

Title: Many Means to a Mean

Author: Shannon Pileggi

Abstract: This paper is motivated by reflections on an exam question that students struggled with in an introductory SAS programming course: “Suppose we have a SAS data set with three test grades from ten students. How could you get the average score for each of the three tests? How could you get each student’s average test grade?” Students often confused the application of proc means and the means function in a data step for achieving their objectives. In this paper, we provide an overview of various methods to achieve either column-wise means or row-wise means, which include data step methods (with functions or cumulative sums) and procedures (means, tabulate, report, and sql). We will also discuss various characteristics of these methods, including the difficulty level of the syntax, how missing data is treated, and presentation of the output. This talk will be accessible to novice SAS users as well as provide more experienced users alternative methods for calculating a mean.

Working definitions:

Column-wise mean – takes the mean of a column of data (vertical data summary, over observations)

Row-wise mean – takes the mean of an observation (horizontal data summary, over columns)

Summary table:

This table is not finalized, but provides an overview of topics to be discussed in the paper.

Attribute	PROC MEANS	PROC TABULATE	PROC SQL with mean summary	DATA step utilizing cumulative sums	PROC REPORT (summary report)	DATA step utilizing mean function	PROC REPORT (detail report)
<i>Mean type</i>	Column	Column	Column	Column	Column	Row	Row
<i>Best option for beginners?</i>	x					x	
<i>Default number of decimals reported</i>	consistently 7	consistently 2	varies based on sig digits	8 in data set / 4 with the proc print	varies based on sig digits	8 in data set / 4 with the proc print	varies based on sig digits
<i>Missing data</i>	ignored	ignored	ignored	ignored	ignored	ignored	prohibits calculation
<i>Syntax level</i>	Easier	Harder	Harder	Harder	Harder	Easier	Harder

<i>Organization of output</i>	Clean	Clean	Harder to read	Harder to read	Clean	Clean	Clean
<i>Options to combine with a variety of statistics</i>	some	yes	yes	if you calculate them	yes	N/A	N/A
<i>Requires proc print to view results</i>	no	no	no	yes	no	yes	no
<i>Other notes</i>			Must name each statistic	be careful to exclude missing data in counts			
<i>Advantages</i>	Simplest to use, well organized output, many customization options	Well organized output, can produce a variety of statistics	Can produce a variety of statistics	Explicit control over your calculations	Can produce a variety of statistics	Simplest to use	can produce a variety of statistics
<i>Limitations</i>	Limited number/combo of statistics able to produce	More complex syntax	More complex syntax, must name each statistic, organization of means not as readable	Must be careful to have counts that exclude missing data; requires proc print to view results	defining the usage may be challenging for beginners	Requires proc print to view results	More complex syntax

Demonstration SAS code (not yet finalized):

```
data grades;
  input name $ test1 test2 test3;
  datalines;
  Shannon      96      82      83
  Lex          92      81      68
  Becky        92      75      73
  Lora         94      65      70
  Susan        91      77      85
  Hunter       76      72      86
  Ulric        98      71      80
  Richann      90      60      60
  Taylor       97      94     100
  Michael      .       77      60
  ;
run;

title "original data";
proc print;
run;
title;

/*-----
--
COLUMN MEANS
-----*/

title "proc means";
proc means data=grades;
  var test1 test2 test3;
run;
title;

title "proc sql";
proc sql;
  select count(test1) as n_test1,
         mean(test1) as ave_test1,
         count(test2) as n_test2,
         mean(test2) as ave_test2,
         count(test3) as n_test3,
         mean(test3) as ave_test3
  from grades;
quit;
title;

title "proc tabulate";
proc tabulate data=grades;
  var test1 test2 test3;
  table test1*(N mean) test2*(N mean) test3*(N mean);
run;
title;

title "proc report (summary)";
proc report data=grades;
  column test1,(N mean) test2,(N mean) test3,(N mean);
```

```

run;
title ;

*data step;
data grades2;
    set grades;
    if test1 ne . then count1+1;
    if test2 ne . then count2+1;
    if test3 ne . then count3+1;
    sum_test1 + test1;
    sum_test2 + test2;
    sum_test3 + test3;
    running_ave_test1=sum_test1/count1;
    running_ave_test2=sum_test2/count2;
    running_ave_test3=sum_test3/count3;
run;

title "data step results with cumulative sums";
proc print data=grades2;
run;
title;

/*-----
--
ROW MEANS
-----*/
* /
data grades2;
    set grades;
    ave_test = mean(test1,test2,test3);
run;

title "data step results with mean function";
proc print; run;
title;

*detail report;
title "proc report (detail)";
proc report data=grades;
    column name test1 test2 test3 ave_test;
    define ave_test / computed;
    compute ave_test;
        ave_test = ( _C2_ + _C3_ + _C4_ )/3;
    endcomp;
run;
title ;

```