DATA SCIENCE EDUCATION K-12

INDIVIDUAL PAPER: A task design framework for introducing code-driven tools through statistical modelling

Anna Fergusson, University of Auckland, a.fergusson@auckland.ac.nz
Maxine Pfannkuch, University of Auckland, m.pfannkuch@auckland.ac.nz

Session Description

Computer programming has been promoted in teaching and learning materials to support data science at the high school level. However, minimal research exists about the design of tasks that support the introduction of students and teachers to the use of code-driven tools for statistical modelling. Using a design-based research approach, four structured tasks were developed for teaching statistical modelling at the same time as introducing the programming language R. These tasks were implemented with high school statistics teachers and a task design framework for introducing code-driven tools was produced by using retrospective analysis on the four tasks to identify, evaluate, and refine key design principles and processes. The task design framework developed from this research explicates important features of the tasks used in the research, including extending the familiar into the unfamiliar, using the informal before the formal, and carefully targeting, sequencing and connecting specific human-computer interactions for statistical modelling.

Proposal

The advent of data science has led to statistics education researchers re-thinking their ideas about tasks and tools for teaching and learning. A common thread to discussions about data science education is that students need to integrate both statistical and computational thinking to learn from data (e.g., De Veaux et al., 2017), which necessitates students developing at least some coding (computer programming) skills (e.g., Gould, 2010). Using code-driven tools facilitates the use of digital data sources and algorithmic modelling, however, careful considerations are needed to ensure that the development of statistical concepts, thinking, and reasoning are still well supported when learners use such tools. The use of tools is paired with tasks, and the task used will influence the nature of the learning that takes place (Doerr & Pratt, 2008). Within statistics education, research focused on task design has not specifically explored computer programming as the main computational tool (e.g., Ben-Zvi et al., 2017). Hence, the purpose of this research was to explicate design principles for the construction of statistical modelling tasks that introduce code-driven tools by developing a task design framework. The research questions is: *What design principles could guide the construction of statistical modelling tasks that introduce code-driven tools?*

Research approach

From a pragmatic interpretive perspective (e.g., Creswell & Poth, 2016), an exploratory study was conducted over five years using a design-based research approach (e.g., Bakker & van Eerde, 2014). The analysis of practical teaching issues and construction of new learning tasks was informed by both existing design principles and technological innovations (e.g., Edelson, 2002). The participants in the research were twelve teachers from New Zealand (nine female and three male) with experience teaching the national statistics curriculum for the last two years of high school. The high school statistics teachers were positioned as learners for the research. Four new tasks were constructed (e.g., Fergusson & Pfannkuch, 2022a; Fergusson & Pfannkuch, 2022b; Fergusson & Pfannkuch, 2024) that were aligned to the national curriculum for statistics at the Grade 11 or 12 level. The curriculum focus was statistical modelling from a data science perspective. The tasks extended the following curriculum topics: numeric data distributions, probability simulations, simple linear regression, randomisation (permutation) tests. All the tasks introduced the programming language R (R Core Team, 2020). Retrospective analysis was used on the four tasks to identify, evaluate, and refine key design principles and processes, and to develop the Introducing Code-Driven Tools through Statistical Modelling (ICDTSM) task design framework.

The ICDTSM task design framework

The ICDTSM task design framework is summarised in Figure 1.

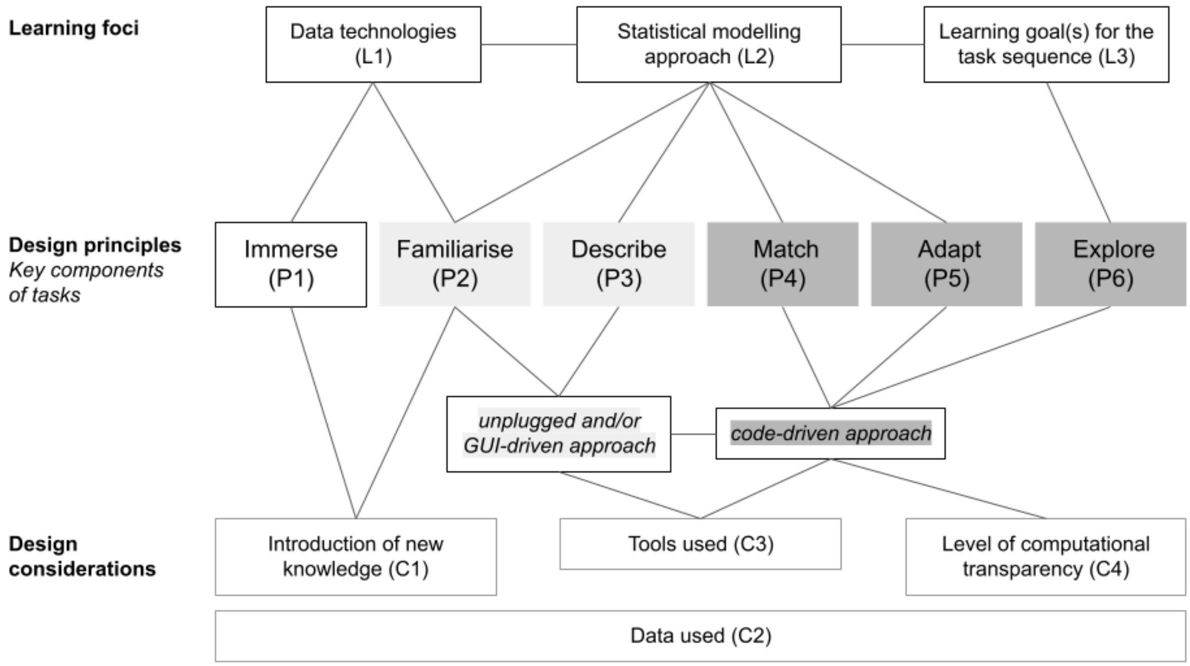

Figure 1: The ICDTSM task design framework

The three core aspects of the ICDTSM task design framework are: learning foci (Table 1), design principles (Table 2), and design considerations (Table 3). To use the design framework to construct a task sequence that introduces a code-driven tool for statistical modelling, the task designer needs to decide what data technologies and statistical modelling approach will be used and determine the learning goal(s) (Table 1).

Table 1: Learning foci for the ICDTSM task design framework

Learning focus	**Task designer action**
Data technologies (L1)	Select data contexts that involve data technologies
Statistical modelling approach (L2)	Identify key statistical modelling actions, representations, and associated language
Learning goal(s) for task sequence (L3)	Set learning goals that require students to create computational products

These learning foci (L1 to L3) are reviewed and refined during the design of the task sequence and finalised at the end of the construction process. The design principles (P1 to P6) are used to inform decisions about features of the learning task sequence in terms of what learners will be asked or encouraged to do, and the chronological order of these actions or experiences (Table 2).

Table 2: Design principles for the ICDTSM task design framework

Design principle	**Task design features (what students are asked to do)**
Immerse in data (P1)	Participate in activities that promote engagement with the data context and data technologies
Familiarise with key statistical modelling actions (P2)	Carry out statistical modelling activities without using code
Describe computational aspects of statistical modelling process (P3)	Use words to describe key computational aspects of statistical modelling actions
Match statistical modelling actions to code chunks (P4)	Read and match lines/chunks of code with statistical modelling actions
Adapt code chunks with slight modifications (P5)	Identify aspects of code to change, in order to carry out statistical modelling actions

Explore “what if?” changes to code (P6)	Modify at least one aspect of provided code to produce new or unexpected outputs

Task sequences can be designed using separate phases for each design principle, or several design principles may be used within the same phase. Alongside the design principles, the construction of the task sequence is simultaneously guided by four design considerations (C1 to C4) that inform broader decisions (Table 3).

Table 3: Design considerations for the ICDTSM task design framework

Design consideration	**Task designer action**
Introduction of new knowledge (C1)	Consider when and how much new knowledge is being introduced within each phase of the task sequence
Data used (C2)	Use one general data context and either one data source that provides different variables or subsets, or closely related data sources
Tools used (C3)	Connect actions and representations when learners move between different computational tools
Level of computational transparency (C4)	Select or develop features of computational tools by considering how obvious the computations performed by the tool are to learners

With respect to computational tools, the immerse (P1) component of the task sequence can utilise any tool, the familiarise (P2) and describe (P3) components should utilise unplugged or GUI-driven tools, and the match (P4), adapt (P5), and explore (P6) components should utilise code-driven tools.

Discussion

Two core threads of the ICDTSM task design framework (Figure 1) are: (1) extend the familiar into the unfamiliar (e.g., Biehler & Schulte, 2017), by using learners’ familiarity with statistical modelling approaches as a basis for introducing new contextual and practical computational ideas and; (2) use the informal before the formal (e.g., Gravemeijer, 2004), for example, when introducing algorithmic models or coding approaches. The use of data technology contexts appeared to stimulate thinking about new computational approaches (cf. Hicks & Irizarry, 2018), and structuring a task sequence towards an end goal of creating a statistical model as a computational product appeared to help teachers appreciate both the purpose and utility of their learning (cf. Ainley et al., 2006). The ICDTSM task design framework also explicates the need carefully target, sequence and connect specific human-computer interactions for statistical modelling, including: creating adaptable code chunks that represent each modelling action; identifying the modelling actions or steps that could transfer across computational tools; and connecting each modelling action across the different computational tools used in the task sequence, both visually and textually. This research contributes to a greater understanding of how data science tasks could be designed to support learners to simultaneously develop conceptual-based and tool-based understanding (cf. Artigue, 2002).

References

Ainley, J., Pratt, D., & Hansen, A. (2006). Connecting engagement and focus in pedagogic task design. *British Educational Research Journal*, 32(1), 23–38. https://doi.org/10.1080/01411920500401971

Artigue, M. (2002). Learning mathematics in a CAS environment: The genesis of a reflection about instrumentation and the dialectics between technical and conceptual work. *International Journal of Computers for Mathematical Learning*, 7(3), 245–274. https://doi.org/10.1023/A:1022103903080

Bakker, A., & van Eerde, D. (2014). An introduction to design-based research with an example from statistics education. In A. Bikner-Ahsbahs, C. Knipping, & N. Presmeg (Eds.), *Approaches to qualitative research in mathematics education* (pp. 429–466). Springer. https://doi.org/10.1007/978-94-017-9181-6_16

Ben-Zvi, D., Gravemeijer, K., & Ainley, J. (2017). Design of statistics learning environments. In D. Ben-Zvi, K. Makar, & J. Garfield (Eds.), *International handbook of research in statistics education* (pp. 473–502). Springer. https://doi.org/10.1007/978-3-319-66195-7_16

Biehler, R., & Schulte, C. (2017). Perspectives for an interdisciplinary data science curriculum at German secondary schools. In R. Biehler, L. Budde, D. Frischemeier, B. Heinemann, S. Podworny, C. Schulte, & T. Wassong (Eds.), *Paderborn Symposium on Data Science Education at School Level 2017: The Collected Extended Abstracts* (pp. 2–14). Universitätsbibliothek Paderborn. http://doi.org/10.17619/UNIPB/1-374

Creswell, J. W., & Poth, C. N. (2016). *Qualitative inquiry and research design: Choosing among five approaches.* SAGE.

De Veaux, R. D., Agarwal, M., Averett, M., Baumer, B. S., Bray, A., Bressoud, T. C., Bryant, L., Cheng, L. Z., Francis, A., Gould, R., Kim, A. Y., Kretchmar, M., Lu, Q., Moskol, A., Nolan, D., Pelayo, R., Raleigh, S., Sethi, R. J., Sondjaja, M., … Ye, P. (2017). Curriculum guidelines for undergraduate programs in data science. *Annual Review of Statistics and Its Application,* 4, 15–30. https://doi.org/10.1146/annurev-statistics-060116-053930

Doerr, H. M., & Pratt, D. (2008). The learning of mathematics and mathematical modeling. In M. K. Heid & G. W. Blume (Eds.), *Research on technology and the teaching and learning of mathematics: Volume 1 Research syntheses* (pp. 259–285). Information Age Publishing.

Edelson, D. C. (2002). Design research: What we learn when we engage in design. *The Journal of the Learning Sciences*, 11(1), 105–121. https://doi.org/10.1207/S15327809JLS1101_4

Fergusson, A., & Pfannkuch, M. (2022a). Introducing high school statistics teachers to predictive modelling and APIs using code-driven tools. *Statistics Education Research Journal, 21*(2). https://doi.org/10.52041/serj.v21i2.49

Fergusson, A., & Pfannkuch, M. (2022b). Introducing teachers who use GUI-driven tools for the randomization test to code-driven tools. *Mathematical Thinking and Learning, 24*(4), 336-356. https://doi.org/10.1080/10986065.2021.1922856

Fergusson, A., & Pfannkuch, M. (2024). Using grayscale photos to introduce high school statistics teachers to reasoning with digital image data. *Journal of Statistics and Data Science Education*. https://doi.org/10.1080/26939169.2024.2351570

Gravemeijer, K. (2004). Local instruction theories as a means of support for teachers in reform mathematics education. *Mathematical Thinking and Learning*, 6(2), 105–128. https://doi.org/10.1207/s15327833mtl0602_3

Gould, R. (2010). Statistics and the modern student. *International Statistical Review*, 78(2), 297–315. https://doi.org/10.1111/j.1751-5823.2010.00117.x

Hicks, S. C., & Irizarry, R. A. (2018). A guide to teaching data science. *The American Statistician*, 72(4), 382–391. https://doi.org/10.1080/00031305.2017.1356747

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