SESUG Paper 175-2018

Using PROC FORMAT to Automate Data correction process

Dalvi Shreyas, University of South Florida

ABSTRACT

SAS® has exceptional analytics capabilities, but to process data we often need to extract, transform, validate and correct the data that we get from various sources to make best use of its capabilities. Suppose in an application, we process customer data where we get information from data entry monthly, with multiple records having data entry errors. Periodically, we need to identify and correct those entries in the final SAS dataset as part of the data validation and correction process. It is time consuming to manually update each record monthly, therefore the need for an automated process arises to produce a final corrected data set. This paper demonstrates how we can update only the incorrect values in a SAS data set by using the external file which provides only the corrected values (finder file). This process does not make any data merges or SQL joins for the data correction. The process will use PROC format and will create the customized formats using CNTLIN for the finder file. Proc format will create the variable to be corrected and a unique master key having several variables concatenated to avoid errors in the correction process. Using this format, code will correct the invalid values in the variable and all remaining variables will remain the same. This paper is intended for intermediate level SAS Developers who want to build data validation and data correction programs using SAS.

INTRODUCTION

In various business processes, we have data correction and validation processes. The organizations always want to automate such processes. This paper explains how we can leverage the PROC FORMAT's format creation using a SAS dataset and proceed with data correction for the required observations. This methodology is useful in the scenario where we have millions of input records with thousands of records to be corrected. Instead of making complex SQL joins and data merges, PROC FORMAT makes use of option CNTLIN= (control in) to create a custom format using the dataset containing the records to be corrected. This helps in setting up an automated correction process where you can place the finder file (file having the values to be corrected) into the specified landing path in UNIX or flat-file in the case of a mainframe environment.

UNDERSTANDING INPUT DATA

Understanding the granularity of input data is very important to decide which variables are needed to create the unique master key. In this paper input data have multiple vehicles and multiple users for a single policy id. This is a reason we are including the id, first Name, Last name and License number to create the master key. The user should decide the variables needed to create unique master key based on the properties of the dataset.

Below is the input SAS dataset with incorrect values as VIN codes.

id	first_name	last_name	email	gender	VIN	License
1403	Daryl	Lorey	dloreyy@ca.gov	Female	WAUDG94F26N289830	FL525134
54918	Diego	Brotherhead	dbrotherheadz...	Male	5XYKT3A17DG881921	FL421318
97265	Martyn	Cestard	mcestard10@f...	Male	WDDHF0GBXXX	FL632505
10689	Darrick	Korneichuk	dkorneichuk11...	Male	JHMZF1D6XXXX	FL416665
46753	Lise	Haselup	lhaselup12@o...	Female	1C4NJCBAXXXX	FL587039
67226	Sayres	Brandino	sbrandino13@...	Male	1N6AA0CC3000	FL251961
23707	Imogene	Walpole	iwalpole14@va...	Female	XXXXKXDM662634	FL802746
39544	Iggy	Tremethack	itremethack15...	Male	1FADP5BU4DL464493	FL634154
71735	Fanya	Piet	fpiet16@thetim...	Female	1FTWX3D57AE153506	FL424688
82779	Gwenny	Dennerley	gdennerley17...	Female	SCFBB04C89G250812	FL622949
99996	Silvio	Euler	seuler18@360....	Male	1D4PU7GX7BW678830	FL621487
5531	Filippa	Plom	fplom19@g.co	Female	WAURGAFD9EN404571	FL600523
5371	Dougy	Dencs	ddencs1a@hh...	Male	2D4RN7DG2BR090899	FL266354
77620	Melessa	Shilleto	mshilleto1b@w...	Female	WAULC68E82A670655	FL686520
58782	Maximilien	Bainbridge	mbainbridge1c...	Male	JHMFA3F20BS458225	FL875554
96232	Devinne	Mallaby	dmallaby1d@g...	Female	WUADU98E58N111364	FL456214
61734	Alyson	Smitham	asmitham1e@...	Female	SAJWA4DC9DM911463	FL113089

Display 1. Input Dataset FILE_MAIN (Highlighted records have incorrect VIN's)

STEP1: FINDER FILE PROCESSING

The finder file consists of corrected records. Suppose we have incorrect VIN codes in our original file which need to be corrected and we have got the list of correct VIN codes from the external agencies which we want to use to update the original file.

This file is used to create the custom format using CNTLIN in PROC FORMAT. Next create a unique key to find the correct record where we update the value of VIN code. This ensures updating of the correct value.

Display2 is a screenshot of SAS dataset of finder file the corrected values of VIN codes:

id	first_name	last_name	VIN	License
97265	Martyn	Cestard	ADDHF1GB4EA490080	FL632505
10689	Darrick	Korneichuk	JHMZF9D61ES368193	FL416665
46753	Lise	Haselup	9X4FJXBAXDD144918	FL587039
67226	Sayres	Brandino	9F6AA4XX3DF499399	FL251961
23707	Imogene	Walpole	JF9XV6EKXDM660634	FL802746
39544	Iggy	Tremethack	9FADP5BU4DL464493	FL634154
71735	Fanya	Piet	9FTAX3D51AE953546	FL424688
82779	Gwenny	Dennerley	SXFBB44X89G054890	FL622949
99996	Silvio	Euler	9D4PU1GX1BA618834	FL621487
5531	Filippa	Plom	AAURGAFD9EF444519	FL600523
5371	Dougy	Dencs	0D4RF1DG0BR494899	FL266354
77620	Melessa	Shilleto	AAULX68E80A614655	FL686520
58782	Maximilien	Bainbridge	JHMFA3F04BS458005	FL875554
96232	Devinne	Mallaby	AUADU98E58F999364	FL456214

Display 2. Dataset FINDER with all correct VIN codes

Below is the code for creating KEYS using the FINDER dataset.

```
DATA FIND;
SET WORK.FINDER;

FORMAT FKEY1 $69.;
FKEY1 = PUT(ID,$8.) !! PUT(FIRST_NAME,$12.) !!
PUT(LAST_NAME,$15.) !! PUT(LICENSE,$34.);

FORMAT FKEY2 $17.;
FKEY2 = PUT(VIN,$17.);
RUN;

PROC SORT DATA=FIND OUT=TEMP NODUPKEY;
BY FKEY1;
```

STEP2: CREATING SAS DATASET TO CREATE THE CORRECTION FORMAT

This step prepares the SAS dataset which is given in the CNTLIN option in PROC FORMAT.

```
DATA VINFMT (RENAME=(FKEY1=START FKEY2=LABEL));
RETAIN FMTNAME '$VINC' TYPE 'C';
SET TEMP END=EOF;
OUTPUT;
IF EOF THEN
DO;
      FKEY1 = ' ';
      FKEY2 = 'XX';
      HLO='O';
      OUTPUT;
END;

PROC FORMAT CNTLIN=VINFMT CNTLOUT=VINFMTOUT;
PROC FORMAT;
SELECT $VINC;
```

FMTNAME	TYPE	ID	…	START	LABEL	HLO
$VINC	C	224	…	224Silvain Neathway FL263157	JH4XL96865X345365	
$VINC	C	585	…	585Joann Welling FL613792	9G6DJ5EVXA4459603	
$VINC	C	824	…	824Boris Hatton FL493688	5F9AF4FA9DF943683	
$VINC	C	1441	…	1441Dorolisa Lowdham FL779021	AP4AA0A95FS014439	
…	…	…	…	…	…	…
$VINC	C	96232	…	96232Devinne Mallaby FL456214	AUADU98E58F999364	
$VINC	C	97265	…	97265Martyn Cestard FL632505	ADDHF1GB4EA490080	
$VINC	C	98892	…	98892Trixi Beaford FL808572	ABAEV53490K001098	
$VINC	C	99996	…	99996Silvio Euler FL621487	9D4PU1GX1BA618834	
$VINC	C	99996	…		XX	O

Table 1. SAS Dataset VINFMT (Input to PROC FORMAT)

CHANGING THE TYPE OF FORMAT (THE 'TYPE' VARIABLE)

Variable TYPE can be used in your formatting dataset to modify the type of format you want to create. Type can have the following values with the following meanings:

Value	Means to Create
C	Character Format
N	Numeric Format
J	Character Informat
I	Numeric Informat
P	Picture Format

Table 2. TYPE of FORMAT

USING THE 'LOW' AND 'HIGH' VALUES IN YOUR FORMAT DATASET (HLO=)

What if you want to specify a starting value of 'LOW' or an ending value of 'HIGH' in your format? In this case, you cannot just put 'LOW' as the value of the 'START' variable or 'HIGH' as the value of the 'END' variable. Instead, you need to use a different variable called 'HLO'. Below are three of the values that HLO can have. There are several others.

HLO Value	Meaning
H	Range's ending value is HIGH (Value in END will be ignored)
L	Range's starting value is LOW (Value in START will be ignored)
O	Range is Other (both Start and End values are ignored)

Table 3. HLO Values and Meanings

HLO can be set to 'O' if you want to include a format range of 'Other' in your CNTLIN dataset. In our example, we don't have any range of values to be formatted. Both the START and END values in our tables would be Other and are assigned as 'XX'.

WHAT ARE CNTLIN AND CNTLOUT?

CNTLIN option can be used to specify a SAS data set for building informats and formats using the PROC FORMAT procedure. This will help in creating the FORMAT using the finder file in this example. CNTLOUT option can be used to generate a SAS data set with information about formats and informats. This is especially useful when we inherit permanent formats and like to learn more about the formats.

STEP 3: USING CUSTOM FORMAT FOR DATA CORRECTION

After creating formats using finder file, you can use this format to correct the input dataset. You should create the Master key with same combination of variables and length as created previously using the finder file. Newly created variable VINCD is used to hold the correct values of VIN from lookup table.

For converting the MASTKEY in $VINC format SAS searches MASTKEY in the lookup table and returns the value specified in LABEL (see Table1). If a match is not found it returns 'XX' as defined in VINFMT dataset.

This corrected VINCD is assigned to the variable VIN only if it needs to be corrected i.e. only if it is present in the lookup table.

Below is the code for final data correction step:

```
DATA FILECORR;
SET WORK.FILE_MAIN;
FORMAT MASTKEY $44.;
FORMAT VINCD  $17.;
MASTKEY = PUT(ID,$8.) !! PUT(FIRST_NAME,$13.) !!
PUT(LAST_NAME,$15.) !! PUT(LICENSE,$8.);
VINCD = PUT(MASTKEY,$VINC.);
IF VINCD NOT EQ 'XX' THEN
DO;
      VIN = VINCD;
END;
DROP VINCD MASTKEY;
RUN;
```

ID	FIRST_NAME	LAST_NAME	EMAIL	GENDER	VIN	LICENSE
1403	Daryl	Lorey	dloreyy@ca.gov	Female	WAUDG94F26N289830	FL525134
54918	Diego	Brotherhead	dbrotherheadz...	Male	5XYKT3A17DG881921	FL421318
97265	Martyn	Cestard	mcestard10@f...	Male	ADDHF1GB4EA490080	FL632505
10689	Darrick	Korneichuk	dkorneichuk11...	Male	JHMZF9D61ES368193	FL416665
46753	Lise	Haselup	lhaselup12@o...	Female	9X4FJXBAXDD144918	FL587039
67226	Sayres	Brandino	sbrandino13@...	Male	9F6AA4XX3DF499399	FL251961
23707	Imogene	Walpole	iwalpole14@va...	Female	JF9XV6EKXDM660634	FL802746
39544	Iggy	Tremethack	itremethack15...	Male	9FADP5BU4DL464493	FL634154
71735	Fanya	Piet	fpiet16@thetim...	Female	9FTAX3D51AE953546	FL424688
82779	Gwenny	Dennerley	gdennerley17...	Female	SXFBB44X89G054890	FL622949
99996	Silvio	Euler	seuler18@360....	Male	9D4PU1GX1BA618834	FL621487
5531	Filippa	Plom	fplom19@g.co	Female	AAURGAFD9EF444519	FL600523
5371	Dougy	Dencs	ddencs1a@hh...	Male	0D4RF1DG0BR494899	FL266354
77620	Melessa	Shilleto	mshilleto1b@w...	Female	AAULX68E80A614655	FL686520
58782	Maximilien	Bainbridge	mbainbridge1c...	Male	JHMFA3F04BS458005	FL875554

Display 3. FILECORR dataset with corrected values

Display3 shows the output SAS dataset with corrected VIN codes. You can make use of this stored Format to correct more files with the same data layout. In this process we don't need to make complex SQL joins SORT/MERGE for correcting multiple files. If the size of the main file is very large compared to the finder file, CPU efforts to SORT/MERGE the huge file can be avoided. In a practical scenario the percent of data to be corrected is much less compared with main file. Therefore, most of the time such approach of correcting the data is advantageous.

CONCLUSION

Correcting a huge dataset using various SORT/MERGE or SQL JOIN techniques consumes a lot of CPU time. This paper shows an alternate way to correct records using PROC FORMAT. If the input file is comparatively large than the finder file, then it is advantageous to use a custom format for the correction process. This way of correcting data will save CPU time as there is no need to MERGE/JOIN or SORT the input dataset. This paper also demonstrates how we can create Custom format from raw data using CNTLIN option. You can get more information on using the CNTLIN and other options in SAS online documentation for PROC FORMAT.

REFERENCES

Wendi L. Wright "Creating a Format from Raw Data or a SAS® Dataset" *Paper 068-2007*, Harrisburg, PA: SAS Global Forum 2007 http://www2.sas.com/proceedings/forum2007/068-2007.pdf.

PROC FORMAT Statement Base SAS(R) 9.2 Procedures Guide

ACKNOWLEDGMENTS

Special thanks to my SAS professor Dr. Alexia Athienitis PhD, University of South Florida for guiding, encouraging me on technical writing and reviewing this paper.

RECOMMENDED READING

- *Base SAS® Procedures Guide*
- *SAS® For Dummies®*

CONTACT INFORMATION

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

Shreyas Dalvi
University of South Florida
Phone: 814-573-4627
E-mail: shreyasdalvi@mail.usf.edu
LinkedIn: https://www.linkedin.com/in/shreyasdalvi/