PharmaSUG 2012 - Paper CC27

“V” Can Transpose: Transposing Data Sets Using Functions

Neha Mohan, PharmaNet/i3, Mumbai, India

ABSTRACT

When a formatted character variable is transposed simultaneously with other variables, the PROC TRANPOSE procedure returns unformatted values of the character variable. In cases when the formatted values associated with the character variable are desired, it is required to either transpose the corresponding variable separately or alternately create a new variable holding the corresponding formatted values, and then transpose this new variable along with other variables.

This paper demonstrates the use of three “**V**” functions namely **V**VALUE, **V**NAME and **V**LABEL in transposing such variables; function VVALUE, to directly return the formatted values associated with the variables (that a PROC TRANSPOSE would not) and functions VNAME and VLABEL, to maintain the list of original variable names and their labels respectively (that a PROC TRANSPOSE would return). In addition, these functions also enable the original variables to be retained in the resultant transposed data set without having to perform any additional processing.

The use of these functions can be further extended to the DOW-loop as well.

PROBLEM

Suppose we have demographic data for patients participating in a clinical trial. Here is a snapshot of the data (sample data set TEST) consisting of a mix of character and numeric variables:

USUBJID	**AGE**	**SEX**	**RACE**	**DOB**
101-00101	1.93840	M	2	18660
101-00102	1.83984	M	1	18689

Table 1: Sample data set TEST

Variable	**Label**	**Type**	**Format**
AGE	Age (Months)	Numeric	8.1
SEX	Gender	Character	$Sex. (‘M’ = 'Male'; ‘F’ = ‘Female’)
RACE	Ethnicity	Character	$Race. (‘1’ = ‘Hispanic’; ‘2’ = ‘Non-Hispanic’)
DOB	Date of Birth	Numeric	Date9.

Table 2: Variable attributes of data set TEST

We need the reporting data set to look like this:

USUBJID	**VAR**	**LABL**	**VAL**
101-00101	AGE	Age (Months)	1.9
101-00101	SEX	Gender	Male
101-00101	Race	Ethnicity	Non-Hispanic
101-00101	DOB	Date of Birth	02FEB2011
101-00102	AGE	Age (Months)	1.8
101-00102	SEX	Gender	Male
101-00102	Race	Ethnicity	Hispanic
101-00102	DOB	Date of Birth	03MAR2011

Table 3: Required reporting data set

If we directly specify PROC TRANPOSE with data set sample data set TEST as the input data set using the following code, not all of the values will be as required; formatted values of only the numeric variables are returned as can be seen in the output data set TEST_T.

```
PROC TRANSPOSE DATA = test
               OUT = test_t (RENAME=(COL1=VAL _NAME_=VAR _LABEL_=LABL));
  BY usubjid;
  VAR age sex race dob;
RUN;
```

USUBJID	VAR	LABL	VAL
101-00101	AGE	Age (Months)	1.9
101-00101	SEX	Gender	**M**
101-00101	Race	Ethnicity	**2**
101-00101	DOB	Date of Birth	02FEB2011
101-00102	AGE	Age (Months)	1.8
101-00102	SEX	Gender	**M**
101-00102	Race	Ethnicity	**1**
101-00102	DOB	Date of Birth	03MAR2011

Table 4: Output data set TEST_T

In order to extract the formatted values of these character variables, we will have to first create new variables that will hold the associated formatted values and then transpose these new variables as follows:

```
DATA new;
  SET test;
  race_ = PUT(race, $race.);
  sex_  = PUT(sex, $sex.);
  LABEL sex_  = 'Gender'
        Race_ = 'Ethnicity';
RUN;

PROC TRANSPOSE DATA = new
               OUT = new_t (RENAME=(COL1=VAL _NAME_=VAR _LABEL_=LABL));
  BY usubjid;
  VAR age sex_ race_ dob;
RUN;
```

USUBJID	VAR	LABL	VAL
101-00101	AGE	Age (Months)	1.9
101-00101	SEX	Gender	Male
101-00101	Race	Ethnicity	Non-Hispanic
101-00101	DOB	Date of Birth	02FEB2011
101-00102	AGE	Age (Months)	1.8
101-00102	SEX	Gender	Male
101-00102	Race	Ethnicity	Hispanic
101-00102	DOB	Date of Birth	03MAR2011

Table 5: Output data set NEW_T

ALTERNATIVE SOLUTION

Alternatively, we can make use of the VVALUE function, in conjunction with ARRAYs in order to get the formatted values directly, without using PROC TRANSPOSE. At the same time, since the name and label of the original variables are also required, then functions VNAME and VLABEL can also be used.

- VVALUE Function
  Returns the formatted value that is associated with the variable that you specify
  Syntax : VVALUE(*var*) , where *var* specifies a variable that is expressed as a scalar or as an array reference.
  Restriction: You cannot use an expression as an argument.

- VNAME Function
  Returns the name of the specified variable.
  Syntax : VNAME(*var*), where *var* specifies a variable that is expressed as a scalar or as an array reference.
  Restriction: You cannot use an expression as an argument.

- VLABEL Function
  Returns the label that is associated with the specified variable. If there is no label, VLABEL returns the variable name.
  Syntax : VLABEL(var), where var specifies a variable that is expressed as a scalar or as an array reference.
  Restriction: You cannot use an expression as an argument.

The code to generate the desired reporting data set directly from the input data set using these three “V” functions looks like this:

```
DATA out1;
  SET test;
  LENGTH var val $50 labl $100;
  BY usubjid;
  ARRAY abc1{*} sex race; /* Array for character variables */
  ARRAY abc2{*} age dob; /* Array for numeric variables */
  /* Processing of character variables */
  DO i=1 TO dim(abc1);
      var  = VNAME(abc1{i});    /* get name of the variable         */
      labl = VLABEL(abc1{i});   /* get label of the variable     */
      val  = VVALUE(abc1{i});   /* get formatted value of the variable */
      OUTPUT;
  END;
  /* Processing of numeric variables */
  DO j=1 TO dim(abc2);
      var  = VNAME(abc2{i}); /* get name of the variable */
      labl = VLABEL(abc2{i});   /* get label of the variable */
      val  = VVALUE(abc2{i});  /* get formatted value of the variable */
      OUTPUT;
  END;
RUN;
```

The resultant output data set (OUT1) has values in the format required by the reporting data set.

USUBJID	AGE	SEX	RACE	DOB	VAR	LABL	VAL
101-00101	1.93840	M	2	18660	AGE	Age (Months)	1.9
101-00101	1.93840	M	2	18660	SEX	Gender	Male
101-00101	1.93840	M	2	18660	Race	Ethnicity	Non-Hispanic
101-00101	1.93840	M	2	18660	DOB	Date of Birth	02FEB2011
101-00102	1.83984	M	1	18689	AGE	Age (Months)	1.8
101-00102	1.83984	M	1	18689	SEX	Gender	Male
101-00102	1.83984	M	1	18689	Race	Ethnicity	Hispanic
101-00102	1.83984	M	1	18689	DOB	Date of Birth	03MAR2011

Table 6: Output data set OUT1

FURTHER SCOPE

The standard PROC TRANSPOSE returns the formatted values of only numeric variables when both character and numeric variables are passed together (as seen in the example described in the paper). On the other hand, if formatted variables, all of the same type (either all character or all numeric) but with different formats are passed, the procedure would not return any formatted values at all. The suggested alternative approach would also be useful in such cases.

In addition, the usage of these functions can be extended to the DOW-Loop as well to perform more complex data set transpositions.

CONCLUSION

The use of the three "**V**" functions (**V**VALUE, **V**LABEL, and **V**NAME), facilitates the extraction of formatted values of variables along with the list of original variable names and labels while transposing data sets using ARRAYs and eliminates the need for a DATA step and PROC TRANSPOSE procedure. In addition, the original variables can also be retained in the resultant transposed data set without having to perform any additional processing.

ACKNOWLEDGMENTS

Thank you to Nancy Brucken (Principal Statistical Programmer, PharmaNet/i3) and my colleagues, Gayatri Karkera and Usha Kumar, for reviewing this paper for me; and a special thanks to Sandeep Sawant (Manager, Statistical Programming, PharmaNet/i3) for the encouragement.

REFERENCES

- SAS® support website: http://support.sas.com/documentation/cdl/en/lrdict/64316/HTML/default/viewer.htm#a000245860.htm

- Brucken, Nancy. *"2 PROC TRANSPOSEs = 1 DATA Step DOW-Loop"*. Proceedings of the PharmaSUG 2011 conference. www.pharmasug.org/proceedings/2011/CC/PharmaSUG-2011-CC26.pdf

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Name: Neha Mohan
Enterprise: PharmaNet/i3
Address: 7th Floor, Corporate Center, Andheri-Kurla Road, Andheri (E)
City, State ZIP: Mumbai-400059, Maharashtra, India
Work Phone: +91-22-30554000
E-mail: nmohan@pharmanet-i3.com ; neha.nm@gmail.com