PharmaSUG 2014 - Paper BB16

Automated Validation – Really?

Sneha Sarmukadam, inVentiv Health Clinical, Pune, India
Seeja Shetty, inVentiv Health Clinical, Mumbai, India

ABSTRACT

In Clinical Programming, data quality takes priority over all other aspects. Since, it is health-related data that is being processed, it becomes crucial that accurate analysis is performed and the results are displayed exactly the way they are in the analysis datasets. There are two ways to validate outputs - Manual method and Automated method. It has been observed that Automated Validation has superseded Manual Validation lately due to many factors such as efficiency, 100% data check, code reusability, etc. However, even if Automated Validation has successfully checked the data, it does not mean that the output generated is perfect. Automated Validation methods are known to have some limitations [1], but there are some other underlying issues which may not be spotted easily and can be missed out. This paper tries to highlight such errors and oversights that occur during validations at data display level which can lead to serious quality issues.

INTRODUCTION

All SAS programmers have been validation programmers at some point in their SAS career. Generally, all validation programmers choose Automated Validation for error free and quick results; especially, when there are large numbers of outputs to be validated. We either check for the "NO UNEQUAL OBSERVATIONS FOUND" message in the PROC COMPARE output, or we check for "0" observations in the "OUT=" dataset generated by PROC COMPARE [2]. Automated validation has many known benefits over manual validation. But, there are some loop holes at data display level which may not be caught even after using PROC COMPARE. Such few examples have been listed below. All efforts should be made to identify and correct such oversights instead of solely depending on PROC COMPARE to do the job right.

INCORRECT FORMAT

This is a scenario where an incorrect format with shuffled values is used for a variable which is to be displayed in an output generated by PROC REPORT.

Example:

In the below Adverse Events listing, a format to display 'Yes' and 'No' is applied to last two columns. The values present in the dataset are in a numeric form – '0' and '1' and the aforementioned format is applied to these values. 'Yes' implies that adverse events are serious and related to the study drug. In reality, the values assigned to the numerals '0' and '1' were exactly opposite from what is being displayed below, which means that the format applied was an incorrect one.

Subject	Date/Time	Preferred Term/ Body system	Severity	Related?	Serious?
001	29APR2013/ 15:20	Chronic obstructive pulmonary disease /Respiratory, thoracic and mediastinal disorders	Severe	Yes	Yes
002	03MAY2011/ 10:10	Cardiac failure /Cardiac disorders	Moderate	No	Yes

Display 1. Example of Incorrect Format

The above finding may not be identified by the validation programmer since PROC COMPARE would not have shown any mismatch. Format is a useful tool which gives meaning to the underlying numbers – of course only if it is used in the right manner!

In this case, if special focus has to be given to understand the meaning of data and question it, only then such potential mistakes can be avoided.

NO FORMAT

It may happen that format is not applied on the required variable to be displayed in an output generated by PROC REPORT which will give unwanted results.

Example:

In the below ECG listing, the highlighted ECG 'Position' column does not have any format applied to it. If not checked carefully, since one may see many numbers together - (time point, date, time etc), it is easy to miss the fact that no format has been applied to the variable. Such a mistake can be avoided during validation if data is looked at more closely. In this case as well, the Automated Validation would not show any mismatches as the numbers would match perfectly well in original and validation datasets.

Subject	Visit	Time Point	Date	Time	Position	Abnormality
001	Screening	0	20JAN2013	19:00	1	No
	Baseline	0	25JAN2013	20:06	1	No
	TRT Day 1	2	31JAN2013	08:15	2	No

Display 2. Example of No Format

ISSUE WITH DATE FORMAT

Sometimes, improper date with seemingly correct format may be displayed in the output which may not be noticed during validation. This may happen in the following scenario:
If we have initialized date of birth variable with a format of DATETIME20. and we try to enter only the datepart in this date variable, then outcome is all dates of birth in the affected dataset would contain values like 01JAN1960!

Example:

The demographic listing below looks to be well formatted at first glance. When looked at closely, we realize that all the birth dates are 01JAN1960. Definitely everyone in the study cannot be born on 01JAN1960. This means that there some issue at the formatting level which are missed by both original and validation programmer.

Subject	Country	Race	Sex	Date of birth	Height (cm)	Weight (kg)
001	IND	Other	Male	01JAN1960	186	88.4
002	RUS	Other	Male	01JAN1960	187	92.0
003	USA	Black	Female	01JAN1960	160	68.0

Display 3. Example of Issue with Date Format

PAGE BREAK

Checking page break is a very crucial aspect in any PROC REPORT output. If a particular data listing is required to be paged by subject (new page per subject), then 'page' option in PROC REPORT should be used to separate out one subject's data from another subject.

```
Break after subject /page;
```

Display 4. Usage of Page Break

If this option is not used, then there is a high probability of mixing up records of sequential subjects.
During Automated Validation, all values would match in the validation and original programmer's dataset. Also, with a cursory check of such a report, the validation programmer may find everything in order. As a result, special attention should be given to this issue.

'FLOW' OPTION

Flow is one of the important options of PROC REPORT 'DEFINE' statement. This should be used in conjunction with the width option for generating error-free outputs. Validation programmer must take a look at the data thoroughly to identify long characters or numbers to avoid truncation.

Below is an *example* which can be missed out unless the data is checked properly.

Subject	Study Start date	Last contact date	Discontinued?	Reason
001	06SEP012	15SEP2012	Yes	Subject withdrew consent

Display 5. Example of 'FLOW' Option

In the above example, as the FLOW Option has not been used in the DEFINE statement of PROC REPORT, the entire text for reason of discontinuation which is "Subject withdrew consent (as she shifted from her current location to another city)" was truncated to display only "Subject withdrew consent". The visual and Automated Validation both would be unable to catch such an oversight.

SORTING PRODUCTION DATASET

The practice of sorting production dataset, during validation, is observed in many a validation codes. It may seem logical to pick up the production dataset, get it in the required sorting order and then compare it with the validation dataset. This may simplify validation marginally. However, when such a method is used to perform Automated Validation of datasets obtained from PROC REPORT, then the sorting issues that are seen in the PROC REPORT display are bound to be missed unless a thorough visual check is performed.

Example:

Below is a lab listing that has not been sorted properly with the required variable 'Time'. However, as the Automated Validation has found no issues due to prior sorting of the production dataset, such a sorting mistake may not be identified easily.

Lab Test	Unit	Visit	Date	Time	Result
Sodium	mmol/L	Screening	30MAR2012	10:47	200
		Baseline	11APR2012	15:30	168
		TRT Day 1	12APR2012	23:59	120
			12APR2012	11:00	135
		TRT Day 2	13APR2012	11:00	110

Display 6. Example of Sorting Production Dataset

Similarly, using NODUPKEY or NODUPRECS options, while sorting the production dataset, should be avoided as these erroneously remove duplicate records from production dataset. Resultantly, the PROC COMPARE output comes out clean but the table or listing which is being validated will contain duplicated records.

DATA MODIFICATION TO PRODUCTION DATASET

a. Usage of sub-setting conditions

There may be a manipulation made to the original dataset using a "where" or "if" condition as per below Example:

```
Proc Compare Base = production.conmed (where= (subject ne "")
             Compare = valdiaiton.conmed (where= (subject ne "")
             Out = compare.conmed outbase outcomp outnoequal outdif;
Run;
```

Display 7. Example of Data Modification to Production Dataset

Since, both production and validation programmer's datasets use the same subset condition, obviously the "OUT=" dataset (compare.conmed) would have "0" unequal records.
The data listing, however, would consist of missing subject numbers which may not be desirable.

b. It may happen that some variables are dropped from both; production and validation datasets. Validation programmer may think these are variables are not required to be compared since these are not displayed in the report. This assumption may lead to missing out display or sorting issues.

RECOMMENDATIONS

To minimize the above issues, there is a strong need to come up with a standard checklist and a set of 'Do's-and-Don'ts' for the validation programmers within each organization.

Having a set of standard production and/or validation codes for reporting would also be beneficial as subjective errors would be reduced.

Lastly, there should be minimal computation of variables handled at the reporting level. All the required derived variables should be computed in the analysis dataset itself and used as they are in the Table/ Listing creation and validation programs.

CONCLUSION

Using Automated Validation has definitely reduced quality issues to a great extent. However, it must be understood that it is not a foolproof method of validating outputs and has some limitations. So in conclusion, Manual and Automated Validation methods should definitely be used in harmony to validate any output. In short "Logical Validation" is a must. Apart from that, some time and effort should be put to thoroughly understand the data. Only then any output can be called as Validated!

REFERENCES

[1] "Don't Get Blindsided by PROC COMPARE" by Joshua Horstman and Roger Muller
(PharmaSUG 2013 Paper CC36)

[2] Automated or Manual Validation: Which One is for You? by Richann Watson and Patty Johnson
(PharmaSUG 2011 Paper AD01)

ACKNOWLEDGEMENTS

We would like to thank our manger Sandeep Sawant and our colleagues for their valuable inputs.

CONTACT INFORMATION

Name: Sneha Sarmukadam
Enterprise: inVentiv Health Clinical
Address: Building No.4, 6th Floor, Commerzone, Yerwada
City, State ZIP: Pune, Maharashtra, India - 411006
Work Phone: +91 20 3056 9113
E-mail: sneha.sarmukadam@inventivhealth.com