Research Education Program Plan

A Significance

> "Does the proposed program address a key audience and an important aspect or important need in training in rigor and reproducibility? Is there convincing evidence in the application that the proposed program will significantly advance the stated goal of the program?"

B Innovation

> "Taking into consideration the nature of the proposed research education program, does the applicant make a strong case for this program effectively reaching an audience in need of the program's offerings? Where appropriate, is the proposed program developing or utilizing innovative approaches and latest best practices to improve the knowledge and/or skills of the intended audience?"

These modules will teach the principals of reproducibility as well as introduce researchers to tools for implementing reproducible research workflows. The implementation portion of these modules will focus on tools from the open-source R programming language. R can be freely, quickly, and easily downloaded and installed to a user's computer, allowing new users to get started quickly, a critical consideration for usable scientific software [1]. R has been maintained for over a decade by the R Development Core Team and works with all major computing platforms, ensuring widespread access, stability, and compatability, also critical for ease-of-use [2, 3]. R offers a well-developed environment for creating new tools that extend the core language [4] and includes ample tools for documenting research workflows [5, 6]. R's status as the *lingua franca* of statisticians and biostatisticians means that its use in early stages of experimental data recording and pre-processing can help foster closer collaborations between laboratory-based scientists and statisticians throughout the research process. R can be scaled as the volume of data in projects grows [1], as it includes tools to interface with distributed computing platforms (e.g., *Hadoop* [7], *Spark* [8]), and its scripts can be integrated within workflow management systems (e.g., *Galaxy* [9, 10]).

C Approach

> "Does the proposed program clearly state its goals and objectives, including the audience to be reached, the content to be conveyed, and the intended outcome? Is there evidence that the program is based on a sound rationale, as well as sound educational concepts and principles? Is the plan for evaluation sound and likely to provide information on the effectiveness of the program?"

C.1 Proposed Research Education Program Plan

> "While the proposed research education program may complement ongoing research training and education occurring at the applicant institution, the proposed educational experiences must be distinct from those research training and research education programs currently receiving federal support. When research training programs are on-going in the same department, the applicant organization should clearly distinguish between the activities in the proposed research education program and the research training supported by the training program. The research education proposed must be **targeted to trainees and investigators at any level**. State the **goals for education** and **justify the area of training** selected for module development in terms of its **relevance and potential impact** on improving the development of skills and knowledge important for conducting rigorous and reproducible research. Describe the **subject material** to be covered. Describe the **format** for the training module proposed and **justify it in terms of the education goals**. The **length** of the proposed training module should be explained in terms of **scope and depth of coverage** of the subject matter. In addition, **how the research education will be utilized by trainees or investigators** should be described—for example, a module on how to avoid confirmation bias to be taken by all beginning laboratory workers, or a module on appropriate design of

> animal studies to be taken immediately prior to beginning such work. Describe the **plans for piloting and evaluating the effectiveness** of the training module. Describe **plans for making the proposed training module section 508 compliant of the Rehabilitation Act** (29 U.S.C. '794 d), as amended by the Workforce Investment Act of 1998 (P.L. 105 220; see http://www.section508.gov/ for additional information). Provide a **timeline for module development, piloting and refinement, dissemination, evaluation, and maintenance**. This timeline must propose **making the training publicly available within two years** of the award date."

C.1.1 Educational goals of the modules

The importance of computational reproducibility of scientific research is increasingly recognized by scientists, journals, and funding agencies, with such "computationally reproducible" research requiring that all data and code for a research project be available and that this data and code can be used to regenerate study findings either by the original researcher or by other researchers [11, 12].

Every extra step of data formatting is another chance to introduce an error in the data. Therefore, by keeping research data pipelines simple—which can be more easily achieved if data is initially recorded in a format amenable to later data pre-processing, analysis, and visualization—researchers can decrease the potential for errors in the data and therefore improve the rigor and reproducibility of their research.

Improving the Reproducibility of Experimental Data Recording One key concept for improving the reproducibility of experimental data collection is understanding how to create and use the "tidy" data format, which enables later data analysis using R's *tidyverse* framework. The *tidyverse* framework enables powerful and user-friendly data management, processing, and analysis by combining simple tools to solve complex, multi-step problems, and this framework is enabled by ensuring those simple tools share a common interface: a "tidy" data format [13, 14, 15, 16]. Working within the R framework facilitates research that adheres to standards of reproducibility through scriptable data analysis that can easily be placed under version control [17]. Since the tools are simple and share a common interface, they are easier to learn, use, and combine than tools created in the classical R framework [13, 18, 19, 20]. This *tidyverse* framework is quickly becoming the standard taught in introductory R courses and books [21, 22, 23, 24, 19, 20] (see also Letters of Support [LOS], Kimmel, Peng), ensuring ample training resources for researchers new to programming, including books (e.g., [25, 26], some freely available online, e.g., [16]), massive open online courses (MOOCs), onsite university courses [22, 23, 24], and Software Carpentry workshops [27, 28]. Further, tools that extend the tidyverse have been created to enable high-quality data analysis and visualization in several domains, including text mining [29], microbiome studies [30], natural language processing [31], network analysis [32], ecology [33], and genomics [34].

RStudio allows users to create their own custom "Project" template, suited to a specific type of data analysis or software development, which can then be registered and accessed by other users [35]. While a "Project" can have any internal structure, a common structure can be enforced for a certain type of project through the creation of a new "Project" template, which defines the required subdirectories, structure, and file names of common elements that must exist in the project [35]. This template, when selected by a future user, will create a new directory with a "skeleton" structure, potentially including templated files (e.g., for metadata). Projects saved in this format can be easily put under *Git* version control in *RStudio*, which includes a pane that allows users to work under version control without learning command-line version control language and, if desired, easily connect the project with an online version of the project hosted on *GitHub*. This "project" framework has recently been encouraged by a number of researchers as a way to enable computationally reproducible research, especially for research conducted by teams [36, 37, 18], and the use of *Git* and *GitHub* has also been encouraged as a tool to enable reproducible research [38, 12, 17, 18, 39].

Improving the Reproducibility of Experimental Data Pre-Processing. Scriptable software tools bring key advantages compared to GUI software in terms of data pre-processing [39, 40, 41, 38], but it is critical to provide some training on the use of these tools for researchers new to programming. Expertise with a scripting language is not universal across the biomedical community, although literacy in programming is increasing in the sciences [12], and many now recommend programming as a critical skill for all biology Ph.D. students [1].

Contrast the lack of guidance on experimental data recording for academic research with the guidelines from industry, including "Good laboratory practice").

C.1.2 Module subject material

We propose to develop two collections of modules, **Improving the Reproducibility of Experimental Data Recording** and **Improving the Reproducibility of Experimental Data Pre-Processing**.

Table 1: Modules for '**Improving the Reproducibility of Experimental Data Recording'**. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black).

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Separating data recording and analysis	Many biomedical laboratories currently use spreadsheets, with embedded macros, to both record and analyze experimental data. This practice empedes the transparency and reproducibility of both data recording and data analysis. In this module, we will describe this common practice and explain how it impedes the transparency and reproducibility of data recording and analysis. We will then outline alternative approaches that separate the steps of data recording and data analysis and explain how these alternative approaches can improve the reproducbility of biomedical research.	• Explain the difference between data recording and data analysis; • Understand why collecting data on spreadsheets with embedded macros impedes transparency and reproducibility; • List alternative approaches that separate data recording and data analysis to improve transparency and reproducibility.	15	• Discussion questions, including describing data recording approaches the trainee has previously used in research projects and the benefits and limitations of those approaches in terms of data transparency and reproducibility; • Short audio recording of two Co-Is giving their own answers to these discussion questions.
Principals and power of structured data formats	In this module, we will explain what makes a dataset 'structured' and why this format is a powerful tool for reproducible research.	• List the characteristics of a structured data format; • Describe how using a structured data format when recording experimental data can improve the transparency and reproducibility of research; • Outline other benefits of using a structured format when recording data.	25	• Applied exercise: For example datasets, specify whether each is in a structured data format and, in cases where it is not, draft a structured format that could be used to record the data; • Video walking trainees through one solution to the applied exercise.

Table 1: Modules for '**Improving the Reproducibility of Experimental Data Recording**'. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
The 'tidy' data format: an implementation of a structured data format	The 'tidy' data format was outlined in a 200[x] paper and has since quickly gained popularity among statisticians and data scientists. By consistently using this data format, researchers have found they can employ combinations of simple, generalizable tools to perform complex tasks in data processing, analysis, and visualization. However, despite the power of this format, it is not yet widely known or used among laboratory scientists when they record experimental data. In this module, we will explain what characteristics determine if a dataset is 'tidy' and how use of the 'tidy' implementation of a structure data format can improve the ease and efficiency of 'Team Science', including collaborations with statisticians.	• List which characteristics to check to determine if a dataset complies with the 'tidy' structured data format; • Explain the difference between the ideas of a structured data format (general concept) and the 'tidy' data format (one implementation of that general format that is now particularly popular in data analysis).	25	• Quiz questions: For example datasets, correctly identify which of the 'tidy' data principals the dataset has or lacks; • Video giving answers and explanations for quiz questions, including showing 'tidy' versions of each example dataset; • Link to paper that established the 'tidy' data format.
Designing templates for tidy data collection	This module will move from the principals of the 'tidy' data format to the practical details of designing a 'tidy' data format to use for a specific research project. We will describe common issues that prevent real datasets from experimental research projects from following a 'tidy' format and show how they can be avoided when deciding the format in which to record experimental data. We will also provide rubrics and a checklist to help determine if a data collection template complies with a 'tidy' format.	• Identify characteristics that keep a dataset from following a 'tidy' format; • Convert data from an 'untidy' to a 'tidy' format.	20	• Applied exercise: Take a dataset in an 'untidy' format, identify what characteristics keep it from being 'tidy', and convert design a 'tidy' form of the data; • Video providing a detailed solution to the applied exercise.

Table 1: Modules for **'Improving the Reproducibility of Experimental Data Recording'**. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Example: Creating a template for data collection	In this module, we will walk through an example of creating a template to collect data in a 'tidy' format for a laboratory-based research project. As an example, we will use a research project headed by one of our Co-Is on tuberculosis [more description of this project]. We will walk through the 'untidy' format initially used to collect data for this project, explain how this format differed from a 'tidy' format, and show how we changed the format to be 'tidy'. Finally, we will show examples of how the experimental data can easily be cleaned, analyzed, and visualized using reproducible tools once it is in a 'tidy' format.	• Understand how the principals of 'tidy' data can be applied to experimental data from a real research project; • Explain the advantages of using a 'tidy' data format for the example project.	15	• Discussion questions, including listing examples of how experimental datasets the trainee has previously worked with or is currently working with are 'untidy' and how they could be converted to a 'tidy' format; • Short audio recording of two Co-Is giving their own answers to these discussion questions.
Power of using a single 'Project' directory for storing and tracking research project files	To improve the computational reproducibility of a research project, researchers can use a single 'Project' directory to collectively store all research data (raw and pre-processed), meta-data, code for data pre-processing, and research products further along the research pipeline (e.g., paper drafts, figures, code for data analysis). In this module, we will show how all research project files can be collected and saved in a single 'Project' directory. We will explain how using this practice from the start of a research project improves the reproducibility of the projects, as well as how this practice facilitates the use of later tools to improve reproducibility, including version control. Finally, we will list some of the common components and subdirectories that are useful to include in the structure of a 'Project' directory, including subdirectories for raw and pre-processed experimental data.	• Describe a 'Project' directory, including common components and subdirectories; • List how collecting all research data and other files related to a research project in a single 'Project' directory improves the reproducibility of a research project; • Describe how experimental data collection can be integrated with a research 'Project' directory.	20	abitem Quiz questions: These will test the trainee's understanding of what a 'Project' directory is, what common components it may include, and the benefits of structuring research project files—including raw and pre-processed experimental data—within a single 'Project' directory from the beginning of the research project; • Video with detailed answers and discussion of quiz questions.

Table 1: Modules for '**Improving the Reproducibility of Experimental Data Recording'**. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Creating 'Project' templates	Researchers can gain even more benefits, in terms of improving both the reproducibility and efficiency of research, by using a consistent structure for the 'Project' directories for all of the research projects for a research group. We will describe the benefits of using a consistent structure for 'Project' directories across different research projects within a research group, including how this practice can facilitate the re-use of code for pre-processing, analyzing, and visualizing data. Further, we will demonstrate how RStudio can be used to create a template of a research group's 'Project' directory structure, so a new project can be initialized with a skeleton directory that follows the 'Project' directory format established by the research group.	• Explain how using a consistent structure for research 'Project' directories can improve the reproducibility and efficient of research projects within a research group; • Understand how RStudio can be used to create a template to use to create consistently-structured research 'Project' directories.	25	• Discussion questions, including descriptions of how the trainee has saved and tracked research project files for previous research projects and what barriers, if any, these practices introduced in terms of the reproducibility and efficiency of research; • Short audio recording of two Co-Is discussing how they have saved and tracked research project files in previous projects and what barriers to reproducibility these practices introduced.
Example: Creating a 'Project' template	In this module, we will walk through a real example of establishing the format for a research group's 'Project' template, creating that template using RStudio, and initializing a new research project directory using the created template. [Further description of the real research project]	• Create a 'Project' template in RStudio to use to initialize consistently-formatted 'Project' directories to store all files related to a research project; • Initialize a new 'Project' directory using this template.	15	• Applied exercise: We will provide a description of the components and subdirectories that a research group has decided to include in their 'Project' template. The trainee will need to use RStudio to create and save a 'Project' template that meets these specifications; • Video demonstrating a detailed solution to the applied exercise.

Table 1: Modules for '**Improving the Reproducibility of Experimental Data Recording**'. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Harnessing version control for transparent data collection	As a research project progresses, a typical practice in many experimental research groups is to save new versions of files (e.g., 'draft1.doc', 'draft2.doc'), so that any changes can be reverted to earlier versions. However, this practice leads to an explosion of research project files, and it becomes hard to track which files represent the 'current' state of a project. Version control allows researchers to edit and change research project files in a way that allows them to identify and undo any previous changes while maintaining a single version of each file. Further, version control requires short messages describing each change made to each file, which improves the transparency and reproducibility of both the recording of experimental data and also the later steps of pre-processing, analyzing, and visualizing the data. In this module, we will explain what version control is and how it can be used in research projects. We will highlight how version control can improve the transparency and reproducibility of research. Finally, we will give examples of version control tools that are popular for research.	• Describe version control and what it does; • List how using version control improves the transparency and reproducibility of research.	10	• Discussion questions, including discussion of how the trainee has managed evolving research project files in previous projects and any barriers those practices introduced in conducting efficient and reproducible research; • Short audio recording of two Co-Is giving their own answers to these discussion questions.

Table 1: Modules for '**Improving the Reproducibility of Experimental Data Recording**'. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Using git and GitLab to enhance the reproducibility of collaborative research	Once a researcher has learned to use git on their own computer for local version control, they can begin using version control platforms like GitLab and GitHub to collaborate with others in their research group while keeping the project under version control. These platforms allow the all collaborators to share a current version of a project directory (similar to Dropbox), but in a way that allows easy use of version control and that is more efficient for exploring (and, when necessary, undoing) the changes each team member has made to project files. In this module, we will describe why a research team may want to use a version control platform like GitLab to work collaboratively on a project. Further, we will show how to use git through RStudio's user-friendly interface and how to connect from a local computer to GitLab through RStudio.	• Explain the benefits of using a version control platform like GitLab, rather than Dropbox, to share project files for collaborative research projects, particularly in terms of increasing the transparency and reproducibility of a research project; • Describe the difference between git and GitLab; • Understand how to set up and use git through RStudio's interface; • Understand how to connect with GitLab through RStudio and how to use these version control and collaboration tools to improve the reproducibility of research projects.	20	• Applied exercise, with detailed instructions for each step: Use RStudio to initialize version control for a directory and to make several tracked changes. Create a matching GitLab repository and use RStudio to connect your local and GitLab versions of the directory. • Video walking trainees through a detailed solution to the exercise.

Table 2: Modules for **'Improving the Reproducibility of Experimental Data Pre-Processing'**. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black).

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Principals and benefits of scripted pre-processing of experimental data	The experimental data collected for biomedical research often requires pre-processing before it can be analyzed (e.g., gating of flow cytometry data, peak finding and quantification for LC / MS metabolomics data). While often proprietary, point-and-click software is available for this pre-processing, use of such software can limit the transparency and reproducibility of this pre-processing stage of the analysis, and point-and-click software is often time-consuming to use for repeated tasks over large research projects. In this module, we will explain how using scripts to apply open source software for this pre-processing step can improve the transparency, reproducibility, and transparency of research.	• Define pre-processing of experimental data and give some examples; • Describe how the use of proprietary software for pre-processing experimental data limits transparency and reproducibility; • Understand what an open source code script is and how it can be used as an alternative in pre-processing experimental data; • List some popular packages in R that can be used to pre-process biomedical experimental data.	15	• Discussion questions, including discussion of which steps are commonly used to pre-process experimental data in the trainee's research area; • Short audio recording of two Co-Is giving their own answers to these discussion questions; • List of some popular R packages for pre-processing different types of biomedical experimental data.
Introduction to R code scripts	In this module, we will explain the difference between interactive software use and the use of code scripts, using examples from R. We will then demonstrate how to create, save, and run an R code script for a simple data cleaning task.	• Describe what an R code script is and how it differs from interactive coding in R; • Create and save an R script to perform a simple data pre-processing task; • Run an R script.	10	• Applied exercise: Given a simple example dataset and a data cleaning task, write and run an R script to perform the task. Then adapt that script to re-use it on a second, similar example dataset. Hints on useful R functions will be provided to help trainees new to the R language; • Video providing a detailed walk-through of a solution to the applied exercise.

Table 2: Modules for **'Improving the Reproducibility of Experimental Data Pre-Processing'**. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Simplify scripted pre-processing through R's 'tidyverse' tools	The R programming language now includes a collection of 'tidyverse' extension packages that enable user-friendly yet powerful work with experimental data, including pre-processing and exploratory visualizations. The principal behind the 'tidyverse' is that a collection of simple, general tools can be joined together to solve complex problems, as long as a consistent format is used for the input and output of each tool (the 'tidy' data format taught in other modules). In this module, we will explain why this 'tidyverse' system is so powerful and how it can be leveraged within biomedical research, especially for reproducibly pre-processing experimental data.	• Define R's 'tidyverse' system; • Explain how the 'tidyverse' collection of packages can be both user-friendly and powerful in solving many complex challenges in working with data; • Describe the difference between 'base R' and R's 'tidyverse'.	15	• Quiz: Questions will test the trainee's understanding of what R's 'tidyverse' is and why it is a powerful yet user-friendly tool for improving the reproducibility, transparency, and efficiency of research projects. • Video with detailed answers and explanations for the quiz questions; • Links to further free sources for developing more 'tidyverse' coding skills.
Complex data types in experimental data pre-processing	Raw data from many biomedical experiments, especially those that use high-throughput techniques, can be very large and complex. Because of the scale and complexity of these data, software for pre-processing the data in R often uses complex, 'untidy' data formats. These complex data formats are necessary for computational efficiency and to aid the structure of the pre-processing software, but the 'untidy' formats add a critical barrier for researchers who wish to explore and visualize the data. In this module, we will describe the complex data formats are often used in open source software for pre-processing experimental data, explain why use of these complex formats is often necessary, and outline how these complex formats create hurdles in implementing reproducibility tools among laboratory-based scientists.	• Explain why R software for pre-processing biomedical data often stores the data in complex, 'untidy' formats; • Describe how these complex data formats can create barriers to laboratory-based researchers seeking to use reproducibility tools for data pre-processing.	15	• Quiz: Determine trainee's understanding of why complex data formats are often used within steps of experimental data pre-processing in open-source software; • Video providing detailed answers to quiz questions.

Table 2: Modules for **’Improving the Reproducibility of Experimental Data Pre-Processing’**. The color of each module’s title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Complex data types in R and Bioconductor	Many R extension packages for pre-processing experimental data use complex (rather than ’tidy’) data formats within their code, and many output data in complex formats. This is necessary for computational efficiency of the pre-processing, but creates a hurdle for using many common tools taught to improve research reproduciblity, including R’s ’tidyverse’ tools. With the rising popularity of the ’tidyverse’ collection of R tools, which require data to be in a ’tidy’ format, R users have recognized that the use of complex, ’untidy’ data formats can complicate reproducible code for data pre-processing, analysis, and visualization. Very recently, some researchers have developed tools (the broom and biobroom R package extensions) that can extract a ’tidy’ dataset from data stored in a complex, list-based format. These tools create a clean, simple connection between the complex data formats often used in pre-processing or modeling experimental data and the ’tidy’ format required to use the ’tidyverse’ tools now taught in many introductory R courses. In this module, we will describe the ’list’ data structure, the common backbone for complex data structures in R, and well as provide tips on how to explore and extract data stored in R in this format. We will then demonstrate how the new *broom* and *biobroom* packages can be used to extract to use to convert output from pre-processing software to ’tidy’ data formats for futher steps of reproducible data visualization and analysis. ’tidy’ versions of pre-processed experimental data from their complex data formats, to allow user-friendly data analysis and visualization using the widely-taught general ’tidyverse’ tools.	• Describe the structure of R’s ’list’ data format; • Take basic steps to explore and extract data stored in R’s complex, list-based structures; • Describe what the *broom* and *biobroom* R packages can do; • Explain why converting data from a complex format to a ’tidy’ format can improve the transparency and reproducibility of a research project.	15	• Applied exercise: We will provide example data in a complex, list-based format. The trainee will explore this data based on step-by-step instructions and will extract specified elements from the data format as well as practice using *broom* and *biobroom* R packages to extract ’tidy’ data from complex data formats.; • Video providing a detailed walk-through of completing this exercise, with explanations for specific steps.

Table 2: Modules for **'Improving the Reproducibility of Experimental Data Pre-Processing'**. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Example: Converting from complex data types to 'tidy' data formats	We will provide a detailed example of a case where data pre-processing in R has resulted in data in a complex, 'untidy' format, and where tools can be used to extract data in a 'tidy' format, which then can easily integrate with general R 'tidyverse' tools for data analysis and visualization. We will walk through an example of applying automated gating to flow cytometry data. We will demonstrate the complex initial format of this pre-processed data and then show trainees how a 'tidy' dataset can be extracted and used for further data analysis and visualization. This example will use real experimental data from research on the immunology of tuberculosis [more details on this project].	• List R package extenstions that can be used to extract 'tidy' data from complex, 'untidy' R data formats; • Describe how these tools can be used in research projects to shift from data pre-processing to analysis and visualization of the processed data.	20	• Applied exercise: Trainees will be given an example dataset in a complex, 'untidy' data format in R and will be instructed in how to convert it to a 'tidy' format and then create some straightforward plots of the data based on this 'tidy' dataset; • Video demonstrating a detailed solution to the applied exercise.
Introduction to reproducible data pre-processing protocols	Reproducibility tools can be used to create reproducible data pre-precessing protocols—documents that combine code and text in a 'knitted' document ... . In this module, we will describe how reproducible data pre-processing protocols can be leveraged early in a research project to improve the reproducibility of the pre-processing of experimental data and to ensure transparency, consistency, and reproducibility across the research projects conducted by a research team.	• Describe a reproducible data pre-processing protocol; • Explain how reproducible data pre-processing protocol can be used to improve the reproducibility of research projects at the data pre-processing phase; • List other benefits of using reproducible data pre-processing protocols, including improving efficiency and consistency of data pre-processing across a research groups research projects.	15	• Discussion questions: Including discussion of how reproducible data pre-processing protocols can make biomedical research more reproducible at the data pre-processing stage; • Short audio recording of two Co-Is giving their own answers to these discussion questions.

Table 2: Modules for **’Improving the Reproducibility of Experimental Data Pre-Processing’**. The color of each module’s title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Introduction to RMarkdown as a tool for creating reproducible data pre-processing protocols	RMarkdown can be used to create documents that combine code and text in a ’knitted’ document, and it has become a popular tool among statisticians and data scientists for improving the computational reproducibility and efficiency of their research. This tool can also be used earlier in the research process, however, to develop well-documented code to pre-process raw experimental data. In this module, we will show trainees the types of documents that can be created and run using RMarkdown. We will describe how RMarkdown is used among statisticians to improve the reproducibility, efficiency, and transparency of data analysis, as well as describe how it can be leveraged earlier in a research project to improve the reproducibility of the pre-processing of experimental data. We will also provide detailed instructions on how to use RMarkdown in RStudio to create documents that combine code and text. We will explain how these documents can be converted into different final file formats (PDF, HTML, Microsoft Word). We will show how an RMarkdown document describing a data pre-processing protocol can be used to efficiently apply the same data pre-processing steps to different sets of raw data.	• Define RMarkdown; • Describe the documents that can be created using RMarkdown; • Explain how RMarkdown can be used to improve the reproducibility of research projects at the data pre-processing phase; • Create a document in RStudio using RMarkdown; • Render the document in multiple file formats; • Apply the document to several different datasets that follow the same format.	15	abitem Applied exercise: Trainees will be asked to create, save, and render their own RMarkdown document through RStudio; • Video providing a detailed walk-through of a solution to the applied exercise.

Table 2: Modules for **'Improving the Reproducibility of Experimental Data Pre-Processing'**. The color of each module's title indicates whether the module focuses on **Principals** (blue), **Implementation** (red), or **Case study examples** (black). *(continued)*

Module title	Description of module content	Objectives (After taking the module, the trainee can ...)	Video Length	Extra educational materials
Example: Creating a reproducible data pre-processing protocol	We will provide an example of creatin a reproducible protocol for the automated gating of flow cytometry data for a project on the immunology of tuberculosis [more details on project]. This data pre-processing protocol was created using RMarkdown and allows the efficient, transparent, and reproducible gating of flow cytometry data for all experiments in a research project. We will walk the trainees through the final pre-processing protocol, how we apply this protocol to new experimental data, and how we developed the protocol initially.	• Explain how a reproducible data pre-processing protocol can be integrated into a real research project; • Describe what is included in a data pre-processing protocol; • Understand how to design and implement a data pre-processing protocol to replace manual or point-and-click data pre-processing tools.	20	• Quiz questions: These will test the trainees understanding of how and why we created well-documented and reproducible data pre-processing protocols for this project, as well as how this helps increase the transparency and reproducibility of the research project; • Short audio recording of discussion with the head of this example research project on how this reproducible data pre-processing fits into her research project and how use of this protocol differs from previous data pre-processing practices in the group.

C.1.3 Format for the training modules

- Online book created through the "bookdown" format, with each module as a book chapter. We can use Git Pages to host this (CSU options for web hosting?).
- Training videos embedded for each module, each 5–30 minutes. Videos will be similar to online course lectures and will be hosted using YouTube. Embedding in the book will allow users to watch videos without leaving the book's webpage.
- Each chapter will end with exercise questions (around 10 questions, combination of discussion questions and applied exercises), as well as an embedded video with discussion of the discussion questions and a detailed walk-through of answers to applied exercises.
- Possibly host this through an online course platform like DataCamp?

Online book. To ensure that these training modules are easy for researchers to access, use, and reference, we will provide all training materials through an online book created with the *bookdown* framework [6] (see LOS, Xie). Through this new framework, we will be able to create a searchable online book that weaves R code into the text and that can include embedded tutorial videos, active weblinks to online references, and computationally reproducible practice examples and exercises. Further, by including R code examples as executable code, we will be able to use this online book to frequently check tutorial code examples to quickly identify and fix any broken tutorial code [6]. Dr. Anderson (PI) has previously created two *bookdown*-based books, *R Programming for Research* and *Mastering Software Development in R*.

C.1.4 Piloting and evaluating effectiveness of training modules

C.1.5 Insuring compliance with Rehabilitation Act

C.2 Team

C.2.1 Program Director/Principal Investigator

> "Is the PD/PI capable of providing both administrative and scientific leadership to the development and implementation of the proposed program? Is there evidence that an appropriate level of effort will be devoted by the program leadership to ensure the program's intended goal is accomplished? If the project is collaborative or multi-PD/PI, do the investigators have complementary and integrated expertise; are their leadership approach, governance and organizational structure appropriate for the project?"
>
> "Describe **arrangements for administration** of the program. Provide evidence that the Program Director/Principal Investigator is actively engaged in research and/or teaching in an area related to the mission of NIH, and can **organize, administer, monitor, and evaluate the research education program**. For programs proposing multiple PDs/PIs, describe the complementary and integrated expertise of the PDs/PIs; their leadership approach, and governance appropriate for the planned project."

C.2.2 Other members of the team

C.3 Institutional Environment and Commitment

> "Describe the institutional environment, reiterating the **availability of facilities and educational resources** (described separately under Facilities & Other Resources), that can contribute to the planned Research Education Program. Evidence of institutional commitment to the research educational program is required. A **letter of institutional commitment** must be attached as part of Letters of Support (see below). Appropriate institutional commitment should include the provision of adequate staff, facilities, and educational resources that can contribute to the planned research education program."

Table 3: Some examples of tracks of training modules for sample trainees

	Long description of the student	Module 2	Another very long description of the research associate
Very long sequence name for sequence 1			
• Separating data recording and analysis	Yes	No	No
• Principals and power of structured data formats	Yes	Yes	No
• The 'tidy' data format: an implementation of a structured data format	Yes	No	Yes
• Designing templates for tidy data collection	Yes	No	No
• Example: Creating a template for data collection	Yes	No	No
• Power of using a single 'Project' directory for storing and tracking research project files	Yes	No	Yes
• Creating 'Project' templates	Yes	No	No
• Example: Creating a 'Project' template	Yes	Yes	No
• Harnessing version control for transparent data recording	Yes	No	Yes
• Using git and GitLab to enhance the reproducibility of collaborative research	Yes	No	No
Sequence 2			
• Principals and benefits of scripted pre-processing of experimental data	Yes	No	No
• Introduction to R code scripts	Yes	Yes	No
• Simplify scripted pre-processing through R's 'tidyverse' tools	Yes	No	Yes
• Complex data types in experimental data pre-processing	Yes	No	No
• Complex data types in R and Bioconductor	Yes	No	No
• Example: Converting from complex data types to 'tidy' data formats	Yes	No	No
• Introduction to reproducible data pre-processing protocols	Yes	Yes	No
• Introduction to RMarkdown as a tool for creating reproducible data pre-processing protocols	Yes	No	Yes
• Example: Creating a reproducible data pre-processing protocol	Yes	No	No

C.4 Evaluation Plan

"Applications must include a plan for evaluating the activities supported by the award in terms of their **frequency of use** and their **usefulness**. The use of **multiple evaluation approaches** is highly encouraged as is **testing several groups with different characteristics**. The application must specify **baseline metrics (e.g., numbers, educational levels, and demographic characteristics of test group)** in a structured format, as well as **measures to gauge the short and long-term success of the research education award in achieving its objectives**. Applicants are expected to **obtain feedback from test group** to help identify weaknesses and to provide suggestions for improvements, and **make the**

Table 4: Pilot testing and evaluation of different groups.

	CSU pilot testers	Non-CSU pilot testers	AAM workshop participants	Online users
What are the characteristics of the trainees?				
• Demographics	Yes	Yes	Yes	Yes
• Highest educational degree	Yes	Yes	Yes	Yes
• Research role	Yes	Yes	Yes	Yes
How accessible were the training modules?				
• Module 4	Yes	Yes	Yes	No
• Module 5	Yes	Yes	No	No
• Module 6	Yes	No	Yes	No

> **evaluation and feedback data** available to NIGMS staff.”

Learning objectives These are what we’re trying to determine were achieved by the training modules.

Pilot / text group evaluation:

- Work with GAUSSI to use some students as pilot testers?
- Recruit researchers / faculty as pilot testers?
- Work with CSU’s Research Ethics group to figure out ways to pilot?

The key goal of this project is to develop tools that are easy to use by a broad range of applied metabolomics researchers. We will therefore conduct regular short (two hours) user testings several times per year and one long (two days) user testing session per year. The user testing groups will consist primarily of student trainees involved in applied metabolomics research from a range of departments at Colorado State University (see LOS, Clark, De Long, Heuberger). The shorter testing sessions will be used to test stable versions of the developed R packages immediately before they are published to the Comprehensive R Archive Network (CRAN). These testing sessions will ask participants to work through package tutorial vignettes and other test cases and will focus on identifying aspects of the package that cause unwanted behavior on certain computer systems or when users provide unexpected input. Further, these shorter testing sessions will be used to identify sections of vignette tutorials or help files that are unclear to targeted users. The longer, two-day testing sessions will be more open and will provide participants with open-ended metabolomics data analysis challenges, using data from *Metabolomics Workbench* and the *National Metabolomics Data Repository* that differ from the data used to develop the tools. Participants will be scheduled into groups, allowing them to participate during the two days while meeting outside obligations like classes and meetings. Participants will be encouraged to use both the tools we develop here, as well as any other available R tools, to complete these challenges in groups in a “hackathon”-style structure. Participants will be given guidance on how to use GitHub to work in groups and share final results from the challenge, a framework Dr. Anderson has successfully used in a previous similar event at Colorado State University (Figure 1). This longer testing will help us identify limitations in the usability of the tools we will develop here, validate the tools using separate data, scope future development aims by identifying analysis tools that users would have liked to have to help with these open-ended challenges but that were not yet available in the R environment, and compare the tools we develop to existing metabolomics data analysis and visualization tools. Dr. Anderson (PI) has run several two-hour user testing sessions with students from various departments of Colorado State University prior to releasing R software packages [**? ?**]. Further, in April 2016, she led a longer, two-day user testing session through a Weather Data Hackathon at Colorado State University (Figure 1). Around 15 people participated, including undergraduate students, graduate students, postdoctoral fellows, and professors from CSU's Departments of Atmospheric Sciences, Civil & Environmental Engineering, Microbiology, and Statistics. Some of the ideas and code developed during this Hackathon have since led to development and publication of open source software [**? ?**].

Participant feedback: "I just wanted to thank you again for the opportunity to participate in the hackathon. It was inspiring to work with students from other departments, and I thought you created a very positive and engaging environment for collaboration. **I feel like this experience has in some ways changed the way I think research should be done**. If you are ever looking to host something like this again, I would certainly be interested in working with you again!

Figure 1: Some of the approximately 15 undergraduate students, graduate students, postdoctoral fellows, and professors who participated in a two-day Weather Data Hackathon at Colorado State University in April 2016.

Long-term evaluation:

- Google Analytics for online book. How often are people accessing the book? How long are they spending on the book website? Where are the people accessing the book?
- YouTube analytics for the embedded videos. How often are people accessing the book? How long are they spending on the book website? Where are the people accessing the book?
- Quiz for each chapter of the book? Use to evaluate how well they've mastered the material? (Possibly could use embedded Shiny apps for this? Other ways to do this?)
- Rating options for each chapter of the online book? Usefulness? What they learned?
- Survey within each chapter of the online book? Educational level, demographic characteristics.

Things we want to know about the final training materials:

Quantitative:

- How many people have accessed the online book?
- How are book users distributed across the U.S.? Are there many international users accessing the book?
- When someone accesses the online book, how long on average do they spend reading or exploring the book? How many users are spending more than 10 minutes on the book per visit (about the minimum time for a module)?
- How many people have downloaded the entire book?
- How many people have commented on or made suggestions for the book through its "Issues" page?

- How many people have viewed each of the tutorial videos?
- What percent of people who begin to watch each video finish it?
- For each video, are there common locations in the video where many people turn it off?
- For online quizzes, how many people take each quiz?
- For each quiz question, what percent of people answer it correctly? Are there some questions that many trainees struggle with after going through the training material?
- For audio files of Co-Is answering discussion questions, how often do people listen to those? How many people listen to the complete file?
- For applied exercise that include data or files to download, how many people have downloaded the files?
- What is the demographic profile of users (age, gender, race, ...)?
- What is the educational profile of users (highest degree, area of research)?
- What is the professional profile of users (current position, typical research tasks for which the trainee is responsible)?
- For each module, which elements did a trainee use (online book text, video tutorial, additional content like discussion questions and quizzes)? How useful did the trainee find each of the elements he or she used (on a Likert scale)?
- For each module, did the user feel that he or she had adequate prior knowledge to follow the material in the module, or were there unstated prerequisites that the trainee lacked and felt limited their ability to learn from the training module?
- For each module, does the trainee anticipate making any changes to his / her research practice based on having taken this module?
- For each module, what percent of the material was new to the trainee (versus things he or she had already learned or heard about outside of the module)?

Qualitative:

- What types of suggestions and comments have people made about the book through its “Issues” page? Are these mostly to fix typos, or are there substantive questions? Have users helped to identify areas with which they struggled?
- What are the users’ goals in taking the training modules?
- How did the user decide which modules to take?
- For each module, what are elements that the trainee wished had been covered but were not?
- For each module, were there elements that the trainee did not find useful that could have been excluded?
- For each module, how did the trainee use any additional materials (discussion questions, quiz questions, applied exercises) that came with the module?
- For each module, how does the trainee plan to change his or her research practice based on having taken the module?
- Did the trainee consider using other training materials to learn this material (e.g., on-campus courses, online MOOCs, other books or video series)? If so, what aspects of our training materials led to the decision to pick them?
- Why did the trainee decide to use these training modules (e.g., requirement for a course, PI told them to, self-motivated to improve their research practices, interested in learning new technology)?

Levels of evaluation:

- **Final users of the online book and videos.** These could potentially be from anywhere in the world, and for many we won't have great ways to contact them.
- **Workshop attendees for the workshops we plan to propose and do at national microbiology / immunology conferences.** For these people, we could definitely do a survey before to get information on demographics, education level, interest in the training materials, etc. We could also do a post survey to find out what they learned, how helpful it was, what they found confusing, etc. Finally, we could get their email addresses to ask some longer-term evaluation questions (e.g., How are they using what they learned 1–2 years after taking the workshop? How much did they retain in terms of principals, implementation, and examples?). We can also use questions that are asked during the workshops and areas where additional materials (applied exercises, quizzes, discussion questions) are problematic to help us hone our training materials.
- **Early online users.** We will plan to develop and post the text and some of the additional educational materials (e.g., quizzes, discussion questions) online through GitHub *as we write the book and develop the materials*. We will use social media to invite people to try out the book as we develop it. Based on previous work developing online books, we have found that this open development process can help attract users very early in the process, and that these users are often very helpful in providing feedback as the book is developed. We will elicit their feedback through GitHub ("Issues" page will be the main forum for them to post comments and suggestions).
- **CSU pilot users.** We can ask these pilot users many questions, both before and after the pilot testing. Further, we will have access to ask them longer-term outcomes, as well as to ask at the department level how the use of these training materials by a number of people in the department has changed research practices and what is considered "best practice" for research in the department (i.e., a 'bubble up" effect).
- **Pilot users from other institutions.** Similar to CSU pilot users, although we'll have a bit less access for determining longer-term and department-wide outcomes.

C.5 Dissemination Plan

> "A specific plan must be provided to disseminate the finished training modules **nationally** and make them **freely accessible**. In addition, links to these modules will be posted and maintained on the NIGMS web site."

We will create an online tutorial book, since providing tutorials, example code, and example datasets can substantially improve the ability of new users to learn software tools [1]. We will use GitHub's free "Pages" web publishing framework to publish the book freely online, and we will also submit it to the *bookdown.org* website under a Creative Commons license. Dr. Anderson (PI) has previously created two *bookdown*-based books, *R Programming for Research* and *Mastering Software Development in R*. Both are publicly and freely available online under the Creative Commons license (see LOS, Xie).

We will publish the video lectures using the YouTube platform and embed these videos within the online book. The videos, like the book, will be published under a Creative Commons license.

For many online training materials on the principals and implementation of computationally reproducible research, the target audience is statisticians and other researchers who focus on data analysis. This audience now has access to many excellent resources for improving research reproducibility at the data analysis stage. Our target audience is instead researchers who focus on conducting experimental, laboratory-based biomedical research and whose research tasks are more focused on the earlier research steps of running experiments, recording experimental data, and pre-processing that data, prior to data analysis steps. We are focusing our dissemination plans on this target audience, for whom training materials on improving the reproducibility of later data analysis steps might be of limited relevance.

We will also take steps to make sure that our target audience—laboratory-based biomedical scientists—are aware of these training materials. We will travel in Years 02 and 03 to one national microbiology ([conference]) and one national immunology ([conference]) conference. We will submit proposals to these conferences to present half-day (student?) workshops covering why and how to improve reproducibility in experimental data recording and pre-processing for biomedical microbiology and immunology research. We will also submit poster abstracts to present and discuss the training materials as part of each conference's poster sessions. We have budgeted for two members of our team (the PI and one co-I) to attend each of these conferences to help disseminate the training materials produced by the project. [CSU's microbiology seminar series? Conferences on-campus at CSU? Are there ones that are repeated every year?]

The PI has previously had substantial success in disseminating online training materials. She is the co-instructor of a five-course specialization on *Mastering Software Development in R* through the Massive Open Online Course platform Coursera. The first course in this series has had over 25,000 participants since it was opened in fall 2016. An accompanying online book on the LeanPub platform has been accessed by [x] people. The PI, however, does not have a presence in the microbiology and immunology community, and so the co-investigators, who are heavily involved in this community, will form a key bridge in making sure our target audience is aware that these training materials are available.

C.6 Timeline

> "Provide a timeline for **module development**, **piloting and refinement**, **dissemination**, **evaluation**, and **maintenance**. This timeline must propose making the training publicly available within two years of the award date."

D Works cited

References

[1] Markus List, Peter Ebert, and Felipe Albrecht. Ten simple rules for developing usable software in computational biology. *PLoS computational biology*, 13(1):e1005265, 2017.

[2] Benjamin S Baumer. Lessons from between the white lines for isolated data scientists. Technical report, PeerJ Preprints, 2017.

[3] Stephen Altschul, Barry Demchak, Richard Durbin, Robert Gentleman, Martin Krzywinski, Heng Li, Anton Nekrutenko, James Robinson, Wayne Rasband, James Taylor, et al. The anatomy of successful computational biology software. *Nature biotechnology*, 31(10):894–897, 2013.

[4] Hadley Wickham. *R packages: organize, test, document, and share your code*. " O'Reilly Media, Inc.", 2015.

[5] Yihui Xie. *Dynamic Documents with R and knitr*, volume 29. CRC Press, 2015.

[6] Yihui Xie. *Bookdown: Authoring Books and Technical Documents with R Markdown*. CRC Press, 2016.

[7] Akshay Mittal Sharvanath Pathak and Trevor Bannard. Rhadoop: an improved execution environment for restricted map reduce programs. *R package*, 2014.

[8] Javier Luraschi, Kevin Ushey, JJ Allaire, and The Apache Software Foundation. *sparklyr: R Interface to Apache Spark*, 2017. URL https://CRAN.R-project.org/package=sparklyr. R package version 0.6.3.

[9] Jeremy Goecks, Anton Nekrutenko, and James Taylor. Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences. *Genome biology*, 11(8):R86, 2010.

[10] Mark A Walker, Ravi Madduri, Alex Rodriguez, Joseph L Greenstein, and Raimond L Winslow. Models and simulations as a service: Exploring the use of galaxy for delivering computational models. *Biophysical journal*, 110(5):1038–1043, 2016.

[11] Shannon E Ellis and Jeffrey T Leek. How to share data for collaboration. Technical report, PeerJ Preprints, 2017.

[12] Karthik Ram. Git can facilitate greater reproducibility and increased transparency in science. *Source code for biology and medicine*, 8(1):7, 2013.

[13] Zev Ross, Hadley Wickham, and David Robinson. Declutter your r workflow with tidy tools. *PeerJ Preprints*, 5:e3180v1, 2017.

[14] Julia Silge and David Robinson. tidytext: Text mining and analysis using tidy data principles in r, 2016.

[15] Hadley Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer, 2nd edition, 2016.

[16] Hadley Wickham and Garrett Grolemund. *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. " O'Reilly Media, Inc.", 2016.

[17] Jennifer Bryan. Excuse me, do you have a moment to talk about version control? Technical report, PeerJ Preprints, 2017.

[18] Julia S Stewart Lowndes, Benjamin D Best, Courtney Scarborough, Jamie C Afflerbach, Melanie R Frazier, Casey C O'Hara, Ning Jiang, and Benjamin S Halpern. Our path to better science in less time using open data science tools. *Nature ecology & evolution*, 1(6):160, 2017.

[19] Allan M Reviewer-Miller. Review of r for data science: Import, tidy, transform, visualize, and model data by hadley wickham and garrett grolemund. *ACM SIGACT News*, 48(3):14–19, 2017.

[20] Amelia McNamara. On the state of computing in statistics education: Tools for learning and for doing. *arXiv preprint arXiv:1610.00984*, 2016.

[21] Stephanie C Hicks and Rafael A Irizarry. A guide to teaching data science. *The American Statistician*, (just-accepted):00–00, 2017.

[22] Ben Baumer. A data science course for undergraduates: Thinking with data. *The American Statistician*, 69(4):334–342, 2015.

[23] Daniel T Kaplan. Teaching stats for data science. *PeerJ Preprints*, 5:e3205v1, 2017.

[24] Julian Stander and Luciana Dalla Valle. On enthusing students about big data and social media visualization and analysis using r, rstudio, and rmarkdown. *Journal of Statistics Education*, 25(2): 60–67, 2017.

[25] Benjamin S Baumer, Daniel T Kaplan, and Nicholas J Horton. *Modern Data Science with R*. CRC Press, 2017.

[26] Rafael A. Irizarry and Michael I. Love. *Data Analysis for the Life Sciences with R*. Chapman and Hall / CRC, 2016.

[27] Greg Wilson. Software carpentry: lessons learned. *F1000Research*, 3, 2014.

[28] Aleksandra Pawlik, Celia WG van Gelder, Aleksandra Nenadic, Patricia M Palagi, Eija Korpelainen, Philip Lijnzaad, Diana Marek, Susanna-Assunta Sansone, John Hancock, and Carole Goble. Developing a strategy for computational lab skills training through software and data carpentry: Experiences from the elixir pilot action. *F1000Research*, 6, 2017.

[29] Julia Silge and David Robinson. *Text Mining with R: A Tidy Approach*. " O'Reilly Media, Inc.", 2017.

[30] Paul J McMurdie and Susan Holmes. phyloseq: an r package for reproducible interactive analysis and graphics of microbiome census data. *PloS one*, 8(4):e61217, 2013.

[31] Taylor Arnold. A Tidy Data Model for Natural Language Processing using cleanNLP. *The R Journal*, 2017. URL https://journal.r-project.org/archive/2017/RJ-2017-035/index.html.

[32] Sam Tyner, François Briatte, and Heike Hofmann. Network Visualization with ggplot2. *The R Journal*, 9(1):27–59, 2017. URL https://journal.r-project.org/archive/2017/RJ-2017-023/index.html.

[33] TC Hsieh, KH Ma, and Anne Chao. inext: an r package for rarefaction and extrapolation of species diversity (hill numbers). *Methods in Ecology and Evolution*, 7(12):1451–1456, 2016.

[34] Tengfei Yin, Dianne Cook, and Michael Lawrence. ggbio: an r package for extending the grammar of graphics for genomic data. *Genome Biology*, 13(8):R77, 2012.

[35] URL https://rstudio.github.io/rstudio-extensions/rstudio_project_templates.html.

[36] Ben Marwick, Carl Boettiger, and Lincoln Mullen. Packaging data analytical work reproducibly using r (and friends). *The American Statistician*, (just-accepted), 2017.

[37] Hilary Parker. Opinionated analysis development. *PeerJ Preprints*, 5:e3210v1, 2017.

[38] Stephen R Piccolo and Michael B Frampton. Tools and techniques for computational reproducibility. *GigaScience*, 5(1):30, 2016.

[39] Mine Cetinkaya-Rundel and Colin W Rundel. Infrastructure and tools for teaching computing throughout the statistical curriculum. Technical report, PeerJ Preprints, 2017.

[40] Wolfgang Huber, Vincent J Carey, Robert Gentleman, Simon Anders, Marc Carlson, Benilton S Carvalho, Hector Corrada Bravo, Sean Davis, Laurent Gatto, Thomas Girke, et al. Orchestrating high-throughput genomic analysis with bioconductor. *Nature methods*, 12(2):115–121, 2015.

[41] Likit Preeyanon, Alexis Black Pyrkosz, and C Titus Brown. Reproducible bioinformatics research for biologists. *Implementing Reproducible Research*, page 185, 2014.