Teaching and Educational Methods

Assessing Student Learning Using a Digital Grading Platform

Andrew W. Stevens
Mississippi State University

JEL Codes: A20, A12
Keywords: Assessment, digital grading, student learning objectives

Abstract
Effective assessment of student learning is challenging for several reasons. Not only do learning assessments usually crowd out instructional time, but it can be difficult to assess higher-order cognitive aspects of student learning. In this commentary, I present a method for assessing student learning through the use of a digital grading platform that addresses both of these issues. I discuss a case study where this method was implemented and utilized to inform course design, and I argue that digital grading platforms expand instructors' options for student learning assessments.

1 Introduction

As instructors, we pursue parallel goals: on the one hand, we want our students to learn specific course content; on the other hand, we want our students to think about the world in fundamentally new ways. In a microeconomics course, for example, we want our students to learn how to use demand and supply curves to determine a market-equilibrium quantity and price. But more broadly, we want our students to understand more deeply the human behavior underpinning market outcomes: that incentives matter, that decisions are made on the margin, that opportunity costs are more relevant than accounting costs, and so on. Through assignments and exams, it is comparatively easy to assess whether our students have mastered course content. It is more difficult to assess whether students have mastered higher-order aspects of cognitive processing. In this commentary, I outline a method for assessing a wide variety of student learning outcomes through the use of a digital grading platform. Specifically, I present a case study in which this method was used to inform course design. The assessment method requires no additional effort by students, and minimal additional effort by instructors, making it a high-value approach to assessing student learning.

1.1 Standards of Learning

Today, it is fairly standard pedagogical practice for instructors to identify several student learning objectives (SLOs) in their courses. Ideally, instructors share these objectives with their students and use the SLOs to inform learning assessments like assignments and exams (Wiggins and McTighe 2005; Banta et al. 2009). An effective set of SLOs will incorporate a variety of learning functions. To borrow from Bloom's classic taxonomy of educational objectives, these could include knowledge, comprehension, application, analysis, synthesis, and evaluation (Bloom et al. 1956).[1] In short, SLOs outline an instructor's goals for a course, and effective assessment will refer back to these goals.

Beyond individual courses, SLOs can also exist for academic programs or entire institutions. For example, a program of study may specify a set of learning objectives for its students to achieve before graduation. When properly designed, course-level SLOs support and contribute to program- or institution-

[1] Bloom's original taxonomy has shortcomings, and subsequent scholarship has proposed an update that includes "creation" as the highest form of cognition (Anderson and Krathwohl 2001). There are also other approaches to categorizing learning. See, for example: Marzano (2001), Erickson et al. (2006), Fry et al. (2009), Biggs and Tang (2011), and Fink (2013).

level SLOs and vice versa (Suskie 2018, pp. 63–84). Consequently, methods that support course-level learning assessment can make it easier to assess higher-level SLOs as well. This in turn can help administrators satisfy institutional reporting requirements around student learning.

In practice, it can be challenging to assess SLOs—especially those objectives that pertain to higher-order aspects of cognition. One issue is that traditional learning assessments (assignments, tests, student surveys, etc.) require time and effort that can detract from instruction activities. Another issue is that traditional quantitative assessments can be poorly suited to measure students' ability to analyze, synthesize, and evaluate information beyond a particular application. In order to overcome these issues, one approach is to integrate the assessment of student learning into existing course activities in order to make efficient use of students' and instructors' time and effort. For example, an instructor can write exam questions that specifically link particular aspects of course content with particular SLOs. This allows the instructor to grade the exams both for mastery of content and achievement of learning objectives.

Indeed, so-called integrated assessment systems can improve teaching pedagogy (Atwood and Singh 2018) and are increasingly becoming a best practice for the assessment of student learning. Such systems are integrated into a course curriculum, embedded in course content, and economical to implement (Birenbaum et al. 2006). Increasingly, technology is making integrated assessment possible through computer-assisted assessment and similar methods (Brown et al. 1999). Such approaches fit naturally in courses where students or instructors are already making heavy use of technology for instruction, coursework, assignments, or exams (Seden 1999). Nonetheless, there remains a need for new methodologies to integrate computer-assisted assessment with other traditional and technology-based teaching and learning methods (Bull 1999), and to assess the full range of student learning and cognitive development. In this commentary, I address this need by presenting a method of assessing SLOs through the use of a digital grading platform. Tech-savvy instructors who teach large courses can easily implement my proposed method and improve the effectiveness of their existing assessment activities while simultaneously supporting institutional reporting of program-level SLOs. The remainder of this article is dedicated to a case study in which I discuss the implementation and evaluation of this method. I conclude by offering some thoughts about when and how the method can be most successfully applied in other settings.

2 Case Study: Background

Several years ago, I was involved in the assessment of a new introductory course in data science at a large public university. The course was conceived as a more holistic approach to data science education on campus than previous course offerings and was initially cross-listed between the computer science and statistics departments. A team of administrators, course instructors, and other campus stakeholders was convened to assess the new course and offer suggestions for future improvement. This team initially developed several high-level questions to guide its work: (1) what did we *want* students to learn in this introductory data science course; (2) what were students *actually* learning in the course; (3) how did the course and its content relate to other curricula on campus; and (4) what role(s) did it play?

To address these questions, I designed and implemented an integrated method to assess student learning in the course. As an initial step, I collaborated with one of the primary instructors to identify a list of SLOs. We settled on twelve distinct objectives, listed in figure 1. The objectives spanned several levels of cognitive thinking from application ("Calculate specified statistics of a given data set") to evaluation ("Given the result of a statistical analysis from the course, form correct conclusions about a question based on its meaning"). The SLOs also addressed dispositional learning objectives, such as "Articulate the benefits and limits of computing technology for analyzing data and answering questions." Once we had finalized the list of SLOs, the next step was to figure out how to assess whether students were achieving them.

The course was large, with roughly four hundred students enrolled in the spring semester's single section. Each week, students attended three lectures and a lab that could be completed in person or remotely. Thematically, the course was organized into several units: (1) data science—an overview of data

Upon completion of this course, students should be able to:

1. Write correct small programs that manipulate and combine data sets and carry out iterative procedures.
2. Extend a program with multiple functions so that it runs correctly with additional functionality.
3. Calculate specified statistics of a given data set.
4. Identify the sources of randomness in an experiment.
5. Formulate a null hypothesis that relates to a given question, which can be assessed using a statistical test.
6. Carry out statistical analyses including computing confidence intervals and performing hypothesis tests in a variety of data settings.
7. Given the result of a statistical analysis from the course, form correct conclusions about a question based on its meaning.
8. Given a question and an analysis, explain whether the analysis addresses the question and how the analysis could change and still address the question.
9. Articulate the benefits and limits of computing technology for analyzing data and answering questions.
10. Correctly generate and interpret histograms, bar charts, and box plots.
11. Correctly make predictions using regression and classification techniques.
12. Assess the accuracy and variability of a prediction.

Figure 1: List of student learning objectives

science; (2) tables—using Python to manipulate information; (3) visualization—interpreting and exploring data through visualizations; (4) sampling—understanding the behavior of random selection; (5) prediction—making predictions from data; (6) inference—reasoning about populations by computing over samples; and (7) probability—making assumptions and exploring their consequences.

In order to accommodate the course's large size, the instructor utilized Gradescope: a digital grading platform developed by several of the instructor's former students.[2] Students enrolled in the course completed their work electronically or by hand on a standardized template, and uploaded a PDF of their completed assignment into Gradescope. At that point, instructors or teaching assistants graded each assignment using a common grading rubric. The rubric was precise, breaking down individual problems into multiple predetermined components, each with their own point values. The course instructor adopted Gradescope primarily to increase grading efficiency among the course's teaching assistants, and to simplify the calculation and management of students' grades.[3]

In order to assess SLOs, I utilized the "assignment statistics" tool in Gradescope. For each distinct question or subquestion that could be graded, I determined whether the question addressed any of the twelve SLOs. I then tagged each question with one or more keywords associated with the SLOs. For example, a question that required students to adapt a piece of python code they had already written in order to add a new column to a data set would be tagged with SLOs 1 and 2: "write correct small programs that manipulate and combine data sets and carry out iterative procedures," and "extend a program with

[2] https://gradescope.com/

[3] For a detailed summary of Gradescope's capabilities, see Singh et al. (2017). At the time, the university's course management system (CMS) did not offer the same digital grading capabilities as Gradescope. Today, CMS products like Canvas and others provide similar options for digital grading in a large course.

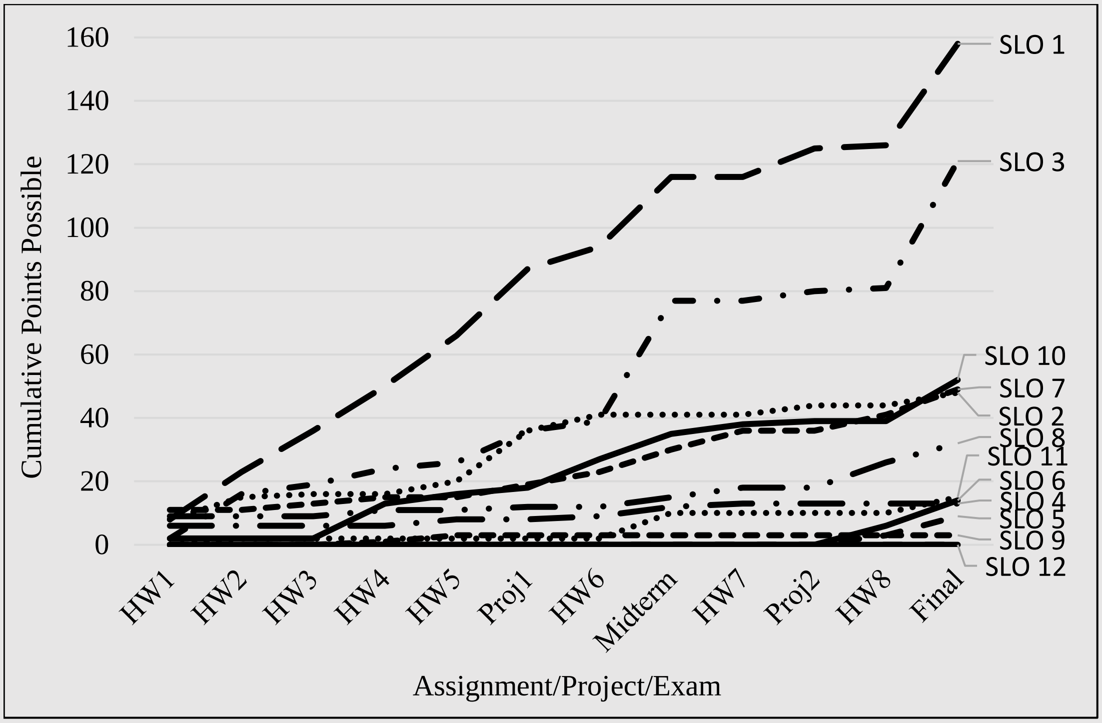

Figure 2: Cumulative points possible by student learning objective (SLO)

multiple functions so that it runs correctly with additional functionality." Once all students' assignments had been graded, Gradescope allowed me to view summary statistics about how students had performed on the questions tagged with each individual SLO. Specifically, I could easily determine how many possible points had been associated with each SLO in the assignment, and how students had performed overall on the questions associated with each SLO.[4]

3 Case Study: Results and Discussion

Over the course of the semester studied, there were a total of 337 possible points students could earn. Overall, students earned on average 82 percent of the points possible in the course. Keep in mind while interpreting the results below that each question in each assignment, project, or exam could have addressed zero, one, or multiple different SLOs.

Table 1 reports assessment results organized both by individual assignment and for the course overall. Several patterns quickly emerge. First, it is clear that most assignments addressed only a handful of the twelve SLOs. The final exam was an exception, in that it addressed nine of the twelve SLOs. Second, several SLOs were much more heavily stressed throughout the course than others, as evidenced by the number of points assigned to each SLO in the "Total" column. This is also visually apparent in figure 2, which shows the cumulative points possible by SLO across different assignments, projects, and exams. SLOs 1 and 3—those focused on writing programs and calculating statistics—were reflected in 158 and 121 points, respectively, throughout the semester. Next, SLOs 2, 7, and 10 were each reflected in roughly 50 points each. And SLO 12, which focused on assessing prediction accuracy, was not captured by any course question. Third, when a particular SLO was included in several sequential assignments, students tended to perform better on the objective over time. Consider, for example, SLO 1 over the first five assignments. As students gained more practice writing programs in python code, they got better at it.

[4] At the time of this case study, Gradescope's tagging capability was not widely available in other CMSs. Now, an increasing number of products are offering similar functionality. For example, instructors using Canvas can achieve similar ends using Gauge, an optional assessment management system (https://www.canvaslms.com/gauge/).

Table 1: Quantitative assessment of student learning objectives (SLOs)

SLO	Assignment 1		Assignment 2		Assignment 3		Assignment 4		Assignment 5		Project 1		Assignment 6	
	Points[a]	% Correct[b]	Points	% Correct	Points	% Correct	Points	% Correct	Points	% Correct	Points	% Correct	Points	% Correct
1. Write programs	8	87	15	89	13	94	14	91	16	95	21	81	7	86
2. Extend a program	8	87	7	92	1	95	-	-	4	89	16	78	5	83
3. Calculate statistics	2	96	14	89	3	91	5	79	2	95	10	66	3	91
4. Identify sources of randomness	6	87	-	-	-	-	-	-	2	90	-	-	1	98
5. Form a null hypothesis	-	-	-	-	-	-	-	-	-	-	-	-	-	-
6. Statistically test a hypothesis	-	-	-	-	-	-	-	-	-	-	-	-	-	-
7. Form correct conclusions	11	84	-	-	2	84	2	88	-	-	4	68	4	92
8. Identify appropriate analyses	9	84	-	-	-	-	2	88	-	-	1	93	-	-
9. Articulate benefits and limits of computing	-	-	-	-	-	-	1	80	2	97	-	-	-	-
10. Generate graphs	2	83	-	-	-	-	11	82	3	84	2	74	9	91
11. Make predictions	-	-	2	78	-	-	-	-	-	-	-	-	-	-
12. Assess prediction accuracy	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Total[c,d]	27	89	25	91	13	100	21	89	22	97	23	85	17	92

SLO	Midterm		Assignment 7		Project 2		Assignment 8		Final Exam		TOTAL	
	Points	% Correct	Points	% Correct	Points	% Correct	Points	% Correct	Points	% Correct	Points	% Correct
1. Write programs	22	67	-	-	9	96	1	73	32	68	158	82
2. Extend a program	-	-	-	-	3	99	-	-	4	70	48	84
3. Calculate statistics	38	60	-	-	3	92	1	73	40	70	121	72
4. Identify sources of randomness	3	87	1	87	-	-	-	-	-	-	13	88
5. Form a null hypothesis	-	-	-	-	-	-	3	83	6	60	9	68
6. Statistically test a hypothesis	-	-	-	-	-	-	6	89	8	63	14	74
7. Form correct conclusions	7	70	6	86	-	-	5	93	8	60	49	79
8. Identify appropriate analyses	3	87	3	93	-	-	8	82	6	67	32	82
9. Articulate benefits and limits of computing	-	-	-	-	-	-	-	-	-	-	3	91
10. Generate graphs	8	43	3	84	1	95	-	-	13	71	52	75
11. Make predictions	8	67	-	-	-	-	-	-	5	71	15	70
12. Assess prediction accuracy	-	-	-	-	-	-	-	-	-	-	0	-
Total	45	64	11	94	26	97	17	88	90	68	337	82

[a] "Points" signifies "points possible" for each SLO (rows 1–12) or in total for an assignment/project/exam (Total row).
[b] "% Correct" signifies the average proportion of possible points earned across all students for each SLO or in total for an assignment/project/exam (**Total** row).
[c] Any question can address zero, one, or multiple SLOs. Therefore, the numbers in the **Total** row need not match the data in rows 1–12.
[d] Extra credit points are available, but do not contribute to any SLO. This is why the "% Correct" value in the Total row may appear unduly large.

The results in table 1 provided the information necessary for course instructors to adjust their curriculum in future semesters. For example, it became clear that instructors would either need to remove assessing prediction accuracy as a SLO, or better integrate it into the coursework. The same conclusion was drawn for SLO 9: articulating the benefits and limits of computing technology. Alternatively, it became clear that hypothesis testing—a core topic for the course—was only being introduced at the very end of the semester. Instructors decided to cover that material sooner in subsequent offerings of the course.

The fundamental value of the results in table 1 is that they reflect course content and student performance in terms of SLOs, rather than in terms of specific exercises or assignments. While the course instructor likely had a good sense of whether students had mastered various aspects of the python coding language, my analysis provided insight into whether students were achieving higher-order objectives, such as their ability to draw appropriate conclusions from data in a general sense. This information was a powerful tool for directing future course offerings, and my method could even be applied throughout a single semester to provide an instructor with real-time feedback about students' learning.

A digital grading platform with tagging functionality such as Gradescope or Gauge is key for this approach to be feasible. Indeed, once such a platform is adopted to support the grading process, little additional effort is required to tag individual questions and analyze those tags' resulting statistics. In the future, digital grading platforms could offer yet more powerful analytic tools, potentially even tracking student-level performance on SLOs over various assignments. In the era of big data, course instructors will be able to take advantage of easily accessible analytics.

4 Conclusion

Once a course instructor decides to adopt a digital grading platform with the ability to tag specific questions, it can be straightforward to effectively assess student learning by linking individual questions to predetermined SLOs. Such an approach addresses two long-standing barriers to effective learning assessment: (1) the extra work usually needed to assess students' learning, and (2) the complex nature of some higher-order learning objectives. While the case study presented in this commentary is specific to data science, the underlying method can be easily applied to almost any field of study. Economics is particularly well-suited to such an approach since it combines tangible skills (mathematics, graphing, calculation, etc.) with higher-order cognitive concepts (utility maximization, budget constraints, weighing marginal trade-offs, etc.).

More broadly, there are many benefits from adopting digital grading—especially in large classes. There are returns to scale both in the efficiency of grading each student's work (Anglin et al. 2008) as well as in the ability to analyze the resulting data. For these reasons, I predict that applied economics instructors will increase their adoption of digital grading platforms in the coming years. As I have demonstrated in this commentary, doing so will open doors to new and powerful methods for assessing student learning outcomes.

Acknowledgements: This work would not have been possible without the collaboration and support of several anonymous administrators and instructors of the course studied in this article. I am greatly appreciative of their partnership. I have no commercial relationship with Gradescope, the digital grading platform discussed in this article, and my commentary should not be interpreted as an endorsement of any particular commercial product. This research has been determined exempt by the Mississippi State University Institutional Review Board for the Protection of Human Subjects in Research under protocol IRB-18-367. Any remaining errors are my own.

About the Author: Andrew W. Stevens is an Assistant Professor, Department of Agricultural Economics, Mississippi State University. (Corresponding Author: a.stevens@msstate.edu)

References

Anderson, L.W., and D.R. Krathwohl, eds. 2001. A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives. New York: Longman.

Anglin, L., K. Anglin, P.L. Schumann, and J.A. Kaliski. 2008. "Improving the Efficiency and Effectiveness of Grading Through the Use of Computer-Assisted Grading Rubrics." *Decision Sciences Journal of Innovative Education* 6(1):51–73.

Atwood, S.A., and A. Singh. 2018. "Improved Pedagogy Enabled by Assessment Using Gradescope." Paper presented at American Society for Engineering Education Annual Conference and Exposition, Salt Lake City UT, 24–27 June. Paper ID #21186.

Banta, T.W., E.A. Jones, and K.E. Black. 2009. *Designing Effective Assessment: Principles and Profiles of Good Practice*. San Francisco CA: Jossey-Bass.

Biggs, J., and C. Tang. 2011. *Teaching for Quality Learning at University: What the Student Does*, 4th. ed. New York: Open University Press and McGraw-Hill.

Birenbaum, M., K. Breuer, E. Cascallar, F. Dochy, Y. Dori, J. Ridgway, R. Wiesemes, and G. Nickmans. 2006. "A Learning Integrated Assessment System." *Educational Research Review* 1:61–67.

Bloom, B.S., M.D. Engelhart, E.J. Furst, W.H. Hill, and D.R. Krathwohl, eds. 1956. *Taxonomy of Educational Objectives: The Classification of Educational Goals. Handbook I: Cognitive Domain.* New York: David McKay.

Brown, S., J. Bull, and P. Race, eds. 1999. *Computer-Assisted Assessment in Higher Education.* London: Kogan Page.

Bull, J. 1999. "Conclusion: A Glimpse of the Future." In S. Brown, J. Bull, and P. Race, eds. *Computer-Assisted Assessment in Higher Education*. London: Kogan Page, pp. 193–197.

Erickson, B.L., C.B. Peters, and D.W. Strommer. 2006. *Teaching First-Year College Students*. San Francisco CA: Jossey-Bass.

Fink, L.D. 2013. Creating Significant Learning Experiences, Revised and Updated: An Integrated Approach to Designing College Courses. San Francisco CA: Jossey-Bass.

Fry, H., S. Ketteridge, and S. Marshall, eds. 2009. A Handbook for Teaching and Learning in Higher Education: Enhancing Academic Practice, 3rd. ed. New York: Routledge.

Marzano, R.J. 2001. *Designing a New Taxonomy of Educational Objectives*. Thousand Oaks CA: Corwin Press.

Seden, R. 1999. "Delivering Automated Innovative Assessment in a Varied Curriculum." In S. Brown, J. Bull, and P. Race, eds. *Computer-Assisted Assessment in Higher Education*. London: Kogan Page, pp. 163–170.

Singh, A., S. Karayev, K. Gutowski, and P. Abbeel. 2017. "Gradescope: A Fast, Flexible, and Fair System for Scalable Assessment of Handwritten Work." *Proceedings of the Fourth (2017) ACM Conference on Learning @ Scale*, pp. 81–88.

Suskie, L. 2018. *Assessing Student Learning: A Common Sense Guide*, 3rd. ed. San Francisco CA: Jossey-Bass.

Wiggins, G., and J. McTighe. 2005. *Understanding by Design*, 2nd. ed. Alexandria VA: Association for Supervision and Curriculum Development.

1(1) doi: 10.22004/ag.econ.294011