Dynamic Documents for Your Research Workflow

Fernando Hoces de la Guardia
BITSS

BITSS Annual Meeting, December 2017

Dynamic Documents for computational reproducibility

- Based on principles of *literate programming* aims at combining code and paper in one single document
- Best framework to achieve the holy grail of **one-click reproducible workflow**
- Best two current implementations: `RMarkdown (R) &` `Jupyter (Python)`. `Stata` is cathching up (more the end)

Currently code and narrative components live in separate universes
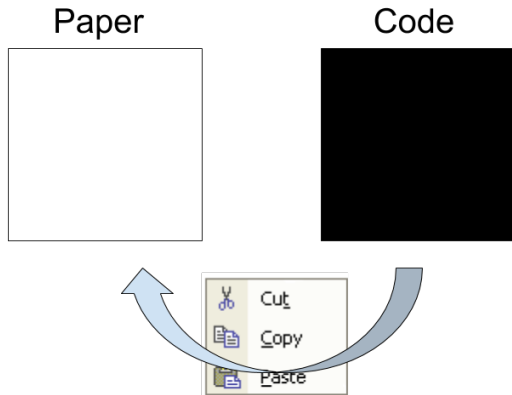
Paper

Code

Dynamic Documents: integrate the two universes!

Paper + Code

Dynamic Documents: A Reciepe

- ► 1 simple language that can combine text and code: `Markdown`
- ► 1 statistical package to do the analysis (R, Python, 3S's?)
- ► 1 machinery to combine analysis and text to create a single output: `Pandoc`
- ► [Optional-but-not-really] 1 program to bring all the elements together: RStudio/RMarkdown, Jupyter

For our excercise: R Markdown

- R: **open source** programming language design for statistical analysis.
- RStudio: free software that provides and Integrated Development Enviroment (IDE)
- RStudio combines all together R + Markdown + Pandoc to produce multiple outputs

R Markdown
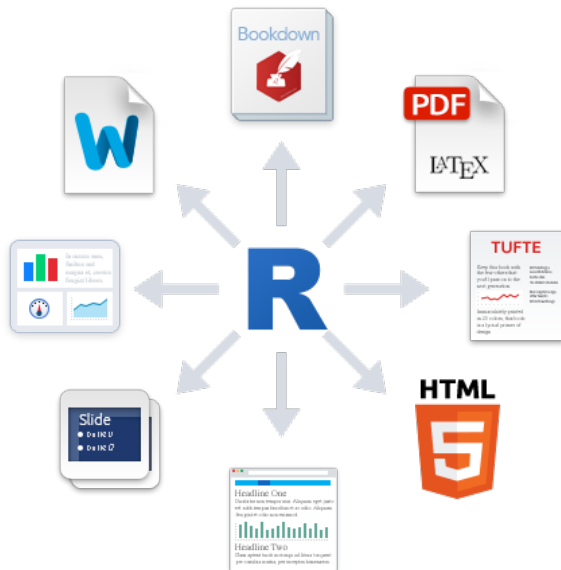
Slide with Bullets

- Bullet 1

Slide with Bullets

- Bullet 1
- Bullet 2

Slide with Bullets

- Bullet 1
- Bullet 2
- Bullet 3

Slide with R Output

```r
summary(cars)
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

Slide with Plot