Carnegie Learning UpGrade / MATHia Digital Learning Platform - Example DataShop Documentation
Stephen E. Fancsali, April Murphy, & Kaleb Mathieu - September 12, 2022
sfancsali@carnegielearning.com

Table of Contents

About This Document

This document illustrates an example of data that are regularly exported from Carnegie Learning's MATHia platform for analysis by learning engineers at Carnegie Learning and often by academic researchers. Column names described in the section DataShop Data Column Descriptions adhere to the format for ingestion and use by Carnegie Mellon University's DataShop repository. This document also includes a brief description of how problem-solving works in the MATHia platform to better ground an understanding of the student learning process that can be captured using these data. The primary role of UpGrade (Ritter et al., 2020) in considering data like those described below is in its specification of experimental designs that randomly assign students to different experimental conditions, often captured in MATHia data exports as "workspace variants" (see "CF (Workspace Variant)" in DataShop Data Column Descriptions), as for example, when an experiment specified in UpGrade tests one or more variants of an instructional experience delivered via MATHia workspaces.

Problem-Solving in MATHia Workspaces

MATHia (formerly Cognitive Tutor; Ritter et al., 2007) content is divided into two types of topical "workspaces,[1]" concept builders and mastery workspaces. Concept builders present a fixed sequence of questions associated with content intended to build conceptual understanding and have no fine-grained knowledge components (KCs) associated with them. Mastery workspaces present multi-step problems on which students demonstrate mastery of KCs. Both types of workspace provide assistance in the form of context-sensitive hints and just-in-time (JIT) feedback. Figure 1 provides a screenshot from a mastery workspace called "Using Scale Factor."
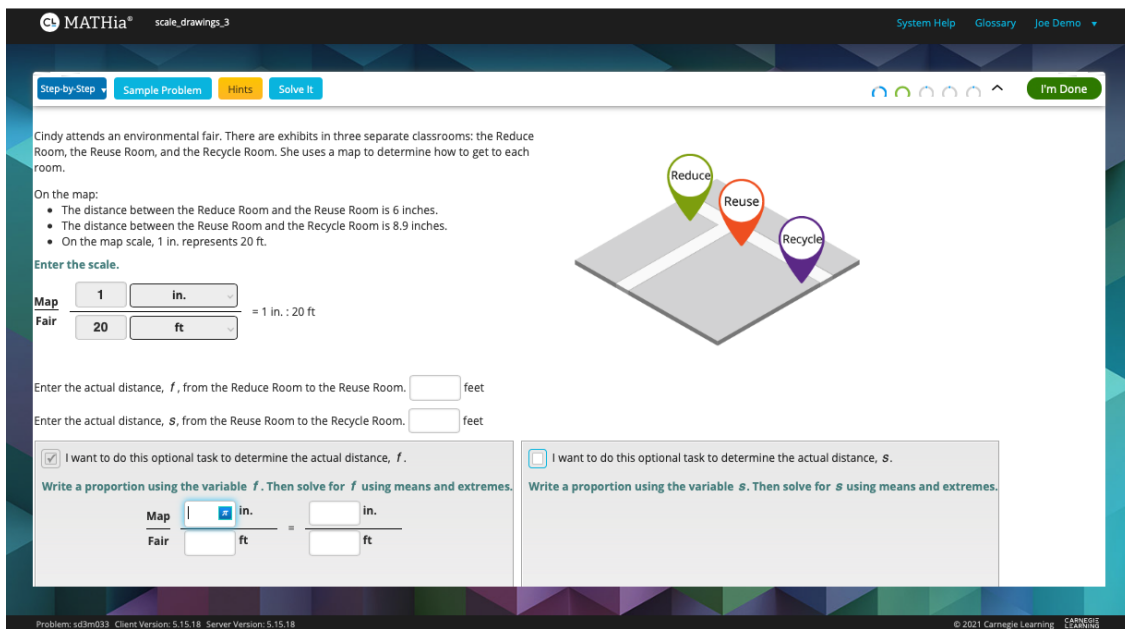

Figure 1. A screenshot of problem-solving in the MATHia workspace "Using Scale Factor."

Each mastery workspace contains a large set of problems, of which each student will complete a subset. When MATHia estimates that students have mastered all KCs associated with a workspace, the student is moved on to the next workspace in their curriculum sequence. KC mastery is estimated using Bayesian Knowledge Tracing (BKT; Corbett & Anderson, 1994). Progress toward skill or KC mastery is

---

[1] Some existing datasets and research use the term "section" or "lesson" for what we call "workspace."

displayed to the student via the Skillometer (see Figure 2). If a student fails to reach mastery of all KCs before reaching a fixed maximum number of problems, they are also moved onto the next workspace, but without mastery (Fancsali et al., 2020). We sometimes call this latter case a "promotion." See Ritter et al., 2016 for details on this approach to mastery learning.

Figure 1 illustrates multi-step problem-solving in a problem within the MATHia mastery workspace "Using Scale Factor." In the partially completed problem in Figure 1, the student has completed four steps of the problem, as entering the "scale" requires providing values and units for both the numerator and denominator. The student has chosen to do an optional task, which introduces four more problem-solving steps that may not always appear in data for this problem (or similar problems within this workspace).
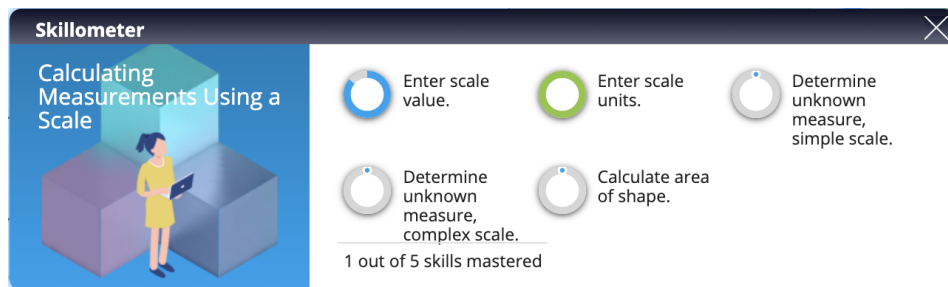

Figure 2. "Skillometer" display for the five KC tracked by "Using Scale Factor." After completing the first four problem steps (see Figure 1), the student is estimated to have mastered the KC "Enter scale units" and has made progress toward mastery of "Enter scale value."

DataShop Data Column Descriptions

Notes: Possible values are often provided in {}; CF = DataShop "Custom Field" associated with transaction-level data.

Anon Student Id: anonymous student identifier

Session Id: {no_session_tracking}; These data do not track login sessions within MATHia, but this is a required field for DataShop, so all values are "no_session_tracking."

Time: Timestamp in UNIX epoch time.

Level (Workspace Id): identifier for the workspace

Problem Name: identifier for the problem

Step Name: identifier for the problem-step

Selection: {Done Button}; This field is typically blank as the selection is usually the "Step Name," but there are entries with the value "Done Button" when the student's action is "Done" (see next column), indicating that they are pressing the "Done Button" in MATHia to proceed to the next problem.

Action: {Attempt, Done, Hint Request, Hint Level Change}
- Attempt: Student made a problem-solving attempt (see input)
- Done: Student clicked the "Done" button required to complete a problem
- Hint Request: Student requests a hint
- Hint Level Change: Student requests a hint at a "deeper" level (see helplevel)

Input: student input at the problem-step (sometimes interpretable a la the actual number/text, etc., input value; other times more difficult to interpret)

Outcome: {OK, BUG, ERROR, INITIAL_HINT, HINT_LEVEL_CHANGE}
- OK = correct
- BUG = error that is tracked for just-in-time (JIT), context-sensitive feedback (e.g., a known misconception, a number as input that appears in the problem but is incorrect, etc.). Some BUG outcomes on first-attempts at a problem-step within a problem do not impact student KC mastery estimates

like an error or JIT typically does. We call these internally "freebie JITs." They are coded as BUG outcomes to comply with DataShop data standards.
- ERROR = error that isn't specifically tracked for JIT feedback
- INITIAL_HINT = first-level hint is provided
- HINT_LEVEL_CHANGE = a "deeper" level hint is provided

Help Level: {0, 1, 2, 3, ...}; Generally (but not always), there are three levels of hints, corresponding to the following (also, in general):
- 0 = not a hint request
- 1 = first-level hint = "goal setting hint" – generally a re-statement of the goal/question relevant to the current problem-step
- 2 = second-level hint = "teach hint" – provides more in-depth insight into how to solve the current problem-step without giving the answer
- 3 = third-level hint = "bottom out hint" – provides the correct answer to the problem-step

Attempt At Step: count of attempts on this step for this problem (or problem-step) (e.g., 1 = first attempt at this problem-step for this problem; 2 = second attempt, etc.; 0 = not an attempt)
- DataShop re-calculates this column during import. See DataShop documentation for datasets that aren't attached as files to the project.

KC Model(MATHia): The skill or knowledge component (KC) tracked by MATHia for this problem-step; When the student's skill values are updated, the change in this BKT estimate is tracked in custom fields CF (Skill Previous p-Known) & CF (Skill New p-Known).
- For student actions on steps associated with multiple KCs that trigger a change in the BKT estimate for these KCs, multiple rows (one per KC) with the same CF (Semantic Event Id) will provide these KCs (and their corresponding changes in BKT p-Known values). However, for instances in which a student action on a step associated with multiple KCs does not lead to a change in

MATHia's BKT estimate, only one of the KCs associated with the step will appear in this column in the row associated with this action.

CF (Ruleid): sub-skill related to KC with some problem-solving context encoded; used, e.g., to determine what hint or just-in-time (JIT) feedback is appropriate; not always interpretable.

CF (Etalon): The correct value for a particular problem-step that MATHia is expecting.

CF (Skill Previous p-Known): BKT skill estimate prior to this action (i.e., semantic event)

CF (Skill New p-Known): BKT skill estimate after this action (i.e., semantic event)

CF (Workspace Progress Status): {GRADUATED, NOT_COMPLETED, PROMOTED}
This is repeated for every entry that corresponds to work in a workspace.
- GRADUATED = The student eventually masters all skills/KCs in this workspace and moves on to the next workspace of content (or finishes the last workspace of assigned content).
- PROMOTED = The student reached the maximum number of problems allowed in this workspace without having mastered all skills/KCs. Student is moved on ("promoted") to the next workspace of content despite a lack of mastery.
- NOT_COMPLETED = The student's work for this workspace is incomplete (i.e., they neither graduated nor were promoted from this section). This could happen for any number of reasons. Given the bevy of factors that might lead a student to not complete a workspace, *it is worth considering filtering out this student work in many analyses.*

CF (Semantic Event Id): a unique id for a semantically meaningful student action/click/input; could have multiple rows associated with it (e.g., when multiple skills are updated for a single student attempt).

CF (Workspace Variant): id of the variant (if any) of the workspace (see Level (Workspace Id) column) in which the student is working. Different workspace variants correspond to different experimental conditions (according to experimental designs specified, for example, by a researcher using UpGrade) and/or variants of workspaces. For example, "introduction_to_a_quadratic_function_control" and "introduction_to_a_quadratic_function_intervention" are variants of the workspace "introduction_to_a_quadratic_function" that correspond to the control and intervention conditions, respectively.

CF (Module): id of the MATHia "module" within which the student was working. Modules group together units, each of which is a set of workspaces. Some module ids are human-interpretable, often corresponding to "shipped" or "suggested" modules grouping together workspaces and units according to Carnegie Learning recommendations for various curricula. Other module ids are not human-interpretable, often corresponding to "custom" modules created by schools and teachers, grouping units and workspaces as they see fit.

CF (Encounter): starting at zero (first encounter), the count of the student's encounter with this workspace over all of their work in MATHia for the school year.

References

Corbett, A.T., Anderson, J.R. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Model User-Adap Inter* 4, 253–278. https://doi.org/10.1007/BF01099821

Fancsali, S.E., Holstein, K., Sandbothe, M., Ritter, S., McLaren, B.M., & Aleven, V. (2020). Towards practical detection of unproductive struggle. In: Bittencourt, I., Cukurova, M., Muldner, K., Luckin, R., Millán, E. (Eds.) *Artificial Intelligence in Education. AIED 2020*. Lecture Notes in Computer Science, Vol. 12164. Springer, Cham. https://doi.org/10.1007/978-3-030-52240-7_17

Ritter, S., Anderson, J.R., Koedinger, K.R., & Corbett, A. (2007). The Cognitive Tutor: Applied research in mathematics education. *Psychonomic Bulletin & Review,* 14(2), 249–255.

Ritter, S., Murphy, A., Fancsali, S. E., Fitkariwala, V., Patel, N., & Lomas, J. D. (2020). UpGrade: An open source tool to support A/B testing in educational software. In *Proceedings of the First Workshop on Educational A/B Testing at Scale*. EdTech Books.

Ritter, S., Yudelson, M., Fancsali, S.E., & Berman, S.R. (2016). How mastery learning works at scale. In: Aleven, V., Haywood, J., Kay, J., Roll, I. (Eds.) *L@S '16: Proceedings of the Third (2016) ACM Conference on Learning @ Scale*. pp. 71–79. https://doi.org/10.1145/2876034.2876039