Yang, J. C. et al. (Eds.) (2018). Proceedings of the 26th International Conference on Computers in Education. Philippines: Asia-Pacific Society for Computers in Education

Identifying Student Learning Patterns with Semi-Supervised Machine Learning Models

Jeffrey MATAYOSHI[a*] & Eric COSYN[a]
[a]*McGraw-Hill Education/ALEKS Corporation, USA*
*jeffrey.matayoshi@aleks.com

Abstract: One of the benefits of adaptive learning systems is that they allow students to work at their own pace. Because of this, students may exhibit drastically different learning patterns, some of which are symptomatic of misuse or suboptimal use of the system, or simply of possible inadequacy in the system. Identifying such patterns allows the system or the instructor to take corrective action to ensure that students are having a successful learning experience. ALEKS, which stands for "**A**ssessment and **LE**arning in **K**nowledge **S**paces", is a web-based artificially intelligent learning and assessment system. In this work we attempt to identify and classify various learning patterns that students exhibit while working in the ALEKS learning mode. To do this, we first build a set of statistical features for describing the learning behaviors that students exhibit. After using these features to identify an example set of students, we use semi-supervised machine learning techniques combined with an artificial neural network to apply these classifications to the rest of our dataset.

Keywords: Semi-supervised machine learning, adaptive learning, knowledge space theory, artificial intelligence, neural network

1. Introduction

ALEKS, which stands for "**A**ssessment and **LE**arning in **K**nowledge **S**paces", is a web-based artificially intelligent learning and assessment system (Falmagne et al., 2006). The artificial intelligence of ALEKS is a practical implementation of knowledge space theory (KST), a mathematical theory that employs combinatorial structures to model the knowledge of learners (Doignon and Falmagne, 1985; Falmagne et al., 2013; Falmagne and Doignon, 2011). KST has been successfully applied to such subjects as math (Huang et al., 2016; Reddy and Harper, 2013), chemistry (Taagepera and Arasasingham, 2013) and even dance education (Yang et al., 2012). Using KST, the ALEKS system assesses a student's knowledge in a particular academic course, and it then places her at the appropriate place in the course so that she can begin with the material that she is most prepared to learn. In a typical ALEKS course, the bulk of a student's time is spent in this "learning mode", where the student receives targeted practice and instruction on the concepts that she has not yet mastered.

In this paper, we attempt to understand and classify the various learning patterns that students exhibit in the ALEKS learning mode. We start by building a set of statistical features that helps us separate the students with extreme learning patterns from the students with more typical behaviors. Once we have these features, we devise four different learning classes, based on the behaviors that students exhibit, and then manually validate and assign a sample of students to each class. Finally, we use semi-supervised machine learning techniques to combine this small amount of labeled data with the much larger amount of unlabeled data and build a classification model that allows us to label all of the students in our dataset. A first implementation uses a logistic regression classifier. To overcome its observed limitations, our final model uses an artificial neural network.

2. Background

In KST, an *item* is a problem that covers a discrete skill or concept. Each item is composed of many examples called *instances*, which are carefully chosen to be equal in difficulty and cover the same

content. A *knowledge state* in KST is a collection of items that a student could conceivably know at any one time. In other words, a set of items is a knowledge state if one can expect a student to know all of the items in the set, while not knowing any of the items outside the set (after discounting lucky guesses and careless errors). For example, the empty set and full set are always considered valid knowledge states.

A student begins an ALEKS course by taking an initial assessment. The initial assessment is an adaptive assessment designed to determine what the student knows and, based on this information, what he is ready to learn next. We refer the reader to chapters 2 and 8 of Falmagne et al. (2013) for more detailed information on how the ALEKS assessment works.

After the initial assessment, the student enters the ALEKS learning mode and works on the specific material that the system deems he is ready to learn. The ALEKS learning mode functions by giving a student targeted instruction and practice on a single item at a time. The student is presented several instances of the item, and he must correctly answer these instances until he demonstrates mastery of the item. During this time, the student is allowed to view an explanation of the current instance, which instructs him on how to solve the given problem. Once the student has finished viewing the explanation, he is then given a new instance to solve. At any time in the ALEKS learning mode, the student has the option of working on any of the items that the ALEKS system deems him ready to learn.

3. Measures of Student Behavior (Features)

The dataset under consideration from the ALEKS learning mode consists of three types of actions, or events: correct answers, wrong answers, and viewing of the explanation. Based on these actions, the following time-dependent rate functions of student behavior are used for our model.

- Items learned per hour of login time (login hourly learning rate)
- Proportion of actions consisting of using the explanation (login explanation rate)
- Proportion of actions consisting of a correct answer (login correct answer rate)
- Items learned per day in course (daily learning rate)

For each of these rates, we are looking for a single statistic that captures the "steadiness" of a student's performance with respect to that rate. Thus, for each rate function f we compute the score

$$\frac{\int_0^T |f(t) - \mu(f)|dt}{\int_0^T f(t)dt},$$

where t is time and $\mu(f)$ is the mean of the student's rate up to time T. This gives a unitless measure that takes values on the interval [0, 2). It is worth noting that this measure of steadiness does not depend on how a student performs relative to the other students or on the magnitude of the rate. Instead, it only depends on the (time) variation of the rate around its mean. Additionally, we are also interested in how the hourly learning rate evolves over the length of a course. To measure this evolution, a linear regression is fit for each student's hourly learning rate, and the slope of this regression is used as a measure of the student's "learning trend". Combining these measures, the five features for our model consist of the steadiness measures for the four rates, and the learning trend.

4. Constructing the Student Labels

The five features in the previous section are designed to identify and differentiate the following classes of student behavior.

- Erratic (suspicious) learning

- Procrastination/cramming
- Plateaued learning ("hitting a wall")
- Normal

Our emphasis is not on identifying a student as being, for instance, a slow or a fast learner—for which the mean hourly learning rate would be an appropriate measure. Rather, we are interested in identifying changes in learning dynamics that point to potential underperformance in the use of the learning system. Consequently, feedback from instructors using ALEKS in a classroom setting, along with exploratory analyses of learning data from 2484 students in a college-level Basic Math course, lead to the selection of the above behavioral classes and to the use of rate steadiness to capture them.

The students in the erratic, or suspicious, learning class are identified mostly by large fluctuations in their hourly learning rate. Anecdotally, we have been informed of several cases where students are having someone else do their work in the ALEKS learning mode, resulting in short bursts of abnormally rapid progress. An example of this behavior is shown in Figure 1 (blue curve). From this plot it seems very likely that we are dealing with two distinct learning patterns, one during the first 28 hours or so of the course, and then another from there on. Patterns such as these are what motivated the design of our features, which are engineered to find learning patterns that exhibit a large amount of variation.

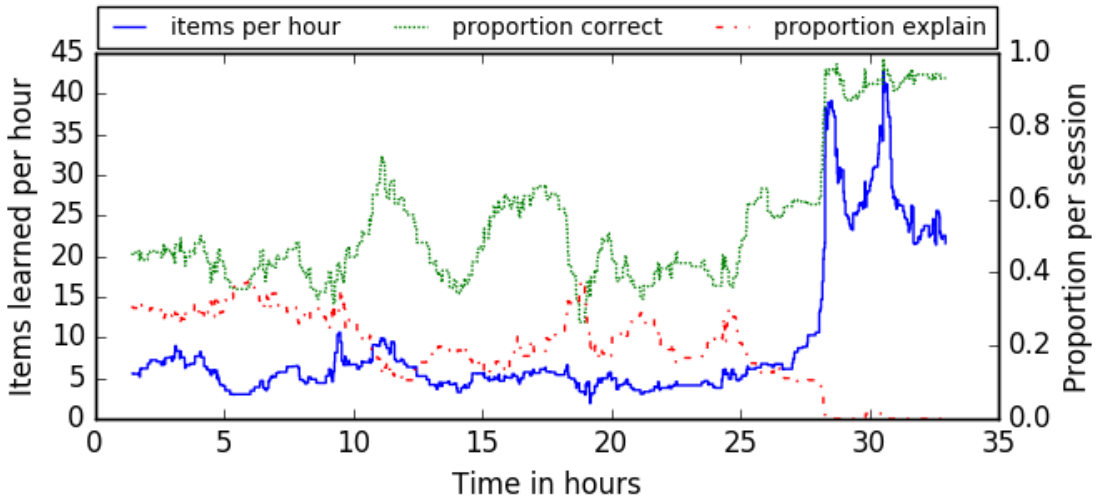

Figure 1. Example student with a highly variable hourly learning rate

Interestingly, while the variation in the hourly learning rate is the most obvious, the erratic students also exhibit large amounts of variation in the rates in which they answer correctly, and also in which they use the explanation. These effects are also shown in Figure 1. As with the hourly learning rate, at about 28 hours both of these values show a distinct transition, with the correct rate starting below 0.6 and eventually peaking at over 0.95. Conversely, the usage of the explanation becomes essentially non-existent after this time.

The next class of student learning we encounter is that of procrastination and/or cramming. In contrast to the students with erratic learning, these students have a relatively consistent hourly learning rate, which is evidence that we are dealing with one actual student learning profile. However, the larger fluctuations appear in the daily learning rate, where for the most extreme cases it is clear that we are seeing students cramming their learning into a short period either before the end of the course, or before an important exam. An example of such a pattern is shown in Figure 2, where the majority of the learning is concentrated in two periods near the end of the course. For this same student, Figure 3 shows the hourly rate, along with the correct and explanation rates. In contrast to the erratic profile in Figure 1, the shift for this student is more subtle. While there is an increase in the hourly learning rate towards the end of the course, which corresponds to the spikes in the daily learning rate, the correct rate during this time is not significantly different from the time period just before (as opposed to the large spike in Figure 1). Additionally, the increase in the hourly

learning rate can be reasonably explained by the large drop in the usage of the explanation, which previously constituted a significant amount of the student's learning events (roughly 30% for most of the course). Thus, rather than this being the work of a separate person, this is most likely a student who had previously spent a significant amount of time using the explanation, which in turn lowered the hourly learning rate, and who thereafter made a more concerted effort to solve and master the items at a faster pace.

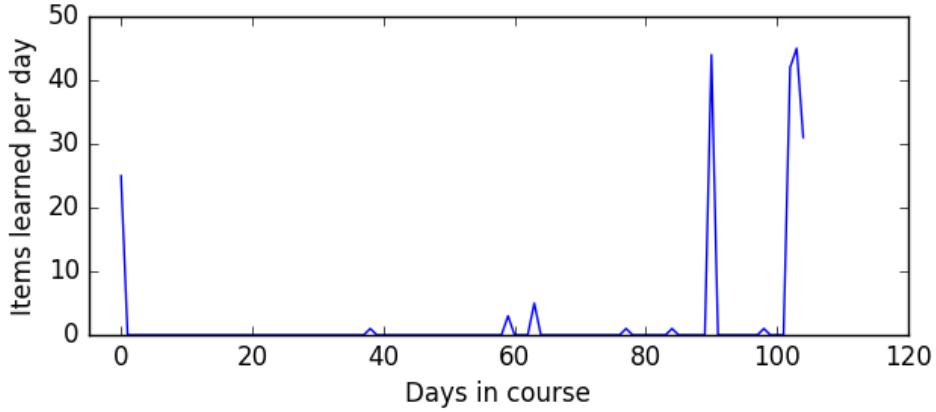

Figure 2. Example student with a highly variable daily learning rate

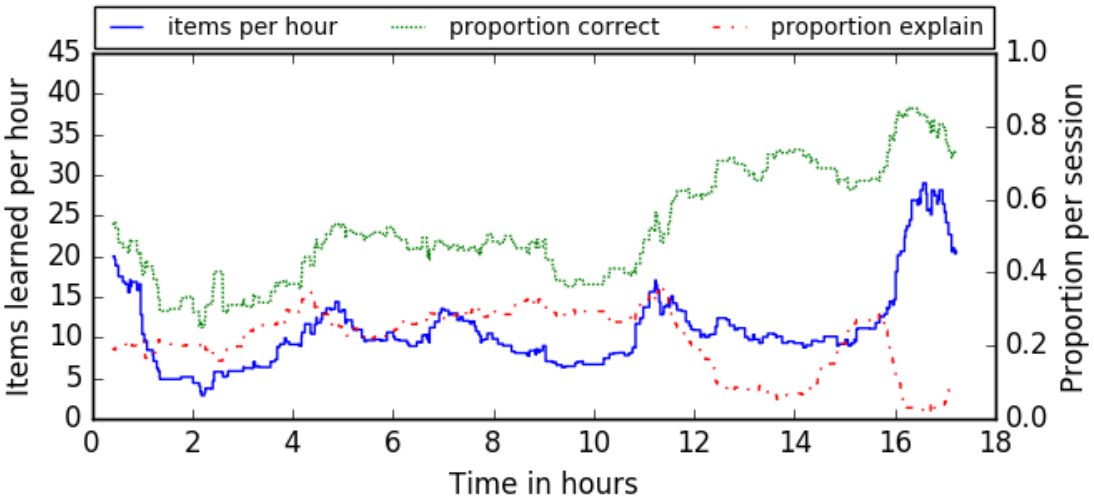

Figure 3. Login learning rate plots for student in Figure 2

The third class of learning pattern consists of students whose learning has slowed down considerably, or even stopped completely, over the length of the course. Like students with erratic learning, these students with plateaued learning (or, "hitting a wall") also show variability in their hourly learning rate, though it tends not to be as extreme. Additionally, they are also characterized by a negative trend in their hourly learning rate. Figure 4 shows one example of such a student. The student has a consistent hourly learning rate through the first few hours of the course, but after that time the rate begins to decline. Interestingly, we can see that the correct rate is mostly steady, until it declines and reaches its minimum near the end of the course. At the same time, while the correct rate is declining, we see a corresponding increase in the usage of the explanation, giving more evidence that this particular student is struggling to make progress in the learning mode.

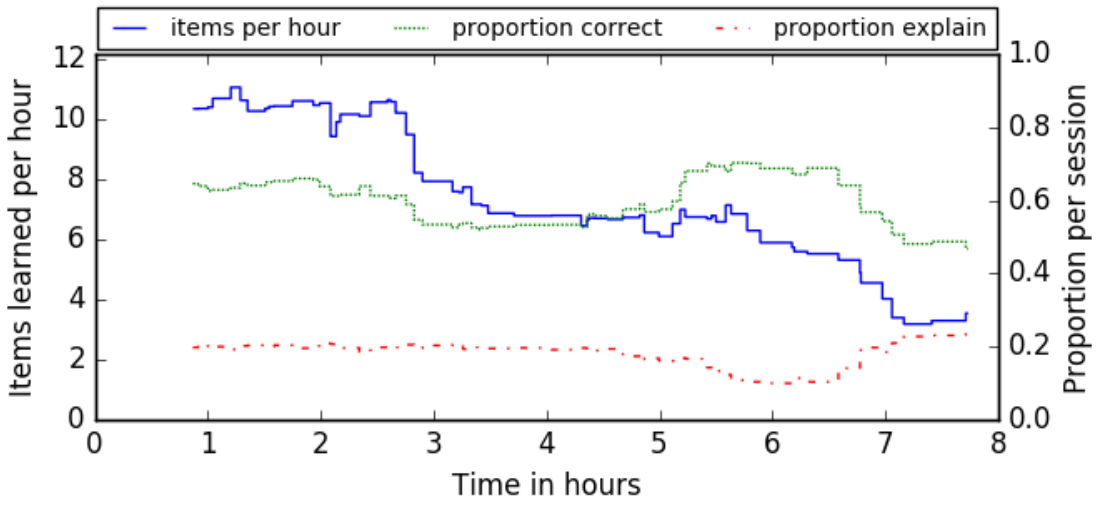

Figure 4. Example student with declining hourly learning rate

The final class consists of the rest of the students who do not fit into any of the previous three classes, and it comprises the bulk of the data.

Based on the insights described in this section, the following procedure was used to assign class labels to a sample of students. The five features were computed for each of the 2484 students in the Basic Math course, and these features were then used to identify a sample of 32 students containing potential examples from each class. Typically, for students potentially in one of the three extraordinary classes, the values of one or more features were above the 95th percentile (see Table 1). These 32 students were then manually evaluated and labeled independently by two experts, who based their evaluations on the complete learning profiles of the students (i.e., extra data that were not part of the model were used for these evaluations). The experts had an initial inter-rater agreement for these labels, as measured by Cohen's kappa, of 0.77. As a last step, the experts discussed the discrepancies and agreed on the final labels of the 32 students. Table 1 shows the features for example students from each of the classes.

Table 1

Examples of learning patterns and their feature values for 7 of the 32 labeled students. The 5th and 95th percentiles for each feature are listed beneath the column headers, and values outside these bounds are highlighted in bold in the body of the table.

	Steadiness Rates				
Class	**Hourly** **(0.22, 0.53)**	**Explanation** **(0.16, 0.62)**	**Correct** **(0.11, 0.29)**	**Daily** **(0.94, 1.65)**	**Trend** **(-0.44, 0.32)**
Normal	0.2848	0.1641	0.1659	1.3738	-0.0475
Erratic	**0.7547**	**0.929**	**0.2952**	1.5842	**2.985**
Erratic	**0.6264**	0.5197	**0.3883**	**1.6749**	**-0.6103**
Hitting a wall	0.4362	0.2538	0.1509	1.3399	**-0.510**
Hitting a wall	0.423	0.3478	0.2002	**1.7609**	**-1.088**
Cramming	0.2855	0.3756	0.2462	**1.8265**	**0.340**
Cramming	**0.1812**	0.2025	0.1131	**1.8537**	-0.184

5. Model Building

To build our student classifier, we will use a semi-supervised machine learning model. Semi-supervised learning models lie between supervised and unsupervised learning, and the unique feature of these models is that they are able to take advantage of both labeled and unlabeled data. Such techniques can be very useful in situations where the amount of labeled data is small in comparison to the amount of unlabeled data. This situation can occur, for example, when assigning accurate labels to data takes a considerable amount of manual (human) effort. In such a case, adding the large amount of extra unlabeled data can possibly give a large increase in the accuracy of the model (Mitchell, 1999; Nigam et al., 2000). For a thorough introduction to semi-supervised learning, we refer the reader to Chapelle et al. (2006) or Zhu and Goldberg (2009)

Our first attempt at building a classifier consists of a logistic regression model that is modified to take advantage of the unlabeled data using entropy regularization (Grandvalet and Bengio, 2006; Zhu and Goldberg, 2009). The key assumption made by entropy regularization in a semi-supervised classification problem is that the classes are separated by low density regions (see Grandvalet and Bengio, 2006, section 9.2; Zhu and Goldberg, 2009, sections 6.3 and 6.4). In our particular problem, we are looking for extreme student behaviors that fall outside the normal learning patterns. Thus, we expect that any students fitting these patterns are well-separated from students exhibiting more standard learning patterns. Using entropy regularization, the cost function for logistic regression can be modified with an additional summand of the form (see Grandvalet and Bengio, 2006, section 9.2; Zhu and Goldberg, 2009, section 6.3)

$$\frac{\lambda}{u}\sum_{i=1}^{u}\sum_{k=1}^{K} P(k|\,\mathbf{x}_i)\ln P(k|\mathbf{x}_i), \qquad (1)$$

where u is the number of unlabeled examples, K is the number of classes, $\boldsymbol{P(k|\mathbf{x}_i)}$ is the predicted probability of class k, and $\boldsymbol{\lambda}$ is a scale factor that determines the contribution of the unlabeled data.

The results using a logistic regression classifier are shown in Figures 5 and 6, where we can see that the classes seem to be relatively well-separated from each other. For example, the students with plateaued learning are separated from the other classes by their low hourly learning rate trend values. Similarly, the students exhibiting erratic learning are mostly identifiable by the large values of the hourly learning rate steadiness.

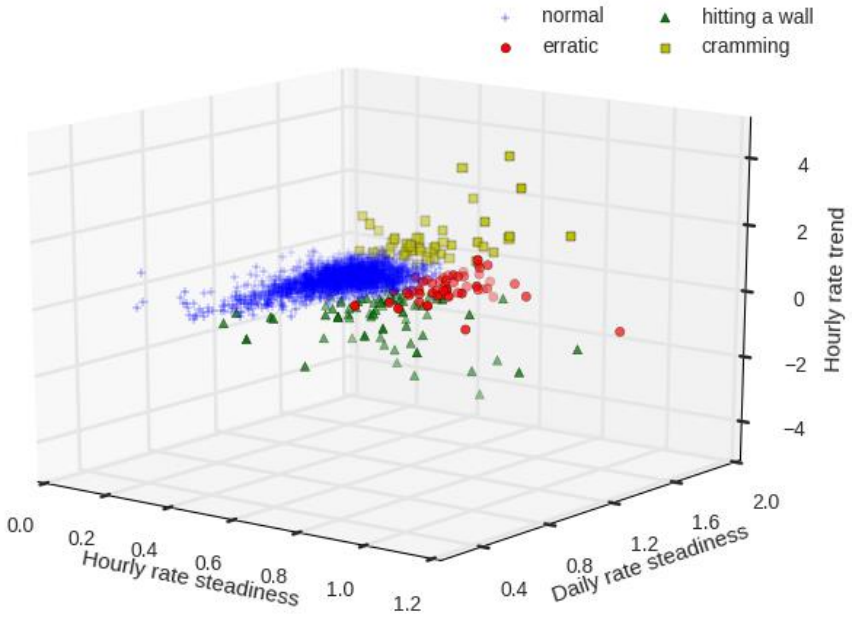

Figure 5. Classifications using logistic regression and entropy regularization

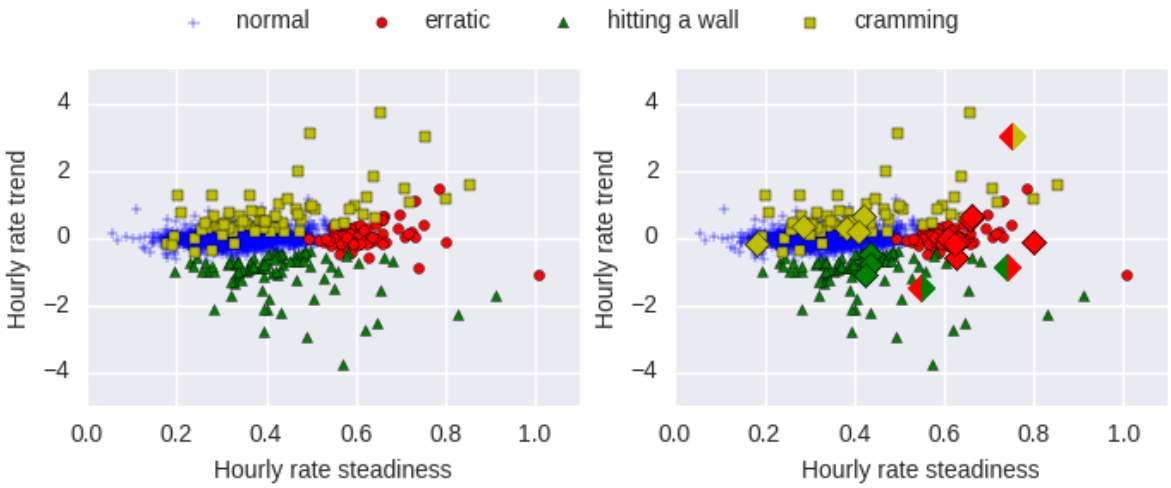

Figure 6. Classifications using logistic regression and entropy regularization

While the logistic classifier gives good results on the whole, upon closer examination it becomes apparent that there are some limitations to the model. To better illustrate this point, the right plot in Figure 6 shows the performance of the classifier on the labeled data. Each diamond represents a labeled data point from one of the three extraordinary classes. The solid colored diamonds are correctly labeled by the classifier after training, while the diamonds with two colors are not. For these mislabeled points, the left color represents the actual ground truth label, while the right color shows the predicted label assigned by the classifier.

One issue has to do with the learning trend values. As previously mentioned, a low value for this feature is typically a good indication that the student's learning has stalled. Furthermore, when numerous example classifications from the logistic classifier are checked manually, this assumption holds for the vast majority of them. However, there are a few specific cases where an extremely low learning trend value does not fit the intuitive definition of plateaued learning. These cases follow a similar pattern where a student, at some point in the course, shows a large (and suspicious) spike in the hourly learning rate. This spike is short-lived, and then the hourly learning rate returns to its previous level. Thus, these students are better labeled as having erratic learning patterns, rather than having plateaued, because the increase in the learning rate is dramatic enough that it seems likely to have come through some sort of suspicious behavior (such as using unapproved outside resources, or having someone else doing the work for them). However, if this spike in learning happens to have occurred early in the course, the overall hourly learning rate shows a large downward trend, which in turn (most likely) causes the logistic classifier to label the student as hitting a wall. Thus, we are simply running into the limitations of a linear model (i.e., the logistic classifier) trying to separate classes with a non-linear boundary.

To better handle the non-linear boundaries in the data, we next build a classifier using an artificial neural network. Deep learning, of which neural networks are an important component, has recently achieved dramatic successes in various fields (LeCun et al., 2015) and is beginning to move into the education domain. One successful application employed deep learning for affect detection (Botelho et al., 2017), while another technique known as Deep Knowledge Tracing (DKT) is being actively studied (Khajah et al., 2016; Piech et al., 2015; Xiong et al., 2016). Our specific architecture consists of a small multilayer perceptron with 3 hidden layers of 10 units (nodes) each. For the activation function of our hidden units we use a rectified linear unit (ReLU), which is a common and successful non-linear activation function (LeCun et al., 2015). To handle the unlabeled data, we again use entropy regularization as shown in equation (1).

The results using the neural network are shown in Figure 7, where in the right plot we can see that all the labeled data for the extraordinary classes are now correctly classified. Additionally, the neural network also shows improvement with other unlabeled extreme values. For example, the logistic model has a tendency to classify students with extremely large positive learning trend values

as being procrastinators/crammers. However, many of these students are more correctly classified as showing erratic or suspicious behavior, as the large value of the learning trend is (typically) caused by a large spike in the hourly learning rate at a later point in the course. Similarly, the students with extremely low learning trend values are better classified by the neural network as being erratic learners, rather than as hitting a wall.

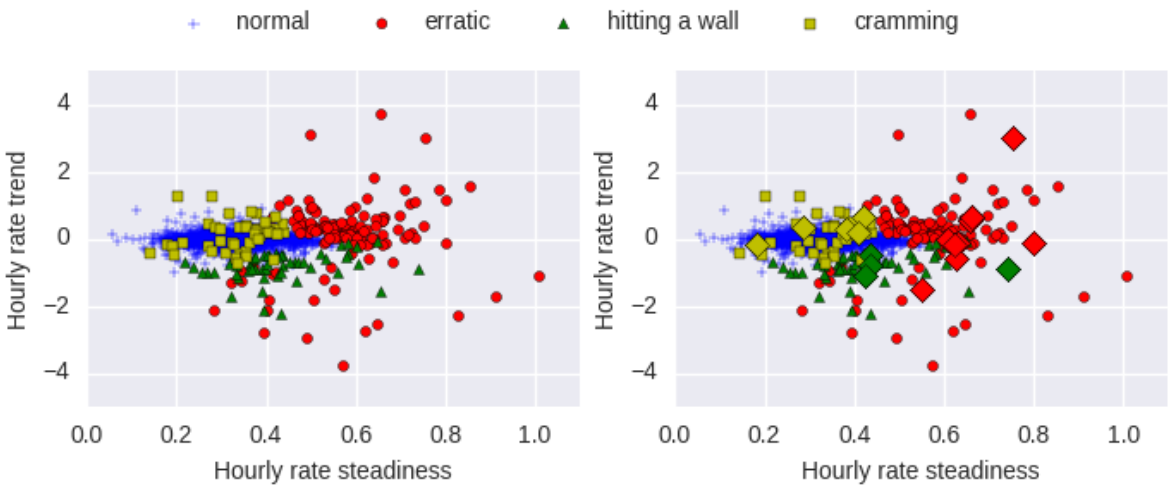

Figure 7. Classifications using multilayer perceptron and entropy regularization

6. Model Validation

Now that we have evidence that the neural network approach improves on the limitations of the logistic regression, we will next validate the performance of the neural network model. To do this, we begin by supplementing our data with an additional dataset from a college-level Intermediate Math course. This new dataset consists of data from 8315 students, 32 of whom have been manually assigned a label from one of the four learning pattern classes (using a similar procedure to the one outlined at the end of section 4). In total, we now have a combined dataset of 10799 students, 64 of whom have been assigned labels. The 64 labeled students consist of 18 in the erratic class, 11 in the hitting a wall class, 14 in the cramming class, and 21 in the normal class.

Due to the small amount of labeled data, we use cross-validation to estimate the accuracy of our model, rather than partitioning the data into fixed training and test sets. Additionally, since we use the architecture described in the previous section (i.e., 3 hidden layers of 10 nodes each, combined with a ReLU activation function), the only hyperparameter that we vary is a scale factor determining the contribution of the unlabeled data to the loss function. This scale factor is represented by λ in equation (1). Thus, in order to vary this factor and still obtain an accurate estimate of the model's performance, we apply nested *k*-fold cross-validation to the labeled data. Using nested *k*-fold cross-validation, rather than *k*-fold cross-validation without nesting, reduces the bias when evaluating a model's expected performance (Cawley and Talbot, 2010; Varma and Simon, 2006). We use 16 folds of 4 data points each for the outer loop, and 15 folds of 4 data points each for the inner loop, while using the same complete set of unlabeled data for all training iterations. That is, at each iteration the training set consists of either 60 (for the outer loop) or 56 (for the inner loop) labeled examples, in addition to all 10735 unlabeled examples, while the test set simply consists of 4 labeled examples. Using this procedure, 57 of the 64 labeled examples are correctly classified in the outer loop, for an overall estimated accuracy of 89.1%.

Independently of the overall accuracy of the model, we are also interested in measuring the relative effect of adding the unlabeled data to the model. Thus, as an additional analysis separate from the procedure discussed in the previous paragraph, we use repeated *k*-fold cross-validation, without nesting, for various values of the unlabeled data scale factor. For reference, a value of 0 for this factor means the unlabeled data have no effect on the loss function, while a value of 1 means

the unlabeled data have equal weight in comparison to the labeled data (with the effect of the other values then falling somewhere in-between these two extremes). Furthermore, we also evaluate the performance when entropy regularization is combined with a newer and more advanced method known as virtual adversarial training (VAT), a combination that has recently achieved state-of-the-art performance on several standard benchmarks (Miyato et al., 2018; Oliver et al., 2018). Using 100 separate trials of *k*-fold cross-validation (with the trials being run independently for the two different semi-supervised models), Table 2 gives the average accuracy for several values of the scale factor.

Table 2

Results from 100 trials of k-fold cross-validation for unlabeled data scale factor (0=no weight given to unlabeled data; 1=equal weight given to unlabeled and labeled data). Averages computed on the 64 labeled examples over the 100 trials.

	Entropy Regularization		**Entropy Regularization plus VAT**	
Unlabeled Data Scale Factor	**Average Correct**	**Average Accuracy**	**Average Correct**	**Average Accuracy**
0.00	55.85	0.873	55.86	0.873
0.01	56.27	0.879	56.67	0.885
0.05	56.39	0.881	57.31	0.895
0.10	56.50	0.883	57.41	0.897
0.25	55.94	0.874	56.84	0.889
1.00	55.51	0.867	54.04	0.844

The results in the table show the best performance for both models for the scale factor 0.1, which, in the case of entropy regularization plus VAT, gives a roughly 2.4 percentage point increase in accuracy over the fully supervised model (scale factor 0). Also, note that giving too much weight to the unlabeled data appears to degrade the performance of the classifier, with the lowest accuracy reached for the scale factor 1, again for both models.

7. Discussion

We have studied the various learning patterns that students exhibit while working in the adaptive learning mode of the ALEKS system. After developing a measure for the steadiness of a student with respect to informative time-dependent variables, we classified a sample of the students into one of four different learning behaviors. We then combined these labeled students with the large amount of unlabeled students and built two different semi-supervised machine learning models. Using only a small set of carefully chosen and engineered features, we saw that the neural network model was able to separate and classify the unlabeled students in an intelligent and intuitive manner.

Practically speaking, classifying and differentiating the students in the ALEKS system based on their learning patterns is an important step in ensuring that students are having an optimal learning experience. If such information can be quickly and effectively communicated to the instructor, corrective intervention can be performed. For instance, a student whose learning stalls will benefit from a diagnostic of the possible cause by the instructor. Similarly, a student whose learning is suspected to have been performed by someone else will deserve closer scrutiny from the instructor. Finally, such information can also be used to improve the ALEKS system itself, where students who are struggling can be targeted for further review of the material.

An actual implementation of such an alert system would also take advantage of the semi-supervised aspect of the model, in which feedback from instructors can be used to retrain and improve the classifier. Specifically, once a student is identified by the model as exhibiting one of the learning patterns, this can be communicated to the instructor. The instructor can then verify that the label is correct, or even propose a different label if they disagree with the system's classification. In either case, this new piece of labeled data can be fed back into the semi-supervised learning

algorithm to improve the predictive performance of the model.

Acknowledgements

We would like to thank Ryan Baker for his helpful comments upon reading a previous version of this paper. We would also like to thank Hasan Uzun for his input and suggestions during the course of this work.

References

Amershi, S., & Conati, C. (2009). Combining unsupervised and supervised classification to build user models for exploratory learning environments. *Journal of Educational Data Mining, 1*(1), 18-71.

Botelho, A., Baker, R., & Heffernan, N. (2017). Improving sensor-free affect detection using deep learning. *Artificial Intelligence in Education-18th International Conference, AIED* 2017, pp. 40-51.

Cawley, G., & Talbot, N. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine learning Research, 11*, 2079-2107.

Chapelle, O., Schölkopf, B., & Zien, A. (Eds.). (2006). *Semi-Supervised Learning*. Cambridge: MIT Press.

Doignon, J.-P., and Falmagne, J.-C. (1985). Spaces for the assessment of knowledge. *International Journal of Man-Machine Studies, 23*, 175-196.

Falmagne, J.-C., Albert, D., Doble, C., Eppstein, D., & Hu, X. (Eds.). (2013). *Knowledge Spaces: Applications in Education*. Heidelberg: Springer-Verlag.

Falmagne, J.-C., Cosyn, E., Doignon, J.-P., & Thiéry, N. (2006). The Assessment of Knowledge, in Theory and in Practice. In: Missaoui, R., Schmidt, J. (Eds.), *Formal Concept Analysis: Foundations and Applications*. Heidelberg: Springer-Verlag.

Falmagne, J.-C., & Doignon, J.-P. (2011). *Learning Spaces*. Heidelberg: Springer-Verlag.

Grandvalet, Y., & Bengio, Y. (2006). Entropy regularization. In Chapelle, O., Schölkopf, B., & Zien, A. (Eds.), *Semi-Supervised Learning*, pp. 151-168. Cambridge: MIT Press.

Huang, X., Craig, S., Xie, J., Graesser, A., & Hu, X. (2016). Intelligent tutoring systems work as a math gap reducer in 6th grade after-school program. *Learning and Individual Differences, 47*, 258-265.

Khajah, M., Lindsey, R., & Mozer, M. (2016). How deep is knowledge tracing? *Proceedings of the 9th International Conference on Educational Data Mining*, pp. 94-101.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*, 436-444.

Oliver, A., Odena, A., Raffel, C., Cubuk, E. D., & Goodfellow, I. J. (2018). Realistic evaluation of deep semi-supervised learning algorithms. arXiv preprint arXiv:1804.09170.

Miyato, T., Maeda, S., Koyama, M., & Ishii, S. (2018). Virtual adversarial training: a regularization method for supervised and semi-supervised learning. arXiv preprint arXiv:1704.03976.

Mojarad, S., Essa, A., Mojarad, S., & Baker, R. S. (2018). Data-driven learner profiling based on clustering student behaviors: learning consistency, pace and effort. *Proceedings of the 14th International Conference on Intelligent Tutoring Systems*, pp. 130-139.

Mitchell, T. (1999). The role of unlabeled data in supervised learning. *Proceedings of the Sixth International Colloquium on Cognitive Science*, pp. 103-111.

Nigam, K., McCallum, A., Thrun, S., & Mitchell, T. (2000). Text classification from labeled and unlabeled documents using EM. *Machine Learning, 39*, 103-134.

Piech, C., Bassen, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L., & Sohl-Dickstein, J. (2015). Deep knowledge tracing. *Advances in Neural Information Processing Systems*, pp. 505-513.

Reddy, A., & Harper, M. (2013). Mathematics placement at the University of Illinois. *PRIMUS, 23*, 683-702.

Taagepera, M., & Arasasingham, R. (2013). Using knowledge space theory to assess student understanding of chemistry. In Falmagne, J.-C., Albert, D., Doble, C., Eppstein, D., & Hu, X. (Eds.). *Knowledge Spaces: Applications in Education*, pp. 115-128. Heidelberg: Springer-Verlag.

Varma, S., & Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics, 7*, 91.

Xiong, X., Zhao, S., Vaninwegen, E., & Beck, J. (2016). Going deeper with knowledge tracing. *Proceedings of the 9th International Conference on Educational Data Mining*, pp. 545-550.

Yang, Y., Leung, H., Yue, L., & Deng, L. (2012). Automatic dance lesson generation. *IEEE Transactions on Learning Technologies, 5*(3), 191-198.

Zhu, X., & Goldberg, A. (2009). *Introduction to Semi-Supervised Learning*. Morgan & Claypool.