PharmaSUG 2018 - Paper AA-13

Challenges and Strategies in PKPD Programming

Jing Su and Jiannan (Jane) Kang, Merck & Co., Inc.

ABSTRACT

PKPD Programming specializes in handling Pharmacokinetic (PK) data and creating data sets for PK analysis and PKPD modeling. The PK data creation process faces many difficulties including lack of industry standards on PK data sets, disparate data sources, evolving data collection standards over time, source data accuracy and completion, and complicated programming logic due to the additional requirements of the specialized software for PKPD modeling. These difficulties make it even more challenging to achieve quick turn-around with good quality and traceability. In this paper, these challenges will be explained in detail and the strategies to overcome them will be discussed. This paper will also highlight the differences between statistical clinical programming and PKPD programming, and emphasize the importance of the collaboration between these two programming groups on consistent usage of clinical data.

INTRODUCTION

Pharmacokinetics (PK) describes what the body does to a drug; whereas pharmacodynamics (PD) describes how the body responds to a drug (measured in terms of AEs and efficacy). PK samples are collected not only in phase I clinical studies, but also in phase II and III studies. Depending on the needs, PK samples may be collected in post-approval studies too. PK analysis and pharmacometric (PK-PD) modeling are conducted at different phases to answer different questions. For example, at phase I, analysis and modeling results are used to support go/no-go decision, safe dose selection, and regimen selection; at phase II, besides go/no-go decision, the primary focus is to select optimal phase III dose and identify intrinsic/extrinsic factors; while at phase III, usually PK-PD modeling is for dose modification for efficacy and safety.

PKPD Programming refers to the programming support for non-compartmental PK analysis and pharmacometric modeling. Its deliverables include analysis or modeling data sets and report-ready table, figure and listings. Pharmacometricians use the datasets for analysis and modeling. The results are summarized in Clinical Study Report (CSR), Common Technical Document (CTD), and modeling and simulation reports. Some of the information is eventually included in drug label.

PKPD PROGRAMMING DELIVERABLES

PKNCA data

PKNCA data is to support non-compartmental PK analysis, which calculates PK parameters like AUC, Cmax, Tmax, etc. without model assumptions. In PKNCA data, besides PK concentration and PK sampling time, it also includes dosing time, dose amount and some demographic covariates. Some derived variables are required, like actual time since first dose and actual time since last dose prior to the sample. At most companies, PK sampling time is in the clinical database, but PK concentration data is in a different database usually owned by the PK group.

Usually actual sampling and dosing time are not used for statistical analysis. But time variables, especially derived time since last dose prior to the sample, are very important to PK analysis. Incorrect time since last dose can cause incorrect analysis and result interpretation.

Per FDA guidance, all analysis datasets that support CSR should be in ADaM format for submission and all TLFs should be generated using ADaM datasets to keep traceability. PKNCA data can be referred as ADPC or ADNCA.

PopPK data

PopPK data supports population PK (PopPK) modeling. Common software for PopPK modeling is NONMEM, MONOLIX, R, MATLAB, etc. In 2016, International Society of Pharmacometrics (ISoP) Data Standards Group conducted an informal survey with 20+ participants from different pharmaceutical companies, CRO's, and institutions. Based on that survey, it is clear that NONMEM is the leading tool for PopPK and PKPD modeling. NONMEM stands for NON-Linear Mixed Effects Modeling. This software requires data in a very specific format and includes a mixture of data components like dosing records, PK and/or PD observations, and covariates (time independent or time dependent).

If ADPC data is used as input to create NONMEM PopPK data, one just needs to insert dosing records, add NONMEM reserved variables like EVID and AMT, etc. and then add additional covariates as needed. The data has to be sorted by subject ID and actual time since first dose because NONMEM expects the data in chronological order within each subject. On dosing records, every dose event needs actual dosing date and time, and actual dose amount. But for studies, like oral daily dose late stage studies, dosing time and dose amount are not always collected for every dose by study design. In this situation, imputation is required.

Other deliverables

- PKPD data: PKPD data supports exposure-response modeling, which provides additional critical information to determine efficacy and safety besides dose-response analysis. Depending on modeling purpose and measurements available, exposure can be PK concentration values over time or calculated PK parameters like model-predicted AUC, Cmax, etc. Response variables can be efficacy endpoints, biomarkers, or AE of interests. If PK variable is PK concentration values over time, usually the data structure is in NONMEM format.
- PKQTc data: PK-QTc modeling assessment in early phase development can, in most cases, avoid the need for a subsequent expensive thorough QTc study, and is required for regulatory decisions. PKQTc analysis data consists of time-matching pairs of drug concentrations and QT/QTc measurements.
- PKADA data: PK-ADA (anti-drug antibody) analysis data is to support immunogenicity assessment which is required by regulatory agencies to ensure efficacy and safety for therapeutic protein products.
- Report-ready table, figure, and listing.
- Submission deliverables like XPT files, and define.pdf for modeling datasets.

CHALLENGES IN PKPD PROGRAMMING

Challenges due to Source data

Multiple data sources

Raw PK concentration data and other clinical data are not in the same database. Sometime there are no consistent common variables for merging. For example, in PK concentration data, it labels the 24hr post-dose sample after day 1 dose as "Day 1 time 24hr"; however, in clinical data, the same sample is labeled as "Day 2 time 0". Some manipulation has to be done before merging. The fact that the data are from different databases also makes it difficult to check data integrity. Some records in PK don't have matching records in clinical, or vice versa.

Lack of standard on raw PK data in terms of file structure and/or variable names

Usually raw PK data is stored in Watson software, which is quite flexible on extracting data. Without clear guidance or tools, people can rename variables based on their preference and save the file either as excel or csv format. Sometime, even for the same protocol, the 2nd version of PK may have different variable names or file format comparing to the 1st version. Lack of standard on raw PK data format makes it harder to re-use SAS® code, and may cause unexpected errors.

Evolving data collection/mapping standards over years

In clinical data, there are standards about data collection and mapping. But the standards have changed over years. It is quite common to pool multiple protocols together for integrated modeling. Due to the changes in standards, it is difficult to reuse SAS codes.

Missing or incorrect sampling time or dosing time

Sampling time and dosing time are important. However, it is not uncommon to have missing or incorrect PK sampling time and/or dosing time, especially for late stage studies. In this situation, imputation rules and exclusion rules should be carefully discussed and applied.

Challenges due to process

Cross-functional collaboration

By nature, PKPD programming is a cross-functional collaborated process. On one hand, PKPD programmers communicate with pharmacometricians to understand modeling purpose and programming requirements. On the other hand, PKPD programmers need to communicate with biostatisticians and statistical programmers to get the right clinical data and to use it correctly. The communication is very important for studies with multiple database locks (DBL). Which locked data should be used? It depends on the modeling purpose. If it is for filing, then the same source data should be used for both statistical deliverables and PKPD deliverables to ensure consistency. If there is a database relock, or ADaM data sets are updated for any reason after DBL, then PKPD programmers need to request the updated data from statistical programmers, and update all modeling datasets and results accordingly to keep traceability and consistency. Otherwise, in the same submission package, there might be unexpected discrepancy between statistical analysis and PKPD modeling results.

Individualized NONMEM PopPK data specification

Individualized data specification is usually due to a lack of standards and clarification. It is hard and inefficient to program and validate against this kind of data spec.

Quick turn-around after database lock

Usually statistical programmers can use blinded clinical data to prepare analysis datasets and TLFs before DBL. In theory, once they get unblinded data, the statistical programmers can rerun their code with little to no modifications to generate final analysis datasets and TLFs.

In blinded data, PK concentration values are masked to prevent unblinding because only non-placebo subjects have positive concentration values. Data issues related to concentration values cannot be identified and cleaned using blinded data. In PK modeling datasets, critical variables need to be derived based on dosing information and concentration values. So draft PK modeling data can be prepared using blinded PK and dosing data, but it may require lots of modification after DBL to get the correct data.

In order to achieve quick turnaround after DBL, sometimes pharmacometricians and PKPD programmers may have to request early unblinding on PK related data before DBL.

Challenges due to analysis data content

Two or more dependent variables in a NONMEM dataset.

When there are two or more time-varying dependent variables, some people tend to use horizontal data structure, in other words, one variable one column. But this structure causes unnecessary programming

difficulties. First of all, merging can be a challenge if those dependent variables come from different source data. Secondly, since samples/measurements were not collected at the same time, it is hard to define and derive a reasonable actual time since last time variable for each record.

It is much easier to handle this type of data using vertical data structure. Similar to ADaM BDS structure, use a flag variable to differentiate dependent variables.

Imputation due to incomplete dosing history

As mentioned above, at each dosing record, dosing date and time are needed to calculate time since first dose. When actual dosing time is not available, imputation is needed. Depending on the study design and collected information, imputation methods can be different.

Common imputation methods are the following:

- use the time of first dose
- use the time of last dose
- add a small number to the time of predose PK sample

No matter which method is used, it is critical to ensure the imputed dosing time does not change time of sample relative to dose. For example, the imputed dosing time should not make predose sample become postdose sample or vice versa. Usually the third method listed above is preferred.

Imputation for missing time dependent covariates

In most models, the covariates are baseline demographic, lab or vital sign values, like age, gender, creatinine, weight, etc. Sometimes, incorporation of time-dependent covariates like dynamic lab and tumor size in models is of great interest, but raises some challenges. One concern is that covariates usually are not available at all PK sample time-points, therefore imputation is required. It is well known that common imputation methods including mean imputation and last value carried forward methods may give misleading results. Recent research has proposed quite a few new methods to handle missing time-dependent covariates. Pharmacometrician defines and provides the most appropriate imputation rule(s). PKPD programmer should carefully implement and discuss with pharmacometrican if any concern.

Using concomitant therapies as covariates

There are different ways to construct concomitant medication (CM) variables. The CM flag variables can be defined based on one drug or a class of drugs.

1. Binary variable: as long as a patient took the drug(s) of interest, no matter how long he/she took, the flag variable is 1 (yes).
2. Another way is to define the CM flag variable based on how long a patient took the drug(s). For example, "no" = 0% covered; "some" = >0% and <90% covered; and "yes" = over 90% covered.

If drug start date or end date is not complete in CM data, imputation is required.

STRATEGIES TO IMPROVE EFFICIENCY AND ACCURACY

The key is to standardize input source data format, improve source data quality, and standardize output modeling data requirement and specification. Once both input and output format are standardized and the source data quality is good, SAS macros can be created to automate the data creation processes.

Standardization

Standardize raw PK data by following SDTM standards: At each company, the PK data process can be different, but regardless of the process, SDTM standards can be adopted. Starting in SDTM IG 3.1.2, there is a specific domain for PK concentration data - the PC domain. FDA guidance requires all studies starting in or after December 17, 2016 to submit SDTM (3.1.3 or later) and ADaM.

Adopt ADaM standards in PKPD data when ADaM datasets are the input: FDA doesn't require PKPD modeling datasets to follow ADaM standards. But if the input are in ADaM format, like ADSL, ADPC, ADAE, ADEEF, etc, then it is better to follow ADaM standards for PKPD modeling data sets as much as possible to maintain traceability, minimize data manipulation, and improve efficiency.

Develop PopPK data standards – an ISoP initiative: Many companies realized the challenge of using individulaized data specifications and started to standardize them within the organization. In late 2015, ISoP initiated an effort to develop industry-wise PopPK data standards including guidance on missing time imputation. The first draft was released for public review in late 2017.

Quality improvement

PK reconciliation/cleaning to improve source data quality: In the last few years, many companies implemented new methods and processes to improve source PK data quality. For example, at Merck, raw PK data is loaded into clinical database, and unique sample ID is available to link PK concentration records with PK time records. This makes PK reconciliation and cleaning much easier. This also makes it possible to automate the PK analysis data creation process.

Streamline analysis/modeling data set validation process: At Merck, the analysis/modeling data set validation process has been streamlined to improve accuracy and efficiency. Based on each standard data spec, pharmacometricians and PKPD programmers collaboratively developed validation checklist which includes mandatory checks before delivery. For example, for PopPK data, time since first dose must be checked and can't be negative. Otherwise, it indicates a possible source data error or programming error. SAS outputs are generated based on the pre-defined validation checklist. Developer carefully reviews the outputs to ensure data quality. The same SAS outputs are shared with programmer validator and modeler reviewer. So they don't have to write seperate code to check on the same things.

Automation

SAS macros can be created to standardize common data manipulations and imputations. Macros will help improve efficiency, reduce validation needs and enforce standardization.

Build talent and keep experienced PKPD programmers

Last but not least, one strategy is to build talent and keep experienced PKPD programmers. A good PKPD programmer must have not only SAS skills and CDISC knowledge, but also NONMEM knowledge and a science background. On the job market, there are many excellent statistical programmers, but not many experienced PK programmers. It is important to build the talent and keep the talent.

CONCLUSION

Pharmacometrics is a growing field, so is PKPD programming. There are inevitably challenges and obstacles. This paper helps new PKPD programmers better understand and prepare for those challenges. This paper also gives statistical programmers some ideas on why PKPD programmers may request clinical data and how they use clinical data. It is important for the two programming groups to work together to ensure the consistent usage of clinical data.

REFERENCES

Florian, J. 2017. "Electronic Submission of Pharmacometrics Data Sets and Reports for Regulatory Submissions." Presented at the Joint Statistical Meeting 2017. Available at http://ww2.amstat.org/meetings/jsm/2017/onlineprogram/AbstractDetails.cfm?abstractid=323584

"Analyses and Displays Associated to Non-Compartmental Pharmacokinetics – With a Focus on Clinical Trials" A White Paper by the PhUSE CSS Development of Standard Scripts for Analysis and Programming Working Group. Available at http://www.phusewiki.org/wiki/images/e/ed/PhUSE_CSS_WhitePaper_PK_final_25March2014.pdf

Hwang, R., 2017. "CDISC Compliant NCA PK Parameter Analysis When Using Phoenix WinNonlin". Presented at PharmaSUG 2017. Available at https://pharmasug.org/proceedings/2017/PO/PharmaSUG-2017-PO01.pdf

Koukuntla, R.K., 2014. "NONMEM® – A Programmer point of view". Presented at PhUSE 2014. Available at https://lexjansen.com/phuse/2014/pp/PP12.pdf

High-level results from the survey by ISoP Data Standard Working Group in 2016 is available at http://www.go-isop.org/data-standards-working-group.

ACKNOWLEDGMENTS

The author would like to thank all Merck colleagues who provided their valuable input for this paper.

CONTACT INFORMATION

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

Jing Su, jing_su@merck.com
Jiannan (Jane) Kang, jiannan_kang@merck.com