Reporting Reproducible Research with R and Markdown

Garrick Aden-Buie // April 11, 2014

INFORMS Code & Data Boot Camp

Lots of things to install

- LaTeX
  - Mac: BasicTeX http://www.tug.org/mactex/morepackages.html
  - Windows: MiKTeX http://miktex.org/
  - Linux: `apt-get install texlive`
- pandoc
  - http://johnmacfarlane.net/pandoc
- R
  - http://r-project.org
  - http://rstudio.com
- knitr
  - http://yihui.name/knitr/
- Go all out: git
  - http://git-scm.org

Skip the talk, learn at home

- pandoc user guide
  - http://johnmacfarlane.net/pandoc/README.html
- knitr user guide
  - http://yihui.name/knitr/
  - http://kbroman.github.io/knitr_knutshell/
- git
  - https://bitbucket.org/gadenbuie/intro-to-git-for-scientists

Today we'll talk about

- What's the deal with Reproducible Research?
- What's up with Markdown?
- A complete research flow
- A simple example
- Show and tell

What's the deal with Reproducible Research?

- Kind of a hot topic these days
  - Coursera's course
  - PLoS One Data Policy
  - RunMyCode
- Code & Data are as much a part of research output as pubs

Reproducible research

...is the idea that data analyses, and more generally, scientific claims, are published with their data and software code so that others may verify the findings and build upon them.

Why should you care?

- Version Control and Management!

[1] YMMV

Why should you care?

- Version Control and Management!
- Start to finish, integrate everything

[1] YMMV

Why should you care?

- Version Control and Management!
- Start to finish, integrate everything
- Write once, output to anything

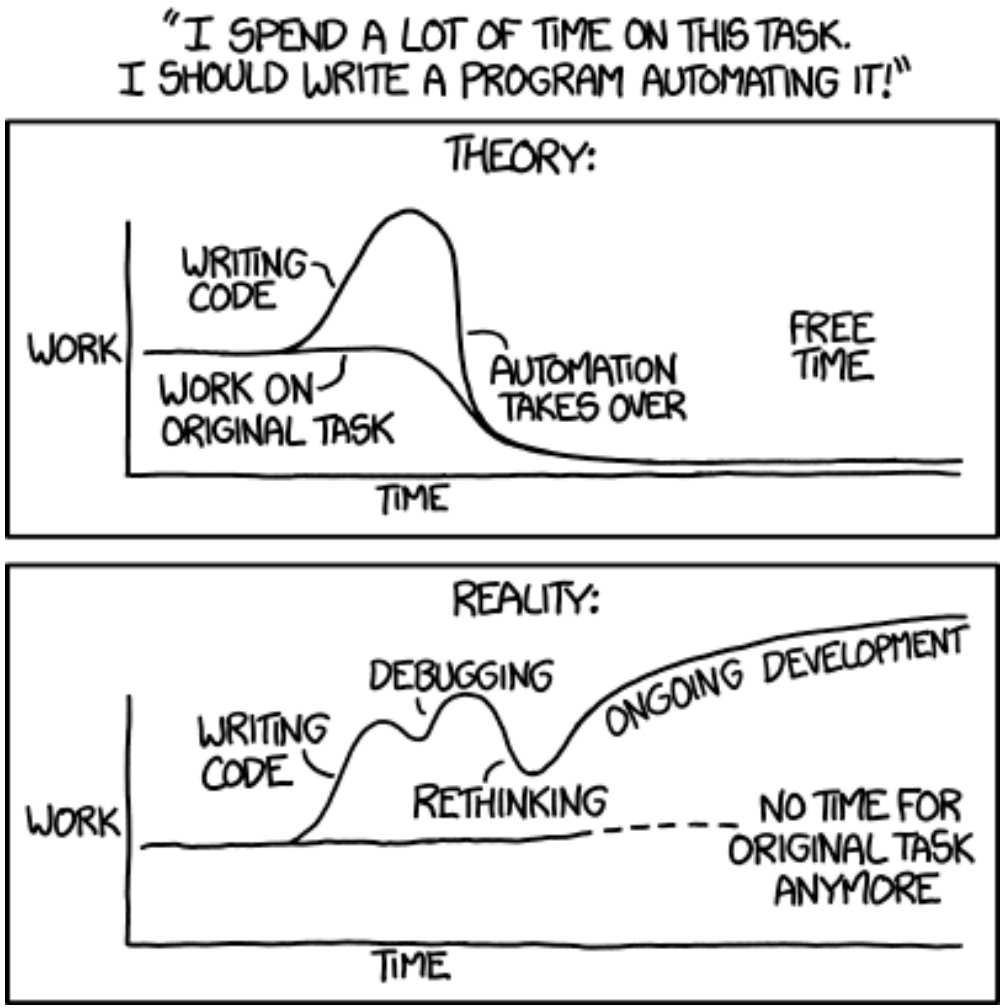
[1] YMMV

Why should you care?

- Version Control and Management!
- Start to finish, integrate everything
- Write once, output to anything
- Make collaboration easier[1] and more scalable

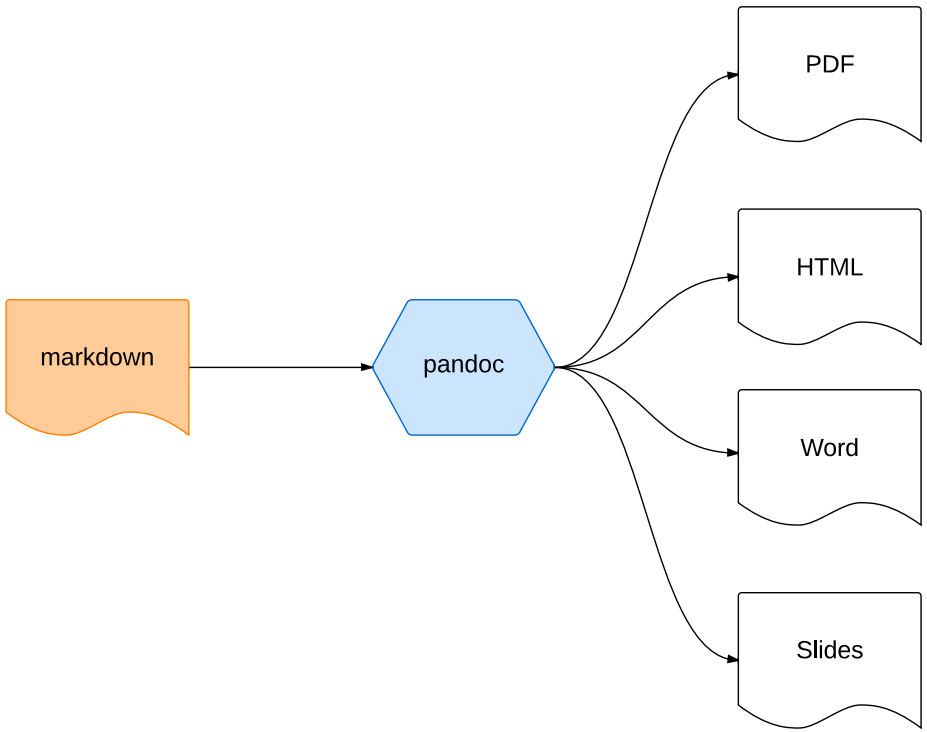
[1] YMMV

Which reminds me...

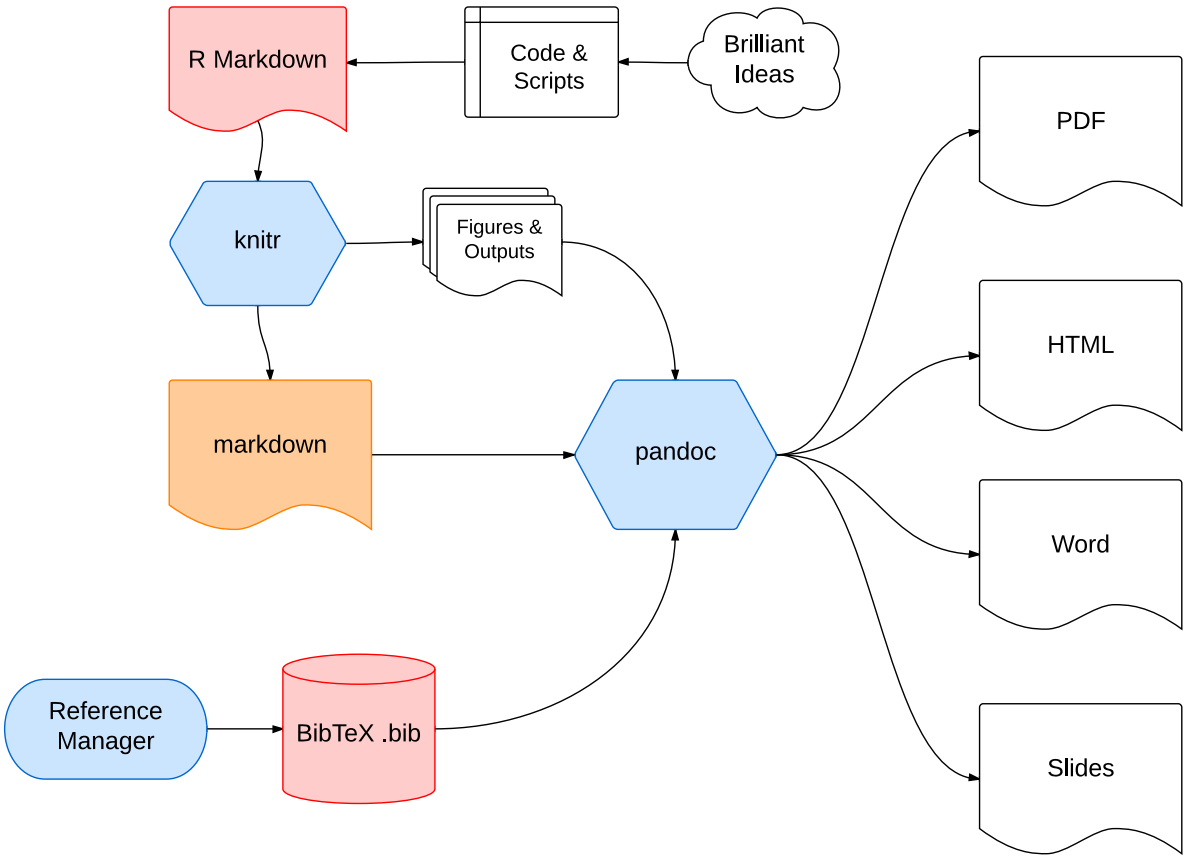


A complete research flow

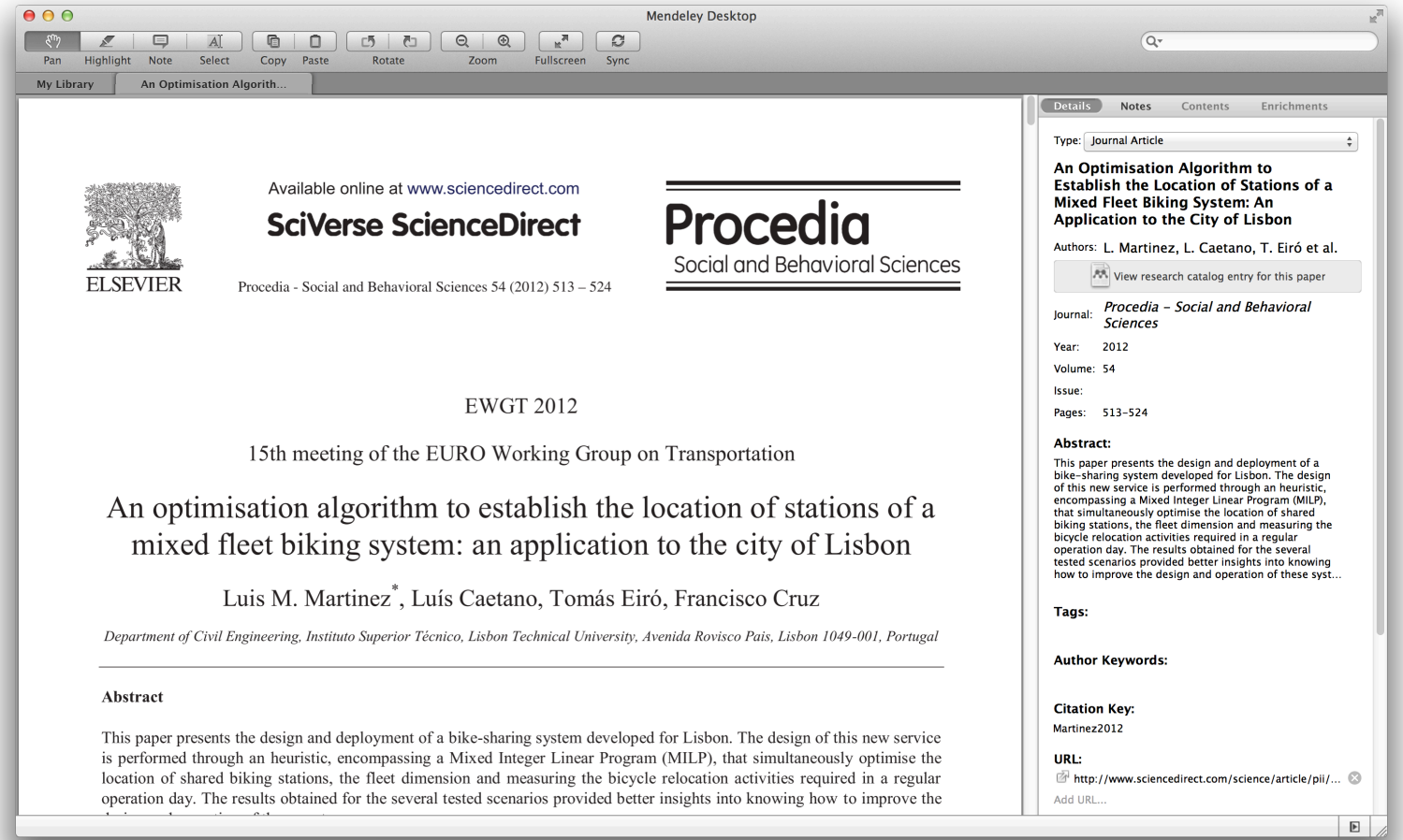
The core workflow



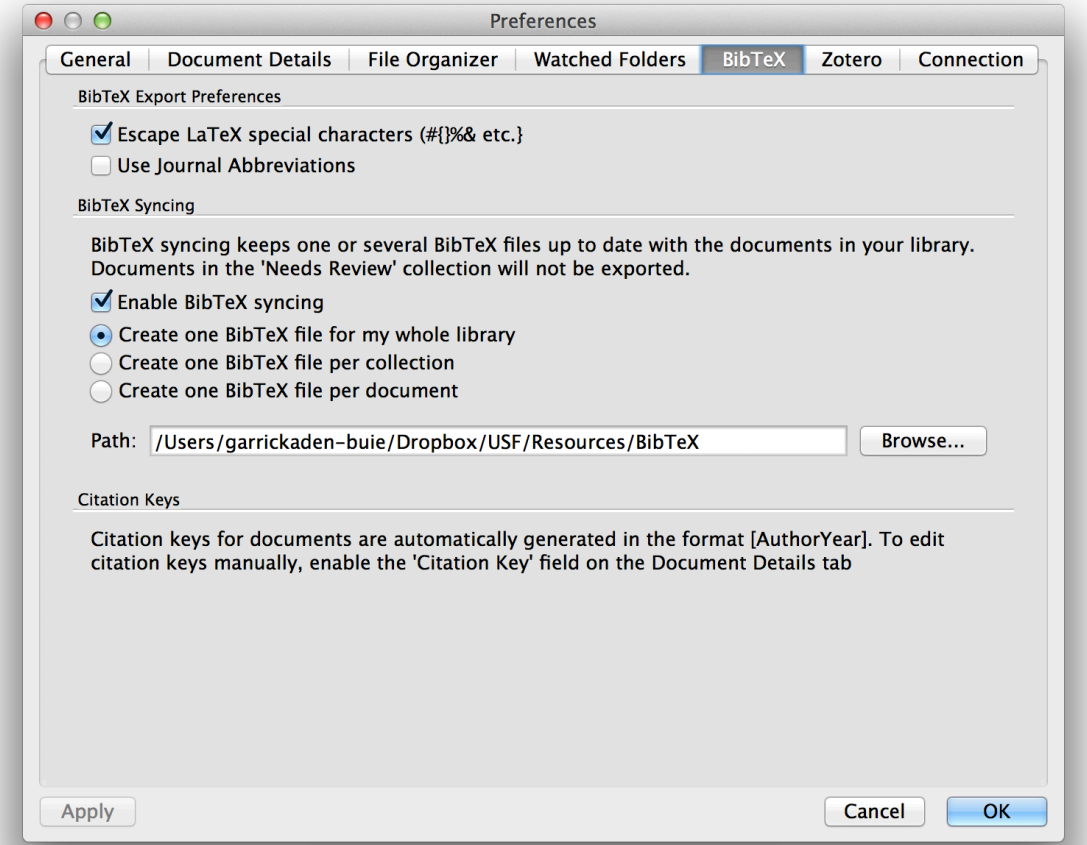
The full workflow



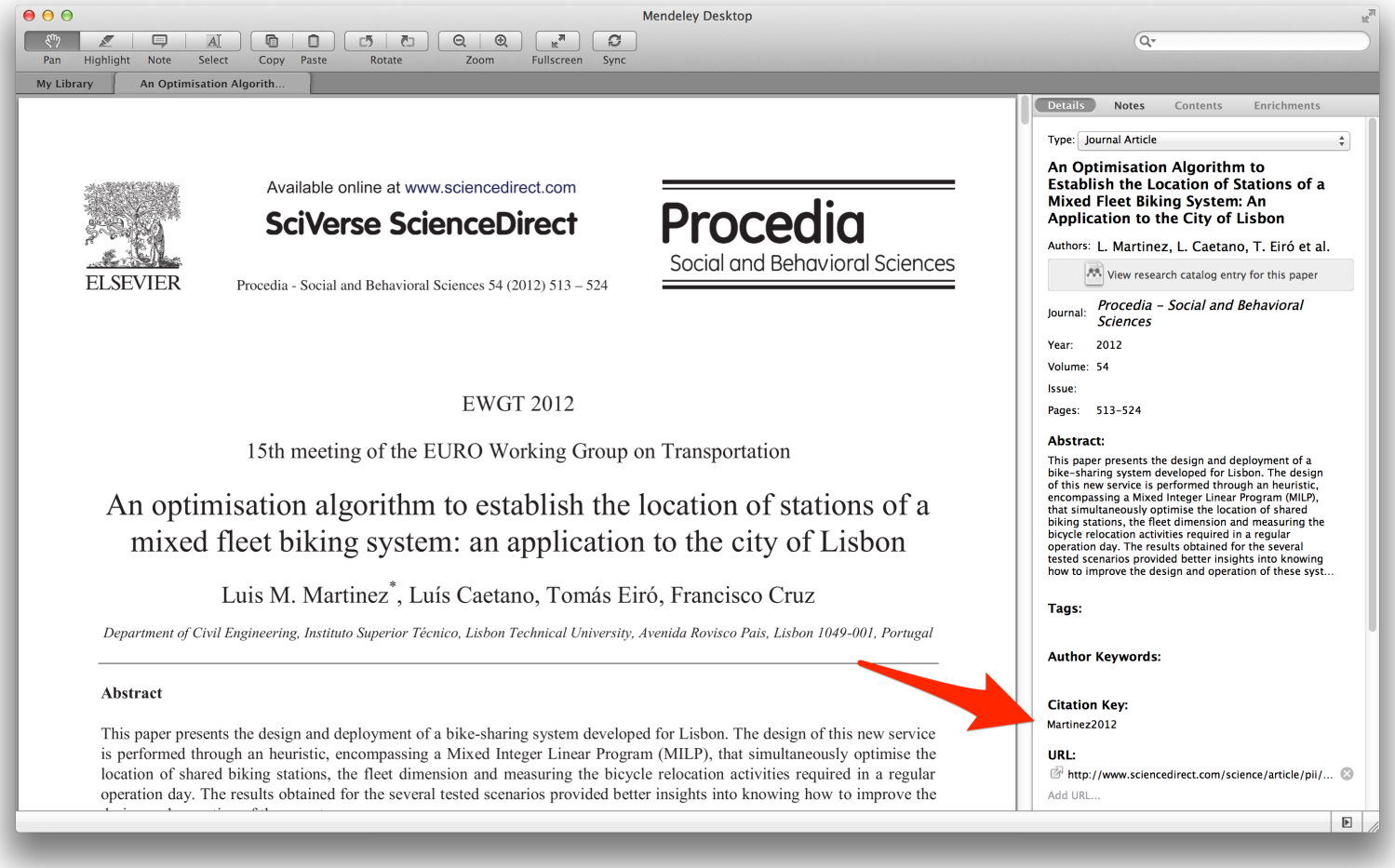
Mendeley


Mendeley Desktop
Pan
Highlight
Note
Select
Copy
Paste
Rotate
Zoom
Fullscreen
Sync
My Library
An Optimisation Algorith...
Available online at www.sciencedirect.com
SciVerse ScienceDirect
Procedia - Social and Behavioral Sciences 54 (2012) 513 – 524
Procedia
Social and Behavioral Sciences
ELSEVIER
EWGT 2012
15th meeting of the EURO Working Group on Transportation
An optimisation algorithm to establish the location of stations of a mixed fleet biking system: an application to the city of Lisbon
Luis M. Martinez*, Luís Caetano, Tomás Eiró, Francisco Cruz
Department of Civil Engineering, Instituto Superior Técnico, Lisbon Technical University, Avenida Rovisco Pais, Lisbon 1049-001, Portugal
Abstract
This paper presents the design and deployment of a bike-sharing system developed for Lisbon. The design of this new service is performed through an heuristic, encompassing a Mixed Integer Linear Program (MILP), that simultaneously optimise the location of shared biking stations, the fleet dimension and measuring the bicycle relocation activities required in a regular operation day. The results obtained for the several tested scenarios provided better insights into knowing how to improve the
Details
Notes
Contents
Enrichments
Type: Journal Article
An Optimisation Algorithm to Establish the Location of Stations of a Mixed Fleet Biking System: An Application to the City of Lisbon
Authors: L. Martinez, L. Caetano, T. Eiró et al.
View research catalog entry for this paper
Journal: Procedia - Social and Behavioral Sciences
Year: 2012
Volume: 54
Issue:
Pages: 513–524
Abstract:
This paper presents the design and deployment of a bike-sharing system developed for Lisbon. The design of this new service is performed through an heuristic, encompassing a Mixed Integer Linear Program (MILP), that simultaneously optimise the location of shared biking stations, the fleet dimension and measuring the bicycle relocation activities required in a regular operation day. The results obtained for the several tested scenarios provided better insights into knowing how to improve the design and operation of these syst...
Tags:
Author Keywords:
Citation Key:
Martinez2012
URL:
http://www.sciencedirect.com/science/article/pii/...
Add URL...

Set up Mendeley + BibTeX



Where to find the citekey



BibTeX entry

~/Dropbox/USF/Resources/BibTeX/library.bib

```
@article{Martinez2012,
  author = {Martinez, Luis M. and Caetano, Luis},
  doi = {10.1016/j.sbspro.2012.09.769},
  issn = {18770428},
  journal = {Procedia - Social and Behavioral Sciences},
  month = oct,
  pages = {513--524},
  title = {{An Optimisation Algorithm to Establish the Location
          of Stations of a Mixed Fleet Biking System: An Application
          to the City of Lisbon}},
  url = {http://www.sciencedirect.com/science/article/pii/S1877042812042310},
  volume = {54},
  year = {2012}
}
```

What's up with Markdown?

What is markdown?

- The `ctrl+B` of plain text
- Many variants, modern markdown father:
  - https://daringfireball.net/projects/markdown/
- Lots of variants, but same idea: plain-text readable markup
  - MultiMarkdown
  - Github-flavored markdown
  - ReST
  - TeX
- pandoc has it's own special features
- General concept: think like HTML or Word "styles"

Markdown crash course

Let's walk through an example together

http://bit.ly/1qIyQbv[2]

[2] http://www.unexpected-vortices.com/sw/gouda/quick-markdown-example.html

Title

Pandoc allows special syntax on the first three lines for document metadata.

```
% Title
% Author
% 2014-04-11
```

Or YAML metadata blocks.

Headers

Two ways to make headers, think <h1>, <h2>, ... levels.

```
An h1 header
============

An h2 header
------------

# Also h1 header

## Also h2 header
```

Paragraphs

```
Paragraphs are separated by a blank line.
I like starting new sentences on a new line.

It's odd, I know.
```

Paragraphs are separated by a blank line. I like starting new sentences on a new line.

It's odd, I know.

Formatting

```
Formatting is easy: *italics*, **bold**, `monospace`,
~~strikethrough~~.

Also H~2~O and Na^+^.
```

Formatting is easy: *italics*, **bold**, `monospace`, ~~strikethrough~~.

Also H_2O and Na^+.

Lists

```
- Item 1
    1. Sub item 1

        Still sub item 1
- Item 2
```

- Item 1

    1. Sub item 1
       Still sub item 1

- Item 2

Block quotes

```
> Block quotes are
> written like so.
>
> They can span multiple paragraphs,
> if you like.
```

> *Block quotes are written like so.*
> *They can span multiple paragraphs, if you like.*

Code Sample

Code samples start with three ‘ characters or three ~ or are indented 4 spaces, and can include the code style.

````
```r
hist(rnorm(100))
```
````

Tables

Tables can look like this:

```
size  material      color
----  ------------  ------------
9     leather       brown
10    hemp canvas   natural
11    glass         transparent
```

size	material	color
9	leather	brown
10	hemp canvas	natural
11	glass	transparent

Tables 2

Or they can also look like this:

```
|size | material     |  color         |
|:----|:-------------|---------------:|
| 9   |  leather     |   brown        |
| 10  |  hemp canvas |   natural      |
| 11  |  glass       |   transparent  |

Table: This table has a caption.
```

size	material	color
9	leather	brown
10	hemp canvas	natural
11	glass	transparent

Table 2: This table has a caption.

Links

```
There are a [couple] of ways to [make][foo]
a [link](http://bing.com).
<http://garrickadenbuie.com>

[couple]: http://google.com
[foo]:    http://xkcd.com
```

There are a couple of ways to make a link.
http://garrickadenbuie.com

Footnotes

```
Footnotes are very similar to links[^disclaimer].

[^disclaimer]: Don't believe everything this guy says.
```

Footnotes are very similar to links[3].

[3] Don't believe everything this guy says.

LaTeX Math

```
Inline math equations go in like so: $\omega = d\phi / dt$.
Display math should get its own line and be put in in
double-dollar signs:

$$I = \int \rho R^{2} dV$$
```

Inline math equations go in like so: $\omega = d\phi/dt$. Display math should get its own line and be put in in double-dollarsigns:

$$I = \int \rho R^2 dV$$

pandoc

Basic pandoc commands

Check out http://johnmacfarlane.net/pandoc/demos.html.

1. Markdown to PDF

```
$ pandoc text.md -o text.pdf
```

2. Markdown to Word

```
$ pandoc text.md -o text.docx
```

3. Markdown to Slides

```
$ pandoc -t beamer --template=mybeamer.template
    text.md -o text.pdf
```

Pandoc-syled citations

- `@<citekey>` – eg. `@Martinez2012`
- `[@<citekey>; @<citekey>]` – `[@smith04, @Martinez2012]`
- Add `# References` to the end of your document.

```
Blah blah [see @doe99, pp. 33-35; also @smith04, ch. 1].
Smith says blah [-@smith04].
@smith04 says blah.
```

Processing citations

Two elements:

1. BibTeX file
2. Citation style `.csl`
    - http://zotero.org/styles

```
$ pandoc text.md -o text.pdf
    --bibliography=/path/to/library.bib
    --csl=/path/to/ieee.csl
```

Keep your `.csl`'s and templates somewhere common.

I use `~/.pandoc/`.

knitr

You already know everything... almost

The easiest way to get started is in R Studio.

Just select `New > R Markdown`.

To tell **knitr** to process code, just add `r` or `{r}` after code-delimiting markdown. You can have *inline* code that runs inside normal `inline code` areas.

You can also have entire code blocks that run R code, called chunks. It's best to keep chunks limited to one or grouped outputs (i.e. one table or figure).

Quick example

````
Inline code evaluations looks like this.
The mean of the sample was `r mean(rnorm(100, mean=10)`
````

````
```{r chunk-name, <chunk-opts>}
hist(rnorm(100))
```
````

Best practices

1. After your document metadata, start with a setup chunk.
   - Use this chunk to set global **knitr** options and load packages.
   - Keep data loading and global functions in separate `.R` files and source them here.
2. Give chunks names for easier navigation
3. Try to keep chunks self-contained. Inter-chunk dependencies get hairy when debugging.

Some important chunk options

Best reference is at http://yihui.name/knitr/options.

Option	Meaning
`echo`	Include R source code in output?
`results`	Options about outputting results
`error`	Hard-fail if error?
`include`	Include any output?
`cache`	Save code chunk results?

A simple example

Grab file from http://bit.ly/USFCodeCamp2014 and switch to R Studio.

Show and tell

Thanks

Contact

- http://garrickadenbuie.com
- [@grrrck](http://twitter.com/grrrck)
- gadenbuie@mail.usf.edu