AR05

R you ready to join the RBQM revolution - Introducing rbqmR

Jeremy Howells (People & Product Leader), Roche
Rohan Palanganda (Principal Analytical Data Scientist), Roche

ABSTRACT

Have you heard the news? ICH guidelines require risk mitigation actions are in place, of which Quality Tolerance Limits (QTLs) (sometimes referred to as acceptable ranges) are now study critical deliverables…and yet we as an industry do not have a standard methodology. We all must be doing something within each of our organisations, further to this we must have some overlap? Can we make this more transparent? Surely there is a case for revolution? Let's revolt…let's join forces…let's open source!

Here at Roche we have an implementation of QTLs, that we are continuously looking to evolve and improve. Is it THE optimal solution? Are you doing anything different?

We see rbqmR as an opportunity to collaborate and engage with industry experts. The aim is to standardise methodology, improve knowledge and develop efficient tools within an open source framework.And why stop at QTLs? We have a real opportunity to work together to improve the whole Risk Based Quality Management strategy space. Let us introduce rbqmR to you.

INTRODUCTION

In this paper we will take you through Roche journey in RBQM adoption, more specifically the ideation process of a fit for purpose tool and methodologies for QTLs, the challenges and opportunities faced through this journey, the inception of opensource RBQM (namely rbqmR), where we are today, and where we'd like to be.

Risk Based Quality Management (RBQM)

As tradition would dictate, perhaps we should start with introducing Risk Based Quality Management (RBQM) and The International Council for Harmonisation of Technical Requirements for Pharmaceuticals for Human Use (ICH) guidelines that help drive this, these being the ICH Good Clinical Practice (GCP) requirements for data collection and review, briefly summarised in the following points:

- E6 R2: "The methods used to assure and control the quality of the trial should be proportionate to the risks inherent in the trial and the importance of the information collected."
- E6 R3: "The sponsor should adopt a proportionate and risk-based approach to quality management… identifying those factors that are likely to have a meaningful impact on participant's rights, safety and well-being and the reliability of the results"
- E8 R1: "Having a risk-based approach to quality management is an ICH recommendation advocated by Regulatory Bodies. Clinical trials should follow a Quality by Design approach"

Quality by design and the risk-based approach to quality management when done well, should and does have a dramatic impact on patients within a trial and ensure efficiency in the trial process. And so with the above in mind, (as we are sure most are aware) Risk Based Monitoring (RBM) and Risk Based Quality Management (RBQM) have evolved and been widely adopted across the industry in recent years, with RBM being a subset of RBQM. Whilst in theory the discipline and interpretation of RBQM should be mostly common across industry, study, and patients, even now there is less known/shared about gold standard practices that connect companies then perhaps there should be.

Roche began its adoption of RBQM in 2016 with the introduction of ICH E6 R2, beginning with Statistical Monitoring, by 2020 it grew into a cross functional effort across Biometrics, Operations and Quality setting up teams of experts who undertook components or capabilities of RBQM. These capabilities consisted of Risk

Assessment, Key Risk Indicators (KRIs), Quality Tolerance Limits (QTLs), Statistical Monitoring, Central Monitoring and Medical Data Review. Each capability's primary focus being to deliver expertise, procedures, tools and guidance in their own particular area whilst connecting centrally to their counterparts to ensure alignment. This approach has seen some great successes such as semi-automated and fully automated user tools being developed, new statistical methodologies, new guidance and procedures being created and adopted and new study support team roles being created, leading to a mostly successful adoption of varying levels of capabilities across a large portion of studies. But with the successes there has also been some major challenges; primarily due to a challenging change management landscape alongside a large mindset shift expected of study teams.

That mindset shift was no easy task, and one many teams across the industry were also facing, distilled into its rawest form: how to move from:- cleaning, checking and verifying all data on a study and treating it all as high risk to taking a risk-based approach, designing trials with quality by design in mind and only considering the required data.The former is how the industry had approached studies for decades and successfully delivered time and time again, so it is easy to see why many were sceptical of how doing what could be perceived as less and adding more “risk” would benefit their study. Nonetheless, the capability teams of Roche tackled this shift with many different approaches to work with end users, adapt to their evolving expectations and hopefully move as many into the Risk based mindset as possible.

One of these approaches, the one this paper will focus on, is the development of simplified tools and analytics that enable quick decisions, and how to clarify the underlying reasoning/context behind the tools so teams better understand why RBQM is important and what is impacting their study at the time. We needed to move the needle from manual, over engineered processes to automated and guided tools that engaged teams.

In this paper we will focus on the QTL and KRI capabilities, the journey to the new way of working for the internal Roche study teams, and how this work evolved into a vision for a framework or ecosystem of RBQM tools, something that could and potentially should be shared, one that could be collaboratively evolved: rbqmR, our attempt at starting an open source RBQM tool ecosystem.

Quality Tolerance Limits (QTLs)

A QTL is defined as “a level, threshold, or value associated with a parameter which is critical to quality. QTLs are set for risks identified at the trial level. A deviation from a threshold during the conduct of the trial may indicate a systematic issue that could impact participants' safety or reliability of trial results.” (Bhagat et al, 2021). Once identified, issues can be addressed by study teams with a number of immediate benefits, including: better patient safety; increased cost effectiveness of running trials; improved data quality; greater reliability of trial results. However, the benefits are not only immediate, as the build and conduct of future trials can also be informed, so as to proactively mitigate any risks or deviations already identified from previous trials.

As mentioned, these are risks identified at the trial level, however some of these risks are common across most trials. To that end, the most common risks across Roche trials (and most other clinical trials) were identified and some methodology developed in order to monitor them, for specific event types: the rate of adverse events (AEs), the rate of serious adverse events (SAEs), proportion of patients with baseline protocol deviations (BPD); and the rate of ongoing protocol deviations (OPD). Bayesian methods were used, with an MCMC (Markov Chain Monte Carlo) poisson model fit to AEs, SAEs and OPD rates and an MCMC binomial model fit to BPD proportions. Along with the results of the fitted models, a number of summary statistics are generated and the data is presented graphically, all of which are compiled into a report using rmarkdown.

There are assumptions that are made and these methods cannot be applied for all studies; for instance, smaller studies, with fewer sites and fewer patients at each site are not suitable for fitting a bayesian model, so the focus instead is on the summary statistics and plots. There is also the use of historical data to set up the prior distributions before a study has any data of its own, these historical studies being identified by study teams based on relevance and similarities to the newer studies.

Whilst the current methodology is considered acceptable, it is by no means a gold standard and is quite rigid, the data being dependent on SDTM structures and with a lot of the functions called within a single large wrapper function in order to generate the report and with little room to deviate from set use. Rather than rewriting and restructuring all the existing code , it was decided instead to move forward with a new approach, which resulted

in the initialisation of the rbqmR package, in order to make the implementation more flexible and ultimately allow for a potential cross-industry collaboration.

Challenges/Opportunities

Some interesting themes came up during the development of rbqmR and the eventual “opensourcifying” of it and in this section we will try to cover some of these.

First of all the decision between “InnerSource” vs “Open Source” and the difference in development or in our case re-development road maps that followed each approach.

Firstly let’s define each approach in relation to software development:

What is open source ? (Definition from opensource.com 2023)
Open source software is software with source code that anyone can inspect, modify, and enhance.

"Source code" is the part of software that most computer users don't ever see; it's the code computer programmers can manipulate to change how a piece of software—a "program" or "application"—works. Programmers who have access to a computer program's source code can improve that program by adding features to it or fixing parts that don't always work correctly.

What is InnerSource? (Definition from gitlab)
InnerSource is a software development strategy in which companies adopt an open source approach and culture to collaborate more effectively.

The differences are quite subtle. If you are developing or have developed a program or software internally at your own company, that you share or have shared, or open up to colleagues and co-workers to use and collaborate on then you potentially have already been “innersourcing” in some way.

The decision between open or inner source does not have to be binary per se; at Roche, what began as developing code to enhance our QTL methodologies/capabilities, slowly evolved as more people contributed to this internally, to the point where an R package was developed. Upon deploying this initial version on internal studies we then discussed the work some of the team members were involved in externally supporting QTLs and RBQM in industry wide working groups. These working groups are in support of sharing guidance and best practices, and so the proposal of open source being the link between the two was discussed. We had already seen the benefits of innersourcing, and how it enabled us to crowdsource solutions and code and so saw the potential for releasing the package externally to potentially support others within the industry but also help us enhance our own methodologies and approach. Ultimately we progressed from siloed code development to innersource to open source.

When the decision was made to opensource, we also took a look internally at the other capabilities. Here we saw similar journeys underway, but still very much in the initial stages of development. We took confidence in seeing that there was an ecosystem of tools in their infancy, but these teams potentially lacked awareness of the potential of an innersource/open source approach. This is where a vision of an RBQM ecosystem of packages or tools evolved. For the first time we saw at least internally that we could or should try to bring this all together under one umbrella of innersource or even potentially open source RBQM.

This decision comes with a number of challenges. For instance, in such a highly regulated and competitive industry, especially considering the sensitive nature of our research, there is always hesitation around sharing anything externally. In pharma we have a rich history of companies developing complex solutions quite often addressing the same thing, yet until recent history (Pharmaverse) we have struggled to truly share and collaborate on shared open source tools. Additionally there are the various legal hurdles and licences to even get this started and so for those that haven’t ventured into this space before this can be quite daunting. Fortunately there were already industry pioneers whose prior experience we could rely on.

Another challenge was the mindset shift. As mentioned, Pharma has not always been an easy industry in which to collaborate, especially considering the wide range of expertise, differing ideologies and varying levels of

scepticism. Even with the existing collaborations (i.e. in OpenPharma) paving the way it would be wrong to assume that any future collaborations would be carried out flawlessly and without complexity and so the same is expected with rbqmR.

rbqmR overview

The rbqmR package, hosted in OpenPharma (https://github.com/openpharma/rbqmR), takes what was previously implemented internally at Roche but applies it with a broader scope, the idea being that the existing methods can be implemented on data types not yet being monitored, such as the number of episodes of rescue medication or time to withdrawal of consent.

Whilst the original inhouse package is quite rigid in its application, rbqmR is built around tidyverse principals (Wickham et al.), allowing for greater variability in the data structures, being able to investigate intermediate steps in the data manipulations and analysis, and in the generation of any reports. Data can be presented in any appropriate format chosen i.e. tables and graphs and can be more easily QC'd when needed.

Most importantly however is that the methodology can be flexibly updated, with the existing functions written more independently and room to expand with the use of further methodologies.

Vision & Conclusion

To conclude let's look at where we are now and what the future potentially holds. Whilst we have released the rbqmR package, it is evident that it is not the finished product, nor was it intended to be and whilst this is not the best practice to acknowledge this, the hope is it does help others within the industry adopting QTLs, and it should highlight the potential of collaboration in this space. The package itself leaves plenty of room for expansion, enhancements and customisation, it also perhaps asks the questions as to whether more capabilities would benefit from adopting similar open source approaches. We have seen from recent history that where we have common industry challenges there have been successful solutions developed in collaboration and so could we see a similar approach in RBQM? Perhaps the solution isn't a one size fits all, but more the development of tools/packages that complement each other and work alongside other solutions which is the current status quo within Roche, or perhaps rbqmR is an experiment that whilst could have had a future, it is in fact more optimal at present to keep current practices. Ultimately we cannot predict the future of RBQM, but one thing we can see is that the RBQM process is being widely adopted, it is needed for efficient trials and to benefit the patients, and so it is not going away anytime soon.

References

Bhagat R, Bojarski L, Chevalier S, Görtz DR, Le Meignen S, Makowski M, Nadolny P, Pillwein M, Suprin M, Turri S. Quality Tolerance Limits: Framework for Successful Implementation in Clinical Development. Ther Innov Regul Sci. 2021 Mar;55(2):251-261. doi: 10.1007/s43441-020-00209-0. Epub 2020 Sep 3. PMID: 32885397; PMCID: PMC7864825.

Definition of innersource according to gitlab.com 2023 - https://about.gitlab.com/topics/version-control/what-is-innersource/

Definition of open source according to opensource.com 2023 - https://opensource.com/resources/what-open-source.

Wickham H, Averick M, Bryan J, Chang W, McGowan LD, François R, Grolemund G, Hayes A, Henry L, Hester J, Kuhn M, Pedersen TL, Miller E, Bache SM, Müller K, Ooms J, Robinson D, Seidel DP, Spinu V, Takahashi K, Vaughan D, Wilke C, Woo K, Yutani H (2019). "Welcome to the tidyverse." Journal of Open Source Software, 4(43), 1686. doi:10.21105/joss.01686.
https://tidyverse.tidyverse.org/articles/paper.htmlx