Analysis of Software Carpentry's Post-Workshop Surveys

Project Lead: Kari L. Jordan
Published: July 1, 2017

For nearly 20 years, Software Carpentry has developed material and trained instructors to teach computing skills to researchers in science, medicine, engineering, and other disciplines. This report is an analysis of the post-workshop survey responses collected for Software Carpentry's workshops from March 2015 through July 2017. In this two year period, nearly 4,000 responses were collected.

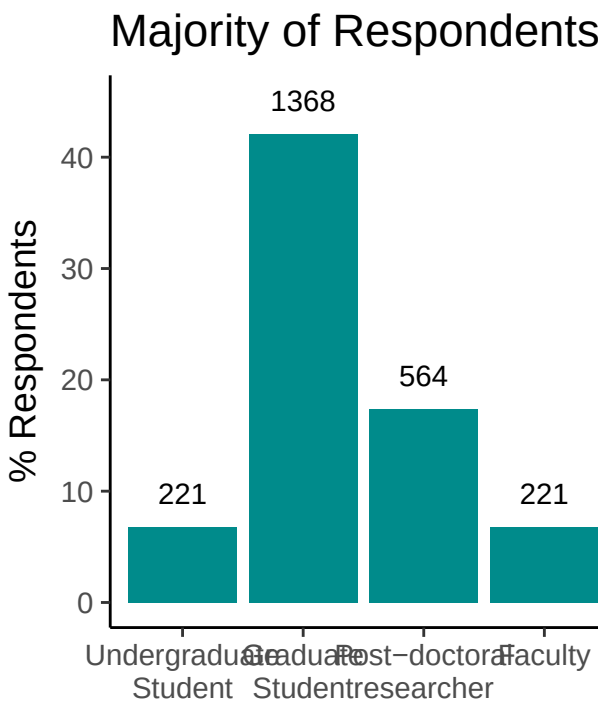
A PDF of the survey questions, the data used in this analysis, and full R code are located on the assessment repo on GitHub. Special thank you to Ben Marwick, Naupaka Zimmerman, Erin Becker, and Jonah Duckles. These individuals made valuable contributions to the code that was used to create the figures in this report.

Community members are invited to contribute code to this analysis. Feel free to use the data and tell us about your findings.

Respondent Demographics

A host of initiatives have been developed and implemented globally to address gender disparities in computing. Software Carpentry's volunteer instructors have hosted hundreds of workshops since 1998, and the post-workshop survey data shows **parity** in attendance of males compared to females.

Gender	n	%
Female	1575	48.5
Male	1597	49.2
Other	10	0.3
Prefer not to say	64	2.0



p

A breakdown of Software Carpentry's learners by **status** is provided below.

42% of Software Carpentry's post-workshop survey respondents are **Graduate Students**.

A breakdown of respondents by research domain/field of work or study is provided below. Respondents were asked to check all that apply. The majority of Software Carpentry learners work in **Life Sciences**.

Research Domain	n	%
Life Sciences (Genetics, genomics, bioinformatics)	927	24.9
Life Science - Organismal/systems (ecology, botany, zoology, microbiology, neuroscience)	894	24.0
Planetary sciences (geology, climatology, oceanography, etc.)	245	6.6
Mathematics/statistics	225	6.0
Physics	217	5.8
Civil, mechanical, chemical, or nuclear engineering	167	4.5
Medicine and/or Pharmacy	161	4.3
Chemistry	149	4.0
Social sciences	149	4.0
Library and information science	121	3.2
Economics/business	98	2.6
Humanities	98	2.6
Psychology	88	2.4
Education	79	2.1
High performance computing	79	2.1
Space sciences	33	0.9

Respondent Perception of Workshop Content and Atmosphere

Software Carpentry has developed an interactive instructional approach that includes direct instruction (i.e. explicit teaching and demonstrations), indirect instruction (i.e. problem solving and discovery), and experiential learning. Respondents have mixed feelings about the pace of the workshop they attended, as outlined below.

Pace	n	%
Just right	1317	38.5
Slightly fast	1057	30.9
Slightly slow	730	21.3
Too fast	157	4.6
Too slow	164	4.8

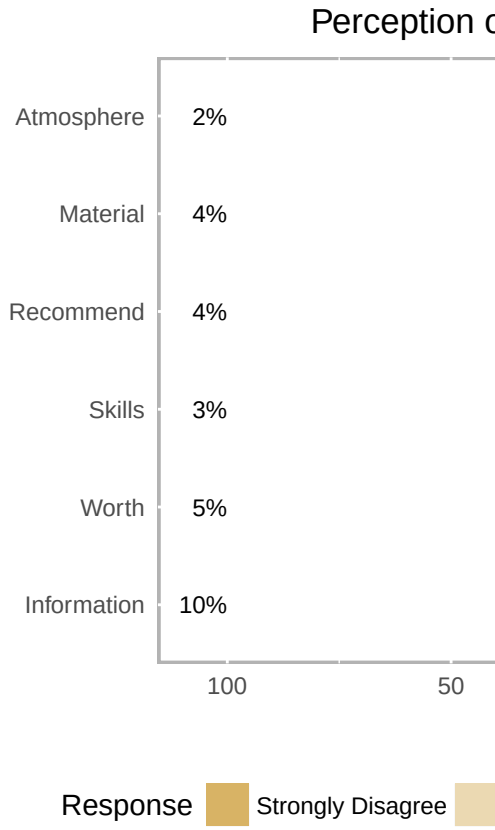
Respondents were asked to indicate their perception of the balance of lecture to hands-on work in the workshop. A breakdown of their responses is provided below.

Balance: Lecture to Hands-On Work	n	%
Too much lecture	59	1.7
Slightly too much lecture	354	10.3
Balanced (lecture/hands-on)	2773	81.0
Slightly too much hands-on	204	6.0
Too much hands-on	35	1.0

81% of respondents felt the workshop they attended was well balanced between lecture and hands-on learning.

Learners were asked to rate their level of agreement on a scale of 1 (Strongly Disagree) to 5 (Strongly Agree) for the following statements regarding the atmosphere and content of the workshop they attended:

- **Atmosphere**: The overall atmosphere of the workshop was welcoming.
- **Material**: The material presented matched the workshop description.
- **Recommend**: I would recommend this workshop to a friend/colleague.
- **Skills**: I learned skills that I will be able to use in my research/work.
- **Worth**: The workshop was worth my time.
- **Information**: The amount of information covered at the workshop was reasonable for allotted time.



The following Likert chart is an analysis of learner responses to the statements above.

The data strongly suggests that Software Carpentry provides a welcoming environment for its learners where the material not only matches the workshop description, but is worth the time learners spend learning it. Learners acquire skills they are able to apply to their research and/or job function in the time allotted over the two-day period. Lastly, learners feel impressed to recommend the workshop to a friend or colleague.

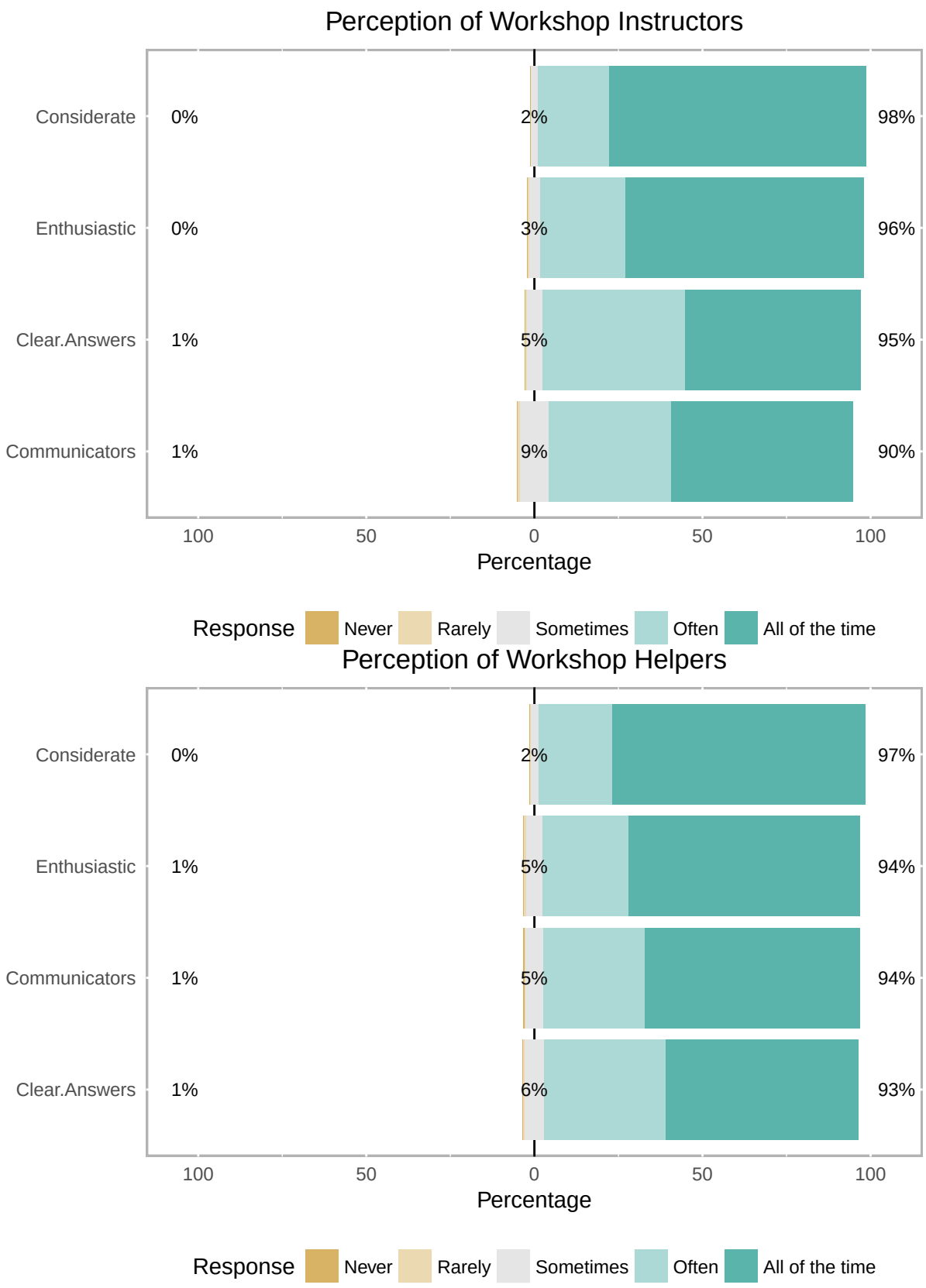
Respondent Perception of Workshop Instructors and Helpers

A strength of Software Carpentry's ecosystem is its instructors and helpers. Learners who responded to Software Carpentry's post-workshop survey were asked to rate how they felt instructors and helpers worked as a team based on the following criteria:

- **Considerate**: Instructors/Helpers were considerate.
- **Enthusiastic**: Instructors/Helpers were enthusiastic.
- **Communicators**: Instructors/Helpers were good communicators.

- **Clear.Answers**: Instructors/Helpers gave clear answers to your questions.

The two Likert plots below provide an analysis of respondent answers.



Software Carpentry's workshop instructors and helpers create an environment where learners are able to receive clear answers to their questions. Additionally, instructors and helpers communicate clearly with

learners and are both enthusiastic about the material being taught and considerate of Software Carpentry learners.

Additionally, 94.1% of respondents felt there were enough helpers in the workshop they attended.

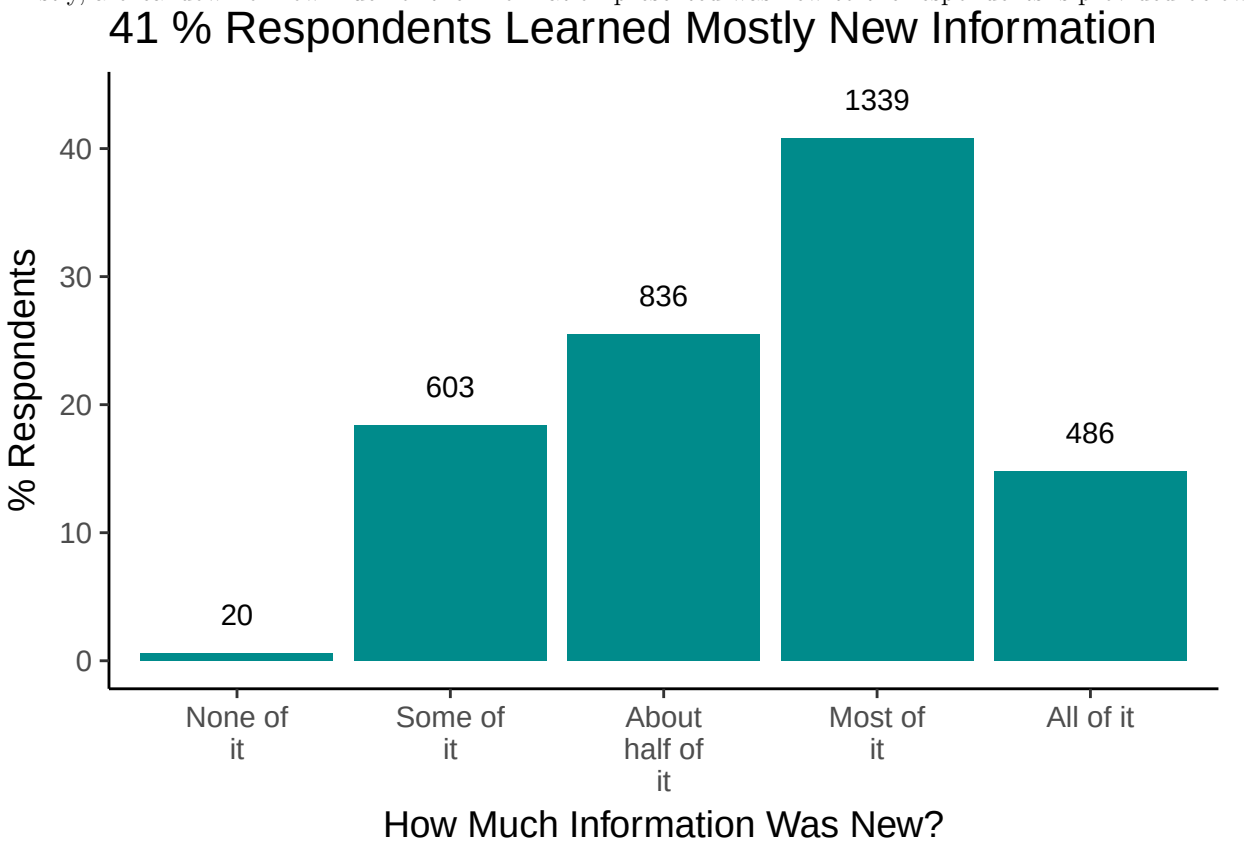
Respondent Self-Reported Knowledge of Tools Covered

Software Carpentry seeks to know more about learners' experience with the tools covered in their workshop before attending, and after attending the workshop. A series of questions were developed around learners' prior knowledge and perception of the tools they learned. Workshops are self-organized, and each workshop includes the following **core topics**:

- Automating tasks using the Unix shell;
- Structured programming in Python, R, or MATLAB; and
- Version control using Git or Mercurial.

Learners were asked to indicate their level of knowledge of the Unix shell, R, Python, Git, and SQL prior to attending a Software Carpentry workshop. Not all workshops cover SQL.

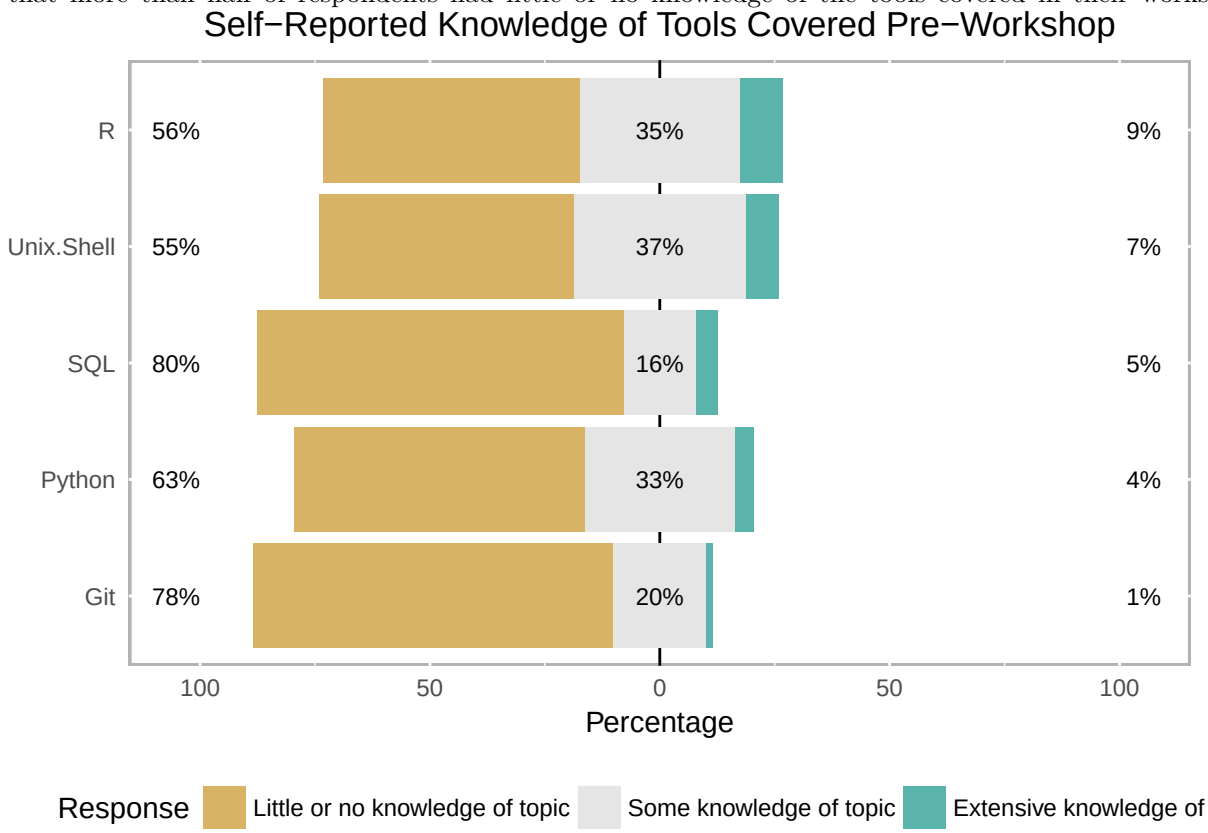
Firstly, a breakdown of how much of the information presented was new to the respondents is provided below.



25% of respondents said about half of the material they learned in a workshop was new to them, and 41% of respondents reported that most of the information they learned in the workshop they attended was new to them, while 15% reported that all of the information they learned was new.

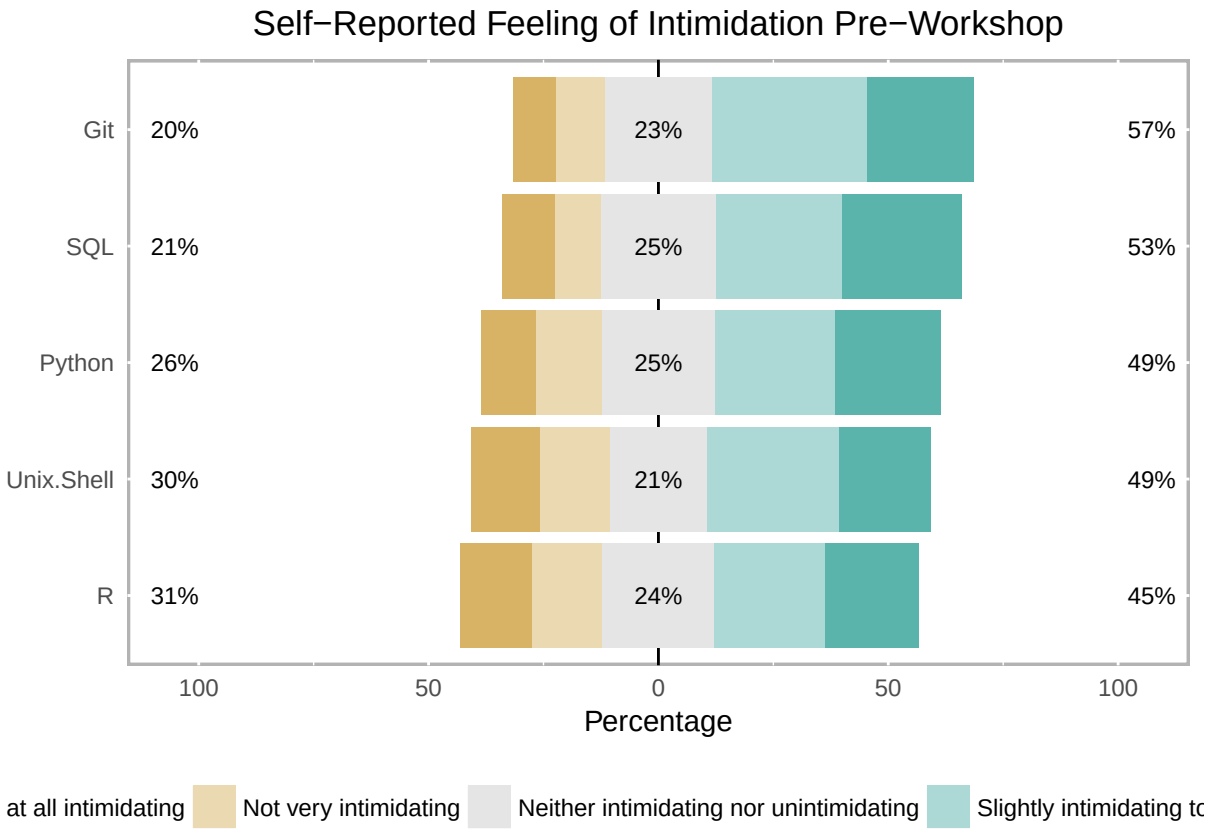
Taking a look at each tool more closely, the Likert plot below provides a breakdown of respondents' self-reported knowledge of the tools covered before attending a workshop. From the figure we see

that more than half of respondents had little or no knowledge of the tools covered in their workshop.



An interesting result is the percentage of learners who had little or no knowledge of Git and SQL (78% and 79% respectively). However, 36%, 39%, and 33% of respondents already had some knowledge of R, the Unix shell, and/or Python, respectively.

Individuals who are new to computer programming tend to be intimidated for lack of familiarity with syntax and terms. As Software Carpentry learners have varying knowledge levels of the tools covered pre-workshop, we are interested in understanding learners self-reported feeling of intimidation with these tools. The Likert plot below provides a breakdown by tool (Git, SQL, etc.).

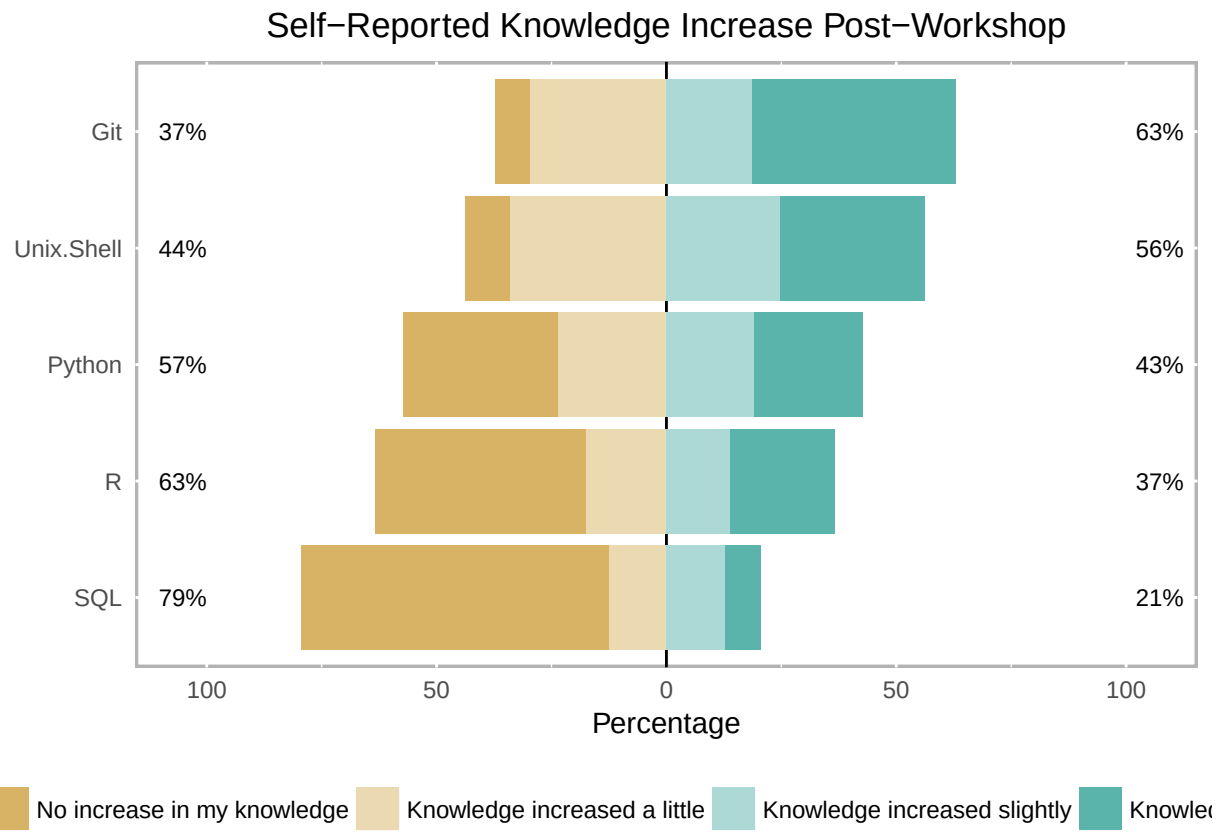


Taking a closer look at the responses by percentage in the table below, we see that at least 44.5% of respondents felt at least one of the tools covered in the workshop they attended was either slightly or very intimidating.

Perception	Git	Python	R	SQL	Shell
Not at all intimidating	9.2	11.9	15.6	11.5	14.9
Not very intimidating	10.5	14.2	15.4	10.0	15.3
Neither intimidating nor unintimidating	23.5	24.9	24.4	25.1	21.3
Slightly intimidating to me	33.7	26.1	24.1	27.5	28.5
Very intimidating to me	23.1	22.9	20.4	26.0	20.0

Self-Reported Knowledge Increase

Software Carpentry is not only interested in creating an atmosphere where learning programming becomes less intimidating, but we want for learners to leave with increased knowledge of the tools that were covered in their workshop. The Likert plot below provides a breakdown of respondents' self-reported knowledge increase.

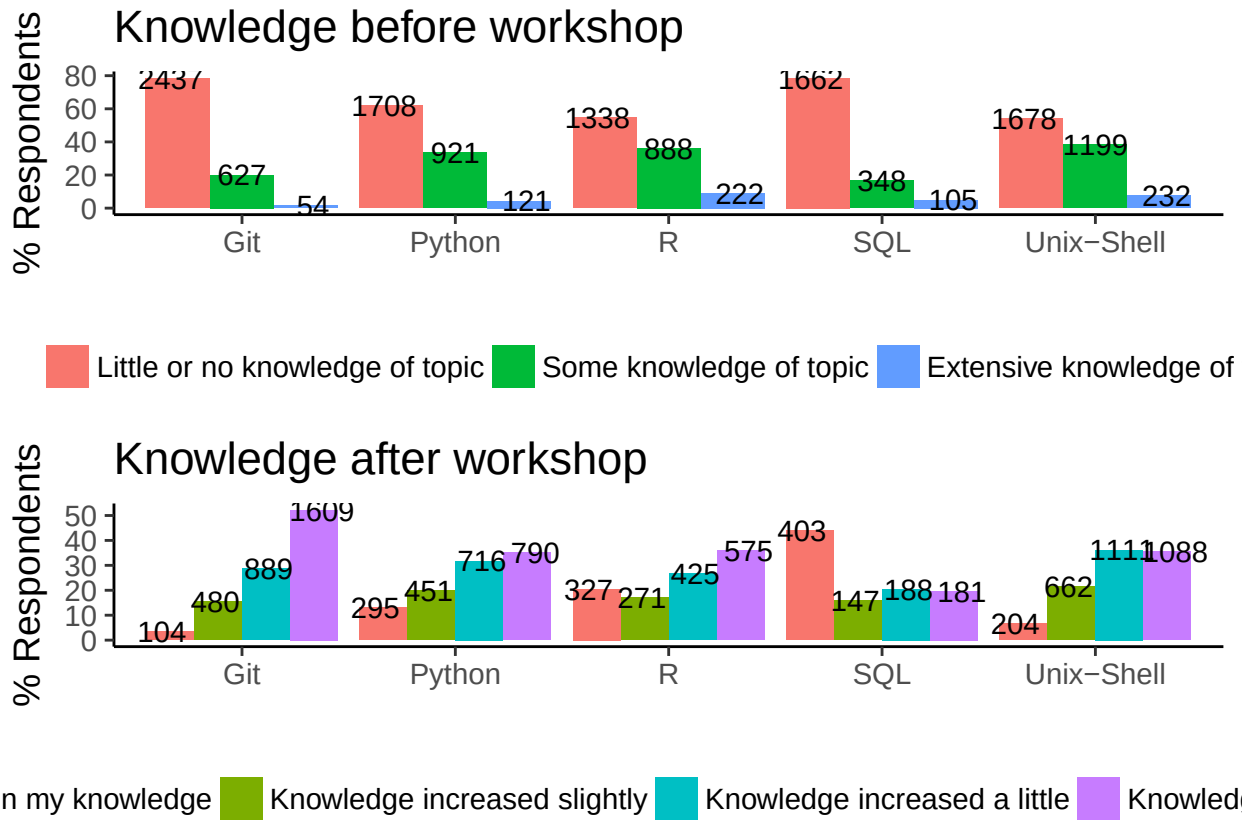


One thing to note is that SQL is not covered in all workshops (60% of respondents said this was not covered). This explains the large portion of respondents that had no increase in their knowledge of SQL.

51% of respondents said their knowledge of Git increased a great deal. This is great news, as 78% of respondents reported having little or no knowledge of Git before attending a Software Carpentry workshop!

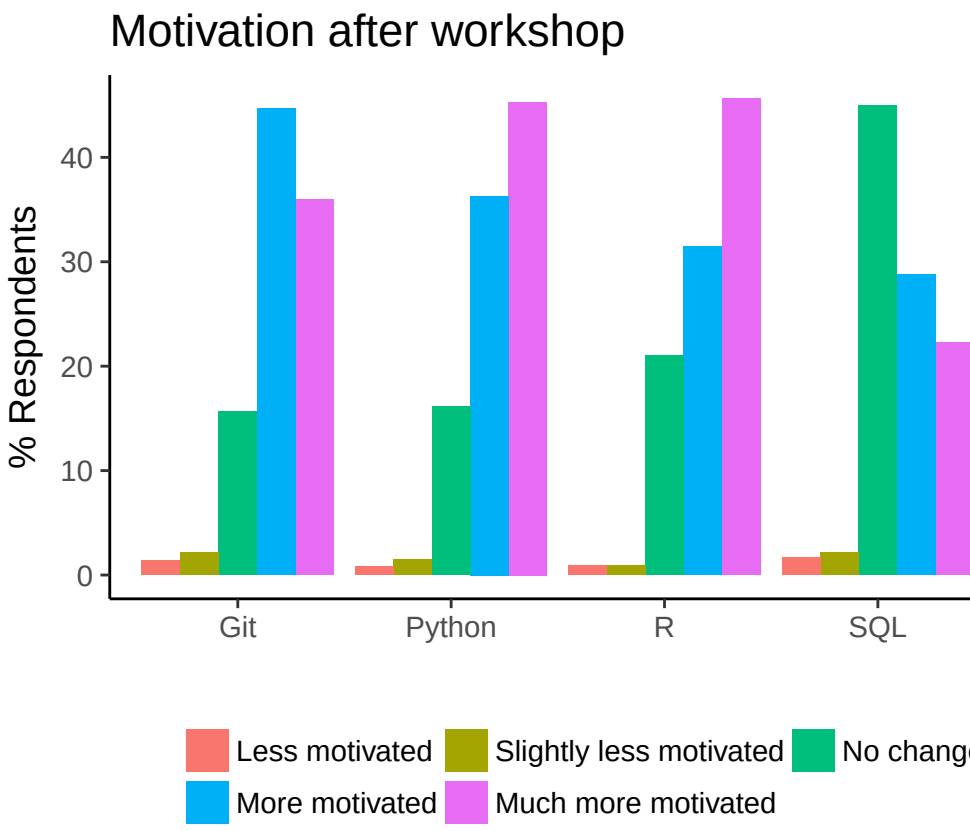
Self-Reported Knowledge Increase: Pre/Post Comparison

Now let's take another look at learners' prior knowledge with the tools covered in the workshop they attended, compared to after the workshop. The grid below provides the breakdown by tool



Motivation to Learn Post-Workshop

Another goal of Software Carpentry is for learners to leave the workshop motivated to continue their learning. From the figure below, we see that learners are more motivated to continue learning and improving upon the



skills that were covered in their workshop.

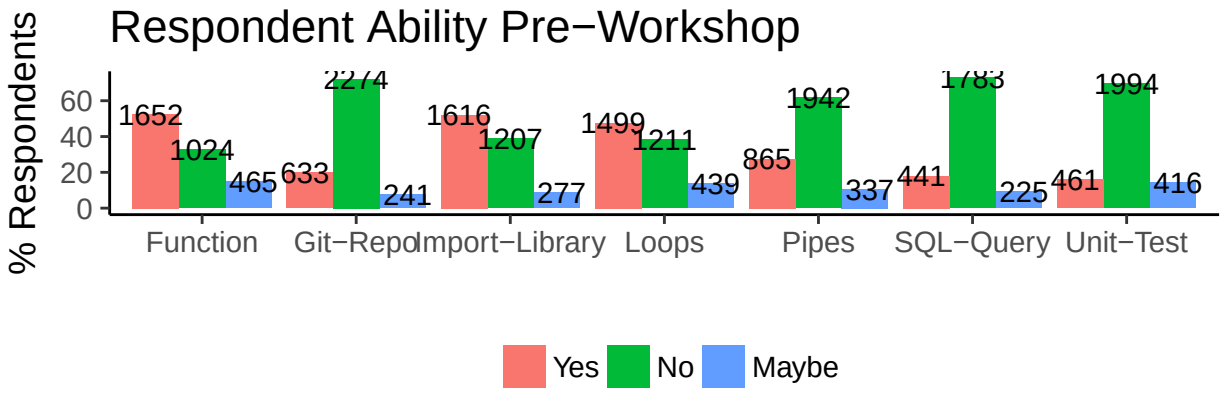
Respondent Ability to Perform Computing Tasks

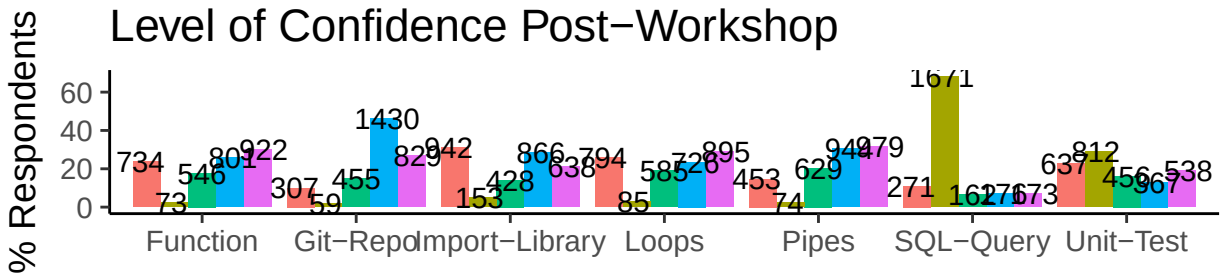
Motivation is important, but being confident in your ability to complete specific computing tasks is an equally important goal of Software Carpentry. The grid below shows respondents' self-reported ability to complete tasks including:

- Using pipes to connect shell commands
- Writing a 'for loop' to automate tasks

- Initializing a repository with git
- Writing a function
- Importing a library or package in R or Python
- Writing a unit test in Python or R
- Writing an SQL query

It also provides their self-reported level of confidence in being able to complete the tasks above after completing





the workshop.

Let's take a closer look with a table. The data shows an increase in confidence for nearly all of the computing tasks outlined above.

Skill	Ability Pre-Workshop	n	%
Pipes	Maybe	337	10.7
Pipes	No	1942	61.8
Pipes	Yes	865	27.5
Loops	Maybe	439	13.9
Loops	No	1211	38.5
Loops	Yes	1499	47.6
Git-Repo	Maybe	241	7.7
Git-Repo	No	2274	72.2
Git-Repo	Yes	633	20.1
Function	Maybe	465	14.8
Function	No	1024	32.6
Function	Yes	1652	52.6
Import-Library	Maybe	277	8.9
Import-Library	No	1207	38.9
Import-Library	Yes	1616	52.1
Unit-Test	Maybe	416	14.5
Unit-Test	No	1994	69.5
Unit-Test	Yes	461	16.1
SQL-Query	Maybe	225	9.2
SQL-Query	No	1783	72.8
SQL-Query	Yes	441	18.0

Skill	Confidence Post-Workshop	n	%
Pipes	Confidence increased a bit	979	31.8
Pipes	Confidence increased greatly	944	30.7
Pipes	Confidence increased slightly	629	20.4
Pipes	N/A - Not covered at workshop	74	2.4
Pipes	No change in confidence	453	14.7
Loops	Confidence increased a bit	895	29.0
Loops	Confidence increased greatly	726	23.5
Loops	Confidence increased slightly	585	19.0
Loops	N/A - Not covered at workshop	85	2.8
Loops	No change in confidence	794	25.7
Git-Repo	Confidence increased a bit	829	26.9
Git-Repo	Confidence increased greatly	1430	46.4
Git-Repo	Confidence increased slightly	455	14.8
Git-Repo	N/A - Not covered at workshop	59	1.9
Git-Repo	No change in confidence	307	10.0
Function	Confidence increased a bit	922	30.0
Function	Confidence increased greatly	801	26.0
Function	Confidence increased slightly	546	17.8
Function	N/A - Not covered at workshop	73	2.4
Function	No change in confidence	734	23.9
Import-Library	Confidence increased a bit	638	21.1
Import-Library	Confidence increased greatly	866	28.6
Import-Library	Confidence increased slightly	428	14.1
Import-Library	N/A - Not covered at workshop	153	5.1
Import-Library	No change in confidence	942	31.1
Unit-Test	Confidence increased a bit	538	19.1
Unit-Test	Confidence increased greatly	367	13.1
Unit-Test	Confidence increased slightly	456	16.2
Unit-Test	N/A - Not covered at workshop	812	28.9
Unit-Test	No change in confidence	637	22.7
SQL-Query	Confidence increased a bit	173	7.1
SQL-Query	Confidence increased greatly	176	7.2
SQL-Query	Confidence increased slightly	162	6.6
SQL-Query	N/A - Not covered at workshop	1671	68.1
SQL-Query	No change in confidence	271	11.0

Summary

Software Carpentry workshops improve learner skill, ability, and confidence in using computing tools like Python, Git, and the Unix shell. Additionally, respondents are satisfied with the caliber of workshop instructors and helpers. To close out this report, I offer a list of interesting questions that could be answered with this data, and encourage community members to get involved by using the data in this analysis to answer these questions:

- Are there differences in gender for intimidation, confidence, and/or motivation to use the tools post-workshop?
- Is there a relationship between respondent confidence and their motivation to use the tools post-workshop?
- Is there a relationship between respondent knowledge increase and their confidence and/or motivation post-workshop?
- Is there a relationship between respondent intimidation pre-workshop and their confidence and/or motivation to use the tools post-workshop?

What other questions can be answered from this data? Additionally, here are a few other questions that we can discuss as a community:

- Why is SQL the least covered workshop topic?
- Has your confidence and/or motivation to use the tools we teach changed as a result of being a member of Software Carpentry's community? How? Why?