SAS program to standardize business names

Nada Wasi
University of Michigan

Ann Rodgers
University of Michigan

Kristin McCue
U.S. Census Bureau

Abstract

Probabilistic record linkage is often a key step in combining information about the same business over time or across data sources. Where string similarity measures are used, standardizing fields is a crucial pre-processing step that improves the accuracy and efficiency of probabilistic linking methods. Finding few publicly available tools adapted specifically to business names, we put together a set of standardization rules. Here we describe how we have implemented them in SAS, and provide examples that illustrate how to use them.

Keywords: standardizing, parsing, record linkage, SAS.

1. Introduction

Databases on businesses have become more widely available to researchers, and linking information about the same business over time or across data sources is often a key step in analyzing such data. Researchers may need to deduplicate records in the same database, or link records across data sources to combine different pieces of information. For example, linking information about respondents' employers from a household survey to the associated records in a business survey would improve our understanding of how employers and their employees affect each other. Ideally, unique identifiers (such as tax identification numbers) would unambiguously identify duplicates or units that should be linked across sources. But often unique identifiers are not available, and linkages must instead rely on partial identifiers such as business names and addresses. In these cases, probabilistic record linkage is commonly used, with string similarity measures applied to the available fields.[1]

With such an approach, standardizing text fields is a crucial pre-processing step that improves the accuracy and efficiency of probabilistic linking methods. There are numerous reasons why the name recorded for a particular employer may differ across sources, including differing use of abbreviations or acronyms; inconsistencies in the amount of detail included; businesses

[1] See Christen (2012) and Winkler (2006) for comprehensive review of probabilistic record linkage.

that have different trade and legal names (e.g. franchisees); or simply misspellings. Finding few publicly available tools designed specifically to standardize business names, a group of researchers at the University of Michigan, the U.S. Census Bureau, and Cornell University collaborated to put together standardization rules. This article describes how we have implemented these rules in Base SAS.

Tables 1 and 2 give some examples that we have created from publicly available data to illustrate ways in which the information reported by businesses themselves (Table 1) often differs from employer information provided by household respondents (Table 2).[2] For example, row 2 in Table 1 and row 1 in Table 2 contain quite similar information, but the household record has more abbreviated business name and address information. Also, without standardization,"AZTEC IND" (row 3 in Table 2) would appear more similar to "ASTEC Inc" (row 7 in Table 1) rather than "ASTEC INDUSTRIES INCORP", the correct one (row 4 in table 1).

obs	Company Name	Street Address	State
1	ABIOMED INCORPORATED	22 CHERRY HILL DR	MA
2	ADVANCED ANALOGIC TECHNOLOGIES LTD	830 E. ARQUES AVENUE	CA
3	ANALYSTS INTERNATIONAL CORP FKA ANALYSTS CORP	3601 WEST 76TH ST	MN
4	ASTEC INDUSTRIES INCORP	1725 SHEPHERD ROAD	TN
5	Aztec	1510 N Liberty Hill Rd # E	TN
6	LA Azteca Mexican Bakery	411 Alexander Dr	TN
7	ASTEC Inc	1699 Commercial Ave	WY
8	Aztec Painting		TN
9	BALCHEM PARTNERS	P O BOX 600	NY
10	BLUEFLY INC PC	42 WEST 39TH ST	NY
11	BROADVIEW INSTITUTE	4455 WEST 77TH STREET	MN
12	CHURCHILL DOWNS COMPANY	700 CENTRAL AVE	KY
13	COLUMBIA LABORATORIES PC	354 EISENHOWER PARKWAY	NJ
14	CONCORD CAMERA CORPORATION	4000 HOLLYWOOD BLVD STE 650	FL
15	COST U LESS INC	3633 136TH PLACE SE, SUITE 110	WA
16	Cost U Less Cars	701 Riverside Ave # 1	CA
17	Cost Plus World Market	10300 NE 8th St	WA
18	Costco	4299 Meridian St	WA
19	FLUSHING FINANCIAL CORP T/A FFC	1979 MARCUS AVENUE , SUITE E140	NY
20	Ruths Chris Steak House, Inc.	3321 HESSMER AVENUE	LA
21	SOUTHERN CALIFORNIA EDISON CO	2244 WALNUT GROVE AVE P O BOX 800	CA
22	STREAMLINE HEALTH SOLUTIONS, PROF CORP	10200 ALLIANCE ROAD SUITE 200	OH

Table 1: Business Reported Names and Addresses (Created from Public Filings)

In general, business-reported names are much more likely to include terms such as INC, CO, LTD, and LLP. Each of our household example records in Table 2 has a corresponding record in Table 1 that is close to it in content, but does not match exactly. If we simply sorted the records and linked only those with exact matches on each text string, none of these records would qualify as a match. The **%stnd_compname** module standardizes how text information appears in different datasets and parses the components of names into separate fields. By standardizing, we mean consistently handling abbreviations of common terms (e.g. replace INTERNATIONAL with INTL, replace INDUSTRIES with IND), removing

[2] None of the information in this or any other table is based in any way on confidential data. We created our examples of business-reported names and addresses by starting from a sample of names and addresses provided by businesses in public filings with the Securities and Exchange Commission. We have edited some of the business names and addresses to work in additional examples of patterns we have come across in working with employer administrative records. We have created examples of "household-reported" names and addresses by editing the original names and addresses to reflect common patterns.

obs	Employer name	Employer address	State
1	ADVANCED ANALOGIC	830 ARQUES AVE	CA
2	ANALYSTS INTL	3601 W 76TH ST	MN
3	AZTEC IND	1725 SHEPHERD ROAD	TN
4	BALCHEM CORP	214 MAIN ST	NY
5	BLUEFLY	39TH ST & 6th AVE	NY
6	CHURCHILL DOWNS	700 RTE 214	KY
7	COLOMBIA LABS	354 EISENHOWER PKWY	NJ
8	COST-U-LESS STORE	3633 136TH PLACE	WA
9	FLUSHING FINANCIAL	1979 MARCUS AVENUE SUITE 140	NY
10	Ruth's Chris Steakhouse	1219 W. 56th St	OH

Table 2: Pseudo Household-Reported Employer Names and Addresses

most unnecessary punctuation (e.g. commas), and trimming extra spaces. Parsing allows researchers to compare analogous pieces of information to each other and to give more weight to some parts of a name than others. Examples of issues handled by our parsing routines include separating a company's trade name from its legal name or separating entity type (e.g. CORP or INC) from more distinctive elements of company names.

The next section gives the command line for the **%stnd_compname** module. Section 3 shows an example using the sample data above. Section 4 discusses how advanced users can customize the program for their applications.

2. %stnd_compname module

%stnd_compname standardizes and parses a string variable containing company names into 6 new components. The new generated outputs are in the following order: official name, Doing-Business-As (DBA) name, Formerly-Known-As (FKA) name, attention name, business entity type, and business entity type for the DBA part of the name. Each component is standardized. When a name cannot be parsed, the original value is recorded in the official name field. If a user specifies only one new variable name for output, only the standardized official name will be output. The module requires at least the first four inputs listed below.

%stnd_compname(

```
(1) name of dataset with company name to be standardized
(2) name of dataset to be created with standardized fields added
(3) name of variable to standardize
(4) standardized (official) name with entity info removed
  (5) doing-business-as, traded-as name (optional)
  (6) fka=formerly known as name (optional)
  (7) attn=mailing name (usually a person) (optional)
  (8) entity type (optional)
  (9) entity type for the DBA part of the name (optional)
```

);

%stnd_compname relies on a sequence of subcommands and a set of ancillary rule-based pattern CSV files. These subcommands and pattern files must also be installed. The pattern files were developed as separate files in CSV format so that they can be customized and can be used by more than one program. For example, Wasi and Flaaen (2015) developed a set of STATA programs to standardize business names which uses these same pattern files.

The base pattern file directory must be specified using the `&pattern_path` macro variable before calling **%stnd_compname**. For example,
`%let pattern_path=c:\SWELL\patternfiles`
tells the program to look for the pattern files in the directory `c:\SWELL\patternfiles\theme\public`. The default directory location for the pattern files that are distributed with **%stnd_compname** is ‘public’. The subcommands and their associated pattern files are listed below.

Subcommands	Default pattern file
%parsing_namefield	P10_namecomp_patterns.csv
%stnd_specialchar	P21_spchar_namespecialcases.csv
	P22_spchar_remove.csv
	P23_spchar_rplcwithspace.csv
%stnd_entitytype	P30_std_entity.csv
%stnd_commonwrd_name	P40_std_commonwrd_name.csv
%stnd_commonwrd_all	P50_std_commonwrd_all.csv
%stnd_numbers	P60_std_numbers.csv
%stnd_nesw	P70_std_NESW.csv
%stnd_smallwords	P81_std_smallwords_all.csv
%parsing_entitytype	P90_entity_patterns.csv
%agg_acronym	

Table 3: Subcommands used in **%stnd_compname**

There is no pattern file for the `%agg_acronym` subcommand. When a particular pattern file is not found, the program will display a warning message and the standardizing or parsing step associated with that pattern file will be skipped. In section 4, we discuss how advanced users can modify these pattern files.

3. Example

The command lines below show an example of how to set up the library and run this macro. The raw input data is called “fileA” and is in the `c:\swell\raw` directory. “name” is the variable that contains the company names in fileA to be standardized. In the example program provided, “fileA”’ contains the data values listed in Table 1. The standardized output’s file name is called “fileAstnd” and it will be stored in the `c:\swell\stnd` directory. ***stnd_nm, stnd_dba, stnd_fka, stnd_attn, stnd_ent, stnd_dbaent*** are the new variables to be saved in this output file. Because we have included the line `%let theme=pass1;` , the program in this example will look for the pattern files in directory `C:\swell\PatternFiles\stndpatterns\theme\pass1`.

```
%let sqlopt=noprint;
filename cenmacro "C:\swell\macros\standardizer";
options nosource mautosource sasautos=(sasautos cenmacro);
libname dataraw "C:\swell\raw";
libname datastnd "C:\swell\stnd";
%let pattern_path=C:\swell\PatternFiles\stndpatterns;
%let theme=pass1;
%stnd_compname(
dataraw.fileA,
datastnd.fileAstnd,
name,
stnd_nm,
stnd_dba,
stnd_fka,
stnd_attn,
stnd_ent,
stnd_dbaent
);
```

Table 4 and Table 5 show the standardized outputs that result from applying **%stnd_compname** to the inputs from Tables 1 and 2, respectively. Columns 2-5 show the standardized names parsed into 4 fields. In order to conserve space, we do not display ***stnd_attn*** and ***stnd_dbaent***, which are blank fields in our examples.

obs	stnd_nm	stnd_dba	stnd_fka	stnd_ent
1	ABIOMED			INC
2	ADVANCED ANALOGIC TECHNOLOGIES			LTD
3	ANALYSTS INTL		ANALYSTS CORP	CORP
4	ASTEC IND			INC
5	AZTEC			
6	LA AZTECA MEXICAN BAKERY			
7	ASTEC			INC
8	AZTEC PAINTING			
9	BALCHEM PARTNERS			
10	BLUEFLY			INC PC
11	BROADVIEW INSTITUTE			
12	CHURCHILL DOWNS			CO
13	COLUMBIA LAB			PC
14	CONCORD CAMERA			CORP
15	COST U LESS			INC
16	COST U LESS CARS			
17	COST PLUS WORLD MARKET			
18	COSTCO			
19	FLUSHING FINANCIAL	FFC		CORP
20	RUTHS CHRIS STEAK HOUSE			INC
21	SOUTHERN CALIFORNIA EDISON			CO
22	STREAMLINE HEALTH SOLUTIONS			PC

Table 4: Standardized output from Table 1

4. Pattern files

`%stnd_compname` relies on a sequence of subcommands and ancillary rule-based pattern files in CSV format.[3] Advanced users may want to customize these pattern files for their own matching projects. This can be done in several ways. Some users may want to use their own

[3] CSV files use encoding WLatin1

obs	stnd_nm	stnd_dba	stnd_fka	stnd_ent
1	ADVANCED ANALOGIC			
2	ANALYSTS INTL			
3	AZTEC IND			
4	BALCHEM			CORP
5	BLUEFLY			
6	CHURCHILL DOWNS			
7	COLOMBIA LAB			
8	COST U LESS STORE			
9	FLUSHING FINANCIAL			
10	RUTHS CHRIS STEAKHOUSE			

Table 5: Standardized output from Table 2

set of pattern files while others may want to run the first pass with the default pattern files and run the second pass with some additional pattern files. The subcommands and their default pattern files are listed in Table 3. Before modifying a pattern file, the users need to understand how these subcommands work and their dependencies on each other. Their sequence is also important because some subcommands and their pattern files are conditional on certain characters being removed or standardized by previous subcommands. We suggest that the users keep the default pattern files in the public theme, and use different themes for a different set of pattern files. The `%stnd_compname` module relies on two parsing modules and seven standardizing modules. Wasi and Flaaen (2015) explain all format requirements for all pattern files. In this paper, we will give more detailed examples but only for two pattern files.

4.1. Examples of pattern files

Example 1: %parsing_namefield module and P10_namecomp_patterns.csv

DBA	DBA
D/B/A	DBA
D.B.A.	DBA
D B A	DBA
T/A	DBA
FKA	FKA
F/K/A	FKA
F.K.A.	FKA
F K A	FKA
FNA	FKA
F/N/A	FKA
F.N.A.	FKA
F N A	FKA
FORMERLY KNOWN AS	FKA
FORMERLY	FKA
AS SUCCESSOR TO	FKA
SUCCESSOR TO	FKA
ATTN	ATTN
C/O	ATTN

Table 6: P10_namecomp_patterns.csv

`%parsing_namefield` is the first step, and it checks whether a given string variable contains more than a single business name. If so, it parses them into separate fields. Some records include a legal name, a trade name and/or a former name in their filing. The content of the default pattern file associated with this command, `P10_namecomp_patterns.csv`, is shown in Table 6. Each row consists of two columns: column 1 is a string pattern to search for (keyword); and column 2 is the associated name component type. For example, for doing-business-as names (DBA) or trade names, the command searches for the string "DBA", "D/B/A", "D.B.A", "D B A" or "T/A". Any text that appears after one of these keywords will be treated as the business DBA name. Applying this command to "FLUSHING FINANCIAL CORP T/A FFC" will split it into "FLUSHING FINANCIAL CORP" and "FFC". [4]

Example 2: %stnd_nesw and P70_stnd_nesw.csv

`%stnd_nesw` is one of the seven standardizing subcommands used. They are all based on word substitution. These subcommands also ensure that the text to be substituted is not a part of a larger string, i.e., the text is surrounded by spaces. The `%stnd_nesw` standardizes directional words appearing in business names by searching for keywords in the pattern file `P70_stnd_nesw.csv`. The content of this pattern file is listed in Table 7. Each row consists of two columns. The first column is the string to be substituted. The second column is the standardized word. For example, "NO", "NOR", "NORTH" will be replaced by "N", but not "NORTHERN ILLINOIS UNIVERSITY".

The users may remove or insert additional rows to these files.

NO	N
NOR	N
NORTH	N
SO	S
SOUTH	S
SOUTN	S
EAST	E
WE	W
WEST	W
WST	W
N E	NE
NORTH EAST	NE
NORTHEAST	NE
N W	NW
NORTH WEST	NW
NORTHWEST	NW
S E	SE
SOUTH EAST	SE
SOUTHEAST	SE
S W	SW
SOUTH WEST	SW
SOUTHWEST	SW

Table 7: P70_std_nesw.csv

4.2. Changing a directory of pattern files

The user can change the path of the pattern file directory using the `pattern_path` and `theme`

[4] We do not see "CORP" in the **stnd_nm** field of Table 4 because in a later step "CORP" is treated as a word indicating an entity type and is parsed into a separate field.

macro variables. For example, if the user wants to further standardize the variable ***stnd_nm*** in the above output with another set of pattern files located in `C:\swell\PatternFiles\stndpatterns\theme\pass2`. The new variable ***stnd_nm2*** is to be added on the same file.

```
libname datastnd "C:\swell\stnd";
%let pattern_path=C:\swell\PatternFiles\stndpatterns;
%let theme=pass2;
%stnd_compname(
datastnd.fileAstnd,
datastnd.fileAstnd,
stnd_nm,
stnd_nm2,
);
```

Note that without specifying new pattern file names, the files located in this theme must have exactly the names listed in Table 3.

4.3. Changing a pattern file name

The default pattern file name setting for individual pattern files may also be changed by setting individual macro variables that correspond to each pattern file. For instance, to use a pattern file called `P60_stnd_numbers_new.csv` rather than the default, `P60_stnd_numbers.csv`, the macro variable `P60` would need to be assigned a value of `P60_stnd_numbers_new.csv`.

```
libname dataraw "C:\swell\raw";
libname datastnd "C:\swell\stnd";
%let pattern_path=C:\swell\PatternFiles\stndpatterns;
%let theme=pass1
%let P60=P60_stnd_numbers_new.csv;
%stnd_compname(
dataraw.fileA,
datastnd.fileAstnd,
name,
stnd_nm,
);
```

The file P60_stnd_numbers_new.csv, in the directory specified by the `pattern_path` and `theme` macro variables, would then be used by the **%stnd_numbers** macro.

5. Discussion

Probabilistic record linkage is commonly used to link records of the same entity over time or across multiple sources. The method heavily relies on approximate string comparator functions, which measure the similarity of two strings. Similarity measures can be inaccurate if records are in different formats and contain unnecessary information. This manuscript explains how our **%stnd_compname** module helps researchers properly prepare data files containing business names before probabilistically linking them. Advanced users can also customize the rules used in the default pattern files for their projects. It should be noted that our default pattern files were developed for U.S. business names. For records including businesses in other English-speaking countries, some common words or other entity types should be added to the pattern files. For example, “Public Limited Corporation” (or “PLC”) is common in the U.K. “Proprietary Limited Company” (or “Pty. Ltd”) is common in Australia. We also recommend that users standardize long words by making them shorter (e.g., changing “Professional Corporation” to “PC”, not “PC” to “Professional Corporation” because “PC” may refer to “Personal Computer”). Lastly, users should carefully examine changes to standardized output when adding rules because they sometimes have unintended consequences. For instance, while standardizing a state abbreviation such as “WASH” to “WA” seems to make sense, this rule will also change a business named “SUPER CAR WASH” to “SUPER CAR WA”.

References

Christen P (2012). *Data Matching: Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection.* Springer.

Wasi N, Flaaen A (2015). “Record linkage using Stata: Preprocessing, linking, and reviewing utilities.” *Stata Journal*, **15**, 672–697.

Winkler WE (2006). *Overview of record linkage and current research directions.* Bureau of the Census.

Acknowledgement

Any opinions and conclusions expressed herein are those of the authors and do not necessarily represent the views of the U.S. Census Bureau. This paper does not use any confidential Census Bureau information. This project is a part of work by The Summer Working-group for Employer List Linking (SWELL)– a collaboration between researchers at the U.S. Census Bureau, University of Michigan, and Cornell University. We gratefully acknowledge support by the Alfred P. Sloan Foundation for the University of Michigan’s Census-Enhanced HRS (CenHRS) Project and by the National Science Foundation (SES1131500) for the University of Michigan node of the NSF-Census Research Network (NCRN).

Affiliation:

Nada Wasi
University of Michigan
E-mail: nwasi@umich.edu

Ann Rodgers
University of Michigan
E-mail: anrodger@umich.edu

Kristin McCue
U.S. Census Bureau
E-mail: kristin.mccue@census.gov