Applied Data Analysis School

LITERATE PROGRAMMING IN R MARKDOWN

Paulo Guimaraes and Miguel Portela

October 27 and 29, 2020

# install pandoc

[Pandoc](https://pandoc.org/)

[R Markdown - Get Started](https://rmarkdown.rstudio.com/lesson-1.html)

[R Markdown - Reference Guide](https://rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf)

Windows and mac latex: miktex or mactex sumatra or kim

R Markdown cheetsheet: check the Meny Help

# yaml

# references

# pandoc examples

# Introduction

* What is [**Literate programming**?](https://en.wikipedia.org/wiki/Literate_programming)

Literate programming refers to melding a descriptive narrative and computer code into a single document, from which both human-friendly documentation and computer readable files can be created

* Your work should be transparent, easy to update, easy to maintain, and easy to replicate
* Data analysis requires reproducibility of results
* Useful for teaching

# Preliminaries: Markdown

* [**Markdown**](https://en.wikipedia.org/wiki/Markdown) is a lightweight markup language with plain text formatting syntax that was invented by John Gruber
* There are several “flavors” of Markdown
* Markdown files can be created in any text editor
* Markdown files can be easily converted to other formats
* They are text files usually with the extension “.md”
* There are many editors specific for Markdown (eg: [**Typora**](https://typora.io/), [**Mou**](http://25.io/mou/), [**Draft**](https://draftin.com/), etc) but you can use general editors (eg: [**Atom**](https://atom.io/), [**Sublime Text**](https://www.sublimetext.com/))
* Used in popular sites such as [**GitHub**](https://github.com/), [**Reddit**](https://www.reddit.com/),[**Stack Exchange**](https://stackexchange.com/), etc

# Markdown Syntax: Quick Introduction

* Creating Headers

This is header 1  
================

or

Another header  
--------------

* or headers with different levels

# This is Header 1  
...some text...  
## And now Header 2  
...more text...  
### Finally Header 3  
... and a little more text.

# Markdown Syntax (continued)

* Emphasizing text

this is \*Italic\* but \_this\_ also works

this will write in \*\*bold\*\* but you can also \_\_do this\_\_

It is also possible to ~~Strikeout~~ text

* Creating unordered lists (use "\*" “+” or “-”")

\* Point 1  
+ Point 2

# Markdown Syntax (continued)

* You can create structured lists

- Point 1  
 - Point 12  
- Point 2  
 - Point 21

* insert links to text. Simply write

[text](url)

* insert images

![text](file "Description")

* You can add tables
* and even references

# Markdown

* You can find a quick guide for syntax [**here**](https://www.markdownguide.org/cheat-sheet/) or more detailed information [**here**](https://daringfireball.net/projects/markdown/syntax)
* To get an idea how Markdown works you can use an online editor such as [**Dillinger**](https://dillinger.io/) or [**Markdown here**](https://markdown-here.com/livedemo.html)
* and if you need to create tables you can use [**this**](http://www.tablesgenerator.com/markdown_tables) great online tool

# Markdown Syntax (continued)

* Markdown documents are highly flexible because they can incorporate other languages
* An example is . To add code you need to enclose it in “$” If you write

$a^2+b^2=c^2$

you will see

* Markdown also accepts raw HTML code
* You can incorporate code from other languages (eg: Python, Java, R, Stata, etc)
* This is typically done in a “code fence”: lines with three or more backticks or tildes inserted before and after

# Preliminaries: Pandoc

* [**Pandoc**](https://pandoc.org/) is a command line tool to convert across different document formats. Try it online [**here**](https://pandoc.org/try/)
* It supports many formats such as Markdown, HTML, docx, pdf, , etc
* It is simple to use: to convert the file *example1.md* to
  + HTML
* pandoc -s example1.md -o example1.html
  + PDF
* pandoc -s example1.md -o example1.pdf
  + docx
* pandoc -s example1.md -o example1.pdf
* Note: -s stands for source and -o for output

# Markstat

* a user written *Stata* command by German Rodriguez
* lets you embed *Stata code* in Markdown documents
* It produces dynamic presentations in
  + HTML
  + DOCX
  + PDF
  + Beamer
* *Markstat* requires *Pandoc*
* To produce PDF and Beamer files it requires an installation of
* You will also need to copy the file “stata.sty” to the working directory. This file is part of the “sjlatex” package.

# Installing Markstat

* Install in *Stata* by typing

ssc install markstat

* *Markstat* also requires the *whereis* ado. To install it do:

ssc install whereis

* use *whereis* to let *Stata* know where *pandoc* is in your system. For example:

whereis pandoc "c:\pandoc\pandoc.exe"

* if you plan on generating PDF files via you also need to provide the path for *pdflatex*. For example:

whereis pdflatex "C:\MiKTeX 2.9\miktex\bin\pdflatex"

# Marsktat syntax

markstat using filename [, options]

* filename is the name of a Stata-Markdown file. This file should have an extension “.stmd”
* The options **pdf**, **docx**, **slides** and **beamer** identify the type of document to be produced
* **mathjax** renders equations in HTML documents
* **bundle** generates self-contained HTML documents
* **bibliography** is used to create citations
* **strict** - specifies that you are using the strict syntax
* additional information about *markstat* can be found at the author’s website [**here**](https://data.princeton.edu/stata/markdown)

# Adding Stata code to a Markdown document

* Markstat interprets anything after a tab or 4 spaces as Stata code
* It is safer to use “code fences”. This is the **strict** mode
* For Markstat to understand Stata code the opening fence has to be followed by {s} or s
* To turn off echoing Stata the opening fence must be followed by {s/} or s/
* You can add inline code using `s [fmt] expression`
* you can also add Mata code `m [fmt] expression`
* You can even add R code!

# Metadata

* You can add a title, author and date as metadata. Simply start the document as

% Fill in title  
% Fill in Name  
% 23 January 2019

* Or you can use the [**YAML**](https://en.wikipedia.org/wiki/YAML) format. See [**Pandoc User’s Guide**](https://pandoc.org/MANUAL.pdf) for more info. To add title, author and date place at the top of the document

---  
title: Fill in title  
author: Fill in name  
date: 23 January 2019  
---

# Adding citations to the document

* it is possible to add citations from a [**bibtex**](http://www.bibtex.org/) file
* place the “bib” file (say “references.bib”) in the working folder
* modify the YAML by adding a line pointing to the bib file

bibliography: references.bib

* At the bottom of the document add a line

## References

* use the syntax [@key] to identify the reference
* when running markstat add the **bib** option
* For more information check [**here**](https://data.princeton.edu/stata/markdown/citations)

# Advanced use:customizing outputs

* Customizing HTML outputs
  + HTML outputs may be customized using a [“css”](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) file
  + Copy the “markstat.css” file from the installation folder to your current working folder
  + Edit the “markstat.css” file in the working folder
* Customizing DOCX outputs
  + DOCX files may be customized. Instructions are [**here**](https://data.princeton.edu/stata/markdown/docx)
  + Copy the “markstat.docx” file from the installation folder to your current working folder
  + Use *Word* to Edit the “markstat.docx” file in the working folder

# Advanced use:customizing outputs

* The default style for citations is the Chicago Manual of Style author-date format.
* But you can use any style available in Citation Style Language (CSL) in the [Zotero Style Repository](https://www.zotero.org/styles).
* To change style download the “csl” file and add a reference to it in the YAML block. for example

---  
title: Title of my presentation  
author: My name  
date: 23 January 2019  
bibliography: references.bib  
csl: thisstyle.csl  
---

# Other Stata Tools for Reporting

* *dyndoc* (Official Stata) - convert dynamic Markdown document to an HTML file or Word document
* *docx2pdf* (Official Stata) - convert a Word document to a pdf file
* *markdown* (Official Stata) - convert Markdown document to an HTML file
* *dyntext* (Official Stata) - process Stata dynamic tags in text file
* *dynpandoc* - convert file with dynamic tags in one markup format to another using pandoc