“Rectangularizing” a Data File in SAS[1]

It is sometimes necessary to reshape data so that you can do the kind of analysis that you want to do. If the following example, there are multiple data records for individuals describing events in their lives. An example of this type of file might be arrests. But, you may need to have detailed information about each arrest on one record. This document will demonstrate a couple of ways of creating a one record per case from many. The method you would use would depend on what kind of data you wanted to end up with.

The first method demonstrated will capture all of the information from each individual record. The process is a little cumbersome, and the final product is not always what you need. The second method will show how to create summary data for the individual. Some of the details from the first file will not be carried into the second.

For these examples, I will use the following data - seven arrests for two individuals, and only four variables.

ID	Year	Crime	convicted
1	1988	burglary	1
1	1992	murder	0
1	1993	assault	1
1	1992	DUI	1
2	1989	DUI	1
2	1992	DUI	1
2	1994	Cocaine	1

Method 1

The following program creates new variables for each case. The result will be two cases - one for each ID, with information for up to four crimes (since four is the maximum cases per ID in this example). You may copy this program into an SAS syntax window and run it to see exactly how it works.

```
* rectangle1.sas ;
title1 'rectangle1 - reshaping a file - long to wide - keeping all information' ;
data arrest ;
input id year crime $15. convict ;
cards ;
1     1988  burglary          1
1     1992  murder            0
1     1993  assault           1
1     1992  DUI               1
2     1989  DUI               1
2     1992  DUI               1
2     1994  Cocaine           1
;
* Sort the data by ID - we need the cases to be grouped together    ;
* If you would like the variables to be in a certain order, include ;
* that variable in the sort.                                        ;
proc sort; by id year ;
* Create a new data set - and use a "by" statement.                 ;
* This will create a "first" and "last" variable that we will need. ;

data two ; set arrest ; by id ;
* First you must find out the maximum number of cases for an ID.    ;
* The following code will create a counter within each ID.          ;
* We will need this count variable later too. ;
if first.id then n = 0 ;    n + 1 ;
```

[1] Prepared by Patty Glynn, University of Washington. 7/16/02, updated 8/9/02 C:\all\help\helpnew\rectangle_sas.wpd

```
** Use proc freq to find out how many is max cases.  In this case, 4.;
 /*  proc freq;  tables n ;  run ;  */

* The variable names that will be used for each case follow.  A retain statement
* is required because the values need to be retained across cases. ;
retain year_1 conv_1 crim_1
       year_2 conv_2 crim_2
       year_3 conv_3 crim_3
       year_4 conv_4 crim_4 ;

* The following arrays and loops will assign values to each of the variables.  ;
** Character and numeric variables must be processed separately. ;
** Arrays cannot have both in them ;

** Numeric variables, you need as many arrays as the MAX number of cases. ;
array orig year convict  ;
array for1 year_1 conv_1 ;
array for2 year_2 conv_2 ;
array for3 year_3 conv_3 ;
array for4 year_4 conv_4 ;

** Set variables to missing for first.id ;
do over orig ;
if first.id then do ;
 for1 = . ;
 for2 = . ;
 for3 = . ;
 for4 = . ;
end ;
** Assign values to variables - depending on which case for ID. ;
 if n = 1 then for1 = orig ;
 if n = 2 then for2 = orig ;
 if n = 3 then for3 = orig ;
 if n = 4 then for4 = orig ;
end ;

** Character variable(s):  One array for each case - up to maximum N ;
array origc $ crime  ;
array for1c $ crim_1 ;
array for2c $ crim_2 ;
array for3c $ crim_3 ;
array for4c $ crim_4 ;

** Set values to missing for first case. ;
do over origc ;
if first.id then do ;
 for1c = '' ;
 for2c = '' ;
 for3c = '' ;
 for4c = '' ;
end ;
** Assign values to variables. ;
 if n = 1 then for1c = origc ;
 if n = 2 then for2c = origc ;
 if n = 3 then for3c = origc ;
 if n = 4 then for4c = origc ;
end ;

* Keep only the last case for each ID - it will have all variables with values assigned ;
if last.id ;
drop year crime convict ;
proc print uniform ;  run ;
```

Method 2

This method uses proc summary to create a data set with summarized information for each ID. For this example, I have added a variable “time_in”, which is time incarcerated.

ID	Year	Crime	convicted	time_in
1	1988	burglary	1	12
1	1992	murder	0	.
1	1993	assault	1	15
1	1992	DUI	1	1
2	1989	DUI	1	1
2	1992	DUI	1	2
2	1994	Cocaine	1	3

```
* rectangle2.sas ;
title1 'rectangle2 - a way of reshaping a file - long to wide - SUMMARIZING information' ;
data arrest ;
input id year crime $15. convict time_in ;
cards ;
1     1988        burglary    1     12
1     1992        murder            0     .
1     1993        assault           1     15
1     1992        DUI               1     1
2     1989        DUI               1     1
2     1992        DUI               1     2
2     1994        Cocaine           1     3
;
data two ; set arrest ;
* create variables for variables of interest ;

* Use code like the following to determine values that need to be categorized ;
UPCRIME = UPCASE(CRIME) ;
/* PROC FREQ; TABLES UPCRIME ; RUN ; */

if UPCRIME = 'BURGLARY' and convict = 1 then propconv = 1 ;
if UPCRIME = 'MURDER'   and convict = 1 then violconv = 1 ;
if UPCRIME = 'ASSAULT'  and convict = 1 then violconv = 1 ;
if UPCRIME = 'DUI'      and convict = 1 then alcconv   = 1 ;
if UPCRIME = 'COCAINE'  and convict = 1 then drugconv = 1 ;

if year gt . and year lt 1990 then pre1990  = 1 ;
if year               ge 1990 then post1990 = 1 ;

proc summary nway ;
class ID ;
var convict propconv violconv alcconv drugconv pre1990 post1990 time_in ;
output out = sumdat sum = ;

data fin ; set sumdat ;
narrest = _freq_ ;
label narrest  = 'N of arrests for person' ;
label propconv = 'Property crime conviction' ;
label violconv = 'Violent crime conviction' ;
label alcconv =  'Alcohol crime conviction' ;
label drugconv = 'Drug crime conviction' ;
label time_in  = 'Months in prison' ;
drop  _freq_  _type_ ;
* change missing to zeros ;
array ze propconv violconv alcconv drugconv pre1990 post1990 time_in ;
do over ze ; if ze = . then ze = 0 ; end ;

proc print uniform ; run ;
```