THE 5Ws AND 1H OF TERM PROJECTS IN THE INTRODUCTORY DATA SCIENCE CLASSROOM

MINE ÇETINKAYA-RUNDEL
Duke University,
RStudio
mc301@duke.edu

MINE DOGUCU
University of California Irvine
mdogucu@uci.edu

WENDY RUMMERFIELD
University of California Irvine
wrummerf@uci.edu

ABSTRACT

Many data science applications involve generating questions, acquiring data and preparing it for analysis—be it exploratory, inferential, or modeling focused—and communicating findings. Most data science curricula address each of these steps as separate units in a course or as separate courses. Open-ended term projects, however, allow students to put each of these steps into practice, sequentially and iteratively. In this paper, we discuss what we mean by data science projects, why they are crucial in introductory data science courses, who works on these projects and how, when in the term they can be implemented, and where they can be shared.

Keywords: *Statisitcs education research; Data science; Teaching; Curriculum; R language; Git*

1. INTRODUCTION

Many data science applications involve generating questions, acquiring data, and preparing it for analysis—be it exploratory, inferential, or modeling focused—and communicating findings. Most data science curricula address each of these steps as separate units in a course or as separate courses. Open-ended term projects, however, allow students to put each of these steps into practice, sequentially and iteratively. Such projects also provide an opportunity for learning by doing, which encourages student creativity, engagement, and independence. Even though term projects can be highly beneficial for mastery, they can also be a source of frustration for learners and instructors if not designed and executed well.

In an effort to streamline the narrative, going forward we refer to these open-ended term projects as "projects." We purposefully avoid the term "final project" or "end-of-term project" as we believe that these projects will be much more successful in achieving their learning goals if students start working on them early in the term and develop them throughout.

There are many questions that instructors may be asking regarding projects: Should I assign a project? What should its scope be? When should it run? How should it be assessed? Making decisions that answer these questions prompt even more specific questions such as, "If students work in teams, how do we ensure individual accountability and learning?" There is not a one size fits all answer to such questions since their answers depend on many factors such as class size, student background, and instructor teaching load.

One challenge in being prescriptive about answers to these questions is the lack of consensus on topics that should be covered in introductory data science courses and the amount of emphasis placed on each topic (Çetinkaya-Rundel & Ellison, 2021). Some introductory data science courses mainly discuss data cleaning and visualization, while others place a large emphasis on probability. In our courses, we cover

Statistics Education Research Journal, 21(2), Article 4. https://doi.org/10.52041/serj.v21i2.37

exploratory data analysis (numerical and visual summaries), data wrangling (importing, tidying, and transforming), modeling (linear and logistic regression), inference (primarily bootstrap intervals), and data science ethics. Our courses are non-calculus and non-linear algebra-based, designed for an audience with little to no experience with statistics, computer science, data science, or mathematics at the university level. Our courses are targeted at lower-level undergraduates (i.e., first and second years) majoring in any discipline. We teach R for data analysis (R Core Team, 2020) using RStudio as the integrated development environment in which students write their R code (RStudio Team, 2021), primarily along with their narrative in R Markdown documents (Allaire et al., 2021). We also use *Git* for version control and *GitHub* for hosting the Git repositories as well as for facilitating collaboration between team members.

In this paper, we pose questions that we have asked ourselves over the past few years of data science teaching and answer them based on our experience, supported by the literature on data science, statistics, computer science, and broader education literature. We aim to (1) convince data science instructors that incorporating projects into their curriculum is not just beneficial, but also crucial, (2) help them answer questions they will inevitably need to answer as they design projects by providing our opinionated suggestions, and (3) provide concrete resources for project design and assessment that instructors can adopt and reuse in their courses. Additionally, we frame projects based on the computational infrastructures we use; however, many of the recommendations and examples we present can be adapted for data science courses using different computational tools.

2. WHAT?

2.1. *WHAT* EXACTLY DO WE MEAN BY A PROJECT?

Our project assignments come with the following:

> *TL; DR (too long; didn't read): Ask a question you're curious about and answer it with a dataset of your choice. This is your project in a nutshell.*

This short prompt is indicative of the amount of freedom students were given in their projects, which were designed to promote creativity and individuality in the research question, dataset, approach, analysis, and communication of results. We followed this short prompt with a thorough assignment description, clearly stating expectations and assessment criteria so that students knew what exactly what was expected of them and how they would be assessed. A sample project assignment is provided in Appendix A.

Specifics of project assignments vary from course to course; however, there are a few pillars we recommend sticking to:

- Students should find their own dataset and generate their own research question(s).
- There should be a proposal stage where students get ample feedback.
- Projects should be evaluated based on depth (appropriateness and implementation of methods that answer the research questions), not breadth (using all methods taught in the course).
- For team projects, individual student scores should have a personal accountability component.
- Students should have an opportunity to see other projects in the class through peer review and/or in-class presentations.
- The project should be a substantial component of the course grade (e.g., 30%–40%), reflecting the percentage of the academic term students are expected to be primarily working on their projects.

To get an idea of what a project submission may look like consider the following example: A team may want to work with data from the National Student Satisfaction Survey, an annual United Kingdom survey that asks students what they like and what could be improved about their universities (https://www.thestudentsurvey.com). The team augments the survey data with information scraped from Wikipedia, such as university location, public/private status, and size, as well as latitude and longitude information using the Google Maps API. At the exploratory stage before the proposal, the team designates the university satisfaction score as the outcome and proposes to investigate if there is a north-south divide

for university satisfaction scores. At this stage, the instructor provides feedback on how response rates for various universities should be addressed as well as suggesting other possible explanatory variables and recommendations for visual display of the data.

2.2. *WHAT* ARE THE LEARNING GOALS OF A PROJECT?

Projects are a wonderful way to assess the student learning goals of a course, but they should not be used to gauge whether every single learning goal of the course has been achieved. Instead, the expectation should be that students demonstrate proficiency broadly by:

- asking meaningful questions,
- searching for answers using the relevant methods and techniques from the course,
- using the computational output to answer these questions,
- interpreting and communicating the results appropriately, and
- being transparent and critical about reliability, validity, and ethical considerations in all stages of the project.

Projects can also allow students to go beyond the course content, therefore, "relevant" methods do not have to be limited to what is covered in the course if it is feasible to support students who choose to pursue alternative methods. As well, finding and preparing data to be analyzed is an important data science skill. It also usually requires being creative with found data, which is another important learning outcome that projects can address.

3. WHY?

Why should an introductory data science course include a project?

Timed exams have long been used for assessment; however, achievement-based testing only measures the specific content that is being tested (Bell, 2010) and is not representative of authentic problems faced by data scientists, who combine skills from mathematics, computer science, and statistics to use data to describe the world (De Veaux et al., 2017). Projects allow students to apply the methodological knowledge they acquire to any dataset they choose, highlighting the interdisciplinary nature of data science. In the course described above, the mathematical prerequisites have been considerably flattened, making this type of course accessible to a wide variety of learners. This means that regardless of students' backgrounds, they can learn to handle data at all stages: acquisition, transformation, and analysis. We can teach this most effectively by allowing students to work on novel projects with real datasets that have not been pre-processed (Wild et al., 2011). Further, it has been shown that students who create their own research questions and collect their own data may feel more invested in the project, increasing the perceived value of statistics and data science (Bailey et al., 2013). Both the *Guidelines for Assessment and Instruction in Statistics Education: College Report 2016* (GAISE College Report ASA Revision Committee, 2016) and the *Curriculum Guidelines for Undergraduate Programs in Data Science* (De Veaux et al., 2017) recommend that students work with real data on topics that are relevant and interesting to them. Projects comply well with these recommendations.

In undergraduate statistics curricula, capstone courses and consulting courses aim to help students synthesize the courses taken throughout their degree (Smucker & Bailer, 2015) as a valuable way to help students prepare for the professional world. We propose incorporating projects into introductory courses for early exposure to and experience with the full data science cycle. Starting with projects in the introductory course may give a better idea of what is to follow in the major and prepare students for working on longer projects with higher stakes (Spurrier, 2001). It can also be a good preparation for extra-curricular data-centric activities like the American Statistical Association, *DataFest*, an undergraduate data science competition where teams of students work over a weekend to find and share meaning in a large, rich, and complex data set (Gould & Çetinkaya-Rundel, 2013). Experience with open-ended projects, especially ones

completed in teams, not only prepares students for such events, but it also gives them the confidence and motivation to participate in them in the first instance.

Projects that span the full data science cycle also provide an opportunity to consider each stage of this cycle through the lens of ethics. For example, if students are scraping data off the web for their project, they need to first confirm that it is indeed ethical to do so. Additionally, if they are building predictive models, they should consider the ethical implications of the models they fit. Regardless of which methodology they use, students should also consider any bias they might have introduced at any stage of the project as well as potential misuses or unethical uses of their findings.

If designed correctly, projects also give learners the opportunity to hone their professional skills like written and oral presentations, technical writing, collaboration, and effective communication (Wild et al., 2011). Students can enhance skills useful to the workforce such as time management and communication, both oral and written. In team projects, students get experience working with their peers, practice active listening, self-reliance, and creative thinking.

Studies have reported the effectiveness of using projects in the classroom through increased exam scores and higher grades (Geier et al., 2008). Projects also offer an opportunity to go beyond the course curriculum (e.g., new modeling techniques like Poisson regression or new R packages like ggplot2 extensions). Getting students to a point where they can learn more on their own to build on the foundations acquired in the course is a very welcome outcome, but it should be noted that this typically requires more guidance from the teaching team, which may be quite demanding in courses with large cohorts of students.

When working on projects, students are often inspired to learn more (Lazar et al., 2011). Many institutions who perform projects like these are "blown away" by the level and quality of the final submissions with some students going above and beyond by submitting papers to undergraduate journals (White, 2019). Students such as these can choose to publicly share their projects and build their data science portfolio in preparation for future interviews.

4. WHO?

In order to answer, "*Who* works on the projects?" instructors need to decide whether students work in teams or individually. We have taught introductory data science courses in different educational contexts with class sizes ranging from 7 to 300+ at liberal arts colleges, as well as public and private research universities. Regardless of the educational context, we recommend that data science projects are designed as team projects to better reflect how data science is done in the real world. We recommend teams of 4–5 students. This size allows for a functioning team even if a team member is not present for a meeting or drops the class, while still not being too large that it becomes difficult to assign tasks to each member or schedule meetings outside of class. In this section, we discuss the benefits and challenges of teamwork from cognitive, social, and logistical perspectives, as well as share strategies for overcoming the challenges.

When students work individually, they are in charge of the entire cognitive contribution to the project. In team projects, however, each student is in charge of the task/s assigned. Without intentional strategies, however, a student may end up completing a team project with little to no contribution. To avoid this, we recommend two strategies:

1. Use task management systems or apps that transparently link progress to contribution. Hosting projects in a GitHub repository allows users to create *issues* (tasks) and assign them to members (GitHub, 2021a), creating a to-do list that invites threaded discussions that can be *closed* (marked as done) with specific *commits* (changes) that are *pushed* (submitted) to the project repository. This not only helps students be more deliberate with their contributions, but it also makes it clear to team members, and instructors, how they contributed.
2. Enable individual accountability by allocating points for team members to evaluate each other's contribution. Ideally, all team members contribute equally, and these points do not end up changing individual project scores. In instances of unbalanced individual contribution to teamwork, these points reward those who contributed more.

Neither the number of commits nor peer evaluations alone tell the full story of an individual students' contribution to the project; however, these pieces of information in combination and over time can help draw a fairly accurate picture of how individual team members contributed to the project. While quality and quantity of commits are not the same thing, zero commits throughout the duration of the project is an informative piece of information. In our experience, scores of students in most teams tend to vary based only on their peer evaluation score. However, in few cases each semester where a student has been disengaged from the teamwork for an extended period of time, the score awarded for the project can differ greatly from the other team member scores, or the person might not be eligible for any of the project points.

These strategies help students manage teamwork and team dynamics, but do not ensure well-working teams. Therefore, while logistically teamwork may seem appealing for instructors from a workload perspective, time needs to be allocated to managing team dynamics. We recommend starting teamwork on day one, despite formal project work starting later in the course. One approach, supported by the team-based learning literature (Michaelsen & Sweet, 2004), is having students stay in the same teams throughout the entire term and work in those teams on all team-based assignments. Another approach is to shuffle the teams for each assignment. Either way, periodic within team feedback supplemented by instructor intervention (e.g., team meetings) early in the course can help teams gradually settle into a well-working dynamic.

Teams can be formed by instructor assignment or self-selection. In our experience, when self-selection option is given, students tend to team with people that they are already friends with or happen to be sitting next to. We believe that a more principled approach for forming teams is needed as students will be working together for a long time on an assignment that is a big portion of their final grade. When doing team assignments, instructors should be: wary of logistical considerations (e.g., grouping students in the same lab section together); should consider leveraging data collected via pre-course surveys on statistical, computing, communication, and social skills; and should recruit the help of teaching assistants who might get to know the students better as they tend to see them in smaller groups. It is also crucial to consider diversity and aim to ensure that no student feels under-represented in their team. This is likely the hardest goal to achieve fully since there is no perfect measure of diversity. To avoid inquiring about self-identity we instead rely on information like major, class year, and so on.

In this section we have described our approach for designing the teamwork aspect of projects in introductory data science courses. We also mentioned that teams will work best for projects if students work in teams throughout the course as well. Vance (2021) provides a comprehensive overview of using team-based learning to teach data science that includes valuable insights into how to fully integrate teams into the data science classroom. Last but not least, when students work in teams, the quality of work is better than even the best student's individual work. In our experience, the benefits of teamwork outweigh the challenges that come with it. We regard working in teams as an essential skill to be acquired in a data science course.

5. HOW?

5.1. *HOW* ARE PROJECT DATA SELECTED?

As Cobb (2015) says, "Nothing motivates students like choosing their own question and being the first to offer an answer" (p. 277). Logistically, "choose your own dataset" is a feature that is difficult to incorporate into weekly homework assignments and computational labs. Success with such an approach, at a minimum, requires a proposal and feedback stage that is difficult to turn around on a regular cadence in short time intervals. It is possible to give students opportunities for some creativity and personalization into these smaller assignments by asking students to pick variables of interest from a given dataset, fit various models, and compare their predictive performance. However, allowing students to completely design their own assignment is practically not feasible. Therefore, the project is the perfect opportunity for students to both find their own dataset and define their own research question. Additionally, the ownership students have for their projects can help increase the motivation throughout the semester. Even more, variety in

projects can help the final presentations not feel repetitive and showcase the diversity of applications in data science.

Implementing projects is easier said than done as there are many logistical challenges that come with this approach. It also means that the course should explicitly teach importing data from external sources as well as cleaning and tidying the data. Importing rectangular data (e.g., CSV files and Excel spreadsheets) should, at a minimum, be covered. Since data scientists regularly work with non-tabular data as well (e.g., hierarchical data from JSON files and text data), it is important for students to learn how to work with such data within the course curriculum or be allowed to use them for their projects. The real challenge teams picking their own data is being able to gauge whether a dataset of interest is one they can meaningfully engage with using methods they learned in the course. Therefore, we recommend that instructors provide guidance on where (and where not) to find datasets, without limiting students' freedom to go beyond those resources. Note that guidance on picking data is different than providing a list of datasets to pick from, however broad that list might be.

Although the "list of datasets to pick from" approach might be productive for shorter courses (without enough time to provide feedback on proposals), it ultimately does not give students the chance to learn how to assess the feasibility and appropriateness of working with a particular dataset. If timing or resources necessitate not leaving the dataset choice completely open, it is useful to leave that option open for motivated students. If any of the new datasets work well for the project, they can be used to enhance the list for future iterations of the course.

An approach where students are given only guidance, but not a list to pick a dataset from, gives them full control over their project. With a built-in proposal stage, the instructor still has the option to reel them back in, if need be. Below we enumerate a few guidelines we use in our classes.

1. *Think about the question you want to answer first, then look for the appropriate dataset to answer that question. However, the dataset you need to answer the question as you phrased it might not exist or might be difficult to find. Iterate between these two steps until you find a happy medium.*

   Searching for the right dataset means looking for relevant variables and observational units (e.g., person-level data vs. country level data). Library services can be especially helpful in locating data to answer specific questions as well as helping students restate their questions to better match the data available. We allow students to use rich and interesting datasets regardless of whether they can draw inferential conclusions or not. Data that are appropriate for conducting statistical inference tend to come from published studies that include a complete analysis, leaving little room for students' own creativity to generate and answer a question. Additionally, the course curriculum should keep in mind the types of questions students are drawn to for their projects. For example, students are often interested in problems that predict binary outcomes, hence, instructors should consider covering logistic regression.

2. *A good question requires considering multiple variables so that relationships between variables can be evaluated with potential confounding variables in mind.*

   We generally recommend students look for a dataset with at least 50 observations and between 10 to 20 variables ideally of varying types: categorical, discrete numerical, and continuous numerical. These numbers are somewhat arbitrary but communicating these bare minimums to students (allowing for exceptions where necessary) is useful guidance for them.

3. *A comprehensive data dictionary that contains information on how and when the observations were sampled (if relevant) and data were collected is crucial.*

   Without these, it is impossible for students to evaluate the reliability, validity, and ethical considerations of the data for their projects.

4. *Be selective with data from data aggregators like Kaggle* (https://www.kaggle.com/) *and TidyTuesday* (https://github.com/rfordatascience/tidytuesday). The documentation of datasets from these resources varies greatly. These datasets also tend to have sample data analyses of varying quality associated with them. Instructors should provide clear guidelines on how, if at all, to use or

get inspired by sampling these analyses and set clear rules for what constitutes plagiarism vs. inspiration.

These pieces of guidance are relevant for using datasets found "in the wild." Another option is for students to collect their own data, which might be even more motivating for some. Designing a study and collecting data provides an opportunity to think deeply about sampling and/or experimental design. However, not all students embrace this opportunity fully, and many gravitate towards data collection via surveys from friends, distributed via social media. Two common issues with such survey data collected are leading questions (as survey construction is often not part of the curriculum) and convenience sampling bias. These can be mitigated, at least partially, with a proposal stage where students' survey questions and delivery mechanisms are pre-approved. This feedback can prevent teams from making overreaching statements when interpreting their results.

Additionally, some guidance should be provided on working with human subjects' data. Instructors might choose to require Institutional Review Board (IRB) approval regardless of whether the institution requires it for class projects or not. Incorporating the IRB process into the course might require substantial planning, necessitating earlier project proposal deadlines, which may not be feasible. A practical alternative is to address concerns around working with human subjects' data as part of an ethics module that comes before the project proposal deadline.

Yet another approach for collecting data is web scraping. Web scraping allows students to work with current and rich data and provides data wrangling opportunities that meet the learning goals of an introductory data science course (Dogucu & Çetinkaya-Rundel, 2021). This option requires covering web scraping prior to (or as) the project proposal is assigned as well as a discussion on legal and ethical considerations around web scraping.

5.2. *HOW* ARE THE DELIVERABLES DESIGNED?

The first deliverable is a project proposal submitted roughly in the middle of the term, after students have learned data wrangling, tidying, visualization, and importing. The proposal should state the research question, include a preliminary exploratory data analysis, outline the data analysis plan, and contain the dataset to be analyzed in the project along with a data dictionary. This helps students get a real sense of whether the dataset they chose is feasible. Peer review at the proposal stage is also very helpful for getting additional feedback to students and for students to learn from others, as good habits tend to be contagious in a good way.

The second deliverable is the final project. We recommend presentations be a part of the final project. Other parts might be a short executive summary, a full-length write-up, a Shiny app (Chang et al., 2020), or a dashboard with an accompanying summary. The choice of deliverable depends on course length as well as course learning goals.

Requiring presentations might appear daunting at first, especially since they present logistical challenges for large classes and classes with geographically distributed students. Additionally, they can be harder for students who have anxiety about presenting in front of their peers under time pressure. If live presentations are not feasible or preferable, pre-recorded videos can be used. Regardless of how the presentations are delivered, we recommend making time for questions and answers, especially if the presentations make up a significant portion of the course grade. If class time does not allow for all videos to be played, students can be asked to watch them for peer review. An important consideration is projects on sensitive topics (e.g., cancer, suicide, police brutality). While this can create yet another logistical challenge, we recommend sharing titles of projects prior to peer review and presentations and offering students an option to opt out of reading or hearing about a particular project if the topic is triggering for them.

When considering which elements to include in the final deliverable, it may feel like a presentation is too little. Before assigning additional components, think about the amount of effort that students will put into a presentation. If the final assessment includes a written component in addition to a presentation, consider a one-page executive summary, instead of a full report. If requiring a full report, it is helpful to

provide a template, especially for introductory courses as it is likely to be the students' first experience writing such a report. Full reports will take longer to grade, so instructors should judge how much feedback they will be able to give at the end of the course, as well as whether students will read that feedback. Instead, a draft deliverable where students get instructor or peer feedback to improve their writing might be more effective.

In our courses, we require projects to be submitted as GitHub repositories that evolve with the project, from proposal to final submission. We also ask that students present their work as a webpage, via GitHub's Pages feature (GitHub, 2021b). The project website, by default, includes the project summary, links to presentation slides, and, if available, the pre-recorded video. The structure and contents of a template repository we recommend providing to students is given in Appendix B. This includes links to a sample repository on GitHub, the website generated based on the repository, sample automated checks that can be run on the project repositories, and sample issue templates you can use for project feedback.

5.3. *HOW* DO STUDENTS GET FEEDBACK?

Given that the project spans almost half the course, it is important that students get periodic feedback on both the content of their project and their individual contribution to the team. They can either make improvements accordingly or, at a minimum, feel good about the status of their work. We recommend two types and phases of feedback on the content. One of these should come from the instructor (or the teaching team, which might include teaching assistants) and the other should come from peers. Detailed rubrics for both types of feedback can be found in Appendix A and we discuss some of the implementation details below.

Instructor feedback on project proposals is crucial for success on the project, as it can save teams from going down paths that might be frustrating (e.g., the question they want to answer requires a more advanced technique than what will be covered in class) or lead to failure (e.g., the question they want to answer cannot be answered with the data they are planning to use). We recommend giving such feedback in the form of GitHub issues, where each suggested improvement is filed as a new issue (GitHub, 2021a), that students can close with specific commits as they address them. The project proposal should make up a considerable but not substantial portion of the project grade (e.g., 10%).

After one round of instructor feedback and improvements, we recommend one more round of feedback on the content from their peers. Peer review not only introduces students to a standard and valued academic and industry practice, but it also allows students to learn and get inspired by each other's work while they are still working on their own projects. Additionally, it increases the feedback received on projects without adding a heavy burden to instructors—instructors still need to handle the logistics, but this usually takes less effort than providing feedback to all teams.

One challenge with implementing between team peer review is timing. While there isn't a universal solution to this challenge, one way to address it is to dedicate class time (e.g., lab time) for students to work on their projects and another is to fold peer review into another assignment.

The second challenge is deciding on the right level of blinding (single-blind, double-blind, or completely open) and implementing it. Evidence from literature on writing courses suggests that students that participate in double-blind review perform better and provide more critical feedback (Lu & Bol, 2007). Non-blinded review, however, is likely what students will encounter in industry, for example, as part of code review in a data science role. As for implementation, the **ghclass** package offers a set of functions that facilitate peer review on GitHub at various blinding levels (Rundel & Çetinkaya-Rundel, 2021).

While peer review at the proposal stage can be formative, we recommend summative peer review at the presentation stage. Our experience is that students tend to be quite generous on these assessments but with this component constituting no more than 5% of the overall project grade, it tends to be low risk and high reward as it helps students keep their attention during the presentation.

During the development of the project, it is also important that students get feedback on their individual contribution from their teammates. We recommend running team reviews after the proposals as well as shortly before the projects are completed. Giving and receiving feedback at these time points allows

students to understand their team's perception of their contribution and gives learners the opportunity to adjust their behavior if others in their team think they are not contributing sufficiently or effectively. The first of these can be formative and the second summative, feeding into the individual scores of students in a team.

6. WHEN?

Introducing projects too early means students may struggle to pick a topic that is feasible, while too late will not give students time to dive deep and get creative. It is important to teach data import, preferably including web scraping, in time for students to be ready to bring their own datasets to the project proposal.

In Figure 1, we outline timelines for 10-week and 15-week course sessions (typically quarter and semester, respectively) for project components. One crucial difference to note is that in the shorter session, students receive peer and instructor feedback on their proposals in the same week and the re-do of the proposal is optional, due to the tight timeline, while in the longer session students first receive peer feedback and make improvements to their proposals and then submit again for instructor feedback. Additionally, dedicating class time sends a message to the students the projects are just as important as anything else covered in class.

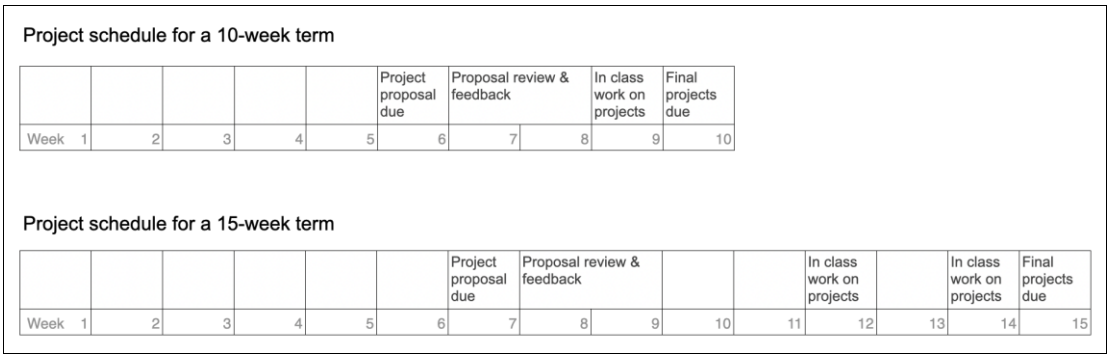

Figure 1. Timing of project components based on session length

Students are not expected to be working on their projects continuously throughout the term but having interim deadlines and periodically allocating class time to the project allows them to keep making progress and not leave everything until the last minute.

7. WHERE?

Where can student projects be shared beyond the classroom?

Due to privacy concerns, we do not require students to share their work publicly. However, we offer them the opportunity to make a copy of their project repositories and make them public after the projects have been graded. We discuss what it means to share work publicly and ask that all team members give consent before any student can *fork* (make a copy that they own) and make their project repository public.

Other potential venues for students to share their work include competitions and conferences. For example, the Undergraduate Class Project Competition (USCLAP) hosted by the Consortium for the Advancement of Undergraduate Statistics Education (CAUSE) takes place twice a year and specifically accepts submissions from class projects (USCLAP, 2021). In addition to potentially winning a monetary award and name recognition in this competition, students can also present their work at Electronic Undergraduate Statistics Research Conference. If the project includes a web application produced with Shiny, the Shiny Contest is another good potential venue because it specifically has a category for new Shiny users (Çetinkaya-Rundel, 2020). If executed well, projects can find a place outside of the classroom.

Whether it is via GitHub, a national competition, or another venue, students should consider sharing their work if it reflects their true potential in data science. Many students list these projects on their resumes, and they regularly come up in student interviews and reference calls with potential employers.

8. DISCUSSION

We cannot envision an introductory data science course without projects. In this paper, we asked questions that any instructor might ask themselves and answered them within the scope of the paper. We have presented what we consider to be good practices for structuring projects in the introductory data science classroom. These practices could be used as a roadmap or inspiration, depending on the goals of each educator and the constraints within which they need to work to achieve them.

Throughout the paper, we have referred to the project workflow from our courses using Git and GitHub. We recommend adopting this workflow for all course assignments, not just the project as it might be challenging to switch workflows between regular course content and projects. We recommend that instructors familiarize students with this workflow starting in the first week of the course. Beckman et al. (2020) and Fiksel et al. (2019) provide further guidance on how to incorporate GitHub in the statistics and data science classrooms. For instructors who want to learn more about using Git and GitHub with R, we recommend *Happy Git with R* (Bryan et al., 2020).

The ideas as well as the infrastructure for the projects that we have presently are based on a specific set of tools that we use in our courses. In addition, the overall goals of the project match closely with our course goals. However, just like our own courses, the projects have their own set of limitation. For instance, we do not cover data collection principles and thus do not allow our students to collect their own data via surveys or experiments. What we presented here is not the only way of adopting projects in the data science classroom. However, based off the blueprint we provide in this paper, instructors could design alternative projects based on the tools that they teach (e.g., Python) and/or the learning objectives of their courses (e.g., data collection).

Another limitation of these projects is that, despite the iterative development of project focus and research questions through the proposal and feedback stages, it is difficult, if not impossible, to ensure that all teams come up with research questions that are both scientifically sound and carry societal importance. While possible in an upper-level, smaller consulting course for students to work with field experts to develop a question (or work on a question already developed by an expert), this strategy is not scalable to large, introductory courses.

We have discussed the many opportunities projects provide students; however, it is important to also acknowledge the additional work that instructors must put into planning and organizing projects. To provide concrete help with implementation, we have provided templates in the appendices of this paper. Instructors who are inspired to incorporate a project should note that something will have to adjust their existing courses, for example, reduce the number of other assignments or drop particular topics. While neither of these is easy, we believe that the benefits of including an open-ended project in the introductory data science course outweigh its costs. Despite the challenges, projects provide concrete benefits for instructors as well. They are a great source of new datasets, they enable instructors to get to know their students' interests, and most importantly, they provide a wonderful opportunity for instructors to watch their students be creative with what they have learned.

REFERENCES

Allaire, J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., … Iannone, R. (2021). *RMarkdown: Dynamic documents for R*. https://github.com/rstudio/rmarkdown

Bailey, B., Spence, D. J., & Sinn, R. (2013). Implementation of discovery projects in statistics. *Journal of Statistics Education*, *21*(3), Article 1. https://doi.org/10.1080/10691898.2013.11889682

Bell, S. (2010). Project-based learning for the 21st century: Skills for the future. *The Clearing House: A Journal of Educational Strategies, Issues and Ideas*, *83*(2), 39–43. https://doi.org/10.1080/00098650903505415

Bryan, J., STAT 545 TAs, & Hester, J. (2020). *Happy Git and GitHub for the useR*. https://happygitwithr.com

Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2020). *Shiny: Web application framework for R*. https://CRAN.R-project.org/package=shiny

Cobb, G. (2015). Mere renovation is too little too late: We need to rethink our undergraduate curriculum from the ground up. *The American Statistician*, *69*(4), 266–282. https://doi.org/10.1080/00031305.2015.1093029

Çetinkaya-Rundel, M. (2020, February). *Shiny Contest 2020 is here!* RStudio. https://blog.rstudio.com/2020/02/12/shiny-contest-2020-is-here

Çetinkaya-Rundel, M., & Ellison, V. (2021). A fresh look at introductory data science. *Journal of Statistics and Data Science Education*, *29*(S1), S16–S26. https://doi.org/10.1080/10691898.2020.1804497

De Veaux, R. D., Agarwal, M., Averett, M., Baumer, B. S., Bray, A., Bressoud, T. C., … Ye, P. (2017). Curriculum guidelines for undergraduate programs in data science. *Annual Review of Statistics and Its Application*, *4*(1), 15–30. https://doi.org/10.1146/annurev-statistics-060116-053930

Dogucu, M., & Çetinkaya-Rundel, M. (2021). Web scraping in the statistics and data science curriculum: Challenges and opportunities. *Journal of Statistics and Data Science Education*, *29*(S1), S112–S122. https://doi.org/10.1080/10691898.2020.1787116

Fiksel, J., Jager, L. R., Hardin, J. S., & Taub, M. A. (2019). Using GitHub classroom to teach statistics. *Journal of Statistics Education*, *27*(2), 110–119. https://doi.org/10.1080/10691898.2019.1617089

GAISE College Report ASA Revision Committee. (2016). *Guidelines for assessment and instruction in statistics education (GAISE): College report 2016*. https://www.amstat.org/docs/default-source/amstat-documents/gaisecollege_full.pdf

Geier, R., Blumenfeld, P. C., Marx, R. W., Krajcik, J. S., Fishman, B., Soloway, E., & Clay-Chambers, J. (2008). Standardized test outcomes for students engaged in inquiry-based science curricula in the context of urban reform. *Journal of Research in Science Teaching*, *45*(8), 922–939. https://doi.org/10.1002/tea.20248

GitHub. (2021a). *Mastering issues - GitHub guides*. https://guides.github.com/features/issues

GitHub. (2021b). *GitHub pages*. https://docs.github.com/en/github/working-with-github-pages

Gould, R., & Çetinkaya-Rundel, M. (2013). Teaching statistical thinking in the data deluge (pp. 377–391). Springer Fachmedien Wiesbaden. https://doi.org/10.1007/978-3-658-03104-6_27

Lazar, N. A., Reeves, J., & Franklin, C. (2011). A capstone course for undergraduate statistics majors. *The American Statistician*, *65*(3), 183–189. https://doi.org/10.1198/tast.2011.10240

Lu, R., & Bol, L. (2007). A comparison of anonymous versus identifiable e-peer review on college student writing performance and the extent of critical feedback. *Journal of Interactive Online Learning*, *6*(2). https://digitalcommons.odu.edu/cgi/viewcontent.cgi?article=1002&context=efl_fac_pubs

Michaelsen, L., & Sweet, M. (2004). *Team-based learning*. Sterling. https://digitalcommons.georgiasouthern.edu/ct2-library/199

R Core Team. (2020). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org

RStudio Team. (2021). *RStudio: Integrated development environment for R*. RStudio, PBC. http://www.rstudio.com

Rundel, C., & Çetinkaya-Rundel, M. (2021). *ghclass: Tools for managing classes on GitHub*. https://rundel.github.io/ghclass-dev/articles/articles/peer.html

Smucker, B. J., & Bailer, A. J. (2015). Beyond normal: Preparing undergraduates for the work force in a statistical consulting capstone. *The American Statistician*, *69*(4), 300–306. https://doi.org/10.1080/00031305.2015.1077731

Spurrier, J. D. (2001). A capstone course for undergraduate statistics majors. *Journal of Statistics Education*, *9*(1). https://doi.org/10.1080/10691898.2001.11910643

USCLAP. (2021). *USCLAP Competition*. https://www.causeweb.org/usproc/usclap

Vance, E. (2021). Using team-based learning to teach data science. *Journal of Statistics and Data Science Education*. https://doi.org/10.1080/26939169.2021.1971587

White, D. (2019). A project-based approach to statistics and data science. *PRIMUS*, *29*(9), 997–1038. https://doi.org/10.1080/10511970.2018.1488781

Wild, C. J., Pfannkuch, M., Regan, M., & Horton, N. J. (2011). Towards more accessible conceptions of statistical inference. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *174*(2), 247–295. https://doi.org/10.1111/j.1467-985X.2010.00678.x

MINE ÇETINKAYA-RUNDEL
Duke University
mc301@duke.edu

APPENDIX A: PROJECT ASSIGNMENT

The following is a sample assignment for a project where students find their own dataset and work in teams in a GitHub repository.

TL;DR

Ask a question you're curious about and answer it with a dataset of your choice. This is your project in a nutshell.

May be too long, but please do read

The final project for this class will consist of analysis on a dataset of your own choosing. The dataset may already exist, or you may collect your own data using a survey or by conducting an experiment. You can choose the data based on your teams' interests or based on work in other courses or research projects. The goal of this project is for you to demonstrate proficiency in the techniques we have covered in this class (and beyond, if you like) and apply them to a novel dataset in a meaningful way.

The goal is not to do an exhaustive data analysis i.e., do not calculate every statistic and procedure you have learned for every variable, but rather let me know that you are proficient at asking meaningful questions and answering them with results of data analysis, that you are proficient in using R, and that you are proficient at interpreting and presenting the results. Focus on methods that help you begin to answer your research questions. You do not have to apply every statistical procedure we learned. Also, critique your own methods and provide suggestions for improving your analysis. Issues pertaining to the reliability and validity of your data, and appropriateness of the statistical analysis should be discussed here.

The project is very open ended. You should create some kind of compelling visualization(s) of this data in R. There is no limit on what tools or packages you may use but sticking to packages we learned in class is required. You do not need to visualize all of the data at once. A single high-quality visualization will receive a much higher grade than a large number of poor-quality visualizations. Also pay attention to your presentation. Neatness, coherency, and clarity will count. All analyses must be done in RStudio, using R.

A.1. DATA

The dataset you pick should have at least 50 observations and 10 to 20 variables. If you set your hearts on a dataset that has fewer observations or variables than what's suggested here, that might still be ok; use these numbers as guidance for a successful proposal, not as minimum requirements. Picking a dataset that has a variety of numeric and categorical variables can lead to interesting opportunities, and hence is recommended, but is not required.

You can find data wherever you like, but here are some recommendations to get you started. You shouldn't feel constrained to datasets that are already in a tidy format, you can start with data that needs cleaning and tidying, scrape data off the web, or collect your own data.

- Awesome public datasets
- Bikeshare data portal
- Data.gov
- Data is Plural
- Edinburgh Open Data
- CORGIS: The Collection of Really Great, Interesting, Situated Datasets
- Google Dataset Search
- Harvard Dataverse
- NHS Scotland Open Data
- IPUMS survey data from around the world
- Los Angeles Open Data
- NYC OpenData
- Open access to Scotland's official statistics

- PRISM Data Archive Project
- UCI Machine Learning Repository
- UN data
- UK Government Data
- US Government Data
- Youth Risk Behavior Surveillance System (YRBSS)

A.2. DELIVERABLES

You have three deliverables for this project: proposal, presentation, and write-up. Instructions and requirements for each are outlined below.

Proposal

The `proposal.Rmd` file should contain the three following sections.

1. Introduction: In this part you need to introduce the questions you are interested in answering with the data. What observations and variables this question is concerned with? If you can state why this question is important, even better. You should also address ethical concerns about the data you're using, if any.
2. Data: Place your data in the `data` folder and add dimensions and data dictionary to the README in that folder. For your data dictionary, follow the structure and the formatting outlined in the README in the template repo. Then print out the output of `tibble::glimpse()` or `skimr::skim()` of your data frame.
3. Data analysis plan: Your data analysis plan should clearly state the outcome (Y) and predictor (X) variables you will use to answer your question. You should also conduct simple exploratory analysis (including summary statistics and visualizations) to get some preliminary idea about answers to your questions and discuss what the result of this analysis indicates. Additionally, you should describe any next steps you have planned for your project. Note that your plans do not need to be exhaustive, and you will be free to change them later as you see fit, especially as you learn more about statistical techniques that you can use building on your preliminary exploratory analysis.

Presentation

You have 5 minutes (maximum) to present your findings. You can use this time however you like, but you must make sure that each team member says something substantial. Your presentation should not just be an account of everything you tried ("then we did this, then we did this, etc."), instead it should convey what choices you made, and why, and what you found.

You will prepare slides for your final presentation using the **xaringan** package. A template is provided in the `presentation` folder in your template repo, and it's called `presentation.Rmd`. If you choose to include any images other than the ones you create with R (e.g., a ggplot) in your presentation, make sure to create a folder called in `images` in the `presentation` folder and add these images to this folder. Before you finalize your presentation, make sure your chunks are turned off with `echo = FALSE`, except for any code you particularly want to discuss in your presentation, if any.

Summary

Write a short summary (150–200) words of your project that conveys the main points of your presentation to someone who has not watched your presentation. This summary can include the questions your project addressed, the specific methods you have used (e.g., exploratory analysis, logistic regression,

etc.), your key results, and conclusion. If necessary, you can include a plot in this section to show an important finding of your project. This summary should be included in the `README.Rmd` of your repository. If you choose to include a plot, then make sure that you have the chunk option set to `echo = FALSE` so that the R code for the plot can be hidden. Further details of your results and additional plots should be included in your presentation.

A.3. TIPS

- You're working in the same repo as your teammates now, so merge conflicts will happen, issues will arise, and that's fine! Commit and push often and ask questions when stuck.
- Review the grading rubrics and ask questions if any of the expectations are unclear.
- Set aside time to work together with your team as well as on your own.
- When you're done, review the documents on GitHub to make sure you're happy with the final state of your work. Then go get some rest!

A.4. ASSESSMENT

Rubric for the proposal: Instructor evaluation

Category	Descriptor	Scale
Data	Data are provided. README.md in data folder contains codebook. Contains dataset(s) to be used in the project Instructions removed from `README` and codebook added Metadata on the data clearly stated, e.g., *"The dataset has/contains/is comprised of/etc. R observations, each representing […], and C columns."*	0 - 3
Proposal	Structure of the `proposal.Rmd` is well organized as outlined in the instructions. All the parts of the proposal are provided. (1 pt) Structure: `.Rmd` file is updated and is in the repo `.md` file is updated and is in the repo Figures are visible in the `.md` file Uses section headings to organize each part Content: (1 pt) Introduction: Research question(s) clear, cases are stated, variables to be used are explained (1 pt) Data: `glimpse()` or `skim()` output available (2 pts) Data analysis plan: Variable roles are clear, comparison groups (if any) are clear, preliminary exploratory analysis is included and interpreted	0 - 5
Workflow	Data read in from `data/` folder using the **here** package Reasonable number of commits Meaningful commit messages (or at least not an abundance of not meaningful ones) No unexpected/disallowed files	0 - 1
Group work	All group members have committed to the repo at least once	0 - 1

Rubric for the proposal: Between team peer evaluation

- Describe the goal of the project.
- Describe the data used or collected.
- Describe how the research question will be answered, e.g., what approaches / methods will be used.

- Is there anything that is unclear from the proposal?
- Provide constructive feedback on how the team might be able to improve their project.
- What aspect of this project are you most interested in and would like to see highlighted in the presentation?
- Provide constructive feedback on any issues with file and/or code organization.
- (Optional) Any further comments or feedback?

Rubric for the final project: Instructor evaluation

To be filled out by course instructor and any teaching staff who is present at the presentations.

Category	Descriptor	Scale
Content	Is the research question well designed and are the data being used relevant to the research question?	0 - 5
Content	Did the team use appropriate statistical procedures and interpretations of results accurately?	0 - 10
Creativity and Critical Thought	Is the project carefully thought out? Are the limitations carefully considered? Does it appear that time and effort went into the planning and implementation of the project?	0 - 10
Slides	Are the slides well organized, readable, not full of text, featuring figures with legible labels, legends, etc.?	0 - 10
Professionalism	How well did the team present? Does the presentation appear to be well practiced? Are they reading off of a script? Did everyone get a chance to say something meaningful about the project?	0 - 5
Teamwork	Did the team present a unified story, or did it seem like independent pieces of work patched together?	0 - 6
Time management	Did the team divide the time well amongst themselves or got cut off going over time?	0 - 4
Executive summary	Does it follow guidance? Is it concise but detailed enough?	0 - 10
Comments	*Optional*, but please add some if you've scored on the higher or the lower end for the content or creativity and critical thought categories.	

Rubric for the final project: Between team peer evaluation

Category	Descriptor	Scale
Content	Is the research question well designed and is the data being used relevant to the research question?	0 - 1
Content	Did the team use appropriate statistical procedures and interpretations of results accurately?	0 - 1
Creativity and Critical Thought	Is the project carefully thought out? Are the limitations carefully considered? Does it appear that time and effort went into the planning and implementation of the project?	0 - 1
Slides	Are the slides well organized, readable, not full of text, featuring figures with legible labels, legends, etc.?	0 - 1
Professionalism	How well did the team present? Does the presentation appear to be well practiced? Are they reading off a script? Did everyone get a chance to say something meaningful about the project?	0 - 1
Comments	*Optional.*	

Rubric for the final project: Within team peer evaluation

- Your estimate of how much each team member has contributed, including yourself. Estimate the percentage of the total amount of work/effort done by each member, including. Be sure your percentages sum to 100%! This information will not be shared with your team members.
- Comments and feedback for the team. These comments will be distributed to all team members anonymously, i.e., team members will receive all comments for the team but not see who wrote which comment. Both positive and constructive comments are welcomed. Please make sure that the tone is appropriate for distribution.
- Reflect on your own team contribution, unpacking the contribution score you gave yourself. What have you been doing well on, and what are some areas of improvement, and how do you plan to address them? This information will not be shared with your team members.
- Comments about team dynamics. This information will not be shared with your team members.
- Based on all your answers above, rate each of your teammates' scores, except for your own, on a scale of 0 to 10. This information will not be shared with your team members.
    - Each student receives an average of their teammates' scores. Any outlying scores will be examined in light of verbal feedback and will be considered in the calculation if there is verbal feedback to support it as well as an agreement among team members (e.g., if you give a teammate a score of 1 out of 10 below, your verbal feedback above should explain why).
    - Failure to fill out the within team peer evaluation form, including this question, will result in a 0 for the within team peer evaluation portion of an individual's project score. This is meant as an incentive to complete the peer evaluation rather than a prohibitive measure.

APPENDIX B - GITHUB REPOSITORY TEMPLATE

A repository template can be found at https://github.com/mine-cetinkaya-rundel/ds-final-project. The webpage for this repository is at https://mine-cetinkaya-rundel.github.io/ds-final-project

B.1. WHAT'S IN THE REPOSITORY

```
project-repo/
|-- .github/                  Folder for issue templates & workflows
|   |-- ISSUE_TEMPLATE/         Issue templates
|   |-- workflows/              GitHub Actions workflows
|-- data/                     Folder for data
|   |-- README.md               Data dictionary template
|-- extra/                    Folder for extra materials (not graded)
|   |-- README.md               Information on what goes in this folder
|-- presentation/             Folder for presentation
|   |-- presentation.Rmd        xaringan presentation template
|-- proposal/                 Folder for proposal
|   |-- proposal.Rmd            Rmd template for proposal
|-- .gitignore
|-- README.Rmd                Rmd template for summary
|-- README.md                 md output for summary
|-- _config.yml               Theme setting for project webpage
|-- project.Rproj             RStudio project file
```

B.2. CONTENTS OF REPOSITORY `README.RMD`

```
---
title: Project title
author: by Team name
output: github_document
---

## Summary

Write-up of your project and findings go here. Think of this as the text of your pres
entation. The length should be roughly 5 minutes when read out loud. Although pacing
varies, a 5-minute speech is roughly 750 words. The addin will ignore code chunks and
only count the words in prose. You can also load your data here and present any analy
sis results / plots, but I strongly urge you to keep that to a minimum (maybe only th
e most important graphic, if you have one you can choose). And make sure to hide your
code with `echo = FALSE` unless the point you are trying to make is about the code it
self. Your results with proper output and graphics go in your presentation, this spac
e is for a brief summary of your project.

## Presentation

The slides for our presentation can be found [here](presentation/presentation.html).

<!-- Keep only if you have recorded a video of your presentation. -->
The video recording of our presentation can be found [here](INSERT LINK TO VIDEO).
<!-- -->

## Data
```

```
Include a citation for your data here. If you found your data off the web, make sure
to note the retrieval date.

## References (optional)

List any references here. If you don't have any references to list, you can remove th
is section.
```

B.3. CONTENTS OF `README.MD` IN THE `DATA` FOLDER

```
# data

Place data file(s) in this folder.
Then, include codebooks (variables, and their descriptions) for your data file(s) usi
ng the following format.

## name of data file
- `variable1`: Description of variable 1
- `variable2`: Description of variable 2
- `variable3`: Description of variable 3
- ...
```

B.4. CONTENTS OF `README.MD` IN THE `EXTRA` FOLDER

```
Any extra documents you might have go here. This might include Rmd files you're using
to develop your project, any notes, or anything else. The contents of this folder wil
l **not** be graded, it's just a convenient place to store documents and collaborate
with teammates without cluttering the rest of your repo.
```