

Multivariate Procedures in SAS

1) Introduction

- the goal of the data methods that we have been discussing is to prepare a data set (analysis file) that can be used in an empirical investigation
- we have discussed some simple statistics and cross-tabulations that can be performed in SAS
 - these procedures indicate general associations in the data
 - the procedures do not account for indirect and possibly confounding influences from other observed variables
- estimates of “direct” or “partial” associations that account for these indirect influences come from *multivariate* procedures, such as regression analyses, probit procedures, etc.
- SAS has many multivariate routines
 - some of these will be immediately recognizable
 - many others are available but tougher to find
 - ◊ SAS is written for a broad set of users
 - ◊ the descriptions that other scientists use (and that SAS uses) do not always line up with the descriptions that economists use
- should you actually use these routines?
 - it can be convenient to use the routines available in SAS
 - however, economists often find the routines in packages like Stata to be more useful
 - can’t make a good choice without knowing at least some of what’s available in SAS

2) PROC REG¹

- the SAS regression procedure, PROC REG, is the workhorse of multivariate procedures
- it estimates OLS regressions
- basic syntax and operation

```
PROC REG <regression_options>;  
    <model_label:> MODEL <dep_var> = <list_of_explanatory_variables>  
    </ model_options>;
```

- where the *dep_var* is a SAS variable with the dependent, or outcome, variable in the regression
- the *list_of_explanatory_variables* is a list, separated by spaces, of independent variables in the regression
- *model_label* is a label that will appear in all of the output associated with this model; it is useful for producing readable output
- unless asked to do otherwise, the REG procedure automatically includes an intercept term (to drop the intercept, use the NOINT model option)

¹ See http://support.sas.com/documentation/cdl/en/statug/59654/HTML/default/reg_toc.htm.

- the REG procedure outputs estimated coefficients, coefficient standard errors, as well as test statistics and p -values (under the null hypothesis that the true coefficient is zero) for the intercept and independent variables
- the REG procedure also calculates the residual sum of squares, the R^2 and adjusted R^2 statistics, the mean square error, F -test statistics and other statistics
- estimating many models
 - PROC REG can be modified to estimate many models in the same procedure (i.e., without calling another regression procedure)
 - multiple MODEL statements can be included in the same procedure
 - also a list of dependent variables can be provided instead of a single dependent variable; all variables that appear before the equals sign (=) in the MODEL statement are treated as dependent variables
- testing and correcting for heteroskedasticity
 - including the SPEC model option leads to a heteroskedasticity specification test (artificial regression test) being performed
 - including the ACOV model option leads to a heteroskedasticity-consistent variance covariance matrix being calculated
 - note that newer versions of SAS (starting with version 9.2) will have more convenient and flexible options
- testing for first-order autocorrelation
 - the DWPROB model option will lead to a first-order Durbin-Watson test with a corresponding p -value for the null hypothesis of no autocorrelation
 - note that the DWPROB option assumes that the data are sorted in chronological order
- tests of model restrictions
 - restrictions on the coefficients from the last specified model can be tested using the TEST statement
 - syntax

```

<test_label:> TEST <test_expression_1> <, test_expression_2>
                    <... , test_expression_k>;

```

- the test expressions are mathematical expressions involving the explanatory variables (coefficients actually) from the regression model
 - ◊ if equals signs are included in an expression, TEST tests the actual specification
 - ◊ if equals signs are omitted, TEST tests the hypothesis that all of the expressions are jointly equal to zero
- multiple TEST statements can be issued in the same REGression procedure
- examples

```

PROC REG;
  MODEL y = x1 x2 x3;
  TEST x2 = 0, x3 = 0;

```

- ◊ estimates a regression with y as the dependent variable and $x1$, $x2$, and $x3$ as the independent variables

- ◇ it then tests the null hypothesis that $x_2 = x_3 = 0$ (i.e., that x_2 and x_3 are jointly equal to zero); note that TEST x_2, x_3 ; would have tested the same hypothesis

TEST $x_2 - x_3$; and TEST $x_2 = x_3$;

- ◇ test the null hypothesis that $x_2 = x_3$

TEST $x_2 - 1$; and TEST $x_2 = 1$;

- ◇ test the null hypothesis that $x_2 = 1$

- outputting predictions and residuals
 - a data set with predictions and residuals from the regression can be produced using the OUTPUT statement
 - the OUTPUT will produce these statistics for the last MODEL estimated
 - multiple OUTPUT statements can be included
 - the OUTPUT statement must include at least one statistics *keyword*
 - ◇ P = <prediction_variable> generates and stores predictions
 - ◇ R = <residual_variable> generates and stores estimated residuals
 - ◇ see the documentation for additional supported statistics
 - the OUTPUT data set would contain one observation for every observation read into the REGression procedure
 - it also will contain copies of all of the dependent and independent variables used in the procedure
- outputting coefficient estimates
 - including the option OUTEST=<SAS_data_set> in the PROC REG statement causes the procedure to create a SAS data set containing the coefficients for all of the estimated models
 - the data set contains one observation per model estimated
 - models are identified by a _MODEL_ variable and by the names of the dependent variables
 - coefficients are stored in variables with the same names as the explanatory variables

3) PROC LOGISTIC

- the primary procedure for running binary choice models is PROC LOGISTIC
 - as the name suggests, PROC LOGISTIC estimates logit models; however, it also estimates probit and other types of models
 - SAS has a PROC PROBIT that can be configured to estimate binary choice probit models (the standard specification estimates another type of model)
- syntax

```
PROC LOGISTIC <options>;
  MODEL <response_specification> <(resp_variable_options)>
    = <list_of_independent_variables> </ model_options>;
```

- the response specification can be of two types

- ◇ a binary variable (or ordered categorical variable)
- ◇ $\langle \text{positive_outcomes} \rangle / \langle \text{total_outcomes} \rangle$ where the first term is a SAS variable with the number of positive outcomes in a grouped observation and the second term is the total number of outcomes; this is used to estimate grouped binary data
- note that you can only specify one MODEL per LOGISTIC procedure; to estimate multiple models, you need to call the procedure again
- a quirk in PROC LOGISTIC is that its default is to treat the lowest value in a response variable as the outcome of interest
 - ◇ thus, the default for a 0/1 binary variable is to model the probability that the outcome is 0 – that is, the exact opposite of how most social scientists specify these models!
 - ◇ there are several ways to “fix” this; one is to include a response variable option (EVENT='1'); another is to include a response variable option (DESCENDING)
 - ◇ example, let y be a 0/1 binary variable, and let x_1 , x_2 , and x_3 be independent variables; we could specify a model

```
PROC LOGISTIC;
  MODEL y(DESCENDING) = x1 x2 x3;
```

- using the model options, you can also specify the type of distribution that you want to use for the model
 - ◇ the default is the logistic distribution, producing a logit model
 - ◇ to get probit estimates, you would use the LINK=PROBIT model option
 - ◇ from the example above,

```
PROC LOGISTIC;
  MODEL y(DESCENDING) = x1 x2 x3 / LINK=PROBIT;
```

- ◇ to estimate a conditional log-log binary model (useful in event-history analyses), you would use the LINK=CLOGLOG model option
- if you provide an ordered categorical model, the LOGISTIC procedure will estimate an ordered choice model
 - ◇ again, the choices will be modeled focusing on the lowest value
 - ◇ to estimate a model focusing on high values, use the (DESCENDING) response variable option
- outputting predictions
 - a data set with predictions can be produced using the OUTPUT statement
 - two common statistics are
 - ◇ $P = \langle \text{prediction_variable} \rangle$ generates and stores predicted probabilities
 - ◇ $XBETA = \langle \text{latent_pred_variable} \rangle$ generates and stores predicted latent variable
 - ◇ see the documentation for additional supported statistics
 - the OUTPUT data set would contain
 - ◇ one observation for every observation read into the LOGISTIC procedure for binary data

- ◇ one observation for each possible response level and observation for ordered categorical data
- it also will contain copies of all of the dependent and independent variables used in the procedure
- outputting coefficient estimates
 - including the option OUTEST=<SAS_data_set> in the PROC LOGISTIC statement causes the procedure to create a SAS data set containing the coefficients for the estimated model
 - the data set contains one observation
 - coefficients are stored in variables with the same names as the explanatory variables

4) PROC QLIM²

- the QLIM procedure is a new addition to SAS and estimates a variety of qualitative and limited dependent variable models, including logit, probit, tobit, and selectivity-corrected models
- binary-choice specifications
 - syntax for a logit model is

```
PROC QLIM <options>;
  MODEL <dep_var> = <list_of_explanatory_variables>
    / DISCRETE(D=LOGIT) <other_model_options>;
```

- note that unlike the LOGISTIC procedure, QLIM models the probability that the dependent variable equals 1
- probit specification would be estimated using just the DISCRETE option (it is the default binary choice model in QLIM)
- binary choice specifications with controls for heteroskedasticity can also be estimated using the HETERO statement

```
HETERO <dep_var> ~ <list_of_het_explanatory_variables>
  </ h_options>;
```

- ◇ the h_options include the type of LINK, LINEAR or EXPonential
- ◇ they also include whether a SQUARE of the linear combination of the explanatory variables will be used
- ◇ they also include whether the constant should be dropped (NOCONST)
- tobit specification
 - syntax for a “standard” tobit (censored regression model, censored from below at zero) is

```
PROC QLIM <options>;
  MODEL <dep_var> = <list_of_explanatory_variables>;
  ENDOGENOUS <dep_var> ~ CENSORED(LB=0);
```

² See http://support.sas.com/documentation/cdl/en/etsug/60372/HTML/default/qlim_toc.htm.

- with this specification QLIM will estimate a standard ML tobit
- more generally, the lower bound (LB=) in the CENSORED option can be specified as
 - ◇ a numerical constant
 - ◇ a variable
- upper bounds can also be specified by including an UB= in the CENSORED option
- for example, let y be CENSORED from below by 0 and from above by 100000; let $x1$, $x2$, and $x3$ be independent variables; we could specify a model

```

PROC QLIM;
  MODEL y = x1 x2 x3;
  ENDOGENOUS y ~ CENSORED(LB=0 UB=100000);

```

- in addition to these models, QLIM also supports estimation of selection models, bivariate probit models, Box-Cox models, and other models
- it is possible to output predictions and coefficient for this procedure