Title

intro 1 — Introduction to DSGEs and dsge

Description Remarks and examples References Also see

Description

In this entry, we introduce DSGE models and the `dsge` command. We begin with an overview of DSGE models. We then illustrate the complete process of DSGE modeling by doing an example from start to finish. In this example, we demonstrate how to describe a model in its original nonlinear form, write it in a corresponding linearized form, estimate the parameters of the linearized model, and interpret the results. We conclude by showing how this example fits into the general DSGE modeling framework.

Remarks and examples

stata.com

Remarks are presented under the following headings:

Introduction to DSGE models

DSGE models are models for multiple time series used in macroeconomics and finance. These models are systems of equations that are motivated by economic theory and in which expectations of future values of variables play an important role. Because these models come from theory, the parameters of these models can typically be directly interpreted in terms of the motivating theory. DSGE models are used for macroeconomic policy analysis and forecasting.

In DSGE models, individuals' actions are summarized by decision rules that take the form of nonlinear systems of dynamic equations. These decision rules often come from dynamic stochastic optimization problems. A DSGE model consists of these decision rules, plus any aggregation conditions, resource or budget constraints, and stochastic processes for exogenous variables.

Because the model's equations are the solution to dynamic optimization problems, model equations can feature expectations of future variables. These expectations are endogenous. In DSGE models, expectations of future variables correspond to their conditional mean as implied by the model. In other words, individuals' expectations of future values are correct, on average. Such expectations are said to be model-consistent expectations or rational expectations.

There are three kinds of variables in DSGE models: control variables, state variables, and shocks. The terminology is taken from the state-space and optimal control literatures. In DSGE models, the concepts of exogeneity and endogeneity are understood relative to a time period. A state variable is fixed, or exogenous, in a given time period. The system of equations then determines the value of the state variable one period in the future. On the other hand, the system of equations determines the value of a control variable in the current time period. State variables evolve over time and may depend on control variables. State variables can also be correlated. Control variables in a DSGE model can be either observed or unobserved. State variables are always unobserved.

DSGE models can be written in multiple forms. The model based on economic theory may consist of equations that are nonlinear both in the variables and in the parameters. A DSGE model is said to be linear, or linearized, when the model equations are linear in the variables. DSGE models are commonly linearized prior to analysis. After linearization, the model variables are expressed as deviations from a steady state.

The DSGE model must be solved prior to estimation. Dynamic linear models are easier to solve and fit than dynamic nonlinear models, particularly when future expectations are incorporated into the model.

In any analysis of simultaneous equations systems, to solve a model means to write the model's endogenous variables as functions of its exogenous variables. In DSGE models, the analogous solution concept is to write the model's control variables in terms of its state variables. The model's solution consists of a system of equations relating the control variables to the state variables and a system of equations describing the evolution of the state variables over time. The solution to a DSGE model thus takes the form of a state-space model. The solution to a DSGE model is a crucial object for both estimation and analyses after estimation. Both the likelihood function and the impulse–response functions are formed from the model solution.

`dsge` solves and estimates the parameters of linearized DSGE models.

General introductions to DSGE modeling are available in Ljungqvist and Sargent (2012) and Woodford (2003). Canova (2007) and DeJong and Dave (2011) describe parameter estimation using DSGE models.

How to write down a DSGE

Writing down a DSGE is a two-step process. First, we write down (potentially) nonlinear structural equations that come from theory, usually economic theory. In many such theories, individuals do the best that they can given their constraints. This idea is formalized in dynamic stochastic optimization problems. The solutions to these problems are the nonlinear form of DSGE models. Second, we write down the linearized model corresponding to the nonlinear structural model.

Consider the following nonlinear model, similar to that in Woodford (2003, chap. 4). The model consists of equations that describe the behavior of households, firms, and a central bank. Interactions among these actors produce a model of inflation, output growth, and the interest rate. Models of this type are popular in academic and policy settings and are used to describe and analyze monetary policy.

Household optimization generates an equation that relates current output Y_t to the expected value of a function of tomorrow's output Y_{t+1}, tomorrow's inflation Π_{t+1}, and the current nominal interest rate R_t,

$$\frac{1}{Y_t} = \beta E_t \left[\frac{1}{Y_{t+1}} \frac{R_t}{\Pi_{t+1}} \right] \tag{1a}$$

where β is a parameter that captures households' willingness to delay consumption.

Optimization by firms generates an equation that links the current deviation of inflation from its steady state, $\Pi_t - \Pi$, to the expected value of the deviation of inflation from its steady state in the future, $E_t(\Pi_{t+1} - \Pi)$, and to the ratio of actual output, Y_t, to the natural level of output, Z_t,

$$\phi(\Pi_t - \Pi) + \theta - 1 = \theta \left(\frac{Y_t}{Z_t}\right)^{\eta - 1} + \beta\phi E_t(\Pi_{t+1} - \Pi) \tag{2a}$$

where ϕ, η, and θ are parameters linked to the pricing decision of firms. Firms are not affected by inflation per se; they are affected only by deviations of inflation from its steady-state value.

Finally, there is an equation that describes central bank policy. The central bank adjusts the interest rate in response to inflation and, perhaps, to other factors that we leave unmodeled. The equation for the central bank policy is

$$\frac{R_t}{R} = \left(\frac{\Pi_t}{\Pi}\right)^{1/\beta} e^{u_t} \tag{3a}$$

where R is the steady-state value of the interest rate and u_t is a state variable that captures all movements in the interest rate that are not driven by inflation.

Using the dsge command

Before you can use `dsge`, you must make sure certain things about your data and your model are true.

First, you must linearize the model by using deviations from steady state. `dsge` implements a check for linearity and will complain if it detects nonlinearities. We do not discuss how to linearize the equations in this manual. See, for example, DeJong and Dave (2011, chap. 2), which provides an overview of the linearization process.

Second, the data must be `tsset` prior to using `dsge`. Using `tsset` on the data allows us to use time-series operators.

Finally, the series in your model must have zero mean and be weakly stationary. `dsge` will demean the series for you, but you need to ensure that your data are weakly stationary before specifying them in the `dsge` command.

Writing down linearized DSGEs

The model in (1a)–(3a) is nonlinear. We now need to write the model in its corresponding linearized form, which we show below. Throughout this manual, we use lowercase letters to denote percentage deviations of variables from the steady state. The linearized versions of the above equations are

$$y_t = E_t y_{t+1} - (r_t - E_t \pi_{t+1}) \tag{1b}$$

$$\pi_t = \beta E_t \pi_{t+1} + \kappa(y_t - z_t) \tag{2b}$$

$$r_t = \frac{1}{\beta}\pi_t + u_t \tag{3b}$$

The new parameter κ is a complicated function of the underlying parameters from (2a). Those underlying parameters cannot be separately identified in this model, but κ can be.

We have implicitly constrained some of the coefficients; for example, the coefficient on the interest rate in the output equation is constrained to -1.

Note that we have not yet specified stochastic processes for z_t and u_t.

Data preparation

We have data on price levels and interest rates in `rates.dta`. These data were obtained from the Federal Reserve Economic Database (FRED), which contains many macroeconomic and financial time series; see [D] **import fred**.

```
. use http://www.stata-press.com/data/r15/rates2
(Federal Reserve Economic Data - St. Louis Fed, 2017-02-10)

. describe

Contains data from http://www.stata-press.com/data/r15/rates2.dta
  obs:           281                          Federal Reserve Economic Data -
                                                St. Louis Fed, 2017-02-10
 vars:             5                          26 Apr 2017 21:22
 size:         6,182
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
datestr         str10   %-10s                 Observation date
daten           int     %td                   Numeric (daily) date
gdpdef          float   %9.0g                 GDP deflator GDPDEF
r               float   %9.0g                 Federal funds rate FEDFUNDS
dateq           int     %tq                   Quarterly date
-------------------------------------------------------------------------------
Sorted by: dateq
```

The dataset includes the price level and the interest rate. But the model is written in terms of the inflation rate. For quarterly data, the inflation rate is conventionally obtained as 400 times the difference in log of price variable. Therefore, we begin by generating an inflation rate variable `p` by using the `L.` lag operator.

```
. generate p = 400*(ln(gdpdef) - ln(L.gdpdef))
(2 missing values generated)

. label variable p "Inflation rate"
```

We now have inflation and interest rates. They are not mean zero, but `dsge` will demean the data for us.

Specifying the model to dsge

We make one final modification to the model equations before estimating the parameters. Woodford (2003) rewrites the model in (1b)–(3b) by defining $x_t = y_t - z_t$ as the output gap. We do the same. Substituting in x_t gives us the three-equation system

$$x_t = E_t x_{t+1} - (r_t - E_t \pi_{t+1} - g_t) \tag{4}$$

$$\pi_t = \beta E_t \pi_{t+1} + \kappa x_t \tag{5}$$

$$r_t = \frac{1}{\beta}\pi_t + u_t \tag{6}$$

where $g_t = E_t(z_{t+1}) - z_t$ is a state variable. Equations (4)–(6) are the structural form of how the endogenous control variables x_t, π_t, and r_t evolve as functions of the exogenous state variables g_t and u_t.

We complete the model by specifying processes for how the state variables evolve. Per standard practice, both are modeled as first-order autoregressive processes.

$$u_{t+1} = \rho_u u_t + \epsilon_{t+1} \tag{7}$$

$$g_{t+1} = \rho_g g_t + \xi_{t+1} \tag{8}$$

The variables ξ_{t+1} and ϵ_{t+1} are shocks to the state variables.

Equations (4)–(6) are the linearized DSGE model derived from (1a)–(3a). Together with (7)–(8) for the evolution of state variables, this is a complete model whose parameters can be estimated.

The command to estimate the parameters in the system of (4)–(8) is

```
. dsge (p   = {beta}*E(F.p) + {kappa}*x)            ///
       (x   = E(F.x) -(r - E(F.p) - g), unobserved) ///
       (r   = (1/{beta})*p + u)                     ///
       (F.u = {rhou}*u, state)                      ///
       (F.g = {rhog}*g, state)
```

Each equation is bound in parentheses. The equations look almost identical to the system in (4)–(8). Because a model has as many variables as it has equations, each variable will appears on the left-hand side of one and only one equation.

The equation options `unobserved` and `state` modify how `dsge` interprets the equations we supply. Equations may be specified in any order; in this manual, we typically write control equations and then state equations, but you don't have to follow that convention.

`p`, `x`, and `r` are our control variables p_t, x_t, and r_t. Each appears on the left-hand side of one equation and can appear on the right-hand side of as many equations as we like. There are two state equations with shocks. The model must have the same number of shocks as observed control variables. Therefore, we can treat only two of three control variables as observed. Of our three control variables, the output gap is the most plausible to be unobserved. Thus we model the output gap as unobserved, and we model the inflation rate and the interest rate as observed. To specify inflation and the interest rate as observed, we only need to write their equations. We specify the output gap as unobserved using the `unobserved` option.

`u` and `g` are the state variables u_t and g_t. Recall that state variables are fixed in the current period, so we specify how they evolve through time by modeling the one-period lead—hence, the `F.` on the left-hand side of each state equation. The state equations specify how the state variable evolves as a function of the current state variables and, possibly, the control variables.

The shocks ϵ_t and ξ_t enter the system through the state equations of their corresponding variable. By default, a shock is attached to each state equation. So, when we typed

```
. dsge ... (F.u = {rhou}*u, state) ...
```

the underlying equation is what we wrote in (7). If a state variable is treated as deterministic in your model, then it will not have a shock. For example, capital accumulation is often treated as deterministic. To include an equation for a state variable without a shock, we would include the `noshock` option within the equation.

Expectations of future variables appear within the `E()` operator and use Stata's `F.` operator. For example, type `E(F.x)` to specify $E_t(x_{t+1})$.

The parameters we want to estimate are bound in braces.

For more details on the `dsge` syntax, see [DSGE] **intro 2**.

Parameter estimation and interpretation

We estimate the parameters of the model in (4)–(8). These equations are much discussed in the monetary economics literature. Equation (4) is known as the output-gap Euler equation. Equation (5) is known as a New Keynesian Phillips Curve, and the parameter κ is known as the slope of the Phillips curve. In New Keynesian models, prices depend on output, and κ is a measure of that dependence. Equation (6) is known as a Taylor rule, after Taylor (1993). The coefficient on inflation in a Taylor rule is a commonly discussed parameter. β has two roles in the model above. It relates current inflation deviations to expected future inflation deviations, and it relates interest rate deviations to inflation deviations.

```
. dsge (p = {beta}*E(F.p) + {kappa}*x)
>      (x = E(F.x) -(r - E(F.p) - g), unobserved)
>      (r = (1/{beta})*p + u)
>      (F.u = {rhou}*u, state)
>      (F.g = {rhog}*g, state)
(setting technique to bfgs)
Iteration 0:   log likelihood = -13931.564
Iteration 1:   log likelihood = -1301.5118  (backed up)
Iteration 2:   log likelihood = -1039.6984  (backed up)
Iteration 3:   log likelihood = -905.70867  (backed up)
Iteration 4:   log likelihood = -842.76867  (backed up)
(switching technique to nr)
Iteration 5:   log likelihood = -812.04209  (backed up)
Iteration 6:   log likelihood = -786.76609
Iteration 7:   log likelihood = -777.19779
Iteration 8:   log likelihood =  -768.8383
Iteration 9:   log likelihood =  -768.1368
Iteration 10:  log likelihood = -768.09519
Iteration 11:  log likelihood = -768.09383
Iteration 12:  log likelihood = -768.09383
DSGE model

Sample: 1954q3 - 2016q4                         Number of obs     =        250
Log likelihood = -768.09383
```

	Coef.	OIM Std. Err.	z	P>|z|	[95% Conf.	Interval]
/structural						
beta	.5112881	.075791	6.75	0.000	.3627404	.6598359
kappa	.1696296	.0475492	3.57	0.000	.076435	.2628243
rhou	.6989189	.0449192	15.56	0.000	.6108789	.7869588
rhog	.9556407	.0181342	52.70	0.000	.9200983	.9911831
sd(e.u)	2.317589	.2988024			1.731947	2.90323
sd(e.g)	.6147348	.0973279			.4239757	.8054939

Two of the parameters have structural interpretations. The parameter `kappa` is the slope of the Phillips curve. Theory predicts that this parameter will be positive, and indeed our estimate is positive.

The parameter `beta` is the inverse of the coefficient on inflation in the interest rate equation. We can obtain an estimate of $1/\beta$, which is interpreted as the degree to which the central bank responds to movements in inflation, by using `nlcom`.

```
. nlcom 1/_b[beta]
       _nl_1:  1/_b[beta]
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_nl_1	1.955844	.2899255	6.75	0.000	1.387601	2.524088

A typical value for $1/\beta$ in the literature is 1.5. Our point estimate is around 2.

Postestimation

Policy and transition matrices

The matrix of parameters in the state-space form that specifies how the state variables affect the control variables is known as the policy matrix. Each policy matrix parameter is the effect of a one-unit shock to a state variable on a control variable.

▷ Example 1: Obtaining the policy matrix

We use `estat policy` to view these results.

```
. estat policy
Policy matrix
```

	Coef.	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
p						
u	-.4170859	.0389324	-10.71	0.000	-.4933919	-.3407799
g	.881884	.2330573	3.78	0.000	.4251001	1.338668
x						
u	-1.580153	.3926336	-4.02	0.000	-2.3497	-.8106049
g	2.658667	.9045286	2.94	0.003	.885823	4.43151
r						
u	.1842449	.056798	3.24	0.001	.072923	.2955669
g	1.724828	.2210259	7.80	0.000	1.291625	2.158031

Results are listed equation by equation. The first block is the policy equation for inflation `p` and writes it as a function of the state variables alone. A unit shock to the state `u` reduces inflation by an estimated 0.417, and a unit shock to `g` raises inflation by an estimated 0.882.

◁

The matrix of parameters that specifies the dynamic process for the state variables is known as the state transition matrix. The state transition equation relates the future values of the state variables to their values in the current period. Each state transition matrix parameter is the effect of a one-unit shock to a state variable on its one-period-ahead mean.

▷ Example 2: Obtaining the transition matrix

```
. estat transition
Transition matrix of state variables
```

		Coef.	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
F.u							
	u	.6989189	.0449192	15.56	0.000	.6108789	.7869588
	g	1.11e-16	7.11e-12	0.00	1.000	-1.39e-11	1.39e-11
F.g							
	u	0	(omitted)				
	g	.9556407	.0181342	52.70	0.000	.9200983	.9911831

Both state variables are modeled as autoregressive processes, so the results in `estat transition` repeat the estimates of `rhou` and `rhog` from the `dsge` output. In this case, the other entries in the state transition matrix are 0 or differ from 0 only because of lack of numerical precision. In more complicated models, such as those in which a state equation depends on a control variable, the state transition matrix will contain new information about that state variable.

◁

Impulse responses

The state-space form allows us to trace the path of a control or state in response to a shock to a state. This path is known as an impulse–response function (IRF). `irf` after `dsge` estimates IRFs, and it puts the named set of estimates into an `.irf` file, whose results can be displayed using `irf graph` or `irf table`.

▷ Example 3: Graphing an IRF

To graph the IRF, we first create the `nkirf.irf` file and set it as the active `.irf` file using the `irf set` command.

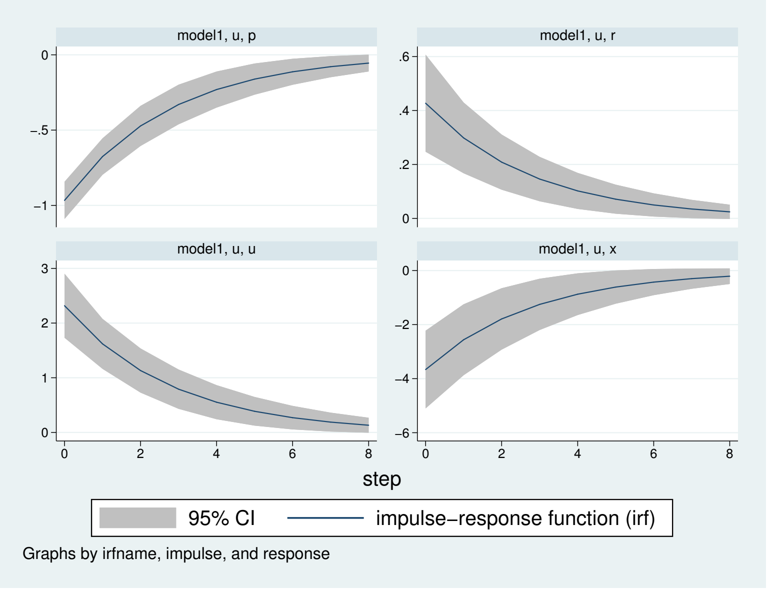
```
. irf set nkirf.irf
(file nkirf.irf created)
(file nkirf.irf now active)
```

Next, we use `irf create` to estimate a complete set of impulse responses based on our `dsge` command. A complete set of impulse responses is an impulse to each shock and the response of each state and control variable to that impulse. For the model in this example, `irf create` generates an impulse to `e.u` and `e.g`, then stores the response to that impulse on `p`, `x`, `r`, `g`, and `u`. The results are stored in the `nkirf.irf` file.

```
. irf create model1
(file nkirf.irf updated)
```

We now use `irf graph` to plot the impulse responses. The `impulse()` and `response()` options control which impulse and which responses are chosen. To view the response of `p`, `x`, `r`, and `u` to a shock to `u`, we type

```
. irf graph irf, impulse(u) response(x p r u) byopts(yrescale)
```



The state variable u models movements in the interest rate that occur for reasons other than the feedback between inflation and the interest rate. A shock to u is effectively a surprise increase in the interest rate, and the IRF traces out how this shock causes temporary decreases to inflation (top-left graph) and to the output gap (bottom-right graph).

◁

Forecasts

The forecast suite of commands produces dynamic forecasts from the fitted model.

▷ Example 4: Out-of-sample forecast

We first store the dsge estimation results.

```
. estimates store dsge_est
```

We use tsappend() to extend the dataset by 3 years, or 12 quarters.

```
. tsappend, add(12)
```

To set up a forecast, we perform three steps. forecast create initializes a new forecasting model, which we name dsgemodel.

```
. forecast create dsgemodel
  Forecast model dsgemodel started.
```

Next, we add the estimates from dsge to the forecasting model using forecast estimates.

```
. forecast estimates dsge_est
  Added estimation results from dsge.
  Forecast model dsgemodel now contains 2 endogenous variables.
```

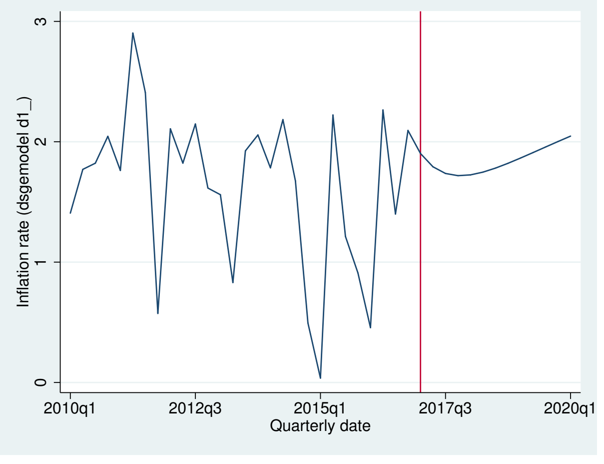
This command adds the estimates stored in dsge_est to the model dsgemodel. We now produce dynamic forecasts beginning in the first quarter of 2017 using forecast solve. The prefix(d1_) option specifies the d1_ prefix that will be given to the variables created by forecast. We also request that dynamic forecasts begin in the first quarter of 2017 with the begin(tq(2017q1)) option.

```
. forecast solve, prefix(d1_) begin(tq(2017q1))
Computing dynamic forecasts for model dsgemodel.
─────────────────────────────────────────────────
Starting period:  2017q1
Ending period:    2020q1
Forecast prefix:  d1_
2017q1:  ..............
2017q2:  ..............
2017q3:  ..............
2017q4:  ..............
2018q1:  ..............
2018q2:  .............
2018q3:  .............
2018q4:  .............
2019q1:  .............
2019q2:  .............
2019q3:  .............
2019q4:  .............
2020q1:  .............
Forecast 2 variables spanning 13 periods.
──────────────────────────────────────────
```

The dynamic forecast begins in the first quarter of 2017, so all forecasts are out of sample.

We can graph the forecast for inflation d1_p using tsline.

```
. tsline d1_p if tin(2010q1, 2021q1), tline(2017q1)
```



The model forecasts that inflation will smoothly return to its long-run value, the sample mean.

◁

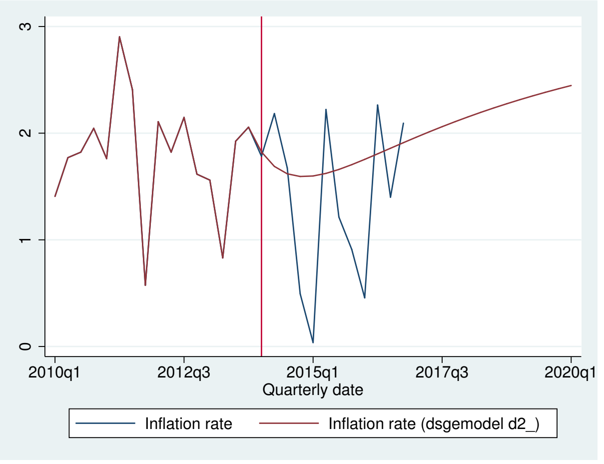
We can also begin the forecast during a time period for which observations are available.

▷ Example 5: Within-sample forecast

Specifying the begin(tq(2014q1)) option produces dynamic forecasts beginning in the first quarter of 2014, so we can compare the forecast for 2014–2016 with the actual observations over that period.

```
. forecast solve, prefix(d2_) begin(tq(2014q1))
Computing dynamic forecasts for model dsgemodel.
------------------------------------------------
Starting period:  2014q1
Ending period:    2020q1
Forecast prefix:  d2_
2014q1:  ..............
2014q2:  ..............
2014q3:  ..............
2014q4:  ..............
2015q1:  ..............
2015q2:  ..............
2015q3:  ..............
2015q4:  .............
2016q1:  .............
2016q2:  .............
2016q3:  .............
2016q4:  .............
2017q1:  .............
2017q2:  .............
2017q3:  .............
2017q4:  .............
2018q1:  .............
2018q2:  .............
2018q3:  .............
2018q4:  .............
2019q1:  .............
2019q2:  .............
2019q3:  .............
2019q4:  .............
2020q1:  .............
Forecast 2 variables spanning 25 periods.
-----------------------------------------
. tsline p d2_p if tin(2010q1, 2021q1), tline(2014q1)
```

We plot both the observed inflation and the forecast.



The forecast captures the general upward trend in inflation from 2014–2016, but it does not predict the variation in inflation around the upward trend.

◁

Structural and reduced forms of DSGE models

Now that we have worked an example, we show how it fits in the more general formulation of DSGE models. The model in (4)–(8) is an example of a linearized DSGE model. In general, a linearized DSGE model can be expressed as

$$\mathbf{A}_0\mathbf{y}_t = \mathbf{A}_1 E_t(\mathbf{y}_{t+1}) + \mathbf{A}_2\mathbf{y}_t + \mathbf{A}_3\mathbf{x}_t \tag{9}$$

$$\mathbf{B}_0\mathbf{x}_{t+1} = \mathbf{B}_1 E_t(\mathbf{y}_{t+1}) + \mathbf{B}_2\mathbf{y}_t + \mathbf{B}_3\mathbf{x}_t + \mathbf{C}\boldsymbol{\epsilon}_{t+1} \tag{10}$$

where $\mathbf{y}_t$ is a vector of control variables, $\mathbf{x}_t$ is a vector of state variables, and $\boldsymbol{\epsilon}_t$ is a vector of shocks. $\mathbf{A}_0$ through $\mathbf{A}_3$ and $\mathbf{B}_0$ through $\mathbf{B}_3$ are matrices of parameters. We require that $\mathbf{A}_0$ and $\mathbf{B}_0$ be diagonal matrices. The entries in $\mathbf{A}_i$ and $\mathbf{B}_j$ are all functions of the structural parameters, which we denote by vector $\boldsymbol{\theta}$. Economic theory places restrictions on the $\mathbf{A}_i$ and $\mathbf{B}_j$ matrices. $\mathbf{C}$ is a selection matrix that determines which state variables are subject to shocks.

The state-space form of the model is given by

$$\mathbf{y}_t = \mathbf{G}\mathbf{x}_t \tag{11}$$

$$\mathbf{x}_{t+1} = \mathbf{H}\mathbf{x}_t + \mathbf{M}\boldsymbol{\epsilon}_{t+1} \tag{12}$$

where $\mathbf{y}_t$ is a vector of control variables, $\mathbf{x}_t$ is a vector of state variables, and $\boldsymbol{\epsilon}_t$ is a vector of shocks. $\mathbf{G}$ is the policy matrix, and $\mathbf{H}$ is the state transition matrix. $\mathbf{M}$ is diagonal and contains the standard deviations of the shocks.

$\mathbf{y}_t$ is partitioned into observed and unobserved controls, $\mathbf{y}_t = (\mathbf{y}_{1,t}, \mathbf{y}_{2,t})$. The observed control variables are related to the control variables by the equation

$$\mathbf{y}_{1,t} = \mathbf{D}\mathbf{y}_t$$

where $\mathbf{D}$ is a selection matrix. Only observed control variables play a role estimation. The number of observed control variables must be the same as the number of state equations that include shocks.

You specify a model of the form (9)–(10) to `dsge`. Many models require some manipulation to fit into the structure in (9) and (10); see [DSGE] **intro 4** for details. Postestimation commands `estat policy` and `estat transition` will display the policy and transition matrices in (11) and (12), respectively.

References

Canova, F. 2007. *Methods for Applied Macroeconomic Research*. Princeton, NJ: Princeton University Press.

DeJong, D. N., and C. Dave. 2011. *Structural Macroeconometrics*. 2nd ed. Princeton, NJ: Princeton University Press.

Ljungqvist, L., and T. J. Sargent. 2012. *Recursive Macroeconomic Theory*. 3rd ed. Cambridge, MA: MIT Press.

Taylor, J. B. 1993. Discretion versus policy rules in practice. *Carnegie-Rochester Conference Series on Public Policy* 39: 195–214.

Woodford, M. 2003. *Interest and Prices: Foundations of a Theory of Monetary Policy*. Princeton, NJ: Princeton University Press.

Also see

[DSGE] **dsge** — Linearized dynamic stochastic general equilibrium models

[DSGE] **dsge postestimation** — Postestimation tools for dsge

[TS] **forecast** — Econometric model forecasting

[TS] **irf** — Create and analyze IRFs, dynamic-multiplier functions, and FEVDs

[TS] **sspace** — State-space models