Jonsdottir, A. H., Jakobsdottir, A., & Stefansson, G. (2015). Development and Use of an Adaptive Learning Environment to Research Online Study Behaviour. *Educational Technology & Society, 18* (1), 132–144.

Development and Use of an Adaptive Learning Environment to Research Online Study Behaviour

Anna Helga Jonsdottir*, Audbjorg Jakobsdottir and Gunnar Stefansson
Department of Natural Sciences, University of Iceland, Dunhaga 5, 107 Reykjavik, Iceland // ahj@hi.is // audbjorg@yahoo.com // gunnar@hi.is
*Corresponding author

(Submitted December 6, 2013; Revised April 5, 2014; Accepted July 1, 2014)

ABSTRACT

This paper describes a system for research on the behaviour of students taking online drills. The system is accessible and free to use for anyone with web access. Based on open source software, the teaching material is licensed under a Creative Commons License. The system has been used for computer-assisted education in statistics, mathematics and fishery science. It offers a unique way to structure and link material, including interactive drills with a purpose of increasing learning rather than mere evaluation. It is of interest to investigate what affects how students learn in such an environment, for example how the system entices students to continue to work. One research question is therefore: When do learners decide to stop requesting drill items? A case study has been conducted including 316 students in an undergraduate course in calculus. Analysis of the data showed that the probability of stopping increases with higher grades but decreases with increased difficulty and the number of questions answered. Also, the probability of stopping decreases if the last question was answered correctly in comparison to when the last question was answered incorrectly.

Keywords

Web-based education, Statistics education, IRT, AIWBES, The tutor-web

Introduction

With the increasing number of web-based educational systems and learning environments several types of systems have emerged. These include the learning management system (LMS), learning content management system (LCMS), virtual learning environment (VLE), course management system (CMS) and Adaptive and intelligent web-based educational systems (AIWBES). The terms VLE and CMS are often used interchangeably, CMS being more common in the United States and VLE in Europe.

The LMS is designed for planning, delivering and managing learning events, usually adding little value to the learning process nor supporting internal content processes (Ismail, 2001). A VLE provides similar service, adding interaction with users and access to a wider range of resources (Piccoli, Ahmad & Ives, 2001). The primary role of a LCMS is to provide a collaborative authoring environment for creating and maintaining learning content (Ismail, 2001). Classes taught on these platforms are accessible through a web-browser but are usually private, i.e., only individuals who are registered for a class have access to the password-protected website.

A number of content providers can be found on the web. Even though they are not educational systems per se, linking them to learning systems would make the content available to a larger audience and save work on creating material within the learning systems. Examples of existing content providers are Khan Academy and Connexions. A number of academic institutions have also made educational material available, including MIT OpenCourseWare and Stanford Engineering Everywhere.

Many systems are merely a network of static hypertext pages (Brusilovsky, 1999) but adaptive and intelligent web-based educational systems (AIWBES) use a model of each student to adapt to the needs of that student (Brusilovsky & Peylo, 2003). These systems tend to be subject-specific because of their structural complexity and therefore do not provide a broad range of content. The first AIWBES systems were developed in the 1990s. These include ELM-ART (Brusilovsky, Schwartz & Weber, 1996; Weber & Brusilovsky, 2001) and the AHA! system (De Bra & Calvi, 1998). Since then, many interesting systems have been developed, many of which focus on a specific subject, often within computer science. Examples of AIWBES systems used in computer science education are SQL-Tutor (Mitrovic, 2003), ALEA (Bieliková, 2006), QuizGuide (Brusilovsky & Sosnovsky, 2005; Brusilovsky, Sosnovsky &

Shcherbinina, 2004) and Flip (Barla et al., 2010) which includes an interesting way of allocating quiz questions to students (discussed further in the following section). AIWBES systems can be found in other fields such language teaching (Chen, Lee & Chen, 2005; Heift & Nicholson, 2001) and teaching modelling of dynamic systems (Zhang et al., 2014) to name some. Systems including competitive web-based drill games are also available, with an overview presented in González-Tablas, Fuentes, Hernández-Ardieta and Ramos (2013).

The goal of the project described here is to build an AIWBES including the functionalities of an LCMS. The system should be open to everyone having access to the web and provide broad educational content including interactive drills with the primary purpose of enhancing learning. Intelligent methods will be used for item allocation in drills and for directing students toward appropriate material. As discussed in Romero and Ventura (2007), great possibilities lie in the use of educational datamining. The behaviour of the students in the system are logged so the system provides a testbed for research on web-assisted education such as drill item selection methods.

It has been described earlier how students tend to strive for higher grades in similar systems (Stefansson, 2004). The present paper considers these drivers more explicitly, namely how the student behaviour, including stopping times, reflects their achievements and likely immediate performance, as predicated by system design.

Item allocation in educational systems

Numerous educational systems with different functionality are available today as discussed in the previous section. The majority permits the creation of quiz questions and administration of quizzes for evaluation or to enhance learning. In most systems these quizzes are static, where the instructor has chosen a fixed set of items. In some cases items are selected randomly from an available question pool so that students are not all presented with the same set of questions. In this section, methods for allocating quiz questions or drill items to learners are discussed.

Although there are a number of educational web-based systems that use intelligent and/or adaptive methods for estimating learner's knowledge in order to provide personalized content or navigation (Barla et al., 2010) only a few systems use adaptive and/or intelligent methods for item allocation (adaptive item sequencing). Even though adaptive item sequencing is not common in educational systems, it has been used in Computerized Adaptive Testing (CAT) (Wainer, 2000) for decades. In CAT the test is tailored to the examinee's ability level by means of Item Response Theory (IRT).

IRT (Lord, 1980) is the framework used in psychometrics for the design, analysis, and grading of computerized tests to measure abilities. It has been used extensively in CAT for estimating students abilities. Within the IRT framework, several models have been proposed for expressing the probability of observing a particular response to an item as a function of some characteristic of the item and the ability of the student, the Rasch model being a common one (Wright, 1977). Another, slightly more complicated model, is the three parameter logistic model, or the 3PL model, including a difficulty parameter β, a discrimination parameter α and a guessing parameter c. The Point Fisher Information (PFI) is then used to select the most informative item in the pool, that is the item that minimizes the variance in the ability estimate. Using IRT, a test developer is able to have items that can discriminate students along a continuum of the hypothesized latent construct. However, IRT requires a large sample size for item calibration (i.e., getting estimates for the parameters of the model) and thus it is typically not done in the classroom. As an example of a system using this technique is the SIETTE system (Conejo et al., 2004), a web-based testing system (i.e., not used for learning purposes).

Research on the application of IRT in learning environments is largely absent (Wauters, Desmet & Van Den Noortage, 2010). Review of available literature found only one system using IRT for adaptive item sequencing with the main focus on enhancing learning, the web-based programming learning system Flip (Barla et al., 2010) developed within the PeWePro1 project (Bieliková and Návrat, 2009). The system uses, among other methods, IRT to select questions with adequate difficulty using the 3PL model, but the parameters (α and β) are set manually for each question. Experiments using Flip showed remarkable improvements in test results (Barla et al., 2010).

System description—the tutor-web

The tutor-web system has been designed by considering four major requirements:

- to be open source and domain/subject independent
- to provide a wide range of open content through a web browser
- to use intelligent methods for item allocation (and grading), amenable to research
- to function as a LCMS

The system has been in development for several years. A pilot version, written only in HTML and Perl, is described in Stefansson (2004). The new implementation described below incorporates fresh approaches to individualized education. It is written in Plone (Nagle, 2010) which is an open source content management system with a usability focus, written in Python on top of the ZODB object database. Plone is flexible, customisable and extended with packages from a worldwide community. It is popular with educational content providers, powering Connexions (http://cnx.org/), MIT's OpenCourseWare (http://ocw.mit.edu/) as well as many OpenCourseWare projects, based on the eduCommons (http://educommons.com/) system.

The educational material available within the tutor-web covers wide areas within mathematics and statistics, although only one course is analysed in this study. Examples of use in other fields include fishery science, earth science and computer science. The system could equally well be used in fields as diverse as linguistics, religion, psychology and English. This contrasts several systems which address very special skills, some named above.

The system offers a unique way to structure and link together teaching material and organize it into manageable units both for instructors and students. A well-defined structure enables instructors to construct, share and re-use material and provides a single repository of teaching material for students minimizing time otherwise wasted on imperfect searching and browsing and eradicating any format/incompatibility issues. Additionally, interactive drills have a primary purpose of increasing learning, placing evaluation a distant second. Though the tutor-web is not originally designed as a remote-learning system it can be used as such if desired. The system also provides a testbed for research on web-assisted education such as drill item selection methods.

The tutor-web system is based solely on generic formats and open source software to provide unrestricted usage of material by institutions of limited resources without overheads in terms of software or licenses. The teaching material is licensed under the Creative Commons Attribution-ShareAlike License (http://creativecommons.org/) and is accessible to anybody having a web-browser and web access. Instructors anywhere can obtain free access to the system for exchanging and using teaching material while students have free access to its educational content including self-evaluation in form of drills.

Content structure

The teaching material is organized into a tree (fig. 1) with departments, courses, tutorials, lectures and slides. The different departments can be accessed from the tutor-web homepage (http://tutor-web.net).

The basic component of the tutor-web is the slide. Students can browse through the slides and view related material relevant to a given slide. Instructors can use the slides as source material for presentations in the classroom. Although electronic slides are useful for presentation purposes, often they are not sufficient material when a course is taught. Typically more detailed, examples or even extensive handouts may be needed. Such additional material can be linked to any slide on the tutor-web. As in a classroom, a typical lecture, including several slides, should be constructed around a specific subject to aid and focus the students understanding on that particular subject.

A tutorial typically contains several lectures and should be based on a distinctive topic. A tutorial can belong to more than one course and should be built up around a single theme. For example, a tutorial on simple linear regression could both be a part of a general course on regression and an introductory statistics course. Having the tutorials allows the student to complete a portion of the material and perform self-evaluation based on smaller blocks than a complete course.

The system uniquely uses the modularity and traceability of content so the instructor can easily demonstrate how examples and images are derived: An image based on data can be drawn in the system using the statistical environment R (R Development Core Team, 2011). Normally such an image is presented statically on a screen for a class. Here, however, the R plotting commands and the data are stored as objects in the system, automatically producing PDF or HTML slides. The student can view the underlying data and R code, making the system an ideal tool to teach not only model output but also modelling.

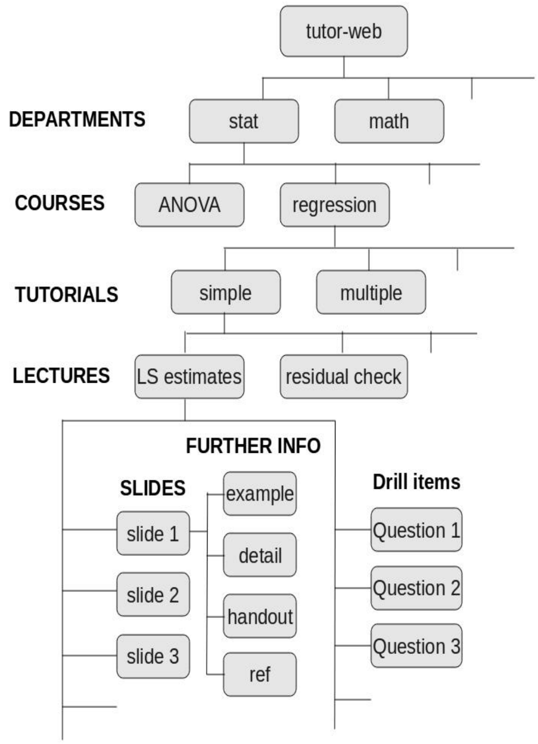

Figure 1. The structure of the tutor-web

One goal of the tutor-web project is to make available a repository of educational material for a BSc degree in mathematics and an MSc degree in applied statistics. Some courses are ready for general use with slides, handouts and questions while others are only placeholders waiting to be filled up with material. In addition to university courses, complete high school mathematics tutorials in Icelandic and English are available with over 2000 drill items.

Drills and item selection

Drills (items) are grouped together so they correspond to material within a lecture. These will be termed “drills” rather than “quizzes” to indicate the emphasis on increasing learning. The drills are multiple choice and the author can choose the number of answers and provide detailed explanation of the correct answer shown to the learners after answering a drill. A drill in the tutor-web system differs from the typical classroom testing methods where a student normally answers a given number of questions during a specific time period. In the tutor-web a student can dynamically start or re-enter a drill, one question at a time and may attempt the drill at his/her leisure, although an instructor might decide on a time limit for recording results.

The intuitive style of the drill in the tutor-web encourages students to improve their results and learn from their mistakes. The learners are provided with knowledge of correct response feedback after answering a drill along with elaborative feedback if provided by the author of the drill. Studies have shown that frequent feedback given to students yields substantial learning gains (Black and Wiliam, 1998). In live applications the students have been encouraged to request answers repeatedly until a decent grade is obtained. Students who do not know the material can test their knowledge and, upon finding it wanting, go back to the online text or textbooks to come back to the drill at a later date.

In the original version of the tutor-web system, drill items within the same lecture were selected randomly with uniform probability (Stefansson, 2004). In the current version of the system a probability mass function that depends on the grade of the student is used to select items of appropriate difficulty. Here, the difficulty of a question is simply calculated as the ratio of incorrect responses to the total number of responses. The questions are then ranked according to their difficulty, from the easiest question to the most difficult one.

A student with a low grade should have higher probability of getting the lowest ranked questions while a student with a high grade should have higher probabilities of getting the most difficult questions. The mass of the probability function should therefore move towards the difficult questions as the grade goes up. The probability mass functions are shown in figure 2 for a lecture with 100 drill items, based on the implemented discrete forms of the beta distribution. Here, beginning student (with a score 0) receives easy items with high probability. As the grade increases the mode of the probability mass functions shifts to the right until the student reaches a top score resulting in high probability of getting the most difficult items.

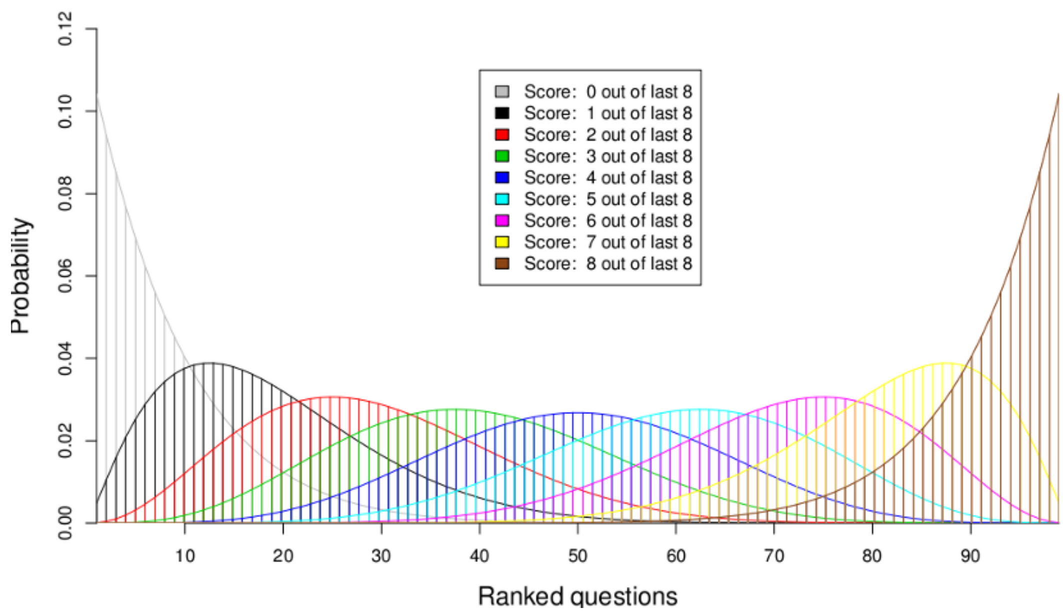

Figure 2. Probability mass functions for item allocation in a lecture with 100 questions

Grading

Although the main goal of the quizzes in the tutor-web is learning there is a need to evaluate the student's performance. The drills are of arbitrary length since the students are permitted to continue to answer questions until they (or the instructor) are satisfied. Because of this, issues regarding how to compute the grade arise.

Currently a student gets one point for answering a question correctly and -1/2 for an incorrect answer. Since the purpose is to enhance learning and thus improve the grade, only the last eight answers are used when the grade is calculated for each lecture. Old sins are therefore forgotten. The student can track the grade and thus monitor personal progress with a single click.

Users and access

Four types of users are defined in the tutor-web: Regular users, students, teachers and managers. There is open access for regular users (anybody having access to the web) to browsing and downloading of material. However, in order to take drills the user needs to log in to the system and become a tutor-web student. When a user initially signs up, the requirement is to provide a full name, a valid email address, choose a unique user name and agree to data being used for research purposes.

Teachers are editors of tutorials and have the ability to edit and insert material as well as quizzes. They also have access to drill results. Managers have the same privileges as instructors with the additional authority to add and edit Departments and Courses and give teacher rights.

Viewing educational material

There are three different ways for a tutor-web user to view the teaching material (fig. 3):

- through a web browser
- as a collection of lecture slides
- as a tutorial printout

The first approach is the simplest one for a student wising to access the educational material. The student simply needs to open the home-page (http://tutor-web.net) and select a department to see courses containing several tutorials and lectures. The student can then enter a lecture to browse through, slide by slide. As an example, one can enter the Math department, click on Computing and calculus for applied statistics (course), Some notes on statistics and probability (tutorial), Multivariate probability distributions (lecture) and Joint probability distribution (slide).

Once “in” a lecture, a PDF document can be downloaded including all the slides for the lecture.

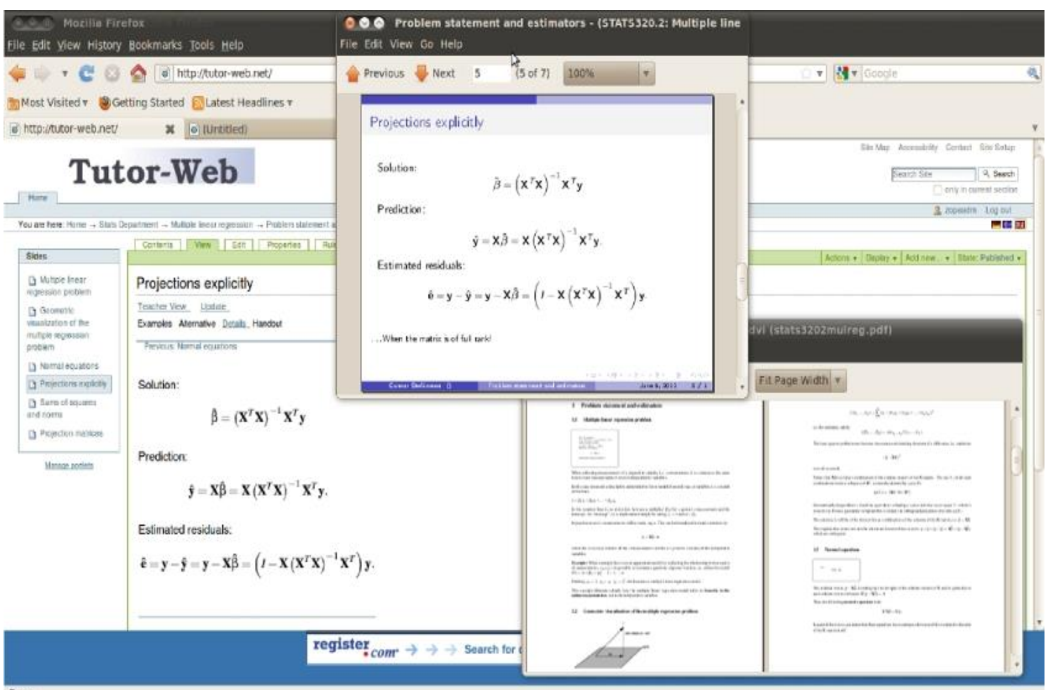

Figure 3. Different views into the database of teaching material in the tutor-web

These PDF slides are made with the LaTeX package Beamer (Tantau et al., 2011) and should be ready for classroom use.

A third way of viewing material is on the tutorial level. Users can download a PDF document including all lectures belonging to that tutorial. Each slide is presented as a figure in this document along with all additional material attached to them providing a handout including all relevant information. In a fully developed tutorial this corresponds to a complete textbook.

The length of earthworms in a certain garden follows a normal distribution with mean 11cm and standard deviation 1.2. If an earthworm is picked at random form the garden what is the probability that it is longer than 12 cm?

a) 0.7967

b) 0.2633

c) 0.8333

d) 0.2033

We need $P(X > 12)$ where $X \sim N(11, 1.2^2)$.

Start by standardizing:

$$z = \frac{12-11}{1.2} = 0.83$$

We use a normal dist. table and see that for $z = 0.83$ we have $\Phi(z) = 0.7967$. Remember that $\Phi(z) = P(Z < z)$.

$$P(X > 12) = 1 - P(X < 12) = 1 - P(Z < 0.83) = 1 - 0.7967 = 0.2033$$

R-command: 1-pnorm(12,11,1.2)

Figure 4. Explanation of the correct answer is given after the student answers a question

The tutor-web drills are accessible within a lecture. When entering one, a Drill button appears which opens a new tab with the first question when pushed. After answering the question the correct answer is indicated along with some explanation. An example is shown in figure 4. The question is taken from a basic course in statistics. The material belonging to that course can be found by choosing the Stats department from the welcoming screen and from there Introductory statistics.

Adding material and content formats

Teaching material can easily be added to the system through a web-browser. It is important that text-based content as well as mathematical equations and figures are correctly displayed and easily manipulated in a standard browser. To achieve this, several predefined content formats are permitted within the system.

Managers can create departments and courses from the tutor-web homepage. After entering a department teachers can create tutorials that then are linked to one or more courses. Within a tutorial, teachers can subsequently add lectures and later slides. Departments, courses tutorials and lectures are simply collection of slides so they require little more than a name.

After creating a lecture, a tutor-web teacher can create a slide. It can consist of a title and three basic units, the main text, a main figure (graphic) and explanatory text. The format of the main text can be LaTeX (Goossens et al., 1994), plain text, or HTML. The figure(s) can be uploaded files (png, gif or jpeg) or they can be rendered from a text based image format (R-image (R Development Core Team, 2011) or Gnuplot (Williams and Kelley, 2010)). Additional material can be attached to the slides which is available when viewing the material through a browser and in the tutorial printout.

Drill items are grouped together so they correspond to material within a lecture. Questions and answers can be added to the system through a browser or be uploaded from a file. A drill item can have as many answers as desired and there is an option to randomize the order of the answers. The format of the text can be LaTeX or plain text. Questions can therefore include formulas, essential to mathematical instruction. The system also permits the use of the statistical package R (R Development Core Team, 2011) when a question is generated. This allows the generation of similar but not identical data sets and graphs for students to analyse or interpret. Alternatively, a large body of such items can be generated outside the system and then uploaded.

For each item it is possible to put an explanation or solution to the problem along with the question. After a student has submitted his answer the correct answer to the question is displayed along with this explanation.

Case study

It is of particular interest to investigate what affects how students learn in a learning environment, such as the tutor-web. How well a system entices students to continue is a particularly important system feature. One research question is therefore: When do learners decide to stop requesting drill items.

The learners' responses to drill items in the tutor-web can be used for research on online learning. In the following, data from 316 students in an undergraduate course in calculus will be used. The students were requested to answer drill items from several lectures covering limits, derivatives, logarithms, integration, sequences and series. Within each lecture the students were required to answer a minimum of eight questions correctly but were allowed to continue as long as they liked, with the final eight answers counting towards their grade in the course.

Since all request for items from within a lecture are logged, these appear in a sequence $n_l = 1,\ldots,m_l$. Data on stopping times can be obtained by looking at the last request of an item from within a lecture, m_l, and we define $S := I_{n_l = m_l}$ as a 0/1-indicator variable. One can now formally test which other variables relate significantly to this stopping variable.

The empirical distribution function (edf) of the number of attempts students made within each lecture is given in figure 5. Recall that within this system students are free to make as many attempts as they desire. As a result, the distribution is heavily right-skewed. A jump is seen at 8 attempts: Few students stop before 8 attempts but there is a smooth change in the edf from then on.

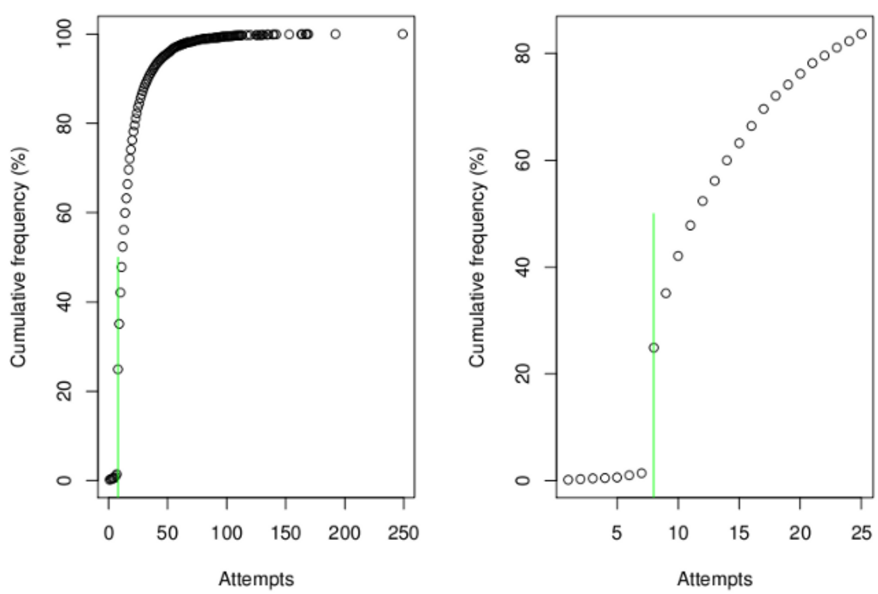

Figure 5. Cumulative distribution (%) of the total number of attempts by each student at each lecture. The right panel expands by only considering attempts 1-25

Finding the drivers

From informal interviews and observations within support sessions that were offered during the semester it seems clear that students have a tendency to continue working within this system until the system reports a high grade. This behaviour is confirmed when looking at the data. Table 1 shows the number of times learners decided to continue requesting drills or decided to stop as a function of the number of correct answers to the last eight items requested within each lecture (the "lecture grade"). As discussed before, only the last eight responses are used to calculate the grade in every lecture and by far, the highest probability (73.3%) is of stopping at the stage when the student has received a full mark (8 out of 8).

Table 1. Stopping percentage (%) as a function of the number of correct in last 8 questions

Number of correct answers	Continue	Stop	Stopping percentage (%)
0	112	1	0.9
1	527	9	1.7
2	2280	30	1.3
3	6612	69	1.0
4	13428	216	1.6
5	20102	438	2.1
6	22482	981	4.2
7	17158	1710	9.1

Consider next the last response before stopping. Table 2 classifies all responses into groups depending on whether this was the final answer within a lecture and whether the answer was correct or not.

Table 2. Classification of answers according to whether the last question was answered correctly (1) of not (0) and whether the student continued or stopped

Last answer	Continue	Stop	Total
0	24665	852	25517
1	59934	7822	67756
Total	84599	8674	93273

Naturally, most cases are in the middle of a sequence so most of the observations fall into the "Continue" column. Only about 10% of terminations follow an incorrect response: Unless an incorrect answer follows a long sequence of correct response, it will be beneficial for the student to request another item if the current answer is incorrect.

It has been suggested in earlier studies with this sort of grading scheme as used here that students may decide to stop early if a run of correct responses is followed by an incorrect answer (Stefansson and Sigurdardottir, 2011). To investigate this, one can consider the fraction of stopping as a function of both the current lecture grade and the most recent grade. This is shown in table 3.

Table 3. Fraction of stopping (%) as a function whether the last questions was answered correctly (0) or not (1) and the number of correct answers in the last eight questions

	0	1	2	3	4	5	6	7
Last = 0	0.9	1.5	1.3	0.8	1.0	2.4	5.4	24.7
Last = 1	-	2.4	1.4	1.4	2.1	2.0	3.9	8.0

It is seen in table 3 that if a run of 7 correct answers is followed by an incorrect answer then the student will in 25% of all cases decide to stop. This is a perfectly logical result since a student who has a sequence of 7 correct and one incorrect, will need another 8 correct answers in sequence to increase the grade. An improved averaging scheme can be used to alleviate this problem. An algorithm which uses the average of the most recent $min(n/2, 30)$ grades after n attempts can give the same full grade after 8 correct responses but a single incorrect answer will get much lower weight as more correct answers come in. An even better option could be to use tapering to downgrade old responses. Given the incentive to work towards a high grade (table 1), this simple change is likely to alleviate the 25% stopping problem.

In principle a generalized linear model (assuming a binomial distribution and logit link) can be fitted to the data shown in table 1. As can be seen in the table, the relationship between the probability of stopping and the grade is not a linear one. A factor variable with four levels (low grade = 0–2, median grade = 3–5, high grade = 6–7 and top grade = 8) will be used. The results are shown in table 4. Here the 0/1 indicator of whether the student stopped is "regressed" against the factor variable grade at that timepoint. The statistical package R (R Development Core Team, 2011) was used for the analysis.

Although this is a useful approach to estimating effects and obtaining an indication of p-values, assumptions can not be assumed to be completely correct since there will be a subject effect (this is considered below). Nevertheless the estimates indicate that the probability of stopping is increasing with higher grade. The difference in probability of stopping between the low grade (0–2) and the median grade (3–5) is not significant. The difference between the high-grade and the low grade as well as the top-grade and low grade are highly significant.

Table 4. Parameter estimates where the 0/1 indicator of whether the student stopped is regressed against the grade at that timepoint

	Estimate	Std. error	z-value	p-value
Intercept	-4.2901	0.1591	-26.97	< 0.0001
Median grade	0.2733	0.1634	1.67	0.09
Hight grade	1.6002	0.160.	9.98	< 0.0001
Top grade	5.3018	0.1613	32.86	< 0.0001

In order to see which other variables are related to the probability of stopping a more complicated model, also including an indicator variable stating if the last answer was correct or not (grade_last), difficulty of the question (diffic) (computed based on the proportion of incorrect responses) as well as the number of attempts (nattl) was fitted to the data. One can see in table 5 that these all appear highly significant. Also notice the sign of the parameter estimates; the probability of stopping increases with higher grades but decreases with increased difficulty and number of questions answered. Also, the probability of stopping decreases if the last question was answered correctly in comparison to when the last question was answered incorrectly.

Table 5. Parameter estimates where the 0/1 indicator of whether the student stopped is regressed against the grade at that timepoint, grade of last answer, item difficulty and number of items answered

	Estimate	Std. error	z-value	p-value
Intercept	-3.7509	0.1645	-22.80	<0.0000
median grade	0.3764	0.1642	2.29	0.0219
high grade	1.7813	0.1631	10.92	<0.0000
top grade	5.4785	0.1656	33.07	<0.0000
grade_last	-0.2801	0.0444	-6.30	<0.0000
diffic	-0.2946	0.1354	-2.18	<0.0296
nattl	-0.0182	0.0010	-18.51	<0.0000

When estimating an even more complicated model, also including the student effect and the lecture (or content) effect it was found that both factors were highly significant.

Conclusions and future work

These simple analyses clearly indicate how one can potentially improve upon the item database and the grading scheme. Given the strong incentive for students to obtain high grades in this kind of system, it is imperative that the system includes a wide variety of very difficult problems, since clearly students continue well into the most difficult parts of the content. It is also clear from this analysis that the behaviour of the students is affected by the grading scheme used here. The effect of different grading schemes on students' behaviour are therefore currently under investigation. Instead of using the last eight responses and giving them equal weight an experiment with tapering schemes is being designed where the most recent answers get more weight than older answers. In these experiments the students will be randomly assigned a value that determines how much weight is put on their most recent answer. Before requesting a new drill item the students are told what their grade will be if they answer the next drill item correctly, with the intent of enticing them to continue requesting drill items and learning more. When large weight is given to the most recent answer and the student answers the item correctly, the student receives an immediate award (large jump up in grade). However if the answer is incorrect the student gets immediate punishment (large jump down in grade). The risk is that the student then stops requesting items but the intent is that information on the potential grade increase will tempt students to continue working within the system.

Currently the tutor-web content provider sets up a concept map by structuring the teaching material within a course into tutorials and lectures within tutorials, providing a linear and logical learning path with respect to prerequisites. Drill items are grouped within lectures and are chosen according to the learners ability resulting in individualized path of drill items for each learner within a specific topic. One of the goals of the tutor-web project is to link appropriate learning material to the drill items so if a student answers an item incorrectly the student will be pointed towards appropriate material to read. These links will be made to material within the system but it would also be interesting to allow users to provide links to other Creative Commons licensed material outside of the system resulting in a completely individualized learning path through an entire course within the system or even the entire web.

The tutor-web is an ongoing research project into the online student's behaviour. An experiment was made in an older version of the system to assess potential difference in student learning while working in the tutor-web versus students handing in written homework (Jonsdottir and Stefansson, 2011). The difference in learning between the groups was not found to be significant but more importantly the confidence bound for the difference was tight, indicating no difference of importance. This implies that time spent on correcting written assignments can be saved by using the tutor-web as homework instead of some written assignments, potentially making a considerable financial difference. The system is under constant development and these results imply that further improvements in learning through the system will enhance it to become better than traditional assignments. Current research is therefore focused on ways to amplify learning rather than changing from one medium to another.

An interesting research question is: How should items be allocated so the students get the most out of the drills? Current CAT techniques tend to use Point Fisher Information (PFI), justified when attempting to evaluate current knowledge since with the PFI the selection criterion is to minimize the variance in estimated subject ability after seeing the item. With the current emphasis on learning, not evaluation, the PFI is no longer central. Although one can in principle still use the PFI methodology, the basic criteria for using the PFI are no longer of interest: Instead, one wants to select each item so as to maximize the amount of learning obtained by showing the item to the subject. In addition, one wants to make sure that this learning is not just transient but committed to long-term memory and, if, at all possible that learning occurs with understanding - not simply learning by rote. Such a mechanism for selecting items could and should take a number of concerns into account.

- If selecting within the current lecture, select an item to give maximum learning
- Within the lecture select easy items for an underachiever, hard items for a good student
- Increase the difficulty level as the student learns, within the lecture
- Select items so that a student can only "successfully" complete the lecture by completing the most difficult items
- Select items from previous lectures (or prerequisite tutorials/courses) if the student can not handle the easiest items within the lecture
- Estimate whether the student is likely to be forgetting earlier material and select earlier items accordingly
- Select an item based on externally supplied metadata on item taxonomy, such as an item containing a cue
- Select items from material which the student has earlier answered incorrectly or is likely to answer incorrectly

Some of these points are already implemented as described in this paper but others form future research projects.

The system has been used by over 2000 students, mostly in courses on statistics and mathematics at the University of Iceland (UI) and the University of Maseno, Kenya. The tutor-web has mainly been used to supplement education in the classroom, but being freely accessible without regard to physical location or registration into a school or university, the potential is much greater. Completion of certain courses has e.g., been used as an entry criterion for PhD applicants at the UI, as an addition to other formal criteria for entering a PhD study. Similarly the system can be used by students lacking in prerequisites.

The tutor-web has considerable potential for low-income regions like Kenya where textbooks are not widely available, and student surveys regarding the tutor-web are highly positive, "I wished to do more" being a typical response (Mokua, Stern, Jonsdottir & Mbasu, 2013). A mobile tutor-web is under development, where the user does not need to be connected to the Internet at all times to answer drill items. This can become a game changer for students in rural areas where Internet access is limited but the number of students with access to smart phones is exploding. Hopefully increase the number of students whose experience will be described by words from the Maseno survey: "Doing maths online was the best experience I ever had with maths."

Acknowledgements

The tutor-web project has been supported by Ministry of Education and the Marine Research Institute of Iceland, the United Nations University, University of Iceland and the EU project MareFrame. The authors would finally like to thank Christopher Desjardins for his helpful comments.

References

Barla, M., Bieliková, M., Ezzeddinne, A., Kramar, T., Simko, M., & Vozár, O. (2010). On the impact of adaptive test question selection for learning efficiency. *Computers & Education, 55*(2), 846–857.

Bieliková, M. (2006). An adaptive web-based system for learning programming. *International Journal of Continuing Engineering Education and Life Long Learning, 16*(1), 122–136.

Bieliková, M., & Návrat, P. (2009). Adaptive web-based portal for effective learning programming. *Communication & Cognition, 42*(1/2), 75–88.

Black, P., & Wiliam, D. (1998). Assessment and classroom learning. *Assessment in Education: Principles, Policy & Practice, 5*(1), 7–74.

Brusilovsky, P. (1999). Adaptive and intelligent technologies for web-based education. *Kunstliche Intelligenz, 13*(4), 19–25.

Brusilovsky, P. & Peylo, C. (2003). Adaptive and intelligent web-based educational systems. *International Journal of Artificial Intelligence in Education*, 13(2), 159–172.

Brusilovsky, P., Schwarz, E., & Weber, G. (1996). Elm-art: An intelligent tutoring system on world wide web. *Intelligent Tutoring Systems* (pp. 261–269). doi: 10.1007/3-540-61327-7_123

Brusilovsky, P., & Sosnovsky, S. (2005). Engaging students to work with self-assessment questions: A study of two approaches. *ACM SIGCSE Bulletin, 37*(3), 251–255.

Brusilovsky, P., Sosnovsky, S., & Shcherbinina, O. (2004). Quizguide: Increasing the educational value of individualized self-assessment quizzes with adaptive navigation support. In J. Nall & R. Robson (Eds.), *World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education* (pp. 1806-1813). Chesapeake, VA: AACE.

Chen, C., Lee, H., & Chen, Y. (2005). Personalized e-learning system using item response theory. *Computers & Education, 44*(3), 237–255.

Conejo, R., Guzmán, E., Millán, E., Trella, M., Pérez-De-La-Cruz, J., & Ríos, A. (2004). Siette: A web-based tool for adaptive testing. *International Journal of Artificial Intelligence in Education, 14*(1), 29–61.

De Bra, P. & Calvi, L. (1998). Aha! an open adaptive hypermedia architecture. *New Review of Hypermedia and Multimedia, 4*(1), 115–139.

González-Tablas, A. I., de Fuentes, J. M., Hernández-Ardieta, J. L., & Ramos, B. (2013). Leveraging quiz-based multiple-prize web tournaments for reinforcing routine mathematical skills. *Educational Technology & Society, 16*(3), 28–43.

Goossens, M., Mittelbach, F., & Samarin, A. (1994). *The LaTeX Companion*. Citeseer. Reading, MA: Addison-Wesley.

Heift, T., & Nicholson, D. (2001). Web delivery of adaptive and interactive language tutoring. *International Journal of Artificial Intelligence in Education, 12*(4), 310–325.

Ismail, J. (2001). The design of an e-learning system: Beyond the hype. *The Internet and Higher Education, 4*(3), 329–336.

Jonsdottir, A., & Stefansson, G. (2011). Enhanced learning with web-assisted education. In D. Kaplan, *JSM Proceedings*, Section on Statistical Education (pp. 3964–3975). Alexandria, VA: American Statistical Association.

Lord, F. (1980). *Applications of item response theory to practical testing problems*. L. Hillsdale, NJ: Erlbaum Associates.

Mitrovic, A. (2003). An intelligent sql tutor on the web. *International Journal of Artificial Intelligence in Education, 13*(2–4), 173–197.

Mokua, V., Stern, D., Jonsdottir, A. H., & Mbasu, Z. (2013, June). *Using tutor-web and video making to improve first year service mathematics teaching at Maseno University, Kenya*. Paper presented at the Africme 4, Maseru, Lesotho.

Nagle, R. (2010). *A user's guide to Plone 4*. Houston, TX: Enfold Systems Inc.

Piccoli, G., Ahmad, R., & Ives, B. (2001). Web-based virtual learning environments: A research framework and a preliminary assessment of effectiveness in basic it skills training. *Mis Quarterly*, *25*(4), 401–426.

R Development Core Team (2011). *R: A language and environment for statistical computing*. Vienna, Austria: R Foundation for Statistical Computing.

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

Stefansson, G. (2004). The tutor-web: An educational system for classroom presentation, evaluation and self-study. *Computers & Education*, *43*(4), 315–343.

Stefansson, G., & Sigurdardottir, A. J. (2011). Web-assisted education: From evaluation to learning. *Journal of Instructional Psychology, 38*(1), 47–60.

Tantau, T., Wright, J., & Mileti, V. (December 25, 2013). *The Beamer Class*. Retrieved from http://www.tex.ac.uk/ctan/macros/latex/contrib/beamer/doc/beameruserguide.pdf

Wainer, H. (2000). *Computerized adaptive testing*. Hillsdale, NJ : L. Erlbaum Associates.

Wauters, K., Desmet, P., & Van Den Noortgate, W. (2010). Adaptive item-based learning environments based on the item response theory: possibilities and challenges. *Journal of Computer Assisted Learning*, *26*(6), 549–562.

Weber, G., & Brusilovsky, P. (2001). Elm-art: An adaptive versatile system for web-based instruction. *International Journal of Artificial Intelligence in Education*, *12*, 351–384.

Williams, T., & Kelley, C. (November 12, 2011). *Gnuplot 4.4-an Interactive Plotting Program*. Retrieved from http://www.gnuplot.info/docs_4.4/gnuplot.pdf

Wright, B. D. (1977). Solving measurement problems with the rasch model. *Journal of Educational Measurement*, *14*(2), 97–116.

Zhang, L., VanLehn, K., Girard, S., Burleson, W., Chavez-Echeagaray, M. E., Gonzalez-Sanchez, J., & Hidalgo-Pontet, Y. (2014). Evaluation of a meta-tutor for constructing models of dynamic systems. *Computers & Education*, *75*, 196–217.