PharmaSUG 2017 - Paper QT03

What Are Occurrence Flags Good For Anyway?

Nancy Brucken, inVentiv Health, Ann Arbor, MI

ABSTRACT

The ADaM Structure for Occurrence Data (OCCDS) includes a series of permissible variables known as occurrence flags. These are optional Y/null flags indicating the first occurrence of a particular type of record within a subject. This paper shows how occurrence flags can be used with PROC SQL to easily produce tables summarizing adverse events (AEs) by System Order Class (SOC) and dictionary preferred term.

INTRODUCTION

The ADaM Data Structure for Adverse Event Analysis (ADAE) introduced the concept of occurrence flags as a way to easily identify which in a series of event or intervention records was being used for analysis, and those variables were carried over into OCCDS. As stated in the OCCDS, “Occurrence flags can be used to prepare data for analysis. They are typically created by sorting the data in the required order and then flagging the first treatment emergent record.” Note that in this case, “first” does not necessarily denote chronological order. For example, in studies where the data includes large numbers of records with partial start dates, it may be easier to sort by USUBJID and AESEQ.

COMMONLY-USED OCCURRENCE FLAGS

The OCCDS describes six specific occurrence flags, along with a general placeholder to handle additional analysis needs. This paper will focus on the following three:

Variable Name	Variable Label	Type	Code List / Controlled Terms	Core	CDISC Notes
AOCCFL	1st Occurrence within Subject Flag	Char	Y	Perm	Character indicator for the first occurrence of any event/intervention/finding within the subject. Example derivation: Sort the data in the required order and flag the first treatment emergent record for each subject.
AOCCSFL	1st Occurrence of SOC Flag	Char	Y	Perm	Character indicator for the first occurrence of the system organ class within the subject. Example derivation: Sort the data in the required order and flag the first treatment emergent record for each body system for each subject.
AOCCPFL	1st Occurrence of Preferred Term Flag	Char	Y	Perm	Character indicator for the first occurrence of the preferred term within the subject. Example derivation: Sort the data in the required order and flag the first treatment emergent record for each --DECOD for each subject.

Table 1. Occurrence Flag Variables

These flags are used to mark records to be pulled into a standard table summarizing treatment emergent AEs by SOC and body system, which might look something like this:

System Organ Class **Preferred Term**	**Drug A** **(N=xxx)** **n (%)**	**Drug B** **(N=xxx)** **n (%)**
Subjects reporting at least 1 event	xxx (xx.x)	xxx (xx.x)
Blood and lymphatic system disorders	xxx (xx.x)	xxx (xx.x)
Anemia	xxx (xx.x)	xxx (xx.x)

Table 2. Sample table summarizing AEs by SOC and preferred term

These three occurrence flags directly correspond to the rows shown on the table. The number of records with AOCCFL=’Y’ represents the number of subjects reporting at least 1 AE. The number of records with AOCCSFL=’Y’, when summarized by SOC, represents the number of subjects reporting an AE within that SOC. And finally, the number of records with AOCCPFL=’Y’ represents the number of subjects reporting an AE coded to that preferred term.

USING OCCURRENCE FLAGS

Once the occurrence flags have been created in the analysis dataset, we can take advantage of their presence when programming the table. Note that in the example shown above, the overall count of subjects with AEs is displayed first, followed by the counts of subjects by SOC, and then within SOC, counts of subjects by preferred term. For simplicity, let’s assume both SOCs and preferred terms are displayed in alphabetical order.

The first step is to pre-process the data so as to achieve that sort order. There are many ways to accomplish this, but here is one:

```
data ae;
  set save.adae (keep=usubjid trtemfl aebodsys aedecod trtan trta saffl ao:
                 where=(trtemfl='Y' and saffl='Y'));

  *** Preferred terms;
  output;

  *** SOC terms;
  aedecod = 'AAAAA';
  output;

  *** All subjects;
  aebodsys = 'AAAAA';
  output;
run;
```

This simply makes three copies of each record, replacing the preferred term with ‘AAAAA’ on the second copy, and the SOC with ‘AAAAA’ on the third. Thus, the SOC total counts will appear at the top of the preferred term list, and the overall subject counts will appear as the first SOC record.

Once the dataset is ready, the occurrence flags for all 3 summarization levels can be counted in a single PROC SQL step:

```
proc sql noprint;
  create table aects as
    select aebodsys, aedecod, trtan
         , sum(aoccfl='Y') as nsubfl
         , sum(aoccsfl='Y') as nsocfl
         , sum(aoccpfl='Y') as nptfl
      from ae
      group by aebodsys, aedecod, trtan;
quit;
```

This step produces a dataset containing one record per each SOC, preferred term and treatment arm, with the overall count of subjects with AEs stored as the first record.

Next, the separate count variables can be combined into a single variable for display, and the percentages computed:

```
data aectout (drop=nsubfl nsocfl nptfl);
  set aects;

  if aebodsys='AAAAA' then subct = nsubfl;
    else if aedecod='AAAAA' then subct = nsocfl;
    else subct = npfl;

  *** Add population counts;
  popn = input(symget(cats('n', trtan)), best6.);

  *** Combine counts and percentages;
  length coltxt $ 20;
  if subct > 0 then coltxt = catx(' ', subct,
                                   cats('(',
                                        (put((100 * subct) / popn, 5.1)),')'));
    else coltxt = '0';

run;
```

Finally, using a DOW-Loop, the output dataset can be transposed so the treatment arm counts appear on the same row, and the text in the first column is aligned for display:

```
data final (keep=aebodsys aedecod disterm col1-col2);
  *** Initialize array;
  array cols(*) $20 col1 col2;
  do i=1 to 2;
    cols(i) = '0';
  end;

  do until (last.aedecod);
    set aectout;
    by aebodsys aedecod;

    cols(trtan) = coltxt;
  end;

  *** Format first column;
  length disterm $ 100;

  if aebodsys='AAAAA' then disterm = "Subjects reporting at least 1 event";
    else if aedecod='AAAAA' then disterm = aebodsys;
    else disterm = '  ' || aedecod;

run;
```

CONCLUSION

The occurrence flags described in the OCCDS model can be used to simplify programming of tables summarizing AEs by SOC and preferred term. Additional occurrence flags may be added to analysis datasets to support summarization of AEs by intensity, relationship to study medication, and other attributes; the code above can be easily converted to a macro, with the appropriate occurrence flags represented by macro variables. In addition, similar occurrence flags can be created for summarizing prior and concomitant medications, medical history terms, and other types of data fitting into the OCCDS.

REFERENCES

CDISC Analysis Data Model Team. 2015. “The ADaM Structure for Occurrence Data (OCCDS).” Version 1.0. Available at http://www.cdisc.org/standards/foundational/adam.

CDISC Analysis Data Model Team. 2012. “Analysis Data Model (ADaM) Structure for Adverse Event Analysis.” Version 1.0. Available at http://www.cdisc.org/standards/foundational/adam.

Dorfman, Paul D and Koen Vyverman. 2009. “The DOW-Loop Unrolled.” Available at http://support.sas.com/resources/papers/proceedings09/038-2009.pdf.

ACKNOWLEDGMENTS

Thanks to my colleagues for reviewing this paper.

CONTACT INFORMATION

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

Nancy Brucken
inVentiv Health
(734) 887-0255
Nancy.Brucken@inventivhealth.com