

**PRELOADFMT comes to your rescue, it brings missing categories to life in summary reports**

Niraj J. Pandya, Independent Consultant, CT  
 Ramalaxmareddy Kandimalla, KRL Solutions, CT

**ABSTRACT**

In clinical trials data analysis and reporting, one always comes across producing categorical tables used for analysis. Provided with lots of different kind of data types such as labs, adverse events, physical examination, concomitant therapies etc and multiple categories for several categorical variables, it becomes difficult sometimes to impute 0 values for categories which are missing altogether for a particular variable. SAS provides multiple options to handle such cases and produce output with all the categories even if some don't exist in the data. This paper concentrates on different work around methods and will discuss details of implementing each technique.

**INTRODUCTION**

During Analysis and reporting in SAS, it is common practice to summarize all possible combinations of class level variables even if few categories are altogether missing in the database. As for instance, Case Report Form has an option as 'OTHER' in Race category but none of the subjects qualifies for it in current data release and thus race category 'OTHER' is missed in the database. But in the above scenario, report still needs to summarize 'OTHER' category under race with zero value assuming this category might appear in future data views. There are few techniques which we will discuss in this paper to summarize all the categories in the report and fill with zero those categories which are missing from the data.

We will use HDL Lab data as shown in Figure 1 for illustration purpose throughout this compilation. We want to create a descriptive summary report of this data categorized by sex and abnormality ranges. This HDL Lab data can be categorized into 'Low', 'Normal' or 'High' categories based on following criteria.

<b>HDL (Male)</b>	< 25 mg/dL	Low
	25 mg/dL - 65 mg/dL	Normal
	> 65 mg/dL	High
<b>HDL (Female)</b>	< 38 mg/dL	Low
	38 mg/dL – 94 mg/dL	Normal
	> 94 mg/dL	High

	USUBJID	SEX	VISIT	LBTEST	LBRSLT	LBUNITS
1	1001	1	Week2	HDL	46	mg/dL
2	1002	2	Week2	HDL	95	mg/dL
3	1003	1	Week2	HDL	94	mg/dL
4	1004	1	Week2	HDL	39	mg/dL
5	1005	2	Week2	HDL	33	mg/dL
6	1006	2	Week2	HDL	97	mg/dL
7	1007	2	Week2	HDL	31	mg/dL
8	1008	1	Week2	HDL	91	mg/dL
9	1009	1	Week2	HDL	42	mg/dL
10	1010	2	Week2	HDL	28	mg/dL

**FIGURE 1: HDL Lab Data**

Following user defined formats will also be used in all the examples explained in this paper.

```
Proc Format;
Value range 1='Low'
            2='Normal'
            3='High'
            ;
Value sex 1 = 'Male'
         2 = 'Female'
         ;
Run;
```

## METHOD 1: USE OF DUMMY DATASET

A simple method to achieve our purpose can be to create a dummy hard coded dataset with all the categories that we expect in the report. Get the output dataset by running any statistical procedure on original data and merge this output dataset with previously created dummy dataset. Following piece of code explains this in detail.

```
/* Hard coded dummy dataset in the format same as expected in final output */
Data Dummy(drop = i j);
Do i = 1 to 3;
  Do j = 1 to 2;
    SEX = j;
    RANGE = i;
    N = 0;
    MEAN = .;
    MEDIAN = .;
    STD = .;
    MIN = .;
    MAX = .;
    Output;
  End;
End;
Run;

/* Create a dataset containing summary statistics */
Proc Means Data = HDL noprint;
  Class SEX RANGE;
  Var LBRSLT;
  Output out = Stat(keep = SEX RANGE N Mean Median Std Min Max)
           N = N
           Mean = Mean
           Median = Median
           Std = Std
           Min = Min
           Max = Max;
Run;

/* Sort Stat and Dummy dataset by SEX and RANGE and Merge them together */
Proc Sort Data = Stat;
  By SEX RANGE;
  Where SEX > . and RANGE > .;
Run;

Proc Sort Data = Dummy;
  By SEX RANGE;
Run;
```

```

Data Stat;
  Merge Dummy Stat;
  By SEX RANGE;
  Format SEX sex. RANGE range.; /* Apply previously defined formats */
Run;

Proc Print noobs;
Run;

```

The above code will create the output as show in Figure 2.

SEX	RANGE	N	MEAN	MEDIAN	STD	MIN	MAX
Male	Low	0	.	.	.	.	.
Male	Normal	3	42.3333	42.0	3.51188	39	46
Male	High	2	92.5000	92.5	2.12132	91	94
Female	Low	3	30.6667	31.0	2.51661	28	33
Female	Normal	0	.	.	.	.	.
Female	High	2	96.0000	96.0	1.41421	95	97

FIGURE 2: Output from METHOD 1

## METHOD 2: PROC MEANS USING 'COMPLETETYPES' AND 'PRELOADFMT' OPTION

The COMPLETETYPES option available in PROC MEANS creates all possible combinations of class variables even if combination doesn't exist in the input dataset. If this option is used in conjunction with PRELOADFMT option in CLASS statement, PROC MEANS will write all user-defined format ranges or values for the combinations of class variables to the output, even when a frequency is zero.

Following code will demonstrate the use of these two options. We will use the same dataset from FIGURE 1 and previously created user defined formats RANGE and SEX for this illustration.

```

Proc Means Data = HDL noprint completetypes;
  Format SEX sex. RANGE range.;
  Class SEX RANGE/preloadfmt;
  Var LBRSLT;
  Output out = Stat(keep = SEX RANGE N Mean Median Std Min Max)
                N = N
                Mean = Mean
                Median = Median
                Std = Std
                Min = Min
                Max = Max ;
Run;

Proc Print noobs;
  Where SEX > . and RANGE > .;
Run;

```

The above code will create the output as show in Figure 3

SEX	RANGE	N	MEAN	MEDIAN	STD	MIN	MAX
Male	Low	0	.	.	.	.	.
Male	Normal	3	42.3333	42.0	3.51188	39	46
Male	High	2	92.5000	92.5	2.12132	91	94
Female	Low	3	30.6667	31.0	2.51661	28	33
Female	Normal	0	.	.	.	.	.
Female	High	2	96.0000	96.0	1.41421	95	97

FIGURE 3: Output from METHOD 2

It is important to note at this point that above code without PRELOADFMT option will work same as SPARSE option in Proc Freq.

### METHOD 3: PROC TABULATE USING 'PRINTMISS' AND 'PRELOADFMT' OPTION

The PRINTMISS option in TABLE statement of PROC TABULATE enables you to display all the possible combinations of CLASS variable values that exist in the data set. Thus, if the CLASS variable value is present in one crossing but not another, all you may need to add is the PRINTMISS option to the TABLE statement. Use of PRELOADFMT option in conjunction with PRINTMISS option works the same way as explained above in METHOD 2.

Following piece of code explains the use of these two options in Proc Tabulate.

```
Proc Tabulate Data = HDL;
  Format SEX sex. RANGE range.;
  Class SEX RANGE/preloadfmt;
  Var LBRSLT;
  Table SEX='SEX'*RANGE='RANGE',
        LBRSLT=' '*N='N'
        LBRSLT=' '*MEAN='MEAN'
        LBRSLT=' '*MEDIAN='MEDIAN'
        LBRSLT=' '*STD='STD'
        LBRSLT=' '*MIN='MIN'
        LBRSLT=' '*MAX='MAX' / printmiss;
Run;
```

The above code will create the output as show in Figure 4

SEX	RANGE	N	MEAN	MEDIAN	STD	MIN	MAX
Male	Low	0.00	.	.	.	.	.
	Normal	3.00	42.33	42.00	3.51	39.00	46.00
	High	2.00	92.50	92.50	2.12	91.00	94.00
Female	Low	3.00	30.67	31.00	2.52	28.00	33.00
	Normal	0.00	.	.	.	.	.
	High	2.00	96.00	96.00	1.41	95.00	97.00

FIGURE 4: Output from METHOD 3

### METHOD 4: PROC REPORT USING 'COMPLETEROWS' AND 'PRELOADFMT' OPTION

Using PRELOADFMT option in conjunction with COMPLETEROWS option in Proc Report outputs all the combinations of category variables even if they are absent in the dataset. Following code demonstrates how to calculate summary statistics using these two options in Proc Report.

Since Proc Report will not allow calculating and displaying different summary statistics for same variable (LBRSLT in our case) in the output, we will create dummy variables (one for each required statistic) first in the input dataset.

```

/* Create Dummy variables for each expected statistics */

Data HDL1;
  Set HDL;
  N=LBRSLT;
  MEAN=LBRSLT;
  MEDIAN=LBRSLT;
  STD=LBRSLT;
  MIN=LBRSLT;
  MAX=LBRSLT;

Run;

Proc Report Data=hdl1 completerows nowd BOX;
  Format SEX sex. RANGE range.;

  Column SEX RANGE N MEAN MEDIAN STD MIN MAX;

  Define SEX / order = internal group preloadfmt;
  Define RANGE / order = internal 'RANGE' group preloadfmt;
  Define N / analysis n;
  Define MEAN / analysis mean;
  Define MEDIAN / analysis median;
  Define STD / analysis std;
  Define MIN / analysis min;
  Define MAX / analysis max;

Run;

```

The above code will create the output as show in Figure 5

SEX	RANGE	N	MEAN	MEDIAN	STD	MIN	MAX
Male	Low	0	.	.	.	.	.
	Normal	3	42.333333	42	3.5118846	39	46
	High	2	92.5	92.5	2.1213203	91	94
Female	Low	3	30.666667	31	2.5166115	28	33
	Normal	0	.	.	.	.	.
	High	2	96	96	1.4142136	95	97

FIGURE 5: Output from METHOD 4

## CONCLUSION

When all the expected data categories are not available in the data, but still need to be presented in the categorical summary reports, PRELOADFMT option comes very handy. To generate custom reports that also include all missing categorical combinations, usually programmer has to go through lot of data manipulation and many times programmer has to do some hard coding. Use of PRELOADFMT option in conjunction with other procedure specific options like COMPLETETYPES, COMPLETEROWS, and PRINTMISS reduces the hassle of data manipulations and it also takes care of user defined format for the categories in the report. This reduces programming efforts significantly and also keeps the entire reporting macro free of unnecessary steps that will come up otherwise.

## **ACKNOWLEDGEMENTS**

We would like to acknowledge Sapan Raval for his help in putting this paper together.

## **CONTACT INFORMATION**

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

Name: NIRAJ J PANDYA  
Phone: 201-936-5826  
E-mail: [npandya@elementtechnologies.com](mailto:npandya@elementtechnologies.com)

Name: RAMALAXMAREDDY KANDIMALLA  
Phone: 408-207-2293  
E-mail: [krlaxmareddy@gmail.com](mailto:krlaxmareddy@gmail.com)

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.