PROC TRANSPOSE--A Powerful Tool for Data Restructuring
Bernard Pardo, Alza Corporation

Introduction

Data generated by clinical pharmaceutical studies at Alza Corporation are frequently structured in a manner dictated by the layout of the data collection forms, the format of both in-house and out-of-house blood and urine analytical data, physician and nurse data forms, as well as other constraints. Data files created from such data sources often do not have a structure which can be readily input to a SAS PROC or listed in a report table, and therefore these data files need to be restructured. While such transformations can be accomplished with the DATA step, PROC TRANSPOSE is often the most simple and powerful method of restructuring data files.

How Data Structures are Born

The data collected in clinical studies come from several possible sources: analytical laboratories, medical research facilities, hospitals, clinical monitor documents, and various data collection forms. Data may be recorded electronically or by entering data on hard-copy forms. Data collection forms should be designed to facilitate the accurate and expeditious capture of data, and to allow easy transfer of data from the forms to computer data files. The computer data files resulting from these various sources, however, have structures which usually do not directly lend themselves to analysis or listing in reports. For example, each record may contain one observation and the time of the observation per subject, and a given subject may have observations over a number of timepoints. This file can be restructured to present all data for a single subject in one record. The corresponding report table would list each subject in a row, with columns containing the data for a given timepoint.

Simple Reversing of Rows and Columns (Observations and Variables)

The rows and columns of a file can be easily reversed using PROC TRANSPOSE. In the following example, 9 blood analysis values were recorded for each study subject according to a fixed timetable. Each record of the resulting file contains all of the subject's data in the same order. In order to produce a table where the time parameter corresponds to rows, the rows and columns must be reversed. The following program accomplishes this task:

```
PROGRAM
DATA tran1;
  INPUT subject lev1-lev9; CARDS;
    3 0 0 0 3.9 3.5 2.9 2 1.4 2
    14 0 0 0.2 2.6 3.2 2.1 6.1 5
    16 0.2 0.2 0.3 1.8 1.6 3.5 2.5 2.1 2.4
    10 0 0 1.3 3.5 4.3 2.8 3.3 2.7
    2 0 0 0 1.3 2.5 2.3 2.3 1.2 1.4
    13 0.2 0.2 1.2 3.5 5.7 3.6 4.1 3.4 3.1
    6 0 0 1 4.5 5.4 4.8 4.2 4.6 2.5
  RUN;
  PROC SORT; BY subject; RUN;
  PROC TRANSPOSE; RUN;
  PROC PRINT NOOBS; RUN;
  
OUTPUT
```

```
 _NAME_  COl1  COl2  COl3  COl4  COl5  COl6  COl7
SUBJECT  2.0  3.0  6.0  10.0  13.0  14.0  16.0
LEV1     0.0  0.0  0.0  0.0  0.2  0.0  0.2
LEV2     0.0  0.0  0.0  0.0  0.2  0.0  0.2
LEV3     0.0  0.0  1.0  0.0  1.2  0.2  0.8
LEV4     1.3  3.9  4.5  1.3  3.5  2.6  1.8
LEV5     2.5  3.5  5.4  3.5  5.7  3.0  1.6
LEV6     2.3  2.9  4.0  4.3  3.6  3.2  3.5
LEV7     2.3  2.0  4.2  2.8  4.1  2.3  2.5
LEV8     1.2  1.4  4.6  3.3  3.4  1.6  2.1
LEV9     1.4  2.0  2.5  2.7  3.1  1.5  2.4
```
One Output Record from Multiple Input Records

Using the optional statements of PROC TRANSPOSE provides some interesting data restructuring possibilities. Consider a data set where several measurements were made on a subject over time, each record containing the subject number and one time and measurement. The goal is to create a table where each line of the table presents all the data for a single subject. The following program uses the BY statement to group the data by subject, and the ID statement to label the data columns of the output:

```
PROGRAM

DATA tran2;
  INPUT subject time value;
  CARDS;
  1 0800 1.1
  1 0900 2.1
  1 1000 3.1
  1 1100 4.1
  2 0800 1.2
  2 0900 2.2
  2 1000 3.2
  3 0800 1.3
  3 0900 2.3
  4 0900 1.4
  4 1200 2.4
RUN;

PROC TRANSPOSE; BY subject; ID time; RUN;
```

One advantage of this particular data restructuring is that it provides a quick summary of missing values.

Another possible application with this data set is to produce a table where each row of the table contains all subjects’ data for a single timepoint. Using the same input data, the following program produces the desired result, with columns labeled by subject number:

```
PROGRAM

DATA tran4;
  INPUT subject time value;
  CARDS;
  1 0800 1.1
  1 0900 2.1
  1 1000 3.1
  1 1100 4.1
  2 0800 1.2
  2 0900 2.2
  2 1000 3.2
  3 0800 1.3
  3 0900 2.3
  4 0900 1.4
  4 1200 2.4
RUN;

PROC SORT; BY time; RUN;
PROC TRANSPOSE; BY time; ID subject; RUN;
PROC PRINT NOOBS; RUN;
```

```
OUTPUT

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

927
Multiple Output Records from One Input Record

The data structure which corresponds to a report table format will generally not be suitable for statistical analysis with PROC MEANS, for example. In the following example, each record of the input file contains the subject number and 7 measurements corresponding to various times. The BY statement forces each measurement for a subject to a single record:

```
PROGRAM
DATA tran5;
  INPUT scn h00 h01 h02 h06 h24 h48 h72;
  CARDS;
  1 72  72  56  96  72  72  80
  2 80  72  80  88  76  76  88
  3 64  72  92 106  80  72  72
  4 58  52  68  76  64  60  64
  5 52  62  66  84  56  58  60
RUN;
PROC TRANSPOSE NAME=hour; BY scn; RUN;
PROC PRINT NOOBS; RUN;
```

```
PROGRAM
PROC MEANS FW=9 MAXDEC=1 MEAN STD;
CLASS HOUR;
VAR COL1; RUN;
```

```
OUTPUT
Analysis variable : COL1
```

```
<table>
<thead>
<tr>
<th>HOUR</th>
<th>N Obs</th>
<th>Mean</th>
<th>Std Dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>H00</td>
<td>5</td>
<td>67.2</td>
<td>10.4</td>
</tr>
<tr>
<td>H01</td>
<td>5</td>
<td>65.0</td>
<td>8.9</td>
</tr>
<tr>
<td>H02</td>
<td>5</td>
<td>71.6</td>
<td>14.4</td>
</tr>
<tr>
<td>H06</td>
<td>5</td>
<td>90.0</td>
<td>11.5</td>
</tr>
<tr>
<td>H24</td>
<td>5</td>
<td>69.6</td>
<td>9.5</td>
</tr>
<tr>
<td>H48</td>
<td>5</td>
<td>67.6</td>
<td>8.0</td>
</tr>
<tr>
<td>H72</td>
<td>5</td>
<td>72.8</td>
<td>11.5</td>
</tr>
</tbody>
</table>
```

Conclusion

The data files encountered in clinical studies often require restructuring. PROC TRANSPOSE provides some powerful options to accomplish various restructuring needs, such as table presentation, statistical analysis, and summarizing missing values.

SAS is the registered trademark of SAS Institute Inc., Cary, NC, USA.