The Report Committee for Nathaniel David Raley
certifies that this is the approved version of the following report:

Learning Analytics in Large College Courses: Facilitating Retention and Transfer of Learning Through Targeted Retrieval Practice

APPROVED BY

SUPERVISING COMMITTEE:

S. Natasha Beretvas, Supervisor

Andrew C. Butler, Co-Supervisor

Learning Analytics in Large College Courses: Facilitating Retention and Transfer of Learning Through Targeted Retrieval Practice

by

Nathaniel David Raley, B.A.

REPORT

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Statistics

THE UNIVERSITY OF TEXAS AT AUSTIN

May 2016

Learning Analytics in Large College Courses: Facilitating Retention and Transfer of Learning Through Targeted Retrieval Practice

Nathaniel David Raley, M.S. Stat.
The University of Texas at Austin, 2016

Supervisors: S. Natasha Beretvas
Andrew C. Butler

Spaced retrieval practice is known to benefit both long-term retention and transfer of learning, two important goals of education. However, most classes are not designed in a way that facilitates frequent quizzing or revisiting previously covered topics; this is particularly true in higher education, where a small number of exams typically account for the bulk of a student's grade. Recently, a large undergraduate course at the University of Texas has implemented a new class structure that replaces high-stakes tests with daily quizzes administered during class via computer; furthermore, quiz items previously answered incorrectly can appear at random on future quizzes. Together, these innovations are an excellent first step toward bringing spaced retrieval practice into the college classroom. However, I propose that technology can be further leveraged in classes such as these to more optimally choose repeated items. Given graded student quiz data from one semester of this course, I use Multidimensional Item Response Theory (MIRT) and Sparse Factor Analysis (SPARFA) to jointly estimate concepts underlying the items and each students' mastery of these concepts. After comparing these factor-analytic methods, I also explore free-response and student chat data using basic natural language processing. It is concluded that techniques from learning analytics can help realize the full potential of spaced retrieval practice in the classroom by optimizing the selection of repeated items so as to target remediation. Furthermore, such techniques can be used to introduce variability into retrieval practice, encouraging a deeper understanding of the content which is more likely to transfer to novel problems.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Any system of education makes many assumptions about how people learn. While some of these assumptions may be grounded in scientific research, far more are products of political decision-making, results of practical compromise, or pieces of received tradition handed down through the generations. In particular, the structural features of schooling are often heavily dependent on political considerations, practical expedients, and accidents of history.

Things like the length of a standard school-year, a semester, a school-day, or a class-period; things like the division of students into different schools based on grade-level and the division of students into different grade-levels based on age. Assumptions are made about which subjects are to be taught, when, and in what sequence; about whether subjects should comprise multiple courses, and if so, which should be taught, when, and in what sequence. Within a given course, there are assumptions about the progression through distinct units of material. Within a unit, which activities and exercises are best, and in what sequence? Traversing levels of granularity in this way, we see an exponential growth of decision points that underlie the way learning is structured. In the face of such a daunting chain of dependencies, it is easy to see why the traditional and efficient approaches persist, often unquestioned.

Perhaps the most crucial of these assumptions are those made about assessing what students have learned and when they can be said to have learned it. Implicit in the structure of schooling—with its teaching and testing—is the notion that if the weighted average of graded student work is above a given level, they "pass" the class. If students pass all of their classes, they advance to the next grade; if they pass enough grades, they graduate. Within a given course, subject matter is typically broken down into several stand-alone units of material. Students are taught the material and then tested over it soon or immediately thereafter; then the class moves on to the next unit and repeats the process. Tests are often weighted more heavily than other assignments and thus account for the greatest proportion of the final course grade, largely determining advancement. Notice in all of this that learning has been implicitly equated with passing: in general, students are considered to have "learned" the material if they have "passed" each unit in a given course; if they have "passed" enough courses, then they are assumed to have learned enough to graduate.

As I see it, the goal of schooling is to teach students knowledge and skills that can be flexibly applied outside of the classroom and that remain accessible to them over time. Grades given for coursework are at best indicative of only short-term knowledge gains, yet there is a widespread belief that this learning will persist over time and transfer usefully into the "real-world." To begin this report, I would like to point out three common assumptions made about student learning that are incorrect and ultimately detrimental. The first is the assumption that testing is for assessment purposes only: that a test is a learning-neutral event for measuring what a given student knows. The second assumption is that learning, as measured by tests, will persist over time. The third assumption is that learning, as measured by tests, will transfer out of the classroom and be widely applicable across situations. We will see that these misguided assumptions are actually quite closely related. After describing each in some detail, I will introduce a college course that has challenged some of these assumptions and is reaping the benefits. Finally, I will explore how courses like this can avoid the negative effects of these assumptions and maximize their students long-term learning outcomes given the constraints of a semester-long college course. I will look at data generated from this course and employ several techniques from learning analytics in order to make further recommendations for improving students' learning outcomes, specifically long-term retention and transfer.

1.1 Testing and Spacing

The idea that testing could have beneficial effects on learning is gradually making inroads into our modern educational practice, but it is still far from universally embraced. Notwithstanding this lukewarm reception, a long history of research (e.g., Gates, 1917) has shown that retrieving information from memory increases the likelihood that the information will be retrievable in the future, a robust finding known as the testing effect (Carrier & Pashler, 1992; see Roediger & Butler, 2011 for review). Thus, the memory retrieval required by testing is thought to enhance learning by directly modifying the retrieved content (e.g., elaborating upon the representation of this content in memory, increasing its availability and accessibility; Bjork & Bjork, 1992).

What this means is that, after initial learning, being tested over the material produces better memory for that material than does an equivalent amount of time spent restudying the material. The relative benefit of testing over restudy becomes larger as the delay before the final recall test grows longer: relative to restudying, retrieving information results in slower forgetting over time and thus better long-term retention (for discussion of possible mechanisms, see Kornell, Bjork, & Garcia 2011). These benefits are observed both in the laboratory and in the classroom (e.g., McDaniel et al. 2015). What's more, the benefits of testing extend beyond just retention of information: recent research suggests that testing leads to a deeper understanding of the material than does restudying, thus facilitating transfer of learning (Butler, 2010). This topic is of particular importance to the present report and will be given a fuller treatment in an upcoming section (see Section 1.3).

Despite its potential to enhance student learning in the classroom, testing still comes under fire largely because frequent assessments have the appearance of being learning-neutral and thus a poor use of school-time and resources. Worse still, it's not only those outside the education system who are discounting this important effect: many students fail to understand the power of testing to enhance learning; when college students were asked what kind of strategies they used when studying, repeatedly restudying the material was the favored approach, while only 11% reported self-testing (Karpicke, Butler, & Roediger, 2009). Thus, to bring the benefits of the testing effect into real educational settings, it has been recommended by leaders in the field that frequent quizzes be employed in the classroom to engage students in retrieval practice (Bjork, Dunlosky, & Kornell 2013).

Related to the testing effect—and perhaps even more well known—is the spacing effect: the finding that spacing out one's studying or testing sessions produces superior learning relative to an equivalent amount of studying or testing in a single sitting or in sessions occurring closer together in time (Cepeda et al., 2006 for review). That is, those who spread their practice out over time enjoy greater long-term retention of that information than those who practice for the same amount of time but do not space it out. The benefit of spaced practice over massed practice on retention holds across learners of all ages and subject-matter of all kinds, including learning grammar, spelling, reading skills, advanced mathematics, motor skills, foreign language vocabulary, history, and more (Carpenter et al, 2012).

While teachers often admonish their students to study a little every day instead of cramming right before the test—and thus have some intuitive understanding that spaced-out is better than massed together— this principle is seldom reflected in the structure of their courses. At the college level, for example, the standard course format is still such that 2 or 3 high-stakes tests determine the bulk of students' grades. If cramming right before the exam results in just as good (if not better) performance, then students have little reason to space out their studying. If it is also true that spacing results in superior long-term retention, then instructors are unwittingly creating a perverse incentive structure: one which rewards behaviors that lead to transitory learning while penalizing those leading to durable learning. Depressingly, this turns out to be the case. Cramming, though it results in much poorer long-term retention, can produce equivalent—indeed sometimes better—recall on an immediate test (or a test after a short retention interval) than does spacing out study sessions (e.g., Rawson & Kintsch, 2005). Thus, cramming is an effective way to get good grades but a terrible way to achieve durable learning, and it is therefore incumbent on educators to incentivize long-term retention with things like cumulative assignments. In view of the foregoing discussion about the testing effect and the spacing effect, it seems that the most effective assignment scheme would be one in which students took daily quizzes (for retrieval practice) that were cumulative, revisiting previously learned material (for the benefits of spacing). Additional concerns for instructors relate to educational materials such as textbooks: many texts present information in a non-distributed (i.e., massed) way, by having self-contained chapters and practice problems which pertain only to the material just presented.

How do we know that spaced retrieval practice is so good for long-term retention? And just how long is long-term retention? Can we achieve indefinite retention and if not, what's the best we can hope for? Harry Bahrick's pioneering research into long-term retention has offered many exciting answers to these questions. With respect to the first question posed, he conducted a 9-year longitudinal study using his own family as participants (Bahrick et al., 1993). They learned and relearned 300 English-foreign language word pairs, varying both the number of relearning sessions (13 vs. 26) and the interval between sessions (14, 28, or 56 days) within subjects. After the training, retention was tested 1, 2, 3, and 5 years later. He found strong main effects both for the additional sessions and for longer spacing intervals on retention. In fact, 13 retraining sessions spaced 56 days apart resulted in retention benefits comparable to 26 sessions spaced 14 days apart. But while the longer intervals resulted in much better retention 5 years later, they hindered initial learning during the training sessions. Thus, we are again cautioned against the dangers of judging learning from performance on tests given soon afterwards. As Schmidt and Bjork (1992) put it, "manipulations that maximize performance during training can be detrimental in the long term; conversely, manipulations that degrade the speed of acquisition can support the long-term goals of training" (p. 207).

1.2 The Forgetting Curve and The "Permastore"

Analysis of people's memory for things like Spanish and algebra years after they took their last course in the subject has provided many important insights about maximizing long-term retention (Bahrick, 1983, 1984a; Bahrick, Bahrick, & Wittinger, 1975). These cross-sectional studies survey hundreds of people about their background in a given subject—when was their most recent course in it, how many classes total they took in it, what grades did they receive, and to what extent have they used the material since they quit learning it. This results in a sample of participants with varying spans of time since content acquisition (the "retention interval") and varying degrees of initial learning, all naturalistically acquired. Then, these participants are tested over their retention of the subject (e.g., an introductory-level Spanish language test of reading comprehension, vocabulary, and grammar).

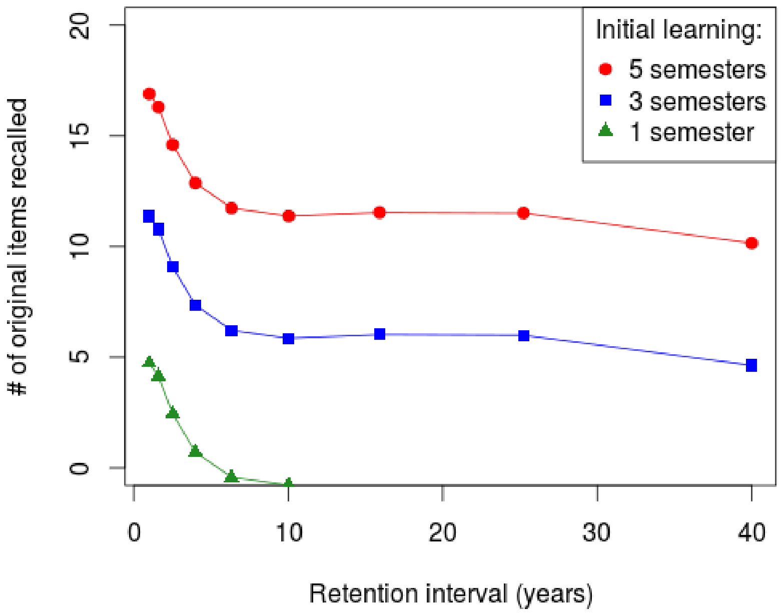


Figure 1.1: Retention of Spanish-English vocabulary (recall) by level of initial learning. Figure adapted Bahrick's (1984a) Fig. 6, using the regression equation given in Table 8, transformed back to linear from a logarithmic scale

From this data, researchers can generate memory curves by plotting retention over time for different degrees of initial learning (see Figures 1.1 and 1.2). For example, a retention function of constant slope would indicate that a constant number of things are forgotten per unit time. One extremely interesting finding from these analyses is that in general, memory curves decline exponentially for the first 3 to 6 years after learning has ceased, but that after this time retention almost asymptotes, remaining largely unchanged even after periods of up to 50 years. Concretely, 3 years after taking a single semester of Spanish, almost all of the Spanish-English vocabulary covered in the course was lost. However, those who took five semesters of Spanish recalled 60% of their original recall score more than 25 years later (Bahrick, 1984a). Still more robust findings are observed with retention of basic math: it has been shown that people who take several mathematics courses in college show *no significant declines* in their retention of high school algebra or geometry content during a 50-year retention interval, even if they have not used or in any way rehearsed the material during that time (Bahrick & Hall, 1991; see Figure 1.2).

In general, if your initial learning was high (i.e., multiple learning sessions spaced out over time), your long-term retention stabilizes at a higher level than would be the case if your initial learning was low. With more initial learning comes an increased portion of content remaining accessible over an extremely long time period: the information in this so-called "permastore" is content that is destined to survive for 25+ years in the absence of rehearsal (Bahrick, 2000). "The most important predictors of the rate of performance loss pertain to the conditions of original exposure or practice. When rehearsals or exposures are extended over several years, performance levels remain stable for half a century without the benefit of further practice. When the same content is acquired over a shorter period, performance tends to decline rapidly and continuously." (Bahrick & Hall, 1991, p. 30).

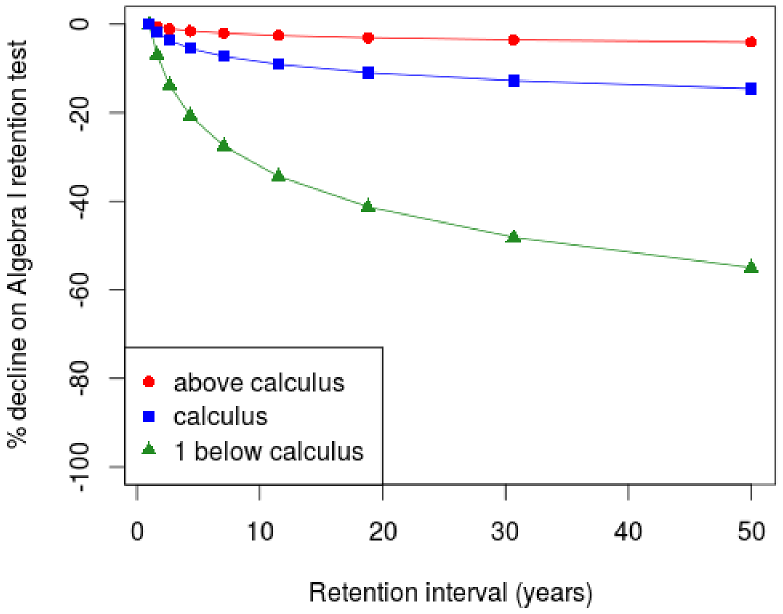


Figure 1.2: Percent decline of Algebra I knowledge for different numbers of college mathematics courses taken. Figure adapted Bahrick & Hall's (1991) Fig. 2, using the regression equation and settings given in their Tables 7 and 8, transformed back to linear from a logarithmic scale.

As an authorial aside, I am currently in my 7^{th} year of post-secondary education and during this time I have never been in a class that gave weekly (to say nothing of daily) quizzes. Only very rarely were tests cumulative, homework was sporadic at best, and I can't recall spending any significant class time revisiting previous material. This is regrettable, because these are some of the simplest things educators can do to implement spacing and retrieval practice into their classes!

Not only would restructuring courses in this way benefit students' long-term retention, it appears also to be more egalitarian. While individual-difference variables such as SAT scores and course grades (in traditional courses) predict acquisition and therefore final test performance in studies of long-term retention, these variables *do not* significantly affect the rate of decline of performance over time. Bahrick & Hall (1991) found no significant interaction between standardized test scores or grades and the rate of decline in retention. What this means is that, since aptitude predicts acquisition, the standard course model (where units are covered in isolation and a couple high-stakes exams determine your final grade), which incentivizes cramming, benefits high-aptitude students. However, if learning was structured differently in these courses—allowing spaced retrieval practice to bring all students up to a high level of initial learning—then retention over time appears to be equally good, irrespective of ability. Though this hypothesis hasn't been formally evaluated, it appears that a curriculum based on spaced retrieval practice could serve to reduce the grade disparity between low and high ability students by equalizing levels of initial learning and avoiding the illusory learning gotten by cramming that differentially favors high-ability students.

1.3 Transfer of Learning

> "The effectiveness of a training program should be measured not by the speed of acquisition of a task during training or by the level of performance reached at the end of training, but rather by the learner's performance in the post-training tasks and real-world settings that are the target of training."
>
> –Robert Bjork (1991, p. 47)

Before delving into the present study, I want to describe some exciting research indicating that retrieval practice benefits more than just retention of information: it appears to also facilitate a deeper understanding of the material so practiced, resulting in the increased ability to transfer one's learning to new problems and in different situations (Butler, 2010; Carpenter, 2012). Butler (2010) demonstrated that, relative to repeated studying, repeated retrieval of material via testing promoted transfer by increasing performance on new inferential questions in different knowledge domains.

Indeed, it appears that even very simple spaced-practice interventions can have a large impact on both knowledge retention and transfer into real-world context, as evidenced by Dolan et al. (2015). These authors conducted a randomized controlled study with medical students who were completing their residency. After receiving a 1-hour case-based lesson on osteoporosis care and fracture prevention, students in the control group received one email containing a 25-item multiple choice self-assessment. Students in the intervention group received the *same 25 multiple choice items*, but instead of being delivered all at once, 1-3 questions were emailed over a 3-6 month period. Items answered correctly were repeated 1 time 28 days later, while items answered incorrectly were repeated twice at 14 day intervals (the variability in the length of time for treatment was due to differences in the number of incorrect responses among students). Ten months after the start of the intervention, the treatment group significantly outperformed the control not only on a bone-health knowledge assessment, but also on real clinical outcome measures: they screened more patients for low bone density, screened them more accurately, and effectively treated more who were at risk for fracture. Studies like this are especially important, given that medical students have been shown to forget a substantial portion of basic knowledge by the time they begin clinical rotations (Butler & Raley, 2015).

Up to this point, “retrieval practice” has signified repeated opportunities to retrieve some information from memory, typically when prompted to do so by test questions. Both the questions and the information retrieved has been assumed to be the same, or at least very similar, for each retrieval attempt. But what would happen if students were asked *different* questions tapping the *same* underlying concept. Instead of being prompted by the same question three times, what if students had to answer three different questions about the same concept? Recent research is revealing that retrieval practice with variable examples of a concept results in greater retention and transfer of learning than does retrieval practice with the same questions (Butler et al. in press). This has important implications for structuring learning in the classroom; typically, teachers have access to large test-banks and item pools for assessment purposes, with many different items that measure the same concepts. These items with conceptual redundancies can be used not just to create alternate forms of a test but to give students variable retrieval practice. By providing spaced retrieval practice with varying examples, teachers may be able to boost student performance above and beyond the benefits of testing and spacing alone. All of this is to drive home the vastly underappreciated finding that retrieval practice through frequent testing facilitates both long-term retention and transfer of learning, thus calling into question several unstated assumptions made by our education system. We now turn to learning analytics and how technology can be used to leverage these findings from cognitive psychology and implement them seamlessly in the classroom.

1.4 Learning Analytics

This report uses techniques from learning analytics to study data generated in a college course that has recently implemented spacing and retrieval practice into their curriculum, doing away with their previous high-stakes testing format. According to the 1^{st} International Conference on Learning Analytics and Knowledge, "learning analytics is the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimising learning and the environments in which it occurs."

Thus, learning analytics examines the relationship between learners, content, institution, and educators. Long and Siemens (2011) propose that it comprises at least five distinct endeavors, generally progressing in the following order:

1. **Course-level:** learning trails, social network analysis, discourse analysis
2. **Educational data-mining:** predictive modeling, clustering, pattern mining
3. **Intelligent curriculum:** the development of semantically defined curricular resources
4. **Adaptive content:** adaptive sequence of content based on learner behavior, recommender systems
5. **Adaptive learning:** adaptive learning process (social interactions, learning activity, learner support)

The focus of the present report will be focused on #1 and #2 (course-level and educational data-mining) in order to make general recommendations about #3, 4, and 5. To make use of a common analogy, we are performing an autopsy after learning has taken place rather than a biopsy to insure healthy learning continues in real time.

1.4.1 Background for the Present Study

The Department of Psychology at the University of Texas at Austin has been developing an innovative online learning platform called TOWER (Texas Online World of Educational Research) which, among other features, is able to implement daily "in-class" quizzes and to provide immediate feedback on quiz performance (Pennebaker, Gosling, & Ferrell, 2013). In 2011-2012, two large Introductory Psychology courses were taught using TOWER; these courses were taught by two instructors who had previously taught the same course every year from 2006-2011 using traditional approaches. In the traditional course approach, four in-class exams of 40-45 multiple choice items were given across the semester; these exams accounted for 86% of the final grade, while writing assignments accounted for the remaining 14%. The TOWER-based course differed from the traditional course model in that there were no exams at all; instead, students were required to take an 8-item online quiz at the beginning of each class period. These 26 quizzes accounted for 86% of students' final grade, thus replacing the exams. The only other substantive difference between the two courses was that, unlike the traditional course, the TOWER-based course assigned weekly readings from online sources instead of from a textbook. Things like lecture format and sequencing of material were intentionally kept constant.

Results

The authors compared performance in the TOWER group (Fall 2011, $n = 901$) to that in the traditionally taught Comparison group (Fall 2008, $n = 935$). The Comparison group semester was chosen because Fall 2008 was the most recent year in which they had used the same demographic survey. This survey included items about parental education, which were used as a proxy for socio-economic status (SES). They evaluated student performance in several ways, the simplest of which was computing overall course grades for each student. The authors found that the mean course grade for the TOWER-based group was significantly lower than that for the Comparison group, even after controlling for parental education and year of course ($t = 2.5,\ p = .01,\ d = .12$). But they cautioned that, because the Comparison group had their exam grades curved upwards, these findings are perhaps misleading.

The second performance measure compared performance on specific items used for the TOWER-based daily quizzes that had previously been used in Comparison classes' exams. An item that had previously appeared on exams in years past was included in 17 of the 26 quizzes; on these 17 items, the TOWER group performed better (77.1% *vs.* 71.2%), albeit with marginal significance ($t = 2.01,\ p = .06,\ d = 1.01$).

The authors also wanted to compare student performance in other classes taken concurrently as well as performance in classes taken in the following semester, hypothesizing that the TOWER-based quizzes could encourage self-regulatory skills and good study habits which would then generalize to other courses. They ran a 2(Course: $TOWER\ vs.\ Comparison$)×3(Grade: $Fall{-}Psychology$, $Fall{-}other$, $Spring{-}all$) repeated measures ANOVA and found a Course-by-Grade interaction ($F(2, 1756) = 23.6,\ p < .001,\ d = 0.33$). Examination of this interaction effect revealed that TOWER students' grades were higher in their other concurrent courses (3.07 *vs.* 2.96 for the Comparison students) as well as in the following Spring semester (3.10 *vs.* 2.98).

Finally, the authors examined SES disparities and class performance. Though this has little direct bearing on the present report, their findings are worth reporting for completeness and general interest. In brief, they regressed course performance on SES, Course (TOWER *vs.* Comparison), and the interaction, while controlling for standing (years of college completed). They found a significant interaction ($B = -.05,\ t = 2.25,\ p = 0.03,\ d = .10$) indicating that grade differences between SES levels were greater in the Comparison courses than in the TOWER courses.

1.5 The Present Study

In the Fall of 2013, the TOWER-based version of this Introductory Psychology course was transformed into a Synchronous Massive Online Course (SMOC), a class taught entirely online to large numbers of students who participate remotely but in real time as the class occurs. All of the important aspects of the TOWER course discussed above remained the same in the SMOC: instead of exams, all students took one 10-minute benchmark quiz (BM) at beginning of each class, for a total of 28 BMs that account for 85% of their grade in the course. Of these 28, 26 featured 8 questions; on the 8-item BMs, 7 of the items addressed readings and lectures from the previous class, and the remaining item was one that the student had previously answered incorrectly. In rare cases where students had not previously missed any questions, the 8^{th} item was randomly selected from material earlier in the course. The penultimate BM quiz in the course had only 6 questions (5 new, 1 repeated), while the final BM quiz had only 2 items and was unrelated to course content. This last BM was excluded from the analysis, leaving a total of 27 BMs in the data (26 8-item BMs and 1 6-item BM). In an effort to combat cheating on the BMs, new items on each BM were drawn from a pool of items covering that day's material (see Figure 3.7). Note that giving students different items on quizzes poses issues from an test-equating perspective, and precautions should be taken to insure fairness in grading when different people answer different questions. However, this procedure results in multiple items about a given concept, which enables the introduction of variability during retrieval practice. See Section 2.1 for a detailed description of the data.

In this particular semester of the course, no textbook was assigned and all course readings were from outside/online sources. The main difference between this course and its previous instantiation (see Section 1.4.1) is that students "come to class" remotely via the internet, logging on at the same time watch a live, interactive video lecture delivered by the instructors. These are the same instructors described above, teaching the same course described above, and the lecture content was similar to that of years past. However, the format of the lecture itself was different: the live-streamed class has a TV talk-show feel, with the instructors sitting behind a desk discussing course content with engaging banter, interspersing demonstrations and video clips. An additional difference is that students were assigned to small groups and required to participate in online chats to discuss various topics covered in class. The data generated from this course and the methods used to analyze them are discussed at length in the next chapter (Section 2.1).

Chapter 2

Data and Methods

2.1 Overview of Data

I have been granted anonymous access to the data generated during the Fall 2013 Synchronous Massive Online Course (SMOC) version of Psychology 301. This dataset includes each student's response to each benchmark quiz item, their free-response output on 4 writing assignments, and their contributions to online discussions. Initially, 939 students were registered for the course, of which 845 took at least 1 benchmark quiz (BM). Of these 845, 5 had only a single incorrect response (and no correct responses) while 3 had only a single correct response (and no incorrect responses); these were dropped from the analysis in order to compare IRT and SPARFA (see Section 2.2), thus bringing the effective sample size to 838 (see Figure 2.1 for histogram of students by number of items attempted).

Each student was assigned 27 benchmark quizzes (BMs) that related to material covered in the course, 26 of which contained 8 questions and 1 of which contained 6 questions, for a total of 214 questions per student. All but one of the items on each quiz were new items relating to the lecture and assigned readings; the other item was one that had been answered incorrectly by the

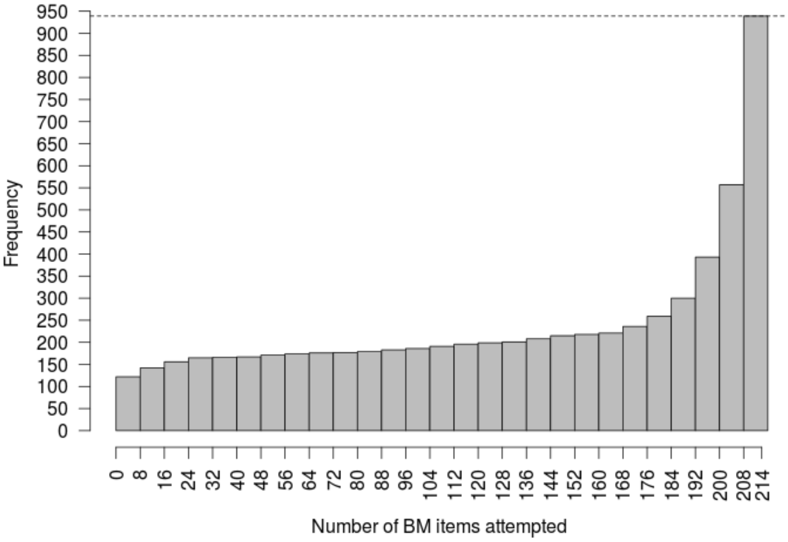


Figure 2.1: Cumulative histogram of students by number of benchmark items completed

student on a previous quiz (i.e., 7 “new” items and 1 “old” item). Each time students answered an item incorrectly, it was put into a pool of missed items associated with that student and from which items were drawn at random to appear in later quizzes (see Figure 2.2). In the rare event that students had not yet missed any items, a past item was simply chosen at random. Thus, students were exposed to a maximum of $7 \times 27 = 187$ unique items; 27 of those items repeated, making 214 items total as noted above (27 benchmarks with 8 items plus 1 benchmark with 6 items). This maximum was not attained; even the best student in the class missed 11 of their 214 attempts.

The size of the item pool from which BM questions were drawn—the total number of unique questions across all students—was 540. With our sample of 845 students, there are $540 \times 845 = 456,300$ possible student-question tuples, and thus the same number of potential responses. However, as noted above, students each answered a maximum of 187 unique items, and so the maximum of possible observed student-question tuples was only $187 \times 845 = 158,015$. That is, if students answered every item they were assigned, they would have answered at most 34.6% of the items in the item pool. In practice, because of student absences and students who dropped the course, the number of observed student-item tuples was 142,390 or 31.2% of the possible item-student tuples, meaning that the "gradebook" matrix of graded responses of all students to all items was 68.8% unobserved, and thus somewhat sparse (see Section 2.2).

Three datasets were created for use in these analyses (see Figure 2.3 for corresponding histograms). First, the full data for all students who completed at least a single question on a single assignment ($n = 845$). Second, a subset of this data consisting of students who "completed" the course ($n = 677$; see description of this subset below). Finally, a third subset was created consisting of only those students who completed all 27 quizzes ($n = 381$). Most of the following analyses were conducted on the largest of these datasets; unless it is specified to be otherwise, the dataset should be assumed to be the most inclusive of the three ($n = 845$).

The last time to officially drop an undergraduate course or change to a pass/fail basis at the university occurred during the tenth week of class; as there were 2 benchmarks per week, this would happen after the 20th benchmark. Therefore, if students did not complete any more work after this dead-

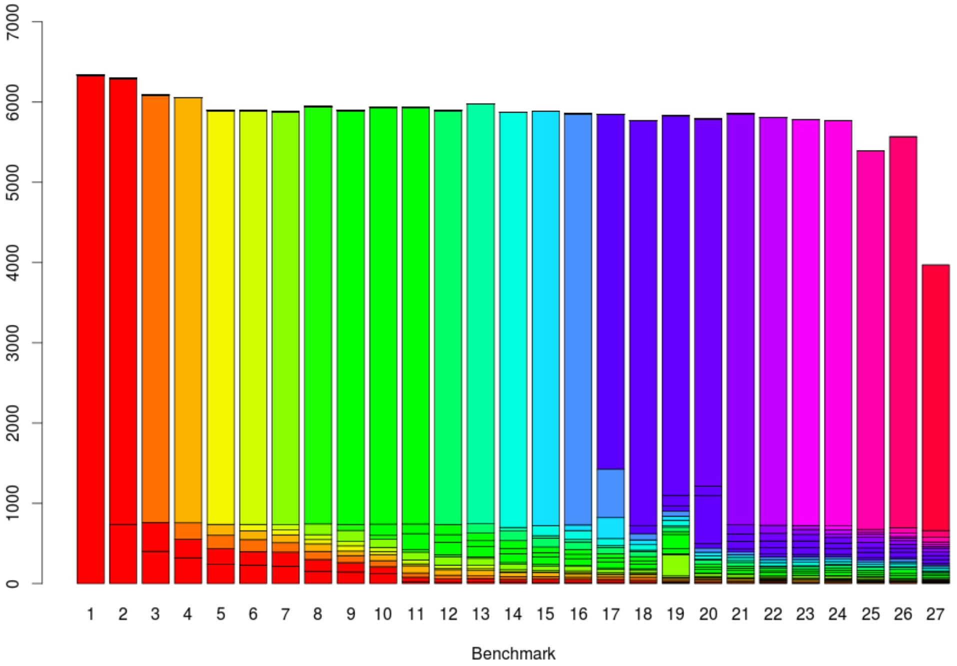


Figure 2.2: Cumulative histogram of items given to students on each BM. From left to right, the first appearance of a color indicates the BM of origin of those items. Note the regularity of repeated items in the bottom $1/8^{th}$ of each bar produced by the random sampling procedure. Note also the three "blip" BMs (17,19,and 20), which depart from this regularity.

line, it was assumed that they had dropped the course and their data was not included in the "completed" subset. As an additional qualification, students' overall class grade was computed by taking $(0.85)(bm\ score){+}(0.15)$. If this total was lower than 60, these students were also removed from the "completed" subset. This total represents the highest possible score given their benchmark performance, assuming perfect performance on the writing activities. Perfect performance was assumed because I did not have access to students' writing grades or their final course grades, but only their final benchmark grades; thus, I am choosing to err on the side of inclusivity by using this generous criterion.

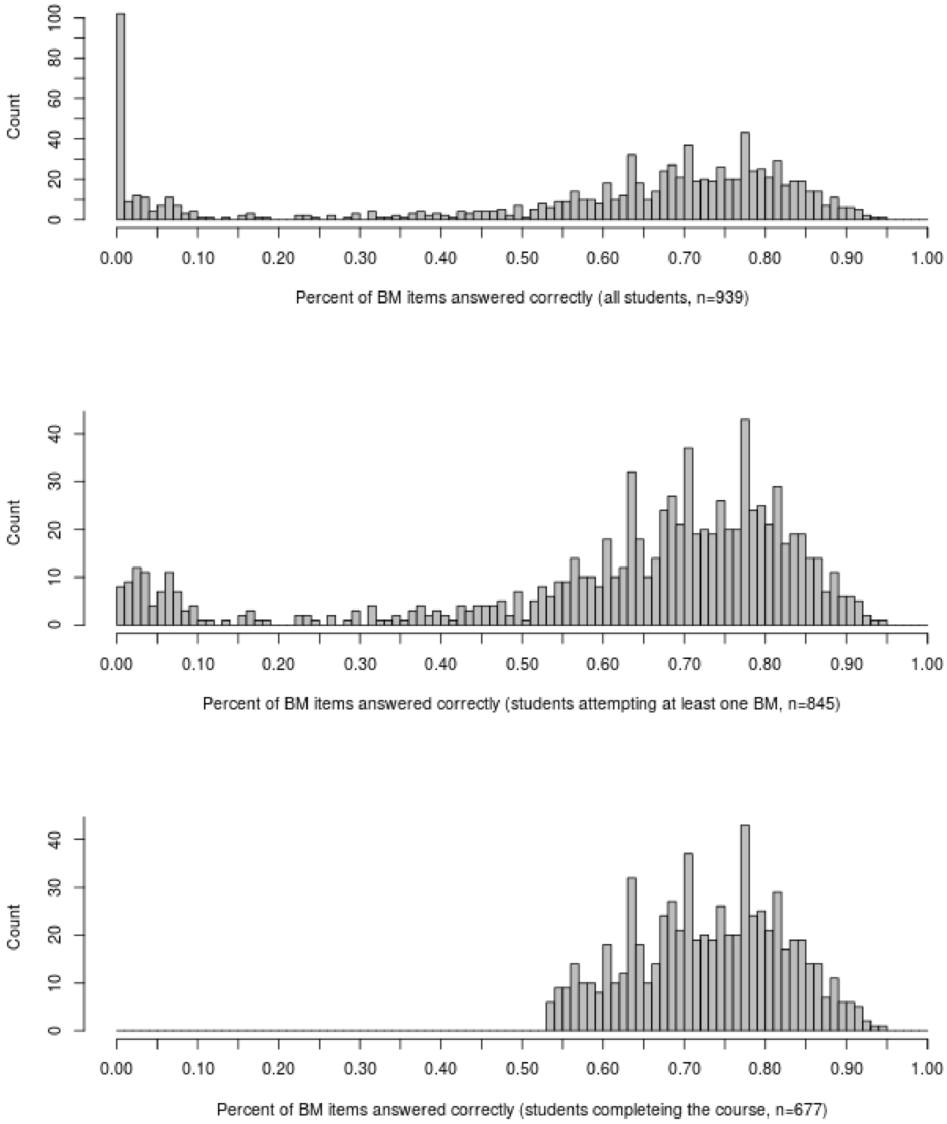


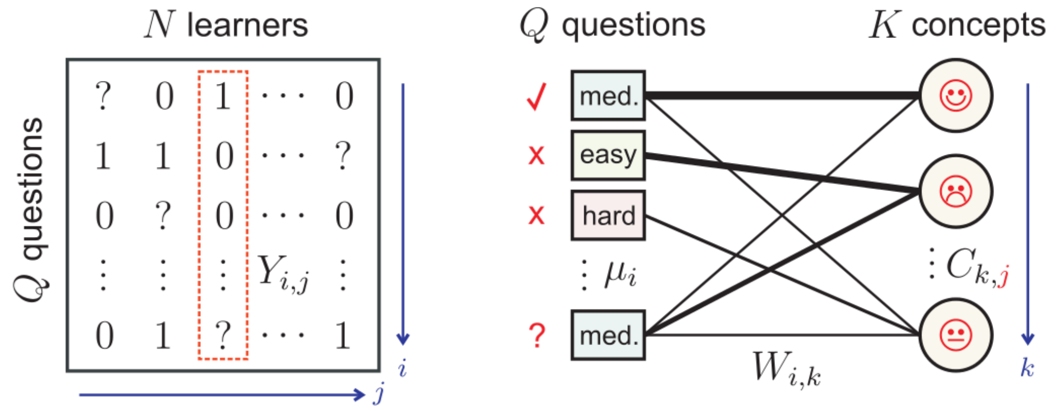
Figure 2.3: Histogram of approximate final grade in PSY 301, Fall 2013. Note the difference in y-axis scale for the top histogram

2.2 Methods: Estimating Student Knowledge

2.2.1 Sparse Factor Analysis (SPARFA)

Sparse Factor Analysis, or SPARFA, describes a set of machine learning techniques being developed for learning analytics by a research team based largely at Rice University (`www.sparfa.com`). I was granted permission by these researchers to look at their source code for the original instantiation of the SPARFA algorithms and to adapt it to the PSY 301 dataset as needed. SPARFA is currently an important component of several personalized learning systems being used in classrooms nationwide, where it is used to analyze student interactions (such as responses to quiz items) in order to assess their understanding on the fly so that student-specific feedback, remediation, enrichment, and sequencing decisions can be made optimally and in real time.

Within this framework, the probability that a student gives a correct response to an item is decomposed into three educationally relevant factors: the intrinsic difficulty of the item, the student's knowledge of a set of underlying concepts, and the degree to which a given item involves each concept. These three factors are estimated from *gradebook* formatted data: an $Q \times N$ matrix where each row i represents an item, each column j represents a student, and each entry $Y_{i,j}$ corresponds to whether question i was answered correctly (1) or incorrectly (0), or left unanswered (?) by student j (see Figure 2.4(a)). Thus, given a matrix of graded learner responses to questions, SPARFA provides estimates of (1) each student's knowledge of each concept, (2) each question's association with each concept, and (3) each question's intrinsic difficulty.



(a) Graded learner–question responses.

(b) Inferred question–concept association graph.

Figure 2.4: High-level SPARFA schematic. SPARFA fits a structural model to student "gradebook" data (binary, sparse), resulting in (b) a concept-mastery profile for each student (how well student j has mastered each concept k, indicated by smiley-faces), a concept-question association profile (how important each concept k is for each question i, indicated by the line-thickness) and estimates of how difficult each question is. Figure adapted from Lan et al. (2014).

SPARFA is one in a long tradition of machine learning algorithms that have been applied in educational settings. Other approaches to modeling student knowledge and analyzing response data have included Bayesian belief networks; these have the drawback of relying on prespecified item-concept dependencies, and typically only estimate one underlying concept. Indeed, most intelligent tutoring systems that are capable of probabilistically modeling concept-item associations, including Khan Academy, are only able to deal with a single latent concept (e.g., Dijksman & Khan 2011). In contrast, SPARFA estimates multi-concept question dependencies solely from graded student response data. I will provide a detailed overview of solving SPARFA with maximum likelihood, including a description of the model and algorithms, but for complete details, as well as other SPARFA models and algorithms, see the authors' original paper (Lan et al., 2014)

More formally, their model assumes that questions $(1, \ldots, i, \ldots, Q)$ are related to a relatively small number of underlying concepts $(1, \ldots, k, \ldots, K)$. The goal is to use the responses of students $(1, \ldots, j, \ldots, N)$ to the Q questions—a $Q \times N$ gradebook data matrix—to generate estimates of the three other, much more informative matrices $\mathbf{W}$,$\mathbf{C}$, and $\mathbf{M}$. The first, $\mathbf{W} \in \mathbb{R}^{Q \times K}$ is the *question-concept association* matrix $\mathbf{W}$ whose entries $w_{i,k}$ represent the degree to which question i involves concept k, large positive values indicating a higher degree of involvement. The second, $\mathbf{C} \in \mathbb{R}^{K \times N}$ is the *concept knowledge* matrix $\mathbf{C}$ whose entries $c_{k,j}$ represent the degree to which student j understands concept k, large positive entries indicating a higher degree of understanding. And the third, a conformal $\mathbf{M} \in \mathbb{R}^{Q \times N}$ *intrinsic difficulty* matrix $\mathbf{M}$ with the intrinsic difficulty of question i, μ_i repeated N times along row i. In Figure 2.4 (b), questions and concepts are represented by rectangles and circles, respectively, and the degree to which question i involves concept k, $w_{i,k}$, is represented by the line connecting them.

$$\mathbf{W}_{Q,K} = \begin{pmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,K} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ w_{Q,1} & w_{Q,2} & \cdots & w_{Q,K} \end{pmatrix}, \mathbf{C}_{K,N} = \begin{pmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,N} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ c_{K,1} & c_{K,2} & \cdots & c_{K,N} \end{pmatrix},$$

$$\mathbf{M}_{Q,M} = \begin{pmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_Q \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{pmatrix}^T = \begin{pmatrix} \mu_1 & \mu_1 & \cdots & \mu_1 \\ \mu_2 & \mu_2 & \cdots & \mu_2 \\ \vdots & \vdots & \ddots & \vdots \\ \mu_Q & \mu_Q & \cdots & \mu_Q \end{pmatrix}$$

Notice that each $Z_{i,j} = \hat{\mathbf{w}}_i^T \mathbf{c}_j + \mu_i$, is a factor analysis with sparse vectors which poses an inverse problem. To regularize this problem, prevent overfitting, and improve identifiability of the factor solution, the authors Lan et al. (2014) lay out three fundamental assumptions of SPARFA: (1) the number of underlying concepts K is small compared to the number of students N and the number of questions Q Therefore, $\mathbf{W}$ will have many more rows than columns, and $\mathbf{C}$ will have many more rows than columns. Note that choice of K is important, with smaller values of K extracting just a few broad concepts and larger values of K extracting more fine-grain concepts. (2) Each question will involve only a few of the abstract concepts, rendering matrix $\mathbf{W}$ sparse. And (3), it is assumed that having more knowledge of a concept will not negatively impact a student's probability of answering a question correctly; that is, the question-concept association matrix $\mathbf{W}$ has no negative entries. These assumptions are reasonable in most real-world educational settings and they help alleviate the identifiability issue inherent to factor analysis. Given these constraints (low-dimensionality, sparsity of $\mathbf{W}$ and non-negativity of $\mathbf{W}$), the authors define *SPARse Factor Analysis* (SPARFA) as the estimation of $\mathbf{W}$, $\mathbf{C}$, and $\mathbf{M}$ given observations $\mathbf{Y}$. They have approached the estimation of these matrices in several ways, but in the present study I make use of a matrix factorization method using bi-convex optimization described below.

The probability that the students answer questions correctly is calculated by $\mathbf{WC}+\mathbf{M}$, transformed via a probit or logit link function (Rasmussen & Williams, 2006). Specifically, let matrices $\mathbf{W} \in \mathbb{R}^{Q\times K}$, $\mathbf{C} \in \mathbb{R}^{K\times N}$, and $\mathbf{M} \in \mathbb{R}^{Q\times N}$ be defined as they are above. The authors model the binary-valued graded response (correct = 1, incorrect = 0) variable $Y_{i,j} \in 0, 1$ for student j on question i as

$$Y_{i,j} \sim Ber(\mathbf{\Phi}(Z_{i,j})), (i,j) \in \Omega_{obs} \text{ with } \mathbf{Z} = \mathbf{WC} + \mathbf{M} \tag{2.1}$$

where $Ber(z)$ is the Bernoulli distribution with probability of success z, and $\mathbf{\Phi}(z)$ is the inverse link function that takes the real value z and outputs the success probability of a binary random variable. Thus, $\mathbf{\Phi}(Z_{i,j})$ gives the probability of student j answering question i correctly. In the following discussion, the inverse link function $\mathbf{\Phi}(x)$ will either be the inverse-probit or the inverse-logit function. The inverse-probit function $\mathbf{\Phi}_{pro}(x)$ is essentially the CDF of the standard normal distribution: it gives the area under the curve of a standard normal distribution up to x.

$$\mathbf{\Phi}_{pro}(x) = \int_{-\infty}^{x} N(t|0,1) \text{ d}t = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{t^2/2}$$

The inverse-logit function $\mathbf{\Phi}_{log}(x)$, also called the logistic function, is closely related to the inverse-probit function.

$$\mathbf{\Phi}_{log}(x) = \frac{1}{1+e^{-x}}$$

Both are cumulative density functions which map $(-\infty, \infty) \to [0, 1]$.

2.2.2 Maximum Likelihood for Sparse Factor Analysis

To estimate $\mathbf{W}$, $\mathbf{C}$, and $\mathbf{M}$ given observations $\mathbf{Y}$, it is helpful to partition matrix $\mathbf{C}$ into columns $(\mathbf{c}_1, \dots \mathbf{c}_N)$ so that each column $\mathbf{c}_j \in \mathbb{R}^K$ represents student j's concept knowledge. Likewise, partition $\mathbf{W}$ into rows $(\hat{\mathbf{w}}_1, \dots \hat{\mathbf{w}}_Q)^T$ so that each row $\hat{\mathbf{w}}_i \in \mathbb{R}^K$ represents question i's concept associations.

From 2.1, the likelihood of observing $Y_{i,j} \in 0, 1$ given the question i's concept associations $\hat{\mathbf{w}}_i$ and student j's concept knowledge $\mathbf{c}_j$ is given as

$$\prod_{(i,j)\in\Omega_{obs}} p(Y_{i,j}|\hat{\mathbf{w}}_i, \mathbf{c}_j) = \prod_{(i,j)\in\Omega_{obs}} \mathbf{\Phi}(\hat{\mathbf{w}}_i^T \mathbf{c}_j)^{Y_{i,j}}(1 - \mathbf{\Phi}(\hat{\mathbf{w}}_i^T \mathbf{c}_j))^{1-Y_{i,j}}$$

The goal is to maximize the likelihood of the observed data $Y_{i,j} \in \Omega_{obs}$ over $\mathbf{W}$ and $\mathbf{C}$. The important constraints are that the concept associations vector $\hat{\mathbf{w}}_i$ is sparse and (2) that concept knowledge will not negatively impact the probability of answering a given question correctly, so all entries $W_{i,j}$ must be non-negative. To achieve constraint (2), they authors use the zero-norm of $\hat{\mathbf{w}}_i$, $\|\hat{\mathbf{w}}_i\|_0$ (which has the effect of counting non-zero entries and thus indexing sparsity). They also constrain the lengths of the $\hat{\mathbf{w}}_i$s for their proof of convergence and normalize matrix $\mathbf{C}$ by taking its Frobenius norm to suppress arbitrary scalings between entries in $\mathbf{W}$ and $\mathbf{C}$.

For convenience, the natural logarithm of the likelihood function is maximized, because the logarithm of a function increases monotonically and attains its maximum at the same points as the function it modifies. Thus, the goal of SPARFA can be summarized by the following problem:

$$\max_{\mathbf{W},\mathbf{C}} \log \left(\prod_{(i,j)\in\Omega_{obs}} p(Y_{i,j}|\hat{\mathbf{w}}_i, \mathbf{c}_j) \right) = \max_{\mathbf{W},\mathbf{C}} \sum_{(i,j)\in\Omega_{obs}} \log p(Y_{i,j}|\hat{\mathbf{w}}_i, \mathbf{c}_j) \tag{2.2}$$

$$\text{subject to } \|\hat{\mathbf{w}}_i\|_0 \leq s\forall i,\ \|\hat{\mathbf{w}}_i\|_2 \leq \kappa\forall i,\ W_{i,k} \geq 0\forall i,k,\ \|\mathbf{C}\|_F = \xi. \tag{2.3}$$

To achieve a working maximum-likelihood algorithm with minimal computational complexity, the authors relax the sparsity constraints to the vector 1-norm, the sum of the absolute values of the entries. Lan et al. (2014) then take the constraints and move them into the objective function and restate the problem as minimization, giving

$$\min_{\mathbf{W},\mathbf{C}:W_{i,k}\geq 0\forall i,k} \sum_{(i,j)\in\Omega_{obs}} \text{-log } p(Y_{i,j}|\hat{\mathbf{w}}_i, \mathbf{c}_j) + \lambda \sum_i \|\hat{\mathbf{w}}_i\|_1 + \frac{\mu}{2} \sum_i \|\hat{\mathbf{w}}_i\|_2^2 + \frac{\gamma}{2}\|\mathbf{C}\|_F^2.$$

Here, $\lambda \sum_i \|\hat{\mathbf{w}}_i\|_1$ maintains the sparsity constraint with parameter $\lambda \geq 0$ controlling degree of sparsity, and all other terms and their respective parameters being for regularization of scaling, as before.

In brief, the algorithm the authors develop depends on the nature of the problem being biconvex. Looking at the minimization problem statement above, we can see that the first term, the negative log-likelihood, is convex in $\mathbf{WC}$ for both probit and logit link functions, while the rest of the terms are convex with respect to either $\mathbf{W}$ *or* $\mathbf{C}$. Their SPARFA-M (for maximimum likelihood) algorithm iteratively optimizes the function by alternating between holding $\mathbf{W}$ constant while optimizing $\mathbf{C}$ and holding $\mathbf{C}$ constant while optimizing $\mathbf{W}$. Specifically the two subproblems are as follows:

$$(1)\quad \underset{\hat{\mathbf{w}}_i\,:\,W_{i,k}\geq 0}{\text{minimize}} \sum_{(i,j)\in\Omega_{obs}} \text{-log } p(Y_{i,j}|\hat{\mathbf{w}}_i,\mathbf{c}_j) + \lambda\sum_i \|\hat{\mathbf{w}}_i\|_1 + \frac{\mu}{2}\sum_i \|\hat{\mathbf{w}}_i\|_2^2$$

$$(2)\quad \underset{\hat{\mathbf{c}}_j}{\text{minimize}} \sum_{(i,j)\in\Omega_{obs}} \text{-log } p(Y_{i,j}|\hat{\mathbf{w}}_i,\mathbf{c}_j) + \frac{\gamma}{2}\|\mathbf{C}\|_F^2$$

The authors solve these optimization subproblems in using the FISTA framework, an iterative method that breaks each objective function into two functions, at least one of which is continuously differentiable. Each iteration uses a gradient descent step for the smooth part and a shrinkage step (non-negative soft-thesholding) for the potentially unsmooth part of the objective function (Beck & Teboulle, 2009). For algorithmic details, proofs, and convergence analysis, see Lan et al. (2014). In the present study, the algorithm suite was implementing using Python.

2.2.3 Item Response Theory

Item response theory (IRT) describes a group of latent variable techniques that have been developed specifically to model the interaction between participants' ability (or other latent traits) and the characteristics of test items (indexed by parameters like difficulty, discrimination, and guessing; see Reckase, 2009 for overview). IRT can be used to obtain estimates of these item parameters as well as estimates of latent ability for individual participants. Indeed, these models were originally developed to model how latent ability θ was related to answering a test item (1=correct, 0=incorrect), given item parameters such as difficulty d. IRT models are usually based on the logistic function $\mathbf{\Phi}_{log}(x) = \frac{1}{1+e^{-x}}$, described above as the inverse-logit link function (note again that its domain is $\mathbb{R}$ while its range is 0 to 1). Item parameters are added to the logistic function and change its shape accordingly: parameter b shifts the horizontal scale, while parameter a stretches the vertical scale. A pseudo-guessing parameter c will compress the vertical scale from (0,1) to (c,1), thus acting as an asymptotic minimum.

These parameters are immediately interpretable: b represents the ability level at which there is a 50% chance of answering the question correctly, thus indexing item difficulty. Parameter a is the maximum slope of the curve, or the slope of the line tangent to the point on the curve where $\theta = b$; it indexes how much the probability of a correct answer increases as ability level increases, or equivalently, how well performance on the item discriminates between ability levels. Finally, c is used to account for guessing by raising the lower bound to the probability of a correct response from 0 to the probability of a correct response due to chance. For example, if the item is a five-option multiple choice test, then guessing randomly results in a 1/5 chance of getting the item correct. Thus, even students with very low ability still have at least a 20% chance of responding correctly, so $c = 0.2$. IRT models that estimate only the single parameter b are called 1-parameter logistic (1PL) while models that estimate parameters b, a or b, a, c are called 2-parameter logistic (2PL) and 3-parameter logistic (3PL) models, respectively. The 1PL, 2PL, and 3PL models are shown below (Equations 2.4, 2.5, and 2.6 respectively). Notice that the 3PL reduces to the 2PL when $c = 0$, which itself reduces to the 1PL when

$a = 1$. Indeed, more general cases such as the 4PL do exist, but we will not go into them here. For item i and person j,

$$\begin{aligned} P(y=1|\theta_j, b_i) &= \frac{1}{1+e^{-(\theta_j - b_i)}} && (2.4) \\ P(y=1|\theta_j, a_i, b_i) &= \frac{1}{1+e^{-a_i(\theta_j - b_i)}} && (2.5) \\ P(y=1|\theta_j, a_i, b_i, c) &= c + \frac{1-c}{1+e^{-a_i(\theta_j - b_i)}} && (2.6) \end{aligned}$$

As functions of θ, these functions are often referred to as item-response functions or item characteristic curves (ICCs).

As with SPARFA, IRT uses the responses of a set of people to a set of items (scored correct=1, incorrect=0) as its input information source. Estimating the person and item parameters entails some strong assumptions, including the assumption of a unidimensional latent trait θ and that item responses are uncorrelated after controlling for this latent trait (local independence).

Multidimentional IRT (MIRT)

For the sake of simplicity, we will only be dealing with the 2PL model in this report. Let $P(\theta_j, \phi_i) = P(y = 1|\theta_j, a_i, b_i) = (1 + e^{-a_i(\theta_j - b_i)})^{-1}$ as before. Note that we are positing and estimating a single underlying trait θ_j per student and single discrimination parameter a_i per item. But what if there are multiple traits which additively determine a students performance on question i? We can simply replace the single values θ_j and a_i with vectors

$\theta_{\mathbf{i}} = \theta_{i,1}, \theta_{i,2}, ..., \theta_{i,K}$ and $\mathbf{a_i} = a_{i,1}, a_{i,2}, ..., a_{i,K}$, giving us

$$P(\theta_{\mathbf{j}}, \phi_{\mathbf{i}}) = P(y = 1|\theta_{\mathbf{j}}, \mathbf{a_i}, b_i) = (1 + e^{-\mathbf{a_i}^T(\theta_{\mathbf{j}} - Ib_i)})^{-1}$$

Where I is a vector of ones of length N. Thus, this is the probability that student i responds correctly to question j given their $\theta_{\mathbf{j}}$, and so $1 - P(\theta_j, \phi_i)$ is the probability that student i answers question j incorrectly.

Estimating IRT/MIRT parameters

Proceeding with θ and a without loss of generality, IRT/MIRT models the observed response variable $Y_{i,j} \in 0, 1$ (the score of person j on item i) as

$$Y_{i,j} \sim Ber(P(\theta_j, \phi_i)), (i, j) \in \Omega_{obs} \tag{2.7}$$

For clarity, note that this means each person-item combination is a different Bernoulli random variable. Let the set of all j people's responses to all i items, $Y_{i,j} \forall (i, j) \in \Omega_{obs}$, be the $M \times N$ matrix $\mathbf{Y}$. Further, let the set of all ability parameters θ_i and the set of all item parameters ϕ_j be vectors $\mathbf{\Theta}$ of length M and $\mathbf{\Phi}$ of length N respectively, where M is the total number of items and N is the total number of participants as before. Then the value of the parameters a_i and b_i for each item (i.e., $\mathbf{\Phi}$) are chosen so as to maximize the probability of observing all $Y_{i,j} \in \Omega_{obs}$. That is,

$$\max_{\mathbf{\Phi}} P(\mathbf{Y}|\theta, \mathbf{\Phi}) = \max_{\phi = (\mathbf{a_i}, \mathbf{b_i})} \prod_{(i,j) \in \Omega_{obs}} P(\theta_j, \phi_i)^{Y_{i,j}} (1 - P(\theta_j, \phi_i))^{1 - Y_{i,j}}$$

Which is equivalent to

$$\max_{\mathbf{\Phi}} \log\left(P(\mathbf{Y}|\theta, \mathbf{\Phi})\right) = \max_{\phi_{\mathbf{i}}=(\mathbf{a_i},\mathbf{b_i})} \log\left(\prod_{(i,j)\in\Omega_{obs}} P(\theta_j, \phi_i)^{Y_{i,j}}(1-P(\theta_j,\phi_i))^{1-Y_{i,j}}\right) \tag{2.8}$$

$$= \max_{\phi_i} \sum_{(i,j)\in\Omega_{obs}} (Y_{i,j}\log P(\theta_j, \phi_i) + (1-Y_{i,j})\log(1-P(\theta_j,\phi_i)) \tag{2.9}$$

This second function is the log-likelihood function which we maximize to determine the parameters of interest. We have $\mathbf{Y}$, and if we knew $\mathbf{\Theta}$ we could simply take the partial derivatives of the log-likelihood with respect to a_i and b_i for each item, set each equal to zero, and solve the resulting system of equations (checking to be sure we have found maxima). But unfortunately we do not; both the latent trait values and the item parameters must be estimated from the observed pattern of responses. One way around this is to assume some distribution $g(\theta)$ for θ and then integrate it out, resulting in what's know as the "mariginal likelihood" in a Bayesian framework. That is,

$$P(\mathbf{Y}|\mathbf{\Phi}) = \int_\theta P(\mathbf{Y}|\theta, \mathbf{\Phi})g(\theta)\mathrm{d}\theta$$

By maximizing this marginal likelihood we get our parameter estimates, but as the number of items grows this maximization can get ugly.

One improvement on this idea makes use of the expectation-maximization (EM) algorithm, an iterative method for finding maximum likelihood estimates of parameters of models in which there are latent variables. As its name suggests, the algorithm alternates between two calculation steps: (1) the expected values of (E) of the log-likelihood at the current parameter estimate, and the maximization (M) of this expected log-likelihood, yielding new parameter estimates which are used to determine the distribution of the latent variables in the subsequent expectation step. Here, the trick is to consider θ a latent variable. Thus, the likelihood function is now $\log P(\mathbf{Y}, \theta | \mathbf{\Phi}^{(t)})$. Nothing has changed except our interpretation of θ, which is now considered to be unobserved, latent data. The EM algorithm alternates back and forth between (1) calculating the expected value of the log-likelihood with respect to θ (given the observed data $\mathbf{Y}$ and the current parameter estimate $\mathbf{\Phi}^{(t)}$) giving us a guess at a probability distribution over θ, and (2) taking the result and finding the parameters $\mathbf{\Phi}$ that maximize it, thus becoming the new parameter estimates $\mathbf{\Phi}^{(t+1)}$. In steps,

- Initialize an estimate/guess for the parameters $\mathbf{\Phi}^{(t)}$
- E Step: compute $E_{\theta|\mathbf{Y},\mathbf{\Phi}^{(t)}} \log P(\theta, \mathbf{Y} | \mathbf{\Phi}^{(t)})$
- M Step: pick $\mathbf{\Phi}^{(t+1)}$ to be the $\mathbf{\Phi}$ that maximize the expectation
- Continue with E Step until stopping criterion is met or the algorithm fails to converge

Another way to think of this is that each expectation step results in a posterior likelihood, which is then maximized, etc. The EM algorithm is the default method used in most software packages for IRT parameter estimation. In what follows, I make use of the R package `mirt` which defaults to EM estimation but supports other estimation techniques including Metropolis-Hastings in conjunction with standard numerical optimization. For the higher-dimensional models, I use the "quasi-monte carlo" version of the EM algorithm as recommended by the package developer.

2.2.4 Comparing SPARFA and MIRT

From the PSY 301 data I have created a subset consisting of all student responses to items on their first presentation (i.e., no repeated items). I will compare SPARFA to MIRT by fitting each to this data given a different number of underlying traits/concepts K and comparing their parameter estimates predicted probabilites. In the original SPARFA paper, the authors acknowledge similarity to MIRT models but note that "the design of these algorithms leads to poor interpretability of the resulting parameter estimates" (Lan et al., 2014, p. 34). Assessing this claim and comparing the two models is one goal of the present report. As described above, the SPARFA algorithm was written in Python and implemented using iPython through a command-line interface. MIRT was performed using the `mirt` package in R (Chalmers, 2012), implemented through the R-studio graphical user interface.

2.3 Content-level Methods: Chats & Free-Response

Chats

Students participated in 25 small-group online chats during class throughout the semester, each consisting of 2-5 participants and each lasting 5-10 minutes in duration. Several chats were about material that was previously featured on that day's benchmark quiz. I will examine the overall relationship between the total number of chat contributions across all chats and overall score across all 27 benchmarks (irrespective of content overlap). I will then examine the relationship between performance specific benchmark quizzes and subsequent chats about relevant material.

Text Mining Student Free-Response

Approximately 15% of students' grades came from their performance on 4 free-response writing exercises. In these assignments, students were required to write a narrative about an ambiguous picture (Thematic Apperception Test activity), to describe a recent dream in detail, or to type in a stream-of-consciousness fashion for 20 minutes, the latter task being repeated twice in the semester. One of the instructors of the course whence this data comes is a leading expert on text analysis and has even invented his own software package to perform such tasks (LIWC; Pennebaker & Francis, 1999). Accordingly, there is very little I could do with students' text output that hasn't already been thought of by him and his team of researchers. They have used things like the Categorical Dynamic Thinking Index (CDI; Pennebaker et al., 2014), Language Style Matching (LSM), etc. However, I wanted to explore these techniques a bit myself in this report using a lexical measure that their team has never examined: idea density. Idea density (also called proposition density) is "the number of propositions divided by the number of words," where a proposition is anything in speech that can be true or false. So, for example, “the quick brown fox jumps over the lazy dog” has five propositions: (1) quick, (2) brown, (3) jumped, (4) over, (5) lazy. Thus, propositions correspond roughly to verbs, adjectives, prepositions, and subordinating conjunctions. The theory is that each proposition entails a certain amount of mental processing effort, and high idea density—by packing in the propositions—makes for slower processing by increasing the amount of work the reader must do to understand the text. Covington (2008) showed that “popular” texts (e.g., magazines, pulp fiction) and “introductory” texts (addressed to serious nonspecialists) always have an idea density below 0.5, while technical documents are always above 0.5.

I was introduced to this construct by the “nun study”, an influential research project done to assess whether linguistic ability in early life was associated with cognitive impairments in old age and the development of Alzheimer’s disease (Snowdon et al, 1996). These authors measured both the idea density and the grammatical complexity of autobiographical essays that 93 nuns had written as an entrance requirement upon first taking their monastic vows at a mean age of 22 years. They found that both measures were associated with low cognitive performance in late life, but that idea density was a stronger and more consistent predictor. Most strikingly, Alzheimer’s disease was present in *all* of the nuns whose essays exhibited low idea density in early life and *none* of those with high idea density.

Though this college sample is comparable in age range, the text submissions differ in important ways from an autobiographical essay. I chose to apply measures of idea density to the two stream-of-consciousness exercises and to see if idea density scores were predictive of students total grade across all of the BM quizzes. This was purely exploratory, though I hypothesized that idea density in a stream-of-consciousness writing exercise would be associated with higher grades on the BM quizzes, perhaps due to individual differences in text processing for the class reading assignments upon which many BM questions were based.

I used the program CPIDR (version 5.1) to calculate idea density; it uses a part-of-speech tagger to count true propositions in text, and it achieves very high interrater reliability with human raters (Brown et al., 2008). CPIDR was developed by Michael Covington at the University of Georgia and the software is free for non-commercial use. I then regressed total BM grade (proportion of BM items answered correctly) on idea density score for the stream-of-consciousness assignment. I used several regression techniques and I also examined the correlation between idea density on the first and second of these assignments.

Chapter 3

Results and Discussion

3.1 Post-Processing and Analysis of SPARFA Output

3.1.1 Tagging Items

After running SPARFA on the student-item response data $\mathbf{Y}$, we have estimated the *question-concept association* matrix $\mathbf{W}$ (with $w_{i,k}$ representing the degree to which question i involves concept k), the *concept knowledge* matrix $\mathbf{C}$ (with $c_{k,j}$ representing the degree to which student j understands concept k), and the *intrinsic difficulty* matrix $\mathbf{M}$, with the estimated difficulty of item i repeated along row i.

The next question of interest becomes how to interpret the latent concepts given by $w_{i,k}$ and $c_{k,j}$. One fruitful way of doing this incorporates user-generated "tags" that describe or classify each item a priori. Given a set of tags for our items, we then need to find the association between the latent concepts and the tags, so that the latent concepts are more readily interpretable.

Though the syllabus for this course was not available for the year 2013, I was able to use the syllabi from 2012 and 2014 to triangulate in order to find the sequence of broad sections covered in the course ($n = 8$) as well as the daily lecture topic as defined by the instructors ($n = 27$). Furthermore, I individually considered each item and its associated answer choices to develop my own finer-grain tags for all 540 unique items, resulting in 110 such tags. See Appendix B for sample items and their tags, topics, and sections.

Tag algorithm (not used)

In general, let M be the number of tags associated with the Q items. To incorporate tag information, we create a tag matrix $\mathbf{T} \in \mathbb{R}^{Q \times M}$ of zeroes and ones, where each column of $\mathbf{T}$ is associated with a tag, and each entry $T_{i,m} = 1$ if the tag m is associated with question i and $T_{i,m} = 0$ otherwise. Lan et al. (2014) show that the item-concept association matrix $\mathbf{W}$ can itself be factored into $\mathbf{TA}$, where $\mathbf{A}$ is an $M \times K$ matrix representing tag-to-concept mapping. The same non-negativity and sparsity assumptions apply in this second factorization as they did to the first.

To estimate the tag-concept association matrix, as well as student-tag knowledge. The authors use a 1-norm regularized least-squares minimization method to obtain each column of $\mathbf{A}$, where η controls the sparsity level. The FISTA framework described above is then used to solve for $\mathbf{A}$ as desired.

$$\min_{\mathbf{a}_k} \ \frac{1}{2}\|\mathbf{w}_k - \mathbf{T}\mathbf{a}_k\|_2^2 + \eta\|\mathbf{a}_k\|_1$$

The resulting matrix $\mathbf{A}$ contains the tag-concept associations; normalizing the entries of each column so they sum to one lets us interpret the entries as the proportion of each tag m contributing to a given concept k. To calculate each student's knowledge of each tag, we simply multiply the the *concept* *knowledge* matrix $\mathbf{C}$ by the tag-concept matrix $\mathbf{A}$ to get $\mathbf{U} = \mathbf{AC}$ Since $\hat{\mathbf{a}}_m^T$ contains the contributions of tag m to all concepts k, and $\mathbf{c}_j$ contains the concept knowledge for all concepts k for student j, $U_{m,j} = \hat{\mathbf{a}}_m^T \mathbf{c}_j$ represents the knowledge of student j of tag m.

Unfortunately, I could not get the tagging functionality to work properly. In the interest of time, I decided to stick with interpreting the question-concept associations given as entries in matrix W instead.

3.2 Analysis of MIRT Output

3.2.1 Model Fit

Five 2PL IRT models with $\theta_{\mathbf{j}}$ of different dimension were fit to the full-student dataset. These 1-, 2-, 3-, 8-, and 27-dimensional models were identical except for the number of θ_js estimated. Likelihood ratio tests were all significant when comparing a model with fewer parameters to a model with more parameters, indicating that additional θ parameters result in better fit, even up to $K = 27$. However, using a penalized likelihood ratio criterion tempers this interpretation. For $K = 1$ *vs* $K = 2$, the Δ_{AIC} ($AIC_{full} - AIC_{constrained}$) was -265.9 while the Δ_{BIC} ($BIC_{full} - BIC_{constrained}$) was 2260.5; For $K = 2$ *vs* $K = 3$, the Δ_{AIC} was 133.2 while the Δ_{BIC} was 2654.7; For $K = 3$ *vs* $K = 8$, the Δ_{AIC} was 200.6 while the Δ_{BIC} was 12738.2; For $K = 8$ *vs* $K = 27$, the Δ_{AIC} was 7226.2 while the Δ_{BIC} was 53789.4. The only equivocal evidence that a model with $K > 1$ is appropriate comes from the initial comparison of AIC for $K = 1$ *vs* $K = 2$; we will proceed with the unidimensional model on the full data for the remainder of the IRT discussion, but we will return to higher-dimensional models when comparing IRT to SPARFA in the next section.

The best fit by all accounts was obtained by fitting a unidimensional model to the dataset consisting of item responses for students who completed the course only ($n = 677$). This model fit better than the unidimensional with full-student data: $\Delta_{AIC} = 16603.6$, $\Delta_{BIC} = 16850$. It also fit better than the 8-dimensional model on the constrained data ($\Delta_{AIC} = 114.5$, $\Delta_{BIC} = 16843.5$), which in turn fit better than the 27-dimensional model on the constrained data ($\Delta_{AIC} = 6058.7$, $\Delta_{BIC} = 50350$). Thus, the unidimensional model provides the best fit with the constrained data (correcting for overfitting). We will return to higher-dimensional models on this constrained data-set when we compare IRT to SPARFA in the next section.

3.2.2 Parameter Estimates

The test characteristic curve plotted in Figure 3.1 gives the expected total score across all benchmark quizzes as a function of θ; this can be thought of as adding together all of the 540 unique item characteristic curves (ICCs) or, equivalently, as adding up the probability of a correct response on each item for each θ). Recall that each student only received 187 of these 540 items; thus, this curve represents the expected total score on the entire item pool as a function of θ, or what would be expected if every student had answered every question.

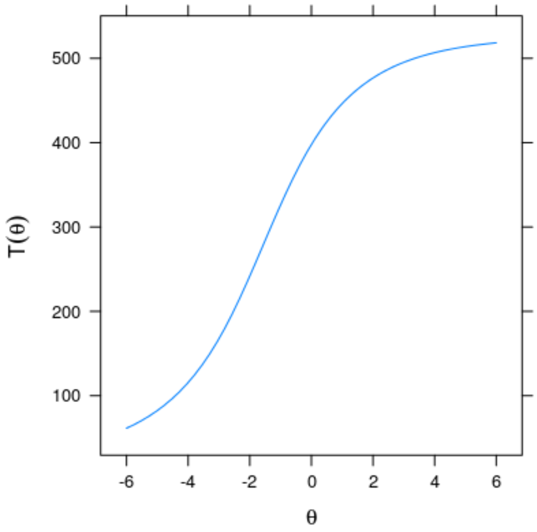
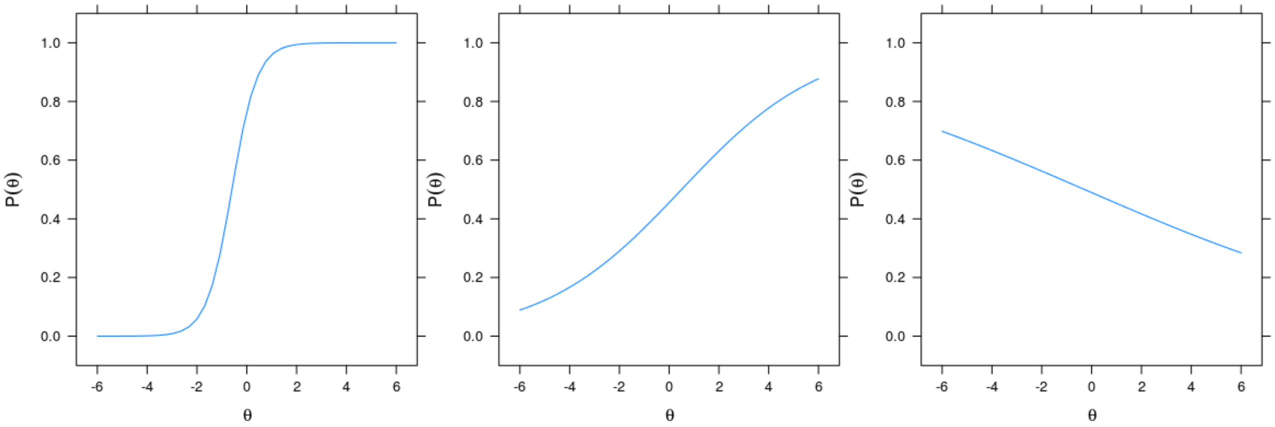


Figure 3.1: Test characteristic curve combining all 540 items over all 27 benchmark quizzes to give a predicted total score if a student were to answer all items.



(a) Good Item

(b) Bad Item

(c) Ugly Item

Figure 3.2: Good, Bad, and Ugly ICCs corresponding to items 206, 169, and 327 respectively for the full-student data ($n = 838$). See Appendix A for ICCs and item fit statistics.

Item characteristic curves (ICCs) for all 540 items are pictured thumbnail-size in Appendix A for the full student data. The three ICCs pictured in Figure 3.2, the "good" item had $a = 1.983$, $d = 1.18$, the "bad" item had $a = 0.358$, $d = -0.179$, and the "ugly" item had $a = -0.147$, $d = -0.042$. Indeed, six of the 540 items had negative discrimination parameters. For these items, students with higher θ scores had a lower probability of answering the item correctly; such items should be checked for accidental miscoding and eliminated from the item pool. Notice that the bad item has a relatively flat slope parameter a; thus, the probability of answering that item correctly does not go up very much as θ increases, and thus whether or not the student gets the item correct does not give strong evidence about their θ score.

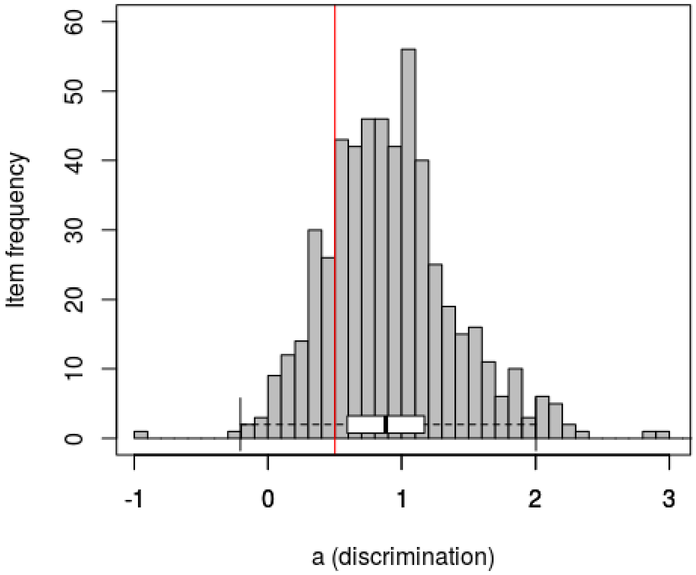


Figure 3.3: Histogram and boxplot of item discrimination parameters for all 540 items. Items with low discrimination indicated by cutoff at $a = 0.5$

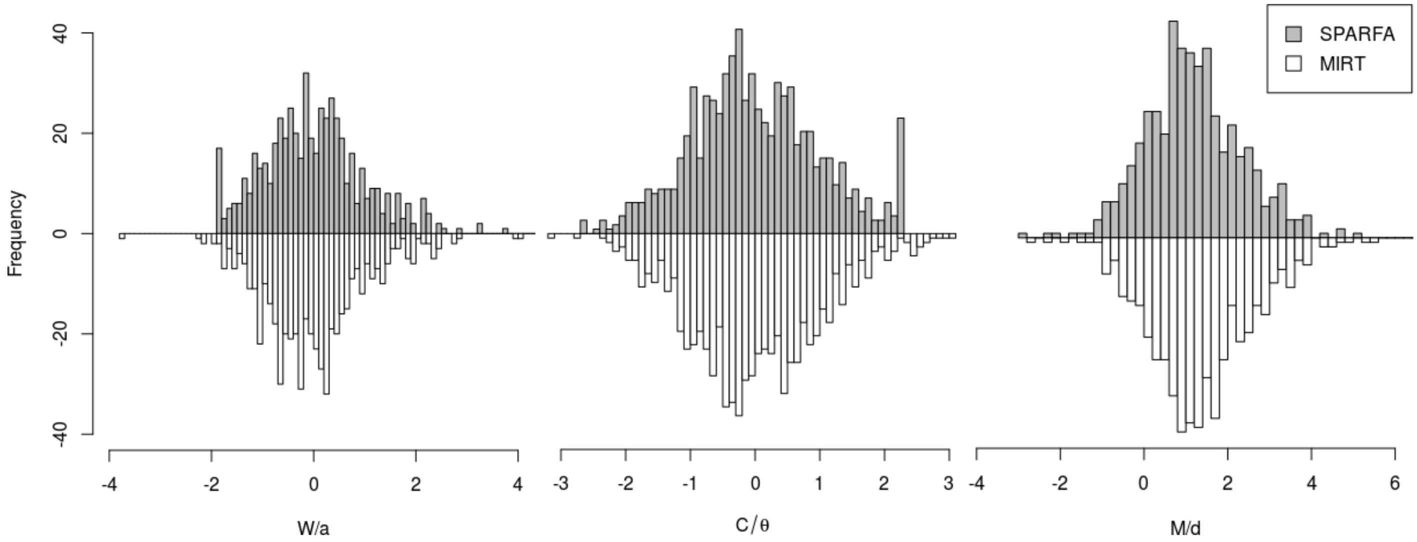
As shown in Figure 3.3, the mean discrimination for the items was 0.920, the median was 0.882, and the first and third quartiles were 0.593 and 1.168 respectively. Of all the items, 97 had discrimination parameters between 0 and 0.5 and 27 items had discriminations between 0 and 0.25. This sort of analysis could be used to alert instructors to problems with their items so that they can be rewritten or discarded.

3.3 Comparison of SPARFA & IRT/MIRT

SPARFA and IRT/MIRT were run on two datasets: the full student data ($n = 848$) and the data consisting only of students who finished the course ($n = 677$). These data did not include repeated items, only items given to each student for the first time. Estimated question-concept associations (SPARFA's $\mathbf{W}$ matrix), concept-knowledge associations (SPARFA's $\mathbf{C}$ matrix), and intrinsic difficulty (SPARFA's $\mathbf{M}$ matrix) were obtained by running SPARFA with the number of concepts $K = 1, 2, 3, 8,$ and 27. The same procedure was repeated using the `mirt` package in R, where K is the number of dimensions modelled. Thus, for each student j there were $\theta_{j,1}, \theta_{j,2}, ..., \theta_{j,K}$ latent trait scores, and for each item i there were $a_{i,1}, a_{i,2}, ...a_{i,K}$ discrimination parameters estimated. A given item's $a_{i,k}$ represents how well that item discriminates among students with different scores on θ_k.

It is instructive to compare the SPARFA likelihood (Eq. 2.1) and the MIRT likelihood (Eq. 2.7) explicitly. Note that the MIRT equation $p(Y_{i,j}|\theta_{\mathbf{j}}, \mathbf{a_i}, b_i) = (1+e^{-\mathbf{a_i}^T(\theta_{\mathbf{j}}-Ib_i)})^{-1}$ can be reparameterized as $(1+e^{-(\mathbf{a_i}^T\theta_{\mathbf{j}}+d_i)})^{-1}$, where $d_i = -b_i \times a_i$. Here we can see that $\mathbf{a_i}^T\theta_\mathbf{j}$ is analogous to $\hat{\mathbf{w}}_i^T\mathbf{c}_j$, so estimates of $a_{i,k}$ should be similar to the elements of $\mathbf{W}$, $w_{i,k}$, which represent the strength of association between item i and concept k. MIRT and SPARFA are at heart quite similar, representing observations as linear combinations of latent factors; the main differences are in (1) the assumptions, and (2) the optimization algorithm. To be able to handle sparse data, SPARFA assumes that the number of concepts is small and that W is sparse and non-negative; these constraints limit the quality of our comparisons. Furthermore, SPARFA breaks log-likelihood into sub-problems with terms regulated by sparsity parameters before optimizing with the FISTA algorithm, while MIRT uses an EM algorithm to maximize the likelihood function. In essence, both are performing factor analyses using slightly different algorithms and under different assumptions.

Picking 1, 8, and 27 for the number of concepts/dimensions/latent traits was pre-specified because there were 8 sections in the course and 27 weekly quizzes with minimal content overlap. Picking 2 and 3 for the number of concepts/dimensions was done to better assess the divergence between the estimates generated by SPARFA and IRT.

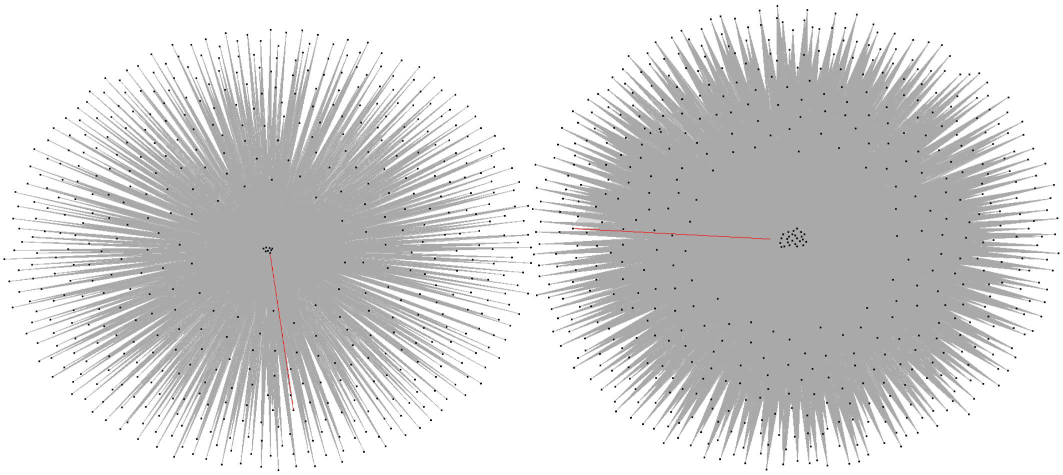


(a) $w_{i,1}$s (top), a_is (bottom) (b) $c_{1,j}$s (top), θ_js (bottom) (c) μ_is (top), d_is, (bottom)

Figure 3.4: Back-to-back histograms of SPARFA/MIRT parameter estimates. Y-axis scale is the same for each plot (note different x-axes).

For the full-student data ($n = 838$), running SPARFA and IRT with 1 concept resulted in a concept-knowledge matrix **C** that correlated $r = .9834$ with IRT theta estimates $\theta_{\mathbf{j}}$. SPARFA question-concept association matrix **W** correlated $r = .9536$ with IRT discrimination parameter estimates $\mathbf{a_i}$. SPARFA difficulty estimates correlated $r = .9959$ with IRT difficulty parameter estimates. The predicted probability of a correct answer for student i on question j for both SPARFA and IRT correlated $r = .9927$. See histograms in Figure 3.5 for a visual comparison of unidimensional SPARFA and IRT "parameters".

Running SPARFA and MIRT with 8 concepts on the full-student data yielded a concept-knowledge matrix **C** that correlated $r = .0293$ with MIRT theta estimates $\theta_{\mathbf{j}}$. SPARFA question-concept association matrix $\mathbf{W}$ correlated $r = .0101$ with MIRT discrimination parameter estimates $\mathbf{a_i}$. SPARFA difficulty estimates correlated $r = .4442$ with MIRT difficulty estimates. The predicted probability of a correct answer for student i on question j for both SPARFA and MIRT correlated $r = .4712$.



(a) Question-concept graph ($K = 8$)

(b) Question-concept graph ($K = 27$)

Figure 3.5: Graphical representation of question-concept associations $w_{i,k}$ for full data (540 questions) with $K = 8$ (a) and $K = 27$ (b) concepts.

Running SPARFA and MIRT with 27 concepts on the full-student data yielded a concept-knowledge matrix $\mathbf{C}$ that correlated $r = -.0162$ with MIRT theta estimates $\theta_{\mathbf{j}}$. SPARFA question-concept association matrix $\mathbf{W}$ correlated $r = .0163$ with MIRT discrimination parameter estimates $\mathbf{a_i}$. SPARFA difficulty estimates correlated $r = .4602$ with MIRT difficulty estimates. The predicted probability of a correct answer for student i on question j for both SPARFA and MIRT correlated $r = .1176$.

For completeness, SPARFA and MIRT were run with $K = 2$ and $K = 3$ so that the parameter and probability correlations could be examined more closely. With $K = 2$ concepts on the full-student data, the correlation between entries of $\mathbf{C}$ and estimated $\theta_{\mathbf{j}}$ was $r = 0.2212$, the correlation between $\mathbf{W}$ and estimated $\mathbf{a_i}$ was $r = .1768$, and the correlation between SPARFA and MIRT difficulty estimates was $r = .9387$. Moreover, with $K = 2$ the correlation between predicted probability of a correct response was $r = .8732$. With $K = 3$ concepts, the correlation between entries of $\mathbf{C}$ and estimated $\theta_{\mathbf{j}}$ was $r = -0.0028$, the correlation between $\mathbf{W}$ and estimated $\mathbf{a_i}$ was $r = .0787$, the correlation between SPARFA and MIRT difficulty estimates was $r = .7971$, and the correlation between predicted probability of a correct response was $r = .7851$. The above analyses were all re-run using only data from students who had completed the course ($n = 677$) to observe how this chance would affect the estimates. All correlations are reported in Table 3.1.

It was speculated that the lack of parameter estimate correlations with increasing K was due both to SPARFA's sparsity constraints on the $\mathbf{W}$ and $\mathbf{C}$ matrices (see Section 2.2.1) and to the extraction of non-existent factors. Determination of the "correct" number of factors to use for adequate representation of our data's intercorrelations required that an ordinary exploratory factor analysis be performed; however, the sparse response matrix Y would preclude such analysis (hence the development of SPARFA). One way around this problem is to impute the missing data. To this end and assuming pairwise independence the R package `missForest` was used, which iteratively fits a ran-

Table 3.1: SPARFA/MIRT estimate correlations, computed by stacking all estimates into vectors and computing Pearson's r on them. Note that W/a and C/θ correlations drop off steeply as K increases, while probability correct and difficulty correlations decline more gradually.

		SPARFA/MIRT Correlations			
		$p(y = 1\|...)$	W/a	C/θ	M/d
N=838	K=1	.993	.954	.983	.996
	K=2	.873	.177	.221	.939
	K=3	.785	.079	.003	.797
	K=8	.471	.010	.029	.444
	K=27	.118	.016	-.016	.460
N=677	K=1	.997	.844	.997	.972
	K=8	.451	.017	.118	.522
	K=27	.133	.007	.003	.494

dom forest on the observed data to predict the missing part. After imputation, factor analysis was performed using principal axis factoring without rotation; using parallel analysis and very-simple-structure criteria, a 3-factor solution was found to be best. Using less robust extraction criteria like Kaiser's rule (all eigenvalues > 1) results in a 2-factor solution, while examination of the scree plot results in a 3-factor solution. Regardless, it is clear that both 8 and 27 are too many, yielding noisy factors that therefore fail to correlate.

3.3.1 Sparfa Proof of Concept

SPARFA was run on the first week's scored benchmark quiz (BM 1) data which consisting of eight questions; I chose 4 concepts (k=4) because I had assigned 4 unique tags to these questions a priori (see Appendix B). This analysis was conducted with full student data ($n = 838$); below are presented the 8 × 4 question-concept association matrix **W**, a portion of the 838 × 4 student concept-student matrix **C**, and the 8 intrinsic difficulty estimates **M**. We can visualize the question-concept association matrix **W** from Table 3.4 using a graph with edge weights corresponding to the strength of the question-concept association (Figure 3.6).

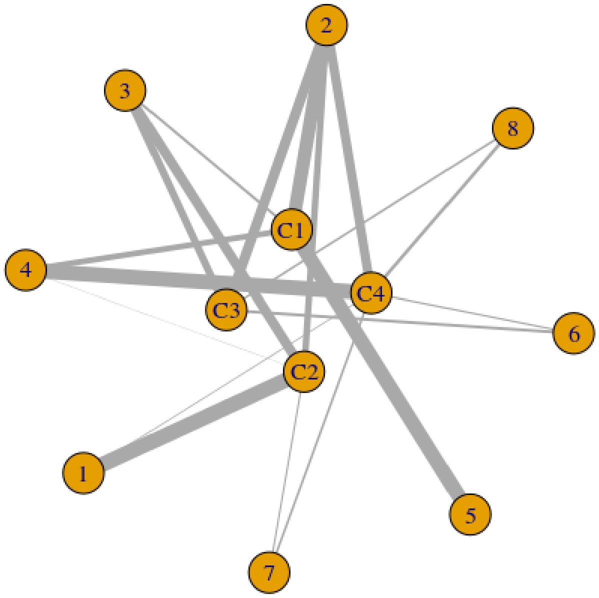


Figure 3.6: Graphical representation of question-concept associations. Nodes C1 through C4 represent Concepts 1 through 4 respectively (see Table 3.4). Nodes 1 through 8 represent Items 1 through 8 respectively. Edge thickness between item i and concept k is proportional to $w_{i,k}$

Looking at Table 3.5, we see the estimated concept knowledge for 12 of the 838 students; on the far left of the table, we have the average concept knowledge for all students. Thus, instead of giving students a question they previously missed at random, SPARFA's **C** matrix lets you know which concepts most urgently need remediation for a given student. For example, we see

Table 3.2: Benchmark 1 tags and questions

	Tag	Question Prompt
1	needs	"...According to the principles of Maslow's heirarchy of needs..."
2	psych perspectives	"...What kind of psychologsts are they most likely to be?"
3	class protocol	"If you do not participate in... what will happen?"
4	psych perspectives	"...How would a developmental psychologist explain this behavior?"
5	psych perspectives	"...Neither is wrong; they're just using different:"
6	psych perspectives	"...Her therapist is taking which of the following psychological perspectives?"
7	conditioning	"Which of the following is the best example of the classical conditioning?
8	class protocol	"Which of these is NEVER permitted when you take a benchmark?

Table 3.3: Student responses to BM 1 items (first 12 of 838)

S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	...
0	1	1	1	1	1	1	1	1	1	0	1	...
1	1	1	1	1	1	1	0	0	1	0	1	...
1	1	1	1	1	1	1	1	1	1	0	1	...
1	1	1	1	1	1	1	1	1	1	0	1	...
0	1	1	0	0	0	1	0	0	1	1	1	...
1	1	1	0	0	0	1	1	1	1	0	0	...
1	1	1	1	1	1	1	1	1	1	1	1	...
1	1	1	1	1	1	1	1	1	1	1	1	...

in Table 3.3 that student S1 missed the first question and the fifth question. The first question is highly associated with Concept 2, while the fifth question is highly associated with Concept 1 (see Table 3.3). The student has missed two questions; given the chance to repeat only one of these questions on the next quiz, which should be repeated? Table 3.5 (top) tells us that student S1's Concept 1 knowledge estimate is low (-1.77) and their Concept 2 estimate is low (-1.16), but compared to the mean scores on both concepts, it can be quickly determined that Concept 2 should take priority: the student's z-score for Concept 1 was -0.70 while their z-score for Concept 2 was -1.27 (Table 3.5, bottom), indicating that their understanding of Concept 2 is in near lowest

Table 3.4: Benchmark 1 question-concept (W) matrix; difficulty estimates appended (right)

	Concept 1	Concept 2	Concept 3	Concept 4	Difficulty
Question 1	0	10.13	0	0.73	3.10
Question 2	8.94	4.39	6.77	6.20	-1.16
Question 3	1.80	6.51	4.96	0	7.06
Question 4	3.88	0.24	0	9.95	-0.55
Question 5	9.73	0	0	0	4.16
Question 6	0	0	1.97	1.14	0.94
Question 7	0	1.06	0	1.49	1.08
Question 8	0	0.04	1.44	2.21	8.42

Table 3.5: Benchmark 1 concept-knowledge (C) matrix for the first 12 students; raw (top), standardized (bottom)

	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	**AVG**	**SD**
C1	-1.77	3.00	3.00	-0.77	-0.77	-0.77	3.00	-3.00	-3.00	3.00	-0.06	2.50	-0.31	2.09
C2	-1.16	3.00	3.00	3.00	3.00	3.00	3.00	0.83	0.83	3.00	-0.60	2.53	1.16	1.83
C3	3.00	1.72	1.73	-3.00	-3.00	-3.00	1.71	0.45	0.45	1.71	-2.71	-3.00	0.78	2.08
C4	3.00	3.00	3.00	2.96	2.96	2.96	3.00	2.40	2.40	3.00	-0.17	0.73	0.47	1.85
	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	**AVG**	**SD**
C1	-0.70	1.58	1.58	-0.22	-0.22	-0.22	1.58	-1.29	-1.29	1.58	0.12	1.34	0	1
C2	-1.26	1.00	1.00	1.00	1.00	1.00	1.00	-0.18	-0.18	1.00	-0.96	0.75	0	1
C3	1.07	0.46	0.46	-1.82	-1.82	-1.82	0.45	-0.16	-0.16	0.45	-1.68	-1.82	0	1
C4	1.36	1.36	1.36	1.34	1.34	1.34	1.36	1.04	1.04	1.36	-0.35	0.14	0	1

decile and thus questions tapping this Concept 2 should take priority upon repetition in order to optimize remediation.

As described earlier, the course whence this data originated implemented the selection of repeated questions in an uninformed way by using the following rule: pick a repeat question at random from the set of all questions a student has previously missed at least once. As it happens, question number 5 was randomly chosen to repeat on the next quiz, and this student did not see question 1 again until BM 5.

Another unfortunate side-effect of drawing repeat questions at random from the pool of all items a student has missed at least once is that all items have an equal probability of reappearing, even those that have already reappeared multiple times; no correction is made based on the students performance with the repeated item. To take an extreme example, say a student only misses one question on the first BM quiz, misses none of the second, none on the third, none on the fourth, etc.; even if this student answers the previously missed item correct when it is repeated on quiz 2, it will repeat again on quiz 3, again on quiz 4, and so on. To take an striking example from the data, one student had to repeat the same 10 times throughout the course, despite answering it correctly on each of the ten repeats. Furthermore, even if a given student has missed multiple items, there is still a chance that an item will keep repeating. If a student misses only two items, there's still a $\frac{1}{32}$ chance that they will only see one of those items on the next five benchmark quizzes. Though this random selection procedure is a simple way to implement spaced practice, it is certainly suboptimal.

It is of interest, then, to compare the repeated items that were *actually* given to students on the second benchmark with the items that SPARFA *would have given* (those tapping the concept most in need of remediation for a given student). To achieve this, each concept-knowledge estimate (each entry in the concept-knowledge matrix $\mathbf{C}$) was centered and scaled by row, yielding z-scores for each student for each of the 4 concepts (bottom of Table 3.5). Then, taking the concept with the lowest z-score for each student, we can see whether the repeated question on BM 2 was one that was strongly associated with their weakest concept. We ask whether the repeated question was one that SPARFA found to have significant association with their weakest concept estimate (Table 3.4); performing this comparison for the first repeated items reveals that the random procedure assigned repeated items that tapped a student's weakest concept only 33.2% of the time. Thus, there is definitely something to be gained by using SPARFA to inform decisions about which items to repeat for which students and when.

As discussed in the introduction, introducing variability during retrieval appears to enhance the transfer of learning to novel problems above and beyond the benefits of retrieval practice (e.g., Butler, 2010). Specifically, practicing retrieval with *different* questions that tap the same underlying concept results in an increased ability to apply one's knowledge of the underlying concept to novel questions relative to retrieval practice with the same question. Given these observations, it would be interesting to know how often "repeated" items tap the same underlying concept without being repeated verbatim. In the present course, the number of possible new items for each benchmark was variable: some weeks, there were only 7 new items and thus all students received the same 7 items; other weeks, there were as many as 36 new items, from which 7 were drawn at random to make up 7 of the 8 BM items for each student (Figure 3.7). Using SPARFA's esimates of students' weakest concepts, and given the items students have already seen which tap that concept, then instead of just repeating those items, it may indeed be more beneficial to give

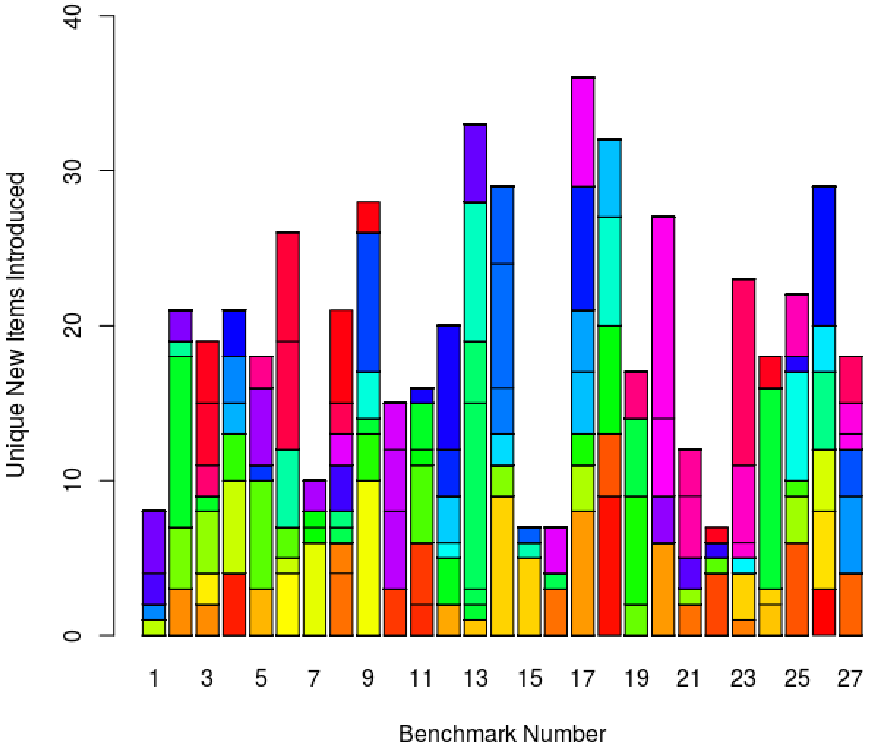


Figure 3.7: Size of new item-pool per benchmark; colors indicate a priori tagged concepts

students a new, previously unseen item tapping that concept. Looking at the figure, we can see the concepts broken down by benchmark (the colors of the stacked bars; items tapping the same concept are all the same color); thus, by capitalizing on the size of each benchmark's item pool, we can use SPARFA to introduce variability into retrieval practice by giving students *new* questions that tap previous concepts. Additionally, if a student is really struggling with a particular concept, item-difficulty estimates can be used to select an easier item tapping that concept to help scaffold their learning (e.g., Murray & Arroyo, 2002).

3.4 Natural Language Processing: Chats and Writing Assignments

As described in the methods section, CPIDR was used to calculate idea density scores for each of the 4 writing assignments completed by each student during the course of the semester. The first was a 20 minute stream-of-consciousness exercise; the second was a projective writing exercise wherein students had to compose a narrative about an ambiguous stimulus picture of two scientists; the third was an exercise where students wrote in detail about a recent dream; the fourth was a second 20 minute stream-of-consciousness exercise. The idea density variables are denoted ID.93, ID.211, ID.232, and ID.442 for the $1^{st}, 2^{nd}, 3^{rd}$ and 4^{th} writing assignments, respectively. The least inclusive data subset was used for these analyses ($n = 677$) in order to restrict our subset to data from students who completed the entire course (who therefore have a predicted course grade of more than 60 and who generated data for most of the writing exercises; see pages 21-22 for inclusion criteria). The figure below shows pairwise scatterplots for the benchmark grade ("Grade") and idea density scores for each assignment.

First, I was curious whether idea density differed across the different writing tasks. For this step, I further subsetted the data to include only those who had completed all 4 writing assignments ($n = 425$). Idea density

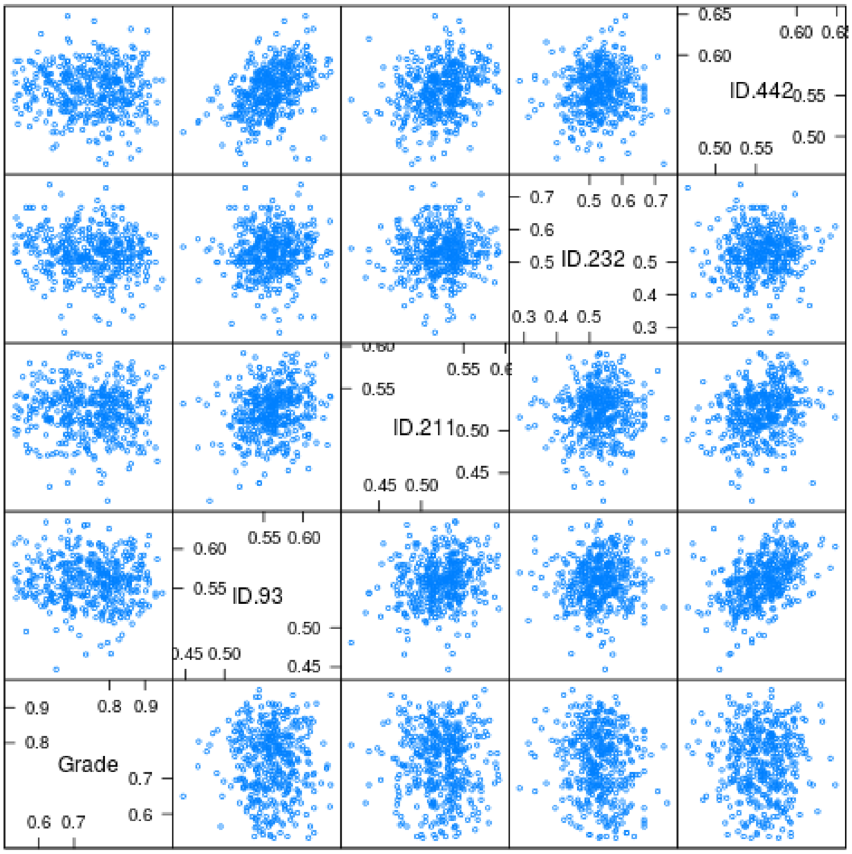


Figure 3.8: Pairwise scatterplots across assignments

scores were calculated for all assignments; I skipped the omnibus F-test and performed all pairwise paired t-tests, adjusting the p-value with the Bonferroni correction. The α-corrected p-values and associated effect sizes for each test are given in Table 3.6 below; we can see that idea density scores were significantly different across all writing exercises except with Thematic Apperception Test (TAT) *vs.* Describe a Dream and the first *vs.* second stream-of-consciousness (SOC) exercises. While in the latter case the lack of difference is to be expected given that the demands of the task were the same, the first difference is curious because the nature of the assignments are quite different. Note that this procedure and the large sample result in very powerful tests: the probability of detecting a standardized mean difference of 0.2 given it exists—a "small" effect (Cohen, 1988)—using a paired t-test with $n = 425$ and $\alpha = .05$ is .98, or 98%. Take note of the mean differences for each comparison to determine practical significance.

Table 3.6: Paired t-tests and effect sizes for differences in idea density across writing assignments. P-values are $p = Pr(T > |t^*| \mid H_0)$ where T follows a t-distribution with 424 degress of freedom under the null, and the alternative hypothesis of non-equality of means is non-directional. Effect sizes are Cohen's d for dependent samples. Means (m) and standard deviations (sd) given in the margins.

	Writing Assignment			
	2 (TAT story)	**3 (Dreams)**	**4 (2nd SOC)**	
3 (Dreams)	$p = 0.5, d = .084$	–	–	$m = .529, sd = .066$
4 (2nd SOC)	$p < .000, d = .975$	$p < .000, d = .402$	–	$m = .557, sd = .029$
1 (1st SOC)	$p < .000, d = 1.00$	$p < .000, d = .436$	$p = 0.81, d = .073$	$m = .559, sd = .029$
	$m = .523, sd = .029$	$m = .529, sd = .066$	$m = .557, sd = .0291$	

To compare visually across writing assignments, frequency wordclouds were generated for each of the four free-response submissions. For each, these consisted of the 150 words that appeared most frequently in students' writing assignments after removing stopwords, numbers, and punctuation. This was performed using R packages `tm` and `wordcloud`. Wordclouds are presented in Appendix C to save space in this section.

Benchmark Performance and Chats

I was also interested in the degree to which performance on the benchmark quizzes was related to behavior in the ungraded online small-group chats. I tried to regress overall benchmark score on total number of chat contributions but model checking revealed that the fit was no good; though robust against non-normality, the equal-variance assumption was clearly being violated and there were several outliers with high leverage. Instead of messing with transformations and omitting outliers, I decided to use a non-parametric kernel regression here (chances are a parametric model wouldn't have been correct in the first place). The specific technique I used estimates the regression function by fitting a "moving-average" smoother known as the Nadaraya-Watson estimator:

$$\hat{f}_\lambda(X) = \frac{\sum_{i=1}^{n} Y_i K(\frac{X-X_i}{\lambda})}{\sum_{i=1}^{n} K(\frac{X-X_i}{\lambda})}$$

Here, bandwidth $\lambda > 0$ depends on the size of the sample and the kernel function K, where $\int K = 1$ (in all analyses, a normal/gaussian kernel function was used). All analyses were performed using the `npreg` package in R. The optimal bandwidth ($\lambda = 53.604$ in this case) was selected automatically by choosing the one with the smallest error under k-fold cross-validation. Significance testing was done bootstrapping to determine the null distribution of the test statistic (see Racine, 2007). The above procedure was followed in all subsequent analyses of this type.

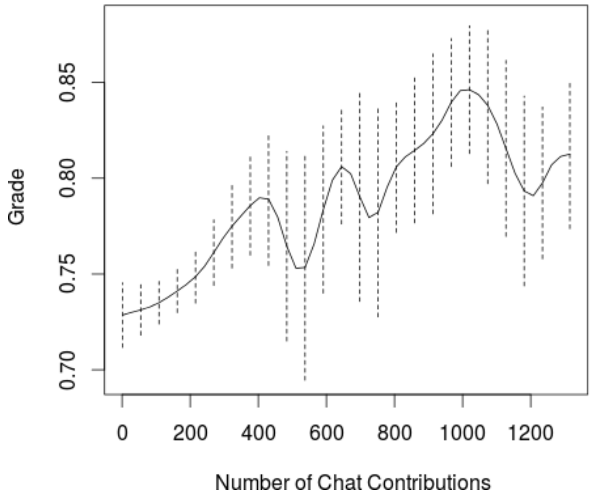
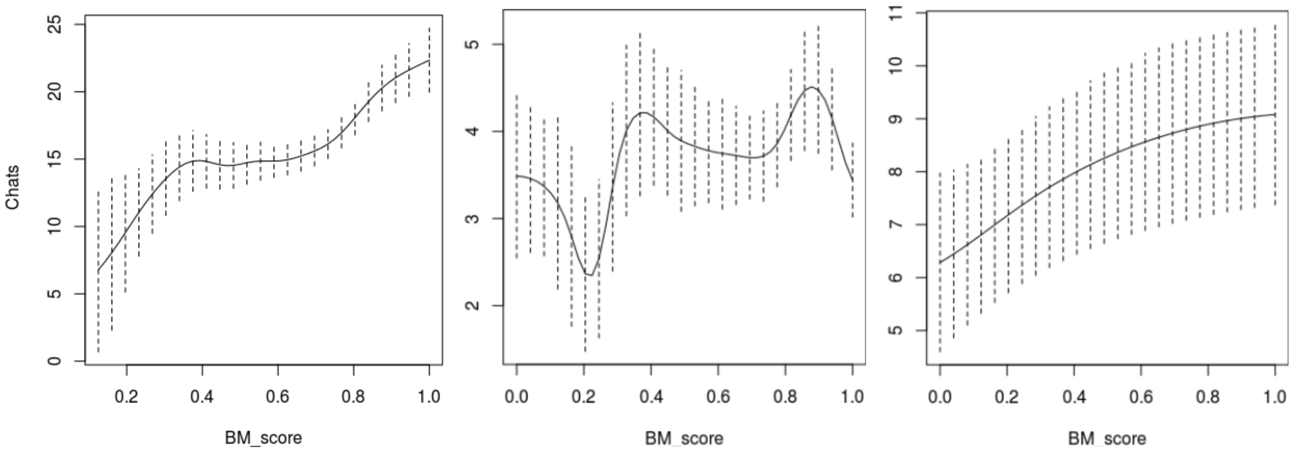


Figure 3.9: Non-parametric regression function predicting overall benchmark grade from total chat contributions

The number of chat contributions was found to significantly predict overall benchmark grade for all students who completed the course ($\lambda = 53.604$, $p = 0.0275$); the non-parametric regression function is shown in Figure 3.9. I also examined the relationship between performance on a benchmark quiz and the number of chat contributions made later that class period in a chat over material related to material covered by the benchmark. There were three benchmarks and three relevant chats: a BM about personality and the “big 5” followed by a chat about the personality and the “big 5”, a BM about correlations/experiments followed by a chat about correlations/experiments, and a BM about learning/conditioning followed by a chat about learning/-conditioning. The total number of BM items correct was regressed on the total number of chat contributions for each topic using nonparametric kernel regression. Results are shown in Figure 3.10. Predicting Chat from BM scores was significant for the correlations/experiments topic using a bootstrap kernel regression significance test ($\lambda = 0.9707,\ p < 0.0000$) Predicting Chat from BM scores was also significant for the personality/"big 5" topic using a bootstrap kernel regression significance test ($\lambda = 0.0590,\ p = 0.0100$). Predicting Chat from BM scores was not significant for the learning/conditioning topic using a bootstrap kernel significance test ($\lambda = 0.2642,\ p = 0.3408$).



(a) Correlations/Experiments (b) Personality/"Big 5" (c) Learning/Conditioning

Figure 3.10: Non-parametric regression function predicting number of chat contributions about various topics (Experimental Design, Personality/"Big 5", Learning/Conditioning) from score on benchmark covering same topic. Bootstrapped errors.

3.5 Discussion and Future Directions

Including SPARFA in the classroom appears to be a very promising way to maximize gains from spaced repetition during retrieval practice (e.g., weekly quizzes). Not only does it give interpretable estimates of what topics need extra attention (both in general and on an individual-student basis) based on quiz performance, but it can make optimal item-choice recommendations in a spaced retrieval practice paradigm by leveraging retrieval variability. Importantly, it can achieve these things based on only a small number of student responses to items, setting it apart from other methods.

With the present class data, SPARFA's student-concept estimates would have resulted in a much better use of the repeated question slot during benchmark quizzes by accounting for their correct re-answers and avoiding unnecessary repetitions. Another way in which the course's quizzing-with-repetition structure could be improved upon is by interleaving some form of restudy between the repetitions. It is known that, after initial learning, test-study-test sequences result in better retention than test-test-test sequences, which are in turn better than study-study-study sequences (Karpicke & Roediger, 2007; McDaniel et al. 2015). One way of achieving this could be through elaborated feedback, possibly given after a delay. In the present course design, feedback to students told them only whether their answer was right or wrong; elaborating upon the correct answer and why it is correct in a feedback message given after some delay could introduce a "restudy" element before the item or a related item appears again on a later quiz.

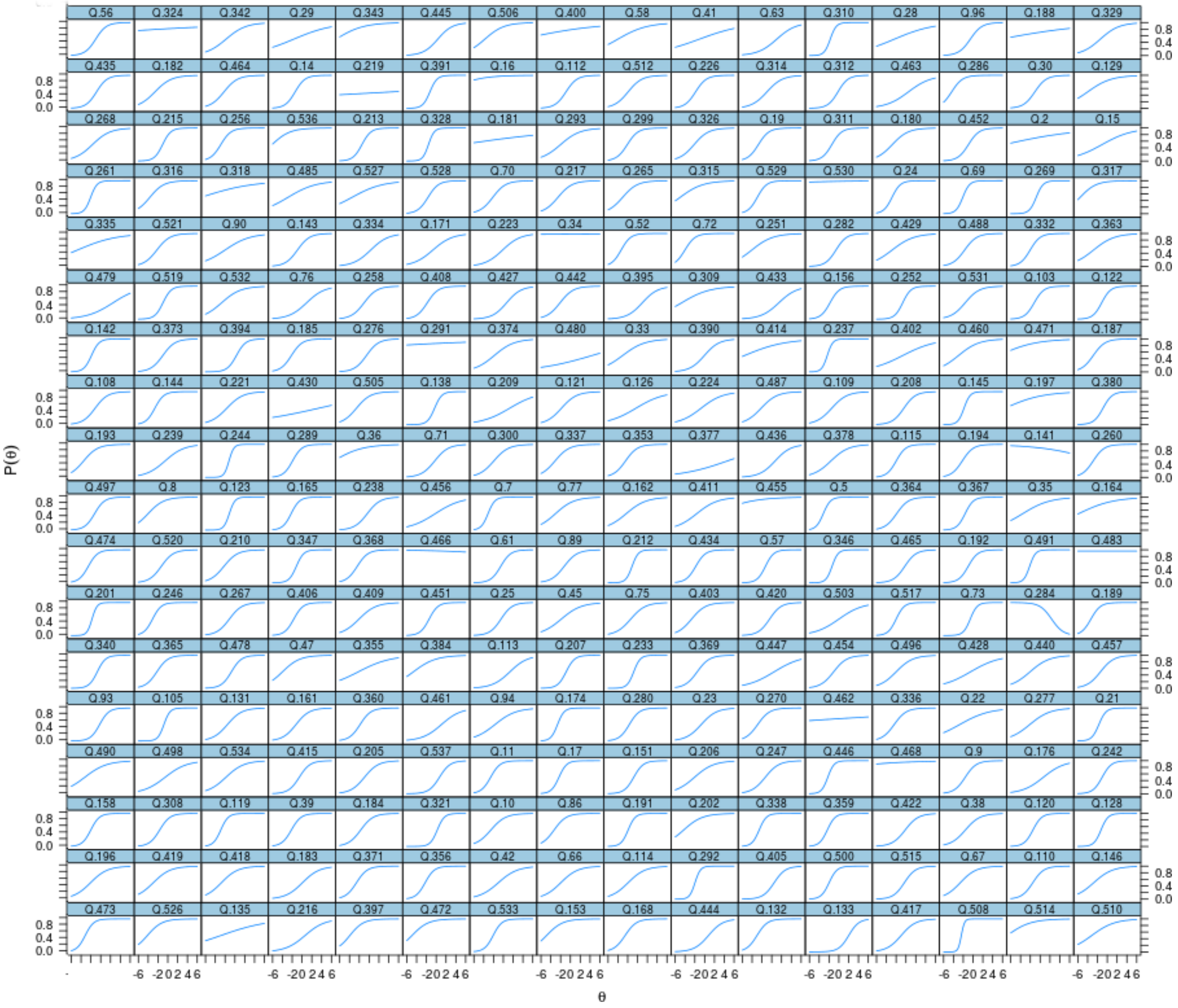
While SPARFA and MIRT can be used in the classroom to improve the efficacy of spaced retrieval practice, they do have several limitations. Importantly, they assume that learners' concept knowledge states remain constant over time, an assumption which is violated when these techniques are applied to student responses from different points in time (during which much learning and forgetting may occur). Also, these frameworks deal only with student-item interactions and thus have no means of accounting for the effects of other learning events that students encounter (reading a textbook, viewing a lecture, interacting with other students, etc.).

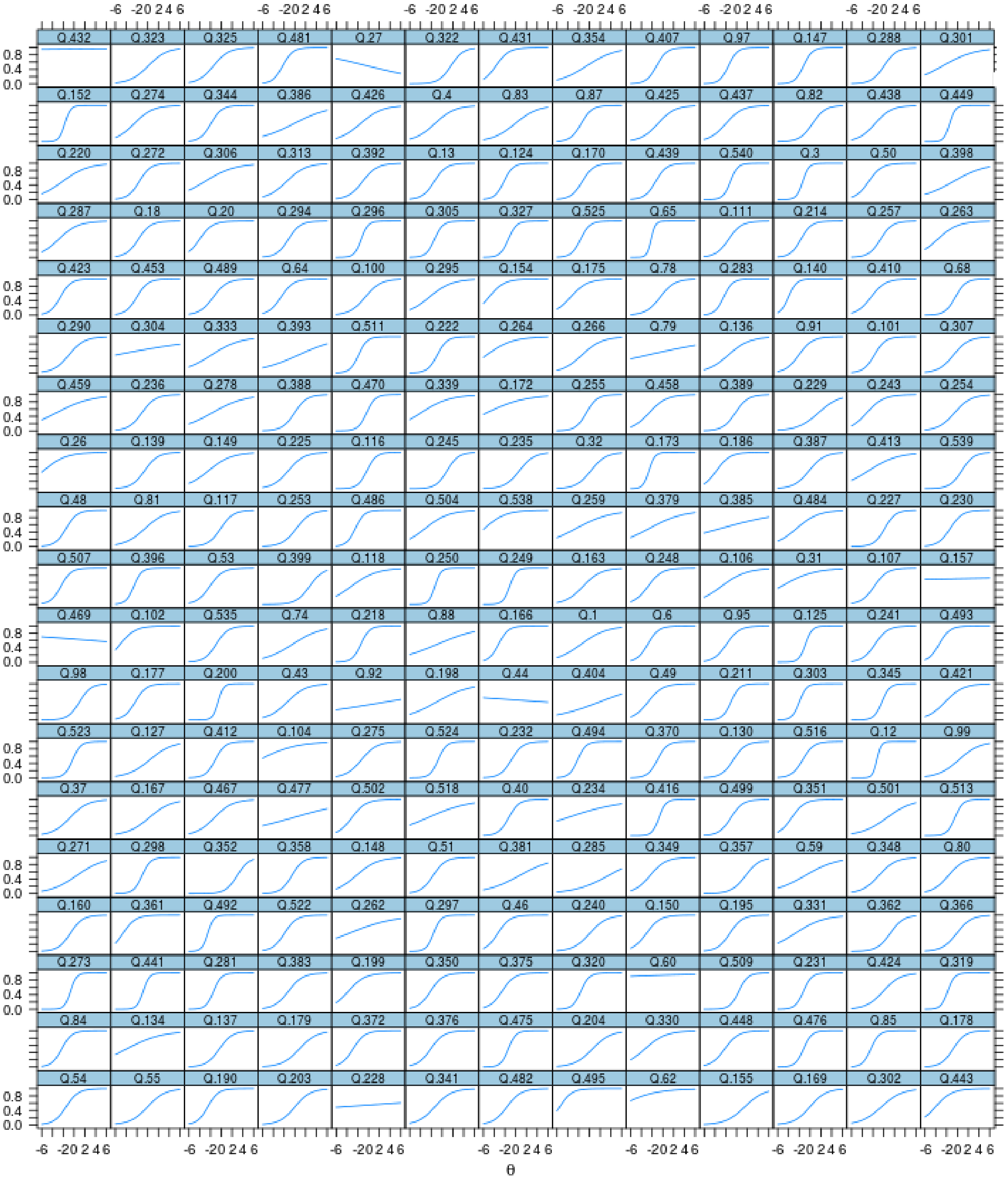
The SPARFA framework has recently been extended to overcome these limitations by including a latent state transition model based on Kalman filtering that traces students' concept-knowledge states over time using incoming data about questions students have answered and activities they have engaged in (Lan et al. 2014b). This new SPARFA-Trace framework takes the same graded learner-response matrix as input, but it additionally requires a *student-resource interaction matrix* that indicates whether or not a student has used certain learning resources between consecutive responses. Not only does this new framework estimate all of the original SPARFA parameters, but it captures learning concept knowledge evolution over time, and it also estimates parameters dealing with the organization and quality of the learning resource content. SPARFA-Trace seems able to provide more accurate assessments of students' concept knowledge while providing feedback to instructors about the efficacy of the learning resources (eg., chats, lectures, readings) used in their classrooms.

Appendices

Appendix A

Item characteristic curves for all items




Q.432
Q.323
Q.325
Q.481
Q.27
Q.322
Q.431
Q.354
Q.407
Q.97
Q.147
Q.288
Q.301
Q.152
Q.274
Q.344
Q.386
Q.426
Q.4
Q.83
Q.87
Q.425
Q.437
Q.82
Q.438
Q.449
Q.220
Q.272
Q.306
Q.313
Q.392
Q.13
Q.124
Q.170
Q.439
Q.540
Q.3
Q.50
Q.398
Q.287
Q.18
Q.20
Q.294
Q.296
Q.305
Q.327
Q.525
Q.65
Q.111
Q.214
Q.257
Q.263
Q.423
Q.453
Q.489
Q.64
Q.100
Q.295
Q.154
Q.175
Q.78
Q.283
Q.140
Q.410
Q.68
Q.290
Q.304
Q.333
Q.393
Q.511
Q.222
Q.264
Q.266
Q.79
Q.136
Q.91
Q.101
Q.307
Q.459
Q.236
Q.278
Q.388
Q.470
Q.339
Q.172
Q.255
Q.458
Q.389
Q.229
Q.243
Q.254
Q.26
Q.139
Q.149
Q.225
Q.116
Q.245
Q.235
Q.32
Q.173
Q.186
Q.387
Q.413
Q.539
Q.48
Q.81
Q.117
Q.253
Q.486
Q.504
Q.538
Q.259
Q.379
Q.385
Q.484
Q.227
Q.230
Q.507
Q.396
Q.53
Q.399
Q.118
Q.250
Q.249
Q.163
Q.248
Q.106
Q.31
Q.107
Q.157
Q.469
Q.102
Q.535
Q.74
Q.218
Q.88
Q.166
Q.1
Q.6
Q.95
Q.125
Q.241
Q.493
Q.98
Q.177
Q.200
Q.43
Q.92
Q.198
Q.44
Q.404
Q.49
Q.211
Q.303
Q.345
Q.421
Q.523
Q.127
Q.412
Q.104
Q.275
Q.524
Q.232
Q.494
Q.370
Q.130
Q.516
Q.12
Q.99
Q.37
Q.167
Q.467
Q.477
Q.502
Q.518
Q.40
Q.234
Q.416
Q.499
Q.351
Q.501
Q.513
Q.271
Q.298
Q.352
Q.358
Q.148
Q.51
Q.381
Q.285
Q.349
Q.357
Q.59
Q.348
Q.80
Q.160
Q.361
Q.492
Q.522
Q.262
Q.297
Q.46
Q.240
Q.150
Q.195
Q.331
Q.362
Q.366
Q.273
Q.441
Q.281
Q.383
Q.199
Q.350
Q.375
Q.320
Q.60
Q.509
Q.231
Q.424
Q.319
Q.84
Q.134
Q.137
Q.179
Q.372
Q.376
Q.475
Q.204
Q.330
Q.448
Q.476
Q.85
Q.178
Q.54
Q.55
Q.190
Q.203
Q.228
Q.341
Q.482
Q.495
Q.62
Q.155
Q.169
Q.302
Q.443
0.8
0.4
0.0
-6 -2 0 2 4 6
θ

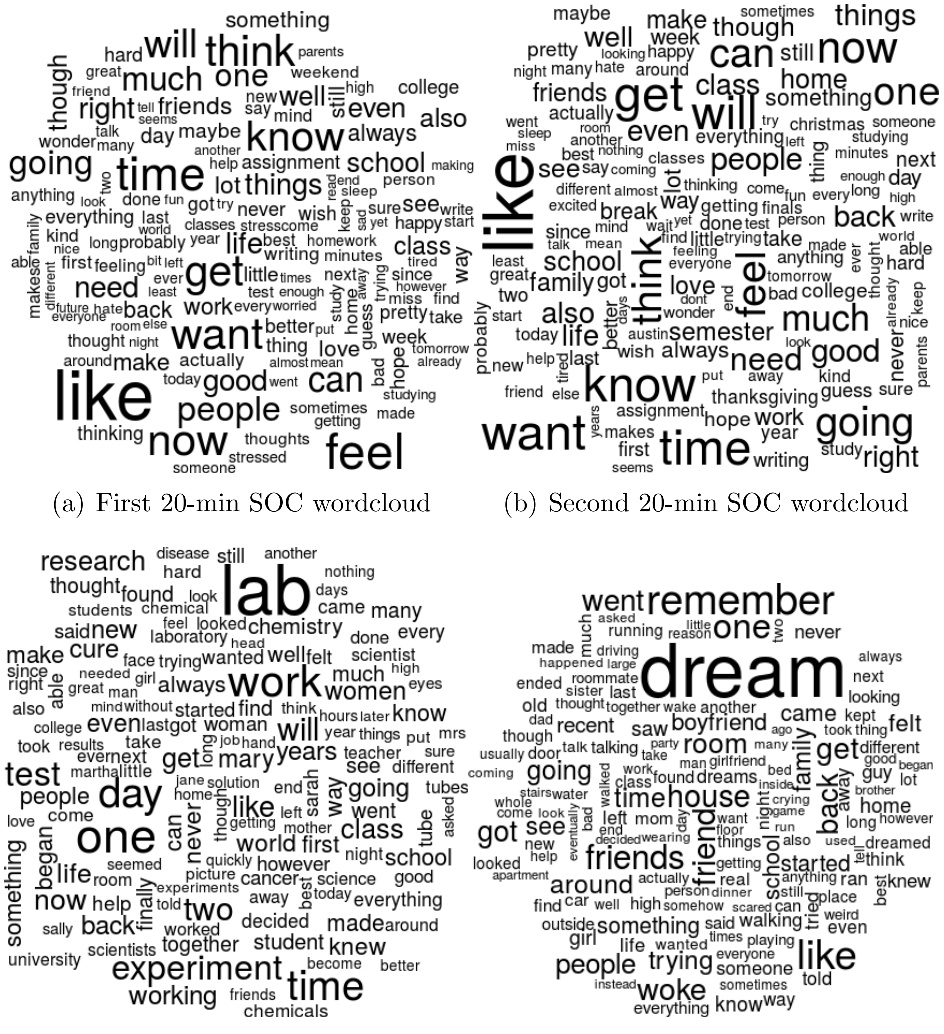
Appendix B

Table of tags for two BMs

Section	Unit	Tag	Question Prompt
Background in Psychology	Philosophy & correlations	bias	Tom decides to study the...
Background in Psychology	Philosophy & correlations	correlation	Which of the following is...
Background in Psychology	Philosophy & correlations	correlation	You read in a newspaper that...
Background in Psychology	Philosophy & correlations	random assignment	Dr. Martin is running an...
Background in Psychology	Philosophy & correlations	experimental design	Jamie interviewed dog owners...
Background in Psychology	Philosophy & correlations	bias	Sue wants to investigate...
Background in Psychology	Philosophy & correlations	correlation	Which of the following is...
Background in Psychology	Philosophy & correlations	correlation	Your friend tells you that...
Background in Psychology	Philosophy & correlations	experimental design	For a class project about...
Background in Psychology	Philosophy & correlations	experimental design	Gina's lab conducts a study...
Background in Psychology	Philosophy & correlations	experimental design	Dr. House surveys individuals...
Background in Psychology	Philosophy & correlations	experimental design	While in the cafeteria at...
Background in Psychology	Philosophy & correlations	experimental design	Yann visits a foreign...
Background in Psychology	Philosophy & correlations	historical	Who is credited with starting...
Background in Psychology	Experimental & causal thinking	variables	Bill ran a study to see if...
Background in Psychology	Experimental & causal thinking	test an intervention	Dr. Jones is teaching a...
Background in Psychology	Experimental & causal thinking	experimental design	Dr. Vale wants to know if...
Background in Psychology	Experimental & causal thinking	validity	Harriet asked a large group...
Background in Psychology	Experimental & causal thinking	variables	S&y is head chef at a...
Background in Psychology	Experimental & causal thinking	broad/narrow	Sarah's husb& travels...
Background in Psychology	Experimental & causal thinking	bias	Which of the following is...
Background in Psychology	Experimental & causal thinking	variables	In order to assess the...
Background in Psychology	Experimental & causal thinking	test an intervention	Pedro, Cindy, & Christie...
Background in Psychology	Experimental & causal thinking	broad/narrow	Tyler saw on the news that...
Background in Psychology	Experimental & causal thinking	bias	Which of the following is...
Background in Psychology	Experimental & causal thinking	variables	Bill ran a study to see...
Background in Psychology	Experimental & causal thinking	validity	Harriet asked a group of...
Background in Psychology	Experimental & causal thinking	validity	In class, Dr. Shine described...
Background in Psychology	Experimental & causal thinking	control group	Ted works at a hospital...

Appendix C

Wordclouds for all free-response assignments



(a) First 20-min SOC wordcloud

(b) Second 20-min SOC wordcloud

(c) Thematic Apperception wordcloud

(d) Dream description wordcloud

Appendix D

R and Python code

```r
dataset<-read.csv("answer_data_working.csv")

#general info
dim(dataset)
str(dataset)

#what are all of the column names?
colnames(dataset)
#how many variables so far?
length(colnames(dataset))

#number of unique eids
length(unique(dataset$eid))
#number of unique student ids
length(unique(dataset$student_id))
#number of unique quiz_answer_ids
length(unique(dataset$quiz_answer_id))
#number of unique activity_ids
length(unique(dataset$activity_id))
#histogram of
#unique prompts for each benchmark (including repeats from previous bms)

qs46<-unique(dataset$prompt[dataset$activity_id==46])
numqs46=length(unique(dataset$prompt[dataset$activity_id==46]))
qs57<-unique(dataset$prompt[dataset$activity_id==57])
numqs57<-length(unique(dataset$prompt[dataset$activity_id==57]))
qs69<-unique(dataset$prompt[dataset$activity_id==69])
numqs69<-length(unique(dataset$prompt[dataset$activity_id==69]))
qs81<-unique(dataset$prompt[dataset$activity_id==81])
numqs81<-length(unique(dataset$prompt[dataset$activity_id==81]))
qs84<-unique(dataset$prompt[dataset$activity_id==84])
numqs84<-length(unique(dataset$prompt[dataset$activity_id==84]))
qs96<-unique(dataset$prompt[dataset$activity_id==96])
numqs96<-length(unique(dataset$prompt[dataset$activity_id==96]))
qs147<-unique(dataset$prompt[dataset$activity_id==147])
numqs147<-length(unique(dataset$prompt[dataset$activity_id==147]))
qs165<-unique(dataset$prompt[dataset$activity_id==165])
numqs165<-length(unique(dataset$prompt[dataset$activity_id==165]))
qs181<-unique(dataset$prompt[dataset$activity_id==181])
numqs181<-length(unique(dataset$prompt[dataset$activity_id==181]))
qs192<-unique(dataset$prompt[dataset$activity_id==192])
numqs192<-length(unique(dataset$prompt[dataset$activity_id==192]))
qs210<-unique(dataset$prompt[dataset$activity_id==210])
numqs210<-length(unique(dataset$prompt[dataset$activity_id==210]))
```

```
qs223<-unique(dataset$prompt[dataset$activity_id==223])
numqs223<-length(unique(dataset$prompt[dataset$activity_id==223]))
qs241<-unique(dataset$prompt[dataset$activity_id==241])
numqs241<-length(unique(dataset$prompt[dataset$activity_id==241]))
qs259<-unique(dataset$prompt[dataset$activity_id==259])
numqs259<-length(unique(dataset$prompt[dataset$activity_id==259]))
qs269<-unique(dataset$prompt[dataset$activity_id==269])
numqs269<-length(unique(dataset$prompt[dataset$activity_id==269]))
qs289<-unique(dataset$prompt[dataset$activity_id==289])
numqs289<-length(unique(dataset$prompt[dataset$activity_id==289]))
qs306<-unique(dataset$prompt[dataset$activity_id==306])
numqs306<-length(unique(dataset$prompt[dataset$activity_id==306]))
qs330<-unique(dataset$prompt[dataset$activity_id==330])
numqs330<-length(unique(dataset$prompt[dataset$activity_id==330]))
qs344<-unique(dataset$prompt[dataset$activity_id==344])
numqs344<-length(unique(dataset$prompt[dataset$activity_id==344]))
qs360<-unique(dataset$prompt[dataset$activity_id==360])
numqs360<-length(unique(dataset$prompt[dataset$activity_id==360]))
qs379<-unique(dataset$prompt[dataset$activity_id==379])
numqs379<-length(unique(dataset$prompt[dataset$activity_id==379]))
qs391<-unique(dataset$prompt[dataset$activity_id==391])
numqs391<-length(unique(dataset$prompt[dataset$activity_id==391]))
qs403<-unique(dataset$prompt[dataset$activity_id==403])
numqs403<-length(unique(dataset$prompt[dataset$activity_id==403]))
qs421<-unique(dataset$prompt[dataset$activity_id==421])
numqs421<-length(unique(dataset$prompt[dataset$activity_id==421]))
qs434<-unique(dataset$prompt[dataset$activity_id==434])
numqs434<-length(unique(dataset$prompt[dataset$activity_id==434]))
qs450<-unique(dataset$prompt[dataset$activity_id==450])
numqs450<-length(unique(dataset$prompt[dataset$activity_id==450]))
qs467<-unique(dataset$prompt[dataset$activity_id==467])
numqs467<-length(unique(dataset$prompt[dataset$activity_id==467]))

#average number of new questions per student for a given activity
sum(dataset$X1st.pres[dataset$activity_id==96])/length(unique(dataset$student_id[dataset$activity_id==96]))
#number of new unique prompts for a given activity
length(unique(dataset$prompt[dataset$activity_id==69 & dataset$X1st.pres==1]))

length(unique(dataset$prompt[dataset$activity_id==101 & dataset$X1st.pres==1]))
length(unique(dataset$prompt[dataset$activity_id==101]))
length(unique(dataset$student_id[dataset$activity_id==101]))

bm1<-dataset[dataset$activity_id==46,]
bm2<-dataset[dataset$activity_id==57,]
bm3<-dataset[dataset$activity_id==69,]
bm4<-dataset[dataset$activity_id==81,]
bm5<-dataset[dataset$activity_id==84,]
bm6<-dataset[dataset$activity_id==96,]
bm7<-dataset[dataset$activity_id==147,]
bm8<-dataset[dataset$activity_id==165,]
bm9<-dataset[dataset$activity_id==181,]
bm10<-dataset[dataset$activity_id==192,]
bm11<-dataset[dataset$activity_id==210,]
bm12<-dataset[dataset$activity_id==223,]
bm13<-dataset[dataset$activity_id==241,]
```

```r
bm14<-dataset[dataset$activity_id==259,]
bm15<-dataset[dataset$activity_id==269,]
bm16<-dataset[dataset$activity_id==289,]
bm17<-dataset[dataset$activity_id==306,]
bm18<-dataset[dataset$activity_id==330,]
bm19<-dataset[dataset$activity_id==344,]
bm20<-dataset[dataset$activity_id==360,]
bm21<-dataset[dataset$activity_id==379,]
bm22<-dataset[dataset$activity_id==391,]
bm23<-dataset[dataset$activity_id==403,]
bm24<-dataset[dataset$activity_id==421,]
bm25<-dataset[dataset$activity_id==434,]
bm26<-dataset[dataset$activity_id==450,]
bm27<-dataset[dataset$activity_id==467,]

bmdata<-rbind(bm1,bm2,bm3,bm4,bm5,bm6,bm7,bm8,bm9,bm10,bm11,bm12,bm13,bm14,bm15,bm16,bm17,bm18,bm19,bm20,bm21,bm22,bm23,bm24,bm25,bm26,bm27)
bmdata<-data.frame(bmdata)

#how many unique bm prompts total?
length(unique(bmdata$prompt))
#540
length(unique(bmdata$response))
#2000 unique responses

#assign question id to first appearance of question; indicates what unit it comes from
bmdata$qid[is.element(bmdata$prompt,qs46)]<-1
bmdata$qid[is.element(bmdata$prompt,qs57[!is.element(qs57,qs46)])]<-2
bmdata$qid[is.element(bmdata$prompt,qs69[!is.element(qs69,tmp1<-union(qs46,qs57))])]<-3
bmdata$qid[is.element(bmdata$prompt,qs81[!is.element(qs81,tmp2<-union(tmp1,qs69))])]<-4
bmdata$qid[is.element(bmdata$prompt,qs84[!is.element(qs84,tmp1<-union(tmp2,qs81))])]<-5
bmdata$qid[is.element(bmdata$prompt,qs96[!is.element(qs96,tmp2<-union(tmp1,qs84))])]<-6
bmdata$qid[is.element(bmdata$prompt,qs147[!is.element(qs147,tmp1<-union(tmp2,qs96))])]<-7
bmdata$qid[is.element(bmdata$prompt,qs165[!is.element(qs165,tmp2<-union(tmp1,qs147))])]<-8
bmdata$qid[is.element(bmdata$prompt,qs181[!is.element(qs181,tmp1<-union(tmp2,qs165))])]<-9
bmdata$qid[is.element(bmdata$prompt,qs192[!is.element(qs192,tmp2<-union(tmp1,qs181))])]<-10
bmdata$qid[is.element(bmdata$prompt,qs210[!is.element(qs210,tmp1<-union(tmp2,qs192))])]<-11
bmdata$qid[is.element(bmdata$prompt,qs223[!is.element(qs223,tmp2<-union(tmp1,qs210))])]<-12
bmdata$qid[is.element(bmdata$prompt,qs241[!is.element(qs241,tmp1<-union(tmp2,qs223))])]<-13
bmdata$qid[is.element(bmdata$prompt,qs259[!is.element(qs259,tmp2<-union(tmp1,qs241))])]<-14
bmdata$qid[is.element(bmdata$prompt,qs269[!is.element(qs269,tmp1<-union(tmp2,qs259))])]<-15
bmdata$qid[is.element(bmdata$prompt,qs289[!is.element(qs289,tmp2<-union(tmp1,qs269))])]<-16
bmdata$qid[is.element(bmdata$prompt,qs306[!is.element(qs306,tmp1<-union(tmp2,qs289))])]<-17
bmdata$qid[is.element(bmdata$prompt,qs330[!is.element(qs330,tmp2<-union(tmp1,qs306))])]<-18
```

```
bmdata$qid[is.element(bmdata$prompt,qs344[!is.element(qs344,tmp1<-union(tmp2,qs330))])]<-19
bmdata$qid[is.element(bmdata$prompt,qs360[!is.element(qs360,tmp2<-union(tmp1,qs344))])]<-20
bmdata$qid[is.element(bmdata$prompt,qs379[!is.element(qs379,tmp1<-union(tmp2,qs360))])]<-21
bmdata$qid[is.element(bmdata$prompt,qs391[!is.element(qs391,tmp2<-union(tmp1,qs379))])]<-22
bmdata$qid[is.element(bmdata$prompt,qs403[!is.element(qs403,tmp1<-union(tmp2,qs391))])]<-23
bmdata$qid[is.element(bmdata$prompt,qs421[!is.element(qs421,tmp2<-union(tmp1,qs403))])]<-24
bmdata$qid[is.element(bmdata$prompt,qs434[!is.element(qs434,tmp1<-union(tmp2,qs421))])]<-25
bmdata$qid[is.element(bmdata$prompt,qs450[!is.element(qs450,tmp2<-union(tmp1,qs434))])]<-26
bmdata$qid[is.element(bmdata$prompt,qs467[!is.element(qs467,tmp1<-union(tmp2,qs450))])]<-27

#questions unique to each unit
un1<-unique(bmdata$prompt[bmdata$qid==1])
un2<-unique(bmdata$prompt[bmdata$qid==2])
un3<-unique(bmdata$prompt[bmdata$qid==3])
un4<-unique(bmdata$prompt[bmdata$qid==4])
un5<-unique(bmdata$prompt[bmdata$qid==5])
un6<-unique(bmdata$prompt[bmdata$qid==6])
un7<-unique(bmdata$prompt[bmdata$qid==7])
un8<-unique(bmdata$prompt[bmdata$qid==8])
un9<-unique(bmdata$prompt[bmdata$qid==9])
un10<-unique(bmdata$prompt[bmdata$qid==10])
un11<-unique(bmdata$prompt[bmdata$qid==11])
un12<-unique(bmdata$prompt[bmdata$qid==12])
un13<-unique(bmdata$prompt[bmdata$qid==13])
un14<-unique(bmdata$prompt[bmdata$qid==14])
un15<-unique(bmdata$prompt[bmdata$qid==15])
un16<-unique(bmdata$prompt[bmdata$qid==16])
un17<-unique(bmdata$prompt[bmdata$qid==17])
un18<-unique(bmdata$prompt[bmdata$qid==18])
un19<-unique(bmdata$prompt[bmdata$qid==19])
un20<-unique(bmdata$prompt[bmdata$qid==20])
un21<-unique(bmdata$prompt[bmdata$qid==21])
un22<-unique(bmdata$prompt[bmdata$qid==22])
un23<-unique(bmdata$prompt[bmdata$qid==23])
un24<-unique(bmdata$prompt[bmdata$qid==24])
un25<-unique(bmdata$prompt[bmdata$qid==25])
un26<-unique(bmdata$prompt[bmdata$qid==26])
un27<-unique(bmdata$prompt[bmdata$qid==27])

AllUniqueQs<-list(un1,un2,un3,un4,un5,un6,un7,un8,un9,un10,un11,un12,un13,un14,un15,un16,un17,un18,un19,un20,un21,un22,un23,un24,un25,un26,un27)

sink("AllUniQuestions")
lapply(AllUniqueQs,print)
sink()
lapply(AllUniqueQs, function(x) write.table( data.frame(x), ’test.csv’ , append= T, sep=’,’ ))
```

```

#going to write a few things to file:
write.csv(bmdata, file="bmdata.csv")
write.csv(uniquedf,file="allQuestions.csv")

#get only first-presentation data, remove students who were absent
#ie, answers not graded 1 or 0
bmdata1n<-subset(bmdata, X1st.pres==’1’ & correct!=’NA’)

#create unique ids for items
bmdata1n<-transform(bmdata1n,item_id=as.numeric(factor(prompt)))
#unique response id
bmdata1n<-transform(bmdata1n,resp_id=as.numeric(factor(response)))

###################new bmdata with respLab fixed, plus corresp and corresplab!!!!!!!!!!!!
bmdata1n<-read.csv("bmdata1n.csv")

#func<-function(x){
#  as.numeric(factor(x))
#}
#respLab<-by(bmdata1n$response, bmdata1n$prompt, func)
#head(respLab)
#unlist(respLab,use.names=F)

write.csv(bmdata1n,"bmdata1n.csv")

#allresp<-list()
#for(i in 1:27){
#allresp[i]<-bmdata1n$response[bmdata1n$resplab==c(1,2,3,4,5) & bmdata1n$qid==i]
#}

#histogram of qids
hist(bmdata1n$qid,breaks=1000)

#new bm
bmdata2<-read.csv("bmdata2.csv",header=T,sep="\t")

#PLOTS
barplot(lis,main="Number of new unique questions per benchmark", xlab="Benchmark",names.arg=seq(1:27))

barplot(tab<-table(bmdata2$qid,bmdata2$activity_id),col=rainbow(27),names.arg=seq(1:27),main="Questions per Benchmark Colored-coded by Original Presentation",xlab="Benchmark", ylim=c(0,7000))

#barplot(table(bmdata$activity_id,bmdata$qid),col=rainbow(27),names.arg=seq(1:27))

#item origin per benchmark
table(bmdata2$qid[bmdata2$BM==17])
table(bmdata2$qid[bmdata2$BM==19])
table(bmdata2$qid[bmdata2$BM==3])/sum(table(bmdata2$qid[bmdata2$BM==3]))

#students per benchmark
table(bmdata2$eid[bmdata2$BM==17])

####################################################################################
```

```
#compare sparfa's predicts on bm1 to repeated items on bm2:
firsts<-subset(bmdata2,BM_2=='1' & X1st.pres=='1')
repqs<-subset(bmdata2, BM_2=='2' & X2nd.pres=='1')

Cscal<-scale(t(C),center=T,scale=T)
Cscal<-as.data.frame(Cscal)
names(Cscal)<-c("F1","F2","F3","F4")

minfs<-as.matrix(apply(Cscal,1,which.min))

newv<-vector()

for(i in 1:length(minfs)){
if(minfs[i]==1){if(repqs$item_id[i] %in% c(228,55) == T){newv[i]<-1}else{newv[i]<-0}}
else if(minfs[i]==2){if(repqs$item_id[i] %in% c(54,190,55) == T){newv[i]<-1}else{newv[i]<-0}}
else if(minfs[i]==3){if(repqs$item_id[i] %in% c(55,190) == T){newv[i]<-1}else{newv[i]<-0}}
else if(minfs[i]==4){if(repqs$item_id[i] %in% c(203,55) ==T){newv[i]<-1}else{newv[i]<-0}}
}
############################################################################

eid_grades<-read.csv("bm_grade_byEID.csv",header=T)

par(mfrow=c(3,1))
hist(eid_grades$grade,breaks=c(seq(0,1,.01)), axes=F, main="",ylab="Count",xlab="Percent of BM items answered correctly (all students, n=939)",col="grey")
axis(1,at=c(seq(0,1,.05),1))
axis(2,at=c(seq(0,100,10),100))

hist(eid_grades$grade[eid_grades$total>0],breaks=c(seq(0,1,.01)), axes=F, main="",ylab="Count",xlab="Percent of BM items answered correctly (students attempting at least one BM, n=845)",col="grey")
axis(1,at=c(seq(0,1,.05),1))
axis(2,at=c(seq(0,50,10),50))

hist(eid_grades$grade[(eid_grades$grade*.85+.15)>=.6],breaks=c(seq(0,1,.01)), axes=F, main="",ylab="Count",xlab="Percent of BM items answered correctly (students completeing the course, n=677)",col="grey")
axis(1,at=c(seq(0,1,.05),1))
axis(2,at=c(seq(0,50,10),50))

#cumulative histogram of answered questions
h<-hist(eid_grades$total,breaks=seq(0,216,8))
h$counts<-cumsum(h$counts)

barplot(eid_grades$total,names.arg=seq(1,27,1),ylim=c(0,1100))
barplot(h$counts,names.arg=seq(0,208,8),ylim=c(0,1100))
barplot(h$counts)

plot(ecdf(eid_grades$total),verticals=T,do.points=F)
abline(h=949,lty=2)

plot(h,main="Cumulative histogram of students by BM items attempted",axes=F,xlab="Number of BM items attempted",xlim=c(0,214),col="grey")
axis(1,at=c(seq(0,208,by=8),214),lwd=1,lwd.ticks=1,las=2)
```

```r
axis(2,at=seq(0,950,by=50),las=2)
abline(h=939,lty=2)
abline(v=)
totAns<-eid_grades$total

###wide data for sparfa etc
#create unique ids for items
bmdata2<-transform(bmdata2,item_id=as.numeric(factor(prompt)))
#unique response id
bmdata2<-transform(bmdata2,resp_id=as.numeric(factor(response)))
write.csv(bmdata2,"bmdata2new.csv")
bmdata2<-read.csv("bmdata2new.csv")
responses=subset(bmdata2[,c(5,26,29,30,31,32,33)], bmdata2$X1st.pres=="1" )

#remove items given only to one person
which(rowSums(table(responses$item_id,responses$student_id))==1)
responses=subset(responses, item_id!=159 & item_id!=279 & item_id!=382 & item_id!=401)
wideresp=reshape(responses,timevar="eid",idvar=c("item_id", "correct", "resp_id"), direction="wide")
write.csv(wideresp,"responses.csv")

itembystudent<-subset(responses[,c(1,5,6)])

itembystudent<-reshape(itembystudent, timevar="student_id", idvar="item_id",direction="wide")
irtdata<-reshape(itembystudent, timevar="item_id", idvar="student_id",direction="wide")
irtdata<-reshape(itembystudent, timevar="item_id", idvar="student_id",direction="wide")
write.csv(itembystudent, "itembystudent.csv")

#qid by student to code as incorrect unattempted quizzes
# cut responses up into separate files by benchmark
for(i in 1:27){
  abc<-subset(responses[responses$qid==i,c(1,5,6)])
  abc<-reshape(abc,timevar="student_id", idvar="item_id", direction="wide")
  filename <- paste(i, ".csv", sep="")
  write.csv(abc,filename)
  }
#######################PCA on all questions?
pcadata<-subset(responses, correct!=’NA’)
pcafit1<-princomp(cbind(pcadata$item_id[q],pcadata$correct))
princomp(subset(pcadata$correct, pcadata$qid=="1"))

#########################################################################
#impute data #might need to go long to wide to long to get NAs

library(missForest)
dat.imp<-missForest(itembystudent[,-1])
impute2<-as.matrix(dat.imp$ximp)

pca1<-princomp(t(impute2))
paf8<-fa(t(impute2),8,,fm="pa",rotate="none")
eigs<-pca1$sdev^2
percentvar<-eigs*100/sum(eigs)

##number of NAs
sum(is.na(itembystudent))
```

```r
#[1] 313910
#percent of data unobserved for n=845
313910/(845*540)
#0.6879% of the data is unobserved


library(qgraph)
barplot(table(tagdata$V13,tagdata$numbs),col=rainbow(length(unique(tagdata$V13))))
par(mfrow=c(1,3))
## W/a hist:
h1 = hist(scale(W), plot=FALSE, breaks=50)
h2 = hist(scale(disc), plot=FALSE,breaks=70)
h2$counts = - h2$counts
hmax = max(h1$counts)
hmin = min(h2$counts)
X = c(h1$breaks, h2$breaks)
xmax = max(X)
xmin = min(X)
plot(h1, ylim=c(-40,40), col="grey", xlim=c(-4,4), main="",xlab="W/a")
lines(h2, col="white")
legend("topright", c("SPARFA", "MIRT"), fill=c("grey","white"))

# C/theta hist:
h1 = hist(scale(t(C)), plot=FALSE, breaks=50)
h2 = hist(scale(thet), plot=FALSE,breaks=50)
h2$counts = - h2$counts
hmax = max(h1$counts)
hmin = min(h2$counts)
X = c(h1$breaks, h2$breaks)
xmax = max(X)
xmin = min(X)
plot(h1, ylim=c(-45, 45), col="grey", xlim=c(xmin, xmax),main="", xlab=expression(C/paste(theta)),yaxt="n", ylab=NA)
lines(h2, col="white")

#diff/M hist:
h1 = hist(M, plot=FALSE, breaks=60)
h2 = hist(difs, plot=FALSE,breaks=60)
h2$counts = - h2$counts
hmax = max(h1$counts)
hmin = min(h2$counts)
X = c(h1$breaks, h2$breaks)
xmax = max(X)
xmin = min(X)
plot(h1, ylim=c(hmin, hmax), col="grey", xlim=c(-4,6), main="", xlab="M/d", yaxt="n", ylab=NA)
lines(h2, col="white")
legend("topright", c("SPARFA", "MIRT"), fill=c("grey","white"))
#text(5,20,"SPARFA (top)")
#text(5,-20,"MIRT (bottom)")

hist(difs,breaks=40,col="grey",main="",xlab="a (discrimination)",ylab="Item frequency")
abline(v=0.5,col="red")
boxplot(disc,add=T,boxwex=5, staplewex=3,ylim=c(-1,4),horizontal=T, outline=F, at=2)

########################################
```

```
# SPARFA STUFF
rm(C,W,M,est,probs,probsv,probab,as,theta,probabv)

C<-read.csv("new_sparfac1.csv",header=F)
W<-read.csv("new_sparfaw1.csv",header=F)
M<-W[,ncol(W)]
W<-W[,-ncol(W)]
C<-as.matrix(C)
W<-as.matrix(W)
M<-as.matrix(M)
M<-matrix(rep(M,ncol(C)),nrow=nrow(W))

est<-(W%*%C)+M

#Csd<-apply(C,1,sd)
#Cmean<-apply(C,1,mean)
#Cstd<-(C-Cmean)/Csd

logistic<-function(x){1/(1+exp(-x))}
rbern<-function(p,n=1){sims<-sample(0:1, size=n, replace=TRUE,prob=c(1-p,p))
return(sims)}

probs<-logistic(est)
probsv<-matrix(probs,ncol=1,byrow=FALSE)
predict<-sapply(probs,rbern)

as<-read.csv("mirt2_coefs_formatted.csv",sep="\t",header=F)
theta<-read.csv("mirt2_fscores_formatted.csv",sep=’\t’,header=F)
fls<-read.csv("small_mirt3_summary.csv",sep=’\t’,header=F)

as<-read.csv("small_mirt3_coefs.csv", sep=’\t’,header=F)
theta<-read.csv("small_mirt3fscores.csv",sep=’\t’, header=F)
fls<-read.csv("small_mirt3_summary.csv",sep=’\t’,header=F)

as<-read.csv("mirtk2_coefs.csv",sep=’\t’,header=F)
theta<-read.csv("mirtk2fscores.csv",sep=’\t’,header=F)
d<-as.matrix(as[,ncol(as)])
as<-as.matrix(as[,-ncol(as)])
theta<-as.matrix(theta)
fls<-as.matrix(fls)

Tsd<-apply(theta,2,sd)
Tmean<-apply(theta,2,mean)
Tstd<-(theta-Tmean)/Tsd

twopl<-function(as,theta){
  probs1<-matrix(0,nrow(as),nrow(theta))
for(i in 1:535){
  for(j in 1:838){
  probs1[i,j]<-as.numeric((1/(1+exp(-1*(as[i,-1]*theta[j,1]+as[i,2])))))}}
  return(probs1);
  }

probab<-matrix(twopl(as,theta),nrow=nrow(as),ncol=nrow(theta),byrow=T)

probab<-logistic((as[,-ncol(as)]%*%t(theta))+rep(as[,ncol(as)],nrow(theta)))
```

```

xx<-((as%*%t(theta))+rep(d,nrow(theta)))
probab<-logistic(xx)
probabv<-matrix(probab,ncol=1,byrow=FALSE)

cor(probabv,probsv)
cor(as.vector(C),as.vector(t(theta)))
cor(as.vector(W),as.vector(as))
cor(M[,1],d)

cor(probs,probab)

itemplot(mirt2,8)
###########################################################################
#IRT?

#gbook<-cbind(bmdata1n[,c(2,22,26)])
#gbook<-gbook[which(gbook$item_id!=448 & gbook$item_id!=399 & gbook$item_id!=278),]
#remove items only given to one person
#edit(gbook)
#library(ltm)
#or
library(mirt)

#reshape data to wide format with just id by item (for irt)
#gbook_wide<-reshape(gbook, idvar="student_id", timevar="item_id", direction="wide")
#lots of missing data because everyone hasn't seen the same questions
#dim(gbook_wide)
#descript(gbook_wide)

#remove student column
#gbw<-gbook_wide[,-1]
#write.csv(gbw, "responses.csv")

#determine how many NAs in each column
which(sapply(itembystudent, function(y)sum(length(which(is.na(y))))/length(y))==1)
#plain irt
irtdata<-itembystudent
irtdata$correct_na.1842<-NULL
irtdata$correct_na.1385<-NULL
irtdata$correct_na.1391<-NULL
irtdata$correct_na.1622<-NULL
irtdata$correct_na.1580<-NULL
irtdata$correct_na.221<-NULL
irtdata$correct_na.1162<-NULL
irtdata$correct_na.1812<-NULL
##super annoying bc wants student by item, not item by student
##remove missing rows
irtdata<-as.data.frame(t(as.matrix(irtdata)))
irtdata$"81771"<-NULL
irtdata$"83044"<-NULL
irtdata$"155758"<-NULL
irtdata$"83830"<-NULL
irtdata$"66662"<-NULL

##############################
```

```r
#items with no score
irtdata$Q.159<-NULL
irtdata$Q.382<-NULL

#items with only NAs
irtdata$Q.279<-NULL
irtdata$Q.450<-NULL
irtdata$Q.401<-NULL

#how many zero rows?
irtdata<-irtdata[-(which(rowSums(irtdata[,-1],na.rm=T)<=1)),]
irtdata$correct_na.450<-NULL

which(colSums(irtdata[,-1],na.rm=T)==0)
#761   5497   5793  24769 111473 111505
#96    688    725    838    842    843
studentlist<-irtdata[,1]

irtdata2<-irtdata[which(((rowSums(irtdata[,-1],na.rm=T)/214)*.85+.15)>.57),]
irtdata2$correct_na.508<-NULL
irtdata2$correct_na.140<-NULL
irtdata3<-irtdata[which(((rowSums(irtdata[,-1],na.rm=T)/214)*.85+.15)>.80,]
irtdata3<-t(irtdata2)
write.csv(irtdata3, "irtdata3.csv")

#mirt1<-mirt(irtdata[,-1],1,technical = list(removeEmptyRows=T), SE=T)

mirt2<-mirt(irtdata[,-1],2, method="QMCEM")
mirt3<-mirt(irtdata[,-1],3, method="QMCEM")

mirt31<-mirt(irtdata[,-1],8,method="QMCEM")
mirt4<-mirt(irtdata[,-1],27,method="QMCEM")
mirt5<-confmirt(irtdata[,-1],27)
plot(mirt2, type="trace")

mirt1s<-mirt(irtdata2[,-1],1, method="QMCEM")
mirt2s<-mirt(irtdata2[,-1],8,method="QMCEM")
mirt3s<-mirt(irtdata2[,-1],27,method="QMCEM")

#summary
summary(mirt1)
#latent trait estimates
fscores(mirt1)

#withimputeddata
mirt_imp<-mirt(itembystudent.imp$ximp,1)

#polytomous
mirt_pol<-mirt()

cat(capture.output(coef(mirt2, QMC=T)),file="mirtk2_coefs.txt",sep="\n")
cat(capture.output(summary(mirt2)),file="mirtk2_summary.txt",sep="\n")
cat(capture.output(fscores(mirt2, QMC=T)),file="mirtk2fscores.txt",sep="\n")
cat(capture.output(fscores(mirt4,method="MAP"),file="small_mirt1fscores.txt",sep="\n")

fitmirt3<-mirt(gbw,1,...)
```

```
ItemResponseTheoryData<-gbook
names(ItemResponseTheoryData)<-c("Subj","Correct","Item")
write.csv(ItemResponseTheoryData, file="ItemResponseTheoryData.csv")

##########make IRT figures
par(mfrow=c(1,3))

curve((1/(1+exp(-(x-.5)))),main="1PL (Rasch)",from=-4,to=4, ylim=c(0,1), yaxt="n",xaxt="n",xlab=expression(paste(theta)), ylab="P( y= 1 | b= .5)")
axis(2,at=seq(0,1,.1))
axis(1,at=seq(-4,4,1))
segments(.5,0,.5,.5, lty=2)
segments(-5,.5,.5,.5,lty=2)
text(-2,.8,expression(paste(frac(1,1+e^{-(theta-.5)}))))

curve((1/(1+exp(-2*(x-.5)))),main="2PL",from=-4,to=4, ylim=c(0,1), yaxt="n",xaxt="n",xlab=expression(paste(theta)), ylab="P( y=1 | b=.5, a=2 )")
axis(2,at=seq(0,1,.1))
axis(1,at=seq(-4,4,1))
segments(.5,0,.5,.5, lty=2)
segments(-5,.5,.5,.5,lty=2)
text(-2,.8,expression(paste(frac(1,1+e^{ -2(theta-.5)}))))

curve(.2+((1-.2)/(1+exp(-2*(x-.5)))),main="3PL", from=-4,to=4, ylim=c(0,1), yaxt="n",xaxt="n",xlab=expression(paste(theta)), ylab="P( y=1 | b=.5, a=2, c=.2 )")
axis(2,at=seq(0,1,.1))
axis(1,at=seq(-4,4,1))
abline(h=.2,lty=2)
segments(.5,0,.5,.6, lty=2)
segments(-5,.6,.5,.6,lty=2)
text(-2,.8,expression(paste(.2+frac(1-.2,1+e^{-2(theta-.5)}))))
###############################################

#number of total responses
length(dataset$correct)
#number of total correct responses
sum(dataset$correct,na.rm=T)

#number of first tries
num1<-sum(dataset$X1st.pres)
#number of second tries
num2<-sum(dataset$X2nd.pres)
#number of third tries
num3<-sum(dataset$X3rd.pres)
#etc
num4<-sum(dataset$X4th.pres)
num5<-sum(dataset$X5th.pres)
num6<-sum(dataset$X6th.pres)
num7<-sum(dataset$X7th.pres)
num8<-sum(dataset$X8th.pres)
num9<-sum(dataset$X9th.pres)
num10<-sum(dataset$X10th.pres)
#concatenate
attempts<-c(num1,num2,num3,num4,num5,num6,num7,num8,num9,num10)
#show plot
```

```
plot(1:10,attempts,type="l")
####################################################################################

#only mcqs
dataset_mcq<-subset(dataset, type=="choice")
#only certain columns

length(unique(dataset_mcq$prompt))
length(unique(dataset_mcq$quiz_answer_id))

table3=table(dataset_mcq$activity_id,dataset_mcq$prompt)
#total pool of items for each activity?
rowSums(table3 !=0)

table4=table(dataset_mcq$prompt,dataset_mcq$student_id)
#items per student
colSums(table4 !=0)
summary(colSums(table4 !=0))

############################################################################
#data for first attempts only, and for MCQ
firstdata<-subset(dataset, X1st.pres=='1' & type=='choice')
firstdata<-firstdata[-c(6,7,8,9,13:21,24:35)]
#what represents unique questions?
length(unique(firstdata$prompt))
length(unique(firstdata$quiz_answer_id))
#create unique ids for items
firstdata<-transform(firstdata,item_id=as.numeric(factor(prompt)))

#remove rows with NA prompts

#get rid of answers that are not graded 1 or 0
firstdata<-firstdata[-which(is.na(firstdata$correct)),]
firstdata1<-firstdata[c(1,2,3,4,9,11)]

#plot item frequency
hist(firstdata$item_id,breaks=length(firstdata$item_id))
hist(firstdata$activity_id,breaks=100)

length(unique(firstdata$item_id))
#544
length(unique(firstdata$student_id))
#844
length(unique(firstdata$activity_id))
#28

#how many questions for each activity id?
plot(firstdata$activity_id,firstdata$item_id)
table1<-table(firstdata$activity_id,firstdata$item_id)
margin.table(table1,1)
#number of unique questions per activity_id
rowSums(table1 !=0)

table2<-table(firstdata$item_id,firstdata$student_id)

#students per item
```

```
rowSums(table2 !=0)
#items per student
colSums(table2 !=0)

#reshape data to wide format with just id by item (for irt)
firstdata_wide<-reshape(firstdata1[order(firstdata$student_id),], timevar="item_id", idvar=c("eid","student_id","activity_id","quiz_answer_id"),direction="wide")
#lots of missing data because everyone hasn't seen the same questions

library(ltm)
descript(firstdata_wide)

###########################################################################
#time to deal with time; first convert to POSIX
dataset$graded_at<-strptime(dataset$graded_at, "%m/%d/%y %H:%M")
dataset$created_at<-strptime(dataset$created_at, "%m/%d/%y %H:%M")
dataset$updated_at<-strptime(dataset$updated_at, "%m/%d/%y %H:%M")
#I think graded_at and updated_at are the same, but let's check
x<-dataset$updated_at==dataset$graded_at
table(x)
#not quite, got 8 false, 159754 true; where are those falses?
falses<-dataset[which(dataset$graded_at!=dataset$updated_at),]

#"created at" appears to be when students started, while "updated at" appears to be when they stopped
length(unique(dataset$graded_at))
#2245
length(unique(dataset$updated_at))
#3611
length(unique(dataset$created_at))
#3611

table(dataset$type)
#choice    text
#166493   3380
sum(is.na(dataset$type))
#[1] 1906

#there are 14151 rows with correct = NA
sum(is.na(dataset$correct))
#[1] 14151
length(dataset$correct)
#[1] 171779

#create an assignment duration variable
dataset$duration<-dataset$updated_at-dataset$created_at
head(dataset$duration)
str(dataset$duration)
table(dataset$duration)

#calculate time difference between presentations
timebtw12<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,13]=='1'){timebtw12[i]<-dataset[i,26]-dataset[i-1,26]}else {timebtw12[i]<-'NA'}}
timebtw23<-dataset$duration
```

```r
for(i in 1:length(dataset$eid)){
  if(dataset[i,14]=='1'){timebtw23[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw23[i]<-'NA'}
timebtw34<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,15]=='1'){timebtw34[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw34[i]<-'NA'}
timebtw45<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,16]=='1'){timebtw45[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw45[i]<-'NA'}
timebtw56<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,17]=='1'){timebtw56[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw56[i]<-'NA'}
timebtw67<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,18]=='1'){timebtw67[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw67[i]<-'NA'}
timebtw78<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,19]=='1'){timebtw78[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw78[i]<-'NA'}
timebtw89<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,20]=='1'){timebtw89[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw89[i]<-'NA'}
timebtw910<-dataset$duration
for(i in 1:length(dataset$eid)){
  if(dataset[i,21]=='1'){timebtw910[i]<-dataset[i,26]-dataset[i-1,26]}else timebtw910[i]<-'NA'}
dataset<-cbind(dataset,timebtw12, timebtw23, timebtw34, timebtw45, timebtw56, timebtw67, timebtw78, timebtw89, timebtw910)
###########################################################################

sp8<-read.csv("bm1k4w.csv",header=F)
sp8<-read.csv("new_sparfaw8.csv",header=F)
dad8<-cbind(rep(c("C1","C2","C3","C4"),rep(8,4)), rep(seq(1,8,1),4))
dad8<-cbind(rep(c("F1","F2","F3","F4","F5","F6","F7","F8"),rep(535,8)), rep(seq(1,535,1),8))
dad8<-cbind(rep(c("F1","F2","F3","F4","F5","F6","F7","F8","F9","F10","F11","F12","F13","F14","F15","F16","F17","F18","F19","F20","F21","F22","F23","F24","F25","F26","F27"),rep(535,27)), rep(seq(1,535,1),27))
sp8d<-sp8[,ncol(sp8)]
sp8<-sp8[,-ncol(sp8)]
sp8<-as.matrix(sp8)
sp8v<-matrix(sp8,ncol=1,byrow=F)
dad8<-cbind(dad8,sp8v)

library(qgraph)
library(igraph)

g<-graph.edgelist(dad8[,1:2],directed=F)
E(g)$weight=as.numeric(dad8[,3])
plot(g,layout=layout.fruchterman.reingold,edge.width=E(g)$weight)
tkplot(g,edge.width=E(g)$weight/15,vertex.size=1)
```

```r

GradeIdea<-read.csv("GradesAndIdeaDensity2.csv", sep="\t",header=T)
bmdata2<-read.csv("bmdata2new.csv")
bms<-bmdata2[,c("eid","BM_2","correct")]

detach(GradeIdea)
attach(GradeIdea)
GradeIdea<-GradeIdea[(GradeIdea$grade*.85+.15)>.6,]
GradeIdea<-GradeIdea[(GradeIdea$Density93>.1 & GradeIdea$Density211>.1 & GradeIdea$Density232>.2 & GradeIdea$Density442>.1),]
GradeIdea_nona<-missForest(GradeIdea)

pairwise.t.test(x=adat$Density,g=adat$Assignment, p.adjust.method="bonf",paired=T)

adata<-GradeIdea[,c(2,6,9,15,21,27)]
adat<-reshape(adata,varying=c("Density93","Density211","Density232","Density442"),v.names="Density",timevar="Assignment",times=c("Density93","Density211","Density232","Density442"),direction="long")
adat$Assignment<-as.factor(adat$Assignment)

names(adata)[names(adata)=="grade"]<-"Grade"
names(adata)<-c("EID", "Grade", "ID.93", "ID.211","ID.232","ID.442")

aov.out=aov(Density~Assignment+ Error(EID/Assignment),data=adat)
summary(aov.out)

fit93<-lm(GradeIdea$grade~GradeIdea$Density93)
#363, 315, 784, 773, 637
fit211<-lm(GradeIdea$grade~GradeIdea$Density211)
#428, 785, 741 (542,243)
fit232<-lm(GradeIdea$grade~GradeIdea$Density232)
#325,674, 406 (406,16,818)
fit442<-lm(GradeIdea$grade~GradeIdea$Density442)
#637,815,212,730 601
fitall<-lm(GradeIdea$grade~GradeIdea$Density442+GradeIdea$Density93+GradeIdea$Density232+GradeIdea$Density211)

adat$Assignment<-as.factor(adat$Assignment)
library(lme4)
library(missForest)

adat_nona<-missForest(adat)
m1<-lmer(grade~Density+(1+Density|EID),adat)
m2<-lmer(grade~Density+(1|EID),adat)
m1<-lmer(grade~Density+(1|Assignment),adat_sc)
adat_sc<-adat
adat_sc$grade<-scale(adat_sc$grade)
adat_sc$Density<-scale(adat_sc$Density)

library(np)
fitnp93<-npreg(GradeIdea$grade~GradeIdea$Density93)
fitnp211<-npreg(GradeIdea$grade~GradeIdea$Density211)
fitnp232<-npreg(GradeIdea$grade~GradeIdea$Density232)
fitnp442<-npreg(GradeIdea$grade~GradeIdea$Density442)

plot(GradeIdea$grade~GradeIdea$Density93)
```

```
abline(fitnp93)
par(mfrow=c(1,4))
plot(fitnp93,ylim=c(0,1),plot.errors.method="bootstrap")
points(GradeIdea$grade~GradeIdea$Density442)

fitw93<-lm(GradeIdea$grade~GradeIdea$Words93)
fitw211<-lm(GradeIdea$grade~GradeIdea$Words211)
fitw232<-lm(GradeIdea$grade~GradeIdea$Words232)
fitw442<-lm(GradeIdea$grade~GradeIdea$Words442)
plot(GradeIdea$grade~GradeIdea$Words93)

ChatData<-read.csv("chat_data.csv",header=T,sep="\t")
#calculate the number of contributions
numContribsTot<-as.data.frame(table(ChatData$eid[ChatData$role=="Student"]))

exptd<-as.data.frame(table(ChatData$eid[ChatData$role=="Student"& ChatData$topic_id==64]))
exptd<-exptd[exptd$Freq>0,]
exptdq<-as.data.frame(table(bms$eid[bms$correct=="1" & (bms$BM_2==2 | bms$BM_2==3)]))
exptd<-merge(exptd,exptdq,by="Var1")
names(exptd)<-c("eid","Chats","BM_score")
plot(exptd$BM_score,exptd$Chats)
big5$BM_score<-big5$BM_score/max(big5$BM_score)

expnp<-npreg(Chats~BM_score,data=exptd,gradients=TRUE)
plot(expnp,plot.errors.method="bootstrap",xaxt=’n’)
npsigtest(expnp)

big5<-as.data.frame(table(ChatData$eid[ChatData$role=="Student"& ChatData$topic_id==311]))
big5<-big5[big5$Freq>0,]
big5q<-as.data.frame(table(bms$eid[bms$correct=="1" & (bms$BM_2==18)]))
big5<-merge(big5,big5q,by="Var1")
names(big5)<-c("eid","Chats","BM_score")
big5$BM_score<-big5$BM_score/max(big5$BM_score)

expnp<-npreg(Chats~BM_score,data=big5,gradients=TRUE)
plot(expnp,plot.errors.method="bootstrap",xaxt=’n’)
npsigtest(expnp)

learning<-as.data.frame(table(ChatData$eid[ChatData$role=="Student"& (ChatData$topic_id==256 | ChatData$channel_name=="classChat-4-760" | ChatData$channel_name=="classChat-4-746")]))
learning<-learning[learning$Freq>0,]
learning<-merge(GradeIdea[,1:6],learning,by.x="EID",by.y="Var1")
plot(learning$Freq,learning$grade)
learningq<-as.data.frame(table(bms$eid[bms$correct=="1" & (bms$BM_2==15)]))
learning<-merge(learning,learningq,by="Var1")
plot(learning$Freq.y,learning$Freq.x)

expnp<-npreg(Chats~BM_score,data=learning,gradients=TRUE)
plot(expnp,plot.errors.method="bootstrap",xaxt=’n’)
npsigtest(expnp)

chatlm<-lm(grade~Freq,data=bigd)
chatnp<-npreg(Grade~Freq,data=bigd,gradients=TRUE)
```

```
plot(chatnp,plot.errors.method="bootstrap",xaxt='n')
npsigtest(chatnp)

phobia<-as.data.frame(table(ChatData$eid[ChatData$role=="Student"& (ChatData$channel_name=="classChat-4-2236" | ChatData$channel_name=="classChat-4-2223")]))
phobia<-phobia[phobia$Freq>0,]
therapy<-as.data.frame(table(ChatData$eid[ChatData$role=="Student"& ChatData$channel_name=="classChat-4-2223"]))
therapy<-therapy[therapy$Freq>0,]

bigd<-merge(GradeIdea,numContribsTot,by.x="EID",by.y="Var1")
names(bigd)[names(bigd)=="grade"]<-"Grade"
names(bigd)[names(bigd)=="Number of Chat Contributions"]<-"Freq"

plot(bigd$Grade,bigd$Freq,xlab="Number of Chat Contributions")

chatlm<-lm(grade~Freq,data=bigd)
chatnp<-npreg(Grade~Freq,data=bigd,gradients=TRUE)
plot(chatnp,plot.errors.method="bootstrap",xaxt='n')
npsigtest(chatnp)

#crosstabs for number of contributions per topic per person
bigtable<-table(ChatData$topic_id, ChatData$eid)
freqChatEid<-data.frame(bigtable)

plot(freqChatEid)
#wordclouds #notworking...

library(tm)
library(wordcloud)
library(psych)
source93<-VectorSource(GradeIdea$resp442)
corpus93=Corpus(source93)
corpus93=tm_map(corpus93,content_transformer(tolower))
corpus93=tm_map(corpus93,removeWords,stopwords("english"))
corpus93=tm_map(corpus93,removePunctuation)
corpus93=tm_map(corpus93,removeNumbers)
corpus93=tm_map(corpus93,removeWords,stopwords("english"))
corpus93=tm_map(corpus93,removeWords,c("just","really"))
#corpus93=tm_map(corpus93,stemDocument)
dtm<-DocumentTermMatrix(corpus93)
dtm<-as.matrix(dtm)
wfreq<-colSums(dtm)
wfreq<-sort(wfreq,decreasing=T)
words<-names(wfreq)
wordcloud(words[1:150], wfreq[1:150])

############################################################
#creating images from Bahrick articles
y<-function(x){6.30+0.94*x-6.09*x*x+2.96*x*x*x-.41*x*x*x*x+3.88*lvl-.14*lvl*lvl-5.86*grd+1.55*grd*grd+0.15*1.41}

lvl=5; grd=2.0
pcol<-c("red","blue","forestgreen")
x<-seq(0,log(40),by=log(40)/8)
lvl=5;
```

```r
plot(exp(x),y(x), type='o', ylim=c(0,20), col=pcol[1],pch=19, xlab="Retention interval (years)", ylab="# of original items recalled")
lvl=3;
points(exp(x),y(x),type="o",pch=15,col=pcol[2])
lvl=1;
points(exp(x[exp(x)<=15]),y(x[exp(x)<=15]), col=pcol[3],type="o",pch=17,xlim=c(0,10))
legend("topright",legend=c("5 semesters", "3 semesters", "1 semester"), pch=c(19,15,17), col=pcol,title="Initial learning:")

#created from figure
y2<-function(x){-6.751+.187*x-1.058*x*x-.023*x*x*x+algnum*9.525+hsmnum*4.825-4.922*canum+0.852*cmnum+1.103*relnum+14.360*malev+9.181*grade+0.874*(relnum^2)-3.356*(malev^2)+rscale*2.174-rtime*3.48+gender*.392
  -2.046*algnum*atime+.899*hsmnum*atime+1.751*cmnum*atime+0.528*relnum*atime+2.688*malev*atime-1.134*grade*atime-0.446*rscale*atime-0.315*rtime*atime-2.011*gender*atime}

rscale=0;rtime=1;canum=0
hsmnum=1; algnum=1;cmnum=1;relnum=1;grade=4.0;malev=1; gender=0
atime=seq(0,log(50),by=log(50)/8)
plot(exp(atime),y2(atime),ylim=c(-100,0),type="o", col=pcol[1],pch=16,xlab="Retention interval (years)",ylab="% decline on Algebra I retention test")

#rscale=0;rtime=1;canum=0
#hsmnum=2; algnum=2;cmnum=0;relnum=0;grade=4.0;malev=-3; gender=0
#atime=seq(0,log(50),by=log(50)/8)
#points(exp(atime),y2(atime),pch=18,type="o")

rscale=0;rtime=1;canum=0
hsmnum=1; algnum=1;cmnum=0;relnum=0;grade=4.0;malev=-3; gender=0
atime=seq(0,log(50),by=log(50)/8)
points(exp(atime),y2(atime),pch=17,col=pcol[3],type="o")

rscale=0;rtime=1;canum=0
hsmnum=1; algnum=1;cmnum=1;relnum=1;grade=4.0;malev=0; gender=0
atime=seq(0,log(50),by=log(50)/8)
points(exp(atime),y2(atime),pch=15,col=pcol[2],type="o")

legend("bottomleft",col=pcol,legend=c("above calculus", "calculus", "1 below calculus"), pch=c(16,15,17))
#created from Figure 2 on Bahrick and Hall, 1991
```

Bibliography

[1] H. P. Bahrick, P. O. Bahrick, and R. P. Wittlinger. Fifty years of memory for names and faces: A cross-sectional approach. *Journal of Experimental Psychology: General*, 104(1):54–75, March 1975.

[2] Harry P. Bahrick. Semantic memory content in permastore: Fifty years of memory for Spanish learned in school. *Journal of Experimental Psychology: General*, 113(1):1–29, 1984.

[3] Harry P. Bahrick. Long-Term Maintenance of Knowledge. In Endel Tulving, editor, *The Oxford Handbook of Memory*, pages 347–362. Oxford University Press, 2000.

[4] Harry P. Bahrick. *Life-span Maintenance of Knowledge*. Essays in cognitive psychology. Psychology Pr, 2012.

[5] Harry P. Bahrick, Lorraine E. Bahrick, Audrey S. Bahrick, and Phyllis E. Bahrick. Maintenance of Foreign Language Vocabulary and the Spacing Effect. *Psychological Science*, 4(5):316–321, 1993.

[6] Harry P. Bahrick and Lynda K. Hall. Lifetime maintenance of high school mathematics content. *Journal of Experimental Psychology: General*, 120(1):20–33, 1991.

[7] Harry P. Bahrick and Elizabeth Phelphs. Retention of Spanish vocabulary over 8 years. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 13(2):344–349, 1987.

[8] Ryan S. J. d Baker and Kalina Yacef. The State of Educational Data Mining in 2009: A Review and Future Visions. *JEDM - Journal of Educational Data Mining*, 1(1):3–17, October 2009.

[9] A. Beck and M. Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, January 2009.

[10] Robert A. Bjork, John Dunlosky, and Nate Kornell. Self-Regulated Learning: Beliefs, Techniques, and Illusions. *Annual Review of Psychology*, 64(1):417–444, 2013.

[11] Cati Brown, Tony Snodgrass, Susan J. Kemper, Ruth Herman, and Michael A. Covington. Automatic measurement of propositional idea density from part-of-speech tagging. *Behavior Research Methods*, 40(2):540–545, May 2008.

[12] Andrew C. Butler. Repeated testing produces superior transfer of learning relative to repeated studying. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 36(5):1118–1133, September 2010.

[13] Andrew C. Butler and Nathaniel D. Raley. The Future of Medical Education: Assessing the Impact of Interventions on Long-Term Retention

and Clinical Care. *Journal of Graduate Medical Education*, 7(3):483–485, September 2015.

[14] Shana K. Carpenter. Testing Enhances the Transfer of Learning. *Current Directions in Psychological Science*, 21(5):279–283, October 2012.

[15] Shana K. Carpenter, Nicholas J. Cepeda, Doug Rohrer, Sean H. K. Kang, and Harold Pashler. Using Spacing to Enhance Diverse Forms of Learning: Review of Recent Research and Implications for Instruction. *Educational Psychology Review*, 24(3):369–378, August 2012.

[16] Shana K. Carpenter and Jonathan W. Kelly. Tests enhance retention and transfer of spatial learning. *Psychonomic Bulletin & Review*, 19(3):443–448, February 2012.

[17] Mark Carrier and Harold Pashler. The influence of retrieval on retention. *Memory & Cognition*, 20(6):633–642, November 1992.

[18] Nicholas J. Cepeda, Harold Pashler, Edward Vul, John T. Wixted, and Doug Rohrer. Distributed practice in verbal recall tasks: A review and quantitative synthesis. *Psychological Bulletin*, 132(3):354–380, 2006.

[19] R. Philip Chalmers. mirt: A multidimensional item response theory package for the r environment. *Journal of Statistical Software*, 48(1):1–29, 2012.

[20] M. A. Covington. Idea density #x2014; A potentially informative characteristic of retrieved documents. In *IEEE Southeastcon 2009*, pages 201–203, March 2009.

[21] Michael A. Covington. Idea density: A potentially informative characteristic of retrieved documents. In *IEEE Southeastcon 2009*, pages 201–203, March 2009.

[22] Eugne J F M Custers and Olle T J ten Cate. Very long-term retention of basic science knowledge in doctors after graduation. *Medical Education*, 45(4):422–430, April 2011.

[23] Brigid M. Dolan, Maria A. Yialamas, and Graham T. McMahon. A Randomized Educational Intervention Trial to Determine the Effect of Online Education on the Quality of Resident-Delivered Care. *Journal of Graduate Medical Education*, 7(3):376–381, July 2015.

[24] William Kaye Estes, Alice F. Healy, Stephen Michael Kosslyn, and Richard M. Shiffrin. *From Learning Theory to Connectionist Theory.* Psychology Press, 1992.

[25] Arthur I. (Arthur Irving) Gates. *Recitation as a factor in memorizing.* New York, The Science press, 1917.

[26] Jeffrey D. Karpicke, Andrew C. Butler, and Henry L. Roediger III. Metacognitive strategies in student learning: Do students practise retrieval when they study on their own? *Memory*, 17(4):471–479, April 2009.

[27] Jeffrey D. Karpicke and Henry L. Roediger. The Critical Importance of Retrieval for Learning. *Science*, 319(5865):966–968, February 2008.

[28] Jeffrey D. Karpicke and Henry L. Roediger III. Repeated retrieval during learning is the key to long-term retention. *Journal of Memory and Language*, 57(2):151–162, August 2007.

[29] Nate Kornell, Robert A. Bjork, and Michael A. Garcia. Why tests appear to prevent forgetting: A distribution-based bifurcation model. *Journal of Memory and Language*, 65(2):85–97, August 2011.

[30] Andrew S. Lan, Christoph Studer, and Richard G. Baraniuk. Time-varying Learning and Content Analytics via Sparse Factor Analysis. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 452–461, New York, NY, USA, 2014. ACM.

[31] Andrew S. Lan, Andrew E. Waters, Christoph Studer, and Richard G. Baraniuk. Sparse Factor Analysis for Learning and Content Analytics. *J. Mach. Learn. Res.*, 15(1):1959–2008, January 2014.

[32] Mark A. McDaniel, Julie M. Bugg, Yiyi Liu, and Jessye Brick. When does the test-study-test sequence optimize learning and retention? *Journal of Experimental Psychology: Applied*, 21(4):370–382, 2015.

[33] James W. Pennebaker, Cindy K. Chung, Joey Frazee, Gary M. Lavergne, and David I. Beaver. When Small Words Foretell Academic Success:

The Case of College Admissions Essays. *PLOS ONE*, 9(12):e115844, December 2014.

[34] James W. Pennebaker, Samuel D. Gosling, and Jason D. Ferrell. Daily Online Testing in Large Classes: Boosting College Performance while Reducing Achievement Gaps. *PLOS ONE*, 8(11):e79774, November 2013.

[35] Katherine A. Rawson and Walter Kintsch. Rereading Effects Depend on Time of Test. *Journal of Educational Psychology*, 97(1):70–80, 2005.

[36] M.D. Reckase. *Multidimensional Item Response Theory.* Springer New York, New York, NY, 2009.

[37] Henry L. Roediger III and Andrew C. Butler. The critical role of retrieval practice in long-term retention. *Trends in Cognitive Sciences*, 15(1):20–27, January 2011.

[38] C. Romero and S. Ventura. Educational data mining: A survey from 1995 to 2005. *Expert Systems with Applications*, 33(1):135–146, July 2007.

[39] Richard A. Schmidt and Robert A. Bjork. New Conceptualizations of Practice: Common Principles in Three Paradigms Suggest New Concepts for Training. *Psychological Science*, 3(4):207–217, July 1992.

[40] Snowdon, D.A, Kemper, S.J., Mortimer, J.A., Greiner, L.H., Wekstein, D.R., and Markesbery, W.R. Linguistic ability in early life and cognitive function and alzheimer's disease in late life: Findings from the nun study. *JAMA*, 275(7):528–532, February 1996.