The Pragmatics of Pedagogical Datasets: Methods and Results

Austin Cory Bart
Virginia Tech
Blacksburg, Virginia
acbart@vt.edu

Dennis Kafura
Virginia Tech
Blacksburg, Virginia
kafura@vt.edu

Clifford A. Shaffer
Virginia Tech
Blacksburg, Virginia
shaffer@vt.edu

Eli Tilevich
Virginia Tech
Blacksburg, Virginia
tilevich@vt.edu

ABSTRACT

Lorem ipsum dolor sit amet consectetur adipiscing elit, ante sodales morbi torquent vehicula lobortis convallis erat, fusce viverra diam condimentum vivamus cras. Eget euismod purus aptent netus porta dictum risus dui ullamcorper magnis, arcu integer per natoque litora vulputate hendrerit dictumst rhoncus. Fringilla neque penatibus vivamus ridiculus nec blandit congue lobortis, mattis eros tellus tortor facilisi inceptos aenean, etiam feugiat velit dictumst est ad sociis. Mollis natoque montes mauris etiam fames suscipit condimentum augue luctus ut, orci torquent fermentum euismod curae ridiculus ornare massa aptent. Aliquet maecenas himenaeos consequat est posuere parturient eu et commodo facilisis phasellus, ac sollicitudin quam fusce felis tellus turpis risus aenean. Blandit penatibus turpis ridiculus platea libero sociis risus vitae inceptos dui, pulvinar cras class suscipit dignissim phasellus et quisque nostra convallis, non est dapibus laoreet ante vivamus parturient a mollis.

KEYWORDS

Datasets, Data Science

ACM Reference format:
Austin Cory Bart, Dennis Kafura, Clifford A. Shaffer, and Eli Tilevich. 2018. The Pragmatics of Pedagogical Datasets: Methods and Results. In *Proceedings of ACM SIGCSE, Baltimore, Maryland USA, February 2018 (SIGCSE'18),* 7 pages.
https://doi.org/10.475/123_4

1 INTRODUCTION

Data science, the scientific use of data sets to explore and answer real-world questions, is a compelling context for introductory computing experiences. The techniques and methods have applicability across a wide array of problems, and presents exciting opportunities for advancing disciplines even outside of computing. Non-computing majors in introductory courses can be paired with data relevant to their long-term career interests, improving the usefulness and relevancy of the learning experience. Further, many of the techniques used in processing datasets align closely with introductory computing course content.

However, data science can be extremely challenging to integrate into courses. First, appropriate datasets, of sufficient size and quality, must be found. Next, these datasets need to be cleaned and organized into a form suitable for novice learners. Finally, these datasets need to be discovered and understood by the beginners in a timely fashion. All of these activities have constraints and complications associated with them.

To ameliorate this problem, we introduce a new guide, "The Pragmatics of Pedagogical Dataset Development". The Pragmatics is an open-sourced, web-based document (freely available at https://goo.gl/CtD8ax). The goal of the document is to help developers create "Pedagogical Datasets" – datasets specifically targeted at novices for learning purposes. The `Pragmatics` are an organized collection of design issues, affordances, considerations, suggestions, and the authors' experiences in creating a large repository of Pedagogical Datasets.

In addition to this guide, we present the results of a study that explored introductory learners experience with pedagogical datasets. This study is a ~~repetition~~ of the study described by Bart et al [4], with a larger body of participants. Although results are largely similar, we highlight some nuances compared to the previous study.

This paper makes the following two major contributions:

- A brief review of data science in computing education,
- A description of the Pragmatics text, and
- Further validation of the efficacy of data science as an introductory context.

The audience of this paper are instructors and curriculum developers who are interested in preparing pedagogical datasets or using datasets in their courses. These developers might be planning to submit to open repositories, or develop datasets specifically for their own students.

2 DATA SCIENCE

In the past two decades, the field of Data Science has emerged as a popular area at the intersection of computer science, statistics, mathematics, and a number of other fields [8]. Data Science is the process of answering questions by building, exploring, and processing datasets. There are many theoretical models that define

SIGCSE'18, February 2018, Baltimore, Maryland USA

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

the term more strictly, but in general it can be described as an iterative model of collecting, sanitizing, processing, rendering, and interpreting information. There are many overlaps between Data Science and topics typically covered in computing curricula.

Data Science can be used as both *content* or *context* [7]. As a content area, specific learning objectives are included in the target learning experience that directly connect to core topics in Data Science, such as data processing techniques, using visualization libraries, or understanding statistical tests. As a context, data science becomes a means to an end – the goal is only to motivate students to learn by anchoring the ~~instructional experience~~ in ~~an attractive wrapper.~~ Data science as a context does not mean that students should learn to become data scientists, anymore than it is the goal of projects like Media Computation to teach students how to be professional computational artists [13]. Introducing new contexts incurs pedagogical penalties, requiring time to be spent on material that potentially distracts from core learning objectives. An instructor can downplay the focus on these topics, or emphasize subject matter's strengths (e.g., a statistics major might find it interesting to use their mathematical background to strengthen their problem-solving investigation).

Data Science has been used as both an introductory context and content in a wide range of courses and research studies. Undergraduate curriculum specifically targeted at teaching Data Science have become increasingly popular [1, 10, 16], a number of instructors have worked to make datasets available for individual projects [2, 9, 14], and some researchers have even explored using datasets in non-formal learning experiences such as "Datathons" [3]. Research projects have emerged to provide collections of datasets [4], visualization tools for those datasets [18], and analysis of the efficacy of data science in classrooms [19]. Readers interested in a comprehensive overview of the field of Data Science, including its roles in both industry and classrooms, should refer to Cao's 2017 survey paper [6].

3 PEDAGOGICAL DATASETS

When teaching data science, projects and assignments will typically be structured around the use of datasets. These "Pedagogical Datasets" are differentiated from regular datasets in that they are specifically targeted at learners. Such datasets can be scaffolded or organized for a wide range of tasks: to make it easier to process, to highlight a particular class of problems, or to provide an authentic learning experience for the student, for example. Meanwhile, conventional dataset preparation is typically only concerned with processing datasets strictly to obtain meaning and significance for a stakeholder. The difference in design goals necessitates different approaches in building Pedagogical Datasets.

3.1 Prior Work

There has been little prior work in the literature on practical design and development of Pedagogical Datasets. The UCI Machine Learning Repository aimed to collect and organize a plethora of pedagogical datasets suitable for introductory Machine Learning topics [15] – although an impressive collection, the materials are targeted for specific tasks in advanced computing topics. The STARS project collected real-world datasets available for introductory statistics courses [5]. Although both projects created repositories of datasets, both seem to have stopped production some time ago.

More crucially, ~~neither project attempted to research~~ the formal process of developing pedagogical datasets, to create enduring lessons for future developers who wished to pick up where these projects left off. Radinksy et al. poses some design principles for integrating datasets into inquiry-based learning curriculum, which includes a call for cultivating data but stops short of best practices for organizing and disseminating said data [17]. Verbert describes organizational principles for pedagogical datasets for educational data, specifically for Learning and Knowledge Analytics [20], but not data meant directly for student ~~consumption.~~ None of the formerly described projects codifies best practices of preparing Pedagogical Datasets.

The design and development of datasets is, of course, a fundamental topic in Data Science. Entire textbooks have been dedicated to the topic, and introductory curriculum spend significant amounts of time covering subproblems in this field. Although Pedagogical Datasets involve many distinct problems and require specialized expertise, we acknowledge that there is significant overlap with traditional dataset development. Readers interested in learning more general-purpose techniques are recommended to consult Fry 2007 [11]. Although this particular text was discovered by the authors after the creation of the Pragmatics, Fry incorporates many of the same principles in his descriptions of the data science process. Our goal in writing the Pragmatics is to convey enough fundamentals that a novice ~~data scientist~~ would not need to refer to external texts, while focusing on topics that require effort unique to Pedagogical Datasets.

3.2 Design Considerations

The design criteria for pedagogical dataset diverges from the criteria for conventional datasets, although there are significant overlaps. Fundamentally, the goal of datasets is to prepare data for 1) consumption for a general audience, 2) transmission to specific stakeholders, or 3) ~~to be immediately processed~~ into a visualization or summarization. Therefore, the proper design of the datasets is usually aimed towards organizing the form of the dataset for the audience without damaging its contents. However, pedagogical datasets can be distinctive in fulfilling the objectives.

The audience for any pedagogical dataset will be learners, ideally at a known level, which means that the designer needs to design for their prior and desired skills. These skills ~~should~~ should be used to determine the appropriate complexity of the *navigability* and *understandability* of the datasets: Is the dataset formatted in such a way that students will know how to navigate it? Do students understand the data without needing to rely heavily on the documentation? In particular, navigation of the dataset should require the desired skills, but should otherwise not involve any skills not identified as prior skills. Pedagogical datasets support assignments and activities, which in turn support learning objectives For example, consider a lesson to teach students how to use homogenous collections (e.g., lists), where learners had previously only learned about a single primitive type (e.g., numbers). A dataset that supports this lesson could be composed of that type, but should not

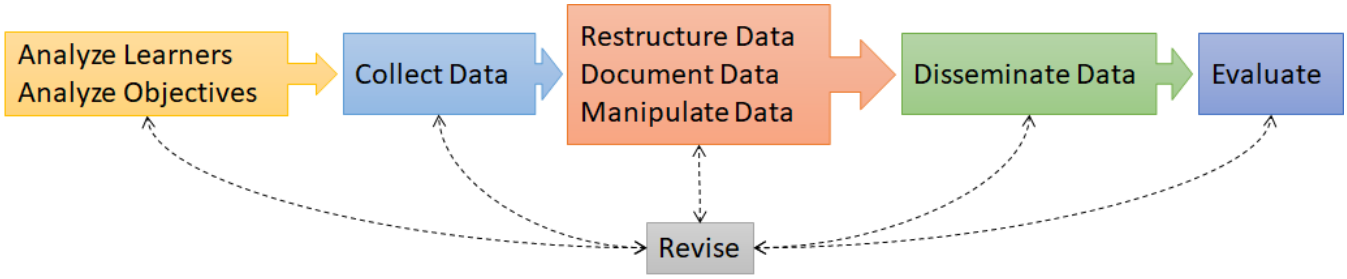

Figure 1: A High-Level Process View of Preparing Pedagogical Datasets

involve heterogeneous collections or other data types, lest they provide distractions.

A distinctive feature of pedagogical datasets is the level of control that the developer has over the narrative of the dataset. A datasets' narrative can be understood as the potential knowledge that can be learned from its contents. This narrative is framed by the context surrounding the provenance of the dataset – how it was collected, who collected it, their purpose in collecting it, and many other important details. A dataset can have multiple narratives, and is partially dependent on the interpretation of its reader. The *authenticity* of the data establishes how accurately it reflects reality. For a conventional dataset, the creator should be a steward of the authenticity of the data, avoiding misrepresenting or modifying the data unduly. However, in a pedagogical dataset, the developer may have freedom to reshape the information to control the narrative more closely. The *flexibility* of the dataset can determine how mutable it is for the developers' needs.

The ultimate goal for a conventional dataset is to be transformed into more consumable knowledge. Typically, this means creating a visualization, a statistical summarization, or some other simpler narrowing of the data. However, in a pedagogical dataset, the developer is intentionally not the agent who finally transforms the data. Although the developer should be informed by the intended approaches that the learner will use, in order to maximize its *manipulability*, they need to focus foremost on the narratives that come out of the dataset.

3.3 Dataset Process

Figure 1 gives a high-level overview of a general process for preparing pedagogical datasets. This process ~~shown~~ is similar to processes for conventional dataset preparation (such as [12]), in particular the middle phases related to collecting and editing the data. First, this process emphasizes the role of the learner and the learning objectives in the design of the dataset; the first phase is to analyze these elements to establish the target audience and complexity of the dataset. Second, the penultimate phase of data dissemination replaces typical final phases that discuss presentation and visualization of data. Finally, the last phase ("Evaluate") is used to show how the generated dataset is meant to be used, observed, and then modified to better suit the needs of the learners, as opposed to a frozen artifact.

4 PRAGMATICS

The Pragmatics text is divided into 6 majors chapters, each of which is subdivided into 5-9 sections. These chapters correspond only approximately to the process outlined before, since some of the advice is not exclusive to a particular phase. In this section, we summarize each chapter, and highlight elements that are particularly unique to pedagogical datasets compared to conventional datasets. Figure 2 gives an outline of the entire text

4.1 General Advice

The Pragmatics begin with a chapter describing advice that either begins the preparation process or cuts across the entire process.

Two pieces of advice in this section ~~ask~~ the reader to consider the learner (as described earlier) and the context. Perhaps more so than in regular dataset preparation, the developer may not be a subject-matter expert with regards to the narratives embedded in the data. If authenticity is an important design criteria, then learning more about the nature of the data becomes crucial.

> "*Find and reach out to subject-matter experts in order to resolve questions. Ensure that the dataset you end up preparing aligns with the kind of data that professionals might expect to see, at least in nature if not shape or format.*"

Much of the advice in this chapter is designed to help scale projects as they grow in complexity or as they extend longitudinally. Although tedious up-front, and possibly unsuitable for smaller scale projects, following these recommendations can pay off later on. Standardizing the development, choosing conventions, and breaking up the process into phases are examples of strategies that can mitigate confusion down the road, and in some cases can have immediate benefit for more intricate projects.

> "*Structure your build pipeline so that phases can be developed in chunks, and individual phases can be debugged independently. For example, by writing out intermediate results to disk, progress can pick up from where it left off.*"

4.2 Collecting Data

The second chapter gives tips and strategies for finding datasets. Typically, data scientists already have a dataset in mind for a particular project. However, instructional developers may be interested in a wide range of pedagogical datasets; rather than finding one for a specific context, they may be searching for one with a particular

set of properties. For example, to satisfy an assignment with learning objectives associated with string processing tools, a text-heavy dataset could be desired. Unfortunately, search tools can still be limited in searching beyond context keywords. This chapter proposes a number of methods for finding data sources. Besides advice on using search engines and reviewing aggregate lists, there are also recommendations on customizing existing data sources. This can be done by scraping web sites, synthesizing existing data sources, or even through mining a real-time data source:

> "*Mining a real-time data source is an easy way to translate a high-velocity, low-volume dataset into a low-velocity, high-volume dataset. The idea is to take a data source that updates regularly and to retrieve data from it on a consistent schedule, aggregating the data over time.*"

4.3 Restructuring Data

When a learner encounters a pedagogical dataset, their first observations will be colored by the structure of the dataset. This chapter covers several strategies for organizing data to minimize students' cognitive load and make it easier for them to comprehend the structure of the data, and pitfalls to avoid when making design trade-offs. Although much of the advice would benefit the creation of any dataset, the advice becomes more urgent for pedagogical dataset design because of the limited abilities of the learners.

> "*When working with spreadsheets that are particularly wide (i.e., have a lot of columns), chunking columns can be an excellent way to help users navigate the data's structure. These chunks of columns can represent sub-abstractions that group related fields. For example, an address can be grouped with latitude and longitude fields under a "location" field.*"

4.4 Manipulating Data

This section contains a number of recommendations to encourage uniformity, simplicity, and readability in the data. Unless the learning objectives specifically require non-uniform data, every diverging element should be cleaned, lest they distract the learner from the intended lesson.

> " *Be as consistent as possible across fields names, types, and their values. Ensure that every field name has the same style of capitalization, spelling, use of symbols, and punctuation, and make sure there are no errors in any of the above. Make sure that every instance of a field has the same type and uses the same kinds of units.*"

This section also covers techniques to change the shape of the data, ideally without changing its nature. These techniques can include imputation (smoothing out missing values), mathematical transformations, and even cleaning up bad data via semi-automatic methods.

4.5 Working with Data Types

~~An entire~~ chapter is devoted to the pedagogical idiosyncrasies of working with common data types and formats. There can be surprising nuances to different kinds of data. For instance, representing dates and times so that beginners can use them is ~~tricky:~~ is a string representation sufficient? What about as a ~~u~~nix epoch timestamp? Although the data can itself have a natural representation, this isn't necessarily ideal either:

> " *When designing for a specific audience, it can be helpful to use a measurement format they are already comfortable with. Inversely, putting data into an awkward format can provide an opportunity to have students practice manipulating data. In other situations, the context may demand or benefit from a certain format (e.g., using metric for scientific data, or sortable dates).*"

4.6 Knowing Data

The final chapter, despite being the shortest, includes a number of important points for considering students' encounters with pedagogical datasets. For example, how will students retrieve and store the necessary data? How will learners be expected to learn about the structure of the data? What documentation needs to be available to supplement the dataset? This last question is particularly ~~tricky,~~ since many novice learners seem to be predisposed to not read the documentation, no matter how conveniently available.

> " *Make sure that the documentation is always readily available, easy to access, and able to answer whatever questions they have. The fewer the barriers, and the better the documentation is at answering questions, the more students will learn to use it and build confidence in documentation as a concept. One method for improving documentation is to log what questions students typically ask about a dataset, and using that data to improve the existing documentation.*"

5 DATASETS IN THE CLASSROOM

As part of our exploration of the efficacy of using pedagogical datasets in the classroom, we have replicated an experiment by Bart et al [4]. The previous study was relatively small, with only 50 participants, but found a number of interesting results. First, the researchers

5.1 Methodology

The replication study was conducted over two semesters of an "Introduction to Computational Thinking" course for non-computing majors, representing a similar population to the original study. A survey was administered at the end of each semester to all students. There were 191 students total, and 176 gave consent for their responses to be used for research purposes (a 92.1% response rate). Student demographic data was only partially retained, but the gender ratio was roughly 60% female. The students skewed towards freshmen and sophomores (roughly 30% each) and the remainder roughly split between juniors and seniors (roughly 20% each).

1. General Advice	1.1. Have a plan 1.2. Build for your audience 1.3. Iterate 1.4. Standardize your process 1.5. Keep a clean workspace 1.6. Manage dataset health 1.7. Beware breaking convention 1.8. Work in phases 1.9. Understand the context
2. Collecting Data	2.1. Hunting sources 2.2. Working with file formats 2.3. Scraping web data 2.4. Mining real-time data 2.5. Legality of your data 2.6. Synthesizing datasets
3. Restructuring Data	3.1. Choose your target structure 3.2. Layering columnar data 3.3. Converting XML to JSON 3.4. Working with indexes 3.5. Collapsing fields 3.6. Stacking data 3.7. Redundant total field
4. Manipulating Data	4.1. Standardize fields 4.2. Names are important 4.3. Working with bad data 4.4. Cleaning up by hand 4.5. Reshaping data 4.6. Extending a dataset
5. Working with Data Types	5.1. Numbers 5.2. Textual 5.3. Dates and times 5.4. Measurements 5.5. Locations 5.6. URLs 5.7 Enumerated data
6. Knowing Data	6.1. No one reads documentation 6.2. Learning the structure 6.3. Learning the distribution 6.4. Disseminating materials 6.5. Monitor usage

Figure 2: Outline of the Pragmatics

The survey instrument used was almost identical to the previous study's instrument. This instrument included a number of questions unrelated to the questions' considered in this study, but were included in order to match the previous research protocol. The relevant questions were divided into three sections. The first section (novel to our ~~instantiation of the~~ survey) described seven potential introductory computing contexts and asked students how much they would prefer or avoid each one:

- "Working with data sets related to your major"
- "Working with pictures, sounds, movies"
- "Making games and animations"
- "Making websites"
- "Making scientific models of real-world phenomenon"
- "Controlling robots or drones"
- "Making phone apps"

The second section asked students how likely they were to continue to learn about computing, apply what they have learned, and to recommend the course to ~~friends;~~ in the previous study, these three subquestions were asked as a single question. The third section was a series of 25 questions organized into 5 groups. Each group corresponded to a component of the MUSIC Model of Academic Motivation, a 5-factor model that hypothesizes that students become motivated when one of the following is present:

(1) feel e**M**powered to direct their own learning,
(2) find the material **U**seful,
(3) believe they will be **S**uccessful,
(4) think that the material is **I**nteresting, and
(5) believe that the course staff **C**ares for them.

Each of these factors was, in turn, connected to one of the 5 core course components: learning about abstraction, learning to write programs, learning about ethics in computation, learning to work with real-world data, and working within a cohort of students (small, 4-person groups). The result of combining these 5 factors and 5 components were statements such as: "I believe it was interesting to learn about abstraction", and "I believe it was useful to my long-term career goals to learn to write computer programs". Students were asked, on a 7-point likert, how much they agreed or disagreed with each statement.

5.2 Results

Figure 3 shows the results of the first question. A one-way ANOVA test reveals that student preference for the Data Science context is significantly higher than any other context. Of course, there is little actual difference in student preference for some of the contexts: particularly, Media Computation was only slightly less preferred. Also notable is students increased neutrality on the other contexts compared to Data Science – perhaps symptomatic of the ambiguity of asking students to consider contexts that they have not personally encountered.

Most of our results for the second and third survey question sections, asking students to agree or disagree with the statements regarding motivation, ~~compared to~~ the results found in the prior study. To demonstrate this, Figure 4 shows students' perception of the usefulness for each course component. This data shows that a significantly higher number of students felt that working with data was more useful to the long-term career goals than learning to program or work with abstractions. This lends further evidence to a major, original hypothesis from that study: that students find a data science context more motivating than the core course content.

Another major result from the original study was the identification of a strong correlation between students' intent to continue learning computing and their attitudes towards learning to program and other core course components (but, notably, not with their attitudes towards the context). Figure 5 presents data that diverges somewhat from the previous study. This table shows the Pearson correlation between students' stated intent to continue

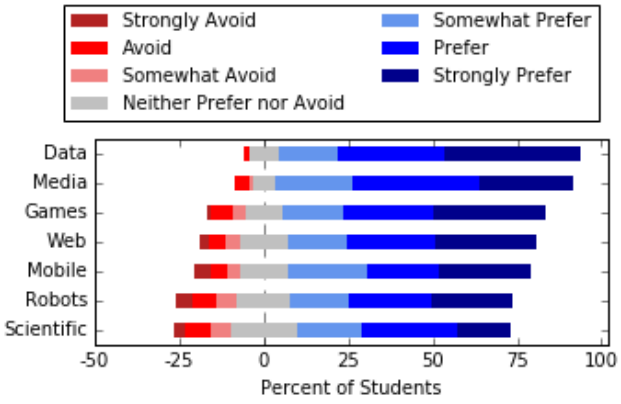

Figure 3: Student Preference for Introductory Contexts

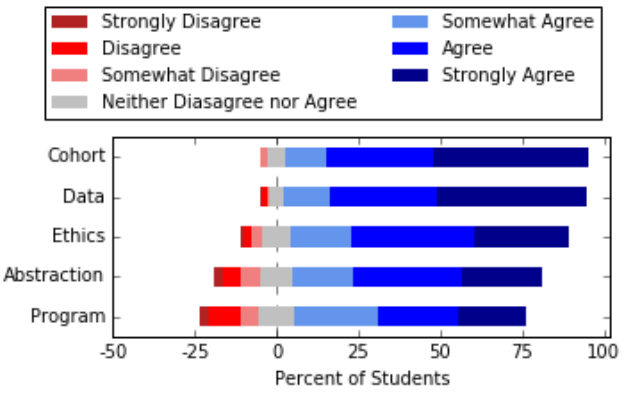

Figure 4: Student Perceptions of Usefulness of Course Components

learning computing, and their self-reported motivation towards each component of the course. The previously high correlations are now considerably lower, although many of the pairwise statements still reach significance. Previously, the strongest correlations were found in students' sense of usefulness of learning to program and work with abstractions. In this study, however, we now find that usefulness is roughly on par with students' self-efficacy and interest – at least in terms of learning to program. However, we can reiterate the interpretation of this result from the previous study: this does not demonstrate that data science is not a motivating context for learners in a course, but instead that, when the goal is to convince students to learn more about computing, a data science context is less important than learning to program.

5.3 Threats to Validity

Although we have improved on the previous study, there are still some threats to validity present in our work. The largest threat in this study is the confirmation bias present. The students surveyed had already completed an introductory computing course contextualized with Data Science; their success in the course might suggest

Component	M	U	S	I	C
Abstraction	**0.198***	**0.230***	**0.174****	0.131	**0.274***
Cohort	0.003	0.114	-0.032	0.024	0.115
Data	-0.005	0.057	-0.014	0.110	0.147
Ethics	0.052	0.146	0.029	0.050	**0.207***
Programs	**0.281***	**0.397***	**0.363***	**0.434***	**0.252***

Figure 5: Correlation between Students' Intent to Continue Learning Computing Vs. Components of the Course with Respect to Motivational Components (at End of Semester)

a predisposition to the context. It is possible that surveying a more general population might reveal that a different course context is more widely preferred.

Statistically, using tests such as the Pearson Correlation on non-continuous data, such as that collected from a Likert, ~~is a somewhat risky proposition~~. We believe that our large survey population and the granularity of the survey instrument make the statistical tests more robust.

6 CONCLUSION

In this paper, we have argued for the continued importance of data science as an introductory context for novice learners. To support the growth of this context, we have authored, and presented here, a guide for educational developers to create new pedagogical datasets. This guide covers strategies, tips, pitfalls, and examples from the authors' own experiences. In addition to presenting the guide, this paper also presents the results of a replication study to evaluate the efficacy of data science as a motivating context. With four times the number of participants as the original study, this replication study largely confirms the original study, albeit with nuances. In conclusion, we feel that this paper demonstrates the rising significance of the power of data science.

REFERENCES

[1] Paul Anderson, James Bowring, Renée McCauley, George Pothering, and Christopher Starr. An Undergraduate Degree in Data Science: Curriculum and a Decade of Implementation Experience. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education.*

[2] Ruth E. Anderson, Michael D. Ernst, Robert Ordóñez, Paul Pham, and Steven A. Wolfman. 2014. Introductory Programming Meets the Real World: Using Real Problems and Data in CS1. In *Proceedings of the 45th ACM Technical Symposium on Computer Science Education (SIGCSE '14).* 465–466. https://doi.org/10.1145/2538862.2538994

[3] Craig Anslow, John Brosz, Frank Maurer, and Mike Boyes. 2016. Datathons: An Experience Report of Data Hackathons for Data Science Education. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education.* ACM, 615–620.

[4] A. C. Bart, R. Whitcomb, E. Tilevich, C. A. Shaffer, and D. Kafura. 2017. Computing with CORGIS: Diverse, Real-world Datasets for Introductory Computing. In *Proceedings of the 48th ACM Technical Symposium on Computer Science Education (SIGCSE '17).*

[5] Penelope Bidgood. 2006. Creating statistical resources from real datasets–the STARS project. In *Seventh International Conference on Teaching Statistics (ICOTS7), Salvador, Bahia, Brazil.*

[6] Longbing Cao. 2017. Data science: a comprehensive overview. *ACM Computing Surveys (CSUR)* 50, 3 (2017), 43.

[7] Jeong-Im Choi and Michael Hannafin. 1995. Situated cognition and learning environments: Roles, structures, and implications for design. *Educational Technology Research and Development* 43, 2 (1995), 53–69. https://doi.org/10.1007/BF02300472

[8] Thomas H Davenport and DJ Patil. 2012. Data Scientist: The Sexiest Job of the 21st Century-A new breed of professional holds the key to capitalizing on big data opportunities. But these specialists aren't easy to findâĂŤAnd the competition for them is fierce. *Harvard Business Review* (2012), 70.

[9] Peter DePasquale. 2006. Exploiting On-line Data Sources As the Basis of Programming Projects. In *Proceedings of the 37th SIGCSE Technical Symposium on Computer Science Education (SIGCSE '06)*. 283–287. https://doi.org/10.1145/1121341.1121430

[10] Roland DePratti, Garrett M Dancik, Fred Lucci, and Russell D Sampson. 2017. Development of an introductory big data programming and concepts course. *Journal of Computing Sciences in Colleges* 32, 6 (2017), 175–182.

[11] Ben Fry. 2007. *Visualizing data: Exploring and explaining data with the processing environment.* " O'Reilly Media, Inc.".

[12] P Guo. 2013. Data science workflow: Overview and challenges. *blog@ CACM, Communications of the ACM* (2013).

[13] Mark Guzdial and Allison Elliott Tew. 2006. Imagineering inauthentic legitimate peripheral participation: an instructional design approach for motivating computing education. In *Proceedings of the second international workshop on Computing education research*. 51–58.

[14] Olaf A. Hall-Holt and Kevin R. Sanft. 2015. Statistics-infused Introduction to Computer Science. In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15)*. 138–143. https://doi.org/10.1145/2676723.2677218

[15] M. Lichman. 2013. UCI Machine Learning Repository. (2013). http://archive.ics.uci.edu/ml

[16] Aparna Mahadev and Karl R Wurst. 2015. Developing concentrations in big data analytics and software development at a small liberal arts university. *Journal of Computing Sciences in Colleges* 30, 3 (2015), 92–98.

[17] Josh Radinsky, Ben Loh, Jennifer Mundt, Sue Marshall, Louis M Gomez, Brian J Reiser, and Daniel C Edelson. 1999. Problematizing Complex Datasets for Students: Design Principles for Inquiry Curriculum. (1999).

[18] Kalpathi Subramanian, Jamie Payton, David Burlinson, and Mihai Mehedint. 2016. Bringing Real-World Data And Visualizations Into Data Structures Courses Using BRIDGES. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*. 590–590.

[19] David G. Sullivan. 2013. A Data-centric Introduction to Computer Science for Non-majors. In *Proceeding of the 44th ACM Technical Symposium on Computer Science Education (SIGCSE '13)*. 71–76. https://doi.org/10.1145/2445196.2445222

[20] Katrien Verbert, Nikos Manouselis, Hendrik Drachsler, and Erik Duval. 2012. Dataset-driven research to support learning and knowledge analytics. *Educational Technology & Society* 15, 3 (2012), 133–148.