Title

intro 2 — Learning the syntax

Description   Remarks and examples   Also see

Description

In this introduction, we demonstrate how to specify a DSGE model using the `dsge` command. We focus on two unique aspects of DSGEs that must be considered when writing the syntax—writing the system of equations and identifying each type of variable within those equations.

Remarks and examples

stata.com

Remarks are presented under the following headings:

Preview of dsge syntax

If you have not read [DSGE] **intro 1**, we recommend that you read it first. In particular, see *Structural and reduced forms of DSGE models* in [DSGE] **intro 1**, where we discuss the structural form of a DSGE that is required by `dsge`. Below we assume that your model has this structural form, and we discuss the syntax for specifying such a model with the `dsge` command.

As an example, if we wanted to fit the DSGE model

$$
\begin{aligned}
p_t &= \beta E_t(p_{t+1}) + \kappa y_t \\
y_t &= E_t(y_{t+1}) - (r_t - E_t(p_{t+1}) - \rho z_t) \\
\beta r_t &= p_t + \beta u_t \\
z_{t+1} &= \rho z_t + \epsilon_{t+1} \\
u_{t+1} &= \delta u_t + \xi_{t+1}
\end{aligned}
$$

we would type

```
. dsge (p        = {beta}*E(F.p) + {kappa}*y)                          ///
       (y        = E(F.y) -(r - E(F.p) - {rho}*z), unobserved)        ///
       ({beta}*r = p + {beta}*u)                                       ///
       (F.z      = {rho}*z, state)                                     ///
       (F.u      = {delta}*u, state)
```

The syntax looks a lot like the equations, but it does require a little explanation. In what follows, we discuss each element of the `dsge` syntax that you will need to specify the equations of your DSGE model.

In [DSGE] **intro 1**, we told you that three types of variables—control variables, state variables, and shocks—appear in DSGE models. We will demonstrate how to specify equations involving each of these types of variables. We will also show you how to specify the required types of equations that are linear in these variables and nonlinear in the parameters.

Specifying the system of equations

For a DSGE model, we specify an equation for each control variable and an equation for each state variable. The equations are bound by parentheses. Therefore, the basic structure of the `dsge` command is

```
. dsge (eq1)          ///
       (eq2)          ///
       ..., ...
```

where *eq1* and *eq2* will be replaced with the syntax representing one of the equations in our model. We use . . . to indicate that we can include more than two equations as well as options.

The basic form of an equation within the parentheses is

```
term = term [ + term [ + term [...]]]
```

where each *term* includes a variable name. The variable name may be preceded by a parameter or nonlinear combination of parameters. For instance, a valid equation might look something like

```
(y = {kappa}*z)
```

or

```
(y = {kappa}*z + {kappa}*{beta}*x)
```

or

```
(1/{beta}*y = {gamma}*z + x)
```

We will explain this later. For now, simply note that these equations are written in `dsge` syntax much like we would write the math, but the parameters we want to estimate are offset with braces, `{}`. If you have used any other Stata commands that work with substitutable expressions, you may recognize this notation. In fact, `dsge` uses a special form of substitutable expressions.

Before we discuss how to use `dsge`'s substitutable expressions, we will focus on how to specify each type of variable.

Control variables

Control variables can be modeled as a function of other control variables, expectations of the future value of control variables, and state variables. The equations for control variables do not include shocks.

We continue from the basic `dsge` command in the previous section, which was

```
. dsge (eq1)          ///
       (eq2)          ///
       ..., ...
```

If we have an observed control variable y, and for *eq1*, we want to specify that y_t be modeled as a function of z_t and x_t, we can type this equation in our `dsge` command as

```
. dsge (y = pexp*z + pexp*x)          ///
  (eq2)                               ///
  ..., ...
```

where *pexp* is a possibly nonlinear expression of parameters in each term.

Control variables can be observed or unobserved. Because we did not include any options within this set of parentheses, y is assumed to be an observed control variable. If y were unobserved, we would add the `unobserved` option as follows:

```
(y = pexp*z + pexp*x, unobserved)
```

Note that each control variable in the model must be included on the left-hand side of one, and only one, equation.

State variables and shocks

To model a state variable, we specify an equation with the one-period lead of that state variable on the left-hand side. On the right-hand side of the equation, we can include state variables, control variables, expectations of the future value of control variables, and shocks.

When we specify an equation for a state variable, we include the `state` option within the parentheses defining the equation.

Continuing with the syntax above, suppose *eq2* is an equation for a state variable x, and we model x_{t+1} as a function of x_t. We expand our `dsge` command to

```
. dsge (y = pexp*z + pexp*x)          ///
  (F.x = pexp*x, state)              ///
  ..., ...
```

We used the `F.` lead operator to specify the one-period lead of x as `F.x`; see **[U] 11.4.4 Time-series varlists**. However, notice that the full list of time-series operators is not available here. We can specify only equations for one-period leads of state variables; we could not replace `F.x` with `F2.x` in the equation above. Note that this restriction does not limit the types of models we can fit; see **[DSGE] intro 4c**.

By default, the equation for a state variable includes an unobserved shock. However, equations for state variables are not required to include a shock. Within the system of equations, the number of shocks should be equal to the number of observed control variables. If we did not wish to include a shock in the equation for x_{t+1}, we could add the `noshock` option,

```
(F.x = pexp*x, state noshock)
```

Note that the one-period lead of each state variable must be included on the left-hand side of one, and only one, equation.

Expectations of future values of control variables

Expectations of the one-period lead of control variables can appear in equations for both control and state variables. Mathematically, we write these expectations as $E_t()$. For instance, we write the expectation of y in time $t+1$ as $E_t(y_{t+1})$. In `dsge`, we write this expectation as `E(F.y)`, where `E()` represents an expectation and `F.y` is the one-period lead of y. Other than using the special `E()` notation, expectations are included in the model in the same way as variables.

If we model y_t as a function of $E_t(y_{t+1})$ in addition to z_t and x_t, we can expand our previous `dsge` command to

```
. dsge (y = pexp*E(F.y) + pexp*z + pexp*x)          ///
  (F.x = pexp*x, state)                            ///
  ..., ...
```

Expectations in E() are strictly for one-period leads of control variables. You cannot, for instance, use E(F2.y) to include the expectation of y in time $t+2$ in the model. This does not prevent you, however, from including such terms in your model. See [DSGE] **intro 4d** for details of fitting models including expectations of control variables more than one period in the future.

Specifying parameters using substitutable expressions

At this point, we know how to specify each type of variable that may arise in our DSGE model. We now turn to specifying the parameters that we want to estimate.

Recall that the basic form of an equation is

term = *term* [+ *term* [+ *term* [...]]]

where each *term* includes a variable name. The variable name may be preceded by a parameter or a nonlinear combination of parameters. For *terms* on the right-hand side of the equation, variable names can also be followed by a parameter specification.

Observed control variables are variables in your dataset. Unobserved control variables and state variables are not variables in your dataset.

We specify the *terms* using a special type of substitutable expressions that we call scalar substitutable expressions. In scalar substitutable expressions, parameters are enclosed in braces, {}, and may enter the model either linearly or nonlinearly. The restriction that each term includes only one variable implies that the variables must enter the equation linearly with these scalar substitutable expressions.

If our model for y in the equation above is

$$y_t = \beta E_t(y_{t+1}) + \kappa z_t + \gamma x_t$$

we could extend our previous dsge command as follows:

```
. dsge (y = {beta}*E(F.y) + {kappa}*z + {gamma}*x)   ///
  (F.x = pexp*x, state)                              ///
  ..., ...
```

However, the equation for y need not be linear in the parameters. If instead we wanted to model y as

$$y_t = (1/\beta) E_t(y_{t+1}) + \kappa z_t + (\gamma/\beta) x_t$$

we would change our command to

```
. dsge (y = 1/{beta}*E(F.y) + {kappa}*z + ({gamma}/{beta})*x)  ///
  (F.x = pexp*x, state)                                         ///
  ..., ...
```

We can even include parameters on the left-hand side of the equation. For instance, we could model y as

$$(1/\beta) y_t = E_t(y_{t+1}) + \kappa z_t + (\gamma/\beta) x_t$$

and change our dsge command to

```
. dsge ((1/{beta})*y = E(F.y) + {kappa}*z + ({gamma}/{beta})*x)  ///
  (F.x = pexp*x, state)                                           ///
  ..., ...
```

Finally, there is an extension to the basic form of the equation that we should mention. Sometimes, it is more convenient to write an equation so that a scalar value or a parameter or a nonlinear combination of parameters is multiplied by a linear combination of terms. The dsge command allows this. For instance, instead of

$$y_t = (1/\beta)E_t(y_{t+1}) + (\gamma/\beta)x_t + \kappa z_t$$

we can write

$$y_t = (1/\beta)\{E_t(y_{t+1}) + \gamma x_t\} + \kappa z_t$$

Similarly, we could write this equation in the dsge command as

```
(y = (1/{beta}) * (E(F.y) + {gamma}*x) + {kappa}*z)
```

Now you know the general syntax of dsge and are ready to fit your own DSGE model with dsge.

For examples of fitting classic DSGE models using this syntax, see [DSGE] **intro 3**. If you find some of the rules of this syntax too restrictive for your model—say you want a shock in an equation for a control variable, or you need to include the lag of a state variable—see [DSGE] **intro 4** for examples of rewriting these types of models so that they can be fit using dsge.

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

Also see

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

[DSGE] **intro 3** — Classic DSGE examples

[DSGE] **intro 4** — Writing a DSGE in a solvable form