Paper ST-13

The Lag-o-Matic: An Improved Method for Selecting the Lag Structure of Multiple Predictor Variables in the Absence of Theory

David C. Sharp, University of Southern Mississippi-Gulf Coast, Long Beach, MS
Stephen M. Finnigan, U.S. Department of State, Washington, DC

ABSTRACT

In applied econometrics, *theory* generally determines the appropriate lag of each independent (i.e., predictor) variable in a time series regression model. But, how does one determine the lag structure of each predictor variable without theory? Various schemes exist, but—without theory--variable-by-variable lag selection can be a daunting task that usually results in a time-intensive, "trial and error" approach. To facilitate the selection of lags, this paper introduces the "Lag-o-Matic," a SAS® program that eliminates many of the difficulties associated with lag selection for multiple predictor variables in the face of uncertainty. The Lag-o-Matic automatically (1) lags the predictor variables over a user-defined range; (2) runs regressions for all possible lag permutations in the predictors (i.e., $VAR1_{t-1}$, $VAR2_{t-1}$; $VAR1_{t-1}$, $VAR2_{t-2}$; $VAR1_{t-2}$, $VAR2_{t-1}$, etc.); and (3) allows users to restrict results according to user-defined selection criteria (e.g., "face validity," significant t-tests, R^2, etc.). Lag-o-Matic output generally contains a short list of models from which the researcher can make quick comparisons and choices.

INTRODUCTION

When specifying a time series regression model, applied econometricians generally rely upon theoretical underpinnings, not only in the selection of the independent (predictor) variables themselves, but for the lag structure of each predictor variable, as well. Selection of the various predictor variables for the multiple regression model is generally straightforward; it typically relies upon solid theory and/or common sense. Similarly, the appropriate lag length of each predictor variable as it relates to the dependent (response) variable is often well established in theory or is fairly obvious from casual observation of the data. At times, however, there are no reliable theories regarding the lag of a relevant predictor variable as it relates to the response variable. Furthermore, cursory examination of the data may provide no immediately obvious insights regarding lag length. Since there is, of course, no compelling reason why lag lengths must be symmetrical (i.e., the same for each predictor variable), the dilemma is compounded as more predictor variables with ambiguous lag lengths are added to the model.

How does the practitioner usually determine the lag structure of each predictor variable within a multiple regression model in the absence of theory? Although techniques vary from the simple to the sophisticated, most rely, in one way or another, upon *crosscorrelations* between a response time series and past values of each predictor, variable by variable. To assist with this task, SAS® users have the benefit of the CROSSCOR option in PROC ARIMA (see, for example, *SAS/ETS User's Guide, Version 6, Second Edition,* p. 137 and *Introduction to Time Series Forecasting Using SAS/ETS Software Course Notes,* p. 277). Unfortunately, however, reliance upon crosscorrelations are frequently unproductive because—after the lag for each predictor variable has been chosen—"face validity" and statistical significance of the coefficients can be compromised when the lagged predictors are finally regressed together. In other words, lag selections that emerge from independent crosscorrelations for each predictor variable frequently do not work when all predictors are regressed in concert. Relationships that appeared to be significant in the crosscorrelations will become insignificant in the regression, and perhaps even bear the wrong sign.

At times, the conflict between the crosscorrelations and the multiple regression is a function of preventable interactions in the predictor variables (e.g., multicollinearity). To support identification of the relationship between the response variable and lagged values of the predictor variables, Box and Tiao (1975) recommend that series be "pre-whitened" (i.e., eliminated of univariate trend, seasonal, or other time based relationships) in order to eliminate spurious correlations before they are crosscorrelated. However, bizarre and unexpected results frequently occur even when all variables have been carefully pre-whitened and the predictor variables contain minimal multicollinearity. Such unacceptable results typically force the practitioner to make additional trips "back to the drawing board" to find just the right combination of predictor lags. Typically, in the end, just the right lag selection for each of the predictor variables is eventually achieved, but only after daunting, time intensive, trial and error endeavors on the part of the practitioner.

To facilitate the selection of lags in a model with many predictor variables, this paper introduces the "Lag-o-Matic," a SAS macro that eliminates many of the problems associated with lag selection in the face of uncertainty. The Lag-o-Matic automatically lags the predictor variables over a user-defined range and runs regressions for all possible lag permutations (i.e., $VAR1_{t-1}$, $VAR2_{t-1}$; $VAR1_{t-1}$, $VAR2_{t-2}$; $VAR1_{t-2}$, $VAR2_{t-1}$, etc.). The Lag-o-Matic allows users to restrict results and select models according to their own selection criteria (e.g., "face validity," significant t-tests, R^2, etc.).

The following sections describe a specific multiple regression-modeling problem, present the Lag-o-Matic, and explain how it was used to handle lag selection and facilitate the model's specification. The paper concludes with brief comments regarding the Lag-o-Matic.

THE LAG-O-MATIC IN PRACTICE: AN EXAMPLE

One of the projects in which the Lag-o-Matic was successfully utilized involved forecasts of employment for the Biloxi-Gulfport-Pascagoula Mississippi MSA. The parties requesting the forecast indicated that it needed to be relatively straightforward methodologically for the intended lay audience. They also requested that the model incorporate the roles of relevant measures of economic activity in the MSA (as opposed to a univariate forecast).

It was theorized that current employment in the MSA (employ) is primarily a function of the following five, local, predictor variables:

- previous employment (employ_)
- previous gambling revenue (gamerev)
- previous taxable sales (taxsale)
- previous ratio values of initial to continued unemployment claims (claimrat); and
- previous values of building permits (permval).

Notably, although there was a strong consensus that previous values of these predictor variables would predict current employment, there was strong dissention regarding how each of the five predictors should be lagged in the regression model; opinions ranged from three to six months per variable.

Monthly data from January 1995 to October 2001 were collected for each of time series. Following Box and Tiao (1975), each series was pre-whitened. Specifically, each dollar-based time series (i.e., gambling revenue, taxable sales, and value of building permits) was adjusted for inflation to 1996 dollars using the GDP deflator. Furthermore, all series were seasonally adjusted using

the Census X11 method through the PROC X11 procedure in SAS/ETS (see, for example, *SAS/ETS User's Guid*e, p 898).

CODE AND RESULTS

The Lag-o-Matic is comprised of five steps. The first step defines the macro and %DO loops with "start" and "stop" parameters indicating where the lag iterations should begin and end for each predictor. The second step creates a dataset with lagged values of the predictors for a given iteration. The third step runs the regression for that iteration, and the forth step limits the output to information of interest. The process continues iteratively until all possible lagged combinations of the predictor variables have been regressed against the response variable. The fifth step appends the results of all of the regressions.

Notably, the number of iterations, *I*, is defined by

$$I = L^X$$

where *L* is the number of lags per predictor variable, and *X* is the number of predictor variables. Since the number of iterations can be quite high, writing to the output and log windows can become extensive. For example, in the following example, lags are limited to run from three to six months (or four potential lags) for each variable. The number of predictors is five. Thus, the Lag-o-Matic will automatically perform (4^5=) 1,024 iterations, and calculate 1,024 alternative regressions. In situations where the number of iterations is very high, such as this, output to the log and output windows will exceed capacity. To avoid this, we have included a statement that writes the log window to a separate file. Likewise, "noprint" options are also used throughout to prevent excess writing to the output window.

The following is the Lag-o-Matic code for the five predictor variable case. Each of the steps is discussed more fully following the code.

```
libname LAGO 'C:\lag';
options nodate pageno=1;

proc printto log="C:\lag\log.log";
run;

/* Step 1: Define macro & assign %DO loops */
%macro iterate
(var1,starti,stopi,var2,startj,stopj,
var3,startk,stopk,var4,startl,stopl,
var5,startm,stopm);
   %do  i=&starti %to &stopi;
    %do j=&startj %to &stopj;
     %do k=&startk %to &stopk;
      %do l=&startl %to &stopl;
       %do m=&startm %to &stopm;

/* Step 2: Create regression dataset */
data regress (keep=date employ &var1. &var2.
&var3. &var4. &var5.);
    set LAGO.MSAdata;
      &var1.=lag&i.(&var1.);
      &var2.=lag&j.(&var2.);
      &var3.=lag&k.(&var3.);
      &var4.=lag&l.(&var4.);
      &var5.=lag&m.(&var5.);

      attrib
         &var1. label="&var1. - &i. Mo. Lag"
         &var2. label="&var2. - &j. Mo. Lag"
         &var3. label="&var3. - &k. Mo. Lag"
         &var4. label="&var4. - &l. Mo. Lag"
         &var5. label="&var5. - &m. Mo. Lag" ;
run;

/* Step 3: Run regression */
proc reg data = regress outest=reg1 tableout
rsquare;
     model employ = &var1 &var2 &var3 &var4
     &var5 /noprint;
run; quit;

/* Step 4: Clean regression output */
data reg2 (drop=_rmse_ _model_ _depvar_
employ _in_ _p_ _edf_);
    length model $175.;
    set reg1;
    model="EMPLOY=f(L&i&VAR1. L&j&VAR2.
    L&k&VAR3. L&l&VAR4. L&m&VAR5.)";
    rename _type_=type;
    if _type_ not in("PARMS","T") then delete;
run;

/* Step 5: Create final output dataset for
all regressions*/
%if &i=&starti and &j=&startj and &k=&startk
and &l=&startl and &m=&startm %then
   %do;
      data results;
          set reg2;
          attrib
             &var1. label="&var1."
             &var2. label="&var2."
             &var3. label="&var3."
             &var4. label="&var4."
             &var5. label="&var5.";
          run;
          /* proc datasets library=work;
             delete reg1 reg2 regress;
          run; */ quit;
        %end;
       %else
 %do;
     proc append out=results new=reg2 force;
     run;
     /* proc datasets library=work;
        delete reg1 reg2 regress;
     run; */ quit;
    %end;

       %end;
      %end;
     %end;
    %end;
   %end;
   %mend;

/* End of Step 1 */
/*iterate(var1,starti,stopi,var2,startj,stopj
,var3,startk,stopk,var4,startl,stopl,
var5,startm,stopm) */
%iterate(employ_,3,6,gamerev,3,6,taxsale,3,6,
claimrat,3,6,permval,3,6);
```

Step 1 defines the macro ("iterate") and creates %DO loops for each of the five predictor variables. Start (i.e., &starti) and stop (i.e., &stopi) parameters define where lags should begin and end for each variable, respectively. At the very end of the Lag-o-Matic code, each predictor is listed, followed by its respective start and stop values. In this particular example, each predictor is allowed to lag three, four, five or six months. In general, however, the start and stop parameters can take on any values (where start ≤ stop). Furthermore, start and stop values need not be the same for each predictor.[1]

Step 2 creates a dataset ("regress") for each of the lag iterations from the raw dataset (i.e., LAGO.MSAdata) that contains the response variable and the unlagged predictors. In this specific case, the first "regress" dataset will contain lagged values of the

[1] A minor "quirk" with The Lag-o-Matic is that it will not run if it sees a predictor variable with the same name as the response variable. In this case, the predictor version of employment is defined as "employ_" to differentiate it from the response variable "employ."

five predictor variables of the order *t-3, t-3, t-3, t-3, t-3,* respectively; the second "regress" dataset will contain lags of the order *t-3, t-3, t-3, t-3, t-4*; the third "regress" dataset will contain lags of the order *t-3, t-3, t-3, t-3, t-5*, and so on. With its current specification, there will be 1,024 such iterations of the "regress" dataset. The last iteration of the "regress" dataset will contain lags of the order *t-6, t-6, t-6, t-6, t-6*. A portion of this last iteration of the "regress" dataset is shown in Figure 1.

Step 3 runs the regression for each "regress" dataset. The OUTEST option of PROC REG creates an output dataset ("reg1") that contains the parameter estimates. In addition, the TABLEOUT and RSQUARE options (see, for example, *SAS/STAT User's Guide, Version 6, Fourth Edition, Volume 2,* p. 1359) write the *t* and the R^2 statistics to this ("reg1") dataset.

Step 4 cleans the regression output by limiting the regression output dataset ("reg1") to include only those statistics of interest. In this particular case, statistics of interest include the model identifier, the coefficient values, the *t* statistics, and R^2, all of which are included in a new dataset ("reg2"). Figure 2 shows the contents of the "reg2" dataset for the final iteration (where, again, the lag structure is *t-6, t-6, t-6, t-6, t-6*). Notably, Step 4 can be easily tailored to contain statistics other than those presented in Figure 2 that may be of interest to the individual user.

Step 5 appends all of the "reg2" datasets (i.e., one from each iteration) into a single dataset (i.e., "results"). The new ("results") dataset contains the results from all of the models. In this case, the "results" dataset contains 2,048 rows, since calculations from the 1,024 regressions contain one row for coefficients and one row for *t* statistics for each regression. A portion of the final dataset ("results") is shown in Figure 3.

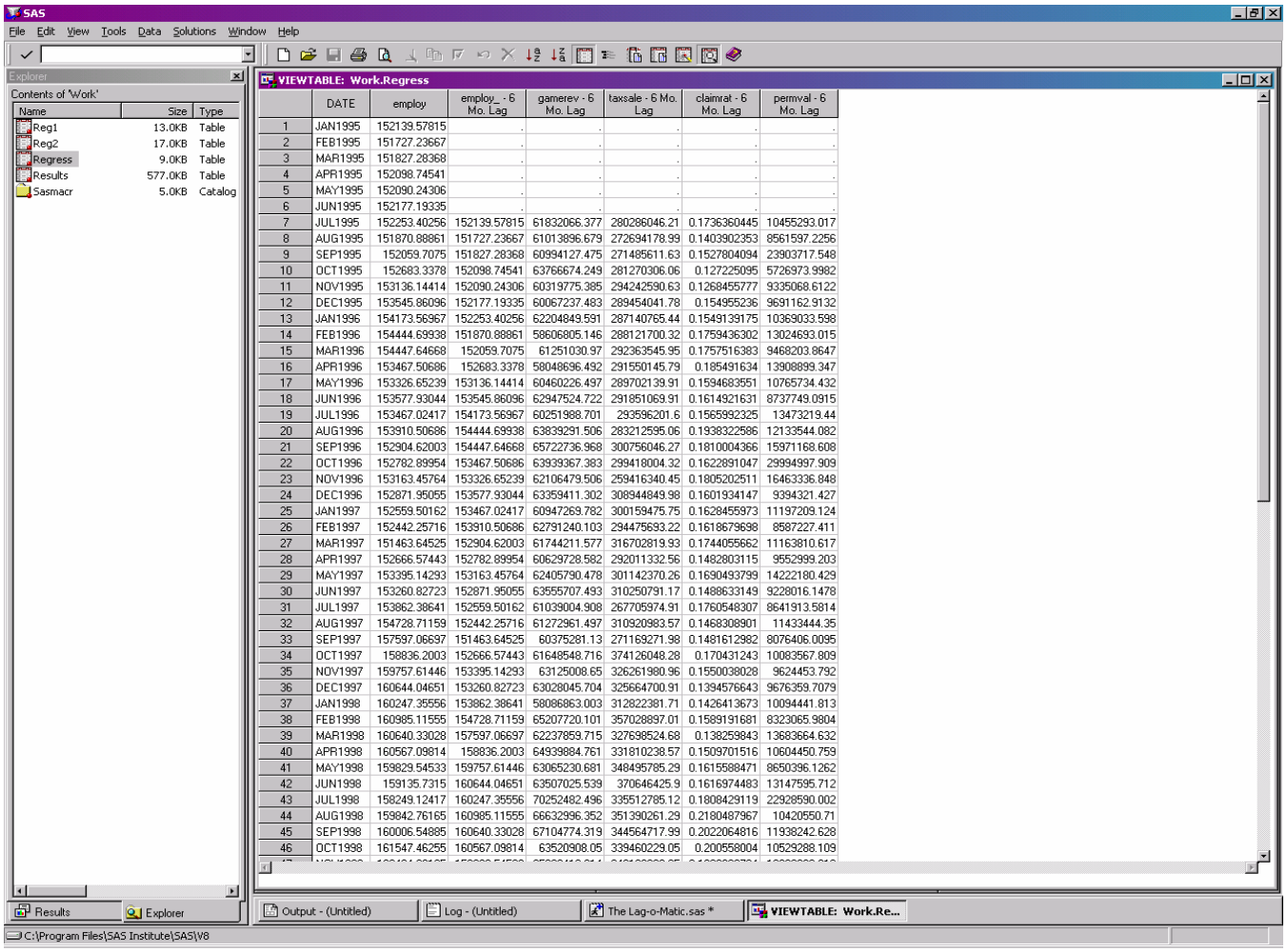

Figure 1. "Regress" Dataset Example

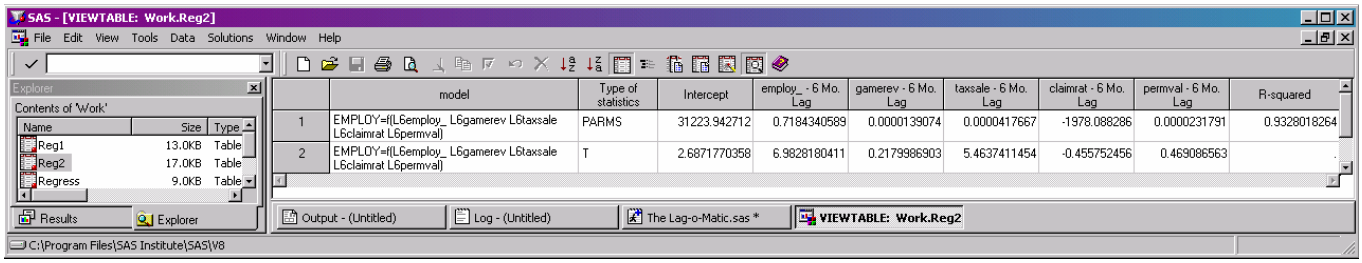

Figure 2. "Reg2" Dataset Example

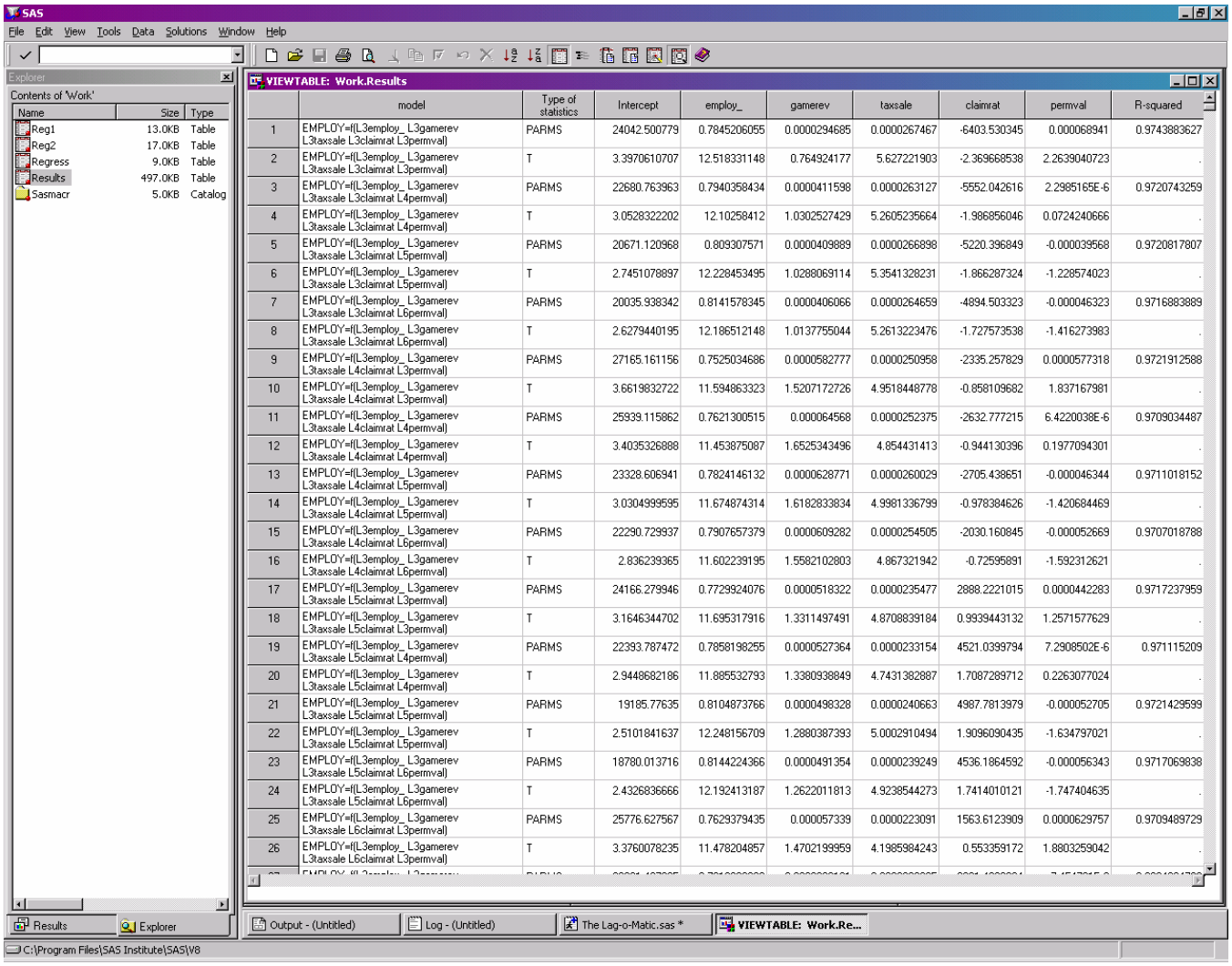

Figure 3. "Results" Dataset Example

The final "results" dataset allows users to restrict results according to their own selection criteria. For example, the user may want to examine only those models with the expected signs (i.e., "face validity") and significant *t*-tests for all coefficients of the predictor variables. In the context of this specific example, positive signs are expected for the coefficients of all variables except "claimrat." Assuming we would be willing to accept significance at α = .10 or lower (corresponding to a critical *t* of approximately 1.671 or higher), we could limit the output to contain only those models with expected signs and significant *t*-statistics on all coefficients with the following code:

```
data good(keep=model);
   set results;
   criterion=1.671;
   if type="T" and employ_ ge criterion
               and gamerev ge criterion
               and taxsale ge criterion
               and ABS(claimrat) ge criterion
               and permval ge criterion
   then output;
run;
```

In this specific case, the output dataset from the code above (i.e., "good") reveals that only one model out of 1,024 meets the selection criteria. As shown in Figure 4, the output of the dataset "good" contains a single model, and that model has a lag structure of

$employ_t = f(employ_{t-4}, gamrev_{t=3}, taxsale_{t-3}, claimrat_{t-3}, permval_{t-3})$.

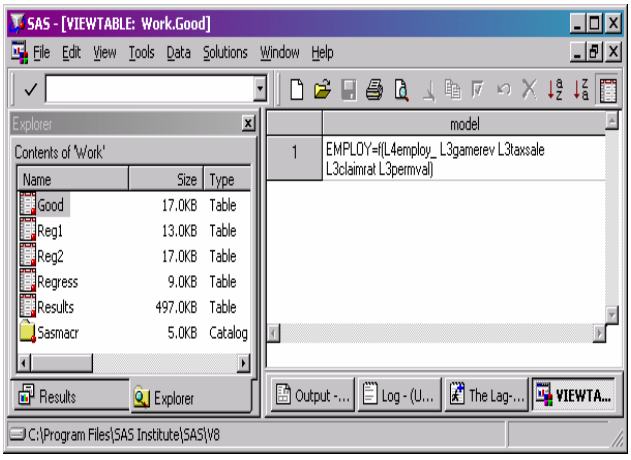

Figure 4. "Good" Dataset Example

For this particular project, only one regression out of the 1,024 alternative regressions run met the relevant selection criteria. A situation such as this makes model selection easy, of course. However, if multiple models had passed the initial screening, we could have further limited the relevant "good" models based upon other statistics of interest. If we wanted to examine the models based upon, say, R^2, we could do so with the following code:

```
proc sql;
    create table goodcoef as
      select * from results
    where model in (select * from good) and
    type = "PARMS";
    create table goodts as
      select * from results
    where model in (select * from good) and
    type = "T";
 quit;

proc sort data=goodcoef;
   by descending _RSQ_;
run; quit;
```

The code above places the coefficients and the *t*-statistics in separate databases ("goodcoef" and "goodts," respectively) and sorts the equations in the coefficient dataset ("goodcoef") by R^2, from high to low.

CONCLUSION

Although theoretical underpinnings are preferred by most applied econometricians in the selection of lag lengths for predictor variables, theory is frequently unavailable. In the absence of theory, most users rely upon crosscorrelations between a response time series and past values of each predictor variable to determine the lag length, one variable at a time. Unfortunately, variable-by-variable lag selection without the benefit of theory can be a daunting task. Moreover, when this task is completed and all of the various lagged predictors are finally placed together on the right-hand-side of the model and regressed together, "face validity" and statistical significance of the coefficients can be compromised, even when multicollinearity is minimal and the time series have been pre-whitened. Often, these results lead the researcher "back to the drawing board" in an attempt to find just the right combination of lags for the predictor variables in the model.

In an effort to provide users with an improved method of lag selection in the face of uncertainty, this paper introduced the Lag-o-Matic. The Lag-o-Matic is a time-saving macro we have written which automatically (1) lags the predictor variables over a user-defined range; (2) runs regressions for all possible lag permutations among the predictor variables (i.e., $VAR1_{t-1}$, $VAR2_{t-1}$; $VAR1_{t-1}$, $VAR2_{t-2}$; $VAR1_{t-2}$, $VAR2_{t-1}$, etc.); and (3) allows users to restrict results according to user-defined selection criteria (e.g., "face validity," significant t-tests, R^2, etc.). Output from the Lag-o-Matic generally contains a short list of models from which the researcher can make quick comparisons and choices.

This paper demonstrated the Lag-o-Matic in the context of a model with five predictor variables. However, the Lag-o-Matic can be easily edited to specify equations with any number of predictor variables. Also in the specific example of this paper, "face validity" and the significance of *t*-statistics were presented as the relevant criteria for model selection. Importantly, however, alternative criteria could be used. Users who are most interested in explained variability could simply select models based upon R^2. In addition, for forecasters, it is a relatively straightforward process to reconfigure the Lag-o-Matic to calculate *MAE*, *MAPE*, *MSE*, or *RMSE* over a holdout sample to assist in model selection.

Of course, the Lag-o-Matic has some limitations. It will not, for example, determine the number of predictor variables that should be included in the model. Likewise, it will neither determine best functional form (e.g., logarithms, etc.) of the predictors as they relate to the response variable nor make adjustments for autocorrelation. Results are, of course, a function of the quality of the predictors selected. It is therefore important to minimize multicollinearity and seasonality prior to use of the Lag-o-Matic.

Despite its limitations, the benefits of the Lag-o-Matic are self-evident. When examining relationships between a response variable and lagged values of any given number of predictor variables—and when only the proper lag lengths are unknown—the Lag-o-Matic can be a useful, timesaving and user-friendly tool for determining just the right combination of predictor lags.

REFERENCES

Box, G.E.P., and Tiao, G.C. (1975), "Intervention Analysis with Applications to Economic and Environmental Problems," *Journal of the American Statistical Association*, 70, 70-79.

SAS Institute Inc. (2000), *Introduction to Time Series Forecasting Using SAS/ETS® Software Course Notes,* Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1997), *SAS® Macro Language: Reference, First Edition*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1993), *SAS/ETS® User's Guide, Version 6, Second Edition,* Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1990), *SAS/STAT® User's Guide, Version 6, Fourth Edition, Volume 2,* Cary, NC: SAS Institute Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

David C. Sharp, Ph.D.
Department of Economics
University of Southern Mississippi-Gulf Coast
Long Beach, MS 39560
Phone: (228) 867-2621
Fax: (228) 865-4588
Email: david.sharp@usm.edu