SAS examples

Every SAS program starts with these paragraphs of code. This is to download data from an excel 97-2003 file with no passwords and no crazy formatting

```
libname disk "C:\Users\Acer\Dropbox\SAS\adulths";
libname library "C:\Users\Acer\Dropbox\SAS\adulths";
options LS=140 pageno=1;

PROC IMPORT
DATAFILE="C:\Users\Acer\Dropbox\SAS\adulths\adulths2.XLS"
DBMS=EXCEL2000
OUT= WORK.MASTER replace;
GETNAMES=YES;

PROC CONTENTS POSITION DATA=master; RUN;
```

Data Statements. Put the name of your dataset right after data and then after set put master all the time

```
data putnameofdatasethere; set master;

if age > 168;
if surgdate >mdy(10,22,2008);
if ageflow < 8 then qmax=((5.7244*log(flowvol+1))-13.6033);
if mdy(10,22,2008) le surgdate le mdy(4,26,2011) then era=0;else era=1;
If age le 192 and tanner <4 then delete;
If age le 192 and tanner=. then delete;
If fumonths < 1 then fu=0;
if fumonths=. then fu=0;
if fumonths ge 1 then fu=1;
```

Other Data useful examples

```
if age gt 45 then age45=1;else age45=0;if age=. then
age45=.;*the last statment excludes missing values";

if age gt 45 then age45=1;else if 0 le age le 45 then
age45=0; *"this line does the same thing as the line
above";

age45=(age gt 45);if age=. then age45=.;*"this line
does the same thing as the line above";*(whatever is in
parenthesis if true equals 1 if false equals 0);

lweight=log(weight);*log tranformation to make a
variable normal;
```

PROC PRINT AND PROC SORT

```
proc print data=losttofu; var name MR;run;

proc sort data=szabo out=szabo3;
by location ; run;

proc sort data=szabo out=szabo2;
by surgtype ageflow ; run;
proc print data=szabo2; var  ageflow surgtype maxflow
flowvol qmax ;run;


proc print data=szabo3; var  location maxflow ;run;
```

PROC FREQ

Can do it for one or more variables. If using 3 variables, the first variable is the one you are controlling for. If using 2 variable the risk factor

(type of suture, meatal location, redo etc) is the first variable

```
proc freq data=adults; tables test*surgtype*redo; run;
*Three way crosstab, the first variable of the list is the stratafication variable, the second is the row, the third one is the column.
proc freq data=szabo; tables suture*UC/ chisq; run;
proc freq data=LAPHERNIA; tables suture*recur/chisq fisher;
run;
```

PROC MEANS

```
proc means n nmiss median p25 p75 data=adultsfu;
var fumonths edad age dateuroflow maxflow flowvol; run;

proc means n nmiss mean min max std data=androgen; var glanspreop;
class test; types (test); run;

proc means n nmiss mean std data=androgen;
var age glanspreop glansop;
class test era; types ()(test)test* era; run;

proc means n nmiss mean std data=androgen;
var glanspreop glansop;
class test era dose; types () test*era*dose ; run;
```

PROC EXPORT

```
proc export data=adultsfu
outfile="c:\Users\Acer\Dropbox\SAS\adulths\adultlist.xls" dbms=excel replace; sheet='sheet-name'; run; quit;
```

PROC LOGISTIC AND PROPENSITY SCORE ANALYSIS

```
proc logistic descending  data=logistic; model uc= redo
edad prox ms  /  details;run;
proc logistic descending data=szabo; model weak2=
meatus redo buccal age uc/selection=stepwise details;
run;

***propensity score analysis;

We want to determine if testosterone (Yes or Not given)
decreases hypospadias complications or not. The
propensity score is obtained by using the variables
size of glans preop and location of the meatus:

proc logistic descending data=hypospadias;
model testosterone= GlansSizePreop meatus ;
output out=propensityscore pred=PS; run;

The last command created a new dataset:
propensityscore. Then we do a few logistic regressions
trying different variables modeling urethroplasty
complications (UC) and include the propensity score
(PS):

proc logistic descending data=propensityscore; model
uc= testosterone PS meatus
redosurgery/selection=stepwise  details; run;

proc logistic descending data= propensityscore; model
uc= test ps meatus redo glansop/selection=stepwise
details; run;

proc logistic descending data= propensityscore; model
uc= test ps meatus redo glanspreop/selection=stepwise
details; run;
```

```
proc logistic data=test descending; model hcv= hxtransf
ivda tatoo hcworker beer   heroine cocaine amphets /
selection=stepwise details sle=.10 sls=.10 include=1;
run; * includes 1 means it keeps the first variable
entered
*significance level to enter the model sle, and
significance level to stay in the model sls
so that you can increase significance to 10 percent
instead of 5 percent
*the above checks each varible to see which one is the
strongest of the model*;
*on the output c at the bottom is like r for regression
models only that c=0.50 is no association, positive
association is between 0.50 to 1;
```

PROC UNIVARIATE

```
proc univariate normal data=logistic; var edad ledad;
run; ***gives you statistics about a variable***;

proc univariate data=correlation;
var change; run;

proc univariate normal data=correlation; var gclinic
glansop; run;

proc univariate normal data=test; var lweight;
run;*checks to see if the variable is normal for sure,
skewness of 0 is perfect, kurtosis of 0 is perfect
normal
skewness is the measure of left right skey, kurtosis is
the peak, shapiro-wilk is the test of normality most
commonly used, if significant then
is not normal...
```

PROC CHART

```
proc chart data=logistic; hbar edad ledad;run;

proc chart data=test;hbar lweight;run; *checks to see
if the data is normal, checks the distribution of the
data gives off a c graph*;
```

PROC NPAR1WAY (NON PARAMETRIC TEST)

```
proc npar1way data=escalating wilcoxon median;
      class dose;
      var glanspreop;
         run;
```

PROC CORR (pearson, spearman)

```
proc corr data=correlation pearson spearman
          plots=matrix(histogram);
   var gclinic glansop;
run;

proc corr spearman pearson kendall data=test; var
weight; with height; run;***kendall correlation handles
better ordinal variables**;
```

PROC PLOT

```
proc plot data=test; plot weight*height; by sex; run;
***graph of linear regression with every point with a
name in it**;
```

PROC LIFETEST (Kaplan meyer)

```
proc lifetest data=curves plots=(s);
time time*censored(0);
strata surgery;
run;
```

```
proc lifetest data=curves plots=(s);
time kaplandate*failure(0);
strata surgery;
run;
```

PROC PHREG (COX REGRESSION MODELING)

```
proc phreg data=curves;
model kaplandate*failure(0)= surgery age sex ; run;
```