

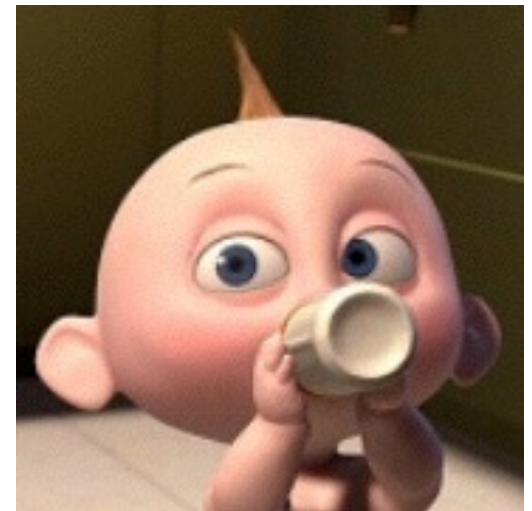
Introduction to Reproducible Research in Bioinformatics

Nan Xiao @road2stat

CRI Bioinformatics Workshop

About me

- <http://nanx.me>
- Statistics background
- Previous experience in statistical machine learning , systems pharmacology, translational bioinformatics
- R developer (7 R/Bioconductor packages; 4 web applications; 4 translated books)



Agenda

- Concept
- Principles
- Tools
- Challenges

We love copy & paste



Data updated today ...

Concept

- Allowing other researchers to replicate your (computational) analysis of the data
- Reproducibility doesn't ensure **correctness**, but still helpful
- Not only required in **bioinformatics** research, but also required in **statistical** research

Why is RR important?

- Reduce (honest) mistakes
- Improve productivity for the long run
- More likely to be used, extended, and cited

General Principles of RR

- Keep track of how every result was produced:
 - Avoid manual data manipulation
 - Version control all custom code
 - Provide public access to data and code

One Principle:
Everything Automated with Code

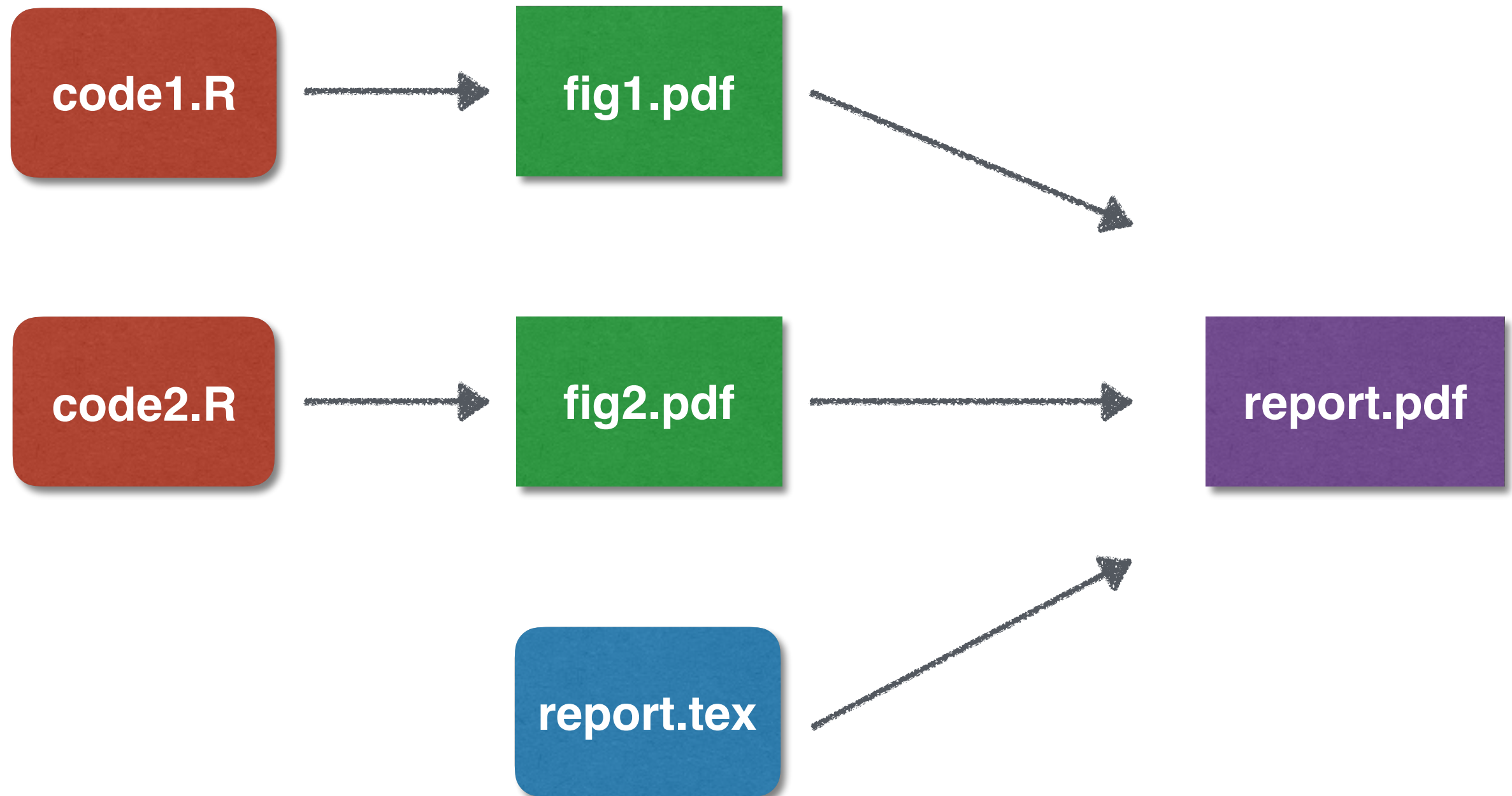
Tools

1. Workflow automation: GNU make + CLI tools instead of GUI tools; Workflow & pipeline systems
2. R / Python packages instead of code snippets
3. knitr / IPython Notebook + Markdown instead of Word
4. Code version control: git / GitHub
5. Package dependency management: packrat / virtualenv
6. System dependency management: Docker

1. Make & its friends

- Make can be used for computational project workflow automation
- Organises code & data dependencies naturally
- Works seamlessly with Linux/Unix CLI tools, such as sed, awk, and many others

Example Makefile



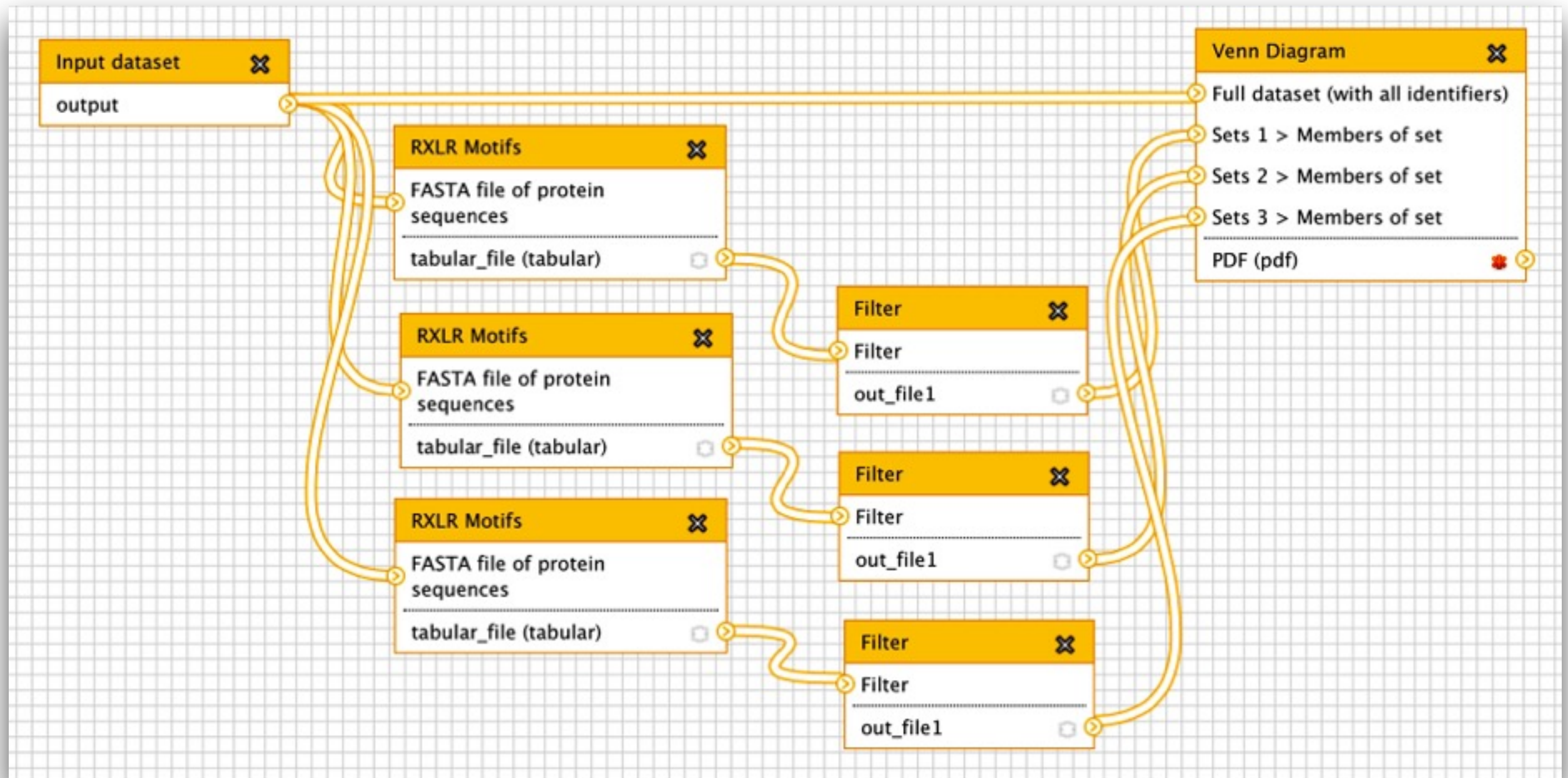
Example Makefile

```
# Example Makefile for a LaTeX report
report.pdf: report.tex img/fig1.pdf img/fig2.pdf
pdflatex report

# Run R code to reproduce figures
img/fig1.pdf: Rcode/code1.R
cd Rcode; R CMD BATCH code1.R code1.Rout
img/fig2.pdf: Rcode/code2.R
cd Rcode; R CMD BATCH code2.R code2.Rout
```

Workflow Systems

- Galaxy
- bpipe
- systemPipeR
- Rabix
- ... and many others:
- <https://github.com/pditommaso/awesome-pipeline>



Galaxy Workflow

Projects ▾
Data ▾
Tools & Pipelines ▾

brandi ▾

Back to Pipeline List

Revision 1 ▾
Publish
Delete
Clone ▾
Run

RNA-Seq Alignment - TopHat

Created by brandi. Last updated by brandi on May 4, 2015.

Align sequence reads from RNA-Seq experiments using the split-read aligner, TopHat.

```

graph LR
    Reads((Reads)) --> TopHat((TopHat))
    RefGenome((Reference Genome)) --> TopHat
    TransAnnotations((Transcript Annotations)) --> TopHat
    TopHat --> UnmappedBAM((Unmapped Reads (BAM)))
    TopHat --> AlignedBAM((Aligned Reads (BAM)))
    TopHat --> BAMToolsIndex((BAMTools Index))
    TopHat --> AlignedBAI((Aligned Reads Index (BAI)))
    
```

Apps

Change Log

Show All ▾

Collapse ▾

TopHat (2.0.4) ▾

Mate inner distance	75
Mate standard deviation	20
Minimum anchor length	8
Splice mismatches	0
Minimum intron length	70
Maximum intron length	500000
Maximum insertion length	3
Maximum deletion length	3
Integer quals	No value
Maximum multihits	20
Report secondary alignments	False
No discordant	False
No mixed	False

Pipeline in NCI Cancer Genomics Cloud
<http://www.cancergenomicscloud.org>

If you want to go beyond CLI and Galaxy ...
R / Python packages can save the day.

2. R Packages

- readr: data loading
- httptr: web scraping
- tidyr: data cleaning
- dplyr: data mangling
- stringr: string data
- lubridate: time data
- ggplot2: visualization
- devtools: package dev
- roxygen2: documentation
- testthat: unit testing



knitr

3. knitr

- knitr report = code + text

Let's explore our ChIP-seq's peak length:

```
```{r}  
counts = read.table("peaks.broadPeak")
hist(counts$end - counts$start)
```
```

The median of the peak length is

```
`r median(counts$end - counts$start)`.
```

Demo with RStudio

- RStudio
- knitr + RMarkdown
- Analyze ChIP-seq peak length distribution

4. Code Version Control

The screenshot shows a GitHub repository page for 'road2stat / liftr'. The page displays the commit history, with the latest commit 'v0.3 - submitted to CRAN' by 'road2stat' on Oct 11. The commit message is 'v0.3 - submitted to CRAN'. The commit details show 1 parent (95a4ab5) and the commit hash c43c74483c3ecfb282f41469162c7362adc2ef94. The commit shows 9 changed files with 45 additions and 21 deletions. The diff view is shown, highlighting changes to the DESCRIPTION file. The diff shows the version number updated from 0.2 to 0.3, and the date updated from 2015-07-30 to 2015-10-10.

road2stat / liftr

Unwatch 2 Unstar 4 Fork 1

Code Issues 13 Pull requests 0 Wiki Pulse Graphs Settings

v0.3 - submitted to CRAN [Browse files](#)

road2stat committed on Oct 11 1 parent 95a4ab5 commit c43c74483c3ecfb282f41469162c7362adc2ef94

Showing 9 changed files with 45 additions and 21 deletions. [Unified](#) [Split](#)

| 4 | DESCRIPTION | View |
|-----|--|--|
| ... | @@ -1,8 +1,8 @@ | |
| 1 | Package: liftr | 1 Package: liftr |
| 2 | Type: Package | 2 Type: Package |
| 3 | Title: Dockerize R Markdown Documents | 3 Title: Dockerize R Markdown Documents |
| 4 | -Version: 0.2 | 4 +Version: 0.3 |
| 5 | -Date: 2015-07-30 | 5 +Date: 2015-10-10 |
| 6 | Authors@R: c(| 6 Authors@R: c(|
| 7 | person("Miaozhu", "Li", email = | 7 person("Miaozhu", "Li", email = |
| 8 | "miaozhu.li@duke.edu", role = "ctb"), | 8 "miaozhu.li@duke.edu", role = "ctb"), |
| | person("Tengfei", "Yin", email = | person("Tengfei", "Yin", email = |
| | "tengfei.yin@sbgenomics.com", role = "ctb"), | "tengfei.yin@sbgenomics.com", role = "ctb"), |

Easy to see when and where the code was changed

5. Package Dependency

- R: **packrat**
- Python: **virtualenv**
- Manages project package dependency, to make sure every newly created environment has the same versions of packages

6. System Dependency



Challenges

- Allow for some level of interactivity: test & debug (knitr, IPython Notebook)
- Handling large-scale computation (Docker + AWS)
- Dependency deprecation (packrat)
- OS-level reproducibility (Docker)
- Data privacy concerns



liftr.me

Idea

- A framework for dockerizing R markdown documents
- Built-in Rabix (bioinformatics pipelines) support
- Reproducible **bioinformatics** and **statistical** analysis



liftr workflow

`liftr("foo.Rmd")` → `drender("foo.Rmd")` → Read and share!

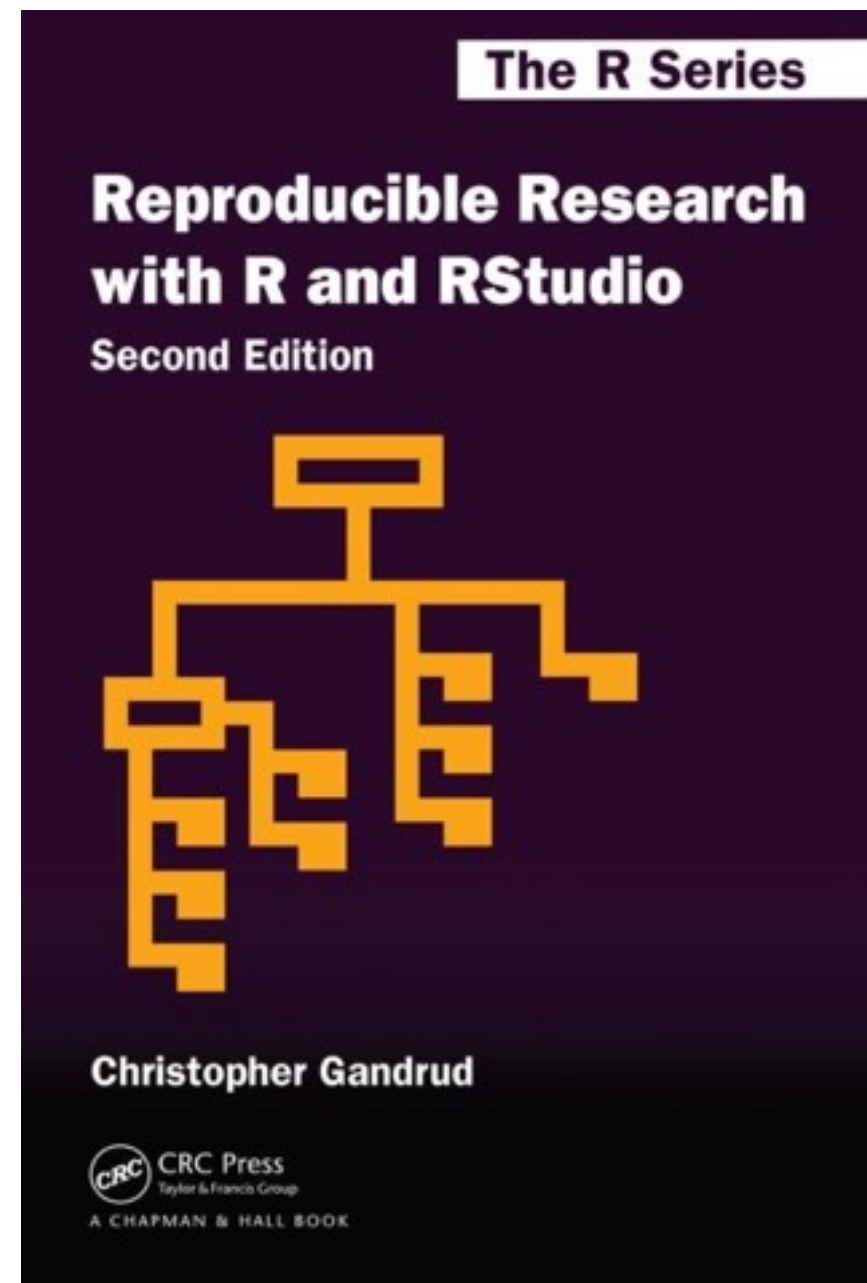
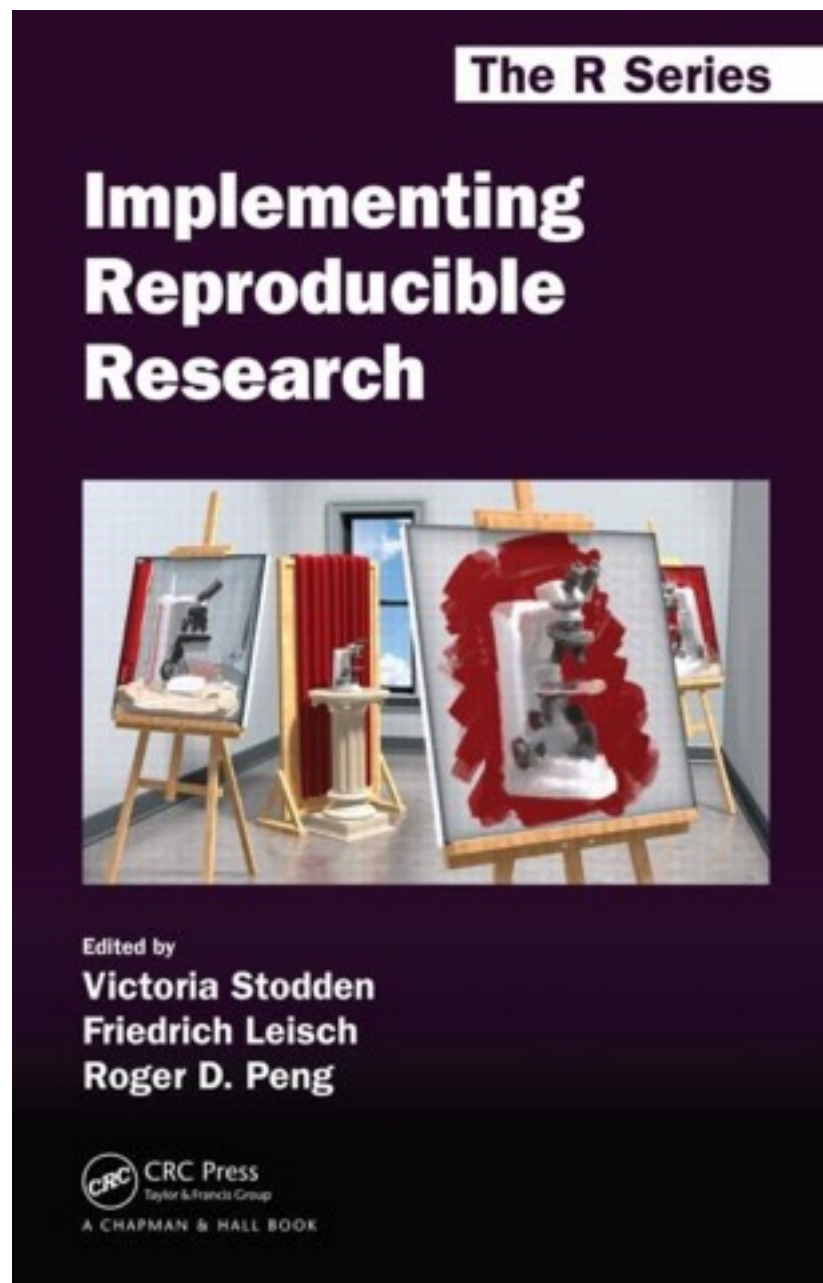


Rmd Documents
with ``liftr``
options in metadata

Generated Dockerfile
(Rabixfile)

Rendered HTML/
PDF/Docx Reports

Further Readings



Q & A

nanx.me

nanx@uchicago.edu