CT-30

Beyond "If then"--

Three Techniques for Cleaning Character Variables from Write-in Questions

Yusheng Zhai, American Cancer Society, Atlanta, GA
Ryan Diver, American Cancer Society, Atlanta, GA
Xia Lin, American Cancer Society, Atlanta, GA

ABSTRACT

In survey studies, cleaning answers to write-in questions can be difficult and time consuming, especially when the same response may be written in multiple ways. Misunderstanding of the survey question, unrecognizable handwriting, and negligence in data entry are major factors leading to data inaccuracies that are almost impossible to avoid. Writing a series of "if then" statements is a classic solution to cleaning data. However with growing datasets, the number of conditions to be tested grow too, until thousands of "if then" statements may be required.

This paper presents three techniques that we used to clean up the country of birth questions in the Cancer Prevention Study-3 (CPS-3). Combining data merging and Excel spreadsheets, using LIKE and SOUND LIKE operators, and implementing join tables with compare functions in the SQL procedure not only reduced the workload and eases the stress in cleaning character variables but also added some flavors to this tedious task.

INTRODUCTION

The Epidemiology Research Program at the American Cancer Society is currently conducting the Cancer Prevention Study-3 (CPS-3) in order to better understand the genetic, behavioral, environmental, and lifestyle factors that may cause or prevent cancer. The goal is to recruit a diverse group of 300,000 men and women across the US and Puerto Rico. The initial survey contains these questions: "Where were you born", "Father's country of birth", and "Mother's country of birth". Five common answers are provided: U.S., Canada, Mexico, India, and China. If none of the above five answers is applicable, the answer of "other" is available and a 15-digit field was provided for reporting the name of the country. The data for the first recruitment period were recently received and is undergoing a cleaning process. A frequency table shows that there are 402 distinct reported countries for the question "Where were you born", 628 distinct reported countries for the question "Father's country of birth", and 664 distinct answers for the question "Mother's country of birth". After merging the three datasets by reported country, there are 1015 distinct answers reported by participants.

The ISO-3611-1 is a widely accepted standard published by the International Organization for Standardization (ISO) on defining the names of countries, dependent territories, and special areas of geographical interest. Based on the ISO-3611-1, there are currently 249 countries with assigned official codes. The ISO-3611-1 contains three sets of country codes: the two-letter country codes alpha-2 (named ISO3611A2); the three-letter country codes alpha-3 (named ISO3611A3), and three-digit country codes (named ISONUM). The alpha-2 codes are the most widely used of the three while alpha-3 allows a better recognition of the country name. The three-digit country code offers exceptional advantages in script independence, and hence is necessary for non-Latin script societies. The ISO 3166 Maintenance Agency (ISO 3166/MA) also establishes official English short names that are commonly accepted. Table 1 shows a partial list of the ISO 3166-1 code from the ISO official site.

SHORTNAME	LONGNAME	ISO3611A2	ISO3611A3	ISONUM
AFGHANISTAN	Afghanistan, Islamic Republic of	AF	AFG	004
ALBANIA	Albania, Republic of	AL	ALB	008
ALGERIA	Algeria, People's Democratic Republic of	DZ	DZA	012
AMERICAN SAMOA	American Samoa	AS	ASM	016
ANDORRA	Andorra, Principality of	AD	AND	020
ANGOLA	Angola, Republic of	AO	AGO	024
ANGUILLA	Anguilla	AI	AIA	660

Table 1.Partial list of the ISO 3166-1

The objective of the cleaning procedure reported here is to assign the reported country names to the official English short names according to the ISO-3611-1 standard, and then assign the corresponding ISO-3611-1 codes to them.

TECHNIQUE 1: DATA STEP MERGE AND EXCEL SPREADSHEETS

Instead of writing hundreds of lines of "if then" statements, we propose a simple technique that combines data merging and Excel spreadsheet. Before the cleaning process, we will need to create a dataset with complete set of

answers to the question. In the example, we create a dataset (ISOLIST) with all the country names from the ISO 3166-1 code. Second, a matching process is employed to screen out the corrected answers from the original dataset. The remaining incorrect answers are exported to EXCEL for manual correction.

The frequency table for all the answers to the question is obtained via the FREQ procedure and output to Excel.

NODUPKEY and OUT options are used to output a dataset (CFREQ) with all unique reported country names.

```
PROC SORT DATA=ALLCOUNTRY NODUPKEY OUT=CFREQ;
  BY COUNTRY;
RUN;
```

Then, we merge the output dataset (CFREQ) with the ISO-3611-1 dataset by country name, so the exact matched countries will show up in a new column called ‘SHORTNAME’. 121 out of 1015 (11%) reported names are matched exactly.

```
DATA ISO;
  SET  ISOLIST;
       COUNTRY=SHORTNAME;
   KEEP COUNTRY SHORTNAME;
RUN;
DATA ISOCTR;
  MERGE CFREQ(IN=C) ISO;
  BY COUNTRY;
  IF C;
RUN;
```

Next, we export the dataset with all unique reported country names and exact match results into Excel.

```
PROC EXPORT DATA=ISOCRT OUTFILE='S:\USER\Output\CFREQ.xls'
            DBMS=EXCEL REPLACE;
            SHEET = 'CFREQ';
RUN;
```

In the spreadsheet, the first column, ‘COUNTRY’, is the unique reported country name. The second column, ‘SHORTNAME’, contains all the exact match results. Cells left empty in the ‘SHORTNAME’ column are those that need to be manually corrected. Since they are sorted by alphabetical order, it is easy to match them to the standard country name list. Although the remaining 894 unmatched reported country names still need to be corrected manually, it is a lot easier to do it in Excel compared with writing “if then” statements in SAS®. In addition, the Excel spreadsheet is easily ordered allowing better visualization for the cleaning process. All we have to do is to put the official country names on the SHORTNAME column. After the correction, the Excel file (ISOCRTC) is ready to be imported back to SAS. Table 2 demonstrates how it looks in Excel.

COUNTRY	SHORTNAME
ALBANIA	ALBANIA
SLOVINIA	SLOVENIA
CONGO KINSHASA	CONGO REPUBLIC
ENGLAND, UK	UNITED KINGDOM
ITAL	ITALY
NTHERLANDS	NETHERLANDS
TURKET	TURKEY
MADRID SPAIN	SPAIN

Table 2. Columns of “COUNTRY” and “SHORTNAME” in ISOCRTC dataset

To imbed the complete ISO 3166 -1 system, simply merge the ISOCRTC file with the standard ISO 3166-1 table by the official name (SHORTNAME).

The final dataset contains all the unique reported country names, the official names, and the ISO 3166-1 codes. Merging with the individual raw file by reported country name is the final step of the process.

TECHNIQUE 2: UTILIZE LIKE AND SOUND LIKE OPERATORS

The previous method is easy to understand and apply, however the technique only identifies 121 exact matches requiring manual matching of the remaining 894 answers. Is there a way to reduce the workload? Applying the LIKE and SOUND LIKE operators in SAS macros can increase the chance of matching and reduce the manual workload later.

Looking at the spreadsheet, the first problem we noticed is that there are many misspellings on the reported country names. In most circumstances, these misspellings are close enough to the correct names on either writing or phonetics. The second problem is that one country may have several conventional names in addition to its official name. For example, the NETHERLANDS is also known as HOLLAND; GREAT BRITAIN, ENGLAND, SCOTLAND, and WALES should all use UNITED KINGDOM as the official name under the ISO-3611-1 standard. The third problem is that some state names of the U.S. were reported as the country name. Taking advantage of these three typical characteristics of the dataset and utilizing the LIKE and SOUND LIKE conditions can considerably reduce the workload of the manual correction process.

The LIKE condition compares character strings with a pattern-matching specification. The SOUND LIKE (*=) does Phonetic Matching on character strings following the American Soundex Coding rule. The LIKE condition is case-sensitive while the SOUND LIKE condition is not. The LIKE condition is more restricted and less likely to identify large variations. On the other hand, the SOUND LIKE condition can identify a larger range of possible matches, but is more likely to incorrectly match. Whether to use the LIKE or SOUND LIKE condition (or a combination of both) for the best results will depend on the data being cleaned.

The first step of macro MATCH1 (APPENDIX A.) uses the CALL SYMPUT function to create macro variables corresponding to each of the standard country names. The column SHORTNAME in the ISOLIST file contains the official names of all 249 countries. The code below creates 249 macro variables (CTR1-CTR249) and assigns them the official country names.

```
DATA ISO;
SET DEMO.ISOLIST;
  CALL SYMPUT ('CTR' ||LEFT(_N_),SHORTNAME);
RUN;
```

The %put statement shows the value assigned to the macro valuables in the log.

```
%PUT &CTR1 &CTR2 &CTR3;
```

SAS log: AFGHANISTAN ALBANIA ALGERIA

Then the dataset (ISOCRT) with all the unique reported names and exact matched results is divided into subsets by grouping the country names that look like (the LIKE condition) the value of the macro variable. In the WHERE statement, the UPCASE function transfers all characters to capital letters, the COMPBL function compresses multiple blanks into one, and the STRIP function removes any leading and trailing blanks. The value of a macro variable is defined as character with a maximum length of 65,534 and there are often trailing blanks in it. Therefore, the STRIP operator is necessary. The percent sign (%) allows the LIKE condition to ignore the head and the tail of a text string so that it matches any sequence characters in the middle. The official country name (SHORTNAME) is assigned to each of the subsets and all the subsets are merged together afterwards. The LIKE Operator in this macro produces 236 out of 1015 (23%) fairly precise matches compared to the exact match results (121/1015, 11%).

```
%DO  M=1 %TO 249 ;
   DATA COUNTRY&M;
     SET &DATASET;
       WHERE SHORTNAME = ' ' AND
       STRIP(COMPBL(UPCASE(&CVAR))) LIKE '%'||STRIP("&&CTR&M")||'%';
       SHORTNAME = STRIP(COMPBL("&&CTR&M"));
   RUN;

   PROC SORT DATA=COUNTRY&M;
     BY &CVAR;
   RUN;
%END;

PROC SORT DATA=&DATASET;
  BY &CVAR;
RUN;
```

```
DATA &DATASET.ALL;
  MERGE &DATASET COUNTRY1 - COUNTRY249;
  BY &CVAR;
RUN;
```

To use the SOUND LIKE Operator, simply change the WHERE statement to:

```
WHERE STRIP(COMPBL(&CVAR)) = * STRIP("&&CTR&M");
```

The SOUND LIKE condition is less specific when matching, therefore the percent sign (%) is no longer necessary. The macro with this condition found an extra 268 matches compared to the exact matching and produced 389 out of 1015 (38%) less specific matches. The SOUND LIKE matches need to be closely inspected to make sure there are no incorrect matches. The LIKE OR SOUND LIKE combination yields 458 out of 1015 (45%) non-specific matches while the LIKE AND SOUND LIKE combination returns 156 out of 1015 (15%) very precise matches, which do slightly better than the exact matching (112/1015, 11%).

For countries that have more than one conventional name, we create new columns (name2, name3, for the second and third names, etc.). We also create a dataset with all the states in the U.S. By utilizing these two extra features in the macro, a single LIKE Operator returns an extra 226 matches and raises the results to 338 out of 1015 (33%) (see APPENDIX B).

The rest of the unmatched reported country names are handled in the Excel spreadsheet as described above.

TECHNIQUE 3: LIKE AND SOUND LIKE IN SQL JOIN

SQL language is not commonly considered well-visualized and is somewhat difficult to understand. However, the SQL procedure offers a great deal of convenience and flexibility on data merging. Specifically, PROC SQL has promising advantages in terms of shortening the program and performing sophisticated data merging procedures. When joining tables using SQL, sorting the data in advance is no longer necessary and conditions such as LIKE and SOUND LIKE can be incorporated into the merging criteria. The program below demonstrates how to incorporate the LIKE condition in a very short PROC SQL join program that accomplishes similar results to the large macro described above. The LEFT JOIN condition matches both tables and retains all rows from the left table (COUNTRYFREQ). The rows from the left table are preserved and captured exactly as they are stored in the table itself, regardless of whether or not a match exists.

```
PROC SQL;
  CREATE TABLE LJOIN AS
  SELECT CFREQ.COUNTRY , ISO.*
  FROM CFREQ AS A LEFT JOIN ISO AS B
  ON STRIP(COMPBL(UPCASE(A.COUNTRY))) LIKE '%'||STRIP(B.SHORTNAME)||'%';
QUIT;
```

Again, to use the SOUND LIKE Operator, simply change the ON statement as follows:

```
ON STRIP(COMPBL(A.COUNTRY)) = * STRIP(B.SHORTNAME);
```

After the matching with country name, we apply the same process to match the answers with states in the U.S.

```
PROC SQL;
   CREATE TABLE LJOIN2 AS
   SELECT LJOIN.* , US_STATES.*
   FROM LJOIN AS A LEFT JOIN US_STATES AS B
   ON STRIP(COMPBL(UPCASE(A.COUNTRY))) LIKE '%'||STRIP(B.STATE)||'%';
QUIT;
```

It is worth noting that duplicate matches will likely occur when employing the LIKE or SOUND LIKE condition to the SQL joins. The SQL joins paired every match that looked or sounded relatively close. For example, “NIGERA” could be matched with “NIGER” and “NIGERIA” under the LIKE condition. Because of that, in the joined dataset, there will be two matched rows for “NIGERA” instead of one. To evaluate the matching and select the best possible match pair, SAS compare function, COMPARE, COMPGED and CALL COMPCOST, COMPLEV, AND SPEDIS can be used. For more information of how these SAS compare functions work and how to apply them, please refer to Amanda Roesch’s paper. Another concern with the SQL join is that since all matching is done simultaneously and equally, answers that sound like or look like “GEORGIA” will be paired with both Georgia the country and Georgia the U.S. state. The macro we proposed eliminates the duplicated matching problem by assigning priority in matching process (first country and then states in the U.S.) and removing those reported country names from the dataset after the first match condition is satisfied.

After selecting the best matched pair from the duplications using the SAS compare functions, the SQL join produced the same results as the previous macro with 236 out of 1015 (23%) matches using the LIKE condition and 389/1015

(38%) matches using a single SOUND LIKE condition.

CONCLUSION

The old-school method of writing thousands of “if then” statements is mind-numbing and intensive, but data cleaning does not have to be that way! This paper introduces three methods that handle the cleaning of write-in questions differently. The Matching and Excel method gives you the convenience and flexibility to get the job done easily in Excel spreadsheet. The SAS macro method introduces the LIKE and SOUND LIKE conditions and illustrates how these conditions can considerably reduce the workload. PROC SQL can employ the LIKE and SOUND LIKE into the join table process to achieve similar result with a significantly shortened program. The SAS compare functions can be used in evaluating the matching in order to select the best matching results. The selection of the three techniques and the use of LIKE and SOUND LIKE conditions are subject to personal programming preference, and the type of data to be cleaned. Review of the matching is an important quality control step in any data cleaning process, and will help to identify the best strategy to use in your dataset. Although none of the described techniques can get all the cleaning done completely, they do reduce the workload to varying degrees and add some programming excitement to the cleaning process! The benefit of using above techniques will grows as the dataset gets bigger and these ideas can be extended to the cleaning of all types of character variables.

REFERECNES

- SAS 9.2 Help and Documentation
- ISO. 1997. ISO-3166-1. Codes for the representation of names of countries and their subdivisions - Part 1: Country Codes. Fifth edition. ISO 3166 Maintenance Agency at DIN, Berlin
- Roesch, Amanda. Matching Data Using Sounds-like Operators and SAS Compare Functions, Proceedings of the SAS Global 2012 Conference, Orlando, FL. Available at http://support.sas.com/resources/papers/proceedings12/122-2012.pdf

CONTACT INFORMATION

Comments, suggestions, and questions are welcomed at:

Yusheng Zhai MSPH
American Cancer Society Inc.
250 Williams Street NW
Atlanta, GA 30303
Work Phone: 4043295764
E-mail:Yusheng.Zhai@cancer.org

APPENDIX A

```
/*=============================================================================
 Program   : MATCH1.SAS
 Version   : 9.2
 Date      : July 28, 2012
 Purpose   : USE LIKE/SOUND LIKE OPERACTOR TO MATCH THE REPORTED COUNTRY
             NAMES WITH THE ISO COUNTRY NAMES
 Usage     : %MATCH1 (DATASET, CVAR);
=============================================================================;
 PARAMETERS:
------- NAME ------- ------------------------ DESCRIPTION ------------------
    &DATASET         THE DATASET WITH ALL THE UNIQUE REPORTED COUNTRY NAMES
    &CVAR            THE VARIABLE OF THE REPORTED COUNTRY NAMES
-----------------------------------------------------------------------------
 RESULT:  RETURN DATASET &DATASET.ALL WITH MATCHED REPORTED AND ISO
          COUNTRY NAMES
=============================================================================*/

%MACRO MATCH1 (DATASET, CVAR);

* CREATE MACRO VARIABLE FOR EACH OF THE OFFICIAL COUNTRY NAME;
DATA ISO;
  SET DEMO.ISOLIST;
  CALL SYMPUT ('CTR' ||LEFT(_N_),SHORTNAME);
RUN;

%PUT &CTR1 &CTR2 &CTR3;

*** 249 COUNTRIES;
%DO M=1 %TO 249 ;
    DATA COUNTRY&M;
    SET &DATASET;
    WHERE SHORTNAME = ' ' AND
          STRIP(COMPBL(UPCASE(&CVAR))) LIKE '%'||STRIP("&&CTR&M")||'%';
          SHORTNAME = STRIP(COMPBL("&&CTR&M"));
    RUN;

    PROC SORT DATA=COUNTRY&M;
      BY &CVAR;
    RUN;
%END;

PROC SORT DATA=&DATASET;
  BY &CVAR;
RUN;

* MERGED THE SUBSETS;
DATA &DATASET.ALL;
  MERGE &DATASET COUNTRY1 - COUNTRY249;
  BY &CVAR;
RUN;

* DELETE THE TEMPORAL DATASETS;
PROC DATASETS NOLIST;
  DELETE COUNTRY1 - COUNTRY249;
QUIT;

%MEND;

%MATCH1(ISOCTR, COUNTRY);
```

APPENDIX B

```
/*=============================================================================
 Program   : MATCH.SAS
 Version   : 9.2
 Date      : July 28, 2012
 Purpose   : USE LIKE/SOUND LIKE OPERACTOR TO MATCH THE REPORTED COUNTRY
             NAMES WITH THE ISO COUNTRY NAMES AND STATE NAMES IN THE U.S.
 Usage     : %MATCH (DATASET, CVAR);
=============================================================================;
 PARAMETERS:
------- NAME ------- ------------------------ DESCRIPTION ------------------
    &DATASET          THE DATASET WITH ALL THE UNIQUE REPORTED COUNTRY NAMES
    &CVAR             THE VARIABLE OF THE REPORTED COUNTRY NAMES
-----------------------------------------------------------------------------
 RESULT:  RETURN DATASET &DATASET.ALL WITH MATCHED REPORTED AND ISO
          COUNTRY NAMES
=============================================================================*/


%MACRO MATCH (DATASET, CVAR);

* CREATE MACRO VARIABLES FOR EACH OF THE OFFICIAL COUNTRY NAME;
DATA ISO;
SET DEMO.ISOLIST;
 *OFFICIAL COUNTRY NAMES;
  CALL SYMPUT ('CTR'||LEFT(_N_),SHORTNAME);

 *FOR COUNTRIES HAVE OTHER CONVENTIONAL NAMES BESIDES OFFICIAL NAME;
  CALL SYMPUT ('CN1'||LEFT(_N_),CNAME1);
  CALL SYMPUT ('CN2'||LEFT(_N_),CNAME2);
  CALL SYMPUT ('CN3'||LEFT(_N_),CNAME3);
  CALL SYMPUT ('CN4'||LEFT(_N_),CNAME4);
  CALL SYMPUT ('CN5'||LEFT(_N_),CNAME5);
RUN;

* CREATE MACRO VARIABLES FOR EACH OF THE STATE IN THE U.S.;
DATA US_STATES;
SET DEMO.US_STATES;
 *OFFICIAL STATE NAMES;
  CALL SYMPUT ('ST1'||LEFT(_N_),STATE);

 *TWO-LETTER STATE ABBREVIATION;
  CALL SYMPUT ('ST2'||LEFT(_N_),STATEABB);
RUN;

*** 249 COUNTRIES, INCLUDED MULTIPLE COMMON NAMES;
%DO M=1 %TO 249 ;
    DATA COUNTRY&M;
          SET &DATASET;
             WHERE
                SHORTNAME = ' ' AND(
                   STRIP(COMPBL(UPCASE(&CVAR))) LIKE '%'||STRIP("&&CTR&M")||'%'
                %IF (%NRQUOTE(&&CN1&M) NE   )  %THEN  OR
                   STRIP(COMPBL(UPCASE(&CVAR))) LIKE  STRIP("&&CN1&M");

                %IF (%NRQUOTE(&&CN2&M) NE   )  %THEN  OR
                   STRIP(COMPBL(UPCASE(&CVAR))) LIKE  STRIP("&&CN2&M");

                %IF (%NRQUOTE(&&CN3&M) NE   )  %THEN  OR
                   STRIP(COMPBL(UPCASE(&CVAR))) LIKE  STRIP("&&CN3&M");

                %IF (%NRQUOTE(&&CN4&M) NE   )  %THEN  OR
                   STRIP(COMPBL(UPCASE(&CVAR))) LIKE  STRIP("&&CN4&M");

                %IF (%NRQUOTE(&&CN5&M) NE   )  %THEN  OR
                   STRIP(COMPBL(UPCASE(&CVAR))) LIKE  STRIP("&&CN5&M");
```

```
                    );
                  SHORTNAME = STRIP(COMPBL("&&CTR&M"));
    RUN;

    PROC SORT DATA=COUNTRY&M;
      BY &CVAR;
    RUN;
%END;


*** 50 STATES IN THE U.S.;
%DO N=1 %TO 50;

    DATA US&N;
     SET &DATASET;
     WHERE SHORTNAME = ' '
             AND
             (STRIP(COMPBL(UPCASE(&CVAR))) LIKE '%'||STRIP("&&ST1&N")||'%'
             OR
             STRIP(COMPBL(&CVAR)) = STRIP("&&ST2&N")
             );
             SHORTNAME = "UNITED STATES";
    RUN;

    PROC SORT DATA=US&N;
      BY &CVAR;
    RUN;
%END;

PROC SORT DATA=&DATASET;
  BY &CVAR;
RUN;

* MERGED SUBSETS FROM THE COUNTRY MATCHED AND SUBSETS FROM STATE MATCHED;

DATA &DATASET.ALL;
  MERGE &DATASET COUNTRY1 - COUNTRY249 US1 - US50;
  BY &CVAR;
RUN;

* DELETE THE TEMPORAL DATASETs;
PROC DATASETS NOLIST;
  DELETE COUNTRY1 - COUNTRY249 US1 - US50;
QUIT;

%MEND;

%MATCH(ISOCTR, COUNTRY);
```