**mswitch** — Markov-switching regression models

## Description

`mswitch` fits dynamic regression models that exhibit different dynamics across unobserved states using state-dependent parameters to accommodate structural breaks or other multiple-state phenomena. These models are known as Markov-switching models because the transitions between the unobserved states follow a Markov chain.

Two models are available: Markov-switching dynamic regression (MSDR) models that allow a quick adjustment after the process changes state and Markov-switching autoregression (MSAR) models that allow a more gradual adjustment.

## Quick start

MSDR model for the dependent variable `y` with two state-dependent intercepts using `tsset` data
```
mswitch dr y
```
Same as above
```
mswitch dr y, states(2)
```
As above, but with three states and switching coefficients on `x`
```
mswitch dr y, switch(x) states(3)
```
MSDR model with two state-dependent intercepts and variance parameters
```
mswitch dr y, varswitch
```
MSAR model with two state-dependent intercepts and an autoregression (AR) term
```
mswitch ar y, ar(1)
```
As above, but with switching AR coefficients
```
mswitch ar y, ar(1) arswitch
```

## Menu

Statistics > Time series > Markov-switching model
## Syntax

**Markov-switching dynamic regression**

```
mswitch dr depvar [nonswitch_varlist] [if] [in] [, options]
```

### Markov-switching AR

```
mswitch ar depvar [nonswitch_varlist], ar(numlist) [msar_options options]
```

*nonswitch_varlist* is a list of variables with state-invariant coefficients.

### options Description

<table>
<thead>
<tr>
<th>Main</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>states(#)</code></td>
<td>specify number of states; default is <code>states(2)</code></td>
</tr>
<tr>
<td><code>switch([varlist][, noconstant])</code></td>
<td>specify variables with switching coefficients; by default, the constant term is state dependent unless <code>switch(, noconstant)</code> is specified</td>
</tr>
<tr>
<td><code>constant</code></td>
<td>allow a state-invariant constant term; may be specified only with <code>switch(, noconstant)</code></td>
</tr>
<tr>
<td><code>varswitch</code></td>
<td>specify state-dependent variance parameters; by default, the variance parameter is constant across all states</td>
</tr>
<tr>
<td><code>p0(type)</code></td>
<td>specify initial unconditional probabilities where <em>type</em> is one of <code>transition</code>, <code>fixed</code>, or <code>smoothed</code>; the default is <code>p0(transition)</code></td>
</tr>
<tr>
<td><code>constraints(numlist)</code></td>
<td>apply specified linear constraints</td>
</tr>
</tbody>
</table>

### SE/Robust

| vce(vcetype) | `vcetype` may be `oim` or `robust` |

### Reporting

<table>
<thead>
<tr>
<th><code>level(#)</code></th>
<th>set confidence level; default is <code>level(95)</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>nocnsreport</code></td>
<td>do not display constraints</td>
</tr>
<tr>
<td><code>display_options</code></td>
<td>control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling</td>
</tr>
</tbody>
</table>

### EM options

<table>
<thead>
<tr>
<th><code>emiterate(#)</code></th>
<th>specify the number of expectation-maximization (EM) iterations; default is <code>emiterate(10)</code></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>emlog</code></td>
<td>show EM iteration log</td>
</tr>
<tr>
<td><code>emdots</code></td>
<td>show EM iterations as dots</td>
</tr>
</tbody>
</table>

### Maximization

| `maximize_options` | control the maximization process |

### coeflegend

| display legend instead of statistics |
**msar_options**

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>* ar(numlist)</td>
</tr>
<tr>
<td>arswitch</td>
</tr>
</tbody>
</table>

*ar(numlist) is required.

You must tsset your data before using mswitch; see [TS] tsset.

varlist and nonswitch_varlist may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, nonswitch_varlist, and varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
by, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
coeflegend does not appear in the dialog box.

See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

**Options**

**Model**

* ar(numlist) specifies the number of AR terms. This option may be specified only with command mswitch ar. ar() is required to fit AR models.

arswitch specifies that the AR coefficients vary over the states. arswitch may be specified only with option ar().

**Main**

states(#) specifies the number of states. The default is states(2).

switch([ varlist[, noconstant] ]) specifies variables whose coefficients vary over the states. By default, the constant term is state dependent and is included in the regression model. You may suppress the constant term by specifying switch(), noconstant).

constant specifies that a state-invariant constant term be included in the model. This option may be specified only with switch(), noconstant).

varswitch specifies that the variance parameters are state dependent. The default is constant variance across all states.

p0(type) is rarely used. This option specifies the method for obtaining values for the unconditional transition probabilities. type is one of transition, fixed, or smoothed. The default is p0(transition), which specifies that the values be computed using the matrix of conditional transition probabilities. Type fixed specifies that each unconditional probability is 1/k, where k is the number of states. Type smoothed specifies that the unconditional probabilities be estimated as extra parameters of the model.

constraints(numlist) specifies the linear constraints to be applied to the parameter estimates.

**SE/Robust**

vce(vctype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim) and that are robust to some kinds of misspecification (robust); see [R] vce_option.
Reporting
level(#), nocnsreport; see [R] Estimation options.

display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels,
    allbaselevels, nofvlabel, fvwrap(#), fvwrapon(style), cformat(%fmt), pformat(%fmt),
    sformat(%fmt), and nolstretch; see [R] Estimation options.

EM options
emiterate(#), emlog, and emdots control the EM iterations that take place before estimation
switches to a quasi-Newton method. EM is used to obtain starting values.

emiterate(#) specifies the number of EM iterations; the default is emiterate(10).

emlog specifies that the EM iteration log be shown. The default is to not display the EM iteration
log.

emdots specifies that the EM iterations be shown as dots. The default is to not display the dots.

Maximization
maximize_options: difficult, technique(algorithm_spec), iterate(#), [no]log, trace,
    gradient, showstep, hessian, showtolerance, tolerance(#), itolerance(#),
    nrtolerance(#), nonrtolerance, and from(matname); see [R] Maximize for all options
except from(), and see below for information on from().

from(matname) specifies initial values for the maximization process. If from() is specified, the
initial values are used in the EM step to improve the likelihood unless emiterate(0) is also
specified. The coefficients obtained at the end of the EM iterations serve as initial values for
the quasi-Newton method.

matname must be a row vector. The number of columns must equal the number of parameters
in the model, and the values must be in the same order as the parameters in e(b).

The following option is available with mswitch but is not shown in the dialog box:
    coeflegend; see [R] Estimation options.

Remarks and examples

mswitch fits Markov-switching models in which the parameters vary over states. The states are
unobserved and follow a Markov process.

mswitch dr fits MSDR models that allow a quick adjustment after a state change and are often
used to model monthly and higher-frequency data. mswitch ar fits MSAR models that allow a more
gradual adjustment after a state change and are often used to model quarterly and lower-frequency
data. Estimation is by maximum likelihood. You must tsset your data before using mswitch; see
[TS] tsset.

Remarks are presented under the following headings:

Introduction
Markov-switching dynamic regression
Markov-switching AR
Video example

If you are new to Markov-switching models, we recommend that you begin with Introduction. A
more technical discussion and examples are presented in the model-specific sections.
Introduction

Markov-switching models are widely applied in the social sciences. For example, in economics, the growth rate of Gross Domestic Product is modeled as a switching process to capture the asymmetrical behavior observed over expansions and recessions (Hamilton 1989). Other economic examples include modeling interest rates (Garcia and Perron 1996) and exchange rates (Engel and Hamilton 1990). In finance, Kim, Nelson, and Startz (1998) model monthly stock returns, while Guidolin (2011b, 2011a) provide many applications of these models to returns, portfolio choice, and asset pricing. In political science, Jones, Kim, and Startz (2010) model Democratic and Republican partisan states in the United States Congress.

These models are also used in health sciences. For example, in psychology, Markov-switching models have been applied to daily data on manic and depressive states for individuals with rapid-cycling bipolar disorder (Hamaker, Grasman, and Kamphuis 2010). In epidemiology, Lu, Zeng, and Chen (2010) and Martinez-Beneito et al. (2008) model the incidence rate of infectious disease in epidemic and nonepidemic states.


Markov-switching models are used for series that are believed to transition over a finite set of unobserved states, allowing the process to evolve differently in each state. The transitions occur according to a Markov process. The time of transition from one state to another and the duration between changes in state is random. For example, these models can be used to understand the process that governs the time at which economic growth transitions between expansion and recession and the duration of each period.

Consider the evolution of a series $y_t$, where $t = 1, 2, \ldots, T$, is characterized by two states, as in the models below

State 1: $y_t = \mu_1 + \varepsilon_t$

State 2: $y_t = \mu_2 + \varepsilon_t$

where $\mu_1$ and $\mu_2$ are the intercept terms in state 1 and state 2, respectively. $\varepsilon_t$ is a white noise error with variance $\sigma^2$. The two states model shifts in the intercept term. If the timing of switches is known, the above model can be expressed as

$y_t = s_t \mu_1 + (1 - s_t) \mu_2 + \varepsilon_t$

where $s_t$ is 1 if the process is in state 1 and 0 otherwise. The above model is a regression with dummy variables and could be estimated with ordinary least squares using, for example, regress.

However, in the case of interest, we never know in which state the process is; that is to say, $s_t$ is not observed. Markov-switching regression models allow the parameters to vary over the unobserved states. In the simplest case, we can express this model as a MSDR model with a state-dependent intercept term

$y_t = \mu_{s_t} + \varepsilon_t$

where $\mu_{s_t}$ is the parameter of interest; $\mu_{s_t} = \mu_1$ when $s_t = 1$, and $\mu_{s_t} = \mu_2$ when $s_t = 2$.

Although one never knows with certainty in which state the process lies, the probabilities of being in each state can be estimated. For a Markov process, the transition probabilities are of greater interest. One-step transition probabilities are given by $p_{s_t, s_{t+1}}$, so for a two-state process, $p_{11}$ denotes the probability of staying in state 1 in the next period given that the process is in state 1 in the current period. Likewise, $p_{22}$ denotes the probability of staying in state 2. Values closer to 1 indicate a more persistent process, or in other words, that it is expected to stay in a given state for a long time.
Estimation of Markov-switching models proceeds by predicting the probabilities of the unobserved state and updating the likelihood at each period, akin to the Kalman filter. However, while the Kalman filter is concerned with making linear updates on continuous latent variables, the filter developed in Hamilton (1989) is a nonlinear algorithm that estimates the probabilities that a discrete, latent variable is in one of several states. Also see Hamilton (1990) for estimation of the model parameters by an EM algorithm, which is also a robust method to find reasonable starting values.

Markov-switching dynamic regression

In this section, we use a series of examples to describe MSDR models and the mswitch dr command.

MSDR models allow a quick adjustment after the process changes state. These models are often used to model monthly and higher-frequency data. When the process is in state $s$ at time $t$, a general specification of the MSDR model is written as

$$y_t = \mu_{s_t} + x_t \alpha + z_t \beta_{s_t} + \epsilon_{s_t}$$

where $y_t$ is the dependent variable, $\mu_{s_t}$ is the state-dependent intercept, $x_t$ is a vector of exogenous variables with state-invariant coefficients $\alpha$, $z_t$ is a vector of exogenous variables with state-dependent coefficients $\beta_{s_t}$, and $\epsilon_{s_t}$ is an independent and identically distributed (i.i.d.) normal error with mean 0 and state-dependent variance $\sigma_{s_t}^2$. $x_t$ and $z_t$ may contain lags of $y_t$. MSDR models allow states to switch according to a Markov process as described in Markov-switching regression models under Methods and formulas.

In the default model fit by mswitch dr, $s = 2$ and a constant $\sigma^2$ is assumed ($\sigma_1^2 = \sigma_2^2 = \sigma^2$), so three parameters, $\mu_1$, $\mu_2$, and $\sigma^2$, are estimated. There is no $x_t$ or $z_t$. The number of states may be increased with option states(). To include $x_t$, you specify a varlist after the command name, and to include $z_t$, you specify option switch(). The assumption of constant variances over states may be relaxed with option varswitch.

A more complete discussion of the MSDR model is provided in Specification of Markov-switching models under Methods and formulas.

Example 1: MSDR model with switching intercepts

Suppose we wish to model the federal funds rate. One potential model is a constant-only model

$$r_t = \mu_{s_t} + \epsilon_t$$

where $r_t$ is the federal funds rate, $s_t$ is the state, and $\mu_{s_t}$ is the mean in each state. In usmacro.dta, we have data for the series from the third quarter of 1954 to the fourth quarter of 2010 from the Federal Reserve Economic Database, a macroeconomic database provided by the Federal Reserve Bank of Saint Louis. The data are plotted below.
We note that the decades of 1970s and 1980s were characterized by periods of high interest rates while the rest of the sample displays moderate levels.

Thus, a two-state model seems reasonable. $s_t \in \{1, 2\}$ is the state; $\mu_1$ is the mean in the moderate-rate state; and $\mu_2$ is the mean in high-rate state. We can use mswitch dr with dependent variable fedfunds to estimate the parameters of the model.

```
. use https://www.stata-press.com/data/r16/usmacro
(Federal Reserve Economic Data - St. Louis Fed)
. mswitch dr fedfunds
```

```
Performing EM optimization:
Performing gradient-based optimization:
Iteration 0: log likelihood = -508.66031
Iteration 1: log likelihood = -508.6382
Iteration 2: log likelihood = -508.63592
Iteration 3: log likelihood = -508.63592
```

```
Markov-switching dynamic regression
Sample: 1954q3 - 2010q4 Number of obs = 226
Number of states = 2 AIC = 4.5455
Unconditional probabilities: transition HQIC = 4.5760
SBIC = 4.6211
Log likelihood = -508.63592

| fedfunds | Coef. | Std. Err. | z | P>|z| | [95% Conf. Interval] |
|-----------|-------|-----------|---|------|---------------------|
| State1    | _cons | 3.70877   | .1767083 | 20.99 | 0.000 | 3.362428 | 4.055112 |
| State2    | _cons | 9.556793  | .2999889 | 31.86 | 0.000 | 8.968826 | 10.14476 |
| sigma     |       | 2.107562  | .1008692 |       | 1.918851 | 2.314831 |
| p11       |       | .9820939  | .0104002 |       | .9450805 | .9943119  |
| p21       |       | .0503587  | .0268434 |       | .0173432 | .1374344  |
```

The header reports the sample size, fit statistics, the number of states, and the method used for computing the unconditional state probabilities. The EM algorithm was used to find the starting values.
for the quasi-Newton optimizer, and we see that it took three iterations for the model to converge. Finally, the header reports that the transition method was used to compute the unconditional state probabilities as a function of the transition probabilities; see Methods and formulas.

The estimation table reports results for each state-dependent mean and the constant error variance. Below that, the table displays the elements of the first \( k - 1 \) rows of the transition matrix, where \( k \) is the number of states.

State 1 is the moderate-rate state and has a mean interest rate of 3.71%. State 2 is the high-rate state and has a mean interest rate of 9.56%. \( p_{11} \) is the estimated probability of staying in state 1 in the next period given that the process is in state 1 in the current period. The estimate of 0.98 implies that state 1 is highly persistent. Similarly, \( p_{21} \) is the probability of transitioning to state 1 from state 2. The probability of staying in state 2 is therefore \( 1 - 0.05 = 0.95 \), which implies that state 2 is also highly persistent.

Note that it is arbitrary which state is called 1 or 2. Changing the initial values for the iterations, for example, can change the state labels for a given model-data combination. The transition probabilities will get swapped in accordance with the change in labels.

Technical note

As mentioned in Introduction, a model with one state is equivalent to linear regression. To estimate a one-state constant-only model for the data in example 1, you type

```
.mswitch dr fedfunds, states(1)
```

This is equivalent to typing

```
.arima fedfunds, technique(nr)
```

or

```
.regress fedfunds
```

The commands produce nearly identical parameter estimates for the coefficients.

Example 2: MSDR model with switching intercepts and coefficients

Continuing example 1, we specify a more complex model that includes the lagged value of the interest rate and allows its coefficient to switch as well. The respecified model is

\[
r_t = \mu_{s_t} + \phi_{s_t} r_{t-1} + \varepsilon_t
\]
We estimate the switching coefficient by including the `switch()` option.

```
. mswitch dr fedfunds, switch(L.fedfunds)
```

Performing EM optimization:
Performing gradient-based optimization:

```
Iteration 0: log likelihood = -265.37725
Iteration 1: log likelihood = -264.74265
Iteration 2: log likelihood = -264.71073
Iteration 3: log likelihood = -264.71069
Iteration 4: log likelihood = -264.71069
```

Markov-switching dynamic regression

```
Sample: 1954q4 - 2010q4 Number of obs = 225
Number of states = 2 AIC = 2.4152
Unconditional probabilities: transition HQIC = 2.4581
SBIC = 2.5215
Log likelihood = -264.71069
```

| fedfunds | Coef. | Std. Err. | z    | P>|z|  | [95% Conf. Interval] |
|----------|-------|-----------|------|------|----------------------|
| State1   |       |           |      |      |                      |
| fedfunds | .7631424 | .0337234 | 22.63 | 0.000 | .6970457 - .8292392  |
| L1.      | .724457 | .288657   | 2.51 | 0.012 | .1586826 - 1.290231  |

| State2   |       |           |      |      |                      |
| fedfunds | 1.061174 | .0185031 | 57.35 | 0.000 | 1.024908 - 1.097439 |
| L1.      | -.0988764 | .1183838 | -0.84 | 0.404 | -.3309043 - .1331515 |
| _cons    | .6915759 | .0358644 | .6247373 | .7655663 |
| sigma    | .6378175 | .1202616 | .3883032 | .830089 |
| p11      | .1306295 | .0495924 | .0600137 | .2612432 |

The output indicates that the coefficients on the lagged dependent variable in the two states are significant. Also, we favor this model over the constant-only model because the Schwarz Bayesian information criterion (SBIC) for this model, 2.52, is lower than the SBIC for the constant-only model, 4.62.

Example 3: Changing the number of states for an MSDR model

Continuing example 2, we now specify a Taylor-rule model with two and three states and select the preferred number of states.

Taylor-rule models specify that the interest rate depends on its own lag, the current value of inflation, and the output gap. In our dataset, `ogap` is the output gap and `inflation` is inflation.
First, we fit a two-state MSDR Taylor-rule model with `fedfunds` as the interest rate.

```
.mswitch dr fedfunds, switch(L.fedfunds ogap inflation)
```

Performing EM optimization:
Performing gradient-based optimization:
Iteration 0:  log likelihood = -229.43752
Iteration 1:  log likelihood = -229.25718
Iteration 2:  log likelihood = -229.25614
Iteration 3:  log likelihood = -229.25614

Markov-switching dynamic regression

|               | Coef.  | Std. Err. | z     | P>|z| | [95% Conf. Interval] |
|---------------|--------|-----------|-------|------|----------------------|
| **State 1**   |        |           |       |      |                      |
| `fedfunds`    | .8314458 | .0333236 | 24.95 | 0.000 | .7661328 - .8967587  |
| L1. `fedfunds`| .1355425 | .0294113 | 4.61  | 0.000 | .0778975 - .1931875  |
| `ogap`        | -.0273928 | .0408057 | -0.67 | 0.502 | -.1073704 .0525849  |
| `inflation`   | .6554954 | .1373889 | 4.77  | 0.000 | .386218 .9247727    |
| `_cons`       | -.0273928 | .0408057 | -0.67 | 0.502 | -.1073704 .0525849  |
| **State 2**   |        |           |       |      |                      |
| `fedfunds`    | .9292574 | .0270852 | 34.31 | 0.000 | .8761713 .9823435   |
| L1. `fedfunds`| .0343072 | .0240138 | 1.43  | 0.153 | -.0127589 .0813733  |
| `ogap`        | .2125275 | .0297351 | 7.15  | 0.000 | .1542477 .2708072   |
| `inflation`   | -.0944924 | .1279231 | -0.74 | 0.460 | -.3452171 .1562324  |
| `_cons`       | -.0273928 | .0408057 | -0.67 | 0.502 | -.1073704 .0525849  |
| `sigma`       | .5764495 | .0302562 | 18.84 | 0.000 | .5200968 .638908    |
| `p11`         | .7279288 | .0929915 | 7.85  | 0.000 | .5519594 .8703909   |
| `p21`         | .2114578 | .0641179 | 3.29  | 0.001 | .1120643 .3629704   |

The results indicate that inflation does not significantly affect `fedfunds` in state 1 but that it does in state 2.
Would a model with three states be better than the above two-state model?

```
.mswitch dr fedfunds, switch(L.fedfunds ogap inflation) states(3)
(output omitted)
```

Markov-switching dynamic regression

```
Sample: 1955q3 - 2010q4  Number of obs =  222
Number of states = 3     AIC =  1.8819
Unconditional probabilities: transition  HQIC =  1.9995
                                      SBIC =  2.1732
Log likelihood = -189.89493
```

```
\begin{tabular}{l|cccc}
 fedfunds & Coef. & Std. Err. & z & P>|z| & [95\% Conf. Interval] \\
\hline
State1 & & & & & \\
 fedfunds & .8464551 & .0333543 & 25.38 & 0.000 & .781082 & .9118283 \\
 L1. & & & & & \\
o gap & .1201952 & .0232718 & 5.16 & 0.000 & .0745834 & .1658071 \\
i nflation & -.0425603 & .0354281 & -1.20 & 0.230 & -.1119981 & .0268775 \\
_c ons & .5261292 & .1266916 & 4.15 & 0.000 & .2778182 & .7744403 \\
State2 & & & & & \\
 fedfunds & .9690088 & .0264823 & 36.59 & 0.000 & .9171046 & 1.020913 \\
 L1. & & & & & \\
o gap & .0464136 & .0200197 & 2.32 & 0.020 & .0071758 & .0856515 \\
i nflation & .1298906 & .0246793 & 5.26 & 0.000 & .0815202 & .1782611 \\
_c ons & -.0034106 & .1073034 & -0.03 & 0.975 & -.2137206 & .2068995 \\
State3 & & & & & \\
 fedfunds & .4178913 & .0809412 & 5.16 & 0.000 & .2592494 & .5765333 \\
 L1. & & & & & \\
o gap & .1075357 & .1131713 & 0.95 & 0.342 & -.1142761 & .3293474 \\
i nflation & .9099168 & .0732991 & 12.41 & 0.000 & .7662531 & 1.053581 \\
_c ons & .6015991 & .8894513 & 0.68 & 0.499 & -.1416934 & 2.344892 \\
\hline
 sigma & .438375 & .0247954 & & & .3923738 & .4897694 \\
p11 & .7253684 & .0807816 & & & .5440011 & .8539642 \\
p12 & .2564055 & .0784303 & & & .1334279 & .4357355 \\
p21 & .1641252 & .0548584 & & & .0822928 & .3006711 \\
p22 & .7994204 & .0578707 & & & .6626908 & .8899317 \\
p31 & .6178282 & .3481683 & & & .0824324 & .9667674 \\
p32 & .3821718 & .3481683 & & & .0332326 & .9175676 \\
\end{tabular}
```

We favor the three-state model over the two-state model because it has the lower SBIC. The three states, in this case, can be thought of as representing low, moderate, and high-interest rate states.

The results for the three-state model indicate that inflation does not affect the interest rate in state 1, but it does affect the interest rate in states 2 and 3. The results also indicate that when the coefficient on inflation is large and significant in state 3, the output gap coefficient is not significant.
Technical note

In some situations, the quasi-Newton optimization will not converge, which implies that the parameters of the specified model are not identified by the data. These convergence problems most frequently arise when attempting to fit a model with too many states.

Example 4: Switching variances

All examples thus far have assumed a constant variance across states. In some cases, we may wish to relax this assumption. For example, in the snp500 dataset we have weekly data on the absolute returns of the S&P 500 index from the period 2004w17 to 2014w18, which we present below. The graph indicates that there were high-volatility periods in 2008 to 2009 and in late 2011. It would be unreasonable to assume that the variance in this high-volatility state is equal to the variance in the low-volatility state.
Below we fit `areturns`, the absolute returns, with an MSDR model in which the coefficients on the lagged dependent variable and the variances differ by state.

```
. use https://www.stata-press.com/data/r16/snp500
(Federal Reserve Economic Data - St. Louis Fed)
. mswitch dr areturns, switch(L.areturns) varswitch
```

Performing EM optimization:
Performing gradient-based optimization:

```
Iteration 0: log likelihood = -753.27687
Iteration 1: log likelihood = -746.54052
Iteration 2: log likelihood = -745.80829
Iteration 3: log likelihood = -745.7977
Iteration 4: log likelihood = -745.7977
```

```markdown
Markov-switching dynamic regression
Sample: 2004w19 - 2014w18 Number of obs = 520
Number of states = 2 AIC = 2.8992
Unconditional probabilities: transition HQIC = 2.9249
SBIC = 2.9647
Log likelihood = -745.7977
```

|        | Coef. Std. Err. | z    | P>|z| | [95% Conf. Interval] |
|--------|-----------------|------|------|----------------------|
| State1 | areturns        |      |      |                      |
| L1.    | .0790744 .0301862 | 2.62 | 0.009 | .0199105 .1382384   |
| _cons  | .7641424 .0782852 | 9.76 | 0.000 | .6107062 .9175785   |
| State2 | areturns        |      |      |                      |
| L1.    | .527953 .0857841 | 6.15 | 0.000 | .3598193 .6960867   |
| _cons  | 1.972771 .2784204 | 7.09 | 0.000 | 1.427077 2.518465   |
| sigma1 | .5895792 .0517753 |      |      | .4963544 .7003135   |
| sigma2 | 1.605333 .1262679 |      |      | 1.375985 1.872908   |
| p11    | .7530865 .0634387 |      |      | .6097999 .856167    |
| p21    | .6825357 .0662574 |      |      | .5414358 .7965346   |
```

The estimated standard deviations, reported in `sigma1` and `sigma2`, indicate that state 1 corresponds to the low-volatility periods and that state 2 corresponds to the high-volatility periods.

Example 5: An MSDR model of population health

We can apply these same methods to noneconomic data that exhibit similar periods of high and low volatility. For example, in public health and epidemiology, the process that determines the incidence of disease over time may evolve with changes in health practices.

In `mumps pc.dta`, we have monthly data on the number of new mumps cases and the interpolated population in New York City between January 1928 to December 1972. The `mumps pc` variable represents the number of new mumps cases per 10,000 residents. We apply seasonal differencing to the population-adjusted `mumps pc` variable using time-series operators, and we plot the resulting series; see [U] 11.4.4 Time-series varlists.
The data clearly show periods of high and low volatility. We fit a two-state MSDR model to the seasonally differenced dependent variable with state-dependent variances and state-dependent coefficients on the lagged dependent variable. This model does not have a constant term, so we specify suboption noconstant in switch() after the variable with a switching coefficient.

```
. use https://www.stata-press.com/data/r16/mumpspc
(Mumps data from Hipel and Mcleod (1994) with interpolated population)
. mswitch dr S12.mumpspc, varswitch switch(LS12.mumpspc, noconstant)
```

```
Markov-switching dynamic regression
Sample: 1929m2 - 1972m6 Number of obs = 521
Number of states = 2 AIC = -0.4826
Unconditional probabilities: transition HQIC = -0.4634
SBIC = -0.4336
Log likelihood = 131.7225

S12.mumpspc Coef. Std. Err. z P>|z| [95% Conf. Interval]
State1
mumpspc .420275 .0167461 25.10 0.000 .3874533 .4530968
LS12. mumpspc .9847369 .0258383 38.11 0.000 .9340947 1.035379
sigma1 .0562405 .0050954 .0470901 .067169
sigma2 .2611362 .0111191 .2402278 .2838644
p11 .762733 .0362619 .6846007 .8264175
p21 .1473767 .0257599 .1036675 .2052939
```

The output indicates that there are two distinct states; state 1 is the low-volatility state and state 2 is the high-volatility state. While the lagged seasonally differenced number of mumps cases is a significant predictor of current seasonally differenced cases, the estimates differ between states. Both states are persistent.
Markov-switching AR

In this section, we use a series of examples to describe MSAR models and the mswitch ar command.

MSAR models allow a gradual adjustment after the process changes state. These models are often used to model quarterly and lower-frequency data. An MSAR model with two state-dependent AR terms for the dependent variable that is in state $s$ at time $t$ is

$$y_t = \mu_{s_t} + x_t\alpha + z_t\beta_{s_t}$$

$$+ \phi_{1,s_t}(y_{t-1} - \mu_{s_{t-1}} - x_{t-1}\alpha - z_{t-1}\beta_{s_{t-1}})$$

$$+ \phi_{2,s_t}(y_{t-2} - \mu_{s_{t-2}} - x_{t-2}\alpha - z_{t-2}\beta_{s_{t-2}})$$

$$+ \varepsilon_{s_t}$$

where $y_t$ is the dependent variable at time $t$; $\mu_{s_t}$ is the state-dependent intercept; $x_t$ are covariates whose coefficients $\alpha$ are state invariant; $z_t$ are covariates whose coefficients $\beta_{s_t}$ are state-dependent; $\phi_{1,s_t}$ is the first AR term in state $s_t$; $\phi_{2,s_t}$ is the second AR term in state $s_t$; and $\varepsilon_{s_t}$ is the i.i.d. normal error with mean 0 and state-dependent variance. As in MSDR models, $x_t$ and $z_t$ may contain lags of $y_t$.

Note that $\mu_{s_{t-1}}$ is the intercept corresponding to the state that the process was in the previous period and that $\mu_{s_{t-2}}$ is the intercept corresponding to the state that the process was in at $t - 2$. Similarly, $\beta_{s_{t-1}}$ is the coefficient vector on $z_{t-1}$ corresponding to the state that the process was in the previous period, and $\beta_{s_{t-2}}$ is the coefficient vector on $z_{t-2}$ corresponding to the state that the process was in at $t - 2$.

In the default model fit by mswitch ar, $s = 2$ and a constant $\sigma^2$ is assumed ($\sigma^2_1 = \sigma^2_2 = \sigma^2$). In the simplest form, a single AR term is specified and the coefficient is common to both states, so four parameters, $\mu_1$, $\mu_2$, $\phi$, and $\sigma^2$, are estimated. There is no $x_t$ or $z_t$.

The number of AR terms may be increased by specifying a numlist in ar(). To allow the estimated parameters for the AR terms to vary across states, you specify option arswitch(). The number of states may be increased with option states(). To include $x_t$, you specify a varlist after the command name, and to include $z_t$, you specify option switch(). The assumption of constant variances over states may be relaxed with option varswitch.

MSAR models allow states to switch according to a Markov process, as described in Methods and formulas under Markov-switching regression models. A more complete discussion of the MSAR model is provided in Specification of Markov-switching models under Methods and formulas.
Example 6: MSAR model with switching intercepts

Hamilton (1989) and Hamilton (1994, chap. 22) fit an MSAR to the growth of quarterly U.S. real gross national product using data from 1952q1 to 1984q4. We replicate those results here using rgnp.dta.

```
. use https://www.stata-press.com/data/r16/rgnp
(Data from Hamilton (1989))
. mswitch ar rgnp, ar(1/4)
Performing EM optimization:
Performing gradient-based optimization:
Iteration 0: log likelihood = -182.54411 (not concave)
Iteration 1: log likelihood = -182.12714 (not concave)
Iteration 2: log likelihood = -181.68653
Iteration 3: log likelihood = -181.42342
Iteration 4: log likelihood = -181.26492
Iteration 5: log likelihood = -181.26339
Iteration 6: log likelihood = -181.26339

Markov-switching autoregression
Sample: 1952q2 - 1984q4 Number of obs = 131
Number of states = 2 AIC = 2.9048
Unconditional probabilities: transition HQIC = 2.9851
SBIC = 3.1023
Log likelihood = -181.26339
```

|                | Coef.  | Std. Err. | z    | P>|z|     | [95% Conf. Interval] |
|----------------|--------|-----------|------|---------|----------------------|
| rgnp           |        |           |      |         |                      |
| ar             |        |           |      |         |                      |
| L1.            | .0134871 | .1199942 | 0.11 | 0.911   | -.2216971 .2486713   |
| L2.            | -.0575211 | .137663 | -0.42 | 0.676   | -.3273357 .2122934   |
| L3.            | -.2469833 | .1069103 | -2.31 | 0.021   | -.4565235 -.037443   |
| L4.            | -.2129214 | .1105311 | -1.93 | 0.054   | -.4295583 .0037155   |
| State1_cons    | -.3588127 | .2645396 | -1.36 | 0.175   | -.8773007 .1596754   |
| State2_cons    | 1.163517  | .0745187 | 15.61 | 0.000   | 1.017463 1.309571    |
| sigma          | .7690048  | .0667396 | 15.61 | 0.000   | .6487179 .9115957    |
| p11            | .754671   | .0965189 | 7.88  | 0.000   | .5524555 .952432     |
| p21            | .0959153  | .0377362 | 2.56  | 0.011   | .0232569 .1685722    |

The output indicates that the average growth rate of U.S. real gross national product during expansions is 1.16% and during recessions is −0.36%, with each state being persistent.

Example 7: Switching AR coefficients

Continuing example 6, we now fit an MSAR with state-dependent AR coefficients to the same dataset. We include only two AR terms in each state to better estimate the parameters.
Performing EM optimization:
Performing gradient-based optimization:
Iteration 0:  log likelihood = -179.68471
Iteration 1:  log likelihood = -179.56238
Iteration 2:  log likelihood = -179.32917
Iteration 3:  log likelihood = -179.32356
Iteration 4:  log likelihood = -179.32354
Iteration 5:  log likelihood = -179.32354
Markov-switching autoregression
Sample: 1951q4 - 1984q4 Number of obs = 133
Number of states = 2 AIC = 2.8319
Unconditional probabilities: transition HQIC = 2.9114
SBIC = 3.0275
Log likelihood = -179.32354

| Coef. | Std. Err. | z     | P>|z| | [95% Conf. Interval] |
|-------|-----------|-------|-------|----------------------|
| **State1** |          |       |       |                      |
| rgnp  | L1.       | .3710719 | .1754383 | 2.12 | 0.034 | .0272191 | .7149246 |
|       | L2.       | .7002937 | .187409  | 3.74 | 0.000 | .3329787 | 1.067609 |
|       | _cons     | -.0055216 | .2057086 | -0.03 | 0.979 | -.408703 | .3976599 |
| **State2** |          |       |       |                      |
| rgnp  | L1.       | .4621503 | .1652473 | 2.80 | 0.005 | .1382714 | .7860291 |
|       | L2.       | -.3206652 | .1295937 | -2.47 | 0.013 | -.5746642 | -.0666661 |
|       | _cons     | 1.195482 | .1225987 | 9.75 | 0.000 | .9551925 | 1.435771 |
| sigma | .6677098 | .0719638 |       | 5.405648 | .8247604 |
| p11   | .3812383 | .1424841 |       | .1586724 | .6680876 |
| p21   | .3564492 | .0994742 |       | .1914324 | .5644178 |

The results indicate that state 1 has negative average growth that is different than the positive average growth in state 2. The AR coefficients for state 1 indicate that shocks will die out very slowly, while the AR coefficients for state 2 indicate that shocks will die out moderately quickly. In other words, shocks in a recession last a long time, while shocks in an expansion die out moderately quickly.

Example 8: Markov-switching regression model with constraints

mswitch can fit models subject to constraints. To facilitate the optimization, mswitch estimates a logit transform of the transition probabilities (see Methods and formulas) and a log transformation of the variance parameter. Therefore, all constraints must be specified to the transformed parameter.

In example 6, the estimated transition probability of staying in state 1 was about 0.75. In this example, we constrain that probability to be 0.75 and estimate the remaining parameters. For this case, the transformed value is \( q = -\ln(0.75/0.25) = -1.0986123 \). We use the constraint command to define this constraint; see [R] constraint.
. mswitch ar rgnp, ar(1/4) constraints(1)
Performing EM optimization:
Performing gradient-based optimization:
Iteration 0: log likelihood = -182.86708
Iteration 1: log likelihood = -182.05084 (not concave)
Iteration 2: log likelihood = -181.79995
Iteration 3: log likelihood = -181.29355
Iteration 4: log likelihood = -181.26463
Iteration 5: log likelihood = -181.26456
Iteration 6: log likelihood = -181.26456
Markov-switching autoregression
Sample: 1952q2 - 1984q4 Number of obs = 131
Number of states = 2  AIC = 2.8895
Unconditional probabilities: transition  HQIC = 2.9609
SBIC = 3.0651
Log likelihood = -181.26456
( 1) [p11]_cons = -1.098612

| Coef.  | Std. Err.  | z    | P>|z|    | [95% Conf. Interval] |
|--------|------------|------|--------|----------------------|
| rgnp   |            |      |        |                      |
| ar     |            |      |        |                      |
| L1.    | .0133924   | .1196067 | 0.11  | 0.911    | -.2210324  | .2478172 |
| L2.    | -.0591073  | .133834  | -0.44 | 0.659    | -.3214172  | .2032026 |
| L3.    | -.2473259  | .1067244 | -2.32 | 0.020    | -.456502   | -.0381499 |
| L4.    | -.2130605  | .1106088 | -1.93 | 0.054    | -.4298498  | .0037288 |
| State1 |            |      |        |                      |
| _cons  | -.3648129  | .23039 | -1.58  | 0.113    | -.816369   | .0867432 |
| State2 |            |      |        |                      |
| _cons  | 1.163125   | .0738402 | 15.75 | 0.000    | 1.018401   | 1.307849 |
| sigma  | .7682327   | .0644585 | 15.75 | 0.000    | .6517376   | .9055508 |
| p11    | .75 (constrained) | |    |        |                      |
| p21    | .0962226   | .037246 | .0439668 | .1977399 |

The point estimates are similar to those reported in example 6 while the standard errors reported here are slightly smaller.

Note that an MSAR model with no AR terms is equivalent to estimating an MSDR model, so typing

. mswitch ar rgnp, ar(0)

is the same as typing

. mswitch dr rgnp

Technical note

Achieving convergence in Markov-switching models can be difficult due to the existence of multiple local minima. Furthermore, a model with switching variance is able to generate a likelihood function that is unbounded when \( \mu = y_i \) and \( \sigma^2 \rightarrow 0 \) (Frühwirth-Schnatter 2006, chap. 6). Four methods for
overcoming convergence problems are 1) selecting an alternate optimization algorithm by using the `technique()` option; 2) using alternative starting values by specifying the `from()` option; 3) using improved starting values, which can be obtained by increasing the number of EM iterations specified in the `emiterate()` option; and 4) transforming the variables to be on the same scale.

Video example

Markov-switching models in Stata

Stored results

`mswitch` stores the following in `e()`:

Scalars

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>e(N)</code></td>
<td>number of observations</td>
</tr>
<tr>
<td><code>e(N_gaps)</code></td>
<td>number of gaps</td>
</tr>
<tr>
<td><code>e(k)</code></td>
<td>number of parameters</td>
</tr>
<tr>
<td><code>e(k_eq)</code></td>
<td>number of equations in <code>e(b)</code></td>
</tr>
<tr>
<td><code>e(k_aux)</code></td>
<td>number of auxiliary parameters</td>
</tr>
<tr>
<td><code>e(states)</code></td>
<td>number of states</td>
</tr>
<tr>
<td><code>e(ll)</code></td>
<td>log likelihood</td>
</tr>
<tr>
<td><code>e(rank)</code></td>
<td>rank of <code>e(V)</code></td>
</tr>
<tr>
<td><code>e(aic)</code></td>
<td>Akaike information criterion</td>
</tr>
<tr>
<td><code>e(hqic)</code></td>
<td>Hannan–Quinn information criterion</td>
</tr>
<tr>
<td><code>e(sbic)</code></td>
<td>Schwarz–Bayesian information criterion</td>
</tr>
<tr>
<td><code>e(tmin)</code></td>
<td>minimum time</td>
</tr>
<tr>
<td><code>e(tmax)</code></td>
<td>maximum time</td>
</tr>
<tr>
<td><code>e(emiter)</code></td>
<td>number of EM iterations</td>
</tr>
</tbody>
</table>

Macros

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>e(cmd)</code></td>
<td><code>mswitch</code></td>
</tr>
<tr>
<td><code>e(cmdline)</code></td>
<td>command as typed</td>
</tr>
<tr>
<td><code>e(eqnames)</code></td>
<td>names of equations</td>
</tr>
<tr>
<td><code>e(depvar)</code></td>
<td>name of dependent variable</td>
</tr>
<tr>
<td><code>e(switchvars)</code></td>
<td>list of switching variables</td>
</tr>
<tr>
<td><code>e(nonswitchvars)</code></td>
<td>list of nonswitching variables</td>
</tr>
<tr>
<td><code>e(model)</code></td>
<td><code>dr</code> or <code>ar</code></td>
</tr>
<tr>
<td><code>e(title)</code></td>
<td>title in estimation output</td>
</tr>
<tr>
<td><code>e(tsfmt)</code></td>
<td>format for the current time variable</td>
</tr>
<tr>
<td><code>e(timevar)</code></td>
<td>time variable specified in <code>tsset</code></td>
</tr>
<tr>
<td><code>e(tmins)</code></td>
<td>formatted minimum time</td>
</tr>
<tr>
<td><code>e(tmaxs)</code></td>
<td>formatted maximum time</td>
</tr>
<tr>
<td><code>e(vce)</code></td>
<td><code>vcetype</code> specified in <code>vce()</code></td>
</tr>
<tr>
<td><code>e(vcetype)</code></td>
<td>title use to label Std. Err.</td>
</tr>
<tr>
<td><code>e(technique)</code></td>
<td>maximization technique</td>
</tr>
<tr>
<td><code>e(p0)</code></td>
<td>unconditional probabilities</td>
</tr>
<tr>
<td><code>e(varswitch)</code></td>
<td><code>varswitch</code>, if specified</td>
</tr>
<tr>
<td><code>e(arswitch)</code></td>
<td><code>arswitch</code>, if specified</td>
</tr>
<tr>
<td><code>e(ar)</code></td>
<td>list of AR lags, if <code>ar()</code> is specified</td>
</tr>
<tr>
<td><code>e(properties)</code></td>
<td><code>b V</code></td>
</tr>
<tr>
<td><code>e(estat_cmd)</code></td>
<td>program used to implement <code>estat</code></td>
</tr>
<tr>
<td><code>e(predict)</code></td>
<td>program used to implement <code>predict</code></td>
</tr>
<tr>
<td><code>e(marginsnotok)</code></td>
<td>predictions disallowed by <code>margins</code></td>
</tr>
<tr>
<td><code>e(asbalanced)</code></td>
<td>factor variables <code>fvset</code> as <code>asbalanced</code></td>
</tr>
<tr>
<td><code>e(asobserved)</code></td>
<td>factor variables <code>fvset</code> as <code>asobserved</code></td>
</tr>
</tbody>
</table>
Matrices
- \( e(b) \) — coefficient vector
- \( e(Cns) \) — constraints matrix
- \( e(V) \) — variance–covariance matrix of the estimators
- \( e(V_{\text{modelbased}}) \) — model-based variance
- \( e(initvals) \) — matrix of initial values
- \( e(uncprob) \) — matrix of unconditional probabilities

Functions
- \( e(sample) \) — marks estimation sample

In addition to the above, the following is stored in \( r() \):

Matrices
- \( r(table) \) — matrix containing the coefficients with their standard errors, test statistics, \( p \)-values, and confidence intervals

Note that results stored in \( r() \) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

### Methods and formulas

Methods and formulas are presented under the following headings:
- Markov-switching regression models
- Markov chains
- Specification of Markov-switching models
  - Markov-switching dynamic regression
  - Markov-switching AR
- Likelihood function with latent states
- Smoothed probabilities
- Unconditional probabilities

### Markov-switching regression models

Consider the evolution of \( y_t \), where \( t = 1, 2, \ldots, T \), that is characterized by two states or regimes as in the models below

State 1: \[ y_t = \mu_1 + \phi y_{t-1} + \epsilon_t \]
State 2: \[ y_t = \mu_2 + \phi y_{t-1} + \epsilon_t \]

where \( \mu_1 \) and \( \mu_2 \) are the intercept terms in state 1 and state 2, respectively; \( \phi \) is the AR parameter; and \( \epsilon_t \) is a white noise error with variance \( \sigma^2 \). The two states model abrupt shifts in the intercept term. If the timing of switches is known, the above model can be expressed as

\[ y_t = s_t \mu_1 + (1 - s_t) \mu_2 + \phi y_{t-1} + \epsilon_t \]

where \( s_t \) is 1 if the process is in state 1 and 0 otherwise. Estimation in this case can be performed using standard procedures.

In the case of interest, we never know in which state the process is; that is to say, \( s_t \) is not observed. Markov-switching regression models specify that the unobserved \( s_t \) follows a Markov chain. In the simplest case, we can express this model as a state-dependent intercept term for \( k \) states

\[ y_t = \mu_{s_t} + \phi y_{t-1} + \epsilon_t \]

where \( \mu_{s_t} = \mu_1 \) when \( s_t = 1 \), \( \mu_{s_t} = \mu_2 \) when \( s_t = 2, \ldots, \) and \( \mu_{s_t} = \mu_k \) when \( s_t = k \). The conditional density of \( y_t \) is assumed to be dependent only on the realization of the current state \( s_t \) and is given by \( f(y_t|s_t = i, y_{t-1}; \theta) \), where \( \theta \) is a vector of parameters. There are \( k \) conditional densities for \( k \) states, and estimation of \( \theta \) is performed by updating the conditional likelihood using a nonlinear filter.
Markov chains

$s_t$ is an irreducible, aperiodic Markov chain starting from its ergodic distribution $\pi = (\pi_1, \ldots, \pi_k)$; see Hamilton (1994, chap. 22). The probability that $s_t$ is equal to $j \in \{1, \ldots, k\}$ depends only on the most recent realization, $s_{t-1}$, and is given by

$$\Pr(s_t = j | s_{t-1} = i) = p_{ij}$$

All possible transitions from one state to the other can be collected in a $k \times k$ transition matrix

$$P = \begin{bmatrix}
    p_{11} & \cdots & p_{1k} \
    p_{12} & \cdots & p_{2k} \\
    \vdots & \ddots & \vdots \\
    p_{1k} & \cdots & p_{kk}
\end{bmatrix}$$

which governs the evolution of the Markov chain. All elements of $P$ are nonnegative and each column sums to 1. For an excellent introduction on the properties of Markov chains, refer to Hamilton (1994, chap. 22) and Frühwirth-Schnatter (2006, chap. 10). For a deeper treatment, see Karlin and Taylor (1975, chap. 2 and 3).

The fact that $\sum_{j=1}^k p_{ij} = 1$ causes some numerical complications. We handle these complications by estimating functions of $p_{ij}$ and by normalizing $p_{ik}$. In particular, we estimate $q_{ij}$ in

$$p_{ij} = \frac{\exp(-q_{ij})}{1 + \exp(-q_{i1}) + \exp(-q_{i2}) + \cdots + \exp(-q_{ik-1})}$$

for $j \in \{1, 2, \ldots, k-1\}$. We normalize $p_{ik}$ by imposing

$$p_{ik} = \frac{1}{1 + \exp(-q_{i1}) + \exp(-q_{i2}) + \cdots + \exp(-q_{ik-1})}$$

The estimates of $p_{ij}$ are displayed, but the estimates of $q_{ij}$ are stored in $e(b)$.

Specification of Markov-switching models

Consider an AR(1) model given by

$$y_t = \mu + \phi y_{t-1} + \varepsilon_t$$

This model can be rewritten in terms of an AR(1) error specification as

$$y_t = \nu + e_t$$
$$e_t = \rho e_{t-1} + \varepsilon_t$$

which can be written as the single equation

$$y_t = \nu + \rho(y_{t-1} - \nu) + \varepsilon_t$$

such that $\phi = \rho$ and $\mu = \nu(1 - \rho)$. 
This result, however, does not hold in the case of Markov-switching regression models, as seen below in a simple two-state case where the constant term is state dependent. Consider the following models:

- **Model I**: \[ y_t = \mu_{s_t} + \phi y_{t-1} + \varepsilon_t \]
- **Model II**: \[ y_t = \mu_{s_t} + \phi (y_{t-1} - \mu_{s_{t-1}}) + \varepsilon_t \]

Model I is also referred to as a MSDR model or a Markov-switching intercept model \( (Krolzig 1997) \). It may consist of other switching parameters, but for simplicity, we only consider the switching-intercept case. The evolution of \( y_t \) depends on the realization of the switching intercept at time \( t \). The discrete latent state \( s_t \) that governs the value of the intercept at time \( t \) has a transition matrix

\[ P = \begin{bmatrix} p_{11} & p_{21} \\ p_{12} & p_{22} \end{bmatrix} \]

This specification allows for two possible intercepts at any given time \( t \).

By contrast, the evolution of \( y_t \) in model II depends on the value of the switching mean at its current state and its lagged value. Model II is also referred to as MSAR or Markov-switching mean \( (Krolzig 1997) \). At any given time \( t \), there are four possible values of the intercept given by

\[
\mu_1 - \rho \mu_1 \\
\mu_2 - \rho \mu_1 \\
\mu_1 - \rho \mu_2 \\
\mu_2 - \rho \mu_2
\]

which implies that models I and II do not yield equivalent representations as compared with the AR(1) model with no switching.

The MA(\( \infty \)) representation shown below better illustrates the different dynamics of \( y_t \) obtained as a result of these specifications.

- **Model I**: \[ y_t = \sum_{i=0}^{\infty} \phi^i \mu_{s_{t-i}} + \sum_{i=0}^{\infty} \phi^i \varepsilon_t \]
- **Model II**: \[ y_t = \mu_{s_t} + \sum_{i=0}^{\infty} \phi^i \varepsilon_t \]

In model I, the effect of a one-time change in state accumulates over time similar to a permanent shift in the error term \( \varepsilon_t \). In model II, the effect of a one-time change in state is the same for all time periods. Also see Hamilton \( (1993) \).

Model II allows \( y_t \) to depend on lagged values of the state \( s_{t-1} \), which in turn leads to four conditional densities. We define a new state variable \( s^*_t \) such that \( s^*_t \) is a four-state Markov chain and \( y_t \) depends only on the current state as

\[
\begin{align*}
    s^*_t &= 1 & \text{if} & & s_t = 1 & \text{and} & & s_{t-1} = 1 \\
    s^*_t &= 2 & \text{if} & & s_t = 2 & \text{and} & & s_{t-1} = 1 \\
    s^*_t &= 3 & \text{if} & & s_t = 1 & \text{and} & & s_{t-1} = 2 \\
    s^*_t &= 4 & \text{if} & & s_t = 2 & \text{and} & & s_{t-1} = 2
\end{align*}
\]
The corresponding $4 \times 4$ transition matrix is
\[
P = \begin{bmatrix}
p_{11} & 0 & p_{11} & 0 \\
p_{12} & 0 & p_{12} & 0 \\
0 & p_{21} & 0 & p_{21} \\
0 & p_{22} & 0 & p_{22}
\end{bmatrix}
\]
The conditional density of $y_t$ is given by $f(y_t|s_t^* = i, y_{t-1}; \theta)$ for $i = 1, \ldots, 4$. Also see Hamilton (1994, chap. 22). More generally, for MSAR models, $s_t^*$ is a $k(p+1)$-state Markov chain, where $p$ is the number of lagged states. Because MSAR models require larger state vectors, they are often used with low-frequency data and smaller AR lags. However, the state vector in MSDR models does not depend on the AR lags and can thus be used to accommodate high-frequency data and higher AR lags.

**Markov-switching dynamic regression**

A general specification of the MSDR model is written as
\[
y_t = \mu_s + x_t \alpha + z_t \beta_s + \epsilon_s
\]
where $y_t$ is the dependent variable, $\mu_s$ is the state-dependent intercept, $x_t$ is a vector of exogenous variables with state-invariant coefficients $\alpha$, $z_t$ is a vector of exogenous variables with state-dependent coefficients $\beta_s$, and $\epsilon_s$ is an i.i.d. normal error with mean 0 and state-dependent variance $\sigma^2_s$. $x_t$ and $z_t$ may contain lags of $y_t$.

**Markov-switching AR**

A general specification of the MSAR model is written as
\[
y_t = \mu_{st} + x_t \alpha + z_t \beta_{st} + \sum_{i=1}^{p} \phi_{i, st} (y_{t-i} - \mu_{st-i} - x_{t-i} \alpha - z_{t-i} \beta_{st-i}) + \epsilon_{st}
\]
where $y_t$ is the dependent variable at time $t$, $\mu_{st}$ is the state-dependent intercept, $x_t$ are covariates whose coefficients $\alpha$ are state-invariant, and $z_t$ are covariates whose coefficients $\beta_{st}$ are state-dependent. As in MSDR models, $x_t$ and $z_t$ may contain lags of $y_t$.

$\phi_{i, st}$ is the $i$th AR term in state $s_t$. Note that $\mu_{st-i}$ is the intercept corresponding to the state that the process was in at period $t - i$. Similarly, $\beta_{st-i}$ is the coefficient vector on $z_{t-i}$ corresponding to the state that the process was in at period $t - i$.

$\epsilon_{st}$ is the i.i.d. normal error with mean 0 and state-dependent variance.

This representation clarifies that the demeaned, lagged errors depend on the state previously occupied by the process. This dependence is not present in the MSDR model.
Likelihood function with latent states

The conditional density of $y_t$ is given by $f(y_t|s_t = i, y_{t-1}; \theta)$ for $i = 1, \ldots, k$. The marginal density of $y_t$ is obtained by weighting the conditional densities by their respective probabilities. This is written as follows:

$$f(y_t|\theta) = \sum_{i=1}^{k} f(y_t|s_t = i, y_{t-1}; \theta) \Pr(s_t = i; \theta)$$

Let $\eta_t$ denote a $k \times 1$ vector of conditional densities given by

$$\eta_t = \begin{bmatrix} f(y_t|s_t = 1; y_{t-1}; \theta) \\ f(y_t|s_t = 2; y_{t-1}; \theta) \\ \vdots \\ f(y_t|s_t = k; y_{t-1}; \theta) \end{bmatrix}$$

Constructing the likelihood function requires estimating the probability that $s_t$ takes on a specific value using the data through time $t$ and model parameters $\theta$. Let $\Pr(s_t = i|y_t; \theta)$ denote the conditional probability of observing $s_t = i$ based on data until time $t$. Then

$$\Pr(s_t = i|y_{t-1}; \theta) = \frac{f(y_t|s_t = i, y_{t-1}; \theta) \Pr(s_t = i|y_{t-1}; \theta)}{f(y_t|y_{t-1}; \theta)}$$

where $f(y_t|y_{t-1}; \theta)$ is the likelihood of $y_t$ and $\Pr(s_t = i|y_{t-1}; \theta)$ is the forecasted probability of $s_t = i$ given observation until time $t - 1$. Then

$$\Pr(s_t = 1|y_{t-1}; \theta) = \sum_{j=1}^{k} \Pr(s_t = i|s_{t-1} = j, y_{t-1}; \theta) \Pr(s_{t-1} = j|y_{t-1}; \theta)$$

Let $\xi_{t|t}$ and $\xi_{t|t-1}$ denote $k \times 1$ vectors of conditional probabilities $\Pr(s_t = i|y_t; \theta)$ and $\Pr(s_t = i|y_{t-1}; \theta)$. The likelihood is then obtained by iterating on the following equations [Hamilton (1994, chap. 22)]:

$$\xi_{t|t} = \frac{(\xi_{t|t-1} \odot \eta_t)}{1'((\xi_{t|t-1} \odot \eta_t))}$$

$$\xi_{t+1|t} = P \xi_{t|t}$$

where $1$ is a $k \times 1$ vector of 1s. The log-likelihood function is obtained as

$$L(\theta) = \sum_{t=1}^{T} \log f(y_t|y_{t-1}; \theta)$$

where

$$f(y_t|y_{t-1}; \theta) = 1'((\xi_{t|t-1} \odot \eta_t))$$
Smoothed probabilities

Let $\xi_{t|T}$, where $t < T$, denote the $k \times 1$ vector of conditional probabilities $\Pr(s_t = i|y_T; \theta)$, which represents the probability of $s_t = i$ using observations available through time $T$. The smoothed probabilities are calculated using an algorithm developed in Kim (1994).

$$
\xi_{t|T} = \xi_{t|t} \odot \{P'(\xi_{t+1|T}(\div)\xi_{t+1|t})\}
$$

where $(\div)$ denotes element-by-element division. The smoothed probabilities are obtained by iterating backwards from $t = T - 1, T - 2, \ldots, 1$.

Unconditional probabilities

The log-likelihood function has a recursive structure that starts from the unconditional state probabilities $\xi_{1|0}$. These unconditional state probabilities are unknown. There are three standard ways to obtain them.

By default, or by option p0(transition), the unconditional state probabilities are estimated from the conditional transition probabilities and the Markov structure of the model. Specifically, the vector of unconditional state probabilities is obtained as

$$
\pi = (A'A)^{-1}A'e_{k+1}
$$

where $A$ is a $(k + 1) \times k$ matrix given by

$$
A = \begin{bmatrix}
I_k - P \\
1'
\end{bmatrix}
$$

$I_k$ denotes a $k \times k$ identity matrix, and $e_k$ denotes the $k$th column of $I_k$.

Sometimes researchers prefer to estimate unconditional state probabilities by adding $k - 1$ additional parameters to the model. This method is seldom used because it requires enough observations to estimate the additional parameters. mswitch uses this method when option p0(smoothed) is specified.

Sometimes researchers prefer to set the unconditional state probabilities to $1/k$. mswitch uses this method when option p0(fixed) is specified.

References


Also see

[TS] **mswitch postestimation** — Postestimation tools for mswitch

[TS] **arima** — ARIMA, ARMAX, and other dynamic regression models

[TS] **sspace** — State-space models

[TS] **threshold** — Threshold regression

[TS] **tsset** — Declare data to be time-series data

[TS] **ucm** — Unobserved-components model

[U] **20 Estimation and postestimation commands**