Don’t Let PROC COMPARE Catch You Unaware

Joshua Horstman
Nested Loop Consulting

Roger Muller
Data-to-Events.com

Introduction

- PROC COMPARE – procedure to compare the contents of two SAS data sets

- Some common uses:
  - Compare data set against prior version to confirm changes are as expected
  - Validate a new program produces the same results as a legacy system
  - Verify results of “double programming” match

The Holy Grail

The following output is often treated as an unqualified indicator of matching data sets:

NOTE: No unequal values were found. All values compared are exactly equal.

Gotcha #1: Missing Variables

DM_PROD Dataset		
SUBJID	AGE	SEX
101	45	M
102	37	F
103	61	F

DM_QC Dataset	
SUBJID	AGE
101	45
102	37
103	61

```
proc compare base=dm_prod compare=dm_qc;
run;
```

PROC COMPARE Output #1

```
                    The COMPARE Procedure
          Comparison of WORK.DM_PROD with WORK.DM_QC
                        (Method=EXACT)

                        Data Set Summary

Dataset                  Created           Modified  NVar    NObs

WORK.DM_PROD   30MAR13:17:23:18  30MAR13:17:23:18     3       3
WORK.DM_QC     30MAR13:17:23:18  30MAR13:17:23:18     2       3


                       Variables Summary

Number of Variables in Common: 2.
Number of Variables in WORK.DM_PROD but not in WORK.DM_QC: 1.


                      Observation Summary

                Observation      Ba

                First Obs
                Last  Obs

          Observations in Common: 3.
          er of Observations Read from WORK.DM_PROD: 3.
          er of Observations Read from WORK.DM_QC: 3.

          f Observations with Some Compared Variables Unequal: 0.
Number of Observations with All Compared Variables Equal: 3.

NOTE: No unequal values were found. All values compared are exactly equal.
```

No unequal values were found!!

We're DONE!!!!

Or are we???

Gotcha #2: Missing Observations

VS_PROD Dataset			
SUBJID	**VISITNUM**	**SYSBP**	**DIABP**
101	1	120	80
101	2	126	84
102	1	132	90
102	2	131	85

VS_QC Dataset			
SUBJID	**VISITNUM**	**SYSBP**	**DIABP**
101	1	120	80
101	2	126	84
102	1	132	90

```
proc compare base=vs_prod compare=vs_qc;
run;
```

PROC COMPARE Output #2

```
Procedure
                Comparison of WORK.VS_PROD with WORK.VS_QC
                              (Method=EXACT)

                            Data Set Summary

Dataset                 Created           Modified  NVar    NObs

WORK.VS_PROD  30MAR13:17:23:18  30MAR13:17:23:18     4       4
WORK.VS_QC    30MAR13:17:23:18  30MAR13:17:23:18     4       3


                           Variables Summary

Number of Variables

                          Observations

Observation      Base  Compare

First Obs           1        1
Last  Match         3        3
Last  Obs           4        .

Number of Observations in Common: 3.
Number of Observations in WORK.VS_PROD but not in WORK.VS_QC: 1.
Total Number of Observations Read from WORK.VS_PROD: 4.
Total Number of Observations Read from WORK.VS_QC: 3.

Number of Observations with Some Compared Variables Unequal: 0.
Number of Observations with All Compared Variables Equal: 3.

NOTE: No unequal values were found. All values compared are exactly equal.
```

No unequal values were found!!

We're DONE!!!!

Not so fast!!

Gotcha #3: Conflicting Types

LB_PROD Dataset			
SUBJID	VISITNUM	LBTESTCD	LBORRES
101	1	ALB	3.6
101	1	ALP	47.2
101	1	AST	13.5
101	1	BILI	0.8

LB_QC Dataset			
SUBJID	VISITNUM	LBTESTCD	LBORRES
101	1	ALB	13.6
101	1	ALP	57.2
101	1	AST	23.5
101	1	BILI	10.8

```
proc compare base=lb_prod compare=lb_qc;
run;
```

PROC COMPARE Output #3

```
                    The COMPARE Procedure
          Comparison of WORK.LB_PROD with WORK.LB_QC
                        (Method=EXACT)

                        Data Set Summary

Dataset                 Created          Modified  NVar     NObs

WORK.LB_PROD  30MAR13:18:57:21  30MAR13:18:57:21     4        4
WORK.LB_QC    30MAR13:18:57:21  30MAR13:18:57:21     4        4


                        Variables Summary

  Number of Variables in Common: 4.
  Number of Variables with Conflicting Types: 1.


 Listing of Common Variables with Conflicting Types

       Variable  Dataset       Type  Length

       lborres   WORK.LB_PROD  Num        8
                 WORK.LB_QC    Char      12


                      Observation Summary

          Observation      Base  Compare

          First Obs           1        1
          Last  Obs           4        4

r of Observations in Common: 4.
 Number of Observations Read from WORK.LB_PROD: 4.
 Number of Observations Read from WORK.LB_QC: 4.

Number of Observations with Some Compared Variables Unequal: 0.
Number of Observations with All Compared Variables Equal: 4.

NOTE: No unequal values were found. All values compared are exactly equal.
```

N's match

Gotcha!

No unequal values were found... yeah, right!

Can we fix this?

Possible Solution #1:

LISTALL option – lists all variables and observations only found in one data set.

Won't remove the "no unequal values" note and still requires we read the entire output.

Can we fix this?

Possible Solution #2:

ID statement – lists variables to use to match observations

Won't remove the "no unequal values" note.
Caution: PROC COMPARE will warn if multiple rows have same ID values,
but we introduce a new gotcha…

Gotcha #4: Mismatched ID Variables

EX_PROD Dataset		
SUBJID	VISITNUM	DOSE
101	1	3
101	2	4
102	1	5
102	2	6

EX_QC Dataset		
SUBJID	VISITNUM	DOSE
101	1	3
101	2	4
102	1	5
102	3	7

```
proc compare base=ex_prod compare=ex_qc;
     id subjid visitnum;
run;
```

PROC COMPARE Output #4

```
                    The COMPARE Procedure
          Comparison of WORK.EX_PROD with WORK.EX_QC
                        (Method=EXACT)

                        Data Set Summary

Dataset                 Created          Modified  NVar     NObs

WORK.EX_PROD  17JUN13:08:55:29  17JUN13:08:55:29     3        4
WORK.EX_QC    17JUN13:08:55:29  17JUN13:08:55:29     3        4


                        Variables Summary

                  Number of Variables in Common: 3.
                  Number of ID Variables: 2.


                       Observation Summary

 Observation      Base  Compare  ID

 First Obs           1        1  subjid=101 visitnum=1
 Last  Match         3        3  subjid=102 visitnum=1
 Last  Obs           4        .  subjid=102 visitnum=2
                     .        4  subjid=102 visitnum=3

Number of Observations in Common: 3.
Number of Observations in WORK.EX_PROD but not in WORK.EX_QC: 1.
Number of Observations in WORK.EX_QC but not in WORK.EX_PROD: 1.
Total Number of Observations Read from WORK.EX_PROD: 4.
Total Number of Observations Read from WORK.EX_QC: 4.

Number of Observations with Some Compared Variables Unequal: 0.
Number of Observations with All Compared Variables Equal: 3.

NOTE: No unequal values were found. All values compared are exactly equal.
```

N's match

Ah ha!

Only when ID statement is used.

Recommendations

- Know your data.
- Review the entire PROC COMPARE output.
- Remove extraneous/temporary variables before the comparison to facilitate identification of variable mismatches.

Contact Information

Joshua M. Horstman
Nested Loop Consulting
317-815-5899
josh@nestedloopconsulting.com

Roger D. Muller
Data-to-Events.com
317-846-5782
roger.muller@data-to-events.com