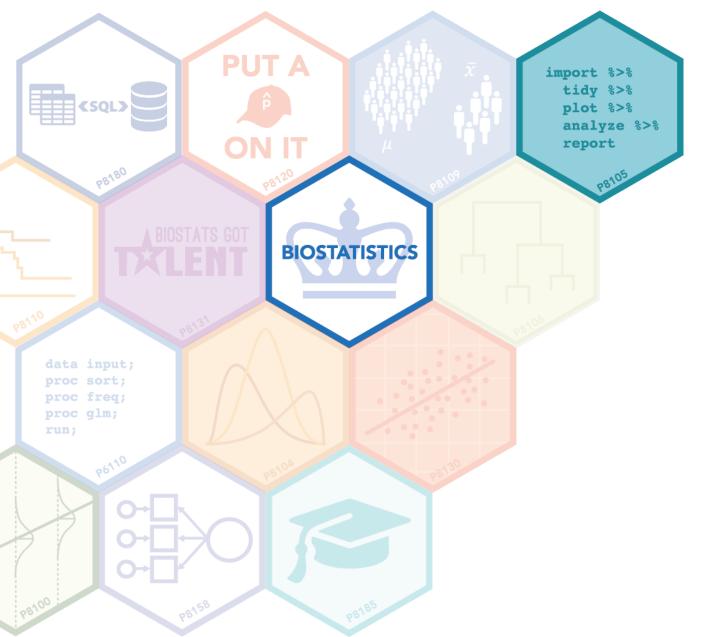
COLUMBIA | MAILMAN SCHOOL UNIVERSITY | of PUBLIC HEALTH



GIT AND GITHUB

Jeff Goldsmith, PhD Department of Biostatistics

Is Git awesome?

• Yes

Is Git awful?

• Also yes

?????

- The good generally outweighs the bad
 - But there is some bad

So ... what is Git?

- Kinda like Google Docs / Dropbox / track changes
- The goal is to avoid this:

	~			ŵ		0	**	Î	Q Search	
		Annelisation			00450000 5		<u>.</u>		•	
Favorites		Application	⊘ ▶	_	_	-		anuscript_JC26.do	DCX	0
Desktop		Drafts			20150330_SM	_	-			0
		Response	Ø 🕨	_	20150406_SM		_		5	0
Documents		Reviewer Comments	s 📀 🕨						Figures_TK_JG.docx	0
😺 Dropbox				_	20150415_SM			AbsMethRes.docx		0
ieffgoldsmith	11							OCX		0
		 20150427_Figure5.docx 20150610_SMARTSKinem_v19_ForJohn.docx 								0
Applications					20150917_SM					ĕ
Devices				_	_		_			ĕ
-		 20150925_SMARTSKinem_v22.docx 20150929_SMARTSKinem_v23.docx 						ĕ		
Remote Disc	5 F				20150929_SM		_			ŏ
Shared	5 1				20150929_SM					ē
<i>_</i>	5 1			_	20151002_SM		_	-		
🎯 All	5 1				20151004_SM					
Tags	2 F				20151009_SM	ARTSKine	m_v28jk	_JG.docx		•
					20151019_SM					0
					20151118_SM	ARTSKine	m_v35.d	ocx		0
					20151210_SM	ARTSKine	m_v36	IG.docx		•
					20151210_SM	ARTSKine	m_v38.d	ocx		<
					20151210_SM	ARTSKine	m_v45_p	oc.docx		0
					20160113_SM	ARTSKine	m_v45.d	ocx		0
				•	20160221_SM	ARTSKine	m_v53.d	ocx		0
					20160224_SM	ARTSKine	m_v54.d	ocx		<
					20160428_SM	ARTSKine	m_v60.d	осх		0
					20160429_SM	ARTSKine	m_v62_1	K_JG.docx		0
					20160801_SM	ARTSKine	m_v64.d	ocx		<
					20160801_SM	ARTSKine	m_v66.d	ocx		0
				_	20161104_SM					0
					20170106_SM					0
				•	20170106_Sup	plementa	al_Materia	als.docx		0

That still doesn't explain it.

- Git watches repositories folders / directories for changes
- It asks that you describe changes when they're made
- It remembers old versions if you need them
- It also keeps an eye out for conflicts, and forces you to resolve them
- It allows multiple people to contribute to the same repository, and does all of the above for everyone at once

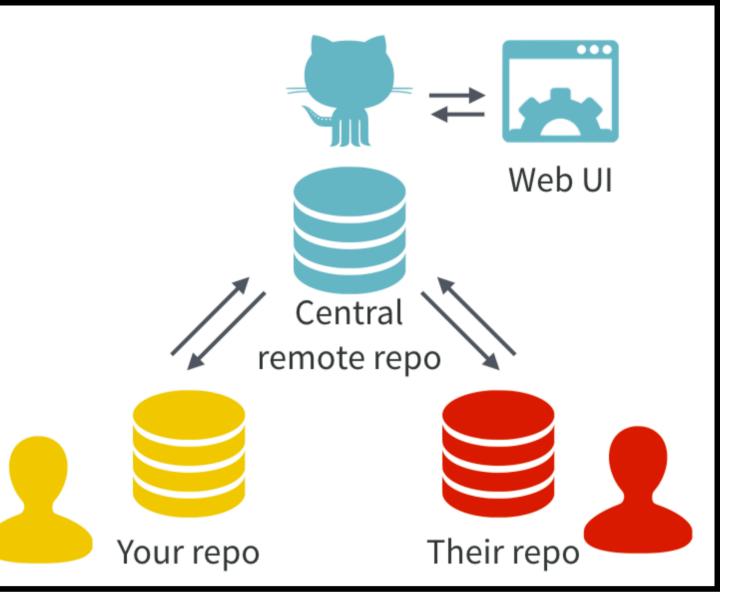
And GitHub?

- Git lives on your computer; GitHub is a web-based platform for storing repositories
 - Think DropBox, but with Git in your folders (watching you)
- GitHub is a great platform for disseminating work
 - You can easily create and host reports; websites; R packages; ...

COLUMBIA UNIVERSITY of PUBLIC HEALTH

And GitHub?

- Git lives on your computer; repositories
 - Think DropBox, but with
- GitHub is a great platform f
 - You can easily create an



"Excuse me, do you have a moment to talk about version control?"

What about RStudio

- Git is something you should be doing, and RStudio tries to make it easy for you to do
- R Projects can initialize Git with a mouse click
- Then, everything in the project is being watched

And a Git client?

- Git is a command-line tool
- Git clients let you do most Git-related stuff in a GUI
 - Git client is to git as RStudio is to R
- RStudio has a bare-bones Git client which will work for most stuff

Workflow

- When starting a new analysis / project / whatever, I
 - Create GH repo
 - Create linked R Project using repo URL
 - Do stuff

"Doing stuff" in a git repo

- You do whatever you would usually do
- Once you've done some amount of stuff, you commit the changes
 - "commit" = "fancy save"
 - Git will keep track of changes between commits
 - Your commit message will summarize what's different
- Then you do more stuff, then you commit, then you do more stuff ...
- Push changes to GitHub more on that soon

"Doing stuff" in a git repo

- You do whatever you would usually do
- Once you've done some amour
 - "commit" = "fancy save"
 - Git will keep track of change
 - Your commit message will su
- Then you do more stuff, then yc
- Push changes to GitHub more



Pros:

- You can revert to earlier commits if you mess something up
- You can quickly review the development process
- You can see what collaborators are doing, where they're doing it, and why
- You're forced to resolve conflicts (two people changing the same thing at the same time) as they arise

Cons:

- There is a lot of overhead, and it's worst at the beginning
- "Resolving conflicts" can be awful
- Everyone on a project is required to stick with the same development pipeline

Vocab

- Repository
- Commit
- Push / Pull
- Branch / Merge
- Cloning? Forking??

Not going to cover

- Messaging
- Issue tracking

Confidentiality

• You have to watch out for data confidentiality – GitHub is public!