OPINION ARTICLE

Hackathon-driven tutorial development [version 1; referees: awaiting peer review]

Bruno M. Grande¹, Arjun Baghela², Anna Cavalla³, Florian Privé⁴, Peter Zhang⁵, Yisong Zhen⁶

¹Department of Molecular Biology and Biochemistry, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada
²Department of Bioinformatics, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada
³Eli Lilly, Windlesham, GU20 6PH, UK
⁴Laboratoire TIMC-IMAG, UMR 5525, Université Grenoble Alpes, Saint-Martin-d'Hères, 38400, France
⁵Department of Medicine, Faculty of Medicine, University of British Columbia, Vancouver, BC, V6T 1Z4, Canada
⁶State Key Laboratory of Cardiovascular Disease, Fuwai Hospital & Cardiovascular Institute, Chinese Academy of Medical Sciences, Peking Union Medical College, Beijing, 100037, China

Abstract
Software is essential for data science. However, several software tools remain out of reach for many users due to a lack of documentation, thus limiting progress in the field. Tutorial development by authors and users can greatly improve a tool's accessibility and accelerate its adoption. In this article, we explore hackathons such as hackseq as a venue for authors and users to develop tutorials to address the lack of documented software. We describe four advantages of hackathon-driven tutorial development as well as three challenges that we faced. We also discuss our experience with remote participation. In short, if properly prepared, hackathons can provide a productive venue for assembling a group of passionate people, including remote participants, to develop a suite of related tutorials and address the growing need for accessible software.

Keywords
software, hackathon, documentation, tutorial, vignettes, programming, data science

This article is included in the Hackathons collection.
Data science is an interdisciplinary field that relies heavily on the use of software tools. These tools require advanced domain-specific knowledge, which is often difficult to acquire and keep up-to-date considering the rate at which new methods become available and best practices evolve. This difficulty primarily stems from incomplete or unclear documentation. In the case of software packages (e.g. in R and Python), minimal documentation consists of describing inputs and outputs for individual functions. The Comprehensive R Archive Network (CRAN) is the de facto package repository for the R programming language and requires that submitted packages at least include this degree of documentation. CRAN also offers a framework for package developers to include additional documentation in the form of vignettes, which typically demonstrate real-world use cases. However, it appears that the majority of CRAN packages do not have vignettes\(^1\). On the other hand, vignettes are required for submission to Bioconductor, a package repository geared towards computational biology. This requirement has made Bioconductor and its numerous packages much more accessible to both new and experienced users. Unfortunately, the practice of including user-friendly vignettes or tutorials remains uncommon. To address this problem, the authors experimented with tutorial development in a hackathon project at hackseq 2017\(^2\). We describe our experience in this article.

The aim of our project was to organize the collective knowledge of a group of computational biologists into modular tutorials that leveraged the same dataset. Tutorial topics were proposed by and then assigned to members of the team. The tutorials were designed to be independent from each other, but they can readily be combined to form workshop lessons that use the same dataset. In this paper, we explore the benefits and challenges associated with hackathon-driven tutorial development, including the trade-offs of remote hackathon participation. Briefly, we believe that hackathons are an excellent venue for tutorial development and are particularly suitable for remote participation.

We found at least four benefits of collaboratively developing tutorials in a hackathon setting. First, interest in a given topic can be assessed based on the voluntary participation of hackathon attendees. The assembly of people interested in a topic can further motivate tutorial development. Second, once a common dataset is selected and processed, team members can efficiently work in parallel. Third, although team members do not have to rely on one another, they may draw on the collective knowledge of the team. The various perspectives and ideas from different research specialties can guide tutorial design, resulting in higher-quality material. In more practical terms, developing tutorials during a hackathon allows problems to be more readily resolved, and team members can perform peer review, leading to more polished tutorials. Fourth, these hackathon projects often bring together community members that have yet to work together and can thus catalyze new collaborations.

That said, there are some challenges or considerations to keep in mind when developing tutorials in the context of hackathons. First, the hackathon project should feature a theme or topic that is focused in scope so that team members can assist one another. The skill level of the target audience should also be determined beforehand. Second, once a theme is decided, any existing tutorials should be identified beforehand to avoid repetition. As mentioned previously, vignettes exist for many software tools, and time is best invested in developing new material that is not yet available. Alternatively, one could build on the work of others by adapting or improving existing open-source tutorials. Third, we suggest that you identify a dataset that can be used in all proposed tutorials and meets the following criteria: openly accessible; properly formatted (i.e. little to no missing or malformed data); relevant to the target audience; and ideally sized (i.e. large enough to be interesting but small enough to fit on a personal computer). For example, in the case of tutorials geared towards computational biologists, there are several interesting human genomic datasets, but access is often restricted for privacy. Accordingly, it may be more practical to select a dataset first and then determine a theme and set of tutorial topics that can be developed using this dataset.

It is often cost-prohibitive to travel for short conferences, especially when travel awards do not cover non-traditional meetings such as hackathons. Fortunately, remote participation is not only possible for hackathons, but relatively straightforward. Several tools exist to support collaborative projects while eliminating the need for collocation. For instance, GitHub offers decentralized code sharing, Skype enables face-to-face team discussions, and Slack is a popular platform for asynchronous communication, which is essential when team members inhabit distant time zones. For example, we successfully managed our project despite being located in Vancouver, BC with remote participants in China and France. We argue that remote participation is especially straightforward for tutorial development because material can be developed independently and thus asynchronously. On the other hand, virtual attendance precludes any participation in social or networking events. There is also additional work involved in ensuring that every local and remote team member have assigned tasks at any given time. Overall, though, we believe that allowing remote participation is a net benefit for a hackathon project, especially for tutorial development.

In conclusion, our experience with developing tutorials at a hackathon with remote participants was positive. We believe that we were able to achieve more together than separately, mostly because we gain access to immediate peer review of our tutorials. We also learned lessons that will help ensure success for future hackathon projects geared toward tutorial development. We believe this approach to be generalizable to other fields and a model for assembling passionate data scientists with similar interests and organizing their collective knowledge into modular tutorials. In turn, these tutorials can greatly benefit the field of data science by facilitating the adoption of powerful tools and accelerating the training of future data scientists.

**Data availability**

No data are associated with this article.
Grant information
The author(s) declared that no grants were involved in supporting this work.

Acknowledgements
The authors acknowledge the hackseq 2017 steering committee for organizing the hackathon.

References

   Reference Source
   PubMed Abstract | Publisher Full Text | Free Full Text
The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias
- You can publish traditional articles, null/negative results, case reports, data notes and more
- The peer review process is transparent and collaborative
- Your article is indexed in PubMed after passing peer review
- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com