

Git and R Markdown

Why they are useful and how to get started

What is Git?

- Distributed version control
- Manages changes without overwriting them

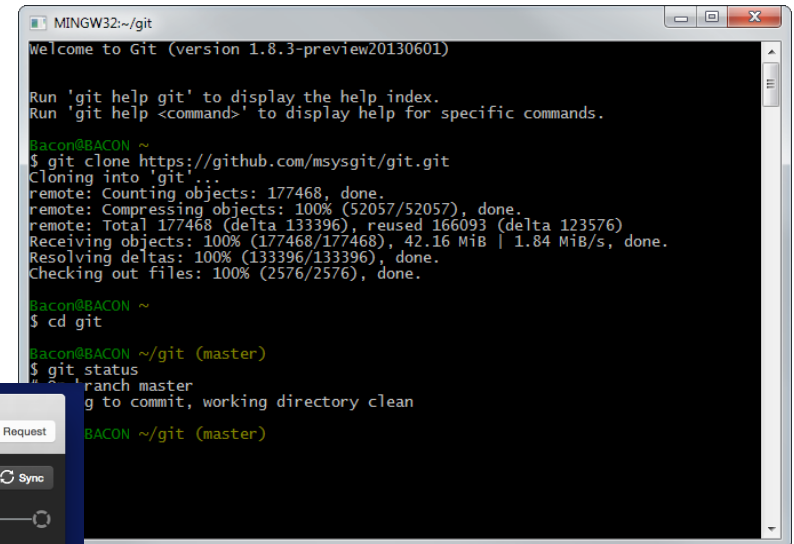
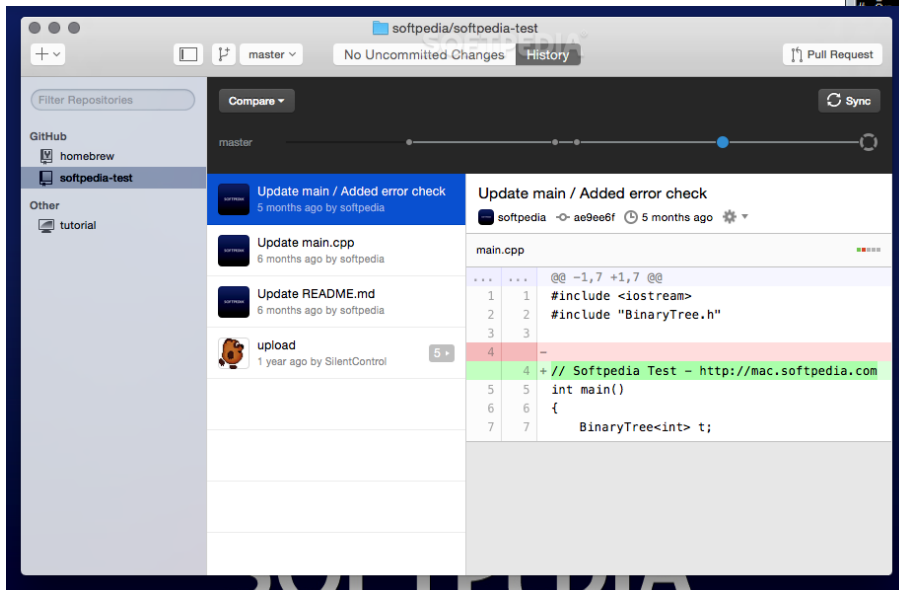
	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT
MESSAGES GET LESS AND LESS INFORMATIVE.



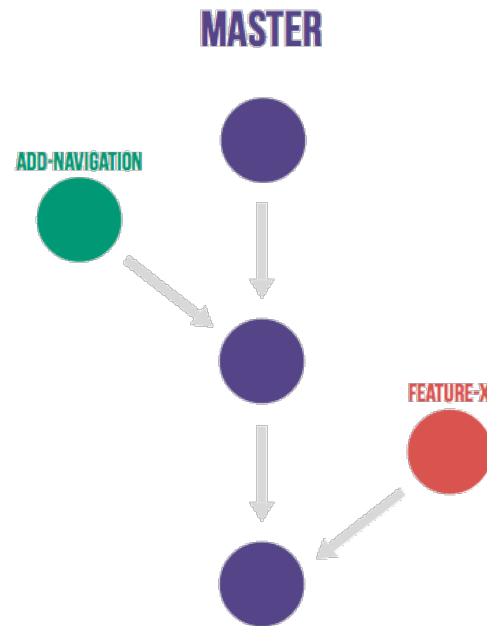
What is Git?

- Distributed version control
- Shell software and GUIs



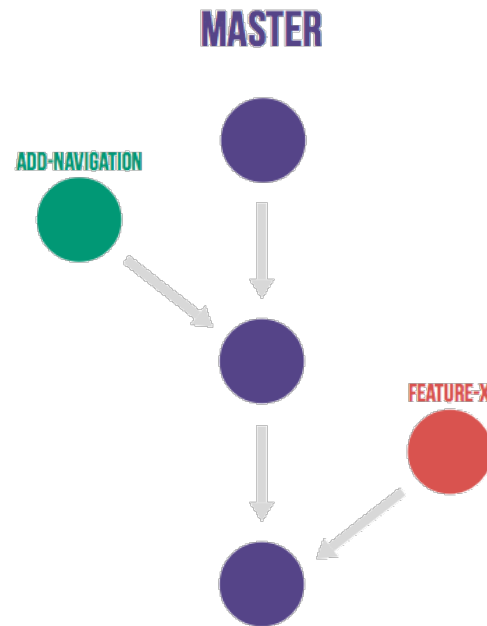
What is Git?

- Distributed version control
- Manages changes without overwriting them
- Local and remote copies



What is Git?

- Distributed version control
- Manages changes without overwriting them
- Local and remote copies
- GitHub: Collaborative platform and hosting service



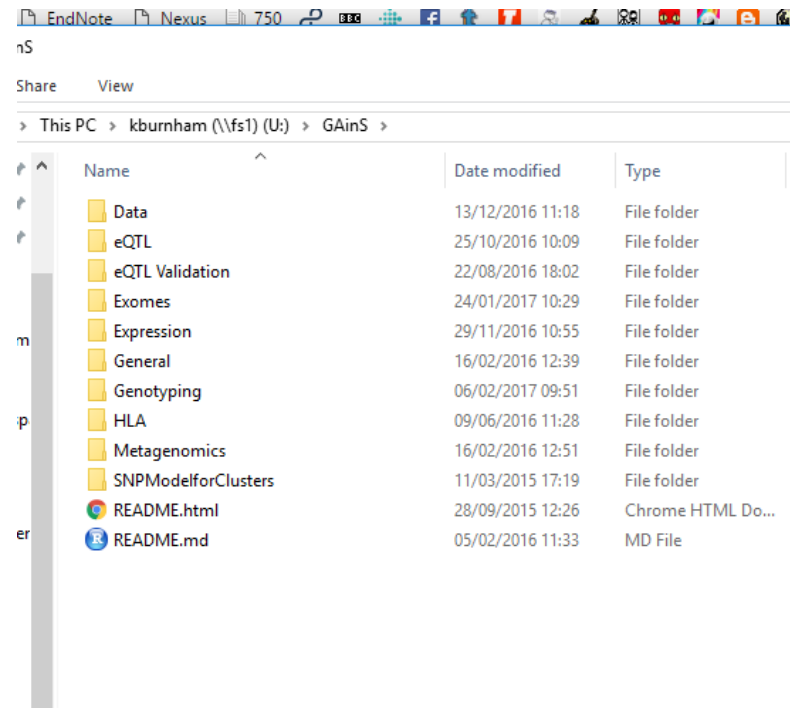
Why use Git?

- Code sharing and publishing, a programmer's Linked In?
- Or: Word track changes + Dropbox shared folders
- Not just for code: any type of file
- Not just for coders: GUIs available
- Manages and stores revisions: filing system for drafts
- Joint projects



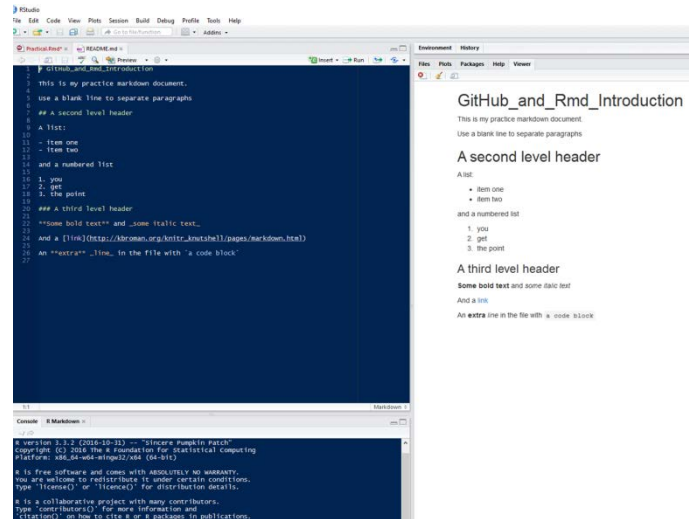
How do you use Git?

- Usual folder structure on your computer



How do you use Git?

- Usual folder structure on your computer
- Do some work (the hardest part)



The screenshot shows the Visual Studio Code editor with a file named 'GitHub_and_Rmd_Introduction.md'. The left pane shows the raw markdown source code, and the right pane shows the rendered HTML output. The source code includes a title, a subtitle, a second-level header, a list, a third-level header, and some bold and italic text. The rendered output shows the corresponding HTML structure with headers, lists, and bold/italic formatting.

```
1 # GitHub_and_Rmd_Introduction
2
3 This is my practice markdown document.
4 Use a blank line to separate paragraphs
5
6 ## A second level header
7
8 A list:
9
10 - item one
11 - item two
12
13 and a numbered list
14
15 1. you
16 2. get
17 3. the point
18
19 ## A third level header
20
21 "some bold text" and "some italic text"
22
23 And a [link](http://bromon.org/knitr-knitrshell/pages/markdown.html)
24
25 An "extra" line in the file with a code block
```

GitHub_and_Rmd_Introduction

This is my practice markdown document.

Use a blank line to separate paragraphs

A second level header

A list:

- item one
- item two

and a numbered list

1. you
2. get
3. the point

A third level header

Some bold text and some italic text

And a link

An extra line in the file with a code block



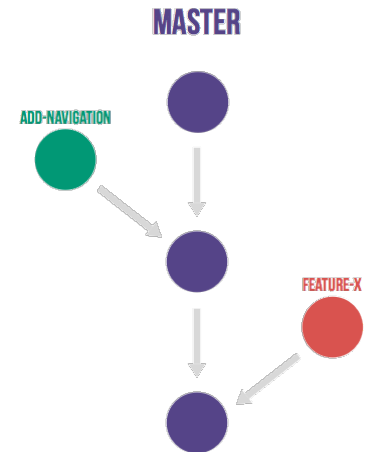
How do you use Git?

- Usual folder structure on your computer
- Do some work
- Save your changes locally; Git notices and tracks them
- Confirm that you want these changes to be remembered
- Synchronise with a central copy of your folder on GitHub



How do you use Git?

- Download from the central copy to another computer (this could be you, or someone else)
- Do some work
- Synchronise (*push*) the changes to GitHub
- *Pull* them down to the first computer



Some terms you will come across

- **Repository** - a directory
- **Remote** - the online copy (also called the **origin**)
- **Clone** - download a whole repository
- **Add** - changes that you **add** are waiting (**staged**) to be stored
- **Commit** - store a snapshot
- **Push** - upload changes
- **Pull** - download changes and merge all together



Some terms you might come across

- **Revert** – reverse previous changes (but still store them)
- **Branch** - a parallel copy of the repository
- **Master** – the main/default branch
- **Fetch** - download changes
- **Checkout** – switch branches or discard changes
- **Merge** - combine branches
- **Fork** – duplicate a repository



What is R Markdown?

- Lightweight markup for interactive documents
- Instruction tags e.g. HTML, LaTeX
 - Word: WYSIWYG
- Readable markup + conversion tool
- Include code (R, Python, SQL) in your text
- Reproducible dynamic documents



Why use Markdown?

- Word + figures = Yuck.
- Generate nice HTML/PDFs/Word docs
- Good integration with GitHub
- REALLY useful for R(/python/bash/SQL) users: integrate code and generate summary docs all in one
- Reproducible and dynamic analysis
- ([Parameters, slideshows, notebooks, websites...](#))



Why use Markdown?

```
<html>  
<body>  
<h1>Title</h1>  
</head>  
</html>
```

Title

```
\documentclass[a4paper,12pt]  
{report}  
\textheight=23.5cm  
\textwidth=15cm
```

```
\section{Title}
```



How do you use R Markdown?

- Open an R Markdown file in R Studio
- *Type as normal*
- Add code in chunks
- Click “Knit” to generate the document
- Open the file created in the same directory

