



Literate programming with {Rmarkdown} and {Quarto}

9 Mars 2023

Nikita Gusarov
University Grenoble Alpes
nikita.gusarov@univ-grenoble-alpes.fr

1 Document creation

2 Different approaches

3 Workflow pipeline

4 Practical part

5 Alternatives

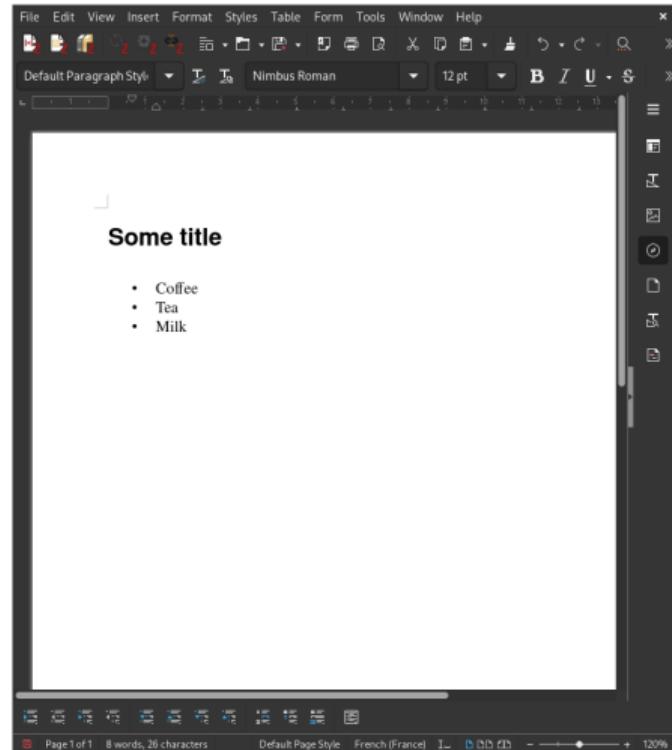
6 References

Document creation

Previously

(Microsoft) Office

- Non-free editing software
- Proprietary format
- Visual editing
- Non-uniform rendering



Beyond the document creation

General drawbacks

- Sharing and collaboration
- Version control
- GitHub / GitLab integration

Prone to errors

- Zeeberg et al. (2004)
- McCullough and Wilson (2005)
- McCullough and Heiser (2008)
- Fetzer and Graeber (2020)

The screenshot shows a Microsoft Excel spreadsheet titled "historical_DXY_XPAR_EUR". The data is presented in a table with columns labeled A through G. Column A contains dates from 2000/01/03 to 2000/02/16. Columns B through G represent financial metrics: Open, High, Low, Close, Volume, and Currency (EUR). The data shows a general upward trend in the exchange rate over the period.

A	B	C	D	E	F	G	
1 Date	B	C	D	E	F	G	
2 2000/01/03 01:00	13.2	14	13.13	13.6	773490	EUR	
3 2000/01/04 01:00	13.38	13.5	12.26	12.26	1220520	EUR	
4 2000/01/05 01:00	12	12.11	11.16	11.31	1214290	EUR	
5 2000/01/06 01:00	11.31	11.58	11.31	11.45	1190945	EUR	
6 2000/01/07 01:00	11.4	12.4	12.11	11.36	12	1031900	EUR
7 2000/01/10 01:00	12.4	12.7	12.2	12.2	1317755	EUR	
8 2000/01/11 01:00	12.1	12.26	11.5	11.98	1317435	EUR	
9 2000/01/12 01:00	11.69	11.84	11.36	11.52	958610	EUR	
10 2000/01/13 01:00	11.42	12.1	11.42	11.98	1382560	EUR	
11 2000/01/14 01:00	12.2	12.8	12.03	12.75	2061130	EUR	
12 2000/01/17 01:00	12.9	12.96	12.36	12.48	1179255	EUR	
13 2000/01/18 01:00	12.5	12.58	12.22	12.5	629710	EUR	
14 2000/01/19 01:00	12.1	12.57	12.1	12.55	565405	EUR	
15 2000/01/20 01:00	12.5	13.1	12.3	13.1	1938700	EUR	
16 2000/01/21 01:00		13.13	12.8	13.38	1294820	EUR	
17 2000/01/24 01:00	13.38	13.38	12.24	12.24	712325	EUR	
18 2000/01/25 01:00	12.3	12.92		12.27	905880	EUR	
19 2000/01/26 01:00	12.98	14.2	12.9	14.2	1174590	EUR	
20 2000/01/27 01:00		14	15.13	8	15	820310	EUR
21 2000/01/28 01:00	14.02	14.86	14.02	14.62	624280	EUR	
22 2000/01/31 01:00	14.4	14.52	13.74	14.37	561565	EUR	
23 2000/02/01 01:00	13.8	14.52	12.94	13.46	827685	EUR	
24 2000/02/02 01:00	13.72	14.36	13.59	14.06	1192805	EUR	
25 2000/02/03 01:00	14.34	15.03	14.02	15.03	1098780	EUR	
26 2000/02/04 01:00	15.03	15.96	15.01	15.2	772055	EUR	
27 2000/02/07 01:00		15.16	15.59	15.15	985870	EUR	
28 2000/02/08 01:00	16.46	17.54	15.24	17.37	2601750	EUR	
29 2000/02/09 01:00	19.1	19.04	17.2	18.24	3679635	EUR	
30 2000/02/10 01:00	17.2	18.98		17.18	44	2627980	EUR
31 2000/02/11 01:00	18.7	19.4	18.2	18.8	1724130	EUR	
32 2000/02/14 01:00	19.4	19.78	18.52	18.52	1342470	EUR	
33 2000/02/15 01:00	19.58	19.85	18.7	18.8	1700860	EUR	
34 2000/02/16 01:00	19.56	21.56	19.2	21.56	3038565	EUR	

abstract_wctr.pdf 96,0% ▾

WTP QUALITY ESTIMATES IN COMMUTE MODE CHOICE:
ISSUES OF PERFORMANCE COMPARISON UNDER SAMPLE SIZE
AND BALANCE IMPACTS.

Nikita Gusarov *
GAEI, Univ. Grenoble Alpes, CNRS, INRAE, Grenoble INP
38000 Grenoble, France
nikita.gusarov@univ-grenoble-alpes.fr

Iragaël Joly
GAEI,
Univ. Grenoble Alpes, CNRS, INRAE, Grenoble INP
38000 Grenoble, France
iragael.joly@grenoble-inp.fr

Pierre Lemaire
G-SCOP
Univ. Grenoble Alpes, CNRS, Grenoble INP
38000 Grenoble, France
pierre.lemaire@grenoble-inp.fr

Extended abstract

Keywords Willingness to Pay - Mode Choice - Model Performance Comparison - Discrete Choice Modelling - Econometrics - Machine Learning

Word count: 991 words in body, 5975 characters in body, 6910 characters in body (including spaces)

In economic studies, the Willingness to Pay (WTP) is the maximum amount an individual is willing to pay for a particular attribute of goods or services. In transportation studies, popular estimations of WTP are Value of Time (VOT) or Value of Comfort (VOC). The WTP elicitation lies at the heart of many various tasks in the transportation mode choice analysis: adoption of emerging sustainable transportation modes (Bahl et al., 2021), attitudes to resilience and sharing in transportation (Ho et al., 2018; Kang et al., 2021), preferences for emergency services (Ardeshiri et al., 2021; Merkert et al., 2022).

There exist multiple concurrent ways to deduce WTP from the data, most of which rely on the *Random Utility Maximisation* (RUM) framework (McFadden, 1974). With time the number of available models and estimation techniques increases, but many of them are still RUM-compliant. Following McFadden (1981) the RUM-compliance translates in the hypothesis that the set of the choice probabilities of the various alternatives to be selected is fully explained by the transformation of the utility function of all potential alternatives (Hensher and Greene, 2002). The family of RUM-compliant models was recently extended by a number of more advanced models, as the mode choice modelling is on the verge of transition to incorporation of novel *Machine Learning* (ML) based analysis methodologies. The ML or *data driven* methodology focuses on the predictive qualities. In contrast, the econometrics or *theory drives* methodology attempts to decipher the underlying properties of the data. The hypothetico-deductive approach of econometrics allows the prediction of economic variables by theory independent of the model structures, on which ML tools do not depend. The transportation studies and mode choice modelling applications in particular were among the first to attempt the implementation of ML methodology for typically *theory driven* tasks. One can observe a

PDF

- Open-source format
- Software agnostic
- Uniform rendering

Rmarkdown

- Markup editing
- Simple version control
- Code integration
- Data management

Different approaches

What You See Is What You Get

Editing content in a form that is identical to its appearance when displayed as a finished product

Examples

- Microsoft Office
- LibreOffice
- Apache OpenOffice
- GNU TeXmacs

Markup languages

Editing content in a plain text format, where the document contains a set of rules that determine its appearance when displayed a finished product.

Examples

- Groff (Troff, Roff)
- TeX (LaTeX)
- HTML
- XML
- Markdown

Workflow pipeline

Pipeline outline

Requirements

- Possibility to render PDF (and potentially other formats)
- Simple citations management
- Easy syntax
- Integration with other activities
 - Code execution
 - Scripting

Pipeline outline

Requirements

- Possibility to render PDF (and potentially other formats)
- Simple citations management
- Easy syntax
- Integration with other activities
 - Code execution
 - Scripting

Solutions

- Pandoc conversion + PDF LaTeX engine
- BibTeX support

- Wide variety of supported formats
- Possibility to combine *markdown* with other markup syntax formats (LaTeX, HTML, ...)
- Custom templates support
- Document composition
 - In-document YAML configuration
 - External features

PDF (and other formats) rendering

Pandoc fully-supported formats

- Markdown
- RTF, docx, ODT
- HTML
- EPUB
- Roff
- LaTeX, BibTeX
- OPML
- Jupyter notebooks

Pandoc output formats

- Chunked HTML
- LaTeX Beamer
- Microsoft PowerPoint
- Slidy
- reveal.js
- S5
- OpenDocument XML
- GNU TexInfo

PDF (and other formats) rendering

Pandoc fully-supported formats

- Markdown
- RTF, docx, ODT
- HTML
- EPUB
- Roff
- LaTeX, BibTeX
- OPML
- Jupyter notebooks

Pandoc output formats

- Chunked HTML
- LaTeX Beamer
- Microsoft PowerPoint
- Slidy
- reveal.js
- S5
- OpenDocument XML
- GNU TexInfo

Pipeline outline

Requirements

- Possibility to render PDF (and potentially other formats)
- Simple citations management
- Easy syntax
- Integration with other activities
 - Code execution
 - Scripting

Solutions

- Pandoc conversion + PDF LaTeX engine
- BibTeX support
- Markdown

Key advantages

- More simple syntax in comparison with pure LaTeX, HTML or Groff
- Best compatibility with Pandoc for conversion into other formats



Easy syntax

LaTeX

```
\begin{itemize}
  \item{Coffee}
  \item{Tea}
  \item{Milk}
\end{itemize}
```

HTML

```
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

Markdown

```
- Coffee
- Tea
- Milk
```

Pipeline outline

Requirements

- Possibility to render PDF (and potentially other formats)
- Simple citations management
- Easy syntax
- Integration with other activities
 - Code execution
 - Scripting

Solutions

- Pandoc conversion + PDF LaTeX engine
- BibTeX support
- Markdown
- R
 - knitr



knitr

- Executes code inside .Rmd document
- Appends the results after the code blocks
- Generates .md document

```
'''{r}
x = rnorm(100); y = 1:100
plot(x, y)
'''
```

Pipeline outline

Requirements

- Possibility to render PDF (and potentially other formats)
- Simple citations management
- Easy syntax
- Integration with other activities
 - Code execution
 - Scripting

Solutions

- Pandoc conversion + PDF LaTeX engine
- BibTeX support
- Markdown
- R
 - knitr
 - rmarkdown

Scripting

Inside document

- Add yaml part

```
---
```

```
title: "Some title"
author: J. Doe
params:
  n: 1000
---
```

Inside body

- Call `params` list to retrieve the parameters

```
'''{r}
n = params$n
x = rnorm(n); y = 1:n
plot(x, y)
'''
```

- knitr (embedded code execution)
- R front-end to pandoc features
- Support for markdown syntax
- Extended YAML configuration
- Wide variety of preconfigured pandoc templates
- Notebook oriented workflow (alternative to Jupyter)

Potential drawbacks

Different flavours of markdown

- CommonMark
- CriticMarkup
- ExtraMark
- GitHub Markdown
- Pandoc's Markdown
- ...

Dependencies to configure

- Pandoc - <https://pandoc.org/installing.html>
- PDF LaTeX engine - <https://yihui.org/tinytex/>
(ex: MikTeX, TinyTeX)
- R - <https://www.r-project.org/>

Extensions

- `kable` and `kableExtra` - toolset for `data.frame` display
- `bookdown` - extra features for academic and professional writing (ex: books and manuals)
- `rticles` - preconfigured templates for scientific articles and conferences
- `blogdown` - blog editing with Hugo
- Python, Julia or C++ for other code block types support
- `htmlwidgets` - bindings R to JavaScript libraries.
- `learnr` - interactive tutorials and quizzes
- `shiny` - interactive documents and reports

For further reading

Manuals

- Xie, Dervieux, and Riederer (2020)
- Mailund (2019)

Potential errors

- Li, Liu, and Meng (2021)

Practical part

Dependencies to configure

- Pandoc - <https://pandoc.org/installing.html>
- PDF LaTeX engine - <https://yihui.org/tinytex/>
(ex: MikTeX, TinyTeX)
- R - <https://www.r-project.org/>

Getting started

- Run your preferred IDE / editor
- Create a new test.Rmd document to experiment with
- Cheat sheets available at
<https://www.rstudio.com/resources/cheatsheets/>

YAML configuration

At the top of the document the YAML part is placed, which communicates parameters to pandoc and R:

Example

```
---
```

```
title: Some title
author: J. Doe
date: March 2023
output:
  pdf_document:
    toc: false
    fig_caption: true
---
```

Basic syntax

- *italics* = *italics*
- **bold** = **bold**
- hyperlink = [hyperlink] (<https://www.rstudio.com>)
- images = ! [image description] (./path/to/image.png)
- lists
 - 1. list
 - * with
 - * nested
 - 2. elements
- headers = # Header
- unnumbered header = # Header {-}

Basic syntax

- quotation => quotation
- footnote = ^ [footnote]
- *inlinemaths* = \$inline maths\$
- maths equations

```
$$  
X = \frac{1}{\sigma}  
$$
```

For full guide see here <https://bookdown.org/yihui/rmarkdown/>

Code integration

Inline code

```
'r x = 10; print(x)'
```

Separate code blocks

```
'''{r, include = TRUE}
x = 10
print(x)
'''
```

Other languages

You can get the available engines with the command:

```
names(knitr::knit_engines$get())
```

Using other languages

```
'''{python, engine.path = '/usr/bin/python3'}
x = 10
print(x)
'''
```

Using custom templates

Create a sample template for LaTeX output and a .Rmd document:

template.tex

```
\documentclass{article}
$if(encoding)$
\usepackage[$encoding$]{inputenc}
$else$
\usepackage[utf8]{inputenc}
$endif$
\begin{document}
$body$
\end{document}
```

somefile.Rmd

```
---
encoding: utf8
output:
  pdf_document:
    template: template.tex
---
```

Some text in body.

Rendering

To convert the document one can:

1. Use the integrated features of the IDE

- Ctrl + Shift + K in VS Code
- knit button in RStudio

2. Call the rendering function directly

```
rmarkdown::render(  
  "path/to/the/file.Rmd"  
)
```

For those who want to see details

Create a new `test.md` markdown document to experiment with.

```
test.md
---
title: Some title
author: J. Doe
---
Some text in body.
```

Convert it to `.tex`

```
pandoc test.md -f markdown -o test.tex -t pdf
```

Alternatives

Language specific

- Pmarkdown (seems to have lost support)
- Jmarkdown
- Jupyter (notebooks)

Language agnostic (standalone)

■ Quarto

- Mostly back-compatible with .Rmd format
- Has dedicated extensions for VS Code, Emacs, etc.
- Specification of knitr options in YAML
- Some packages break

References

- Fetzer, Thiemo, and Thomas Graeber. 2020. “Does Contact Tracing Work? Quasi-Experimental Evidence from an Excel Error in England.” December 15, 2020.
<https://doi.org/10.1101/2020.12.10.20247080>.
- Li, Penghui, Yinxi Liu, and Wei Meng. 2021. “Understanding and Detecting Performance Bugs in Markdown Compilers.” In *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 892–904.
<https://doi.org/10.1109/ASE51524.2021.9678611>.
- Mailund, Thomas. 2019. *Introducing Markdown and Pandoc: Using Markup Language and Document Converter*. Berkeley, CA: Apress.
<https://doi.org/10.1007/978-1-4842-5149-2>.
- McCullough, B. D., and David A. Heiser. 2008. “On the Accuracy of Statistical Procedures in Microsoft Excel 2007.” *Computational Statistics & Data Analysis* 52 (10): 4570–78.
<https://doi.org/10.1016/j.csda.2008.03.004>.
- McCullough, B. D., and Berry Wilson. 2005. “On the Accuracy of Statistical Procedures in Microsoft Excel 2003.” *Computational Statistics & Data Analysis* 49 (4): 1244–52.
<https://doi.org/10.1016/j.csda.2004.06.016>.

References II

- Xie, Yihui, Christophe Dervieux, and Emily Riederer. 2020. *R Markdown Cookbook*. New York: Chapman and Hall/CRC. <https://doi.org/10.1201/9781003097471>.
- Zeeberg, Barry R., Joseph Riss, David W. Kane, Kimberly J. Bussey, Edward Uchio, W. Marston Linehan, J. Carl Barrett, and John N. Weinstein. 2004. “Mistaken Identifiers: Gene Name Errors Can Be Introduced Inadvertently When Using Excel in Bioinformatics.” *BMC Bioinformatics* 5 (1, 1): 1–6. <https://doi.org/10.1186/1471-2105-5-80>.