Making research reproducible using literate programming

On programming a computer, we should concentrate on explaining to humans, what we want the computer to do – instead on focusing on instructing the computer. This idea, termed literal programming, has gained widespread currency in software development communities, but harbors benefit for science as well. In particular, literate programming can be thought of as an necessary condition for reproducible science. Reproducibility is an umbrella term referring to the degree that independent researchers are able to confirm the conclusions of some piece of research. The current "reproducibility crisis" is partly due to the fact that it is often hard or even impossible at all to reproduce the results of an (empirical) research article. For reproducibility, not only data and code must be available, but it must also be possible to easily connect pieces of code for data analysis, with its corresponding figures in the research manuscript. However, often enough researchers separate these three components in different documents, thereby making a reproduction difficult. In this poster, we demonstrate a simple "recipe" for producing a reproducible research report based on literate programming. In this approach, all components of a research paper (data, code, text) will be processed in the same file. Thus, reproducing the claims of a paper build up this way is straight-forward. We explain the use of well-established tools for that purpose. Specifically, we show how to mix data, R code, text (using Latex/Markdown) in one document using a R package called knitr.