

An Alternative Method of Transposing Data Without the Transpose Procedure

Sunil Kumar Gupta

Gupta Programming, Thousand Oaks, CA

ABSTRACT

While the PROC TRANSPOSE procedure may be flexible enough to generate a variety of report formats, it may not be appropriate if the user requests a custom report. In addition, for some cases, it may not be possible to successfully apply the Transpose procedure to obtain the desired output.

A technique is presented that will provide the user with more control on the number and order of the columns generated, enable new columns to be created, and allow for transposing of several variables from different data files. The features of the two methods are compared and discussed.

INTRODUCTION

Often, there is a need to rearrange data to fit a specific format for presentation or statistical analysis. This involves rotating the axis of information from one dimension to another. Mechanically, this means the turning of all columns into rows or rows into columns. This results in observations being new variables. In the final output, information is usually summarized by a given set of variables over another set of variables.

The need for custom report generation often requires an alternative method to the Transpose procedure to obtain the desired output. This can be achieved through a series of PROC SUMMARY and DATA STEPS.

The features of the two methods are compared and discussed.

SIMILARITIES & DIFFERENCES

SAMPLE DATA SET

The sample data set contains the temperature during the winter and summer seasons in several cities. There is one record for each city and season.

```
DATA LIST;
INPUT STATE $2. CITY $15. SEASON $15.
      TEMP 4.1.;

LABEL STATE = 'State'
      CITY = 'City'
      SEASON = 'Season'
      TEMP = 'Temperature';
```

```
FORMAT
      STATE $2.
      CITY $15.
      SEASON $15.
      TEMP 4.1;

CARDS;
NC Raleigh Winter 60
NC Raleigh Summer 80
NC Greensboro Winter 65
FL Miami Summer 82.9
FL Miami Winter 77
;
RUN;
```

PROC TRANSPOSE

The Transpose procedure assumes the data to be summarized by the grouping set of variables.

1. Sort by the grouping variables. These variables will be summarized in the report.

```
PROC SORT;
      BY STATE CITY;
RUN;
```

2. Apply the Transpose procedure. This rotates the data.

The OUT option allows for saving the results into a data set. The BY statement identifies the grouping variables. These grouping variables are not transposed. The ID statement identifies the variable to represent the columns. The VAR statement identifies the measuring variable whose value is reported. A format statement can be applied to the measuring variable.

```
PROC TRANSPOSE DATA=LIST OUT=LISTM3;
      BY STATE CITY;
      ID SEASON;
      VAR TEMP;
      FORMAT TEMP 4.1.;
RUN;
```

Output - The Transposed Data Set

STATE	CITY	_NAME_	WINTER	SUMMER
NC	Raleigh	TEMP	60	80
NC	Greensboro	TEMP	65	.
FL	Miami	TEMP	77	82.9

ALTERNATIVE METHOD TO PROC TRANSPOSE

This method consists of a series of PROC SUMMARY and DATA STEPS to achieve the reporting data set.

1. Summarize by the grouping variables STATE & CITY. This will formulate the master reporting dataset.

```
PROC SUMMARY DATA=LIST NWAY MISSING;
  CLASS STATE CITY;
  VAR TEMP;
  OUTPUT OUT=LISTS1
  (DROP=_TYPE_ _FREQ_) SUM=;
RUN;
```

2. Summarize by the grouping variables STATE, CITY, & SEASON. Depending on what type of data is collected, the MEAN= option may be more appropriate to calculate an average value as compared to the summation of the values. This will summarize the measuring variable by all the possible combinations of the analysis variables.

```
PROC SUMMARY DATA=LIST NWAY MISSING;
  CLASS STATE CITY SEASON;
  VAR TEMP;
  OUTPUT OUT=LISTS2
  (DROP=_TYPE_ _FREQ_) SUM=;
RUN;
```

3. Create column fields in the master dataset and initialize them to zero. These fields will store the summarized measuring variables for the report layout.

```
DATA LISTM1;
  SET LISTS1;
  WINTER=. ;
  SUMMER=. ;
  DROP=TEMP;
RUN;
```

4. Rename grouping fields in the second data set. This will allow for easier identification of the source of common fields. This step is needed to allow the matching of the grouping variables.

```
DATA LISTS2R;
  SET LISTS2;
  RENAME STATE=STATE1;
  RENAME CITY=CITY1;
RUN;
```

5. Construct nested do-loop to store the measuring variable for matching group variables. This step rearranges the data's orientation from columns to rows.

```
DATA LISTM2;
  PUT @1 'START 1';
  OBSNUM=1;
  DO OBSNUM=1 TO LASTR2;
  SET LISTM1 POINT=OBSNUM NOBS=LASTR2;
```

```
OBS=1;
DO OBS=1 TO LASTR;
  SET LISTS2R POINT=OBS OBS=LASTR;

  IF STATE=STATE1 AND CITY=CITY1 THEN DO;

    IF SEASON='Winter' THEN WINTER=TEMP;

    IF SEASON='Summer' THEN SUMMER=TEMP;

  OUTPUT;

  END;
END;

END;
STOP;

RUN;
```

Note: For additional measuring variables, more conditional statements are needed to store the information in the column fields.

6. Select only one record per grouping variables STATE & CITY. Because of the OUTPUT statement, multiple records can be generated for the grouping variable. As a result, the data set needs to be indexed by the grouping variable and only a single copy of the record should be stored in the resulting data set. This constitutes the final reporting data set.

```
DATA LISTM3;
  SET LISTM2;
  BY STATE CITY;
  IF LAST.CITY;
RUN;
```

Output - The Transposed Data Set

STATE	CITY	WINTER	SUMMER
NC	Raleigh	60	80
NC	Greensboro	65	.
FL	Miami	77	82.9

ADVANTAGES

A variety of reports can be generated with the proposed alternative method. Several advantages by the alternative method were found.

- It allows for greater control on the number of columns in the report. New columns can be created. In addition, the order of columns can be specified. Unwanted columns can be removed.
- It is a valid method for custom reports requiring the transposing of several variables from different data files.

- It allows for the addition of a percent column in the report.

DISADVANTAGES

There are several disadvantages to the alternative method.

- The proposed method requires a greater understanding of the DATA STEP. It also requires a solid understanding of PROC SUMMARY. In addition, it may take a longer time to write the program.
- The proposed alternative to the Transpose Procedure works best for a specified number of columns in the report. This is because a field must be defined for each case of the transposed variable. Therefore, if the ID variable contains many types of cases, then the Transpose Procedure is a better choice because it automatically accounts for this. The Transpose Procedure automatically creates a column for each case.

SUMMARY

The SAS programmer has a variety of tools available to complete a reporting objective. The need may arise to become creative in solving reporting obstacles. As often is the case, there is more than one way to achieve an outcome.

REFERENCES

SAS®Procedures Guide, ver. 6, third edition.

TRADEMARK INFORMATION

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

ABOUT THE AUTHOR

The author welcomes your comments & suggestions on other methods of transposing data.

Sunil Kumar Gupta
Gupta Programming
926 Tupelo Wood Ct.
Thousand Oaks, CA 91320-3650
(805) 499-9691
sunilgupta@aol.com



ACKNOWLEDGMENTS

The author would like to thank Dr. Fred Hoehler of Data Management Center, Santa Ana, CA and Kirk Paul Lafler of Software Intelligence Corporation, Spring Valley, CA for their invaluable assistance in the preparation of this paper.