

Use of PROC TABULATE Out File to Customize Tables

Abdus Shahid, Western Oregon University

ABSTRACT

Out file from PROC TABULATE process of SAS® can be used to customize final output tables needed for reports. This method is especially useful when PROC TABULATE is used for several disparate data sets but the final output requires data to be presented in one final table.

INTRODUCTION

Data required for reports do not necessarily come from one set. In many instances, it is not possible to merge all the data into one set. Sometimes, even if all the data is in one set, PROC TABULATE requires different conditional statements to omit missing data from tabulation. Out files from PROC TABULATE process become especially useful in these circumstances as we can select particular rows or columns from out files and merge those in one set for final output.

PROC TABULATE

Institutions of higher education collect student data from various sources and store in different tables. For example, office of admission and registrar's office do not necessarily use same data source or collection and storage methods. Office of admission may collect gender information and high school GPA of students while registrar's office may collect their SAT and ACT scores. Or, both offices may collect gender information but one may store gender data in numeric format while the other in character format. Additionally, office of admission may input missing values as blanks whereas registrar's office may code missing values as zeros. However, the office of institutional research of the same institution may need to produce a report that shows average high school GPA, SAT scores and age by gender in one table. Data sets can be merged into one but that may not necessarily produce a table the way we want to see a table with accurate averages of these variables. Merging different student tables assumes that all the tables have at least one common unique identifier which may not always be the case.

Let us take the following table as example:

	Female	Male
Applicants:		
Average Age:		
Average HS GPA:		
Average SAT Math Score:		
Average ACT Math Score:		

Table 1. Sample Table for Final Report

For the sake of simplicity, discussion from this point will be confined to HS GPA and SAT Math Score only.

First, look at codes and output when we try to use PROC TABULATE where both the variables are in one data set:

```
proc tabulate data = applied_2016 format=15.1 missing
  classdata=ipeds_fmt out = Scores_2016;
  title1 "Applied Average HSGPA and SAT Math";
  class sex;
  var hsgpa SAT_MATH;
  tables (hsgpa SAT_MATH) * (Mean), sex=' '
    /rts=60 misstext='0' box=' ' printmiss;
  format sex sex.;
  keylabel n = ' ';
  where hsgpa ne . or SAT_MATH ne 0;
run;
```

	Female	Male
Applicants:		
Average HS GPA:	3.1	3.1
Average SAT Math Score:	287.9	311.6

Table 2. Averages with First Conditional Statement

Now look at the codes and output when we change the conditional statement:

```
proc tabulate data = applied_2016 format=15.1 missing
  classdata=ipeds_fmt out = Scores_2016;
  title1 "Applied Average HSGPA and SAT Math";
  class sex;
  var hsgpa SAT_MATH;
  tables (hsgpa SAT_MATH) * (Mean), sex=' '
    /rts=60 misstext='0' box=' ' printmiss;
  format sex sex.;
  keylabel n = ' ';
  where hsgpa ne . and SAT_MATH ne 0;
run;
```

	Female	Male
Applicants:		
Average HS GPA:	3.2	3.2
Average SAT Math Score:	447.8	524.4

Table 3. Averages with Second Conditional Statement

Finally, let us tabulate the averages separately:

(only GPA)

```
proc tabulate data = applied_2016 format=15.1 missing
              classdata=ipeds_fmt out = GPA_2016;
  title1 "Applied Average HSGPA";
  class sex;
  var hsgpa;
  tables (hsgpa) * (Mean),sex=' '
        /rts=60 misstext='0' box=' ' printmiss;

  format sex sex.;
  keylabel n = ' ';
  where hsgpa ne .;
run;
```

(only SAT Math score):

```
proc tabulate data = applied_2016 format=15.1 missing
              classdata=ipeds_fmt out = SAT_2016;
  title1 "Applied Average SAT Math";
  class sex;
  var SAT_MATH;
  tables (SAT_MATH) * (Mean),sex=' '
        /rts=60 misstext='0' box=' ' printmiss;

  format sex sex.;
  keylabel n = ' ';
  where SAT_MATH ne 0;
run;
```

	Female	Male
Applicants:		
Average HS GPA:	3.1	3.1
Average SAT Math Score:	448	524

Table 4. Correct Averages

After comparing Tables 2, 3 and 4, it is clear that if we want to produce an accurate report averages have to be tabulated separately. However, when we are dealing with a lot of data such as applicants by year, departments, majors etc. we probably do not want to fill out the tables manually. So, we make use of the out files like those above, e.g., `out = GPA_2016` and `out = SAT_2016`.

Printing GPA_2016 gives us this output:

sex	_TYPE_	HSGPA_ _PAGE_	_TABLE_	Mean
Female	1	1	1	3.13000
Male	1	1	1	3.10391

Transposing GPA_2016 gives us this output:

NAME	_LABEL_	COL1	COL2
sex		1.00	2.00000
PAGE	Page for Observation	1.00	1.00000
TABLE	Table for Observation	1.00	1.00000
HSGPA_Mean		3.13	3.10391

Transpose of SAT_2016 produces this output:

NAME	_LABEL_	COL1	COL2
sex		1.000	2.000
PAGE	Page for Observation	1.000	1.000
TABLE	Table for Observation	1.000	1.000
SAT_MATH_Mean		447.778	524.390

To have all the relevant data in one set, we need to merge two transposed data sets:

```
data gpa_sat;
  length NAME_ $20.;
  set GPA_2016_T SAT_2016_T;
run;
```

The merged data set produces this output:

NAME	_LABEL_	COL1	COL2
sex		1.000	2.000
PAGE	Page for Observation	1.000	1.000
TABLE	Table for Observation	1.000	1.000
HSGPA_Mean		3.130	3.104
sex		1.000	2.000
PAGE	Page for Observation	1.000	1.000
TABLE	Table for Observation	1.000	1.000
SAT_MATH_Mean		447.778	524.390

All we need to do now is to modify this set by renaming some columns and keeping only the rows and columns we are interested in:

```
data gpa_sat (rename = (_NAME_ = Applicants COL1 = Male COL2 = Female));
  set gpa_sat;
  if _NAME_ not in ('HSGPA_Mean', 'SAT_MATH_Mean') then delete;
  drop _LABEL_ ;
run;
```

Now we have correct averages for both GPA and SAT in one output:

Applied	Male	Female
HSGPA_Mean	3.130	3.104
SAT_MATH_Mean	447.778	524.390

Following through the process for other variables and using ODS XML allow us to produce Table 1 as follows:

Applicants:	Female	Male
Average Age:	21	23
Average HS GPA:	3.1	3.1
Average SAT Math Score:	448	524
Average ACT Math Score:	21	21

Table 5. Final Table

CONCLUSION

PROC TABULATE out files can be used along with ODS XML output to produce tables with accurate data without the use of copy and paste option.

RECOMMENDED READING

- *SAS® For Dummies®*
- *Base SAS® 9.3 Procedures Guide, Second Edition*

CONTACT INFORMATION

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

Abdus Shahid
Western Oregon University
Phone: 503-838-9331
E-mail: shahida@wou.edu
Web: <https://www.wou.edu/institutionalresearch/>

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.