White Paper

From Code to Reports with knitr & Markdown

Andrew Brooks
Data Scientist

May 2015

ELDER RESEARCH
DATA SCIENCE · AI · MACHINE LEARNING

Communicating results clearly is critical to the success of an analytics project. However, doing it well can take more time than actually performing the analysis. The process of taking analysis from code to presentation is often messy, involving copying, pasting, and manual steps that make it difficult to trace results back to the processes that generated them. If the data or analysis changes, edits have to be made to both the code and the presentation. Because of this, presentations get stale quickly.

Significant strides have been made in the last few years to address this issue by bringing report and presentation generation closer to code and the processes that generate results. knitr and R Markdown are two such tools for the R environment that embody this paradigm of literate programming. Introduced by Donald Knuth in the early 1980s, literate programming allows data scientists to code and document in a way that is more naturally in tune with their analytics workflow and thought process. The logic of the program is written in English (or other natural language) directly into code, which is then compiled into self-contained documentation.

What exactly is knitr?

knitr is the engine that creates dynamic reports from R Markdown or R scripts. It is an actively developed R package on CRAN, which is also currently supported natively in RStudio. knitr allows coders to create dynamic HTML, PDF, and even Word reports directly from R code and Markdown by encapsulating code, commentary, output (tables, plots, models), and all formatting into one document. Results are reproducible, with no copy and paste necessary.

What exactly is Markdown?

Markdown is an easy and efficient way for analysts to produce HTML documents. Writing content using HTML can be clunky if you're not a web programmer. Writing and reading content using Markdown (a plain text formatting syntax) feels much simpler. The core Markdown syntax fills just one web page. Here is an example Markdown document and the HTML page it produces.

R Markdown is really just a supercharged version of core Markdown. R Markdown documents include chunks (blocks) of R code, which produce the output that is included in the final document. These documents are typically saved with the .Rmd extension and drafted within an R environment. RStudio comes with built-in support to facilitate working with Rmd files.

When can knitr help?

Data exploration

Most data science projects start with exploratory data analysis. This process involves poking and prodding data by executing one-off instructions at the command line, generating temporary plots, and visually inspecting transformations of data, in addition to other ad-hoc approaches that go undocumented. I personally find myself repeating a lot of these unrecorded exercises; my brain can only hold so much for so long.

An analyst asked to transcribe every step of their data exploration might feel like a painter asked to document every brush stroke. After all, data exploration is just as much art as science. However, knitr takes some of the burden of documentation off analysts. If exploration code is simply saved into an Rmd or R script and results and plots are printed to the console, a lightweight report can be generated quickly. If the data changes, it is possible to simply re-run the exploration script to produce another report. I find that working interactively (inspecting results or plots as I generate them without the hassle of constant saving) works well in the moment, but having the full story knitted into a report to review weeks later helps me and team members remember what we've already explored and learned.

Internal communication (team)

Communicating methodologies and workflows

My team and I find that knitr & R Markdown reports are useful for our own understanding, whether or not we end up sharing them with our clients. Reading through a report, even if it was generated directly from an R script with minimal comments, helps us understand what each team member has done more quickly than we would through reading code and executing it line-by-line, block-by-block to see the output. Some of our more technical clients are interested in growing data science teams in-house, and these reports help them understand our process by connecting the code, methodology, and results.

Drafting documents in a team

knitr can turn the same Rmd or R script (among some other types) into an HTML, PDF, or Word file, which is helpful when working collaboratively on teams where members prefer different formats. We often generate drafts for review in Word, where team members can edit in-line and track changes, include the necessary edits in the Rmd file, and then re-knit into an HTML file to share over the web. Since the Word docs are generated from plain text files, drafts can be version controlled with software like Git, which is comparatively more difficult to do with just Word.

On a recent project, we leveraged this functionality to create a supercharged scratchpad for data exploration. We used an automated routine developed in-house to quickly generate a Word document of summary statistics and plots for every variable of each table in a database. Rather than writing commentary about the statistics and plots up front before knitting the document, we included empty sections for this commentary to be written afterwards. It was easier to focus on one document as we analyzed the stats and captured the insights gleaned from our first pass of exploratory data analysis.

External communication (client)

knitr and R Markdown help users seeking time savings, reproducibility, and aesthetically pleasing reports through a range of customization options. On a recent consulting project, we used knitr-generated HTML reports as the primary medium to communicate

technical results with our client. It improved our workflow by encouraging us to self-contain analyses and transparently document the trail from results back to raw data. It's actually possible to convert R scripts directly into reports, which we found introduced the least impedance to our workflow. R scripts can be sourced from other R scripts and easily run interactively in the console by selection. While this is not impossible with Rmd files, it is more cumbersome. To demonstrate this, I put together a sample R script and the HTML report it produces here with some common features and useful tricks. One helpful feature is the ability to link document text directly to objects (statistics or variables) in the R environment. If you expect your data or models to change, this can be more efficient than hardcoding results and numbers in Markdown text.

Education and training materials

I wish my university and grad school courses utilized these tools to supplement textbooks. A number of schools have already started. Many of the helpful data science resources I find online are created using this framework of literate programming. This Johns Hopkins University Data Science course is one example. knitr also integrates LaTeX and R code, which allows mathematical equations and formulas to be expressed clearly—a handy resource for academic papers and lectures.

Web

HTML is the language of much of the content on the web, so reports generated in HTML are naturally suited for sharing on the web. Web programmers who want to customize documents beyond the scope of what's provided in Markdown can include raw HTML and JavaScript code. I use knitr and R Markdown to write data-heavy posts for my Jekyll blog. I find it much more efficient than my previous process of copying and pasting code blocks and embedding plots saved individually into HTML.

Limitations

I list the following as limitations, though I'd be surprised if they couldn't be addressed with some clever techniques.

- **Compiling can be repetitive and slow:** There may be hacks to facilitate this process, but the conventional way to generate a report is to knit the entire document, even if you just made one small change since the previous version. These small changes are relatively less painful with a What You See is What You Get (WYSIWYG) presentation tool like PowerPoint or Word. However, the knitr cache option alleviates some of this pain associated with knitting documents with long run-times. However, care must be taken to ensure that cache is rebuilt when expected.
- **Multiple tools producing analysis:** R Markdown and knitr work most seamlessly when analysis is contained in one language, specifically R. While knitr provides engines that handle a host of other languages when used in

conjunction with R, this may provide a relatively higher level of impedance to your workflow, requiring a merge of multiple languages into one script.

Resources

Many useful resources are included in the hyperlinks above. Two more:

- Yihui Xiu, the author of knitr, has a comprehensive book, Dynamic Documents with R and knitr which details many of the technical options and features available within the knitr framework.
- The official knitr website contains a wealth of information necessary to help you get started.

About the Author

Andrew Brooks is a Data Scientist and project lead at Elder Research. He works with clients through the full analytics life cycle: identifying problems in ambiguous environments, strategizing analytic approaches, creating models using machine learning techniques and operationalizing these models to improve decision making. He has experience in the economic, financial, technology, government oversight and the international domains. Andrew's technical interests span machine learning, model validation, data visualization, building data tools and hacking code. Previously, Andrew worked as a Senior Research Assistant and lead Chile analyst with the Federal Reserve Board in the Division of International Finance where he specialized in forecasting, data visualization and programming.

Andrew earned an MS in Mathematics & Statistics from Georgetown University after graduating magna cum laude from American University with a BS in Economics, BA in International Studies and Minor in Mathematics. Outside of work, Andrew enjoys distance running, cycling, building furniture, board games and any adventure outdoors.

www.elderresearch.com

ELDER RESEARCH

DATA SCIENCE · AI · MACHINE LEARNING