

DARE TO COMPARE – THREE METHODS OF PREPARING YOUR DATA TO COMPARE NUMERIC VALUES

David Maddox, Bellsouth Telecommunications, Inc.

ABSTRACT

In the course of performing basic data analyses, SAS programmers often find it necessary to calculate differences between numeric variables that exist in different data sets. There are various methods in SAS of preparing your data to facilitate comparisons of numeric values. This paper briefly explores three methods of processing, including data set concatenation, match-merging, and the often overlooked, but powerful, PROC COMPARE.

INTRODUCTION

One of the simplest and most basic analysis techniques is the direct comparison of numeric values. Data for a given measure often vary over time and place. While analysts love to have highly refined time series or perfectly compatible cross-sectional data, most are happy with two good values with which they can calculate the difference or percent change. SAS offers a number of methods that can be used to prepare data for analysis. This paper will briefly explore three of those methods – data set concatenation, match-merge, and PROC COMPARE.

THE TASK

The task in this paper is to compare the numeric information that has been collected and captured in two data sets representing different time periods – a typical data scenario. The two sets include a product code, and numeric measures of quantity and revenue. It's assumed that in this case that data for all products appear in both data sets. The data sets are as follows:

YEAR2004				YEAR2005			
OBS	PRODUCT	QUANTITY	REVENUE	OBS	PRODUCT	QUANTITY	REVENUE
1	XL5	21	525.00	1	XL5	19	475.00
2	3Z1	45	459.00	2	3Z1	48	489.60
3	5S7	61	6893.00	3	5S7	71	8023.00
4	7S5	32	556.80	4	7S5	30	582.00
5	4X3	14	1750.00	5	4X3	27	3375.00
6	LX4	44	946.00	6	LX4	48	1032.00
7	7X9	67	904.50	7	7X9	62	837.00
8	HZ1	28	1764.00	8	HZ1	31	2108.00
9	MN4	19	836.00	9	MN4	18	792.00
10	PNR	7	1470.00	10	PNR	9	1890.00

DATA SET CONCATENATION

The first method involves combining the data sets into one data set. This can be done in a DATA STEP by listing the data sets in a SET statement or in a PROC such as PROC APPEND. However, we first must associate each observation with its respective time period or place. In our example, we can add a YEAR variable and assign the pertinent value for each observation.

```
Data year2004_new;
Set year2004;
Year='2004';
Run;
```

```
Data year2005_new;
Set year2005;
Year='2005';
Run;
```

```
Data combined;
Set year2004_new year2005_new;
Run;
```

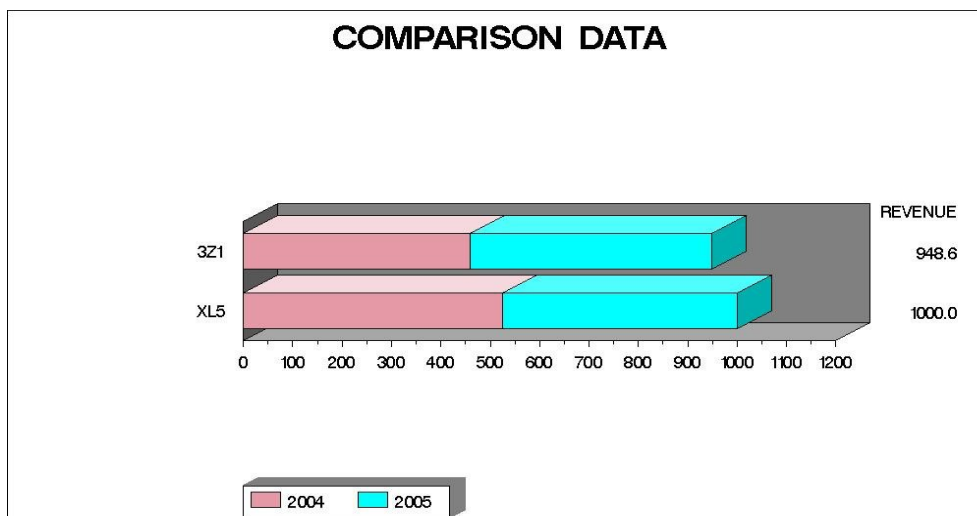
This code produces the following combined data set.

COMBINED

OBS	PRODUCT	QUANTITY	REVENUE	YEAR
1	XL5	21	525.00	2004
2	3Z1	45	459.00	2004
3	5S7	61	6893.00	2004
4	7S5	32	556.80	2004
5	4X3	14	1750.00	2004
6	LX4	44	946.00	2004
7	7X9	67	904.50	2004
8	HZ1	28	1764.00	2004
9	MN4	19	836.00	2004
10	PNR	7	1470.00	2004
11	XL5	19	475.00	2005
12	3Z1	48	489.60	2005
13	5S7	71	8023.00	2005
14	7S5	30	582.00	2005
15	4X3	27	3375.00	2005
16	LX4	48	1032.00	2005
17	7X9	62	837.00	2005
18	HZ1	31	2108.00	2005
19	MN4	18	792.00	2005
20	PNR	9	1890.00	2005

Keep in mind that certain rules must be followed when data sets are combined. Variables must be of the same type. Otherwise, SAS gets totally confused as to which type attributes to carry into the new combined set. Also, the length of variables is governed by which data set is first in the SET list. In general, this rule carries over to labels, formats, and informats. So, the rule here is to be aware of how the variables from the different sets relate and what the attributes of the combined set will be.

Once joined in a unified data set, the data can easily be input into a number of procedures for further display and analysis. For example, this data set can be used to produce graphs such as the one below by specifying YEAR as the subgroup variable in PROC GCHART. Many other analyses can be done on this data without having to make further changes to the set.



MATCH-MERGE (OR PROC SQL)

The second method of preparing data to facilitate numerical comparisons is to merge the sets together so that each observation includes the comparable measures. Merging generally requires more coding and sometimes involves handling large amounts of data in each observation. As with data set concatenation, there are some rules pertaining to merges that analysts must be aware of. PROC SQL provides a more elegant method of merging datasets, but the analyst must still understand how the data relates. The general workflow in merging data sets is to sort the data by the variables by which the observations will be merged. In data set merges, it is imperative that all analysis variables be renamed in order to prevent one variable over-writing another when the merge is performed. Then, the data sets can be merged to produce a single set. In terms of our example, the sets will be merged by product. We're interested in comparing the quantity and revenue produced in the first time period with comparable measures in the second set. The following is a typical merge code sequence:

```
Proc sort data=year2004 out=year2004_srt(rename=(quantity=quantity_2004
revenue=revenue_2004));
    by product_code;
Run;
```

```
Proc sort data=year2005 out=year2005_srt(rename=(quantity=quantity_2005
revenue=revenue_2005));
    By product_code;
Run;
```

```
Data merged_set;
Merge year2004_srt year2005_srt;
    By product_code;
    Drop price;
Run;
```

```
Proc print data=merged_set;
Run;
```

The resultant data set is included here:

Obs	PRODUCT_CODE	quantity_2004	revenue_2004	quantity_2005	revenue_2005
1	3Z1	45	459.0	48	489.6
2	4X3	14	1750.0	27	3375.0
3	5S7	61	6893.0	71	8023.0
4	7S5	32	556.8	30	582.0
5	7X9	67	904.5	62	837.0
6	HZ1	28	1764.0	31	2108.0
7	LX4	44	946.0	48	1032.0
8	MN4	19	836.0	18	792.0
9	PNR	7	1470.0	9	1890.0
10	XL5	21	525.0	19	475.0

Direct comparisons are now greatly simplified. Any number of facilities may be used to calculate or visualize the differences in the data.

PROC COMPARE

The third method of preparing data for numerical comparisons actually requires no pre-processing at all. The data sets may be input and compared in the Base SAS PROC COMPARE for a quick look at numerical comparisons. The PROC offers numerous options for analysis and display. You will note that the PROC has a number of defaults so some of these defaults may need to be overridden in order to obtain a specific result. For example, the PROC prints only a certain number of rows in its basic detailed comparison report. It may be necessary to request that the entire detail comparison be printed. The following is an example of using PROC COMPARE to calculate the differences in quantity and revenue.

```
Proc compare data=year2004 compare=year2005;  
Id product_code;  
Run;
```

The output is as follows:

The COMPARE Procedure				
Comparison of DS.YEAR2004 with DS.YEAR2005				
(Method=EXACT)				
Data Set Summary				
Dataset	Created	Modified	NVar	NObs
DS.YEAR2004	12JUN06:23:00:39	12JUN06:23:00:39	4	10
DS.YEAR2005	12JUN06:23:00:39	12JUN06:23:00:39	4	10
Variables Summary				
Number of Variables in Common: 4.				
Number of ID Variables: 1.				

Observation Summary			
Observation	Base	Compare	ID
First Obs	1	1	PRODUCT_CODE=XL5
First Unequal	1	1	PRODUCT_CODE=XL5
Last Unequal	10	10	PRODUCT_CODE=PNR
Last Obs	10	10	PRODUCT_CODE=PNR
Number of Observations in Common: 10.			
Total Number of Observations Read from DS.YEAR2004: 10.			
Total Number of Observations Read from DS.YEAR2005: 10.			
Number of Observations with Some Compared Variables Unequal: 10.			
Number of Observations with All Compared Variables Equal: 0.			
Values Comparison Summary			
Number of Variables Compared with All Observations Equal: 0.			
Number of Variables Compared with Some Observations Unequal: 3.			
Total Number of Values which Compare Unequal: 23.			
Maximum Difference: 1625.			

PROC COMPARE OUTPUT

The COMPARE Procedure
Comparison of DS.YEAR2004 with DS.YEAR2005
(Method=EXACT)

All Variables Compared have Unequal Values

Variable	Type	Len	Label	Ndif	MaxDif
QUANTITY	NUM	8	QUANTITY	10	13.000
PRICE	NUM	8	PRICE	3	5.000
REVENUE	NUM	8	REVENUE	10	1625

Value Comparison Results for Variables

PRODUCT_CODE	QUANTITY		Diff.	% Diff
	Base QUANTITY	Compare QUANTITY		
XL5	21.0000	19.0000	-2.0000	-9.5238
3Z1	45.0000	48.0000	3.0000	6.6667
5S7	61.0000	71.0000	10.0000	16.3934
7S5	32.0000	30.0000	-2.0000	-6.2500
4X3	14.0000	27.0000	13.0000	92.8571
LX4	44.0000	48.0000	4.0000	9.0909
7X9	67.0000	62.0000	-5.0000	-7.4627
HZ1	28.0000	31.0000	3.0000	10.7143
MN4	19.0000	18.0000	-1.0000	-5.2632
PNR	7.0000	9.0000	2.0000	28.5714

PRODUCT_CODE	PRICE		Diff.	% Diff
	Base PRICE	Compare PRICE		
3Z1	10.2000	10.2000	-1.78E-15	-1.74E-14
7S5	17.4000	19.4000	2.0000	11.4943
HZ1	63.0000	68.0000	5.0000	7.9365

PROC COMPARE OUTPUT

The COMPARE Procedure
Comparison of DS.YEAR2004 with DS.YEAR2005
(Method=EXACT)

Value Comparison Results for Variables

PRODUCT_CODE	REVENUE		Diff.	% Diff
	Base REVENUE	Compare REVENUE		
XL5	525.0000	475.0000	-50.0000	-9.5238
3Z1	459.0000	489.6000	30.6000	6.6667
5S7	6893	8023	1130	16.3934
7S5	556.8000	582.0000	25.2000	4.5259
4X3	1750	3375	1625	92.8571
LX4	946.0000	1032	86.0000	9.0909
7X9	904.5000	837.0000	-67.5000	-7.4627
HZ1	1764	2108	344.0000	19.5011
MN4	836.0000	792.0000	-44.0000	-5.2632
PNR	1470	1890	420.0000	28.5714

You will notice that the PROC provided a large amount of information about the input data sets and the differences between the numerical values. While we've basically taken the default output here, there are a large number of options that can be invoked which will change the appearance of the output. As with most PROCs, data sets can be output for further analytical activities.

CONCLUSION

The methods discussed in this paper are nothing new or revolutionary. SAS programmers have been using variations of these methods for many years. However, it's significant to note that a key attribute of SAS programming is that there are often many different ways to attack the same problem. The richness of the toolset allows that to be the case. As with most endeavors, the preparation methodology will most certainly be influenced by the analyst's objective.

REFERENCES

SAS OnlineDoc® Version Eight. Copyright © 1999. SAS Institute, Inc. Cary, NC, USA.
SAS System Help for Version 9.1.3. Copyright SAS Institute, Inc., Cary, NC, USA.

RECOMMENDED READING

There's a very good paper on PROC COMPARE entitled "How Does Your Data Compare?" SAS's PROC COMPARE PROCEDURE by Jenna Heyen. It can be found at <http://www2.sas.com/proceedings/sugi22/CODERS/PAPER96.PDF>.

CONTACT INFORMATION

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

David Maddox
Bellsouth Telecommunications, Inc.
South E3E, 3535 Colonnade Parkway
Birmingham, AL 35243
Work Phone: 205-977-0415
Fax: 205-977-8719
Email: David.Maddox@Bellsouth.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.