STRATIF: A PROCEDURE FOR PRODUCING STRATIFIED VARIABLES

M.G. Sri Ram, Div. of Computing Services, The Ohio State University College of Medicine

Introduction

Consider the common situation where a data set contains both continuous numeric variables and alphanumeric classification variables. It is desired to study the former within the separate levels of each of the latter, in other words, to stratify the former in terms of the latter. The usual method of accomplishing this is to sort the data set, call the desired procedure with the same BY variable, and repeat this for each of the classification variables. This can be a tedious affair if there are more than a few classification variables, involving multiple calls to SORT and the expenditure of much programming overhead such as disk input/output as well as program coding.

STRATIF represents an alternative approach to the stratification problem. With a few lines of code the SAS programmer can produce a data set in which specified numeric variables are stratified according to the levels of specified classification variables. Appropriate SAS procedures can then be applied to this new data set, producing results identical to those obtained by multiple sorts and multiple calls to the procedure. Further, greater control and flexibility are gained since specific variables can be stratified, rather than the entire data set.

Procedure Description

The procedure is called as follows:

```
PROC STRATIF DATA = name OUT = name/options;
VAR numeric variable list;
CLASS classification variables list;
ID identification variables list;
```

Here 'name' is a valid SAS data set name. The name following DATA is the input data set name, and that following OUT is the output data set generated by STRATIF.

Two options are allowed. These are:

- KEEPOLD or K which outputs the VAR list variables in addition to those generated by the procedure.
- NOPRINT or N which suppresses printed output.

VAR and CLASS are abbreviations for VARIABLES and CLASSES respectively. The usual SAS conventions apply to the DATA and OUT parameters as well as to the VAR statement, i.e., if omitted DATA defaults to the most recently created data set, OUT defaults to the name 'OUT' and the VAR list variables are all the numeric variables in the most recently created data set. However, a CLASS statement must be included.

STRATIF then produces an output data set containing as many observations as the input, whose variables are stratified versions of the VAR variables. The stratification is accomplished by creating a new variable for each VAR variable and each level of each CLASS variable. This new variable is set equal to its VAR 'ancestor' in observations containing that particular level of the CLASS ancestor. In all other observations it is assigned the special missing value '_A'. Clearly, there are $v \times \prod_{1}^{n} l_{i}$ stratified variables, where:

\[
v = \text{number of VAR variables}, \\
n = \text{number of CLASS variables}, \\
l_{i} = \text{number of levels of the } i^{th} \text{ CLASS variable}, \quad 1 \leq i \leq n.
\]

Names and labels of these variables are created by the procedure itself since users should not have to know class level information before using STRATIF. The label contains the ancestry of the stratified variable, describing from which VAR variable and which level of which CLASS variable it was created. When assigning names the procedure has the task of ensuring that duplicates do not occur and also that the names bear some resemblance to the ancestry of the variable. There are several ways of accomplishing this and the present version of STRATIF deals with this by the following rule. Each new variable's name begins with the string 'V', followed by at least three numbers. The first is the position of the VAR ancestor in the VAR list, the second is the position of the CLASS ancestor in the CLASS list and the last represents the order in which this particular level of the CLASS ancestor was found in the data set.

```
STRATIF produces two kinds of printed output. The first, consisting of class level information lists the levels found for the CLASS variables and issues a warning if missing values are encountered. The second is a table containing the names and labels of the stratified variables created. This printed output is provided to help the user keep track of the possibly large number of new variables created by STRATIF.

Example:

In this example data set TEST has two numeric variables COST and WEIGHT, as well as three alphanumeric variables called NAME, ANIMAL and MONTH. The data are read into TEST from cards. Figure 1 is the SAS log for this run. We see from the notes that the output data set called 'ONE' has the same number of observations as TEST but the number of variables is different. Figure 2 is a printout of TEST. Figures 3 and 4 are printed output from STRATIF. Finally, figure 5 is a printout of the data set ONE produced by STRATIF.

Applications

SAS deals with the stratification problem by means of SORT and BY statements. However, it is often easier to conceptualize and manipulate the stratification in the form of variables rather than as a data set sorted in
many ways. Also, specific variables(s) can be stratified according to classification variable(s) while the entire data set is sorted by those or others. Thus, STRATIF is both an alternative and an enhancement to SORT.

Any suitable SAS procedure can be applied to the output data set of STRATIF. Tests of normality can be obtained within levels of CLASS variables by using PROC UNIVARIATE, as is sometimes necessary prior to analysis of variance. Stratified descriptive statistics can be obtained using PROC MEANS. The results are identical to those obtained from PROC SUMMARY but there is no need to manipulate SUMMARY's TYPE variable. In most cases we can call the desired procedure without even specifying a VAR list for that procedure. The labels provided by STRATIF identify the variable completely. Used in this fashion, STRATIF represents an alternative to a macro which repetitively executes calls to SORT and the desired procedure. This may lead to decreased execution overhead.

Here is a data step application. Suppose in the above example it is desired to reorganize the data set such that for each level of ANIMAL there is only one observation in which COST is represented for every level of MONTH. This could be used later to obtain statistics on monthly differences in COST. After deleting the tenth observation, in which MONTH is missing, the following code accomplishes this task.

```sas
PROC SORT DATA = TEST ; BY MONTH ANIMAL ;
PROC STRATIF DATA = TEST OUT = TWO ;
   COST WEIGHT ;
   CLASS MONTH ;
   ID ANIMAL MONTH ;
DATA D1 D2 D3 ;
D1 ( DROP = V111 - V113 ) ;
D2 ( DROP = V211 - V213 ) ;
D3 ( DROP = V311 - V313 ) ;
IF MONTH = 'FEB' THEN OUTPUT D1 ;
IF MONTH = 'JAN' THEN OUTPUT D2 ;
IF MONTH = 'MAR' THEN OUTPUT D3 ;
DATA LAGGED ;
   MERGE D1 D2 D3 ;
   BY ANIMAL ;
   DROP MONTH ;
```

If there is a large number of VAR variables the above code eliminates many RENAMEs.

Suppose next that it is desired to treat the variable COST in a multivariate fashion across the variable MONTH. The following code creates a data set in which COST is transformed into three variables, one for each value of MONTH.

```sas
PROC STRATIF DATA = TEST OUT = ONE ;
   COST ;
   CLASSES MONTH ;
   ID NAME ;
   PROC SORT DATA = ONE ;
   BY NAME ;
   PROC MEANS NOPRINT DATA = ONE ;
   BY NAME ;
   var V111 - V113 ;
   output out = TWO ;
   sum = S1 - S3 ;
```

Multivariate analysis, e.g., correlations, MANOVA, etc., can now be performed on the variables S1, S2, and S3 in data set TWO.

Details:
1) STRATIF deals with missing values in two ways. If an observation contains a missing value for a CLASS variable, a warning is printed but no new variable is created. Missing values for VAR variables are left untouched, i.e., the descendent stratified variables are set equal to missing in that observation.

2) The BY statement is meaningless to STRATIF. If found it is ignored.

3) The ID variable list may be used to add variables to the data set created by STRATIF. These must, of course, be present in the input data set to STRATIF and may include those already present in the CLASS list.

4) STRATIF is written in PL/I and compiled with the optimizing compiler. The parsing module was assembled and loaded using SAS macro libraries.

5) The present version of the procedure has a limit of ten CLASS variables with ten levels each.

Future Directions
Several changes and additions are being considered. These include:

1) Allowing alphanumeric variables in the VAR list.

2) Removing the restriction on CLASS variables mentioned in (5) of Details.

3) A TWOWAY option to obtain two-way stratified variables.

Acknowledgements
The author wishes to express his thanks to Mark McNair, Joe Danico, and Fred Ruland for explaining the SAS Programmers Guide, invaluable programming consultation, encouragement, and suggestions for improvement. Mark is a colleague at the Division of Computing Services, College of Medicine, Ohio State University; Joe and Fred are analysts at the Instruction and Research Computer Center, The Ohio State University.

References
NOTE: THE JOB TS0560874 HAS BEEN RUN UNDER RELEASE 82.3 OF SAS AT OHIO STATE UNIVERSITY.

NOTE: CPUID VERSION = 6B SERIAL = 070206 MODEL = 6478.

NOTE: SAS OPTIONS SPECIFIED ARE:
SORT=.

TITLE PROCEDURE STRATIF EXAMPLE:;
DATA TEST:
   INPUT @1 COST 2.
   4 NAME 6.
   6 ANIMAL 6.
   8 NOTCH 6.
   10 WEIGHT 2.

PROCEDURE STRATIF EXAMPLE

OBS  COST  NAME  ANIMAL  MONTH  WEIGHT
1  18  SALLY  TIGER  FEB  5
2  28  SALLY  TIGER  JAN  6
3  40  SALLY  TIGER  MAR  7
4  50  LINUS  PUPA  MAR  7
5  48  LINUS  PUPA  FEB  9
6  45  LINUS  PUPA  JAN  9
7  40  LUCY  LION  MAR  2
8  40  LUCY  LION  JAN  2
9  45  LUCY  LION  JAN  5
10 50  PUNA  FEB  5

Figure 1: PROC STRATIF Example

PROCEDURE STRATIF EXAMPLE

OBS  COST  NAME  ANIMAL  MONTH  WEIGHT
1  18  SALLY  TIGER  FEB  5
2  28  SALLY  TIGER  JAN  6
3  40  SALLY  TIGER  MAR  7
4  50  LINUS  PUPA  MAR  7
5  48  LINUS  PUPA  FEB  9
6  45  LINUS  PUPA  JAN  9
7  40  LUCY  LION  MAR  2
8  40  LUCY  LION  JAN  2
9  45  LUCY  LION  JAN  5
10 50  PUNA  FEB  5

Figure 2: Input data set to STRATIF
PROCEDURE STRATIF EXAMPLE

CLASS LEVEL INFORMATION

NUMBER OF LEVELS FOUND FOR NAME = 3
LEVEL NAMES FOR NAME ARE: SALLY LINUS LUCY

WARNING: MISSING VALUE(S) FOUND FOR NAME

NUMBER OF LEVELS FOUND FOR ANIMAL = 3
LEVEL NAMES FOR ANIMAL ARE: TIGER PUMA LION

NUMBER OF LEVELS FOUND FOR MONTH = 3
LEVEL NAMES FOR MONTH ARE: FEB JAN MAR

Figure 3: Printed output from STRATIF

PROCEDURE STRATIF EXAMPLE

VARIABLES GENERATED BY STRATIF

<table>
<thead>
<tr>
<th>NAME</th>
<th>LABEL</th>
<th>ATTRIBUTE</th>
<th>NAME</th>
<th>LABEL</th>
<th>ATTRIBUTE</th>
</tr>
</thead>
<tbody>
<tr>
<td>_V111</td>
<td>COST</td>
<td>FOR LEVEL SALLY</td>
<td>OF NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V112</td>
<td>COST</td>
<td>FOR LEVEL LINUS</td>
<td>OF NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V113</td>
<td>COST</td>
<td>FOR LEVEL LUCY</td>
<td>OF NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V121</td>
<td>COST</td>
<td>FOR LEVEL TIGER</td>
<td>OF ANIMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V122</td>
<td>COST</td>
<td>FOR LEVEL PUMA</td>
<td>OF ANIMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V123</td>
<td>COST</td>
<td>FOR LEVEL LION</td>
<td>OF ANIMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V131</td>
<td>COST</td>
<td>FOR LEVEL FEB</td>
<td>OF MONTH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V132</td>
<td>COST</td>
<td>FOR LEVEL JAN</td>
<td>OF MONTH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V133</td>
<td>COST</td>
<td>FOR LEVEL MAR</td>
<td>OF MONTH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V211</td>
<td>WEIGHT</td>
<td>FOR LEVEL SALLY</td>
<td>OF NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V212</td>
<td>WEIGHT</td>
<td>FOR LEVEL LINUS</td>
<td>OF NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V213</td>
<td>WEIGHT</td>
<td>FOR LEVEL LUCY</td>
<td>OF NAME</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V221</td>
<td>WEIGHT</td>
<td>FOR LEVEL TIGER</td>
<td>OF ANIMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V222</td>
<td>WEIGHT</td>
<td>FOR LEVEL PUMA</td>
<td>OF ANIMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V223</td>
<td>WEIGHT</td>
<td>FOR LEVEL LION</td>
<td>OF ANIMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V231</td>
<td>WEIGHT</td>
<td>FOR LEVEL FEB</td>
<td>OF MONTH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V232</td>
<td>WEIGHT</td>
<td>FOR LEVEL JAN</td>
<td>OF MONTH</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_V233</td>
<td>WEIGHT</td>
<td>FOR LEVEL MAR</td>
<td>OF MONTH</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 4: Printed output from STRATIF
### PROCEDURE STRATIF EXAMPLE

<table>
<thead>
<tr>
<th>OBS</th>
<th>_V11</th>
<th>_V12</th>
<th>_V13</th>
<th>_V121</th>
<th>_V122</th>
<th>_V131</th>
<th>_V132</th>
<th>_V133</th>
<th>_V211</th>
<th>_V212</th>
<th>_V213</th>
<th>_V221</th>
<th>_V222</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10 A</td>
<td>A</td>
<td>A</td>
<td>10 A</td>
<td>A</td>
<td>A</td>
<td>10 A</td>
<td>A</td>
<td>5 A</td>
<td>A</td>
<td>5 A</td>
<td>A</td>
<td>5 A</td>
</tr>
<tr>
<td>2</td>
<td>20 A</td>
<td>A</td>
<td>A</td>
<td>20 A</td>
<td>A</td>
<td>A</td>
<td>20 A</td>
<td>A</td>
<td>6 A</td>
<td>A</td>
<td>6 A</td>
<td>A</td>
<td>6 A</td>
</tr>
<tr>
<td>3</td>
<td>40 A</td>
<td>A</td>
<td>A</td>
<td>40 A</td>
<td>A</td>
<td>A</td>
<td>40 A</td>
<td>A</td>
<td>7 A</td>
<td>A</td>
<td>7 A</td>
<td>A</td>
<td>7 A</td>
</tr>
<tr>
<td>4</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>40 A</td>
<td>A</td>
<td>A</td>
<td>40 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
</tr>
<tr>
<td>5</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>40 A</td>
<td>A</td>
<td>A</td>
<td>40 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
</tr>
<tr>
<td>6</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>45 A</td>
<td>A</td>
<td>A</td>
<td>45 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
</tr>
<tr>
<td>7</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>45 A</td>
<td>A</td>
<td>A</td>
<td>45 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
</tr>
<tr>
<td>8</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>49 A</td>
<td>A</td>
<td>A</td>
<td>49 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
</tr>
<tr>
<td>9</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>45 A</td>
<td>A</td>
<td>A</td>
<td>45 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
</tr>
<tr>
<td>10</td>
<td>A</td>
<td>A</td>
<td>A</td>
<td>50 A</td>
<td>A</td>
<td>A</td>
<td>50 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
<td>A</td>
<td>9 A</td>
</tr>
</tbody>
</table>

Figure 5: Output data set produced by STRATIF