PharmaSUG 2018 – Paper BB-05

A Conceptual Strategy and Macro Approach for Partial Date Handling in Data De-Identification

Kelsey Reppert, PPD, Wilmington, NC
Amy Caison, PPD, Wilmington, NC

ABSTRACT

To address the privacy concerns of individual patient health data, regulatory authorities in both the US and Europe have established benchmark requirements for the protection of patient privacy. This is ever more critical now that the European Medicines Agency Policy 0070 is being phased in which will, when fully implemented, require the public disclosure of anonymized patient-level clinical trial data. Data de-identification is the process of anonymizing personal information while maintaining the scientific value of that data. Dates are notable indirect identifiers which, when combined with other potential identifiers, can be used to identify a subject. To counter this adverse outcome and enable the public sharing of clinical trial data, this paper presents an approach to de-identifying all dates, both full and partial, using a random offset methodology with a specific strategy for handling partial dates in a manner that preserves the scientific value of the partial date information in the de-identified dataset and limits the clustering of imputed dates. Since dates are ubiquitous in all domains of clinical trial data, our partial date-handling strategy has been developed as a macro so that dates in a database can be consistently and efficiently de-identified.

INTRODUCTION

The ever-increasing calls from regulatory agencies, the media, and the public for greater clinical trial data transparency has initiated a paradigm shift in the pharmaceutical industry toward greater transparency of trial conduct and public disclosure of trial data and results. In recent years, several initiatives have emerged to provide greater access to clinical trial data such as Project Data Sphere and ClinicalStudyDataRequest.com, both being joint efforts by consortia of pharmaceutical companies. Increasing the pressure for public sharing of clinical trial data, the European Medicines Agency adopted Policy 0070 in October 2014, which requires the publication of clinical trial data for all marketing authorization applications (MAA). This development has several beneficial consequences: duplication of clinical trials can be more easily avoided, the public will likely have more trust in the industry's findings, and researchers can re-assess clinical data. Currently in its first phase of implementation, Policy 0070 requires that only clinical study reports from MAAs are made public but with subsequent phases of implementation, all clinical trial data will be required to be publicly accessible. Policy 0070 requires that personally identifiable data be protected and that all data shared through Policy 0070 be sufficiently de-identified as to protect the identities of subjects participating in the clinical trials. With the increased demand for clinical trial data from both external researchers, the media, and regulatory authorities, the importance of protecting potentially identifying subject-level data is critical. This process of data anonymization, transforms data such that identification of any subject in the data is not likely to take place while simultaneously preserving the scientific utility of clinical trial data.

To address the concerns of privacy of individual patient health data, regulatory authorities in both the US and Europe have established benchmark requirements for protection of patient privacy. In the US, the Health Insurance Portability and Accountability Act (HIPAA) (1996) outlines two methods for de-identification of patient-level data. In the first approach, generally referred to as the Safe Harbor Method, the HIPAA regulations specify a list of 18 data points, including all dates, that must be categorically removed from all data to ensure adequate de-identification. However, removal of such a comprehensive list of data points may compromise the scientific viability of the de-identified data. To address these concerns the HIPAA legislation acknowledges that de-identification based on statistical principles through expert determination can also ensure adequate levels of patient privacy protection while maintaining the scientific and statistical value of the data. Similar guidelines are endorsed by the EMA in its external

guidance document on implementing Policy 0070 (EMA, 2017). While this guidance does not recommend a specific methodology, it does emphasize the importance of anonymizing data while maximizing the scientific utility of the data and references a paper by Hrynaszkiewicz, Norton, Vikers, and Altman (2010) as providing the guidelines for the minimum standard of adequate de-identification.

Hrynaszkiewicz, et al. (2010) distinguish between data that can provide direct and indirect identification of subjects. In their EMA-endorsed approach, they require the complete removal of 14 direct identifiers including variables such as name, initials, address, telephone or other contact information, among others. Further, they recommend assessing the risk of re-identifying subjects on any dataset containing three or more indirect identifiers. Indirect identifiers include, but are not limited to, instances of rare diseases, locations, and date values. By combining multiple indirect identifiers, it is reasonably possible to re-identify a subject. Thus, de-identification of patient data typically includes, but is not limited to, the following basic data transformations:

- Replacement of subject and site identifiers with random, internally consistent values
- Blanking of certain free-text variables which include potentially identifying information such as dates, names, locations, medication or device numbers (e.g., kit numbers), or verbatim terms or events
- Conversion of continuous identifying variables (e.g., AGE) to specified categorical groups (e.g., 5-year age bands, programmatically-determined quartiles or quintiles)
- Collapsing of smaller variable categories into larger, more general categories (e.g., collapsing individual countries in Europe into larger "Eastern Europe" and "Western Europe" groups or collapsing RACE categories with small counts into larger, more general groupings) [1]
- Adjustment or removal of potentially identifying personal characteristics such as heights, weights, or BMIs which are statistical outliers
- Dropping of certain coded terms or events that are deemed rare and would increase the likelihood of re-identification of a subject.
- Intentionally introducing noise into the data
- Where necessary to preserve the scientific value of the data, manual suppression of potentially de-identifying data within free-text fields can be implemented

De-identification of dates is a particularly important element of de-identification since they are so ubiquitous in clinical trial data and they can be used to place subjects in particular locations and in specific health states at exact times and as a result, potentially lead to identification of the subject. However, the de-identification of the dates in such a way as to preserve the temporal relationships inherent in the data while still maintaining patient anonymity is critical. There are two main strategies to accomplish date de-identification while retaining these temporal relationships: converting all dates into a duration from a specified reference date (i.e., date of first dose) and applying a subject-specific random offset to all dates (Hrynaszkiewicz, et al., 2010, pg 2). Both strategies for handling dates relies on a numeric transformation of the date to achieve the de-identification. As is common with clinical trial data, we often encounter partial dates which presents a challenge for de-identification since they cannot be precisely represented as a numeric. While it is certainly possible to handle all full dates as specified and blank out any partial dates, this does not preserve as much information as was contained in the original data. The best approach would be to algorithmically de-identify all dates, even those with missing components, but simply imputing the missing date parts prior to applying the offset won't suffice since care must be taken to avoid clustering of the data around particular dates. Such clustering resulting from date imputation, can potentially provide the means to infer a subject's offset and subsequently unravel the de-identification. To counter this potential adverse situation, this paper presents an approach to de-identifying all dates using the random offset methodology with a specific strategy for handling partial dates in a manner that preserves the scientific value of the partial date information in the de-identified

[1] In addition, it is important to also look at cross-tabulations of variables which, when combined, could lead to small numbers in certain categories and apply a grouping as needed. For example, assessing frequencies of gender by race can result in the need to apply further grouping to ensure sufficient de-identification of subjects.

dataset and limits the clustering of imputed dates (e.g., numerous data points having an imputed day/month of 01Jan). Since dates are ubiquitous in all domains of clinical trial data, our partial date-handling strategy has been developed as a macro so that all dates can be easily and consistently de-identified throughout a database. For actual trial work, we have independently-developed production and validation macros for this task, though only one side is presented here to illustrate the conceptual approach to partial-date de-identification.

METHODOLOGY

The partial date de-identification strategy first begins with the creation of a subject-specific random offset. Once a random number of days for offsetting has been assigned to each subject, the process of de-identifying dates can begin. If the date is complete, one simply increments the date by the random offset. If the date is partial and only has a year and month, one must first impute a temporary partial date, then apply the random offset to the imputed date, then parse the offset back to just present an offset year and month. If the date is partial and only has a year, after imputing to the first day of the year, we then apply the offset and parse it back down to an offset year. One must convert the random offset from days into years and then offset the partial date. We will step through the process in more detail below and discuss examples of each situation.

CREATING AN OFFSET

To generate our random offsets, we used PROC PLAN to produce a dataset of random values, one for each subject. This unique offset is then added to each date for that subject. This way, each date in a subject's data is shifted by the same amount and the relative difference between dates within a given subject's data is preserved. For our implementation, we wished to put some restrictions on the possible random values we generated:

1. No date should be shifted to before the start of the study.
2. No date should be shifted beyond the end date of the study.
3. No date should be shifted by more than ±180 days.

The first two restrictions are put in place since the start and end dates of studies are publicly available and it would not make sense for date values within the final de-identified data set to fall outside of the start and end dates. For example, if a clinical trial ran from 2010 to 2015, it would be nonsensical to offset a patient's date of first dose to some day in 1970 or a subject's date of last dose to some day in 2050.

The third restriction exists to maintain offset dates somewhat close to the true dates while still introducing enough noise to reduce the chance an adversary could identify subjects. We do this because researchers may need to relate the de-identified dates to real world events or other occurrences in the pharmaceutical industry. For example, maintaining de-identified dates relatively close to the original dates would be important in an asthma study where events such as volcanic eruptions or other environmental circumstances that affect asthma suffers could be useful for explaining some poor responses from subjects in the affected region. In addition, as standard treatment practices evolve, ensuring study dates are reflective of the general time in which they were conducted in would enable better comparisons of results to other studies in meta analyses. Note that these three restrictions are optional can be modified to suit specific de-identification implementation situations. However, we recommend creating a pre-specified interval of values from which to generate offset values so the de-identified dates stay within a reasonable time frame from the true date value.

We used the following algorithm to create an interval from which to choose our random offset values:

- min = Study Start Date – Subject Enrollment Date
- max = Study End Date – Subject End of Study Date
- u = Random Number in the Interval (0, 1)
- offset = min + floor((max – min) * u)

This approach guarantees that each offset date will follow the three restrictions. This algorithm should be run for each subject so a different offset is generated for each. Note that MAX ≥ 0 because the subject's end date should fall on or before the study end date and MIN ≤ 0 because the subject's start date should fall on or after the study start date. Once the offset values have been generated, the application of the

offset to de-identify the dates can begin.

MACRO PARAMETERS

Our date de-identification macro works for full dates as well as partial dates and uses the following parameters:

- input_date: name of the date variable that will be de-identified
- output_date: name of the date variable created by the macro that contains the resulting de-identified date (can be set to input_date)
- offset: the number of days used to offset input_date

Below is our date de-identification macro:

```
%macro offset_date(input_date, output_date, offset);

      *** If input date is in the form yyyymmddThh:mm ;
      if length(&input_date) = 16 then do;
            numeric_date = input(scan(&input_date,1,"T"), yymmdd10.);
            offset_date = intnx('day', numeric_date, &offset);
            &output_date  = put(offset_date,yymmdd10.) ||"T"|| scan(&input_date,2,"T");
      end;
      *** If input date is in the form yyyymmdd ;
      else if length(&input_date) = 10 then do;
            numeric_date = input(scan(&input_date,1,"T"), yymmdd10.);
            offset_date = intnx('day', numeric_date, &offset);
            &output_date  = put(offset_date,yymmdd10.);
      end;

      *** If input date is in the form yyyymm;
      else if length(&input_date) = 7 then do;
            numeric_date = input(compress(scan(&input_date,1,"T")||"-01"), yymmdd10.);
            offset_date = intnx('day', numeric_date, &offset);
            &output_date  = substr(put(offset_date,yymmdd10.),1,7);
      end;

      *** If input date is in the form yyyy ;
      else if length(&input_date) = 4 then do;
            numeric_date = input(compress(scan(&input_date,1,"T")||"-01-01"), yymmdd10.);
            offset_date = intnx('day', numeric_date, &offset);
            &output_date  = substr(put(offset_date,yymmdd10.),1,4);
      end;


      *** If the input date is missing, then set the output date to missing ;
      else if &input_date = '' then &output_date = '';

      drop numeric_date offset_date;

%mend offset_date;
```

This macro assumes that input_date is a character ISO8601-formatted date as you would find in CDISC-compliant datasets and works by considering the length of each observation in the date variable. This way, the macro will handle different kinds of partial dates that can occur in ISO8601 format in addition to complete and missing dates. Each section of the macro will determine the form of the observation within the date variable by testing the length of the observation. The de-identification is done in three steps:

1. If the date is partial, then impute the date by setting to the first day of the month and/or year, as appropriate for the situation, and create a numeric version of the date.
2. Use the INTNX function to increment input_date by the specified number of days.
3. Set output_date to the offset date after truncating it to the form of input_date.

The imputation of the missing date parts (by adding 01 or 01JAN) is an intermediate step to facilitate the

application of the offset, and would not survive in the final presentation of the data since those parts are then dropped to return the de-identified date to the partial state it was in before the offset was applied. Thus, the choice of 01 or 01JAN as imputed date parts is arbitrary and could be changed as desired, though whatever the choice for imputed date parts might be, it should be applied consistently across subjects. Note that if input_date contains time information, we use the SCAN function again to parse out the time portion, to be added be added back once the date portion is shifted by the subject's offset. This approach to de-identifying partial dates can also be extended to consider other kinds of partial dates and other kinds of date formats.

EXAMPLE

The PROC SQL statements shown below simply create an example data set containing two date variables to be de-identified and a variable that specifies how many days each date variable will be offset. The offset numbers were selected to illustrate how the macro works; however, in an actual de-identification, this offset variable should be generated randomly through an algorithmic strategy such as that presented above.

Below is code that will generate a dataset which we will use to demonstrate how our macro works:

```
proc sql;
    create table example_input (
           example_date_1 char(20),
           example_date_2 char(20),
           example_offset num
    );

    insert into example_input (example_date_1,example_date_2,example_offset)
            values ("2015-12-14T09:26","1970-01-05T17:03",22)
            values ("2015-12-14","1970-01-05",-10)
            values ("2015-12","1970-01",164)
            values ("2015","1970",801)
            values ("2015-12","1970-01",17)
            values ("2015","1970",72)
            values ("","",377)
           values ("2015-12","2017-01-29",5)
     ;
quit;
```

The below code shows an example of how to use our macro.

```
data example_output;
    set example_input;

    %offset_date(example_date_1, output_date_1, example_offset);
    %offset_date(example_date_2, output_date_2, example_offset);
run;
```

Our macro is designed to run inside a data step so that the macro can be called several times to de-identify different date variables. Note that the input parameter to the macro must specify a variable that exists in the dataset referenced in the encompassing data step. In this example, the variables example_date_1, example_date_2, and example_offset are all in the dataset example_input. The variables output_date_1 and output_date_1 are not in the example_input dataset, but will be created in the example_output dataset shown in Table 1.

DISCUSSION

The examples shown in Table 1 illustrate how this de-identification strategy and corresponding macro addresses key situations commonly found in clinical trial data and successfully de-identifies dates of varying formats.

	example_date_1	example_date_2	example_offset	output_date_1	output_date_2
1	2015-12-14T09:26	1970-01-05T17:03	22	2016-01-05T09:26	1970-01-27T17:03
2	2015-12-14	1970-01-05	-10	2015-12-04	1969-12-26
3	2015-12	1970-01	164	2016-05	1970-06
4	2015	1970	801	2017	1972
5	2015-12	1970-01	17	2015-12	1970-01
6	2015	1970	72	2015	1970
7			377		
8	2015-12	2017-01-29	5	2015-12	2017-02-03

Table 1. Example Output Dataset

EXAMPLE 1: FULL DATE WITH TIME

In row 1 of Table 1, both example dates have time information and since we know that the dates are ISO8601-formatted, we know that the length of the values will be 16. Therefore, the corresponding if-then statement in the macro will calculate the numeric portion of the date part. In this case, example_date_1 will be converted to 20436 and example_date _2 will be converted to 3657. Using the INTNX function, we increment the numeric date value by the offset (22) to get 20458 and 3679, respectively. Finally, we convert these numeric date values and append the time to the end to get the output_date values shown above in ISO8601 format.

EXAMPLE 2: FULL DATE, NEGATIVE OFFSET

In row 2 of Table 1, we only have date information in the example date variables. Therefore, the process is the same as above except we don't have to consider any time information. Since the offset is -10 in this row, we will decrease the SAS date value by 10 days. Example_date_1 is shifted from 2015-12-14 to 2015-12-04. Example_date_2 is shifted from 1970-01-05 to 1969-12-26. Note that, in this situation, the negative offset results in a change in year in output_date_2. This is acceptable because it maintains the temporal relationship between this subject's dates once all dates for this subject are offset.

EXAMPLE 3: MISSING DAY

In row 3 of Table 1, the example dates are missing day information. To apply the offset of 164 days, we want to offset the date by the approximate equivalent number of months. We can do this by imputing the dates to the first day of the month, applying the offset in days using the process outlined above, and removing the day information to get the final output.

- After imputing 2015-12 to 2015-12-01 and shifting 164 days, we get 2016-05-13. We then drop the day information to get 2016-05.
- After imputing 1970-01 to 1971-01-01 and shifting 164 days, we get 1970-06-14. We then drop the day information to get 1970-06.

EXAMPLE 4: MISSING MONTH AND DAY

In row 4 of Table 1, the example dates only have year information. Therefore, we apply a similar process as described above. However, this time, we impute the dates to the first day of the year and then offset by 801 days.

- After imputing 2015 to 2015-01-01 and shifting 801 days, we get 2017-03-12. We then drop the month and day information to get 2017.

- After imputing 1970 to 1970-01-01 and shifting 801 days, we get 1972-03-12. We then drop the month and day information to get 1972.

EXAMPLE 5: MISSING DAY; OFFSET DOESN'T AFFECT DE-IDENTIFIED DATE

In row 5 of Table 1, the dates will be imputed to the first of the month, shifted up by 17 days to the 18th day of the month, and then the day information is dropped leaving the year and month the same. In this situation, the offset algorithm does not result in a de-identified partial date that is different from the original date. This is acceptable as the temporal relationships for this date, as well as other de-identified dates for this subject that do in fact change from their original value, are maintained since the offset is consistently applied to all dates for this subject (see also Example 8).

EXAMPLE 6: MISSING DAY AND MONTH; OFFSET DOESN'T AFFECT DE-IDENTIFIED DATE

In row 6 of Table 1, the dates will be imputed to the first of January, shifted up by 72 days, and then the day and month information are dropped while leaving the year the same.

EXAMPLE 7: MISSING DATE

In row 7 of Table 1, the dates are missing so no offset is applied.

EXAMPLE 8: DE-IDENTIFICATION OF BOTH PARTIAL AND FULL DATE

As in Example 5, the partial date 2015-12 will not be changed as a result of the de-identification, since it is only being offset by 5 days (which is less than one month). However, the date 2017-01-29 will be offset by 5 days and become 2017-02-03.

CONCLUSION

This approach to de-identifying dates has multiple benefits while still ensuring that we preserve as much scientifically-valuable information as possible. This approach works for all dates including partials and ensures that the de-identified dates to not cluster together, thus unintentionally providing clues that could be used by an adversary to reverse the de-identification. Implementing this method in the form of a macro leverages standard data formats and allows de-identification to be done simply and efficiently for multiple variables.

ACKNOWLEDGEMENTS

The authors would like to thank Jurgen Hummel, Senior Statistical Science Director, Irene Ferreira, Principal Statistical Scientist, and Ashley Kesler, Programming Team Leader, for their time and valuable feedback.

CONTACT INFORMATION

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

Name: Kelsey Reppert
Enterprise: PPD
Address: 929 North Front Street
City, State ZIP: Wilmington, NC 28401
Work Phone: 910-558-6116
E-mail: kelsey.reppert@ppdi.com

Name: Amy Caison
Enterprise: PPD
Address: 929 North Front Street
City, State ZIP: Wilmington, NC 28401
Work Phone: 910-558-5918
E-mail: amy.caison@ppdi.com

REFERENCES

European Medicines Agency (2014) *European Medicines Agency policy on the publication of clinical data for medicinal products for human use.* Retrieved 04Dec2017 from: http://www.ema.europa.eu/docs/en_GB/document_library/Other/2014/10/WC500174796.pdf

European Medicines Agency (2017) *External guidance on the implementation of the European Medicines Agency policy on the publication of clinical data for medicinal products for human use.* Retrieved 04Dec2017 from: http://www.ema.europa.eu/docs/en_GB/document_library/Regulatory_and_procedural_guideline/2017/09/WC500235371.pdf

Health Insurance Portability and Accountability Act (1996). 45 C.F.R §164.514(b) (2003).

Hrynaszkiewicz, I., Norton, M.L., Vikers, A.J., and Altman, D.G. (2010). Preparing raw clinical data for publication: Guidance for journal editors, authors, and peer reviewers. *Trials, 11*(9): 1-5.