Paper SAS1879-2018

Regime-Switching Models: Capturing Structural Changes in Time Series

Xilong Chen and Ji Shen, SAS Institute Inc.

ABSTRACT

Stock market conditions, government policy changes, or even weather patterns can be regarded as stochastic processes that are driven by unobserved regimes. A powerful tool to explore these behavioral patterns is the regime-switching model (RSM) that is offered in the HMM procedure and the associated action in SAS® Econometrics software. This model, which is widely used in finance, economics, science, and engineering, has two characteristics: it allows different parameter values for different regimes, and it models the transition probabilities between regimes. These characteristics enable it to fully capture the structural changes in the time series. This paper uses two examples to illustrate how you can use RSMs to better understand the regime patterns in your data and improve your economic analysis. The first example demonstrates how regime-switching autoregression (RS-AR) models help you characterize the volatility and dynamics of stock returns. The second example examines the relationship and movement between the Japanese yen and the Thai baht by using regime-switching regression (RS-REG) models.

INTRODUCTION

Many time series data, such as stock market conditions, government policy changes, weather patterns, and so on, follow different dynamics in different time periods; this behavior is called structural change or regime switching. One type of model for this kind of behavior is the regime-switching model (RSM). RSMs enable you to assign different sets of parameter values to different regimes and model the transition probabilities between regimes. They have been powerful tools for sequential data analysis (including time series analysis) in finance, economics, science, and engineering for several decades. The HMM procedure and the associated action in SAS Econometrics software support RSMs.

The two examples in this paper demonstrate how you can easily specify RSMs in the HMM procedure, perform model selection, and evaluate the predictability performance. The first example shows how regime-switching autoregression models enable you to characterize the dynamics of stock returns, identify the market states, and forecast the value at risk (VaR). The second example uses regime-switching regression to explore the relationship and movement between the Japanese yen and another East Asian currency, the Thai baht.

REGIME-SWITCHING AUTOREGRESSION MODELS

Although bull market and bear market are well-known terms, these market states cannot be directly observed; they can be interpreted only from the observed stock prices and other financial data. In this example, regime-switching autoregression (RS-AR) models are applied to the S&P 500 index weekly returns to analyze the market states and dynamics and to forecast the VaR. The forecasting performance is also assessed.

In the RS-AR models, the observed variables (weekly returns) follow different autoregressive (AR) models in different regimes (market states), and the regimes follow a Markov chain: that is, the transition probability from the current regime to the next regime does not depend on previous regimes.

The S&P 500 index weekly returns from January 10, 1950, to December 15, 2017, are considered (S&P Dow Jones Indices LLC 2018). The original daily data are retrieved from the FRED database at the Federal Reserve Bank of St. Louis and stored in the data set **sp500Original**. The sample is divided into two periods: the in-sample period includes the 2,665 weekly returns before January 1, 2003, and the out-of-sample period includes the remaining 754 weekly returns. The following statements generate the weekly returns from the daily close price and save the client-side data to the server:

```
%let cutDate = '31DEC2002'd;
data sp500w sp500wIn;
   set sp500Original;
   format date MMDDYY10.;
   retain cumReturn 0;
   return = (log(price)-log(lag(price)))*100;
   if(return~=.) then cumReturn + return;
   if(mod(_N_,5)=1 and _N_>1) then do;
      returnw = cumReturn;
      w + 1;
      output sp500w;
      if (date<=&cutDate.) then output sp500wIn;
      cumReturn = 0;
   end;
   keep w date returnw;
run;
data cashmm.sp500wIn; set sp500wIn; run;
data cashmm.sp500w; set sp500w; run;
```

To model the weekly returns, how many market states should be considered? How many AR lags should a regime include? These are common questions in the model selection process. In this paper, the best model is selected using Akaike's information criterion (AIC): the smaller the AIC, the better the model.

An RS-AR(p) model estimates the likelihood of $(T-p)$ observations conditional on the first p observations, where T is the sample size. To compare the AICs for the same number of observations among different RS-AR(p) models with different p values, you need to adjust the sample start dates. In this example, the AICs based on the 2,663 weeks of returns before January 1, 2003, are compared. The following macro variables specify how many weeks should be skipped for each RS-AR(p) model:

```
%let w0 = '31Jan1950'd; * for AR(0), skip first 2 weeks;
%let w1 = '24Jan1950'd; * for AR(1), skip first week;
%let w2 = '17Jan1950'd; * for AR(2), skip no week;
```

In the HMM procedure, when you specify TYPE=AR, NSTATE=k, and YLAG=p in the MODEL statement, you specify the k-state RS-AR(p) model. The estimation of RS-AR models is nontrivial. It is a nonlinear optimization problem. For an RS-AR model, there might be many local optima. To increase the chance of finding the global optimum, the following measures are applied:

1. The initial parameter values are very important. In the estimation, for k-state RS-AR(0) models, where $k = 2, \ldots, 10$, the initial values are obtained randomly by the HMM procedure, and there is no need to use the INITIAL statement. However, for k-state RS-AR(p) models, where $k = 2, \ldots, 10, p = 1, 2$, using the random initial values often leads to bad solutions; an effective approach is to use the INITIAL statement to set the initial values as the final parameter estimates from the corresponding k-state RS-AR($p-1$) models.
2. Although the maximum likelihood (ML) method is commonly applied to estimating RS-AR models, in theory the likelihood of an RS-AR model is unbounded and "the ML estimator as a global maximizer of the likelihood function does not exist" (Frühwirth-Schnatter 2006). The introduction of the proper prior distribution of the parameters in the MAP method can solve the unboundedness problem. You specify METHOD=MAP in the MODEL statement to apply the MAP method. In this example, several flat priors for parameters are applied.
3. A global optimization mechanism, multistart, can be used. When you specify MULTISTART=1 in the OPTIMIZE statement, multistart mode is turned on. This mechanism checks thousands of initial values and finds the best solution among dozens of local optima.

The following statements estimate 2- to 10-state RS-AR(0) models and save the information criterion for model selection. The OUTMODEL= option in the SCORE statement stores model information and parameter estimates from the in-sample data, and later they are applied to score the out-of-sample data (for example, to forecast VaRs). For the best results, multistart mode is strongly recommended.

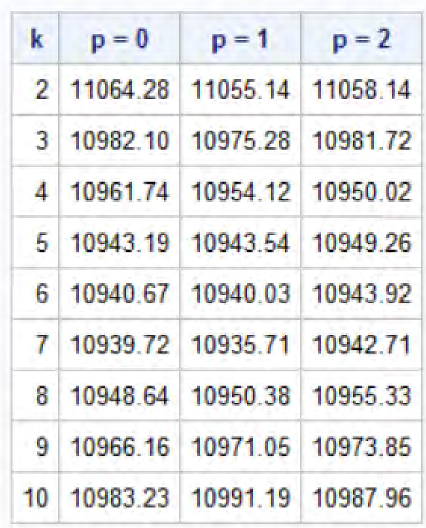
```
* estimate k-state RS-AR(0) models, k from kStart to kEnd;
* when qMultiStart=0, multistart mode is off;
* when qMultiStart=1, multistart mode is on;
%macro estimateRSAR0(kStart, kEnd, qMultiStart);
   %let p = 0;
   %do k = &kStart. %to &kEnd.;
      proc hmm data=cashmm.sp500wIn(where=(date>=&&w&p.))
               outstat=cashmm.sp500StatIn_k&k._p&p.;
         id time=date;
         model returnw / type=ar nstate=&k. ylag=&p. method=map;
         optimize printLevel=3 printIterFreq=1 algorithm=interiorpoint
            multistart=&qMultiStart.;
         score outmodel=cashmm.sp500ModelIn&k._&p.;
         prior tpm~dir(J(&k.,&k.,1)),
            musigma~niw(J(&k.,1+&p.,0),J(&k.,1,10),
                        J(&k.,1,0.00001)@I(1+&p.),J(&k.,1,4.00001));
      run;
      data sp500StatIn_p&p._k&k.;
         set cashmm.sp500StatIn_k&k._p&p.;
         nStates=&k.; lag=&p.;
         keep nStates lag logLikelihood AIC AICC BIC HQC;
      run;
   %end;
   data sp500SelectModelIn_p&p.;
      set sp500StatIn_p&p._k&kStart. - sp500StatIn_p&p._k&kEnd.;
   run;
%mend estimateRSAR0;

* estimate k-state RS-AR(0), k from 2 to 10, with multistart mode on;
* be aware that the following macro might take tens of hours to finish;
* uncomment it to run;
* even if you do not run this macro here, later you still have a chance to get
* estimates of RS-AR(0) models;
* %estimateRSAR0(kStart=2, kEnd=10, qMultiStart=1);
```

For each of the 18 RS-AR(1) and RS-AR(2) models, the INITIAL statement with the corresponding initial parameter values is specified. In the following code, only one example, a 7-state RS-AR(1) model, is listed to illustrate how to estimate these RS-AR(1) and RS-AR(2) models. The SAS® code for other model estimations is omitted here to save space; you can find it online. Multistart mode is turned off, because it is not necessary in this example for RS-AR(1) and RS-AR(2) models.

```
* for 7-state RS-AR(1) model;
* using parameter estimates of 7-state RS-AR(0) as initial values;
%macro estimateRSAR(k, p, qMultiStart);
   ods output FinalParameterEstimates=myParmEst TPM=myTPM ISPV=myISPV;
   proc hmm data=cashmm.sp500wIn(where=(date>= &&w&p.))
            outstat=cashmm.sp500StatIn_&k._&p.;
      id time=date;
      model returnw / type=ar ylag=&p. nstate=&k. method=map;
      optimize printLevel=3 printIterFreq=1 algorithm=interiorpoint
         Multistart=&qMultiStart.;
      score outmodel=cashmm.sp500ModelIn&k._&p.;
      prior tpm~dir(J(&k.,&k.,1)),
         musigma~niw(J(&k.,1+&p.,0),J(&k.,1,10),
                     J(&k.,1,0.00001)@I(1+&p.),J(&k.,1,4.00001));
      initial tpm={0.67872 0.00000 0.00000 0.00000 0.00000 0.02177 0.29951,
                   0.00000 0.10799 0.00000 0.05413 0.00000 0.83787 0.00000,
                   0.00000 0.09589 0.90411 0.00000 0.00000 0.00000 0.00000,
                   0.59099 0.00000 0.00000 0.16756 0.02889 0.00000 0.21257,
                   0.00000 0.42665 0.48353 0.00759 0.00000 0.00000 0.08223,
                   0.00000 0.30024 0.00000 0.00000 0.03013 0.66964 0.00000,
                   0.02400 0.00000 0.00000 0.24036 0.00000 0.00380 0.73184},
         const={0.61433, 2.47350, -0.18333, 1.73559, -5.19095, -0.61715, -0.47292},
         cov={0.55251, 2.04663, 20.20570, 0.74352, 1.54215, 3.11860, 1.59261};
   run;
   data sp500StatIn_&k._&p.;
      set cashmm.sp500StatIn_&k._&p.;
      nStates=&k.; lag=&p.;
      keep nStates lag logLikelihood AIC AICC BIC HQC;
   run;
%mend estimateRSAR;
%estimateRSAR(k=7, p=1, qMultiStart=0);
```

After running the estimation of all 27 of the 2- to 10-state RS-AR(0) to AR(2) models, you get Table 1, which displays the AICs (the printing code is omitted here). As the table shows, the smallest AIC corresponds to the 7-state RS-AR(1) model.



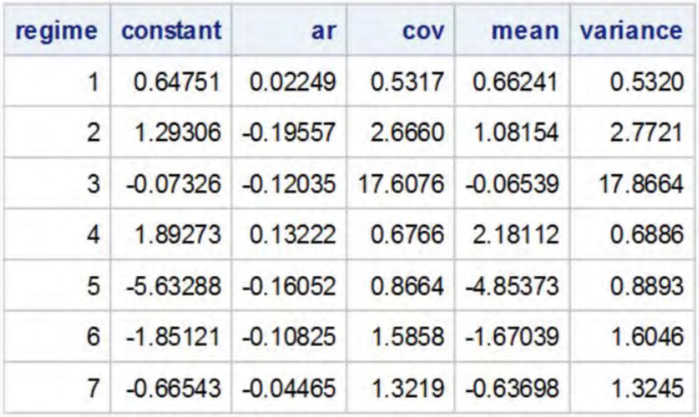
k	p = 0	p = 1	p = 2
2	11064.28	11055.14	11058.14
3	10982.10	10975.28	10981.72
4	10961.74	10954.12	10950.02
5	10943.19	10943.54	10949.26
6	10940.67	10940.03	10943.92
7	10939.72	10935.71	10942.71
8	10948.64	10950.38	10955.33
9	10966.16	10971.05	10973.85
10	10983.23	10991.19	10987.96

Table 1. AICs for 27 RS-AR Models

The 7-state RS-AR(1) model contains 70 parameters. The following statements print the observation parameters and calculate the unconditional mean and variance of weekly returns for each regime (which are displayed in the columns “mean” and “variance”), as shown in Table 2. The values of the AR coefficients show that the process in each regime is stationary. The standard errors of the AR parameter estimates (which are omitted here) show that the AR parameters are significant at the 10% significance level in four out of seven regimes.

```
%Let k=7;
%Let p=1;
%Let nTPM = %SYSEVALF(&k.*&k.);
%Let nObsParms = %SYSEVALF(3*&k.);
%Let nParms = %SYSEVALF(&nObsParms.+&nTPM.);
data obsParms;
   set myParmEst;
   array myest(&nObsParms.) _temporary_;
   retain myest:;
   if _N_ > (&nTPM.) then do;
      myest[_N_-(&nTPM.)] = Estimate;
   end;
   if _N_ = &nParms. then
      do regime=1 to &k.;
```

```
         constant = myest[regime];
         ar = myest[&k.+regime];
         cov = myest[2*&k.+regime];
         mean = constant/(1-ar);
         variance = cov/(1-ar**2);
         output;
      end;
   keep regime constant ar cov mean variance;
run;
proc print data=obsParms noobs; run;
```



regime	constant	ar	cov	mean	variance
1	0.64751	0.02249	0.5317	0.66241	0.5320
2	1.29306	-0.19557	2.6660	1.08154	2.7721
3	-0.07326	-0.12035	17.6076	-0.06539	17.8664
4	1.89273	0.13222	0.6766	2.18112	0.6886
5	-5.63288	-0.16052	0.8664	-4.85373	0.8893
6	-1.85121	-0.10825	1.5858	-1.67039	1.6046
7	-0.66543	-0.04465	1.3219	-0.63698	1.3245

Table 2. Observation Parameters and Unconditional Means and Variances

According to the unconditional mean and variance of weekly returns for each regime, you can draw the Gaussian kernel for each regime and compare them with the histogram of weekly returns by using the following statements. As shown in Figure 1, roughly speaking, three regimes (regimes 1, 2, and 4) could be considered bull market states, where the mean of weekly returns is significantly positive and the risk is relatively low, and four regimes (regimes 3, 5, 6, and 7) could be considered bear market states, where the mean of weekly returns is significantly negative (regimes 5, 6, and 7) or the risk (measured by the unconditional variance) is extremely high (regime 3).

```
data muSigma;
   set obsParms end=eof;
   array mu(&k.); array sd(&k.);
   retain mu: sd:;
   mu(_N_) = mean;
   sd(_N_) = sqrt(variance);
   if eof then output;
run;

%macro plotLearning(myParm,k,myData,myColumn);
   data _NULL_;
      set &myParm.;
      %do i = 1 %to &k.;
         call symputx("mu&i.",mu&i.,'G');
         call symputx("sigma&i.",sd&i.,'G');
      %end;
   run;
   proc sgplot data=&myData.;
      refline 0 / axis=x lineattrs=(thickness=3);
      histogram &myColumn. / nbins=200;
      %do i = 1 %to &k.;
         density &myColumn. / type=normal(mu=&&mu&i. sigma=&&sigma&i.)
            name="regime&i."
            legendlabel="Expected Gaussian Dist. for Regime &i.";
      %end;
      keylegend %do i = 1 %to &k.; "regime&i." %end;;
   run;
%mend plotLearning;

%plotLearning(muSigma,&k.,sp500wIn,returnw);
```

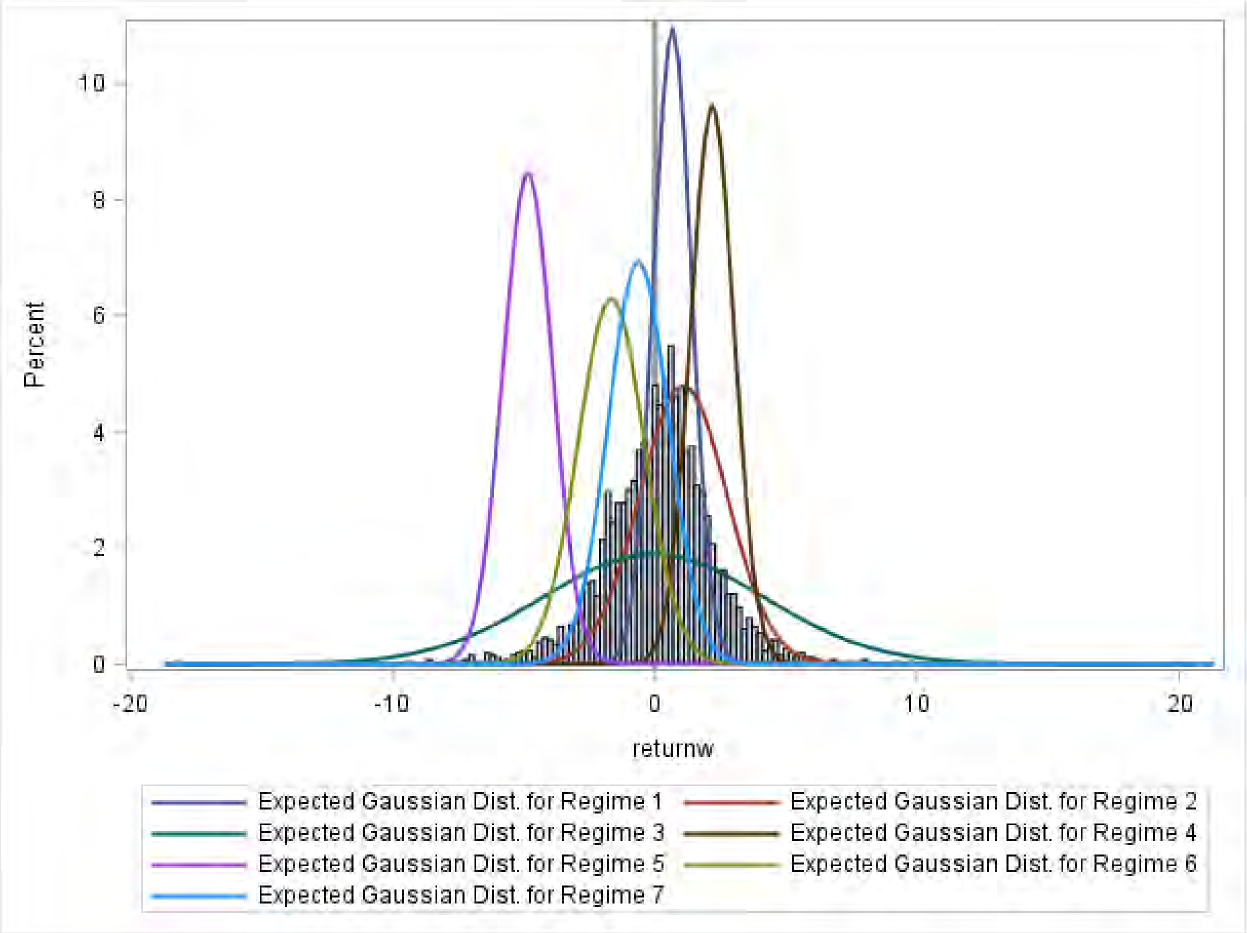
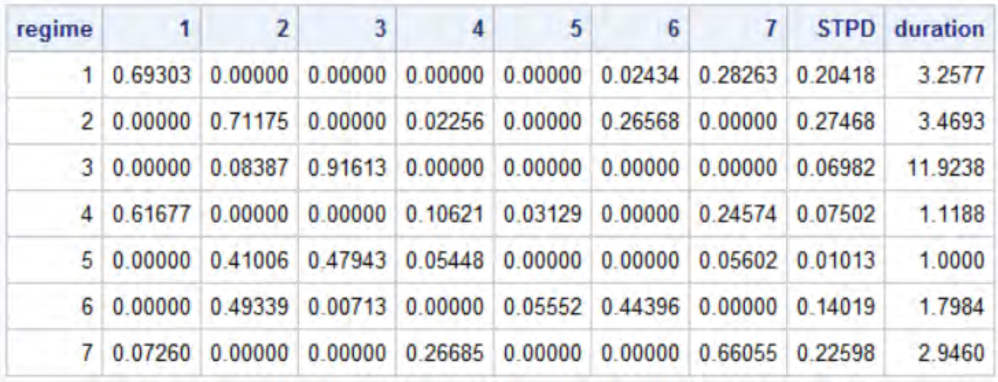


Figure 1. Gaussian Kernels for Seven Regimes

The following statements display the transition probabilities, the steady-state probability distribution (in the column “STPD”), and the expected duration of each regime (in the column “duration”) in Table 3. Regime 5 is an exceptional bear market state: there is only a 1% chance that the market falls in that regime; and even when the market falls in that regime, it quickly recovers. A very bullish market (regime 4) or a very bearish market (regime 6) has about a 10% chance, and neither market lasts long. The market has a 50% chance to be somewhat bullish (regime 1 and 2) or a 20% chance to be somewhat bearish (regime 7). Regime 3 might be highly related to the financial crisis (as discussed later): the risk is extremely high; it has a 7% chance of occurring; and when it does occur, it takes a long time to recover (the expected duration is about 12 weeks).

```
data tp;
   set myTPM;
   array tpm(&k.) state1-state&k.;
   duration = 1/(1-tpm(_N_));
   keep state: duration;
run;
proc print data=tp noobs label; label state='regime'; run;
```



regime	1	2	3	4	5	6	7	STPD	duration
1	0.69303	0.00000	0.00000	0.00000	0.00000	0.02434	0.28263	0.20418	3.2577
2	0.00000	0.71175	0.00000	0.02256	0.00000	0.26568	0.00000	0.27468	3.4693
3	0.00000	0.08387	0.91613	0.00000	0.00000	0.00000	0.00000	0.06982	11.9238
4	0.61677	0.00000	0.00000	0.10621	0.03129	0.00000	0.24574	0.07502	1.1188
5	0.00000	0.41006	0.47943	0.05448	0.00000	0.00000	0.05602	0.01013	1.0000
6	0.00000	0.49339	0.00713	0.00000	0.05552	0.44396	0.00000	0.14019	1.7984
7	0.07260	0.00000	0.00000	0.26685	0.00000	0.00000	0.66055	0.22598	2.9460

Table 3. Transition Probabilities, Steady-State Probability, and Expected Duration

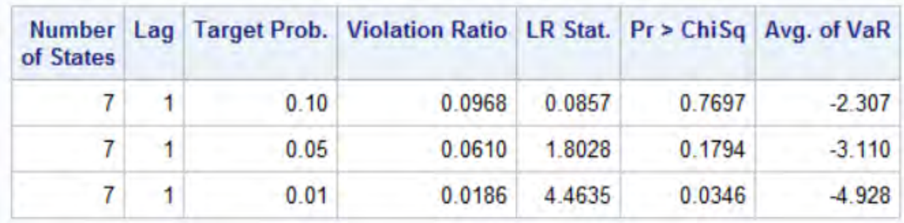
Next, the INMODEL= option in the SCORE statement is used to forecast the VaRs at different levels, such as 1%, 5%, and 10%, as shown in Table 4. The following macros can predict the VaRs for the out-of-sample period and evaluate the predictive performance by the likelihood ratio (LR) test for the unconditional coverage of the VaR forecast (Kuester, Mittnik, and Paolella 2006).

```
%let alpha1 = 0.80;
%let alpha2 = 0.90;
%let alpha3 = 0.98;
%macro VaREstimation(dsModel,k,p,iStart,iEnd,oosStart,oosEnd,dsForecastPrefix);
   %do i = &iStart. %to &iEnd.;
      proc hmm data=cashmm.sp500w(where=(date>=&&w&p.))
               outstat=cashmm.sp500Stat&k._&p.;
         score inmodel=&dsModel.;
         forecast out=cashmm.&dsForecastPrefix.&i. alpha=&&alpha&i. online;
         decode out=cashmm.sp500Decode&k._&p.;
      run;
      data &dsForecastPrefix.&i.;
         set cashmm.&dsForecastPrefix.&i.;
      run;
      proc sort data=&dsForecastPrefix.&i.; by date; run;
      data &dsForecastPrefix.&i.;
         set &dsForecastPrefix.&i. (FIRSTOBS=&oosStart. OBS=&oosEnd.
                                    keep=date returnw_Q1
                                    rename=(returnw_Q1=returnw_Q1_&k.&p.&i.));
         time=_N_; nStates = &k; Lag = &p;
      run;
   %end;
%mend VaREstimation;

%macro VaREvaluate(dsForecastPrefix,k,p,iStart,iEnd,n,dsOut);
   data sp500wout;
      set sp500w;
      if (date>&cutDate.) then output;
   run;
   data forecastData;
      set sp500wout;
      time=_N_;
   run;
   data forecastData;
      merge &dsForecastPrefix.: forecastData;
      by time;
   run;
   data &dsOut.;
      set forecastData;
      retain %do i = &iStart. %to &iEnd.; cviol&k.&p.&i. 0 %end; ;
      retain %do i = &iStart. %to &iEnd.; creturnw&k.&p.&i. 0 %end; ;
      %do i = &iStart. %to &iEnd.;
         if returnw le returnw_Q1_&k.&p.&i. then do;
            cviol&k.&p.&i.=cviol&k.&p.&i.+1;
         end;
         creturnw&k.&p.&i. = creturnw&k.&p.&i. + returnw_Q1_&k.&p.&i.;
      %end;
      if _N_ = &n. then do;
         %do i = &iStart. %to &iEnd.;
            Norminal = (1-&&alpha&i.)/2;
            Viol = cviol&k.&p.&i. / &n.;
            LR = 2*(cviol&k.&p.&i.*log(Viol)+(&n.-cviol&k.&p.&i.)*log(1-Viol)
               -(cviol&k.&p.&i.*log(Norminal)+(&n.-cviol&k.&p.&i.)
                *log(1-Norminal)));
            pValue = 1 - cdf("CHISQUARE", LR, 1);
            meanVaR = creturnw&k.&p.&i. / &n.;
            output;
         %end;
      end;
      label Norminal='Target Prob.' Viol='Violation Ratio' LR='LR Stat.'
            pValue='Pr > ChiSq' meanVaR='Avg. of VaR' nStates='Number of States'
            lag='Lag';
      keep nStates Lag Norminal Viol LR pValue meanVaR;
   run;
   proc print data=&dsOut. noobs label;  format Viol LR pValue meanVaR 6.4; run;
%mend VaREvaluate;

%macro VaR(k,p);
   %VaREstimation(cashmm.sp500ModelIn&k._&p.,&k,&p,1,3,
      %eval(2665-(2-&p.)),%eval(3418-(2-&p.)),sp500Forecastk&k._p&p.);
   %VaREvaluate(sp500Forecastk&k._p&p.,&k,&p,1,3,754,VaR_outputk&k._p&p.);
%mend VaR;

%VaR(7,1);
```

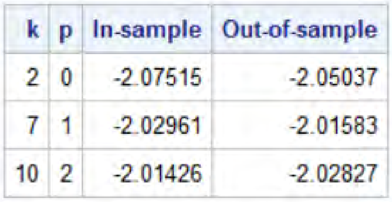


Number of States	Lag	Target Prob.	Violation Ratio	LR Stat.	Pr > ChiSq	Avg. of VaR
7	1	0.10	0.0968	0.0857	0.7697	-2.307
7	1	0.05	0.0610	1.8028	0.1794	-3.110
7	1	0.01	0.0186	4.4635	0.0346	-4.928

Table 4. Predictive Performance of VaR Forecasts

According to the *p*-values in the “Pr > ChiSq” column in Table 4, at the 1% significance level, no tests can reject the null hypothesis that the number of violations is correct. Hence, the 7-state RS-AR(1) model has the correct unconditional coverage for the 1%, 5%, and 10% VaR forecasts.

In fact, besides the good predictability of the tail of the distribution of the weekly returns as shown in the VaR forecast analysis, the 7-state RS-AR(1) model also provides a very good prediction of the whole distribution of the weekly returns, which can be shown by comparing the average weekly log likelihoods of the in-sample period and the out-of-sample period. The SAS code is omitted here. As shown in Table 5, for the 7-state RS-AR(1) model, the average weekly log likelihood in the out-of-sample period is even better than in the in-sample period. Compared to both the simplest model, the 2-state RS-AR(0) model, which has the fewest parameters and the worst in-sample fit, and the most complex model, the 10-state RS-AR(2) model, which has the most parameters and the best in-sample fit, the 7-state RS-AR(1) model has the best out-of-sample forecast ability (that is, the largest average weekly log likelihood in the out-of-sample period).



k	p	In-sample	Out-of-sample
2	0	-2.07515	-2.05037
7	1	-2.02961	-2.01583
10	2	-2.01426	-2.02827

Table 5. Comparison of Average Weekly Log Likelihoods

Finally, the decoded regimes provide a historical view of what happened, given all the available data. The data set **cashmm.sp500Decode7_1**, as a by-product, is generated by the DECODE statement in the VaR forecast. The following statements plot the decoded regimes, as shown in Figure 2. Regimes 1, 4, and 7 seem to belong to one group, the bullish market: although there are very upward-moving (regime 4) and somewhat downward-moving (regime 7) days, the main trend is upward (regime 1). Regimes 2, 3, 5, and 6 seem to belong to another group, the bearish market: although there are some upward-moving days (regime 2), the main trend is downward (regime 6), even very downward (regime 5) or very volatile (regime 3). Regimes 3 and 5 seem to be indicators of a financial crisis; notice their appearance around the 1987 financial crisis, the 1997 Asian financial crisis, the Y2K crash, and the 2008 financial crisis. When the market is in the bullish state, it lasts for a long time before switching to the bearish state, or vice versa.

```
proc sgplot data=cashmm.sp500Decode7_1(where=(state~=.));
   scatter x=date y=state / group=state;
run;
```

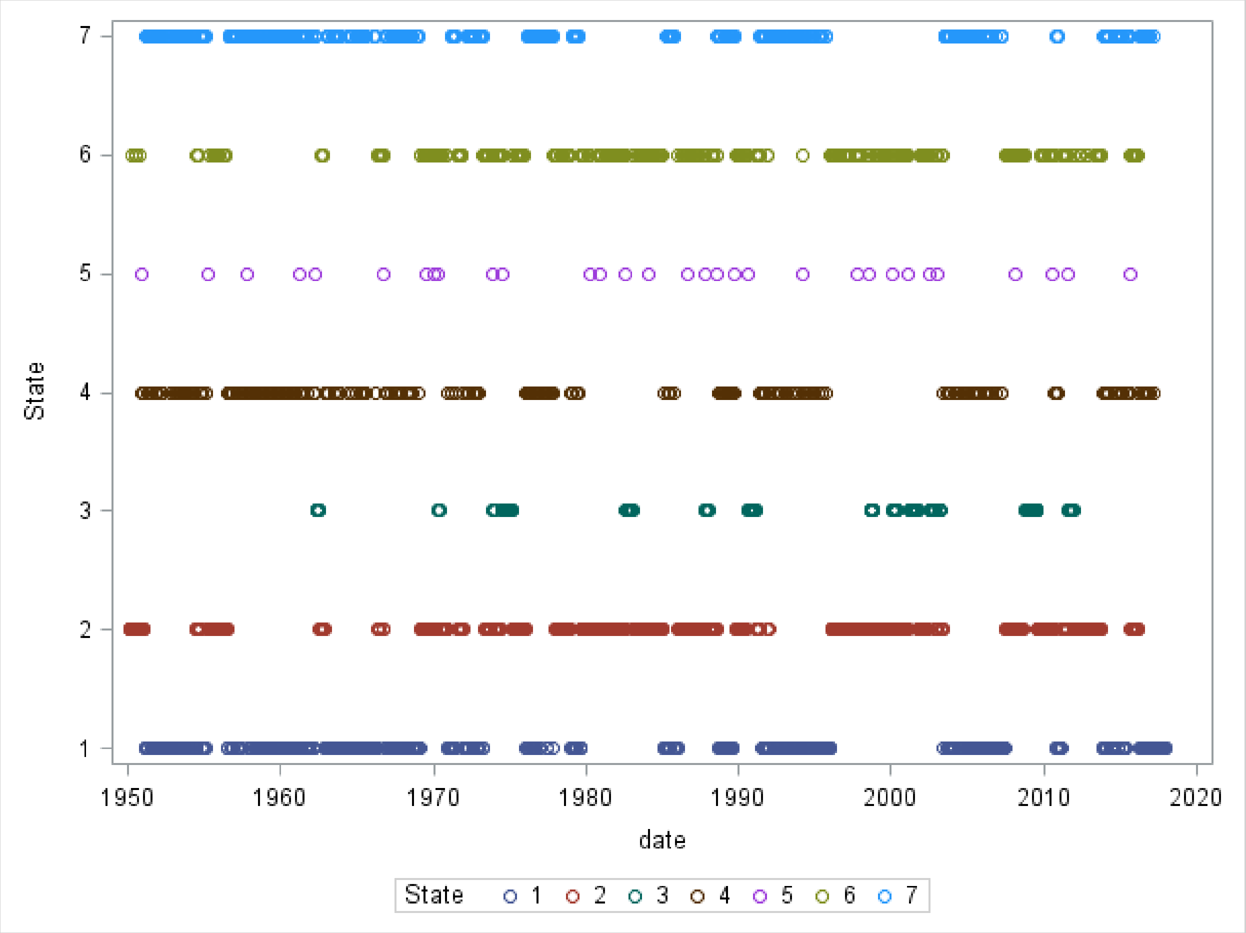


Figure 2. Decoded Regime for Each Week

REGIME-SWITCHING REGRESSION MODELS

A paper by Kim, Min, McDonald, and Hwang (2012) uses the regime-switching regression (RS-REG) models to find that there are synchronization periods (one regime) and desynchronization periods (the other regime) between the Swiss franc exchange rates of floating East Asian currencies and the exchange rate between the Swiss franc and the Japanese yen. This example follows ideas of this paper, using different data and focusing on only one East Asian currency, the Thai baht, and shows how you can use the HMM procedure to estimate the RS-REG models and interpret the results.

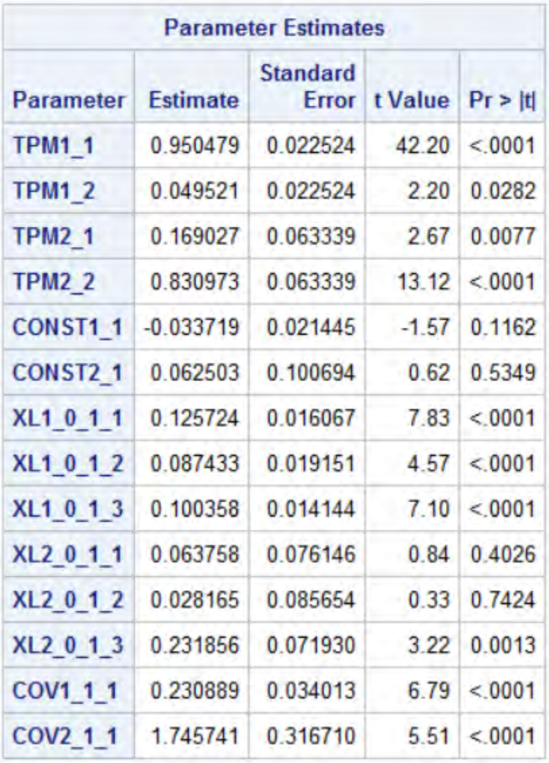
The daily exchange rate data (Board of Governors of the Federal Reserve System (US) 2018a, 2018b, 2018c, 2018d), stored in the data set **ero**, are retrieved from the FRED database, including US dollar (USD) exchange rates from January 1999 to January 2018 for the Australian dollar (AUD), the euro (EUR), the Japanese yen (JPY), and the Thai baht (THB). The following statements prepare the weekly returns of exchange rates for the RS-REG model. The variable DEXTHUSw is the return of the THB-USD exchange rate; DEXJPUSw is the return of the JPY-USD exchange rate; DEXEUUSw is the return of the EUR-USD exchange rate; and DEXALUSw is the return of the AUD-USD exchange rate.

```
data er;
   set ero(where=(DEXUSEU~=.));
   array xr [4] DEXTHUS  DEXJPUS  DEXEUUS  DEXALUS;
   array xrr[4] DEXTHUSr DEXJPUSr DEXEUUSr DEXALUSr;
   array xrc[4] DEXTHUSc DEXJPUSc DEXEUUSc DEXALUSc;
   array xrw[4] DEXTHUSw DEXJPUSw DEXEUUSw DEXALUSw;
   DEXEUUS = 1 / DEXUSEU;
   DEXALUS = 1 / DEXUSAL;
   do i = 1 to 4;
      xrr[i] = (log(xr[i])-log(lag(xr[i])))*100;
      if(xrr[i]~=.) then xrc[i] + xrr[i];
   end;
   if(mod(_N_,5)=1 and _N_>1) then do;
      do i = 1 to 4; xrw[i] = xrc[i]; end;
      w + 1;
      output er;
      do i = 1 to 4; xrc[i] = 0; end;
   end;
   keep date DEXTHUSw DEXJPUSw DEXEUUSw DEXALUSw;
run;
data mycas.er; set er; run;
```

The following statements estimate the bi-state RS-REG model. You specify the regression in the MODEL statement: on the left-hand side is the dependent variable, DEXTHUSw, and on the right-hand side are the regressors, DEXJPUSw, DEXEUUSw, and DEXALUSw. You specify TYPE=REG in the MODEL statement for the RS-REG model. NSTATE=2 in the MODEL statement indicates that there are two regimes. The SMOOTH statement outputs the smoothed probabilities of the two regimes.

```
proc hmm data=mycas.er;
   id time=date;
   model DEXTHUSw = DEXJPUSw DEXEUUSw DEXALUSw / type=reg nstate=2;
   smooth out=mycas.erSmooth;
run;
```

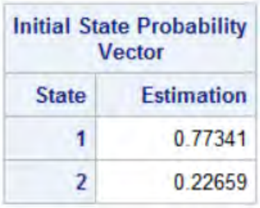
The parameter estimates are shown in Table 6. XLk_l_i_j is the parameter for the jth regressor at lag l in the ith equation for the kth regime. Because this model has two regimes, k can take the value 1 or 2. This is a univariate model, so there is only one equation, and i is always 1. The regression projects only the current regressor; hence lag l is always 0. There are three regressors; in sequence, $j = 1$ for the first regressor, DEXJPUSw, then 2 for DEXEUUSw, and 3 for DEXALUSw. The main interest is in XLk_0_1_1, the relationship between THB-USD and JPY-USD. In the first regime, XL1_0_1_1 is significant at the 5% significance level, indicating that regime 1 is the synchronization regime, where the THB-USD exchange rate is influenced by the JPY-USD exchange rate; and XL2_0_1_1 is not significant at the 5% significance level, indicating that regime 2 is the desynchronization regime, where the THB-USD exchange rate is not influenced by the JPY-USD exchange rate. This result confirms what has been found by Kim et al. (2012). It is also worth mentioning that the covariance of innovations in the synchronization regime is much lower than in the desynchronization regime, which is opposite to another finding of Kim et al. (2012): there is greater volatility during the synchronization period than during the desynchronization period. It might be because the model in this paper has different regressors and uses different data. Further discussion of this topic is beyond the scope of this paper.



Parameter Estimates				
Parameter	Estimate	Standard Error	t Value	Pr > |t|
TPM1_1	0.950479	0.022524	42.20	<.0001
TPM1_2	0.049521	0.022524	2.20	0.0282
TPM2_1	0.169027	0.063339	2.67	0.0077
TPM2_2	0.830973	0.063339	13.12	<.0001
CONST1_1	-0.033719	0.021445	-1.57	0.1162
CONST2_1	0.062503	0.100694	0.62	0.5349
XL1_0_1_1	0.125724	0.016067	7.83	<.0001
XL1_0_1_2	0.087433	0.019151	4.57	<.0001
XL1_0_1_3	0.100358	0.014144	7.10	<.0001
XL2_0_1_1	0.063758	0.076146	0.84	0.4026
XL2_0_1_2	0.028165	0.085654	0.33	0.7424
XL2_0_1_3	0.231856	0.071930	3.22	0.0013
COV1_1_1	0.230889	0.034013	6.79	<.0001
COV2_1_1	1.745741	0.316710	5.51	<.0001

Table 6. Parameter Estimates of RS-REG Model on CHF-PHP

In this model, the Markov chain is assumed to be stationary, and the initial state probability vector (ISPV) is the same as the steady-state probability distribution. The estimates of ISPV are shown in Table 7. The synchronization regime has about a 77% chance. This is confirmed by the smoothed probabilities plot in Figure 3.



Initial State Probability Vector	
State	Estimation
1	0.77341
2	0.22659

Table 7. Initial State Probability Vector (Steady-State Probability Distribution)

The following statements plot the smoothed probabilities of the synchronization regime. Figure 3, which can be compared to Figure 2 in Kim et al. (2012), shows that the degree of synchronization is high for the Thai baht.

```
proc sgplot data=mycas.erSmooth;
   series x=date y=state1;
run;
```

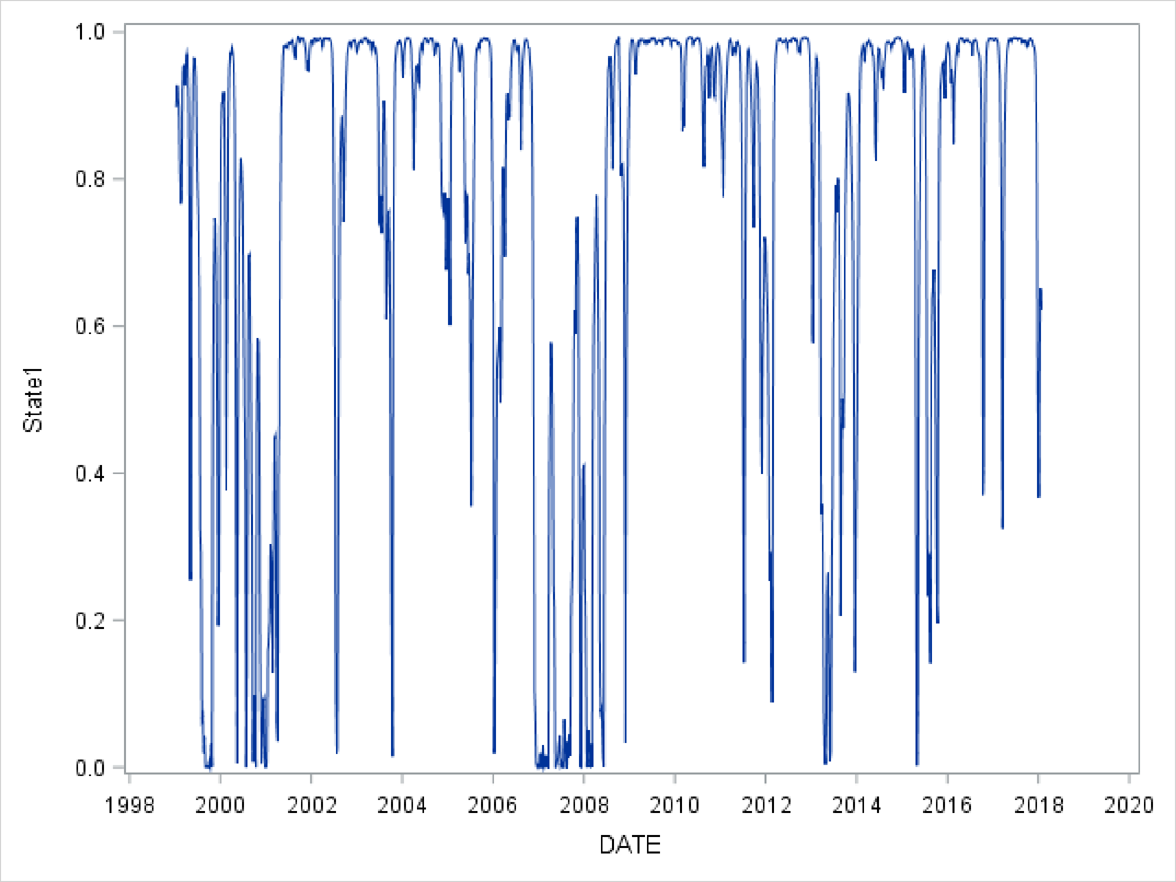


Figure 3. Smoothed Probabilities of Synchronization Regime

CONCLUSION

By using the HMM procedure, you can exploit regime-switching models to better understand the regime patterns in your data and improve your economic analysis. Beyond the topics discussed in this paper, PROC HMM also provides support for Gaussian HMMs and Gaussian mixture HMMs, which are two other powerful tools for time series analysis. You can also use the HMM procedure to analyze cross-sectional time series data (also known as panel data).

REFERENCES

Board of Governors of the Federal Reserve System (US) (2018a). Japan/U.S. Foreign Exchange Rate [DEXJPUS]. Retrieved from FRED database, Federal Reserve Bank of St. Louis. https://fred.stlouisfed.org/series/DEXJPUS.

Board of Governors of the Federal Reserve System (US) (2018b). Thailand/U.S. Foreign Exchange Rate [DEXTHUS]. Retrieved from FRED database, Federal Reserve Bank of St. Louis. https://fred.stlouisfed.org/series/DEXTHUS.

Board of Governors of the Federal Reserve System (US) (2018c). U.S./Australia Foreign Exchange Rate [DEXUSAL]. Retrieved from FRED database, Federal Reserve Bank of St. Louis. https://fred.stlouisfed.org/series/DEXUSAL.

Board of Governors of the Federal Reserve System (US) (2018d). U.S./Euro Foreign Exchange Rate [DEXUSEU]. Retrieved from FRED database, Federal Reserve Bank of St. Louis. https://fred.stlouisfed.org/series/DEXUSEU.

Frühwirth-Schnatter, S. (2006). *Finite Mixture and Markov Switching Models*. New York: Springer.

Kim, B., Min, H., McDonald, J., and Hwang, Y. (2012). "Yen-Synchronization of Floating East Asian Currencies: A Regime-Switching Regression Model and Micro-structural Analysis." *Journal of the Japanese and International Economies* 26:221–232.

Kuester, K., Mittnik, S., and Paolella, M. S. (2006). "Value-at-Risk Prediction: A Comparison of Alternative Strategies." *Journal of Financial Econometrics* 4:53–89.

S&P Dow Jones Indices LLC (2018). S&P 500 [SP500]. Retrieved from FRED database, Federal Reserve Bank of St. Louis. https://fred.stlouisfed.org/series/SP500.

ACKNOWLEDGMENTS

The authors would like to thank Jan Chvosta and Mark Little for their valuable comments and suggestions and Ed Huddleston for his thorough editorial review.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors:

Xilong Chen and Ji Shen
SAS Institute Inc.
Xilong.Chen@sas.com and Ji.Shen@sas.com