PharmaSUG 2021 - Paper AP-120

Programming Time Varying Concomitant Medication Covariates in NONMEM PopPK Dataset

Dan Xiao, Jing Su, Shuqi Zhao, Merck & Co., Inc., Kenilworth, NJ
Xiang Liu, Incyte Corporation, Wilmington, Delaware

ABSTRACT

Concomitant medication (ConMed) covariates are very important time dependent variables for population PK (PopPK) modeling. Since concomitant medication may change the absorption, distribution, metabolism and excretion of the investigated medicine, it is required to construct the concomitant medication covariates and show the usages of ConMed before or on the dosing day. In this paper, we introduce three different strategies to program the time varying ConMed covariates into NONMEM PopPK dataset.

INTRODUCTION

ConMed data is collected in the CM domain per CDISC standards. In some drug and drug interaction studies, modelers may also consider the investigated drug collected in the EX domain as ConMed. In this paper we will focus only on the CM domain because any ConMed data in the EX domain can be analyzed using a similar method.

There are two different categories of covariates for the NONMEM PopPK dataset: non-time varying or time varying covariates. The ConMed covariate is defined as non-time varying when the flag indicates whether or not the subject took the ConMed at subject level; or defined as time varying when the flag indicates whether or not the subject took the ConMed on each dosing day. The ConMed flag can be based on one drug, like Digoxin, or a class of drugs, like CYP3A4 inhibitors. In this paper, we will discuss how to merge the time varying ConMed covariates with the NONMEM PopPK dataset.

The NONMEM PopPK dataset consists of dosing records and PK observations sorted by dosing days and the nominal time points. Various covariates are then added to the resulting structure as unique variables to complete the NONMEM PopPK dataset.

PROGRAMING CHALLENGES ON HANDLING CONMED DATA

- Missing dates
- Overlapping date entries

We introduce imputation on missing dates first and then present 3 strategies of handling overlapped date entries.

In the real world, the CM domain often has an incomplete start date (CMSTDTC) and end date (CMENDTC). At the beginning of our program, we need to impute the incomplete date following the imputation rules for the CM domain. Briefly, if the start date of the ConMed is completely missing, it will be imputed as the study start date; if both month and day are missing, the start date will be imputed as the first day of that year; if only the day is missing, it will be imputed as the first day of that month of that year. On the other hand, if the end date of the ConMed is completely missing, it will be imputed as the cutoff date of the data base; if both month and day are missing, then the end date will be imputed

as the last day of the year; if the day is missing then the end date will be imputed as the last day of the month of the year.

Below is an example of the SAS code to impute missing dates.

```
DATA CM;
  SET SDTM.CM;
  if LENGTH(compbl(CMSTDTC))=4 then CMSTDTC=compress(CMSTDTC||'-01-01', ' ');
  else if LENGTH(compbl(CMSTDTC))=7 then CMSTDTC=compress(CMSTDTC||'-01', '
');

  if LENGTH(compbl(CMENDTC))=4 then CMENDTC=compress(CMENDTC||'-12-31', ' ');
  else if LENGTH(compbl(CMENDTC))=7 then do;
    if substr(cmendtc, 6, 2) = '12' then CMENDTC=compress(CMENDTC||'-31', '
');
     else cmendtc = put(mdy(put(substr(cmendtc, 6, 2) + 1, z2.), '01',
substr(cmendtc, 1, 4)) - 1, yymmdd10.);
  end;

  if CMSTDTC = ' ' then CMSTDTC = 'Study start date';
  if CMENDTC = ' ' then CMENDTC = 'Cutoff date';

run;
```

We assume missing and partial dates have been imputed on the ADCM dataset according to the rules above, so we will now use ADCM directly to illustrate 3 ways of eliminating overlapping CMSTDT and CMENDT data entries, and derive time varying ConMed covariates onto the NONMEM PopPK dataset.

Table 1: Sample ConMed dataset of CYP3A4 inhibitors

SUBJID	CMDECOD	CMSTDT	CMENDT	CYPIH
1002	AZITHROMYCIN	2018-06-16	2018-06-16	1
1002	TACROLIMUS	2017-08-30	2017-10-14	1
1002	CIPROFLOXACIN	2018-06-16	2018-07-01	2
1002	POSACONAZOLE	2017-08-22	2019-10-10	3
1002	POSACONAZOLE	2017-08-31	2017-09-23	3
1002	VORICONAZOLE	2018-08-25	2019-12-10	3

Note: The CYPIH represents the CYP3A4 inhibitor, it is divided to 3 grades.

STRATEGY 1

For this strategy, we use a SAS function to identify and flag CYP3A4 inhibitors related ConMeds in ADCM, and then eliminate overlapping CMSTDT and CMENDT before merging ConMed covariates onto the NONMEM PopPK dataset.

```
DATA cypih;
  SET ADCM;
  BY USUBJID;
```

```
   if PRXMATCH("m/POSACONAZOLE|…/i", cmdecod) then cypih =3;
   else if PRXMATCH("m/CIPROFLOXACIN|…/i", cmdecod) then cypih =2;
   else if PRXMATCH("m/AZITHROMYCIN|TACROLIMUS|…/i", cmdecod) then
cypih =1;

  IF CYPIH>0;

    CMSTDT=INPUT(CMSTDTC, ??ANYDTDTE21.);
    CMENDT=INPUT(CMENDTC, ??ANYDTDTE21.);

  FORMAT cmstdt cmendt is8601da.;
Run;
```

The %cmset macro is used to ensure all overlapping dates are removed in the case of complicated study scenario. If number of records on the output dataset does not change when the cycle number is increased, all overlapping dates are removed.

```
%macro cmset(datain=, dataout=, cycle=);
%do i=1 %to %eval(&cycle);
proc sort data=&datain;
  by subjid cmstdt;
run;

DATA &dataout;
  SET &datain;
  BY subjid cmstdt;

  RETAIN lagA;
  if first.subjid then lagA= cmendt;

  if lagA<cmstdt then lagA=cmendt;
  else if cmendt>lagA>=cmstdt>. then DO;
        cmstdt=lagA+1;
        lagA=cmendt;
  END;
   else if lagA>= cmendt then flag=1;
   if first.subjid then  flag=.;

   if flag=1 then DELETE;
   FORMAT cmstdt cmendt laga date9.;
RUN;
%end;
%mend cmset;
%cmset(datain=cypih, dataout=cypout, cycle=1);
```

If we do not want to consider the CYP3A4 inhibitor grades as defined on table 1, we treat any grade higher than 0 as CYPIH flag and run the %cmset macro to collapse the ADCM dates. The output will then contain no overlapping or duplicate ConMed start and end dates as shown in the example below:

SUBJID	CMDECOD	CMSTDT	CMENDT
1002	POSACONAZOLE	2017-08-22	2019-10-10

1002	VORICONAZOLE	2019-10-11	2019-12-10

However, sometimes modelers want to know the strongest CYP3A4 inhibitors taken on a given day. To do this, we need to set up different covariates for each grade.

If the %cmset macro is used to remove the overlapping dates for each grade, the following resulting output datasets will be produced:

SUBJID	CMDECOD	CMSTDT	CMENDT	CYPIH1
1002	TACROLIMUS	2017-08-30	2017-10-14	1
1002	AZITHROMYCIN	2018-06-16	2018-06-16	1

SUBJID	CMDECOD	CMSTDT	CMENDT	CYPIH2
1002	CIPROFLOXACIN	2018-06-16	2018-07-01	2

SUBJID	CMDECOD	CMSTDT	CMENDT	CYPIH3
1002	POSACONAZOLE	2017-08-22	2019-10-10	3
1002	VORICONAZOLE	2019-10-11	2019-12-10	3

The final step of this strategy is to merge each dataset to the NONMEM PopPK dataset by ADT date.

```
proc sql;
  create table NONMEM1
  as select s.*, a.cypih1, b.cypih2, c.cypih3
  from NONMEM as s
     LEFT join cypih1 as a
  on s.subjid = a.subjid AND  a.cmstdt <= adt <=a.cmendt
     LEFT join cypih2 as b
  on s.subjid = b.subjid AND  b.cmstdt <= adt <=b.cmendt
     LEFT join cypih3 as c
  on s.subjid = c.subjid AND  c.cmstdt <= adt <=c.cmendt
   ;
quit;
```

Table 2: NONMEM PopPK Data with ConMed Covariates Merged

SUBJID	NDAY	NHR	EVID	AMT	DV	ADT	CYPIH1	CYPIH2	CYPIH3	CYPIH
1002	1	0	0	.	0	10/13/2017	1	0	3	3
1002	1	0	1	300		10/13/2017	1	0	3	3
1002	1	1	0	.	72	10/13/2017	1	0	3	3
1002	1	2	0	.	107	10/13/2017	1	0	3	3
1002	1	5	0	.	213	10/13/2017	1	0	3	3
1002	7	0	1	300		10/17/2017	0	0	3	3
1002	7	0	0	.	8.68	10/18/2017	0	0	3	3

1002	7	0	1	300		10/18/2017	0	0	3	3
1002	7	1	0	.	751	10/18/2017	0	0	3	3
1002	7	2	0	.	1010	10/18/2017	0	0	3	3
1002	7	5	0	.	621	10/18/2017	0	0	3	3
1002	28	0	1	300		6/20/2018	0	2	3	3
1002	28	0	0	.	18.8	6/21/2018	0	2	3	3
1002	28	0	1	300		6/21/2018	0	2	3	3
1002	28	1	0	.	850	6/21/2018	0	2	3	3
1002	28	2	0	.	683	6/21/2018	0	2	3	3
1002	28	5	0	.	1200	6/21/2018	0	2	3	3

Note: When EVID=1, ADT is the dosing date. CYPIH is the maximum grade of CYPIH1 to CYPIH3.

STRATEGY 2

For this strategy, we will read in ConMeds of interest from an external file, eliminate overlapping ConMed dates by expanding ADCM records and then merge it with NONMEM PopPK data. In this example, we don't numerically distinguish the CYP3A4 inhibitor grade.

Table 3: Sample external file of CYP3A4 inhibitors

Sheet1:CPYIH1	Sheet2:CPYIH2	Sheet3:CPYIH3
CMDECOD	CMDECOD	CMDECOD
AZITHROMYCIN	CIPROFLOXACIN	POSACONAZOLE
TACROLIMUS		VORICONAZOLE

After we read in the external file as indicated in table 3, and assign 1 to each class of ConMed of interest, the output datasets cypih1, cypih2 and cypih3 will appear as follows:

CMDECOD	CYPIH1
AZITHROMYCIN	1
TACROLIMUS	1

CMDECOD	CYPIH2
CIPROFLOXACIN	1

CMDECOD	CYPIH3
POSACONAZOLE	1
VORICONAZOLE	1

Then output from each pre-defined ConMed list are merged with ADCM by CMDECOD (see table 4). Assuming missing and partial CMSTDT and CMENDT dates have been imputed on the ADCM dataset

according to the rules described for handling missing dates, we keep ADCM records that are on-treatment only to merge with each class of ConMed.

Table 4: ADCM Data with CMDECOD Identified from External File of CYP3A4 inhibitors

SUBJID	CMDECOD	CMSTDT	CMENDT	CYPIH1	CYPIH2	CYPIH3
1002	TACROLIMUS	2017-08-30	2017-10-14	1		
1002	AZITHROMYCIN	2018-06-16	2018-06-16	1		
1002	CIPROFLOXACIN	2018-06-16	2018-07-01		1	
1002	POSACONAZOLE	2017-08-22	2019-10-10			1
1002	POSACONAZOLE	2017-08-31	2017-09-23			1
1002	VORICONAZOLE	2018-08-25	2019-12-10			1

ADCM dosing date is expanded to one day, one record, and NODUPKEY is used to remove duplicated dates by %cm_daily macro.

```
%macro cm_daily (cmf=, dsout=);
data cm_daily;
      set cm_all;
      by usubjid cmstdt;
      where &cmf ne .;
            do i = cmstdt to cmendt;
                  output;
                  end;
      keep usubjid cmdecod cmstdt cmendt i &cmf;
      format i date9.;
run;

proc sort data=cm_daily nodupkey; by usubjid i; run;

data &dsout;
      set cm_daily;
      adt=i;
      format adt date9.;
      keep usubjid adt &cmf;
run;
%mend cm_daily;
%cm_daily (cmf=CYPIH1, dsout=CM1);
%cm_daily (cmf=CYPIH2, dsout=CM2);
%cm_daily (cmf=CYPIH3, dsout=CM3);
```

The expanded dataset of CYPIH1 after removing the duplicated dates appears as follows:

SUBJID	CMDECOD	CMSTDT	CMENDT	CYPIH1	ADT
1002	TACROLIMUS	2017-08-30	2017-10-14	1	2017-08-30
1002	TACROLIMUS	2017-08-30	2017-10-14	1	2017-08-31
1002	TACROLIMUS	2017-08-30	2017-10-14	1	2017-09-01
.	.	.	.	.	.
.	.	.	.	.	.

.	.	.	.	.	.
1002	TACROLIMUS	2017-08-30	2017-10-14	1	2017-10-13
1002	TACROLIMUS	2017-08-30	2017-10-14	1	2017-10-14
1002	AZITHROMYCIN	2018-06-16	2018-06-16	1	2018-06-16

Then CM1, CM2 and CM3 datasets are merged using USUBJID and ADT to create the NONMEN PopPK dataset.

STRATEGY 3

For this strategy, we illustrate the third method of eliminating overlapping ConMed dates and derive CPYIH1-CPYIH3 covariates onto NONMEM PopPK data. In this example, we also do not numerically distinguish CYP3A4 inhibitor grade. Assume temporary flag variables CYPIH1F-CYPIH3F are merged onto ADCM as in table 5, then use macro %conmeds1 to eliminate overlapping ConMed dates.

Table 5: ADCM Data with CMDECOD Identified from External File of CYP3A4 inhibitors

SUBJID	CMDECOD	CMSTDT	CMENDT	CYPIH1F	CYPIH2F	CYPIH3F
1002	TACROLIMUS	2017-08-30	2017-10-14	1		
1002	AZITHROMYCIN	2018-06-16	2018-06-16	1		
1002	CIPROFLOXACIN	2018-06-16	2018-07-01		1	
1002	POSACONAZOLE	2017-08-22	2019-10-10			1
1002	POSACONAZOLE	2017-08-31	2017-09-23			1
1002	VORICONAZOLE	2018-08-25	2019-12-10			1

```
%macro conmeds1 (cm = );

proc sort data=&cm._1 nodupkey;
     by USUBJID CMSTDT CMENDT;
run;

data &cm._2;
     set &cm._1;
     by USUBJID CMSTDT CMENDT;
     retain CMSTDT_ CMENDT_;
     if first.USUBJID then do; CMSTDT_ = CMSTDT ; CMENDT_ = CMENDT ;
end;
     if not first.USUBJID and CMENDT_ >= CMSTDT then do;
          CMSTDT = CMSTDT_;
          if CMENDT_ >= CMENDT then CMENDT = CMENDT_; else CMENDT_ =
CMENDT;
     end;
     if not first.USUBJID and CMENDT_ < CMSTDT then do;
          CMSTDT_ = CMSTDT;
          CMENDT_ = CMENDT;
     end;
run;
```

```
proc sort;
     by USUBJID CMSTDT CMENDT;
run;

data &cm._3;
     set &cm._2;
     by USUBJID CMSTDT CMENDT;
     if last.CMSTDT then output;
run;
```

These are outcomes from the code above. The overlapping ADCM dates are collapsed and adjusted.

CYPIH1 dataset

SUBJID	CMSTDT	CMENDT	CYPIH1F
1002	2017-08-30	2017-10-14	1
1002	2018-06-16	2018-06-16	1

CYPIH2 dataset

SUBJID	CMSTDT	CMENDT	CYPIH2F
1002	2018-06-16	2018-07-01	1

CYPIH3 dataset

SUBJID	CMSTDT	CMENDT	CYPIH3F
1002	2017-08-22	2019-12-10	1

Next assign CMFLAG=1 for CMSTDT and assign CMFLAG=3 for CMENDT.

```
data &cm.;
     set &cm._3;
     adt = input(CMSTDT, yymmdd10.); cmflag = 1; output;
     adt = input(CMENDT, yymmdd10.); cmflag = 3; output;
     format adt yymmdd10.;
     keep STUDYID USUBJID CMSTDT CMENDT &cm.f adt cmflag ;
proc sort;
     by USUBJID adt cmflag;
run;
%mend;
%conmeds1 (cm=CYPIH1);
%conmeds1 (cm=CYPIH2);
%conmeds1 (cm=CYPIH3);
```

We assign a flag variable CMFLAG=2 to an existing NONMEM PopPK dataset “nonmem”, now we append CYPIH1 - CYPIH3 datasets to nonmem data and sorted by USUBJID ADT and CMFLAG.

```
data nonmem_cm01;
     set nonmem CYPIH1 CYPIH2 CYPIH3;
proc sort;
     by USUBJID adt cmflag;
run;
```

Finally, ConMed covariates CYPIH1-CPYIH3 are flagged over CMFLAG=2 NONMEM PopPK data records if they are located between CMFLAG=1 and CMFLAG=3 as in table 6.

```
%macro conmeds2 (cm = );
data nonmem_cm02;
      set nonmem_cm01 (where=(adt^=.));
      by USUBJID adt cmflag;
      retain &cm.;

      if first.USUBJID then &cm. = 0 ;
      if cmflag = 1 and &cm.f = 1 then &cm. = 1;
      if cmflag = 3 and &cm.f = 1 then &cm. = 0;

proc sort;
      by USUBJID;
run;
%mend;
%conmeds2 (cm=CYPIH1);
%conmeds2 (cm=CYPIH2);
%conmeds2 (cm=CYPIH3);
```

Table 6: Output from Macro %conmeds2

SUBJID	NDAY	NHR	EVID	AMT	DV	ADT	CMFLAG	CYPIH1	CYPIH2	CYPIH3
1002						8/22/2017	1	0	0	1
1002						8/30/2017	1	1	0	1
1002	1	0	0	.	0	10/13/2017	2	1	0	1
1002	1	0	1	300		10/13/2017	2	1	0	1
1002	1	1	0	.	72	10/13/2017	2	1	0	1
1002	1	2	0	.	107	10/13/2017	2	1	0	1
1002	1	5	0	.	213	10/13/2017	2	1	0	1
1002						10/14/2017	3	0	0	1
1002	7	0	1	300		10/17/2017	2	0	0	1
1002	7	0	0	.	8.68	10/18/2017	2	0	0	1
1002	7	0	1	300		10/18/2017	2	0	0	1
1002	7	1	0	.	751	10/18/2017	2	0	0	1
1002	7	2	0	.	1010	10/18/2017	2	0	0	1
1002	7	5	0	.	621	10/18/2017	2	0	0	1
1002						6/16/2018	1	1	1	1
1002						6/16/2018	3	0	1	1
1002	28	0	1	300		6/20/2018	2	0	1	1
1002	28	0	0	.	18.8	6/21/2018	2	0	1	1
1002	28	0	1	300		6/21/2018	2	0	1	1
1002	28	1	0	.	850	6/21/2018	2	0	1	1
1002	28	2	0	.	683	6/21/2018	2	0	1	1

1002	28	5	0	.	1200	6/21/2018	2	0	1	1
1002						7/1/2018	3	0	0	1
1002						12/10/2019	3	0	0	0

After CYPIH1 – CYPIH3 are derived, CMFLAG=1 and 3 records are removed to keep original NONMEM PopPK data records.

CONCLUSIONS

Strategy 1 uses the PRXMATCH function to identify pre-defined concomitant medications in ADCM. It uses data step and retain statements to identify overlapping ConMed date entries, and then use PROC SQL to merge ConMed covariates to the NONMEM PopPK dataset by ADT range. The advantages of this strategy are that they are flexible and suitable for a short list of ConMeds, the coding logic is straight forward and easy to modify, and it is good for projects requiring a quick turnaround. The drawback is the overlapping ConMed dates may need a few cycles to be eliminated completely.

Strategy 2 reads each class of ConMed from an external Excel file to SAS datasets. It creates an individual ConMed record for each date for the entire period in which the ConMed was taken, and then removes any duplicates. Then ConMed covariates are merged to a NONMEM PopPK data by ADT through a data step. The advantage of this strategy is the straightforward logic that is used to eliminate overlapping ConMed date entries. The coding is simple, and it is easy to check the output results. The drawback is it could require a large CPU and run time if datasets are large.

Strategy 3 is similar to strategy 2 in that it reads ConMeds of interest from an external file and merges them to ADCM; however, it sorts out and removes the overlapping ADCM dates through data step and retain statements. Then it appends ConMed start dates and end dates to the NONMEM PopPK dataset and sorts by dates and dummy CMFLAGs. It uses retain statements to carry over ConMed covariates from ADCM to NONMEM PopPK data. The advantage of this strategy is that the efficient coding can handle very large datasets, and it requires less CPU and run time. The drawback is that the complicated logic may be difficult to understand and validate the output.

REFERENCES

Jing Su and Jiannan (Jane) Kang, 2018. Challenges and Strategies in PKPD Programming, PharmaSUG 2018 - Paper AA-13

Sree Harsha Sreerama Reddy, Vishak Subramoney, 2018. Dosing In NONMEM® Data Sets an Enigma. PharmaSUG 2018 - Paper BB-02

Florian, J. 2017. “Electronic Submission of Pharmacometrics Data Sets and Reports for Regulatory

Submissions.” Presented at the Joint Statistical Meeting 2017.

Koukuntla, R.K., 2014. “NONMEM® – A Programmer point of view”. Presented at PhUSE 2014.

Irving A. Dark, 2018. “Concomitant Medications: What a Programmer Needs to Know”. SUGI 24

ACKNOWLEDGMENTS

The authors would like to thank all Merck, Incyte colleagues who provided their valuable input for this paper. Thanks Mansur Kazi from Cytel for his contribution. Thanks Dr. Songhui Zhu, A2Z Scientific Inc. for his comments.

CONTACT INFORMATION

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

Dan Xiao, dan.xiao2@merck.com.

Jing Su, jing_su@merck.com.

Shuqi Zhao, shuqi.zhao@merck.com.

Xiang Liu, xiangl@incyte.com.