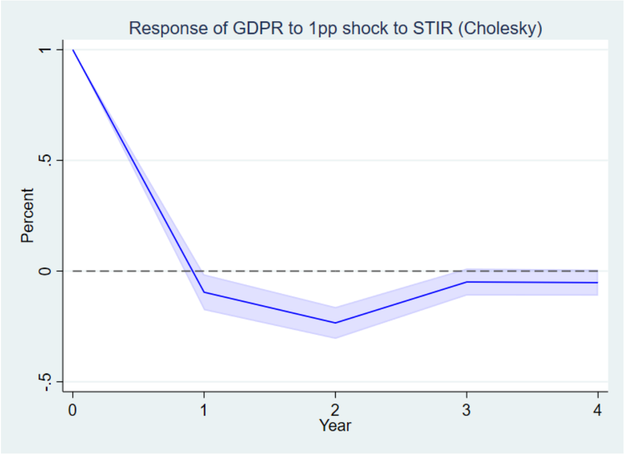
Finally, for estimating the shock to the real interest rate, the shock variable is the same as the dependent variable (STIR), so it is more convenient to use an explicit option for the shock and its lags. Moreover, the first response at ($hor = 0$) is equal to 1 since the response variable and the shock are the same at that horizon period. (STIR responds contemporaneously to all variables)

$$\Delta STIR_{i,t+h} = \beta_h \Delta STIR_{i,t} + \cdots + \theta_h \Delta GDPR_{i,t} + \cdots + \alpha_h \Delta CPIR_{i,t} + \cdots + \rho_h \Delta STIR_{i,t-1} + \cdots$$

```
. locproj d.stir l(0/3).d.lcpi l(0/3).d.lgdpr, s(d.stir) sl(3) h(4) fe cluster(iso) z conf(90) gropt(ylabel(-0.4(0.2)0.2) ytitle(Percent)) tti(Year) title(Response
   of GDPR to 1pp shock to STIR (Cholesky))
```

Impulse Response Function

	IRF	Std.Err.	IRF LOW	IRF UP
0	1.00000	0.00000	1.00000	1.00000
1	-0.09544	0.04657	-0.17646	-0.01442
2	-0.23414	0.04139	-0.30614	-0.16214
3	-0.04929	0.03507	-0.11029	0.01171
4	-0.05203	0.03375	-0.11073	0.00668



Example 3. Control variables that depend on the horizon (leads)

We want to include a dummy variable for the year 2009 because we want to control for the effects of the Global Financial Crisis on the GDP growth rate

D.lgdpr_h0

	Coefficient	Std.Err.	t-stat	p-value
LD.stir	-.3372212	.0723391	-4.66	0.000
L2D.stir	-.0528923	.0493663	-1.07	0.299
L3D.stir	-.1240029	.0397386	-3.12	0.006
L4D.stir	-.0264053	.0294557	-0.90	0.383
L.D.lgdpr	.2742065	.0685837	4.00	0.001
L2.D.lgdpr	-.0041254	.0447188	-0.09	0.928
L3.D.lgdpr	.0248906	.038179	0.65	0.523
LD.lcpi	-.0614964	.0284221	-2.16	0.045
L2D.lcpi	.0648536	.0312744	2.07	0.054
L3D.lcpi	.0111313	.0134321	0.83	0.419
2009.year	-5.715456	.4853945	-11.77	0.000
_cons	1.782528	.1529091	11.66	0.000

D.lgdpr_h1

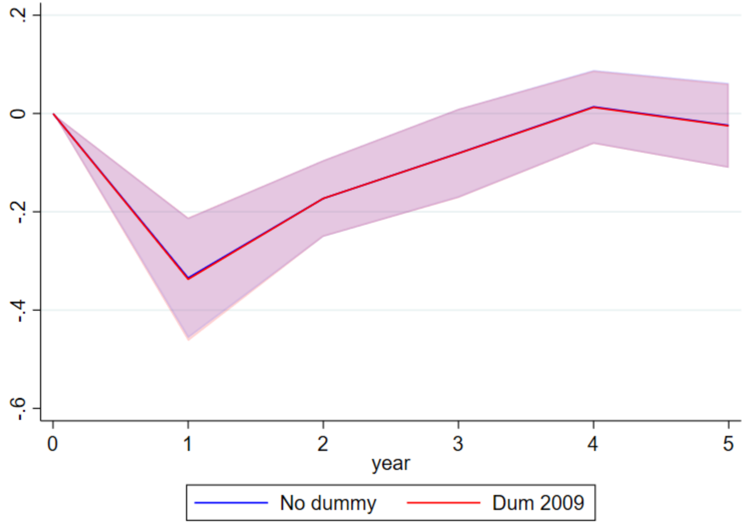
	Coefficient	Std.Err.	t-stat	p-value
LD.stir	-.1728575	.0448718	-3.85	0.001
L2D.stir	-.1012361	.0616746	-1.64	0.119
L3D.stir	-.0866317	.0275724	-3.14	0.006
L4D.stir	-.1065248	.0368599	-2.89	0.010
L.D.lgdpr	.0463748	.0488577	0.95	0.356
L2.D.lgdpr	.0113527	.0308534	0.37	0.717
L3.D.lgdpr	.020195	.0391806	0.52	0.613
LD.lcpi	.0168397	.0160015	1.05	0.307
L2D.lcpi	-.0200341	.0321517	-0.62	0.541
L3D.lcpi	.061711	.0309	2.00	0.062
2009.year	.0460053	.3619759	0.13	0.900
_cons	2.140613	.177486	12.06	0.000

D.lgdpr_h2

	Coefficient	Std.Err.	t-stat	p-value
LD.stir	-.0812572	.0522432	-1.56	0.138
L2D.stir	-.0097422	.0416814	-0.23	0.818
L3D.stir	-.0639559	.0425938	-1.50	0.152
L4D.stir	.0288651	.0558856	0.52	0.612
L.D.lgdpr	.0197249	.0218317	0.90	0.379
L2.D.lgdpr	.0387611	.0332441	1.17	0.260
L3.D.lgdpr	-.0889853	.0414196	-2.15	0.046
LD.lcpi	-.0084503	.0243319	-0.35	0.733
L2D.lcpi	.029928	.0139396	2.15	0.047
L3D.lcpi	.0507179	.0316851	1.60	0.128
2009.year	-.8622562	.4062844	-2.12	0.049

```
locproj lgdpr l.d.stir l(1/3).d.lcpi 2009.year, h(4) yl(3) sl(3) fe cluster(iso) tr(diff)
```

The effect of the dummy changes widely at every horizon and it does not change the IRF:



With LOCPROJ we can include the dummy in a list of control variables that enter with a lead equal to the horizon (*hor=0,…,h*)

D.lgdpr_h0

	Coefficient	Std.Err.	t-stat	p-value
LD.stir	-.3372212	.0723391	-4.66	0.000
L2D.stir	-.0528923	.0493663	-1.07	0.299
L3D.stir	-.1240029	.0397386	-3.12	0.006
L4D.stir	-.0264053	.0294557	-0.90	0.383
L.D.lgdpr	.2742065	.0685837	4.00	0.001
L2.D.lgdpr	-.0041254	.0447188	-0.09	0.928
L3.D.lgdpr	.0248906	.038179	0.65	0.523
LD.lcpi	-.0614964	.0284221	-2.16	0.045
L2D.lcpi	.0648536	.0312744	2.07	0.054
L3D.lcpi	.0111313	.0134321	0.83	0.419
2009.year	-5.715456	.4853945	-11.77	0.000
_cons	1.782528	.1529091	11.66	0.000

D.lgdpr_h1

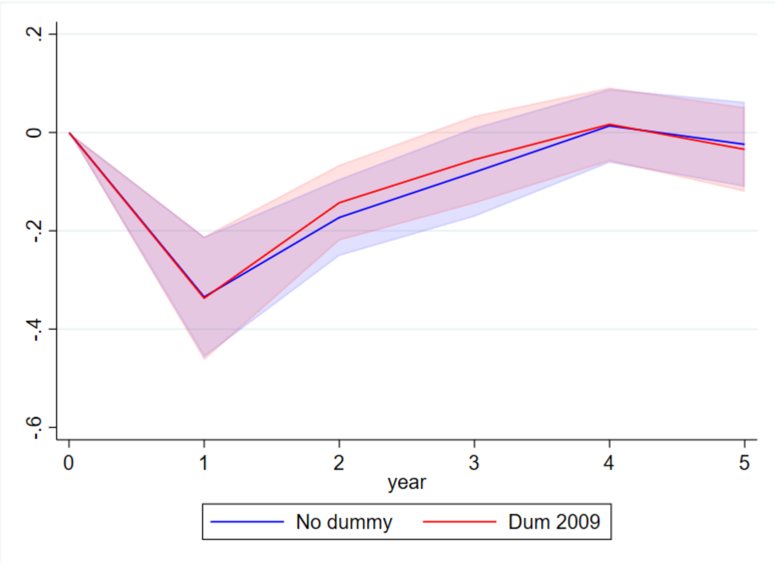
	Coefficient	Std.Err.	t-stat	p-value
LD.stir	-.1428877	.0443136	-3.22	0.005
L2D.stir	-.0716867	.0599295	-1.20	0.248
L3D.stir	-.0700951	.0267016	-2.63	0.018
L4D.stir	-.1020363	.0368719	-2.77	0.013
L.D.lgdpr	.0463441	.0489125	0.95	0.357
L2.D.lgdpr	.010346	.0307829	0.34	0.741
L3.D.lgdpr	.0175054	.0393737	0.44	0.662
LD.lcpi	.0157281	.0161466	0.97	0.344
L2D.lcpi	-.020944	.0323699	-0.65	0.526
L3D.lcpi	.0605461	.0309429	1.96	0.067
2009F.year	-6.202455	.4850499	-12.79	0.000
_cons	2.214083	.1766256	12.54	0.000

D.lgdpr_h2

	Coefficient	Std.Err.	t-stat	p-value
LD.stir	-.0552952	.0512101	-1.08	0.295
L2D.stir	-.0062113	.0419946	-0.15	0.884
L3D.stir	-.0664558	.0432433	-1.54	0.143
L4D.stir	.011575	.0554076	0.21	0.837
L.D.lgdpr	.0205185	.0218734	0.94	0.361
L2.D.lgdpr	.0363792	.033345	1.09	0.290
L3.D.lgdpr	-.089486	.0415394	-2.15	0.046
LD.lcpi	-.009229	.0242664	-0.38	0.708
L2D.lcpi	.0295026	.0139946	2.11	0.050
L3D.lcpi	.0500519	.0317585	1.58	0.133
2009F2.year	-6.279031	.5094145	-12.33	0.000

```
locproj lgdpr l.d.stir l(1/3).d.lcpi, fcontrols(2009.year) h(4) yl(3) sl(3) fe cluster(iso)
tr(diff)
```

Now we do get a different IRF:



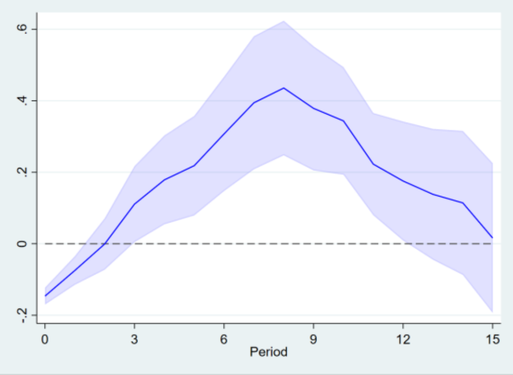
Example 4. Replication of the IRFs in the "LPIV example: lpiv_example"" do-file in Jordà website & use of command LPGRAPH

For estimating the IRF using OLS we need to take into consideration that in the example, the response horizon starts at (hor = 1). However, for exactly replicating the example, we need to use the one-step ahead forecast of UNRATE as the dependent variable instead of UNRATE. We also use Newey-West as the variance-covariance estimation method, which requires defining the option lag() in Newey-West, and thus, we need to use the option hopt() in our locproj command:

```
locproj f.UNRATE DFF l(1/4).DFF l(1/4).UNRATE, h(0/15) hopt(lag) m(newey) z save irfn(ols)
```

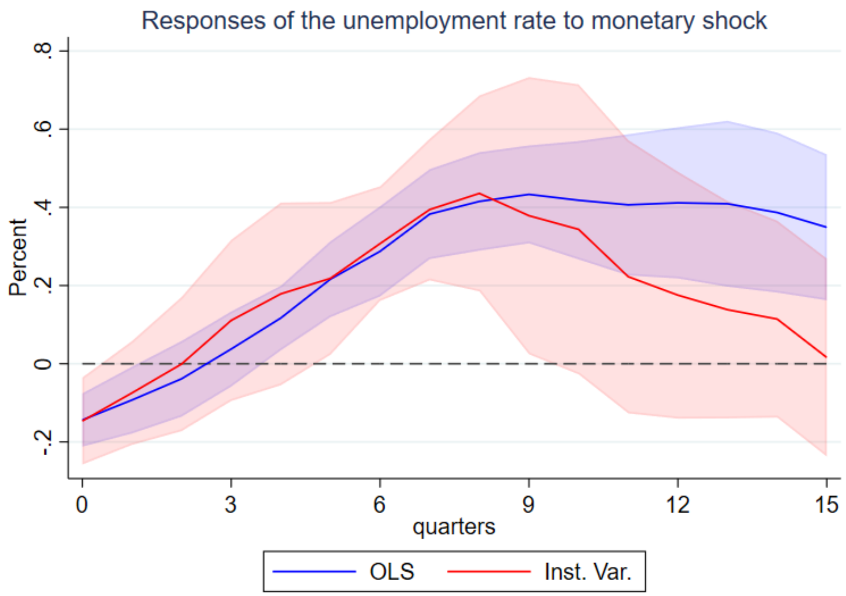
For replicating the instrumental variable IRF approach, we use the option **met()** specifying that the method is **ivregress gmm**. Notice that in this case the **met()** option should include the **gmm** sub-method. Additionally, we need to define the instrument(s) for the shock, which in this case corresponds to the variable **resid_full**. Moreover, we define as an estimation method option the variance-covariance estimator HAC Newest:

```
locproj f.UNRATE DFF l(1/4).DFF l(1/4).UNRATE, h(0/15) met(ivregress gmm) instr(resid_full) vce(hac nwest) z save irfn(ivr)
```



Now we are going to use of the **post-estimation** command **lpgraph**, **also a new command**, that puts together the results of previously estimated IRFs of more than one model into one graph.

```
lpgraph ols ivr, h(15) z title(Response of the unemployment rate to monetary shock, size(*0.8)) ytitle(Percent)
ttitle(quarters) lab1(OLS) lab2(Inst.Var.) graphregion(fcolor(white))
```



Example 5. Non-linear effects and interactions: Using the option lcs()

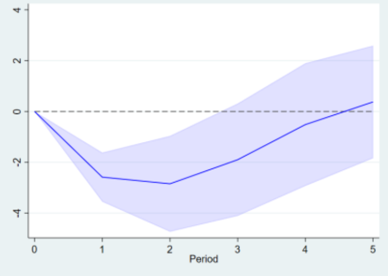
The variable F is a Financial Crisis dummy variable, while the variable N is a Normal Recession dummy variable. We want to look at the cumulative IRF of real GDP per capita to a financial crisis, while controlling for the effect of normal recessions, and viceversa. Thus, we need to use the option **lcs()** since the effect we are looking for is **the sum of the constant term** plus the effect of **F=1** or **N=1**.

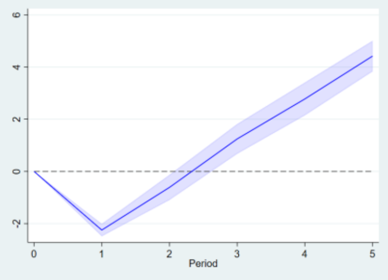
Therefore, we include the expression "**_cons + l.F**" and "**_cons + l.N**" inside the option **lcs()**. When we use the option **lcs()** the order of the variables **l.F** and **l.N** in the main syntax does not matter, since the shock is only determined by what it is defined by the option **lcs()**:

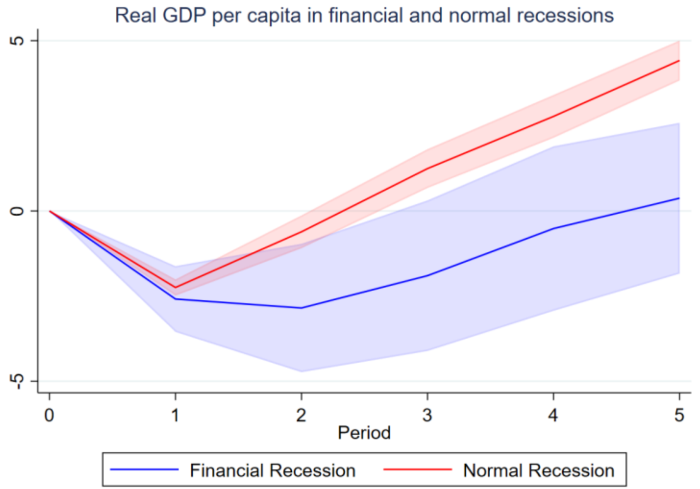
```
locproj rgdppc l.F l.N, fe robust tr(logs cmlt) h(4) z f(100) lcs(_cons + l.F) save irfn(FC)

locproj rgdppc l.F l.N, fe robust tr(logs cmlt) h(4) z f(100) lcs(_cons + l.N) save irfn(NC)

lpgraph FC NC, ti(Real GDP per capita in financial and normal recessions, size(*0.8)) graphregion(fc(white)) lab1(Financial Recession) lab2(Normal Recession)
```

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.00000	0.00000	0.00000	0.00000
1	-2.58574	0.45563	-3.55164	-1.61985
2	-2.84863	0.88923	-4.73371	-0.96354
3	-1.89969	1.04303	-4.11082	0.31143
4	-0.51638	1.13924	-2.93145	1.89869
5	0.37516	1.04460	-1.83929	2.58961

	IRF	Std.Err.	IRF LOW	IRF UP
0	0.00000	0.00000	0.00000	0.00000
1	-2.24700	0.11240	-2.48529	-2.00872
2	-0.61169	0.22738	-1.09372	-0.12965
3	1.24494	0.26917	0.67432	1.81555
4	2.77758	0.29673	2.14854	3.40663
5	4.41904	0.27724	3.83132	5.00676





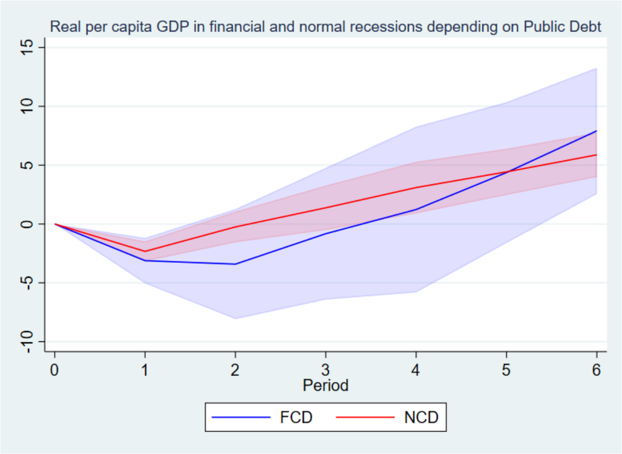


We can include more complicated interactions, for instance we can interact the effect of Financial Crisis (F) or Normal Recessions (N) with the public debt-to-GDP ratio and evaluate the IRF at different levels of such ratio. We first estimate the mean and standard deviation of the Public Debt-to-GDP ratio:

```
. sum debtgdp
. sca dm=r(mean)
. sca dsd=r(sd)
```

Then in the **lcs()** we enter the expansion of the interaction between financial crises and the debt ratio (1.l.F#c.l.debtgdp), multiplied by the mean and std. dev. of the ratio (dm+dsd):

```
locproj rgdppc l.N l.F l.(N#c.debtgdp F#c.debtgdp), fe robust tr(logs cmlt) nograph f(100) lcs(_cons+l.F+1.l.F#c.l.debtgdp*(dm+dsd)) save irfn(FCD)

locproj rgdppc l.N l.F l.(N#c.debtgdp F#c.debtgdp), fe robust tr(logs cmlt) nograph f(100) lcs(_cons+l.N+1.l.N#c.l.debtgdp*(dm+dsd)) save irfn(NCD)

lpgraph FCD NCD, h(6) title(Real per capita GDP in financial and normal recessions depending on Public Debt, size(*0.7))
```



Example 6. Non-linear effects, interactions and binary dependent variable: Using the option margins

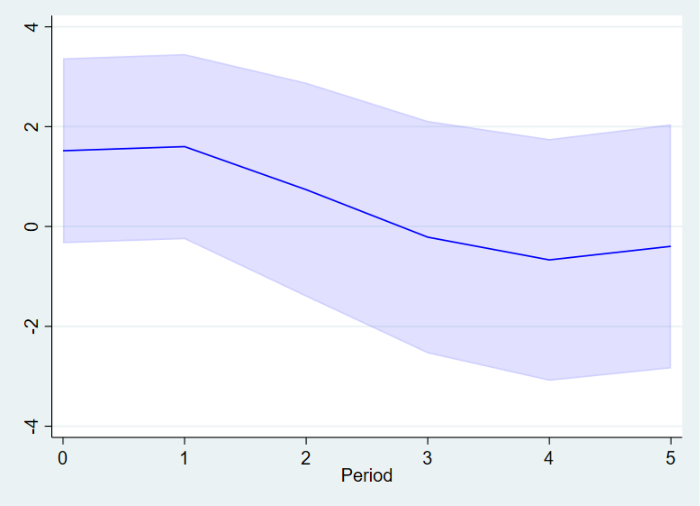
We will estimate the IRF of the probability of a banking crisis to an increase in the USA short-term interest rate (**stir__us**). Our dependent variable in this case is the dummy variable **crisisJST** that is equal to 1 for banking crises.

The option margins estimates the marginal effect of a unit of our shock variable (stir__us) on the probability of a banking crisis, which is our dependent (response) variable. We are using as estimation method the command xtlogit with fixed effects:

```
locproj crisisJST l(0/2).stir_us, margins m(xtlogit) fe f(100)
```

Impulse Response Function

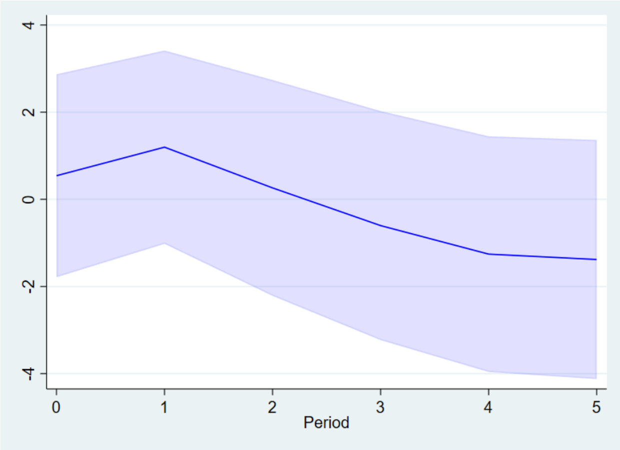
	IRF	Std.Err.	IRF LOW	IRF UP
0	1.51713	0.94456	3.36844	-0.33417
1	1.60001	0.94590	3.45393	-0.25391
2	0.73760	1.09385	2.88151	-1.40630
3	-0.21356	1.18797	2.11482	-2.54195
4	-0.66905	1.23535	1.75219	-3.09029
5	-0.39723	1.24693	2.04671	-2.84117

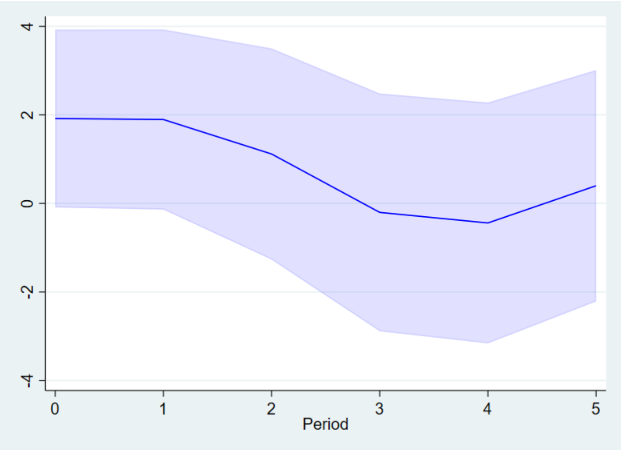


We can also interact the shock variable with a dummy variable, for instance, whether a country has a "PEG" foreign exchange regime.

The option margins allow us to estimate a separate IRF for each category of the dummy variable PEG. For doing that we need to use the option **mrfvar().** In this option we need to specify the expansion of the categorical variable that has been interacted with our shock variable. We also need to use the explicit option to define which variable is our shock without any interaction term, since the command margins does not accept an interaction term expression in its **dydx()** option:

```
locproj crisisJST peg#c.l(0/2).stir_us, s(stir_us) margins m(xtlogit) fe mrfvar(1.peg) f(100)

locproj crisisJST peg#c.l(0/2).stir_us, s(stir_us) margins m(xtlogit) fe mrfvar(0.peg) f(100)
```





References

Some Technical Literature:

- Barnichon, R. and Brownlees, C., 2019. Impulse response estimation by smooth local projections. Review of Economics and Statistics, 101(3), pp.522-530.
- Brugnolini, L., 2018. About Local Projection Impulse Response Function Reliability (No. 440). Tor Vergata University, CEIS.
- Jordà, Ò., 2005. Estimation and inference of impulse responses by local projections. American economic review, 95(1), pp.161-182.
- Jordà, Ò., Singh, S.R. and Taylor, A.M., 2020. The long-run effects of monetary policy (No. w26666). National Bureau of Economic Research.
- Li, D., Plagborg-Møller, M. and Wolf, C.K., 2022. Local projections vs. vars: Lessons from thousands of dgps (No. w30207). National Bureau of Economic Research.
- Montiel Olea, J.L. and Plagborg-Møller, M., 2021. Local projection inference is simpler and more robust than you think. Econometrica, 89(4), pp.1789-1823.
- Plagborg-Møller, M. and Wolf, C.K., 2021. Local projections and VARs estimate the same impulse responses. Econometrica, 89(2), pp.955-980.

Some Applied Literature:

- Berg, K.A., Curtis, C.C. and Mark, N., 2023. GDP and temperature: Evidence on cross-country response heterogeneity (No. w31327). National Bureau of Economic Research.
- Binici, M., Centorrino, S., Cevik, M.S. and Gwon, G., 2022. Here Comes the Change: The Role of Global and Domestic Factors in Post-Pandemic Inflation in Europe. International Monetary Fund.
- Caselli, M.F. and Roitman, A., 2016. Non-Linear Exchange Rate Pass-Through in Emerging Markets (No. 2016/001). International Monetary Fund.
- Ciminelli, G., Rogers, J. and Wu, W., 2022. The effects of US monetary policy on international mutual fund investment. Journal of International Money and Finance, p.102676
- Ferrari Minesso, M., Lebastard, L. & Le Mezo, H. Text-Based Recession Probabilities. IMF, Econ Rev (2022).
- Furceri, D. and Li, B.G., 2017. The macroeconomic (and distributional) effects of public investment in developing economies. International Monetary Fund.
- Jordà, Ò. and Taylor, A.M., 2016. The time for austerity: estimating the average treatment effect of fiscal policy. The Economic Journal, 126(590), pp.219-255.
- Jordà, Ò., 2005. Estimation and inference of impulse responses by local projections. American economic review, 95(1), pp.161-182.
- Jordà, Ò., Schularick, M. and Taylor, A.M., 2016. Sovereigns versus banks: credit, crises, and consequences. Journal of the European Economic Association, 14(1), pp.45-79.
- Jordà, Ò., Schularick, M. and Taylor, A.M., 2020. The effects of quasi-random monetary experiments. Journal of Monetary Economics, 112, pp.22-40.
- Metcalf, G.E. and Stock, J.H., 2020. The macroeconomic impact of Europe's carbon taxes (No. w27488). National Bureau of Economic Research.
- Miranda-Agrippino, S. and Ricco, G., 2021. The transmission of monetary policy shocks. American Economic Journal: Macroeconomics, 13(3), pp.74-107.
- Ramey, Valerie A. and Sarah Zubairy. 2018. Government spending multipliers in good times and in bad: Evidence from U.S. historical data. Journal of Political Economy,126(2):850–901.