/\*----------------------------------------------------

\* \*

\* \*

\* University of Iowa : SAS User's Group \*

\* Useful Tips Code \*

\* Last Modified : 12Aug2013 \*

\* Programmer : JJB \*

\* \*

-----------------------------------------------------

\* \*

\* List of examples: \*

\* \*

\* 1) Using cards to make dataset \*

\* 2) Sorting by using natural "language" rules \*

\* 3) Renaming variables \*

\* 4) Counting values that do not exist \*

\* 5) Combining summary data with individual values \*

\* 6) Attributes of your data are unknown \*

\* 7) Coding patterns of vars. into one variable \*

\* 8) Keeping only the desired output \*

\* 9) Adding footnotes and usernames to output \*

\* \*

---------------------------------------------------\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : including data in a program using cards

data measures;

input f\_name $10. sex $ age height weight ;

cards;

Alfred M 14 69.0 112.5

Alice F 13 56.5 84.0

Barbara F 13 65.3 98.0

Ronald M 15 67.0 133.0

Thomas M 11 57.5 85.0

William M 15 66.5 112.0

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : sorting cases by using "language" rules

data bf\_sort;

input parts $12.;

cards;

10.2.f.ii

10.2.a.iii

10.2.f.i

10.2

8.4.5.b.i

10.2.c.iii

10.2.d.i

10.2.d.ii

8.4.5.c.i

10.2.e.ii

10.2.e.i

7.5.2.b

7.5.2.c

8.4.5.e.i

10.2.b.iii

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : renaming variables

data toons;

input ID $4. yakko wakko dot buttons mindy;

cards;

0001 0 1 1 2 3

0002 2 2 1 2 0

0003 3 4 4 3 2

0004 1 0 0 2 1

0005 4 3 4 4 4

;

run;

\*\*\*\*\* keep and order the variables as desired ;

data toons\_reorder;

retain id yakko dot mindy;

set toons;

keep id yakko dot mindy ;

run;

\*\*\*\*\* rename the variables using an array ;

data toons\_renamed (keep=ID toon1 -- toon3);

 set toons\_reorder;

 array old{3} yakko -- mindy;

 array toon{3} ;

 do i=1 to 3;

 toon[i] = old[i];

 end;

run;

\*\*\*\*\* using the colon ":" operator...

data tmp ; set toons\_renamed;

toon\_sum = sum(of toon: ) ;

run;

proc means data = tmp;

var toon: ;

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : counting values that do not exist

proc format;

 value race 1 = 'American Indian/Alaska Native'

 2 = 'Asian'

 3 = 'Native Hawaiian or Other Pacific Islander'

 4 = 'Black or African American'

 5 = 'White'

 6 = 'More Than One Race'

 7 = 'Unknown or Not Reported';

 value eth 1 = 'Hispanic or Latino'

 2 = 'Not Hispanic or Latino'

 3 = 'Unknown or Not Reported';

 value sex 1 = 'Male'

 2 = 'Female'

 3 = 'Unknown or Not Reported';

run;

data race;

input race eth sex ;

cards;

1 2 1

1 2 1

1 2 2

2 2 1

2 2 2

2 2 1

3 2 2

3 2 2

3 2 1

4 2 2

4 2 1

4 2 2

run;

\*\*\*\*\* Does not work correctly ;

proc tabulate data = race;

class eth sex ;

format eth eth. sex sex. ;

table (eth all = 'Ethnicity Category: Total of All Subjects'),

sex all = 'Total'

 / printmiss misstext='0';

 title 'No Format - Hispanic is Missing';

run;

\*\*\*\*\* Does work correctly ;

proc tabulate data = race;

format eth eth. sex sex. ;

class eth sex / preloadfmt;

table (eth all = 'Ethnicity Category: Total of All Subjects'),

sex all = 'Total'

 / printmiss misstext='0';

 title 'Formatted - Hispanic is Present';

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : combining summary data with individual records

data prop\_scores;

 do i = 1 to 30;

 pred\_prob = ranuni(345343);

 output;

 end;

 keep pred\_prob;

run;

/\* find quintiles of scores \*/

proc univariate data=prop\_scores;

 var pred\_prob;

 output out=Pctls pctlpts = 20 40 60 80

 pctlpre = prop\_

 pctlname = pct20 pct40 pct60 pct80;

run;

/\* add quintiles to data set with scores \*/

data prop\_scores2;

 if \_n\_ = 1 then set pctls;

 set prop\_scores;

run;

/\* create variable prop\_quint that \*/

/\* categorizes propensity scores into quintiles. \*/

data prop\_scores2;

 set prop\_scores2;

 if missing(pred\_prob) then prop\_quint = .;

 if . < pred\_prob <= prop\_pct20 then prop\_quint = 1;

 if prop\_pct20 < pred\_prob <= prop\_pct40 then prop\_quint = 2;

 if prop\_pct40 < pred\_prob <= prop\_pct60 then prop\_quint = 3;

 if prop\_pct60 < pred\_prob <= prop\_pct80 then prop\_quint = 4;

 if prop\_pct80 < pred\_prob then prop\_quint = 5;

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : knowing your data

proc sql;

 create table AllVars as

 select \*

 from dictionary.columns

 where libname = 'SASHELP' and memname = 'CLASS' ;

 \*\*\*\*\* !!!!! note that the libname and memname above must be all upper-case ;

quit;

proc print data=AllVars;

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : combining several variables into one variable

data demos;

input id ethnicity raceai racea raceaa racenh racew raceunk racenr ;

cards;

1 2 0 0 0 0 1 0 0

2 2 0 0 0 0 1 0 0

3 2 0 0 1 0 1 0 0

4 2 0 1 0 0 0 0 0

5 1 0 0 1 0 0 0 0

6 2 1 0 0 0 0 0 0

7 1 0 0 0 0 0 1 0

8 1 0 0 0 0 1 0 0

9 1 0 0 0 0 0 0 1

10 3 0 1 0 1 1 0 0

run;

data race; set demos;

\*\*\*\*\* recode ethnicity to group together Unknown and Not Reported ;

if ethnicity = 3 or ethnicity = 4 then rep\_ethnicity = 3;

else rep\_ethnicity = ethnicity;

format rc\_race $char5. rc\_unk $char2. ;

\*\*\*\*\* this "recodes" the race variables into one variable of length 5 made up of 0's and 1's ;

rc\_race = cats(raceai, racea, raceaa, racenh, racew);

rc\_unk = cats(raceunk, racenr);

\*\*\*\*\* this gives the count of the number of times a 1 occurs in: rc\_race and rc\_unk ;

c\_rc\_race = count(rc\_race, "1") ; \* counts the number of times a 1 occurs in rc\_race ;

c\_rc\_unk = count(rc\_unk , "1") ; \* counts the number of times a 1 occurs in rc\_unk ;

\*\*\*\*\* this gives the race category that matches the pattern ;

if rc\_race = "10000" then rep\_race = 1; \* i.e., american indian / AK native ! and nothing else ;

if rc\_race = "01000" then rep\_race = 2; \* i.e., asian ! and nothing else ;

if rc\_race = "00100" then rep\_race = 3; \* i.e., african american ! and nothing else ;

if rc\_race = "00010" then rep\_race = 4; \* i.e., native hawaiian ! and nothing else ;

if rc\_race = "00001" then rep\_race = 5; \* i.e., white ! and nothing else ;

\* if the count of 1's is greater than 1, the person is multi-racial ;

if c\_rc\_race > 1 then rep\_race = 6; \* i.e., more than 1 race ;

\* if the count of 1's is greater than 0, the person's race is unknown or not reported ;

if c\_rc\_unk > 0 then rep\_race = 7; \* i.e., unknown or not reported ;

label rep\_race = "Racial Categories"

 rep\_ethnicity = "Ethnic Category" ;

run;

proc format;

 value frep\_race 1 = "American Indian / AK Native"

 2 = "Asian"

 3 = "Black or African American"

 4 = "Native Hawaiian or Other PI"

 5 = "White"

 6 = "More than one race"

 7 = "Unknown or not reported" ;

 value frep\_ethnicity 1 = "Hispanic or Latino"

 2 = "Not Hispanic or Latino"

 3 = "Unknown or Not reported" ;

run;

proc freq data = race;

format rep\_race frep\_race. ;

tables rep\_race ;

run ;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : keeping the output you want

data selection;

input weightloss $ surv\_month cens @@;

datalines;

No 1 1 No 11 1 Yes 15 1

No 38 0 No 6 0 Yes 13 0

No 9 0 No 9 1 Yes 14 1

No 9 1 No 4 1 Yes 71 0

No 13 1 No 20 1 Yes 37 0

No 20 1 No 44 1 Yes 17 1

No 8 0 No 81 0 Yes 32 0

No 4 1 No 74 0 Yes 17 1

No 27 1 No 13 1 Yes 13 1

No 62 1 Yes 6 1 Yes 27 1

No 12 1 Yes 39 1 Yes 8 1

No 32 1 Yes 100 1 Yes 63 1

No 17 1 Yes 30 0 Yes 22 1

No 15 1 Yes 1 1 Yes 5 1

No 15 1 Yes 2 1 Yes 62 0

No 8 0 Yes 2 0 Yes 9 1

No 22 1 Yes 6 1 Yes 15 1

No 7 1 Yes 21 1 Yes 5 1

No 16 1 Yes 20 0 Yes 11 1

No 11 0 Yes 5 1 Yes 21 1

;

run;

\*\*\*\*\* selects from the ods output tables: Quartiles, SurvDiff ;

\*\*\*\*\* selects frou the ods output plots: SurvivalPlot ;

ods select Quartiles SurvDiff SurvivalPlot ;

proc lifetest data=selection;

time surv\_month\*cens(0);

strata weightloss/ diff=all test=logrank;

run;

ods select off;

ods trace on ;

proc lifetest data=selection;

time surv\_month\*cens(0);

strata weightloss/ diff=all test=logrank;

run;

ods trace off;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\*\*\*\*\* problem : adding footnotes to keep track of output

%LET USER = Johnny Awesome ;

%LET CURTIME=%SYSFUNC(TIME(),TIMEAMPM11.0);

%LET CURDATE=%SYSFUNC(TODAY(),WORDDATE.);

FOOTNOTE "THIS OUTPUT WAS CREATED AT &CURTIME ON &CURDATE BY &USER" ;

data selection;

input weightloss $ surv\_month cens @@;

datalines;

No 1 1 No 11 1 Yes 15 1

No 38 0 No 6 0 Yes 13 0

No 9 0 No 9 1 Yes 14 1

No 9 1 No 4 1 Yes 71 0

No 13 1 No 20 1 Yes 37 0

No 20 1 No 44 1 Yes 17 1

No 8 0 No 81 0 Yes 32 0

No 4 1 No 74 0 Yes 17 1

No 27 1 No 13 1 Yes 13 1

No 62 1 Yes 6 1 Yes 27 1

No 12 1 Yes 39 1 Yes 8 1

No 32 1 Yes 100 1 Yes 63 1

No 17 1 Yes 30 0 Yes 22 1

No 15 1 Yes 1 1 Yes 5 1

No 15 1 Yes 2 1 Yes 62 0

No 8 0 Yes 2 0 Yes 9 1

No 22 1 Yes 6 1 Yes 15 1

No 7 1 Yes 21 1 Yes 5 1

No 16 1 Yes 20 0 Yes 11 1

No 11 0 Yes 5 1 Yes 21 1

;

run;