Article

Recommendation System for Adaptive Learning

Applied Psychological Measurement
2018, Vol. 42(1) 24–41

DOI: 10.1177/0146621617697959
journals.sagepub.com/home/apm
SAGE

Yunxiao Chen[1], Xiaoou Li[2], Jingchen Liu[3], and Zhiliang Ying[3]

Abstract

An adaptive learning system aims at providing instruction tailored to the current status of a learner, differing from the traditional classroom experience. The latest advances in technology make adaptive learning possible, which has the potential to provide students with high-quality learning benefit at a low cost. A key component of an adaptive learning system is a recommendation system, which recommends the next material (video lectures, practices, and so on, on different skills) to the learner, based on the psychometric assessment results and possibly other individual characteristics. An important question then follows: How should recommendations be made? To answer this question, a mathematical framework is proposed that characterizes the recommendation process as a Markov decision problem, for which decisions are made based on the current knowledge of the learner and that of the learning materials. In particular, two plain vanilla systems are introduced, for which the optimal recommendation at each stage can be obtained analytically.

Keywords

adaptive learning, hidden Markov model, stochastic scheduling, Markov decision process, multi-armed bandit problem, Gittins index, c-μ rule

Introduction

In recent years, efforts have been made to expand learning beyond the traditional classroom setting, with many successful examples (Means, Toyama, Murphy, Bakia, & Jones, 2009). Online learning platforms emerge, such as Knewton, Khan Academy, and massive online open courses (MOOCs). A key advantage of the technology assisted learning experience over classroom learning is the adaptiveness (Webley, 2013). That is, students are enabled to take different learning trajectories according to their unique characteristics, realized by advanced Internet and data science technologies. *Adaptive learning* makes it possible for each student to learn on his or her own pace, so that fast learners do not need to wait for the entire class and slower learners

[1]Emory University, Atlanta, GA, USA
[2]University of Minnesota, Minneapolis, MN, USA
[3]Columbia University, New York, NY, USA

Corresponding Author:
Zhiliang Ying, Department of Statistics, Columbia University, Room 1005 SSW, MC 4690, 1255 Amsterdam Avenue, New York, NY 10027, USA.
Email: zying@stat.columbia.edu

have more time to digest the materials (see Zhang and Chang, 2016, for a review of recent practices, as well as challenges in adaptive learning).

The engine of an adaptive learning system is a recommendation strategy that sequentially makes decisions on what to learn in the next step, based on the currently available information. It is intuitive that a good recommendation strategy makes full use of the information about both the learner and the training materials (e.g., video lectures, practices, etc.) and makes decisions to maximize the overall gain along the whole learning trajectory, instead of only focusing on the gain in the next step. To quantify the recommendation strategies, a mathematical framework that characterizes the learning and recommendation process will be introduced in this article that integrates advanced techniques in psychometrics, statistics, and operations research. The problem is formulated as follows: Assume K skills are available in the learning system and denote $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_K)$ the corresponding proficiency levels of a learner. In addition, consider time epoches $t \in \{0, 1, 2, ...\}$ and $\boldsymbol{\alpha}(t)$ the proficiency levels of the learner at time t. As $\boldsymbol{\alpha}(t)$ is the unobservable latent characteristics of the learner, they can only be measured using assessment items, for which a measurement model is involved. The result from measurement, denoted by $\widehat{\boldsymbol{\alpha}}(t)$ (which is not necessarily a point estimate), is then used to select the next learning material $d(t)$. Given the learning material $d(t)$, the proficiency levels evolve to $\boldsymbol{\alpha}(t+1)$ in a stochastic fashion and the learner receives a reward $R_{d(t)}(\boldsymbol{\alpha}(t))$ from this transition. In summary, this mathematical framework considers three key components of an adaptive learning system, including (a) a *measurement model* that keeps track of the proficiency levels of the learner on different skills, (b) a *learning model* that describes how learning materials improve the proficiency levels, and (c) a *recommendation strategy*. Such an adaptive learning system is illustrated in Figure 1.

To find appropriate measurement models, the authors refer the readers to psychometrics, which studies methodologies of measurement. Considering the multidimensional nature of the learning system (i.e., multiple skills may be available), two families of models may be adopted here, namely, the multidimensional item response theory (MIRT) models (Reckase, 2009) and the diagnostic classification models (DCMs; Rupp & Templin, 2008; Rupp, Templin, & Henson, 2010). In particular, MIRT models assume α_ks to be continuous latent variables, while DCMs assume $\alpha_k \in \{0, 1, ..., M-1\}$, representing proficiency levels from low ($\alpha_k = 0$) to high ($\alpha_k = M-1$). When $M = 2$, only mastery/nonmastery of the skill is modeled. Depending on the specific application and item types, one may choose a measurement model from these two families. Furthermore, one may combine the measurement model with a computerized adaptive test (CAT) design to make the assessment more efficient. Items in a CAT design are selected sequentially and tailored to the learner so that fewer items are needed to achieve the same measurement accuracy comparing with a design that is not optimal. See ''Measurement Model'' section for more discussions.

A learning model associates each learning material with the change in the proficiency levels $\boldsymbol{\alpha}$. In this article, the transition is modeled using a Markov model (e.g., Norris, 1998). That is, the transition only depends on the current state of $\boldsymbol{\alpha}(t)$, but not the time t. In other words, the conditional distribution of $\boldsymbol{\alpha}(t+1)$ given $\boldsymbol{\alpha}(t) = \boldsymbol{\alpha}$ can be written as $f_{\boldsymbol{\alpha}}$, the form of which is specified in the learning model. Moreover, as $\boldsymbol{\alpha}(t)$ are latent variables, the learning model has to be coupled with the measurement model in making inference (such as estimating $f_{\boldsymbol{\alpha}}$). The coupled model then becomes a hidden Markov model (e.g., Cappé, Moulines, & Rydén, 2005). When the proficiency levels are discrete, hidden Markov models have been used to model the learning process (Kaya & Leite, 2016; Li, Cohen, Bottge, & Templin, 2016; S. Wang, Yang, Culpepper, & Douglas, 2016). Details are provided in the ''Learning Model'' section.

Making the right recommendation is a statistical decision problem (e.g., Berger, 2013). Specifically, it is a stochastic scheduling problem that studies priority assignment to jobs with random features, with applications in manufacturing and computer-communication systems

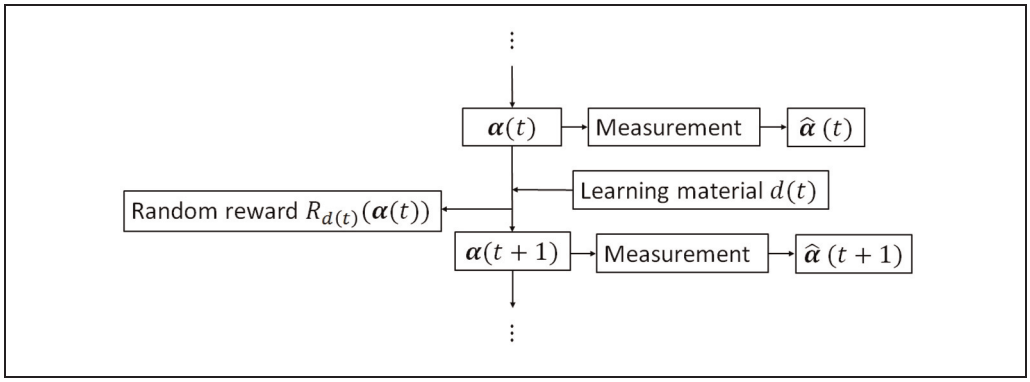

Figure 1. Visualization of the adaptive learning process.

(e.g., Nino-Mora, 2009; Pinedo, 2008). Research on stochastic scheduling starts from the multi-armed bandit problem in statistics (Gittins, Glazebrook, & Weber, 2011; Gittins & Jones, 1974; Lai & Robbins, 1985; Lai & Ying, 1988; Robbins, 1952) and becomes popular in queuing theory, a branch of operations research (Gross, 2008), and in reinforcement learning, a branch of machine learning (Gosavi, 2009; Sutton & Barto, 1998). Under the current setting, the jobs in a stochastic scheduling problem refer to the learning materials. More precisely, at time t, there is a pool of available learning materials denoted by $\mathcal{D}_t$, and a decision has to be made by choosing $d(t)$ from $\mathcal{D}_t$. When making the decision, all information in the system from time 0 to time t can be used to assist the decision making. Given the choice $d(t) \in \mathcal{D}_t$, a random reward $R_{d(t)}(\boldsymbol{\alpha}(t))$ is received, which may depend on the improvement from time t to $t+1$ and/or the learning time spent on the material, and so on. Following the statistical decision theory, one may construct an objective function that can be viewed as a learning goal in the long run. In the present setting, the objective function is the expected rewards aggregated over time t and a recommendation strategy is said to be optimal if the objective function is optimized. Details are provided in the ''Recommendation Strategy'' section, and the optimal strategies in special settings are discussed in the ''Plain Vanilla System I'' and ''Plain Vanilla System II'' sections.

The rest of the article is organized as follows: In the ''Mathematical Framework for Adaptive Learning'' section, a general mathematical framework for an adaptive learning system is proposed. In ''Plain Vanilla System I'' and ''Plain Vanilla System II'' sections, two plain vanilla recommendation systems are introduced, whose optimal recommendation strategies are analytically tractable and interpretable. This is followed by the ''Examples'' section, and finally the ''Discussion'' section.

Mathematical Framework for Adaptive Learning

Measurement Model

The key difference between adaptive and fixed learning designs (e.g., traditional classroom learning) is that in an adaptive learning environment, the information about the learner up to time t can be utilized to help decide the next learning material. Such information may include the proficiency levels of the learner at time t, his or her learning speed, and so on. Psychometrics is the field that studies approach to extract such information from data, which could include item responses, response time, learning time, and so on. In this sense, *psychometrics is an important building block of an adaptive learning system*. When the up-to-date information about the learner

Table 1. An Example of the Q-Matrix.

		Subtraction (α_1)	Multiplication (α_2)
Q=	1. $7-2$	1	0
	2. 5×2	0	1
	3. $(7-2)\times2$	1	1

is fully utilized, as illustrated in ''Examples'' section, learning can be greatly improved upon a fixed design. For ease of exposition, the article focuses on the measurement of proficiency levels. Other factors and information may also be incorporated in the system.

Modern measurement theory adopts a latent variable model framework. Let $\boldsymbol{\alpha}=(\alpha_1,...,\alpha_K)$ be a latent vector, denoting the proficiency levels of a learner. The proficiency levels can be measured by test items. Let $\mathcal{I}$ be the item pool and for each $j\in\mathcal{I}$, the response Y_j from the learner with proficiency level $\boldsymbol{\alpha}=(\alpha_1,...,\alpha_K)$ follows a distribution that only depends on $\boldsymbol{\alpha}$ but not other characteristics of the learner, that is,

$$Y_j|\boldsymbol{\alpha}\sim h_{j,\boldsymbol{\alpha}}(y),$$

where $h_{j,\boldsymbol{\alpha}}(y)$ takes different parametric forms for different response types, latent variable types, and specific models. Specifically, when α_ks are ordinal, that is, $\alpha_k\in\{0,1,2,...,M-1\}$, the model falls into the category of DCMs (Rupp et al., 2010), and when α_k takes continuous values, that is, $\alpha_k\in\mathbb{R}$, the model is categorized as a MIRT model (Reckase, 2009). Moreover, given the proficiency level $\boldsymbol{\alpha}$, the responses are assumed to be independent, which is known as the local independence assumption.

A key quantity in the specification of $h_{j,\boldsymbol{\alpha}}(\cdot)$ is the item–skill relationship, characterized by the so-called Q-matrix (Tatsuoka, 1983). The design matrix $Q=(q_{jk})_{|\mathcal{I}|\times K}$ is a $|\mathcal{I}|$ by K matrix with zero-one entries, each of which indicates whether an item is associated to a skill α_k. More precisely, each row of Q represents an item and each column represents a skill. An example of Q-matrix with two skills, the subtraction and multiplication (represented by α_1 and α_2, respectively), is provided in Table 1. In this example, the first item ''$7-2$'' only measures the subtraction skill and therefore the first row of Q is $(1,0)$. The third item ''$(7-2)\times2$'' measures both skills and thus the corresponding row is $(1,1)$. Technically, the Q-matrix plays a role in the item response distribution through conditional independence. That is, given the levels of the measured skills, the irrelevant skills are independent of the response,

$$h_{j,\boldsymbol{\alpha}}(y)=P\left(Y_j=y|\boldsymbol{\alpha}\right)=P\left(Y_j=y|\left\{\alpha_k:q_{jk}=1\right\}\right).$$

The Q-matrix is typically specified by item designers during test development.

The specification of $h_{j,\boldsymbol{\alpha}}(\cdot)$ varies. Two examples are listed below, the multidimensional two parameter logistic (M2PL) model and the log-linear cognitive diagnosis model (LCDM). Other models, including the DINA (deterministic inputs, noisy ''AND'' gate) model (Junker & Sijtsma, 2001), the DINO (deterministic inputs, noisy ''OR'' gate) model (Templin & Henson, 2006), G-DINA (generalized deterministic inputs, noisy ''AND'' gate) model (de la Torre, 2011), and the general diagnostic model (von Davier, 2008), could also be useful.

Example 1 (M2PL model; Reckase, 2009). The M2PL model is one of the most popular models for measuring multiple continuous latent traits for binary responses, that is, $Y_j\in\{0,1\}$. The M2PL model assumes a logit link between the probability of correct response and the latent traits

$$h_{j,\boldsymbol{\alpha}}(1) = P\left(Y_j = 1|\boldsymbol{\alpha}\right) = \frac{e^{b_j + \sum_{k=1}^{K} a_{jk}\alpha_k}}{1 + e^{b_j + \sum_{k=1}^{K} a_{jk}\alpha_k}}, \tag{1}$$

where $a_{jk} = 0$ when $q_{jk} = 0$. This model is labeled as a compensatory model, as the success probability only depends on the linear combination $b_j + \sum_{k=1}^{K} a_{jk}\alpha_k$, and thus a low ability on one dimension (e.g., small α_1) can be compensated by a high ability on another dimension (e.g., large α_2), when both latent skills are being measured. Moreover, b_j and a_{jk}s are item parameters that can be estimated from data.

Example 2 (LCDM model; Henson, Templin, & Willse, 2009). The LCDM is DCM for binary responses, for which the proficiency levels also take binary values, that is, $\alpha_k \in \{0, 1\}$, where $\alpha_k = 0$ and 1 represents nonmastery and mastery of the skill k. The item response distribution is defined by,

$$logit\left(h_{j,\boldsymbol{\alpha}}(1)\right) = b_j + \sum_{k=1}^{K} a_{j,k}\alpha_k + \sum_{1 \le k_1 < k_2 \le K} a_{j,k_1k_2}\alpha_{k_1}\alpha_{k_2} + \cdots + a_{j,12\ldots K}\prod_{k=1}^{K}\alpha_k, \tag{2}$$

where

$$a_{j,k_1\ldots k_M} = 0 \ when \prod_{t=1}^{M} q_{j,k_t} = 0$$

for any subset $\{k_1, \ldots, k_M\} \subseteq \{1, ..., K\}$. The LCDM is a saturated model that includes many DCMs, such as the DINA, DINO, and NIDA (noisy inputs, deterministic ''AND'' gate), as special cases (see Rupp et al., 2010, for more details).

The authors make a few remarks. First, the above discussion is confined to the cross-sectional setting, that is, the measurement is only considered at a given time. In a learning system, the learner needs to be assessed at every time epoch t. Therefore, the item pool is in fact indexed by t, denoted by $\mathcal{I}_t$ that may vary over time. However, the item parameters are assumed to stay constant over time. Second, the Q-matrix plays an important role in the multidimensional measurement. It helps to identify the skills being measured through the item–skill relationship and thus greatly improves the identifiability of the model parameters (G. Xu & Zhang, 2015). In addition, it naturally partitions the items into groups. Based on the partition, relevant items can be easily targeted from a possibly large item pool when certain skills need more measurement. The Q-matrix is usually provided by the test designers. When information about the Q-matrix is vague, the Q-matrix can be learned from data (Chen, Liu, Xu, & Ying, 2015; Chen, Liu, & Ying, 2015; Liu, Xu, & Ying, 2012, 2013; Sun, Chen, Liu, Ying, & Xin, 2016) and further confirmed by domain experts. Third, the measurement models usually contain unknown parameters that need to be estimated. The estimation of MIRT models and DCMs has been studied in the psychometrics literature (see Liu, Magnus, Quinn, & Thissen, 2016; Rupp et al., 2010, for more details). Fourth, both MIRT models and Cognitive Diagnosis Models (CDMs) can be combined with computerized adaptive testing design to provide efficient measurement. A short but incomplete list of CAT designs for MIRT models and DCMs includes Segall (1996); Mulder and van der Linden (2009); C. Wang and Chang (2011); X. Xu, Chang, and Douglas (2003); Cheng (2009); and Liu, Ying, and Zhang (2015).

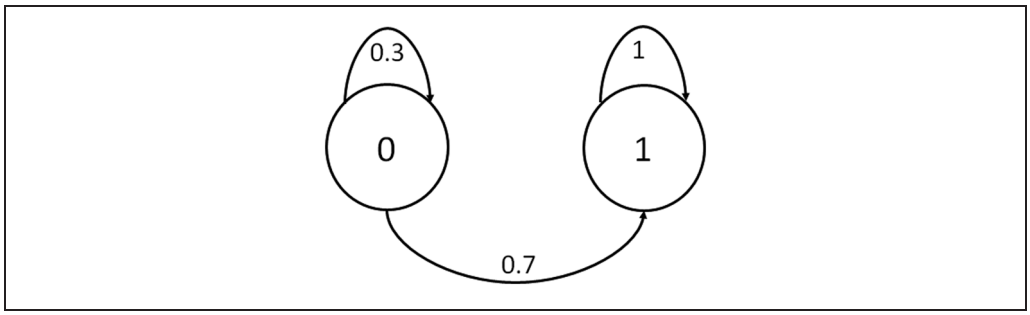

Figure 2. An example of a simple two-state Markov model.

Learning Model

To make better recommendation, one also needs a good understanding of the learning materials; that is, given the characteristics (e.g., proficiency levels) of a learner, how he or she would improve after learning a specific material. This process can be characterized by a learning model. More precisely, a Markov model is assumed for the transition from $\boldsymbol{\alpha}(t)$ to $\boldsymbol{\alpha}(t+1)$, which is the most popular stochastic model for characterizing a changing system with randomness. That is, the conditional distribution of $\boldsymbol{\alpha}(t+1)$ given $\boldsymbol{\alpha}(t)$

$$\boldsymbol{\alpha}(t+1)|\boldsymbol{\alpha}(t) = \widetilde{\boldsymbol{\alpha}} \sim f_{d,\widetilde{\boldsymbol{\alpha}}}(\boldsymbol{\alpha}),$$

where d indicates the learning material and $f_{d,\widetilde{\boldsymbol{\alpha}}}(\boldsymbol{\alpha})$, known as the transition kernel, is a density function when the proficiency levels are continuous and a probability mass function when the proficiency levels are discrete. In addition, the transition only depends on the current state of $\boldsymbol{\alpha}(t)$, but not the whole learning history. Figure 2 presents an example of a simple two-state Markov model. In this example, there is only one skill ($K = 1$)—only mastery/nonmastery of the skill is considered ($\alpha \in \{0, 1\}$)—and only one learning material. The transition kernel is specified by

$$f_0(0) = P(\alpha(t+1) = 0|\alpha(t) = 0) = 0.3; \quad f_0(1) = P(\alpha(t+1) = 1|\alpha(t) = 0) = 0.7;$$
$$f_1(0) = P(\alpha(t+1) = 0|\alpha(t) = 1) = 0; \quad f_1(1) = P(\alpha(t+1) = 1|\alpha(t) = 1) = 1.$$

It says that when the learner has not mastered the skill, his or her chance of mastering it after learning the material is 70%, and once the skill has been mastered, it will not be forgotten. Moreover, the expected learning time of the skill from nonmastery to mastery, defined as $\sum_{t=1}^{\infty} tP\,(\alpha(t) = 1|\alpha(0) = 0)$, is $1/0.7 \approx 1.4$ times.

Similar to the Q-matrix in the measurement model, the authors propose to characterize the relationship between the learning materials and skills using an S-matrix. $S_{|\mathcal{D}|\times K}$ is a $|\mathcal{D}| \times K$ matrix and $S_{dk} = 0$ if training using material d does not help improve skill k. Technically, $S_{dk} = 0$ means that the transition satisfies

$$P(\alpha_k(t+1) = \alpha|\alpha_k(t) = \alpha) = 1,$$

that is, k th skill will remain unchanged when material d is used at time t. Knowing the S-matrix from domain knowledge could greatly reduce the number of parameters in the hidden Markov model and can facilitate the recommendation by providing a natural partition of the learning materials. In practice, an S-matrix may be obtained from the designers of learning materials.

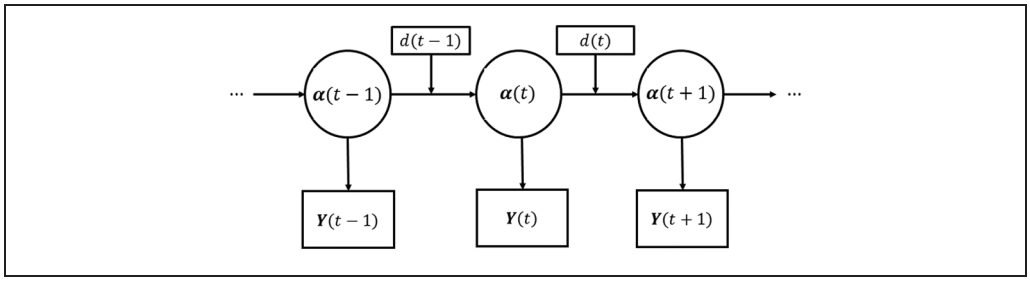

Figure 3. The general architecture of a learning model.

This prespecified S-matrix can be further validated and improved by data-driven approaches that make use of data collected in the adaptive learning system. Further discussion on S-matrix will be provided in the sequel.

The general architecture of the learning model is visualized in Figure 3. One unique feature of the learning model is that the states $\boldsymbol{\alpha}(t)$ are not completely observable. Instead, $\boldsymbol{\alpha}(t)$ is only partially observable through responses to assessment items, denoted by $\mathbf{Y}(t)$. For given learning materials $d(t)$, this learning model naturally becomes a hidden Markov model. Hidden Markov models are widely used in many fields including bioinformatics (e.g., Durbin, Eddy, Krogh, & Mitchison, 1998), econometrics (e.g., Kim, Shephard, & Chib, 1998), and population genetics (e.g., Felsenstein & Churchill, 1996). See Cappé et al. (2005), for the estimation of the unknown model parameters (e.g., unknown transition kernel) and making inference on $\boldsymbol{\alpha}(t)$s.

Recommendation Strategy

Following the previous setup, a recommendation strategy π is to select a learning material $d(t)$ from the material pool $\mathcal{D}_t$ at each time t, based on the information up to time t about the learner. To design and analyze different recommendation strategies, the construction of an objective function is discussed. A key component of the objective function is the reward function. That is, at time t, given a selected learning material $d(t)$, a reward $R_{d(t)}(\boldsymbol{\alpha}(t))$ is obtained. $R_{d(t)}(\boldsymbol{\alpha}(t))$ is random at time t, which may depend on the improvement from the training and possibly other factors such as total training time on $d(t)$. For example, the reward can take the form

$$R_{d(t)}(\boldsymbol{\alpha}(t)) = \sum_{k=1}^{K} c_k(\alpha_k(t+1) - \alpha_k(t)), \tag{3}$$

where c_k is the weight representing the importance of the k th skill and $\boldsymbol{\alpha}(t+1)$ depends on $d(t)$ and $\boldsymbol{\alpha}(t)$ through the transition kernel $f_{d(t),\boldsymbol{\alpha}(t)}(\cdot)$. This sequential process of making recommendations and receiving rewards is known as a partially observable Markov decision process (POMDP; Kaelbling, Littman, & Cassandra, 1998), and a recommendation strategy is also known as a policy in POMDP. *Partially observable* refers to $\boldsymbol{\alpha}(t)$ not being completely observed. Another important component of the objective function is the time horizon. Depending on the learning system, both finite and infinite time horizons can be considered. The objective function for a finite horizon problem can take the form of the expected total reward:

$$V_{\pi}(\boldsymbol{\alpha}) = E_{\pi}\left[\sum_{t=0}^{T-1} R_{d(t)}(\boldsymbol{\alpha}(t))|\boldsymbol{\alpha}(0) = \boldsymbol{\alpha}\right]. \quad (4)$$

An infinite horizon deals with the situation that total number of training can be unlimited $T = \infty$, but early reward is preferred. In that situation, a discount factor $\beta \in (0, 1)$ is typically introduced and the objective function is then defined as the expected discounted total reward:

$$V_{\pi}(\boldsymbol{\alpha}) = E_{\pi}\left[\sum_{t=0}^{\infty} \beta^{t} R_{d(t)}(\boldsymbol{\alpha}(t))|\boldsymbol{\alpha}(0) = \boldsymbol{\alpha}\right]. \quad (5)$$

In particular, based on Equation 5, the reward at time t is discounted by a factor β^t and the discount factor β determines the preference to early rewards. The smaller the β, the more preferred the early rewards. That is, skills that provide large early rewards (e.g., easy to learn or with a large importance index c_k) tend to be prioritized. The choice of β in an adaptive learning system is an open question, depending on the importance of early and future rewards. In practice, one may collect data using different βs in a recommendation system and then choose the best-performing one (determined by some external measures, such as domain experts' opinion). The infinite horizon setting may be suitable for prioritizing which skill to learn first in a long learning process. Both objective functions can be viewed as the total learning achievement during their respective time horizons, with the adaptive learning recommendation strategy π and the starting state $\boldsymbol{\alpha}$.

For an objective function $V_{\pi}(\boldsymbol{\alpha})$, a recommendation strategy π^* is said to be optimal if

$$V_{\pi^*}(\boldsymbol{\alpha}) \geq V_{\pi}(\boldsymbol{\alpha}),$$

for any other strategy π and all possible initial state $\boldsymbol{\alpha}$. In other words, the optimal recommendation strategy maximizes the objective function. Finding an optimal strategy can be challenging. For each time t, one needs to decide $d(t)$ that is a map from the space of responses $(\mathbf{Y}(0), ..., \mathbf{Y}(t))$ to $\mathcal{D}_t$, which can be a very high-dimensional problem. For certain types of problems, the optimal strategy is analytically tractable under an infinite time horizon. For example, the multiarmed bandit problem to be described in the ''Plain Vanilla System I'' section and some of its variants can be solved analytically by the so-called Gittins index (Gittins & Jones, 1974). Finding optimal or near-optimal recommendation strategies is known as the stochastic scheduling problem, which has been extensively studied in statistics, operations research, and machine learning. Existing methods typically require restrictive assumptions that are generally not applicable to the adaptive learning setup. Challenging issues raise with the constraints imposed upon the learning materials pool $\mathcal{D}_t$ and presence of measurement error on $\boldsymbol{\alpha}(t)$. Scheduling problems under finite horizon can be even more difficult to deal with analytically. Although, in principle, such problems can be solved numerically under the framework of dynamic programming (Bellman, 1954), it can be computationally prohibitive due to the curse of dimensionality (Nino-Mora, 2009).

Cognitive theory could also help to build an adaptive learning system. According to cognitive theory, some skills (e.g., multiplication skill) can hardly be mastered before others (e.g., summation skill), creating a hierarchy of skills. The skill hierarchy has been well studied in cognitive theory and is applicable to almost every learner. For example, multiplication skill is never taught before the learner is able to solve summation problems, as the later is a prerequisite for learning the former. Such skill hierarchy information has been used to improve the design of test items in psychometrics, known as the attribute hierarchy method (Leighton, Gierl, & Hunka, 2004). In the learning system, the skill hierarchy imposes constraints to the learning materials $\mathcal{D}_t$. Many

learning materials are frozen until some prerequisite skills are known to have been mastered. Such a priori information needs to be incorporated in to the design and analysis of an adaptive learning system. However, the hierarchy constraint can also make optimal solutions more difficult to obtain analytically.

Plain Vanilla System I

A Plain Vanilla Model for Adaptive Learning

A plain vanilla system is considered that $\boldsymbol{\alpha}(t)$ can be measured with no error. In addition, it is assumed that between time epoches t and $t + 1$, one only receives training on one skill, and there is only one learning material for each skill. That is, $\mathcal{D}_t = \{1, ..., K\}$ and S is a $K \times K$ identity matrix, that is, $S_{dk} = 1$ if and only if $d = k$. It means that if the k th skill is trained, $\alpha_k(t)$ evolves to $\alpha_k(t+1)$ in a Markov fashion and the proficiency levels on the other skills remain unchanged. Thus, with an abuse of notation, the reward is denoted as $R_k(\alpha_k(t))$, if skill k is trained at time t. The expected immediate reward is required:

$$r_k(\alpha_k(t)) = E[R_k(\alpha_k(t))|\alpha_k(t)] \tag{6}$$

to be nonnegative and bounded, for each time t and each state $\alpha_k(t)$ at time t.

Following the discussion in the ''Recommendation Strategy'' section, the authors consider an infinite time horizon and consider to maximize the expected total-discounted reward Equation 5. This problem is a reformulation of the famous multiarmed bandit problem. As introduced in the sequel, its optimal solution is characterized by the well-known *Gittins indices* that greatly reduce the dimensionality of the problem and its solution (Gittins & Jones, 1974).

Optimal Recommendation

The Gittins index strategy, denoted as π^*, makes recommendation as follows: At time t,

$$d^*(t) = \arg\max_k G_k(\alpha_k(t)), \tag{7}$$

where the $G_k(\cdot)$s are known as the Gittins indices. Before formally defining the Gittins indices, a few remarks are made. First, using the Gittins index strategy, the recommendation at time t only depends on the current states $\boldsymbol{\alpha}(t) = (\alpha_1(t), ..., \alpha_K(t))$, but not the learning history. Second, as will be introduced in the sequel, the function $G_k(\cdot)$ only depends on information concerning skill k, which greatly reduces the dimensionality of the problem. Finally, the Gittins index strategy π^* optimizes the total-discounted reward as defined by Equation 5.

The authors proceed to define Gittins index $G_k(\alpha_k)$. To do so, skill k is assumed to be the only available skill, and thus, the subscript k is removed for simplicity. A recommendation strategy π only decides whether to stop or continue at each time epoch t. Specially, each strategy corresponds to a stopping rule, determined by the first time that certain states are achieved. This time point is random and is known as a stopping time, denoted by τ. Then the Gittins index is defined as

$$G(\alpha) = \sup_{\pi} g(\pi, \alpha), \tag{8}$$

where

$$g(\pi, \alpha) = \frac{E_\pi\left(\sum_{t=0}^{\tau-1} \beta^t R(\alpha(t)) | \alpha(0) = \alpha\right)}{E_\pi\left(\sum_{t=0}^{\tau-1} \beta^t | \alpha(0) = \alpha\right)}. \quad (9)$$

The optimality of the Gittins index strategy is described using the following theorem that is due to Gittins and Jones (1974), and a simpler proof can be found in Weber et al. (1992).

Theorem 1: The Gittins index strategy optimizes the total-discounted reward Equation 5. That is, $V_{\pi^*}(\boldsymbol{\alpha}) \geq V_\pi(\boldsymbol{\alpha})$, for any initial state $\boldsymbol{\alpha}$ and any other strategy π.

Although the definition of $G(\alpha)$ looks complicated, it is not difficult to compute (see Chakravorty & Mahajan, 2014, for the computational details). In particular, a largest-remaining-index algorithm for computing $G(\alpha)$ is described in the appendix.

Specifically, when only mastery/nonmastery of the skills is considered, that is, $\alpha_k(t) \in \{0, 1\}$, and retrogress does not exist, that is, $P(\alpha_k(t+1) = 0 | \alpha_k(t) = 1) = 0$, then,

$$G_k(0) = r_k(0) \text{ and } G_k(1) = 0,$$

where r_k is defined in Equation 6. It means that at each time t, one selects to train a skill that has not been mastered and has the largest expected reward. For example, when the reward takes the form of Equation 3, which skill to train depends on the importance of the skill, c_k, and the chance of mastering the skill after one training $P(\alpha_k(t+1) = 0 | \alpha_k(t) = 1) = 0$.

Plain Vanilla System II

In the previous section, an infinite time horizon system is considered. In this section, a system with a finite time horizon is provided and the training time of each skill is taken into account. It is assumed that in between two time epoches t and $t+1$, the learner is only trained on one skill, that is, $\mathcal{D}_t = \{1, ..., K\}$ and $S_{dk} = 1$, if and only if $d = k$. Only the mastery/nonmastery of the skills is considered, that is, $\alpha_k(t) \in \{0, 1\}$. In addition, once skill k starts to be trained, the training will not be stopped until mastery. That means, if $d(t) = k$, $P(\alpha_k(t+1) = 1 | \alpha_k(t) = 0) = 1$ and the other skills remain unchanged. Again, it is assumed there is no measurement error. The randomness of this system comes from the amount of training time. That is, the amount of time spent on skill k is random, denoted by T_k. It is further assumed that $ET_k = \mu_k$, where μ_k is known. When μ_k is unknown, it can be estimated from data and then plugged into the system.

In this system, it is known that all the skills will be mastered after K time epoches, as one masters one skill at a time. Thus, it is a finite horizon problem with $T = K$ time epoches in total. The following reward function at time t is considered:

$$R_k(\alpha(t)) = \sum_{s=1}^{K} -c_s T_k 1_{\{\alpha_s(t+1) = 0\}}, \quad (10)$$

where c_s is the weight associated with skill s representing its importance, and keeping skill s not mastered for T_k amount of time results in a cost of $c_s T_k$, that is, a negative reward $-c_s T_k$. Under this reward function, a large cost occurs if the training of an important skill is postponed or if a skill that takes a large amount of time is prioritized. Therefore, choosing the right skill to train becomes a trade-off between choosing a skill of importance (large c_k) and short training time (small T_k). The result of Smith (1956) leads to the following theorem.

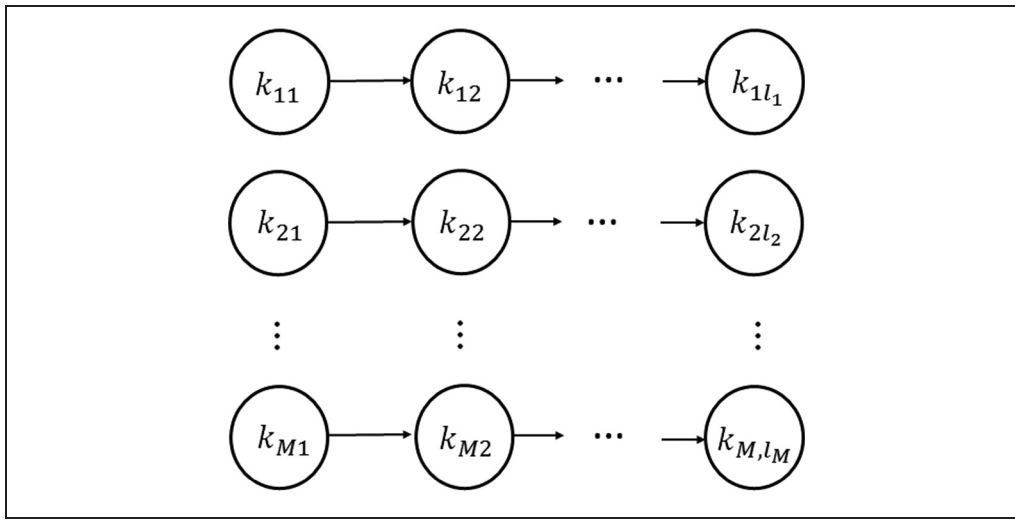

Figure 4. A skill hierarchy with a chain structure.

Theorem 2: To maximize the expected total reward Equation 4, at each time t, the optimal recommendation is to learn a skill with largest value of c_k/μ_k among the unmastered ones.

Despite its simplicity, this learning system does take learning time into account. The resulting optimal strategy is intuitive and easy to implement. The authors would like to point out that the optimal recommendation strategy is derived from the well-known $c-\mu$ *rule* given by Smith (1956) in the operations research literature. There, c refers to the cost of waiting and μ refers to the expected waiting time. A short list of subsequent works includes Cox and Smith (1961); Klimov (1975); Baras, Dorsey, and Makowski (1985); Buyukkoc, Varaiya, and Walrand (1985); Harrison (1988); and Hirayama, Kijima, and Nishimura (1989).

This system and the $c-\mu$ rule may be extended to handle cases with skill hierarchy. In particular, a skill hierarchy with a chain structure is considered as in Figure 4, where the chains may have different number of skills. The previous system can be viewed as its special case, where there are K chains and each chain has only one skill. In such a system, the learner is not allowed to proceed to a skill, until all the prerequisite skills have been mastered. In addition, once a skill has been mastered, a chain structure of skills remains. To describe the optimal recommendation strategy, the remaining skills are indexed as in Figure 4, where there are M chains and the skills in the m th chain are indexed by $k_{m1}, k_{m2}, \ldots k_{m,l_m}$. To describe the optimal recommendation strategy, the priority index is defined first.

$$\rho(m,l)=\frac{\sum_{i=1}^{l} c_{k_{ml}}}{\sum_{i=1}^{l} \mu_{k_{ml}}},$$

and let

$$(m^*,l^*)=\arg\max_{m,\,1\le l\le l_m}\rho(m,l).$$

The optimal recommendations for the next l^* steps are described in the following theorem.

Theorem 3 (Pinedo, 2008): To maximize the expected total reward Equation 4, the optimal strategy is to learn skills $k_{m^*1}, k_{m^*2}, \ldots k_{m^*l^*}$ subsequently, in the next l^* steps.

Examples

Examples of Gittins Index

An example is provided to illustrate the calculation of Gittins index. Consider a system consisting of two skills ($K = 2$). Assume that their proficiency has three levels, corresponding to three states in the Markov learning model, that is, $M = 3$. Also assume that the reward function is defined as in Equation 3, with $c_k = 1$, that is, $R_k(\alpha_k(t)) = \alpha_k(t+1) - \alpha_k(t)$. In addition, the transition probabilities are recorded in matrix $\mathbf{P}_1 = (P_{1,nm})_{M\times M}$ and $\mathbf{P}_2 = (P_{2,nm})_{M\times M}$,

$$\mathbf{P}_1 = \begin{pmatrix} 0.5 & 0.5 & 0 \\ 0 & 0.1 & 0.9 \\ 0 & 0 & 1 \end{pmatrix}, \mathbf{P}_2 = \begin{pmatrix} 0.4 & 0.6 & 0 \\ 0.1 & 0.5 & 0.4 \\ 0 & 0 & 1 \end{pmatrix},$$

where $P_{k,nm} = P(\alpha_k(t+1) = m - 1 | \alpha_k(t) = n - 1)$, $m, n = 1, \ldots, M$. Note that $P(\alpha_k(t+1) = 2 | \alpha_k(t) = 2) = 1$ for both skills, meaning that once the highest level of proficiency has been achieved, one never retrogresses. By comparing P_1 and P_2, it is seen that Skill 1 is relatively more difficult to learn at the beginning ($\alpha_1(t) = 0$). That is, the probability of not making a progress, $P(\alpha_1(t+1) = 0 | \alpha_1(t) = 0) = 0.5$, while $P(\alpha_2(t+1) = 0 | \alpha_2(t) = 0) = 0.4$. In addition, the expected learning time from 0 proficiency to 1 for Skills 1 and 2 are 2 and 1.7, respectively. Moreover, improving from Level 1 to Level 2 is easier for Skill 1, the probability of which is .9, whereas that for Skill 2 is more difficult.

The authors set the discount factor to be $\beta = .9$. Then, the values of the Gittins index can be obtained using Algorithm 1 in the appendix:

$$G_1(0) = 0.63, G_1(1) = 0.9, \text{ and } G_1(2) = 0;$$

$$G_2(0) = 0.60, G_2(1) = 0.34, \text{ and } G_2(2) = 0.$$

Given the Gittins index values and the current proficiency level, the next step can be easily determined. Table 2 enumerates all situations for which at least one skill has not been completely mastered and the corresponding optimal recommendations. A few remarks are made. First, the recommendation strategy based on the Gittins indices is as follows. For a beginner with $\alpha_1 = 0$ and $\alpha_2 = 0$, the optimal strategy is to learn Skill 1 in the next step, as $G_1(0) > G_2(0)$. For a learner with $\alpha_1 = 1$ and $\alpha_2 = 0$, the Gittins design still recommends to learn Skill 1, as $G_1(1) > G_2(0)$. Once Skill 1 has been completely mastered, no more reward can be gained from learning Skill 1, as $G_1(2) = 0$. One then turns to learning Skill 2. Second, the Gittins index takes into consideration all the future rewards, not merely the reward in the immediate step. This is in contrast to the so-called myopic rule, which only looks at the next step. In this example, for $\boldsymbol{\alpha}(t) = (0, 0)$, the expected rewards in the next step are $r_1(0) = 0.5$ and $r_2(0) = 0.6$, respectively. This means that the myopic strategy would choose to learn Skill 2 first, instead of Skill 1. To see why the Gittins index is more preferable to the myopic strategy, it is noticeable that if Skill 1 is trained, the learner has a 50% chance to improve to

Table 2. Current Proficiency Levels and the Corresponding Optimal Recommendations Under the First Example.

$\boldsymbol{\alpha}(t)$	(0,0)	(0,1)	(0,2)	(1,0)	(1,1)	(1,2)	(2,0)	(2,1)
$d^*(t)$	1	1	1	1	1	1	2	2

Table 3. Current Proficiency Levels and the Corresponding Optimal Recommendations Under the Second Example.

$\boldsymbol{\alpha}(t)$	(0,0)	(0,1)	(0,2)	(1,0)	(1,1)	(1,2)	(2,0)	(2,1)
$d^*(t)$	2	1	1	2	1	1	2	2

$\alpha_1(t + 1) = 1$, which can be easily further improved to complete mastery. On the contrary, improving from Level 1 to Level 2 is much more difficult for Skill 2.

In the above example, the optimal strategy is to first learn Skill 1 until it is completely mastered and then learn Skill 2. However, depending on the setting, one may switch between learning two skills. For example, let the transition matrix $\mathbf{P}_2$ be as follows and the rest of the settings remain unchanged:

$$\mathbf{P}_2 = \begin{pmatrix} 0.3 & 0.7 & 0 \\ 0.1 & 0.5 & 0.4 \\ 0 & 0 & 1 \end{pmatrix}.$$

Simple calculation shows that the Gittins indices for Skill 2 becomes $G_2(0) = 0.70$, $G_2(1) = 0.34$, and $G_2(2) = 0$. Under this setting, comparing them with the Gittins indices of Skill 1, it is concluded that when $\alpha_2(t) = 0$, one would choose to learn Skill 2, until it reaches 1, then turn to Skill 1 to completely master it, and finally, back to Skill 2. The optimal recommendations are summarized in Table 3.

Simulated Example: Gittins Optimal Design

A small simulation study is conducted to compare the Gittins optimal design and a randomized strategy. The setting of the first example in the previous subsection is used. The Gittins optimal design is denoted by π^*. For the randomized strategy, an unmastered skill is selected randomly for the next step. This strategy is denoted by π^0. The initial states $(0, 0)$, $(1, 0)$, $(0, 1)$, and $(1, 1)$ are considered. For each initial state, 10,000 trajectories are generated. For each trajectory, the authors compute the observed reward $\sum_{t=0}^{\infty} \beta^t R_{d(t)}(\boldsymbol{\alpha}(t))$ and report the average above 10,000 trajectories. The results are shown in Table 4. As can be seen, for each initial state, the recommendation strategy based on the Gittins index on average generates a larger total-discounted reward than the randomized strategy. Although the differences between two designs are relatively small from Table 4, the Gittins optimal design tends to outperform the randomized strategy by a significant amount in a more complex learning system, with more skills and more states for each skill.

Simulated Examples: $c - \mu$ Rule Design

Two simulated examples of the $c - \mu$ rule optimal design are provided. First, $K = 5$ skills are considered. For skill k, the learning time T_k is assumed to follow the exponential distribution

Table 4. The Mean of Observed Total-Discounted Rewards Above 10,000 Trajectories and Its Standard Error for the Gittins Optimal Design and a Randomized Strategy.

$\boldsymbol{\alpha}(0)$	(0,0)	(1,0)	(0,1)	(1,1)
$\bar{V}_{\pi^*}$	2.89(0.004)	2.43(0.002)	2.30(0.004)	1.71(0.002)
$\bar{V}_{\pi^0}$	2.83(0.005)	2.40(0.003)	2.20(0.005)	1.66(0.003)

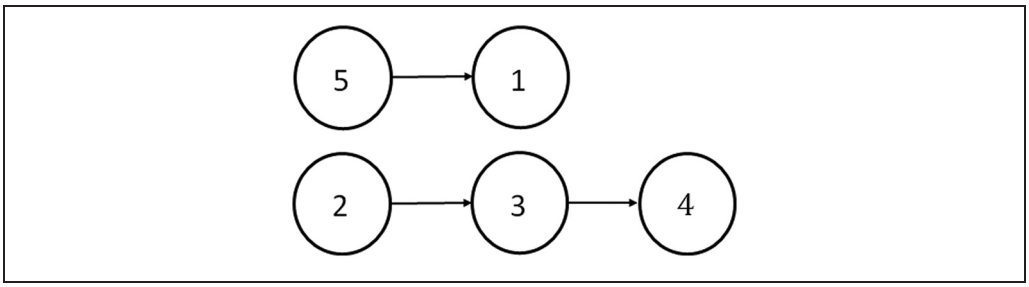

Figure 5. An example of a skill hierarchy with a chain structure.

with mean $\mu_k = k$. In addition, the weights $(c_1, ..., c_5) = (2, 2, 1, 1, 1)$ are set. Under the reward function (10) and the expected total reward (Equation 4) as the objective function, the $c - \mu$ priority indices are $(2, 1, 1/3, 1/4, 1/5)$, respectively. Then the optimal strategy, denoted by π^*, is to learn Skills 1 to 5 sequentially. The authors compare the optimal strategy with a randomized strategy that randomly selects an unmastered skill to learn. This strategy is denoted by π^0. They consider the initial state $\boldsymbol{\alpha}(0) = (0, 0, 0, 0, 0)$, that is, none of the skills has been mastered yet. The mean of observed total reward above 10,000 trajectories and its standard error for π^* are $-21.1(0.11)$ and that for π^* are $-43.1(0.30)$. Therefore, the $c - \mu$ rule optimal design on average has a considerably larger total reward gain over the randomized strategy.

Next, a skill hierarchy constraint is imposed upon the above example, where the hierarchical structure is specified in Figure 5. The optimal strategy based on Theorem 3 is to train skills in the sequence of 2, 5, 1, 3, and 4. Note that it is prioritized, even though the importance of Skill 5 is relatively low ($c_5 = 1$) and the expected learning time is long ($\mu_5 = 5$). This is due to the hierarchical structure of the skills, as Skill 1 can only be unfrozen after the mastery of Skill 5, which is an importance skill ($c_1 = 2$) and can be learned quickly ($\mu_1 = 1$). Again, the optimal design is denoted by π^* and is compared with a randomized strategy π^0 that randomly selects a skill that is currently available. For initial state $\boldsymbol{\alpha}(0) = (0, 0, 0, 0, 0)$, the mean of observed total rewards above 10,000 trajectories and its standard error for π^* are $-35.3(0.23)$ and that for the randomized strategy π^0 are $-43.4(0.28)$. Again, as expected, the optimal design substantially outperforms the randomized strategy.

Discussion

Designing and analyzing an adaptive learning system is a challenging task. It is an interdisciplinary research problem that requires expertise from such fields as psychometrics, education, psychology, statistics, operations research, and machine learning. Existing developments in these fields may provide key building blocks for more comprehensive solutions.

As a first step toward solving this problem, the authors formulate the problem mathematically under the Markov decision framework and borrow ideas from psychometrics, statistics, and operations research. This framework consists of three key components, the measurement model, the learning model, and the recommendation strategy. Concrete examples are provided for each component and relevant literature is discussed to provide intuition, motivation, and justification.

The main result of this article is a recommendation strategy, which is formulated as a stochastic scheduling problem. It associates learning behavior and priority through certain reward functions and evaluates recommendation strategies by the total reward gain. Under this setup, an optimal recommendation strategy is well defined as the strategy that provides the largest total reward gain. Two plain vanilla recommendation systems are presented for adaptive learning, for which the optimal designs are given by the well-known Gittins index and $c - \mu$ rule, respectively.

Many practical issues need to be taken into consideration, before the proposed framework is applied. First, what would be a reasonable reward function and what would be a proper objective function to optimize? This question needs to be answered by experiments and with the help of experts in the subject domain. Second, a reasonable learning model should incorporate more factors. For example, learner specific covariates, such as gender and age, may affect the learning transition and the training time and thus should be incorporated into the learning model. In the plain vanilla systems, it is assumed that each learning material only trains one skill, and that the mastery of one skill does not affect the learning of the others. These assumptions may need to be relaxed, as a learning material may provide training on multiple skills or the mastery of one skill may increase the chance of mastering others. Third, the analytically tractable strategies in the plain vanilla systems are obtained assuming no measurement error. With the presence of measurement error, how would these strategies perform if the estimated proficiency levels are plugged in? Alternatively, measurement error may be incorporated into the system using a Bayesian approach. Fourth, the expected rewards and the transition kernels are unknown in practice and typically need to be learned from data in an online fashion. As such, one has to simultaneously learn the expected rewards and the transition kernels and find a good recommendation strategy. This leads to an exploitation–exploration trade-off. That is, one could miss out on some large rewards if sticking to the current knowledge of the system (exploitation) and lose time by collecting small rewards if exploring too much (exploration). This problem has been studied in the reinforcement learning literature and considerable progress has been made in the domain of robotics (Sutton & Barto, 1998). Further investigations are certainly needed along these directions.

Appendix

The authors present a largest-remaining-index algorithm (Varaiya,Walrand, & Buyukkoc, 1985). It computes the Gittins indices of one skill.

Algorithm 1: Largest-Remaining-Index Algorithm

Input: discount factor β, transition matrix $\mathbf{P} = (P_{nm})_{M \times M}$, and the expected one-step reward $\mathbf{r} = (r(1), \ldots, r(M))$.

1. Find the state with largest expected one-step reward $k_1 = \arg\max_k \{r(k) : k = 1, ..., K\}$, and the corresponding Gittins index is $G(k_1) = r(k_1)$.
2. The algorithm then proceeds recursively. Let $k_1, \ldots, k_{l-1}$, are given. k_l and $G(k_l)$ are obtained as follows. Let $T^{(l)} = (T^{(l)}_{nm})_{M \times M}$

$$T_{nm}^{(l)} = \begin{cases} P_{nm} & \text{if } m \in \{k_1, ..., k_{l-1}\}, \\ 0 & \text{otherwise}, \end{cases}$$
$$\mathbf{d}^{(l)} \triangleq \left(d_1^{(l)}, ..., d_M^{(l)}\right) = \left[I - \beta T^{(l)}\right]^{-1} \mathbf{r},$$
$$\mathbf{b}^{(l)} \triangleq \left(b_1^{(l)}, ..., b_M^{(l)}\right) = \left[I - \beta T^{(l)}\right]^{-1} \mathbf{1},$$

where I is a $M \times M$ identity matrix and $\mathbf{1}$ is a length M vector whose components are all 1. Choose

$$k_l = \underset{k \notin \{k_1, ..., k_{l-1}\}}{\arg\max} \; d_k^{(l)} / b_k^{(l)},$$
$$\text{and } G(k_l) = d_{k_l}^{(l)} / b_{k_l}^{(l)}$$

3. Repeat 2 for $l = 2, ..., M$.

In the above algorithm, ties are broken arbitrarily whenever they exist.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was funded by NSF Grant SES-1323977, NSF Grant IIS-1633360, Army Research Office Grant W911NF-15-1-0159, and NIH Grant R01GM047845.

References

Baras, J., Dorsey, A., & Makowski, A. (1985). Two competing queues with linear costs and geometric service requirements: The μc-rule is often optimal. *Advances in Applied Probability*, *17*, 186-209.

Bellman, R. (1954). The theory of dynamic programming. *Bulletin of the American Mathematical Society*, *60*, 503-515.

Berger, J. O. (2013). *Statistical decision theory and Bayesian analysis*. New York, NY: Springer.

Buyukkoc, C., Varaiya, P., & Walrand, J. (1985). The c μ rule revisited. *Advances in Applied Probability*, *17*, 237-238.

Cappé, O., Moulines, E., & Rydén, T. (2005). *Inference in hidden Markov models*. New York, NY: Springer.

Chakravorty, J., & Mahajan, A. (2014). Multi-armed bandits, Gittins index, and its calculation. In N. Balakrishnan (Ed.), *Methods and applications of statistics in clinical trials: Planning, analysis, and inferential methods* (pp. 416-435). New York, NY: Wiley.

Chen, Y., Liu, J., Xu, G., & Ying, Z. (2015). Statistical analysis of Q-matrix based diagnostic classification models. *Journal of the American Statistical Association*, *110*, 850-866.

Chen, Y., Liu, J., & Ying, Z. (2015). Online item calibration for Q-matrix in CD-CAT. *Applied Psychological Measurement*, *39*, 5-15.

Cheng, Y. (2009). When cognitive diagnosis meets computerized adaptive testing: CD-CAT. *Psychometrika*, *74*, 619-632.

Cox, D. R., & Smith, W. L. (1961). *Queues*. London, England: Methuen.
de la, & Torre, J. (2011). The generalized DINA model framework. *Psychometrika*, *76*, 179-199.
Durbin, R., Eddy, S. R., Krogh, A., & Mitchison, G. (1998). *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. New York, NY: Cambridge University Press.
Felsenstein, J., & Churchill, G. A. (1996). A hidden Markov model approach to variation among sites in rate of evolution. *Molecular Biology and Evolution*, *13*, 93-104.
Gittins, J. C., Glazebrook, K., & Weber, R. (2011). *Multi-armed bandit allocation indices*. West Sussex, UK: Wiley.
Gittins, J. C., & Jones, D. M. (1974). A dynamic allocation index for the sequential design of experiments. In J. Gani, K. Sarkadi & I. Vince (Eds.), *Progress in statistics* (pp. 241-266). Amsterdam, The Netherlands: North-Holland.
Gosavi, A. (2009). Reinforcement learning: A tutorial survey and recent advances. *INFORMS Journal on Computing*, *21*, 178-192.
Gross, D. (2008). *Fundamentals of queueing theory*. New York, NY: Wiley.
Harrison, J. M. (1988). Brownian models of queueing networks with heterogeneous customer populations. In W. Fleming & P. Lions (Eds.), *Stochastic differential systems, stochastic control theory and applications* (pp. 147-186). New York, NY: Springer.
Henson, R. A., Templin, J. L., & Willse, J. T. (2009). Defining a family of cognitive diagnosis models using log-linear models with latent variables. *Psychometrika*, *74*, 191-210.
Hirayama, T., Kijima, M., & Nishimura, S. (1989). Further results for dynamic scheduling of multiclass G/G/1 queues. *Journal of Applied Probability*, *26*, 595-603.
Junker, B. W., & Sijtsma, K. (2001). Cognitive assessment models with few assumptions, and connections with nonparametric item response theory. *Applied Psychological Measurement*, *25*, 258-272.
Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, *101*, 99-134.
Kaya, Y., & Leite, W. L. (2016). Assessing change in latent skills across time with longitudinal cognitive diagnosis modeling: An evaluation of model performance. *Educational and Psychological Measurement*. Advance online publication. doi:10.1177/0013164416659314
Kim, S., Shephard, N., & Chib, S. (1998). Stochastic volatility: Likelihood inference and comparison with arch models. *The Review of Economic Studies*, *65*, 361-393.
Klimov, G. (1975). Time-sharing service systems. I. *Theory of Probability & Its Applications*, *19*, 532-551.
Lai, T. L., & Robbins, H. (1985). Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, *6*, 4-22.
Lai, T. L., & Ying, Z. (1988). Open bandit processes and optimal scheduling of queueing networks. *Advances in Applied Probability*, *20*, 447-472.
Leighton, J. P., Gierl, M. J., & Hunka, S. M. (2004). The attribute hierarchy method for cognitive assessment: A variation on Tatsuoka's rule-space approach. *Journal of Educational Measurement*, *41*, 205-237.
Li, F., Cohen, A., Bottge, B., & Templin, J. (2016). A latent transition analysis model for assessing change in cognitive skills. *Educational and Psychological Measurement*, *76*, 181-204.
Liu, J., Xu, G., & Ying, Z. (2012). Data-driven learning of Q-matrix. *Applied Psychological Measurement*, *36*, 548-564.
Liu, J., Xu, G., & Ying, Z. (2013). Theory of the self-learning Q-matrix. *Bernoulli*, *19*(5A), 1790-1817.
Liu, J., Ying, Z., & Zhang, S. (2015). A rate function approach to computerized adaptive testing for cognitive diagnosis. *Psychometrika*, *80*, 468-490.
Liu, Y., Magnus, B., Quinn, H., & Thissen, D. (2016). Multidimensional item response theory. In P. Irwing, D. Hughes & T. Booth (Eds.), *Handbook of psychometric testing*. Hoboken, NJ: Wiley-Blackwell.
Means, B., Toyama, Y., Murphy, R., Bakia, M., & Jones, K. (2009). *Evaluation of evidence-based practices in online learning: A meta-analysis and review of online learning studies*. Washington, DC: U.S. Department of Education.

Mulder, J., & van der Linden, W. J. (2009). Multidimensional adaptive testing with Kullback–Leibler information item selection. In W. J. van der Linden & C. Glas (Eds.), *Elements of adaptive testing* (pp. 77-101). New York, NY: Springer.

Nino-Mora, J. (2009). Stochastic scheduling. In C. A. Floudas & P. M. Pardalos (Eds.), *Encyclopedia of optimization* (pp. 3818-3824). New York, NY: Springer.

Norris, J. R. (1998). *Markov chains*. New York, NY: Cambridge University Press.

Pinedo, M. L. (2008). *Scheduling: Theory, algorithms and systems*. New York, NY: Springer.

Reckase, M. (2009). *Multidimensional item response theory*. New York, NY: Springer.

Robbins, H. (1952). Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, *58*, 527-535.

Rupp, A. A., & Templin, J. L. (2008). Unique characteristics of diagnostic classification models: A comprehensive review of the current state-of-the-art. *Measurement*, *6*, 219-262.

Rupp, A. A., Templin, J. L., & Henson, R. A. (2010). *Diagnostic measurement: Theory, methods, and applications*. New York, NY: Guilford Press.

Segall, D. O. (1996). Multidimensional adaptive testing. *Psychometrika*, *61*, 331-354.

Smith, W. E. (1956). Various optimizers for single-stage production. *Naval Research Logistics Quarterly*, *3*, 59-66.

Sun, J., Chen, Y., Liu, J., Ying, Z., & Xin, T. (2016). Latent variable selection for multidimensional item response theory models via ${L}_{1}$ regularization. *Psychometrika*, *81*, 921-939.

Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA: MIT Press.

Tatsuoka, K. K. (1983). Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement*, *20*, 345-354.

Templin, J. L., & Henson, R. A. (2006). Measurement of psychological disorders using cognitive diagnosis models. *Psychological Methods*, *11*, 287-305.

Varaiya, P., Walrand, J., & Buyukkoc, C. (1985). Extensions of the multiarmed bandit problem: The discounted case. *IEEE Transactions on Automatic Control*, *30*, 426-439.

von Davier, M. (2008). A general diagnostic model applied to language testing data. *British Journal of Mathematical and Statistical Psychology*, *61*, 287-307.

Wang, C., & Chang, H.-H. (2011). Item selection in multidimensional computerized adaptive testing–gaining information from different angles. *Psychometrika*, *76*, 363-384.

Wang, S., Yang, Y., Culpepper, S., & Douglas, J. (2016). *Tracking skill acquisition with cognitive diagnosis models: Application to spatial rotation skills* (Unpublished manuscript).

Weber, R. (1992). On the Gittins index for multiarmed bandits. *The Annals of Applied Probability*, *2*, 1024-1033.

Webley, K. (2013). A is for adaptive. *Time*, *181*(23), 40-45.

Xu, G., & Zhang, S. (2015). Identifiability of diagnostic classification models. *Psychometrika*, *81*, 1-25.

Xu, X., Chang, H., & Douglas, J. (2003). *A simulation study to compare CAT strategies for cognitive diagnosis*. Paper presented at the annual meeting of the American Educational Research Association, Chicago, IL.

Zhang, S., & Chang, H.-H. (2016). From smart testing to smart learning: How testing technology can assist the new generation of education. *International Journal of Smart Technology and Learning*, *1*, 67-92.