The Code that Made My Data

The relevance of good RSE practices for Open Science Data

It's only FAIR

The relevance of Open Science and Open Data is becoming increasingly obvious in modern day publications. Most scientific publications draw conclusions from measured and raw data, and these conclusions can – to a degree – be validated by other scientists and potentially be used to continue the according research. But this does not generally hold true – in particular where *derived* data is involved, because drawing conclusions from data almost always means processing the data thereby generating new data. Where standard or simple methods are used, these can be referenced or elaborated in the paper – even though this is not even frequently done.

Frequently, however, scientists write their own analysis code (Software Sustainability Institute, 2022), as the complexity of analysis increases and the combination of methods become more relevant – from code conversion, to measuring and comparing. These functions and methods are not stable, are subject to change, are constrained to the use case and data used etc. – with the growing number of AI supported analyses, this becomes even more difficult, as these results can change as the AI method "learns" more.

A FAIRY Tale?

Opening and maintaining data already poses a substantial number of issues, including versioning, provenance, format, and generator specific constraints, such as precision, resolution etc. These problems intensify as we regard code that generates such data, as noted e.g. by Barker et al. (2022). In this presentation we will talk about the problems associated with maintaining research code. As such, for example, code executability is even more difficult to maintain as data readability. This is mostly due to the strong dependencies of code to libraries, drivers, hardware and operating systems. All of these dependencies are subject to frequent changes, which may cause the code to not execute properly anymore or – in the worst case – still execute but deliver different results. Maintaining the code is effort-intensive and therefore basically impossible in the context of a research publication. Obviously, if the code is adapted, we need to maintain all previous versions and refer to the right versions used in the publication to ensure that *in principle* the same results can be reproduced should a divergence arise as a consequence of maintenance.

Due to the specificity of the code, i.e. it being originally developed for a very specific use case and data format, it requires even more effort to adapt that given code to another context, if e.g. the data format or resolution changes, even if the type of analysis and the research question remains the same. Where possible, therefore, the algorithm behind the analysis code may be of more importance than the code itself, given that it is numerically correct. As an implication, implementations may diverge from the numerical results due to the platform accuracy – ideally only minimally, however. As noted, though, this is not appropriate for too complex code where the algorithm would be too difficult to represent and explain, or for AI based and related methods that depend additional data, aka learning context. Implicitly, such methods would have to be treated differently.

Is it REAL?

With respect to ensuring that data is not only FAIR, but also reproducible under any circumstances, we follow the suggestion that code must be treated in the same fashion, by making sure that all algorithmic processes published are

Reproducible in the sense that the results can be achieved again with the same process and context

Executable at any point in time (though not necessarily on any machine)

Attributable to the data and author at the stage of publication and

Literal in so far as that the algorithm is a sound and correct representation of the mathematical methods to be applied.

References

Barker, M., Chue Hong, N.P., Katz, D.S., Lamprecht, A.-L., Martinez-Ortiz, C., Psomopoulos, F., Harrow, J., Castro, L.J., Gruenpeter, M., Martinez, P.A., Honeyman, T., 2022. Introducing the FAIR Principles for research software. Sci. Data 9, 622. https://doi.org/10.1038/s41597-022-01710-x

Software Sustainability Institute, 2022. RSE International Survey 2022. Software Sustainability Institute.