Paper 3675-2019

A scoring algorithm to validate hashed ID linkages using SPEDIS and COMPGED

Nichole Sanders, PhD, Arkansas Center for Health Improvement

ABSTRACT

The Arkansas All-Payers Claims Database (APCD) contains claims from multiple payer sources, as well as other available data sources. It does not have direct personal identifiers (DPI), such as name or date of birth (DOB). Instead it contains a hashed version of the concatenated last name and DOB which allows linkage of individuals across payers and other data sources in the APCD. An expected match rate was derived for matching birth and death certificates to claims in the APCD. DPI are contained on birth and death records, as well as Medicaid claims in the Health Data Initiative (HDI) data warehouse that is housed at the Arkansas Center for Health Improvement (ACHI). Hashed IDs were created for individuals contained in these data sources.

To calculate a match rate, a denominator of "true linkages" was determined using the known DPI contained on the HDI data sources. The match scoring algorithm compared first and last names, DOB, and gender between these sources. The SPEDIS and COMPGED functions were used to compare first and last names and because lower SPEDIS and COMPGED scores represent good matches, exact matches for gender and DOB were set to 0. A rubric was developed based on the scores and visual inspection to determine the true linkages. The numerator consisted of linkages that had a single hashed ID for each record linkage. This rate from this numerator and denominator gave us an estimate of true linkages we could expect when linking hashed IDs in the APCD.

INTRODUCTION

An APCD is a large-scale database that systematically collects healthcare data from a variety of healthcare payer sources. APCDs are tools that can be used to support state health system transformation efforts, increase healthcare transparency, and better understand and address healthcare costs, quality, and utilization. The Arkansas Center for Health Improvement (ACHI), in partnership with Arkansas Insurance Department (AID), has developed the Arkansas APCD, which contains healthcare data from a variety of sources. Direct personal identifiers, or DPI, are not collected (e.g., names, addresses, and Social Security Numbers). However, in order to maximize the use of such a large database, the ability to link data from a single individual across sources is a necessity. In the Arkansas APCD, this linkage is possible using a securely hashed version of the last name, concatenated with the date of birth of an individual. This "hashed ID" makes it possible to link individuals across different payer entities or with other data sources that have been stripped of DPIs, such as birth or death certificates. There are some known limitations to using the hashed ID:

- Collisions can occur where one hashed ID is the same for more than one individual. In other studies we have shown this to be about 2-3 percent of the population, depending on the overall size of the population of interest.
- If the last name is recorded differently across sources, the hashed ID will be different (e.g., when someone changes their last name).

These limitations raise the question: How successful will we, or anyone else subscribing to the Arkansas APCD, be when using the hashed ID to link individuals across the various data

sources available? The opportunity to answer this question became available while studying infant mortality using the Arkansas Health Data Initiative (HDI) data warehouse.

The Arkansas HDI is a comprehensive system that integrates data sets from a variety of state sources and is used to inform a comprehensive understanding of public health in Arkansas. In 2003, the Arkansas General Assembly passed Act 1035, which authorized ACHI to maintain the HDI to support work on data-driven health policy issues — for example, studying healthcare utilization, including prenatally, prior to infant mortality. This type of study is possible using the HDI because it contains the Arkansas Health Department Birth and Death Certificates and the Arkansas Department of Health Services Medicaid claims, along with their corresponding DPI. Besides providing an opportunity to identify areas for possible intervention to decrease infant mortality, this study also gave us the opportunity to determine an expected match rate when using de-identified data in the APCD. We took advantage of this opportunity by matching records across death certificates, birth certificates, and Medicaid enrollment files for both infants and mothers using DPI. Then we evaluated whether or not the created hashed ID matched for each record. During our first attempt to match infant DPI across birth and death certificates in the HDI, we discovered how frequently names can be spelled differently or, especially in the case of last names, how they can completely change in the course of one year. For this reason, we developed a scoring algorithm using both the SPEDIS and COMPGED functions to produce sub-scores based on first and last names.

The use of SPEDIS and COMPGED functions to make fuzzy matches between text variables is not a novel idea. These functions have been applied singularly or in combination with each other or other functions to many situations (Schreier, 2004; Dunham, 2016; Cadieux and Bretheim, 2014; Mullins, 2013; and many others). The COMPGED function [syntax: COMPGED(*string-1, string-2 <, cutoff*><,*modifiers*>)] returns values based on the edit distance between two strings using a generalization of the Levenshtein edit distance. At its most basic level, the function works by assigning a certain score for each type of edit (e.g., deletions, insertions, replacements, etc.) required to transform *string-1* into *string-2*. The function returns values in multiples of 10. The SPEDIS function [syntax: SPEDIS(*query, keyword*)] determines the likelihood of two words matching. This function calculates a "cost" for converting the keyword to the query, but the final score that is returned by the function is also dependent on the length of the query. For instance, a change in the first character costs 200, if the query is only two characters long, then the score will be 100, but if the query is 20 characters long the final score will be 10. For both functions, the lower the returned value the better the match, and 0 indicates a perfect match.

Many SAS users recommend using the COMPGED function instead of the SPEDIS function because it requires less processing time; however, in our experience and others (see Cadieux and Bretheim, 2014), it is difficult to set a cutoff value to indicate consistent matches. As discussed in Cadieux and Bretheim's (2014) paper, it was difficult to find a COMPGED score that included objectively correct matches without including many matches that were too loose. They chose to include a SPEDIS score, as well, which made it possible to fine tune their matches. Ultimately, their goal was to include more matches — which included imperfect matches — because they were using it to match postal addresses. Their paper compares the match rate outcomes of different cut-off scores while they were calibrating their scoring rubric. In this application our goal was to produce as many one-to-one matches for individuals as possible using SAS®, since all individuals with multiple record matches would have to be examined manually. For this paper we will only include our final scoring criteria.

APPROACH

We believe it is important to preface our approach with a disclaimer that this paper came about as an ancillary concern to our primary project goal, which was to develop healthcare utilization profiles of deceased infants. We were aware that we needed to link the records in the HDI using DPI, and we would have needed our scoring algorithm for that purpose alone. But we also want to share our findings on the validity of our hashed IDs, which may be beneficial to other users with similar data structures.

INITIAL PLANNING AN DEVELOPMENT

For our infant mortality study, we extracted death records for all infants that died between 2013 and 2016 based on the age at death on the death certificates (n=1,156). Our approach to determine healthcare utilization for each deceased infant and the prenatal care of the mother is outlined in Figure 1, where the circled numbers represent a linkage that was made. Table 1 lists the DPI for each data source, according to the person to which the linkage is related. Our initial extraction of the birth certificates for the infants identified in the death certificates attempted to use only the DPI listed in Table 1 where the first and last names sounded alike (using =*), the dates of birth were the same, and the sex was the same. Unfortunately, and in hindsight, unsurprisingly, this resulted in very few records being extracted. So to increase the number of potential true matches we added two additional criteria. One was to create the hashed ID before identifying the true matches and the other was to use a non-identifiable ID, or PID, that already existed in the HDI.

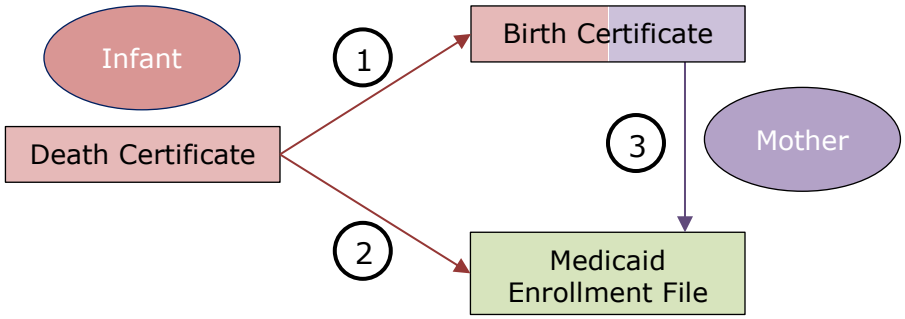

Figure 1. Flow diagram of the process of linking across data sources. Number in circles indicate a linkage that was created.

HDI Data Source	Infant	Mother
Death Certificate	First and Last Name, DOB and Sex	N/A
Birth Certificate		First and Last Name, DOB and SSN
Medicaid Enrollment		

Table 1. DPI Used to Link Across Data Sources

Initially, we planned to apply the APCD hashing methodology only to the records that were linked as true matches based on the DPI. However, we decided to initiate the hashing process prior to finding the true matches so that we could use the limitations of the hashed ID to our advantage. The limitation that was particularly helpful was that some hashed IDs are assigned to multiple individuals, which is frequently is the case with twins. For the death certificates we only hashed the 1,156 infants that we identified. For the birth certificates, on

the other hand, we created hashed IDs for infant and mother on all records from 2012 to 2016. Lastly, we created hashed IDs for all individuals for all years available in the Medicaid enrollment file.

We also took advantage of a non-identifiable personal ID, or PID, that is applied to all records in the HDI. The PID is created using all available DPI from every available HDI data source. The more records that exist for an individual, the stronger the PID becomes. The PID is updated yearly after all state entities have submitted their data, so the DPI on every new record for an existing individual is added to the knowledge repository and is used to generate new PIDs. When there are many records for an individual in the knowledge repository, individuals who have changed their names on different records can still be linked. PIDs that are created for individuals who have only a couple of records in the HDI — primarily the case for deaths of infants — typically have derived PIDs that do not match to any other record. With this limitation in mind we did not plan on using the PID, especially when we attempted to match death certificates to birth certificates. However, after such a limited number of records with our first attempt, we realized that it would not hinder us to include the PID.

Our final criteria to create our subset of possible matched records included the DPI criteria listed above, **OR** a PID match, **OR** a matching hashed ID. For the 1,156 infants we identified, we found 1,408 possible birth certificate matches. This gave us something to work with (spoiler alert: after using our scoring algorithm we found that we truly matched 91 percent of the death certificates to the birth certificates, which according to the Arkansas Department of Health was very reasonable). We also applied this criteria to linkage 2 shown in Figure 1. For the 1,156 infant death certificates, we pulled a subset of 1,005 possible linkages to Medicaid enrollment records. Of the 1,049 birth records that were true matches, we made 813 possible birth certificate mother-to-Medicaid-enrollment-record linkages.

DEVELOPING OUR SCORING ALGORITHM

At last we had three very manageable subsets of possible linkages, so it was time to develop our scoring algorithm to narrow down the list of questionable linkages that would require a visual inspection. For each linkage we removed exact matches, which were defined as exact matches between the DPI for each of the linkages listed in Table 1. We expected the highest number of true matches between death certificates and birth certificates, because both data sets were submitted by the Arkansas Department of Health. So, we started with these linkages. From our death-certificate-to-birth-certificate linkage, we immediately removed the exact matches (985), which were defined as having an exact match for first and last names, date of birth, and sex. We were also able to exclude some clear mismatches after a brief visual inspection. This left us with 306 linkages that would have needed a thorough visual inspection without our scoring algorithm.

SUB-SCORES

Initially, we based our algorithm solely on the SPEDIS and COMPGED return values for first and last names between death and birth records. We created separate SPEDIS and COMPGED sub-scores by adding the output for first and last names, and we created a total score by adding those two sub-scores. We flagged linkages that had SPEDIS sub-scores less than 50 or COMPGED scores of less than 200. When we reviewed the output, we discovered that by using these sub-score cut-offs to indicate possible matches, we would have been including matches with either mismatched sex or mismatched dates of birth. When we tried flagging only those that also had exact matches of date of birth and sex, we ended up excluding linkages where the sex or date of birth might have just been a typo — but based on all the other information in the record they should would have been assigned as a true match on a visual inspection. Because the COMPGED sub-score could be in the hundreds,

we created a sub-score for sex with 0 for an exact match or 100 for a non-match. Originally, we thought that differences in dates of birth might be a typo (perhaps off by a day or two), but in reality, there was no discernable pattern so we did not include date of birth in our final algorithm. The application of SPEDIS and COMPGED to develop sub-scores and the creation of the sex sub-scores are as follows:

```
proc sql;
  create table comparison_table as
    select distinct death_pid
    , birth_pid, death_apcd_hash_id, birth_apcd_hash_id
    , death_fname, birth_fname
    , spedis(death_fname, birth_fname) as fname_spedis_sc
    , compged(death_fname ,birth_fname) as fname_comp_sc
    , death_lname, birth_lname
    , spedis(death_lname, birth_lname) as lname_spedis_sc
    , compged(death_lname, birth_lname) as lname_comp_sc
    , death_sex, birth_sex
    , death_dob, birth_dob
    , case
        when death_sex = birth_sex  then 0
        else 100
    end as sex_score
  from bddiff0;
quit;
```

Initially, we tried to use only a total score that was a sum of all sub-scores. While we found that total scores under 200 were solid matches and scores over 300 clearly were not matches, we couldn't consistently determine whether there was a match or not between scores of 200 to 300. For this reason, we re-introduced the use of separate SPEDIS and COMPGED sub-scores that consisted of the returned value for each function on the first and last names. We also included the sex sub-score in each of these. The final scoring was completed using the following code:

```
proc sql;
  create table comp_table_scored as
    select *
    ,sum(fname_spedis_sc,fname_comp_sc,lname_spedis_sc,lname_comp_sc
    ,sex_score) as total_scores
    ,sum(fname_spedis_sc,lname_spedis_sc,sex_score) as sped_scores
    ,sum(fname_comp_sc,lname_comp_sc,sex_score) as comp_scores
      from comparison_table
        order by total_scores;
quit;
```

SCORING ALGORITHM

Our final scoring algorithm assigned a true match, designated by the variable "group" being equal to "a" whenever the total score was less than 200. When the score was between 201 and 300, the match was dependent on either the SPEDIS sub-score being less than 25 or the COMPGED score being less than 200. The SPEDIS sub-score was checked first for a few reasons. The first was because when we visually inspected the output we realized there were several cases of hyphenated names in some records and not in their likely matches. Hyphenated names can be costlier in SPEDIS than in COMPGED, depending on which variable is the *query* in the SPEDIS function. Because if it was the most restrictive and we knew that if a SPEDIS sub-score was less than 25, then either the total score was much higher, not because of a mismatch of sex, but likely because the first letter of one of the

names was double-typed or had some other error that was very costly in COMPGED. But the rest of the information indicated a true match. If the SPEDIS score was greater than 25, but the COMPGED score was less than 200 there was likely a difference in the sex variable and everything else matched. Finally, records that did not meet the criteria were assigned to group "b." The code below shows the assignment of each record to the match ("a") or non-match ("b") groups:

```
proc sql;
   create table comp_table_flagged as
     select *
       ,case
         when total_scores < 200 then "a"
         when total_scores between 201 and 300 then
           case
             when sped_scores < 25 then "a"
             when comp_scores < 200 then "a"
           end
         else "b"
       end as group
         from comp_table_scored
           order by death_lname;
     quit;
```

The records and their assignments were scanned to see how appropriately the algorithm assigned matches or non-matches, and we found very few cases when the algorithm was incorrect. Very few cases required us to visually inspect all of the variables of the record, but these cases were quickly identified thanks to the scoring algorithm. This was especially important when we were working with the Medicaid enrollment file since the data sources were from two separate state agencies.

HASHED ID VALIDATION

Our scoring algorithm helped us identify our true matches more quickly than visually inspecting all possible matches. Table 2 shows the number of true matches that we found across the three linkages (identified as Step A). These were our "truth" and denominator. We then excluded linkages using our typical methodology when linking records with hashed IDs. When we extract records using a hashed ID, we always concatenate with the sex or gender field to protect us from non-identical twins or other multiple births. So we excluded records that did not have matching concatenated hashed ID and sex (Step B). After we extract data with a concatenated hashed ID and sex, we exclude any hashed ID and sex combinations that have more than one individual's record associated with it. We typically determine this if the records contain conflicting information — for example, if a single hashed ID has more than one insurance IDs for the same payer (Step C). This leaves us with the records we would typically include in an analysis (Step D), which is our numerator.

Step		Infant		Mother
		Death ↕ Birth	Death ↕ Medicaid	Birth ↕ Medicaid
A	Truth (individuals linked with DPI)	1,049	720	764
Linkages *excluded* from the Truth				
B	Hashed ID (+ sex) ≠ between records	35	24	88
C	Colliding Hashed IDs	134	94	157
D	Matching, non-colliding Hashed IDs	880	602	519
	Rate of truly linked records that would have been found using our typical methodology	83.9%	83.6%	67.9%

Table 2. The rate of true matches that we would have found using the hashed ID.

CONCLUSION

Based on another study looking at collision rate that we completed, we initially expected our rate of truth to be in the upper 90s. That study aligned with the expected rate of same sex twin births in Arkansas which is not quite 2 percent (CDC Wonder, 2019). However, after further consideration, it is not surprising that we had higher collision rates — greater than 10 percent — because this particular study population is going to have a higher rate of collisions due to its higher rate of twins and multiple birth siblings (being part of a multiple birth increases the risk of infant mortality). We had hoped the birth-certificate-to-mother linkage would have been higher, but we really had no way of knowing prior to this study because it was a first look. Ultimately, this tells us that any study using the APCD that requires linking women across data sources needs to be very aware that they could lose at least 10 percent because of different names (see Step B).

Regarding the use of the scoring algorithm using the SPEDIS and COMPGED functions, we have already started applying that to other linkages. Particularly when studying provider information in the APCD. We are working on building a master provider table that will draw information from a variety of sources. Not all sources include the providers National Provider Identifier, however, so in those cases we use provider names. Our scoring algorithm has also been adapted slightly to apply to matching addresses across claim sources which has proven very effective.

REFERENCES

Centers for Disease Control and Prevention. CDC Wonder, Natality Information, Live Births. Available at https://wonder.cdc.gov/natality.html

Cadieux, Richard and Bretheim, Daniel. 2014. "Matching rules: Too loose, to tight, or just right?" Proceedings of the 2014 SAS Global Forum. Available at http://support.sas.com/resources/papers/proceedings14/1674-2014.pdf

Mullins, Barry. 2013. "The complexities of an address." South Central SAS Users Group. Available at http://www.scsug.org/wp-content/uploads/2013/11/The-Complexities-of-an-Address-Barry-Mullins.pdf

Schreier, Howard. 2004. "Using edit-distance functions to identify "similar" e-mail addresses." Proceedings of the Twenty-Ninth Annual SAS Users Group International Conference. Available at http://www2.sas.com/proceedings/sugi29/073-29.pdf

ACKNOWLEDGMENTS

I'm thankful to my colleagues at ACHI including Mike Motley, MPH, for leading the Health Policy portion of this study and helping me untangle and explain the nuances of our data to the more right-brained individuals on our team; Brady Rice, the keeper of the HDI; and Tim Holder, our technical editor who frequently saves me from run-on sentences similar to this one. I'm also thankful to Pedro Ramos, PhD, for reviewing my paper as a SAS master when I gave him very short notice.

CONTACT INFORMATION

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

Nichole Sanders, PhD
ACHI
1401 W. Capitol Avenue
Suite 300, Victory Building
Little Rock, AR 72201
501-526-2244
nichole@achi.net
achi.net