Paper CC106

29 Shades of Missing

Darryl Putnam, Pinnacle Solutions, LLC

ABSTRACT

Missing values can have many flavors of missingness in your data and understanding these flavors of missingness can shed light on your data and analysis. SAS® can identify 29 flavors of missing data, and a variety of functions, statements, procedures, and options can be used to whip your missing data into submission. This paper will focus solely on how SAS can use missing values in unique and insightful ways.

INTRODUCTION

In traditional Relational Database Management Systems (RDMS), NULL or missing values are common place. The NULL values are represented by the NULL Set as an off-shoot of the set theory that drives RDMS and the SQL syntax. Although SAS tables have the same look and feel as a database table the underlying constructs are different. The current SQL standard defines NULL as "unknown", even E.F. Cobb the founder of the relational database, notes that the NULL should include the reason for the value being missing. He suggested parsing NULLs in to 2 shades of missing with A-Values(Missing But Applicable) and I-Values(Missing But Inapplicable).

SAS upped to ante and has 3 types of missing values

1. Character
2. Numeric
3. Special Numeric

Expanding on NULLs to include a reason for the NULL value is a good step forward in our quest to wring the meaning out of missing values.

A PRIMER ON MISSING VALUES

Missing values in one's data can wreak havoc on analysis if we are not careful. What happens when we add them, sort them, group by them, print them, or export to Excel? First we need to create a test data set, the below code is used to create our test data set used throughout the remainder of the paper.

In order to assign a missing value for character data types, a blank wrapped in either single or doubles is used (' ', " "). For numeric data types a period ('.') is used to assign a data type. The data contains purposely invalid data that we will use to explore in later sections.

Example 1: Creating a Data Set with Missing Values

```
proc sql;
create table basic_demographics
(ID num,
 ZIPCODE num FORMAT=Z5.,
 MALE_FEMALE char(1),
 AGE num,
 INCOME num FORMAT=DOLLAR16.
 );

 insert into basic_demographics
 values(1,28277,'M' ,48, 45000)
 values(2,28277,''  ,30, -145000)
 values(3,28277,'M' ,200,.)
 values(4,28277,'X' ,48, 15000)
 values(5,00001,'F' ,28, 55000)
 ;
 quit;

proc print data= basic_demographics;
run;
```

Error! Reference source not found.Error! Reference source not found. Result of PROC PRINT

Obs	ID	ZIPCODE	MALE_FEMALE	AGE	INCOME
1	1	28277M		48	$45,000
2	2	28277		30	$-145,000
3	3	28277M		200	.
4	4	28277X		48	$15,000
5	5	00001F		28	$55,000

The table has a number of data issues

- Invalid ZIPCODE
- Missing Male Female Flag
- Invalid Male Female Flag
- Invalid Age
- Missing Income
- Invalid Income

SPECIAL MISSING NUMERIC

SAS can code numeric missing values in 28 different ways. Missing character can only be represented by a blank. Basic numeric missing data is coded as a period ('.') and "special missing value is a type of numeric missing value that enables you to represent different categories of missing data by using the letters A-Z or an underscore." (SAS Online Documentation)

Continuing from the SAS Online Documentation

> "SAS accepts either uppercase or lowercase letters. Values are displayed and printed as uppercase.
>
> If you do not begin a special numeric missing value with a period, SAS identifies it as a variable name. Therefore, to use a special numeric missing value in a SAS expression or assignment statement, you must begin the value with a period, followed by the letter or underscore, as in the following example:
>
> x=.d;
>
> When SAS prints a special missing value, it prints only the letter or underscore."

Let us use this knowledge of special missing numerics to our advantage while we recode the test data set. Invalid data needs to be recoded to missing but information needs to added to explain why the data is missing. This is where the special numeric missing values come into play. The recoding scheme will be as follows:

- I = Invalid
- N=Invalid Negative Value
- . = Value was Missing on Input

In addition special missing values, SAS has many functions that are associated with missing values. The MISSING function is a Boolean function that lets you know whether a value is missing or not and can be used with either character or numeric data types.

Example 2: Recoding Values to Special Numeric Missing

```
data editted_basic_demographics;
  set basic_demographics;
  if ZIPCODE=1 then ZIPCODE=.I;
  if AGE=200 then AGE=.I;
  if NOT MISSING(INCOME) AND INCOME<0 THEN INCOME=.N;
run;
proc print data=editted_basic_demographics;
run;
```

Figure 2: Display of Recoded Values

Obs	ID	ZIPCODE	MALE_FEMALE	AGE	INCOME
1	1	28277M		48	$45,000
2	2	28277		30	N
3	3	28277M		I	.
4	4	28277X		48	$15,000
5	5	IF		28	$55,000

By using a combination of special missing values and PROC FORMAT, we can maximize the information for the end-user. The below code adds labels to the special missing values in order to add the reason for the missing value.

Example 3: Using PROC FORMAT with Special Missing Values

```
proc format;
value NUMISS
.N='Negative Value'
.I='Invalid Value'
. ='Missing from Source';
run;

proc print data=editted_basic_demographics;
format age income numiss.;
run;
```

Figure 3: Display of Recoded Values

Obs	ID	ZIPCODE	MALE_FEMALE	AGE	INCOME
1	1	28277M		48	45000
2	2	28277		30	Negative Value
3	3	28277M		Invalid Value	Missing from Source
4	4	28277X		48	15000
5	5	IF		28	55000

GROUP BY MISSING

Missing data can be summarized by the special missing data and will not be grouped together in a global missing. However, not all procedures group missing values the same way. Let us compare PROC SQL and PROC MEANS;

Example 4: PROC SQL and Missing Class Variables

```
proc sql;
  select Income, count(*) as NROWS
  from editted_basic_demographics
  group by Income
  order by income;
  ;
quit;
```

Figure 4: Output of PROC SQL

INCOME	NROWS
.	1
N	1
$15,000	1

INCOME	NROWS
$45,000	1
$55,000	1

Example 5: PROC MEANS and Missing Class Variable

```
proc means data=editted_basic_demographics n ;
class income;
var income;
run;
```

Figure 5: Notice that the missing values are not displayed.

Analysis Variable : INCOME		
INCOME	N Obs	N
$15000	1	1
$45000	1	1
$55000	1	1

With PROC MEANS one must add the "MISSING" option in order to group by the CLASS variable. According to the documentation; the MISSING option considers missing values as valid values to create the combinations of class variables. Special missing values that represent numeric values (the letters A through Z and the underscore (_) character) are each considered as a separate value.

Example 6: PROC MEANS and Missing Class Variable Plus the Missing Option.

```
proc means data=editted_basic_demographics n missing;
class income;
var income;
run;
```

Figure 6: Now the Missing Data is Exposed

Analysis Variable : INCOME		
INCOME	N Obs	N
.	1	0
N	1	0
$15000	1	1
$45000	1	1
$55000	1	1

CONCLUSION

Missing data can be exploited in your data to add much value in your analysis. Not being limited to missing or not, SAS has 29 shades of missing and has the special numeric missing type to put context behind missing values. Besides the special missing numeric values, SAS has a plethora of functions that can be used to assign missing values and search for missing values. By exploring the concepts in this paper the reader can whip missing data into submission.

REFERENCES

Codd, E.F. 1990. "The Relational Model for Database Management (Version 2 ed.). "

Humphreys, Suzanne M. 2006. "MISSING! – Understanding and Making the Most of Missing Data". *Proceedings of SUGI 31. http://www2.sas.com/proceedings/sugi31/toc.html*

Meneglbier, Magnus. 2011. “Missing Values in SAS”. *Proceedings PhUSE 2012. http://www.phusewiki.org/docs/2011%20Papers/CC05%20paper.*

SAS OnlineDoc® 9.4. Available at http://support.sas.com/documentation/94/index.html

CONTACT INFORMATION

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

Name: Darryl Putnam
Enterprise: Pinnacle Solutions, LLC
E-mail: dputnam@pinnacledatasolutions.net