Some Brief SAS Notes for Epidemiologists

Charles DiMaggio, PhD
College of Physicians and Surgeons
Departments of Anesthesiology and Epidemiology
Columbia University
New York, NY
cjd11@columbia.edu

April 29, 2011

Contents

1 Intro

I have found it useful to have the following bits and pieces of SAS code in one place. I've tried to comment out descriptive text, so you can just cut and paste the syntax into your SAS editor and substitute your own variable names. I make no attempt here to explan the statistics themselves. The material itself comes from standard SAS manuals and some online sources. [1]

2 utility procdedures

2.1 help

```
Find help on SAS procedures:
Help > SAS Help and Documentation > SAS Products > Base SAS > SAS Procedures >
```

2.2 tabulations

```
PROC TABULATE to create a table with sums, medians, etc
proc tabulate data=your.data;
          var  cont_var; /* continuous variable you want stats on*/
          class cat_var; /*categorical variable you want stats by*/
          format cat_var cat.; /*formatting statement using a previously
                     created format for your categorical variable*/
          table cat_var, sum* cont_var pctsum* cont_var median* cont_var;
                           /* requests the sum, percent of sum, median
                    continuous variable for each value of the categorical
                    variable;*/
run;
```

2.3 conversions

```
INPUT function to convert a SAS character value to a numeric value
data  new;  /* creating a new data set with the numeric value */
         set old;   /* your existing data set with the character value */
         new_numeric_var=input(old_char_var, informat);  /* note need specify
                        a numeric informat that best describes how to read
                        the character into the numeric value */
run;
```

[1] By the way, almost all the SAS I know I learned from taking the very useful short courses offered by the SAS Institute. They are even more convenient now, as many of them are offered online.

2.4 output

```
ODS HTML to output a table directly into Excel
ods html file="C:\Folder\file.xls"
                style = minimal;  /*  open ods, specify drive, folder and file,
                   note the .xls extension, minimal style removes formats */
                proc   ;         /* procedure call */
run;
ods html close;
```

3 graphing and simple statistical procedures

3.1 simple plots

PROC GPLOT to graph one variable against another:

```
goptions reset=all;  /* remove any previous settings */
goptions cback=white device=win; /* set background color, specify windows */

axis1  major = (h=2.0  c=black) /* define your vertical axis */
           minor = (h=1.0 c=black)
           order = 0 to 3000 by 100
           label = (h=1.5 a=90 f=arial c=black 'Label for the Axis');

axis2  major = (h=2.0  c=black) /* define your horizontal axis */
           minor = (h=1.0 c=black)
           order = 1 to 132 by 6
           label = (h=1.5 f=arial c=black Label);

proc gplot data= your.data;
plot var1 * var2                                /* specify variables to be plotted */
                             / vaxis = axis1 /* invoke previously defined variables */
                               haxis = axis2;
     symbol v='.' f=arial h=2 i=sm30 c=black; /*define symbol to use for plot
                                              note i= statement defines 30% smoothing
                                              via interpolation*/
      title h=3.0 c=black f=arial j=center Main Title of Plot ;
      title2 h=2.0 c=black f=arial j=center Subtitle of Plot;
run;
quit;
```

3.2 box plots and probability plots

PROC BOXPLOT is a convenient graphical way to get a 5-number summary and sense of normality or symmetry across groups of continuous data:

```
proc boxplot data = your.data;
       plot continous_var * category_var  / boxstyle schematic
cboxes=black;
/* specify type and color of boxplot, note,
if only one category create dummy variable against which to plot */
run;
```

PROBPLOT as an option in PROC UNIVARIATE to graphically assess assumptions of normality for continuous data:

```
proc univariate data = your.data;
       var continuous_var; */ specify your variable */
        id idnumber; /* to identify outliers or illogical observations */
        probplot continuous_var / normal (mu=est  sigma= est color=blue w=1);
             /* invoke the probability plot option, overlay a normal against
                            which to judge, set parameters */
        title;
run;
```

3.3 frequencies and odds ratios

PROC FREQ to calculate an odds ratio and its 95% confidence interval for a 2x2 table:

```
proc freq data =  your.data;
        tables var_a *  var_b / measures cl;  /* variables are both
dichotomous, ' measures' returns the odds ratio, 'cl' returns the
                                        confidence limits*/
run;
```

3.4 standard errors and confidence intervals

STERR and CLM in PROC MEANS to get the standard error of the sample mean and its confidence limits:

```
proc means data=your.data n mean stderr clm; /* invoke procedure, specify stats */
       var continuous_var;
       title Your Title;
run;
```

4 regression

4.1 correlation

PROC CORR to get a correlation coefficient, a summary measure of linear relationship. [2]

```
Proc corr data= your.data rank;  /* rank orders correlations  high to low */
        Var predictor1 predictor2 predictor3 predictor4
        With outcome_var;  /* omit  with  to get a correlation matrix */
         Title correlation of outcome with predictors;
Run;
```

4.2 linear regression

PROC REG for simple linear regression:

```
Proc reg data= your.data;
          Model outcome_var = pred_var;
          Title simple linear regression;
Run;
Quit;
```

4.3 logistic regression

PROC LOGISTIC for categorical outcomes:

```
Proc logistic data = your.data;
     Class cat_var  (param=ref  ref=high); /* class statement to define  variable
                       as a categorical predictor, use reference
                        cell coding with high as the referent */
      Model outcome_var (event=1) = cat_var / clodds=pl /* pl= profile likelihood
                          which will give you better CIs for small sample
                           sizes */
     title1 logistic regression  ;
run;
```

PROC LOGISTIC for ordinal outcomes e.g. Likert Scales seen in many surveys [3]

[2] Caution, this is *not* a measure of a causal relationship. It's thrown off by extreme outliers, mulitcollinearity, etc Always graph the variables first (See PROC GPLOT)

[3] *NB: if complex sampling strategies or correlated responses consider using SUDAAN, PROC SURVEY procedures, or a general estimating equations.

```
proc logistic data=your.data desc; /* SAS will invoke ordinal analyses if outcome
                      variable contains more than 2 responses.  'desc'
                      option defines cumulative probability of being in one of
                      the ordered categories or lower */
   model ordinal_outcome=var;
   title 'Ordinal Logistic Regression Model;
run;
```

4.4 Poisson regression

GENMOD to conduct Poisson Regression for count data. [4]

```
proc genmod data=your.data;
         class catvar1 (param=ref ref=first)  catvar2 (param=ref ref=first); /* define
                              categorical variables in the model as reference
                               type with first level as the referent */
         model depvar = catcar1 catvar2 / dist=poi link=log type3; /* specify model and
                              distribution as poisson with a log link; type=3
                              requests likelihood statistics */
         title ;
run;
```

GENMOD to conduct Poisson Regression for rate data by specifying a population as an offset variable

```
proc genmod data=your.data;
         class catvar1 (param=ref ref=first)  catvar2 (param=ref ref=first);
         model depvar = catcar1 catvar2 /  offset=log_pop dist=poi link=log type3;
                        /* note must be the log of the rate denominator
                        e.g. population; results in incidence density
                        ratio*/
         title ;
run;
```

[4] Non-negative, highly skewed and rare outcomes. Consider negative binomial model (dist=nb) if overdispersed ($\mu > var$)

4.5 factor analysis

VARCLUS to cut down on redundant predictor variables by grouping based on correlations and choosing one from each highly correlated group

```
proc varclus data=your.data maxeigen=.70 short; /* maxeigen sets threshold for
                eigenvalue below which you will no
                 longer cluster variables; cuts down
                           on the voluminous output */
   var var_1 var_2 var_3 var_4 var_5 var_6 var_7 var_8 var_9 var_10 var_11 var_12
              var_13 var_14 var_15 var_16 var_17 var_18 var_19 var_20 var_21 ;
   title 'Variable Clustering';
run;
```