

Challenges Facing the Programmer in Observational Research

Laurence Carpenter, Amgen, Uxbridge, UK

ABSTRACT

Observational Research (OR) is an increasingly important component in drug development and commercialization. OR contributes evidence regarding the frequency, distribution or clinical burden of disease, the natural history or clinical course of disease, the design of clinical trials, cost and utilization patterns, and the safety and effectiveness of interventions.

In the current environment there is increased demand for information from OR studies for regulatory and reimbursement authorities. Regulatory agencies are requiring large-scale observational studies for pharmacovigilance, risk assessment/management, and communication of risk:benefit profiles. Payers are increasingly demanding health technology assessments, the demonstration of real world value of interventions, and comparative effectiveness through OR designs.

Statistical programmers are more experienced in dealing with 'traditional' clinical trial data, but working with OR data can present many different challenges, and these will become more commonplace as the numbers of these types of study increase to match growing demand.

INTRODUCTION

Most statistical programmers working in the pharmaceutical industry are familiar with interventional clinical trials in phases I to IV with strict, well defined efficacy and safety objectives. However, as it becomes harder to develop new drugs and the requirement from regulatory and reimbursement authorities for OR studies grows, programmers will have to become more used to dealing with these types of studies and the challenges they present.

Safety barriers will continue to be raised, and more pharmacovigilance studies will be required to monitor longer term and less common side effects that may not be detected in clinical trials.

Pressure on costs will continue to increase. Reimbursement authorities will look to reduce costs as far as possible, and risk:benefit will need to be assessed and cost effectiveness proven to enable negotiation of best prices.

This paper will examine the similarities and differences between types of OR study and how the information they collect and the way it is collected affects the programmer. It will also consider such issues as how compliant the data needs to be regarding CDISC standards, how the changing OR environment impacts programmers and how the output delivered by the programmer may be used.

TYPES OF OBSERVATIONAL STUDY

There are a number of different types of observational study, and the choice may be dependent on the data required, the practical aspects of data collection and/or the desired timelines. Whereas an interventional clinical trial may compare the effects (both wanted and unwanted) of two or more treatments, using data collected according to a strict schedule of assessments as defined by the study protocol, an observational study will generally use data collected during routine clinical care to analyse safety, effectiveness and costs of treatments or therapies used in clinical practice.

PROSPECTIVE DATA COLLECTION

Two of the most common types of prospective observational studies are registries (longitudinal cohort studies) and prospective chart review studies.

PhUSE 2011

Registries follow a group of subjects over time to observe what happens, with information collected as per the study protocol. These studies are non-interventional (so no investigational treatments are given), but instead patients are observed during their clinical management and study data are collected during these visits.

Prospective chart reviews are so-called because they use data that is recorded not for research but as part of a patients regular medical care, therefore the source data is their medical records (charts). The relevant information is usually extracted from the medical records into the CRF by a chart abstractor. Chart reviews are easier for patients, as they are not required to attend any study visits and informed consent may not be required.

RETROSPECTIVE DATA COLLECTION

The most common observational study using this type of data collection is the retrospective chart review. These also use medical records as source data, but with the key difference being that all data is collected retrospectively, therefore at the time a subject enrolls into the study all of the data that will be needed is already available in their medical charts. As a result, this type of study may be quicker to complete.

The relevant information is usually extracted from these medical records into the CRF by a chart abstractor, although this can be more arduous than for a prospective chart review as here a patients medical records several years old may need to be retrieved, reviewed and abstracted.

One possible benefit of chart review studies may be increased accuracy of data. As all information is taken directly from medical records, this may eliminate or reduce any potential 'recall bias' that may be present when a patient is asked questions at a study visit in a prospective trial.

RETROSPECTIVE DATABASE ANALYSIS

This type of analysis usually involves analysing an often large pre-existing database, which may be of clinical trials data, claims data or another sort of healthcare information.

DATA FORMAT AND COLLECTION

The way the data is collected and the format it is collected in will clearly have implications for the programmer.

In some situations the data may be collected and received in exactly the same way as for an interventional trial, in which case the programmer should be able to use the data as usual. However, on other occasions this may not be the case and this could present additional challenges that must be overcome.

DATA COLLECTION METHODS

In some observational studies data may be collected in a standard way, i.e. using a CRF/eCRF and with a Data Manager overseeing the entry, quality and cleaning of the data. In such cases the programmer should not face any additional challenges, and receive datasets extracted from the database ready for manipulation, analysis and reporting. Even in chart reviews, where an abstractor extracts relevant information from a patients medical records, if this data is entered into a CRF and cleaned it will make little or no difference to the programmer.

However, on other occasions the situation may not be so straightforward, as investigator collected data may be collected in a much more ad-hoc fashion on spreadsheets. Even in studies using a CRF, it may be the case that some further data is collected in addition to the CRF data (such as adjudication data or PRO [Patient Reported Outcome] information).

Also, very large claims or healthcare databases may be received as text files on portable hard drives/DVDs, and in these cases there will be little or no data management or data cleaning involved, and the information will have to be converted into datasets more useful to the programmer.

IMPLICATIONS FOR THE PROGRAMMER

One of the main challenges facing the programmer in observational research is data quality. Even if a Data Management function is involved, it can still be very hard to get clean and complete data if collection is retrospective. In the case of a retrospective chart review, where data is abstracted from medical records, if the information has not been recorded then there is no opportunity to raise a Data Management query to obtain it and the data will remain missing. In the case of a database analysis there is generally no data cleaning at all, and in fact the programmer may have to take on a data management type role by filtering out observations that are clearly incorrect.

PhUSE 2011

Ensuring data is available in a suitable format is also key for the programmer. It is desirable to receive datasets extracted from a validated system (such as a database), rather than having to introduce additional steps converting data from formats such as spreadsheets to something more useful. This is an area that a programmer can add value and one reason they should seek to be involved as early as possible in the study set-up phase, as often they will be closest to the data and using it on a day-to-day basis and therefore in the best position to recommend how it should be collected to facilitate analysis.

Coding techniques can also become important when dealing with observational data. While efficient coding is generally desirable, it rarely becomes critical in clinical trials. However, when dealing with very large datasets (millions of records) efficiency in programming can become more important, programs can take a long time to run and disk space can become an issue. Adaptations when programming using SAS® may include:

- avoiding PROC SORTs where possible
- creating Indexes
- using 'WHERE' instead of 'IF'
- testing programs using a subset of the data

Another potential challenge can be the rate at which data is received. In the case of retrospective chart reviews, at the point of enrollment all of a patient's data has already been recorded in their medical notes. If chart abstraction occurs quickly then large volumes of data can be received very rapidly. While this may be advantageous in terms of quick completion of the study, the programmer must be aware and ready for this challenge - if timelines have been set based on patient recruitment then flexible resourcing will be required.

THE CHANGING ENVIRONMENT

As described earlier in the paper the environment pharmaceutical programmers are operating in is changing, and observational research provides a number of different ways of working.

GOALS AND OBJECTIVES

These may be less clear cut than for a typical Phase II or III study where primary and secondary objectives are clearly laid out in the study protocol. While there are still pre-defined objectives within observational studies, it may be more common to see ideas evolve and develop during a study (especially if outside influences are a factor, e.g. reimbursement authorities or expert opinion) and the balance between pre-specified and ad-hoc analyses may shift.

This can make a major difference to the way a programming team operates, as it makes planning harder and a greater degree of flexibility may be required. Alterations to plans and specifications during the course of a study may need to be absorbed and these can have numerous implications for the programmer.

Additional pressure may be placed on timelines if specifications are changed during the reporting phase, and lead to inefficiency in resourcing if re-work is necessary. A significant volume of ad-hoc work can also lead to resourcing challenges.

In addition, alterations to plans and specifications can also introduce a risk to quality, especially if reporting has already started and completed work has to be re-visited. This is something the programming lead needs to be vigilant of and manage carefully.

USE OF OUTPUT

Another difference encountered in observational research is the end product. In clinical research the derived analysis datasets and the outputs (tables, figures and listings) are usually the final deliverables for a study, and will form the basis of a clinical study report and/or a submission.

However, in observational studies this may not be the case, and tables/figures/listings may just be a part of the process rather than the end product. If a study is being run for reimbursement or costing purposes it is likely that the data generated will be used to populate an economic model, and the results being produced may be needed in an alternative format for use by the modeller.

FUNCTIONAL INTERACTIONS

This is a key area. Programmers in clinical research have generally worked most closely with statisticians, data managers and clinical/medical colleagues. While these interactions do remain within observational research, there will

PhUSE 2011

often be additional team members involved, such as health economists and modellers, some of whom may have little or no knowledge of programming or experience of working with programmers.

It is here that programmers can add significant value by explaining their role and passing on their experience. Conversely there is a lot to be learnt from health economists and other new colleagues, and it is a mutual education process. Communication skills become very important for the programmer in these situations, as new relationships must be fostered and methods of working have to be developed to ensure the smooth running and reporting of studies.

COMPLIANCE TO DATA STANDARDS

A topic of great interest to programmers currently is data standards, and this also applies for programmers working in observational research.

CDISC STANDARDS

One of the important questions when planning an analysis of observational data is whether SDTM and ADaM compliant data is required, as this isn't always clear cut. It will often be the case that observational studies follow on after regulatory approval and will not form part of a submission package. Also, it may be the case in studies focussing on costs and reimbursement that no investigational product is actually received by any patient.

As a result, observational study data may not be submitted to a regulatory body (such as the FDA) that requires CDISC compliant data, and it may not be efficient to work through the steps of creating standard datasets (especially if SDTM domains need to be derived from 'raw' data).

However, if the rest of a program of studies for a particular product has been analysed using CDISC compliant data, or if there is a possibility that submission of the data may be required at a later date then it may still be worthwhile creating SDTM and ADaM datasets to increase efficiency by re-use of standard programs.

In summary, there may not be a requirement to use standard data structures in observational studies, and their use needs to be examined on a case-by-case basis.

CONCLUSION

In conclusion, this paper has highlighted the increasing demand for observational research, and the likelihood that this type of study will become even more common in future.

Observational Research raises many new challenges, and programmers must be able to rise to these. They will require additional knowledge, increased flexibility and project management skills and will test communication skills in addition to the usual technical demands of the role.

CONTACT INFORMATION

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

Laurence Carpenter
Amgen
1 Uxbridge Business Park
Sanderson Road
Uxbridge
Middlesex
UB8 1DH

Tel: +44 (0) 1895 525529
Email: lcarpent@amgen.com