A LATENT HIDDEN MARKOV MODEL FOR PROCESS DATA

XUEYING TANG

UNIVERSITY OF ARIZONA

Response process data from computer-based problem-solving items describe respondents' problem-solving processes as sequences of actions. Such data provide a valuable source for understanding respondents' problem-solving behaviors. Recently, data-driven feature extraction methods have been developed to compress the information in unstructured process data into relatively low-dimensional features. Although the extracted features can be used as covariates in regression or other models to understand respondents' response behaviors, the results are often not easy to interpret since the relationship between the extracted features, and the original response process is often not explicitly defined. In this paper, we propose a statistical model for describing response processes and how they vary across respondents. The proposed model assumes a response process follows a hidden Markov model given the respondent's latent traits. The structure of hidden Markov models resembles problem-solving processes, with the hidden states interpreted as problem-solving subtasks or stages. Incorporating the latent traits in hidden Markov models enables us to characterize the heterogeneity of response processes across respondents in a parsimonious and interpretable way. We demonstrate the performance of the proposed model through simulation experiments and case studies of PISA process data.

Key words: response process, latent variable, hidden Markov models, problem-solving behaviors.

Problem-solving is one of the most important skills in the 21st century (Binkley et al., 2012). Evaluation of problem-solving competency has gained increasing popularity in large-scale assessments. For example, Problem-Solving in Technology-Rich Environments (PSTRE) was one of the three domains of the first cycle of the Programme of the International Assessment of Adult Competencies (PIAAC). In the Programme of International Student Assessment (PISA), surveys were conducted to measure students' problem-solving and collaborative problem-solving skills in 2012 and 2015, respectively. In these assessments, respondents' skills are measured by computer-based interactive items where respondents are required to fulfill a real-life task through an interface in a simulated computer environment. The use of computers in these items not only facilitates the simulation of problem-solving scenarios, but also enables more comprehensive data collection. When a respondent solves an interactive item, computer log files keep track of the actions taken by the respondent within the interface (e.g., mouse clicks and keyboard inputs). After the completion of the item, the recorded action sequence allows us to reproduce the main process of solving the item and are thus called response process data.

Compared to traditional dichotomous or polytomous item responses that only record the final response outcomes, process data contain detailed information on how respondents solved an item. These data not only record whether the item was answered correctly or incorrectly, but also demonstrate how the answer was reached. Thus, process data provide a valuable source for understanding respondents' problem-solving behaviors and improving current psychometric practice. Recent studies have shown that process data are useful for accurate latent trait assessment (Zhang et al., 2023), comprehensive diagnostic classification (Liang et al., 2022; Zhan & Qiao,

2022) , early detection of task failure (Ulitzsch et al., 2022b) , and problem-solving strategy analysis (Wang et al., 2022) .

The rich information in response process data comes with difficulties in data analysis. First of all, process data have a nonstandard format. The traditional item response of an item is a single variable. In contrast, the corresponding response process is a sequence of categorical variables (actions), and the number of categories (unique actions) is often a few dozen or more. Moreover, the length of the sequence varies greatly across respondents. The nonstandard data format prevents the direct application of many conventional item response models such as item response theory models (Lord, 1980) and cognitive diagnostic models (Rupp et al., 2010) to process data. Besides, the dependence structure among actions in a response process is often complicated. Usually, the action to be taken in the next step depends on not only the current action, but also actions taken several steps back, making simple models such as first-order Markov models insufficient for characterizing process data. The unprecedented opportunities and challenges in analyzing response process data call for innovative methods for utilizing the rich behavioral information in the data.

Feature extraction is a common approach to circumventing the difficulties brought by the non-standard format of response processes. In this approach, a fixed number of features are extracted for each process and then fed to regression or other supervised learning methods to explore the relationship between response processes and other variables of interest. For example, Chen et al. (2019a) and Ulitzsch et al. (2022b) used a collection of summary statistics of response processes to predict the final response outcome to achieve early detection of failure. He and von Davier (2016) and Stadler et al. (2019) identified the differences in the behavior patterns among various groups of respondents with the help of n-gram features. Zhang et al. (2023) developed methods for obtaining accurate latent trait assessment through process features.

Traditionally, process features are summary statistics of a response process such as sequence length, action counts, and item response time (Chen et al., 2019a; Ulitzsch et al., 2022b) . Features derived from existing cognitive theory are also used in top-down research studies (Greiff et al., 2016; von Davier et al., 2019) . Such features usually do not comprehensively summarize response processes and thus may overlook important information for understanding respondents' problem-solving behaviors. Moreover, theory-based features are often item-specific and time-consuming to construct if a wide range of items are available. Recently, data-driven feature extraction methods (Tang et al., 2020, 2021) have been developed. These methods do not require substantial prior knowledge about the item. They, with the help of machine learning and statistical tools, compress as much information in response processes into the extracted features as possible. However, the relationship between the extracted features and the original response process is often not explicitly defined. As a result, although the extracted features can be easily incorporated in psychometrically meaningful machine learning tasks and often lead to performance improvement, identifying the characteristics of response processes that contribute to the improvement requires additional careful examination.

In this paper, we take a different route to analyzing process data. We propose a latent hidden Markov model (LHMM) to describe respondents' problem-solving processes directly. The proposed model offers three distinct features that set it apart from previous models. First, it naturally accounts for the long-term dependence among actions by adopting hidden Markov models as the basic framework for characterizing response processes. Second, the hidden states in HMMs can be interpreted as problem-solving states or subtasks, providing a detailed understanding of response processes from a subtask perspective (Wang et al., 2022) . Third, a latent trait variable is introduced in HMM to characterize the respondents' heterogeneous behaviors in transitioning between subtasks and completing a subtask parsimoniously. Compared to the analyses based on feature extraction methods, the proposed model directly links respondents' latent traits to their problem-solving behaviors, providing more straightforward and interpretable results.

Although several models have been recently proposed to characterize response processes (Chen, 2020; Han et al., 2021; Xiao et al., 2021; Xu et al., 2020) , none of them possesses all three features of the proposed model. Xiao et al. (2021) also used HMMs to describe response processes, but they did not take into account respondents' differences in problem-solving processes. Xu et al. (2020) proposed a latent topic model with Markovian transition for process data. This model is essentially an HMM with personalized state transition probability matrices but a shared state-action probability matrix. However, since their model does not use latent trait variables to parameterize the personalized state transition probability matrices, it is difficult to visualize the differences in the response behaviors. Without considering the long-term dependence among actions, Han et al. (2021) proposed a sequential response model which assumes response processes follow a first-order Markov model after conditioning on the latent trait. Finally, Chen (2020) developed a dynamic choice measurement model for process data based on marked point processes. This model takes into account both the heterogeneity of respondents' behaviors and the complex action dependence structure. However, it requires a significant amount of expert input to specify the dependence structure.

The rest of the paper is organized as follows. We first present an example of problem-solving items and process data in Sect. 1 to motivate the proposed latent hidden Markov model described in Sect. 2. In Sect. 3, we explain how the parameters and latent variables in the proposed model are estimated. Case studies of process data from two problem-solving items in PISA 2012 are presented in Sect. 4 to demonstrate the performance of the proposed model. Simulation studies examining the performance of statistical inference in different scenarios are presented in Sect. 5. We conclude with final remarks in Sect. 6.

1. An Example of Problem-Solving Items and Process Data

In this section, we describe the climate control (CC) item in PISA 2012 as an example of items producing process data. This item is one of the 42 items in the survey for assessing students' problem-solving skills. In the CC item, students are asked to figure out how to use a new air conditioner to control room climate. The item interface, presented in Fig. 1, includes an air-conditioner with three control bars. Each bar, controlling either the temperature or the humidity of the room, can be placed at five different positions, "−−" (−2), "−" (−1), default position (0), "+" (1), and "++" (2). Students are required to determine which climate variable that each bar influences by conducting experiments through the interface. They can slide the bars and click the APPLY button to read the humidity and temperature under the current setting from the charts in the interface. Based on the results, they need to match the control bars with the two climate variables.
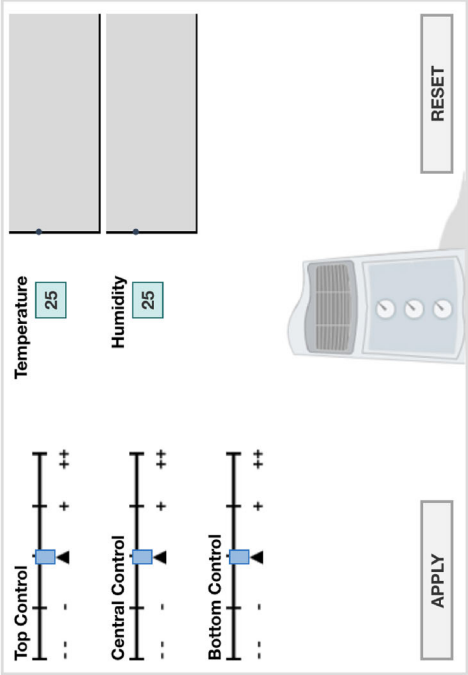
When students solve the item, the experiments they conduct and the buttons they click are recorded sequentially in the computer log files. The CC item has 126 recorded actions, including one action of clicking the RESET button and 125 ($= 5^3$) actions describing the setting of an experiment conducted by a student. Each of the experiment setting actions describes the positions of the three control bars when the APPLY button is clicked. For example, action "−1_2_0" means a student places the top, middle, and bottom controls at "−," "++," and the default position, respectively, and then clicks the APPLY button. A recorded sequence describes the entire response of a student and is one observation of process data. For example, the sequence "1_0_0, RESET, 0_0_ − 2" shows that the student conducted two experiments when solving the item. In the first experiment, the top control was moved to "+," followed by a click of the APPLY button. Then, the positions of the three controls were reset by a click of the RESET button. Finally, the bottom control was moved to "−−," and the APPLY button was clicked to conduct the second experiment.

CLIMATE CONTROL

You have no instructions for your new air conditioner. You need to work out how to use it.

You can change the top, central and bottom controls on the left by using the sliders (➡). The initial setting for each control is indicated by ◀.

By clicking APPLY, you will see any changes in the temperature and humidity of the room in the temperature and humidity graphs. The box to the left of each graph shows the current level of temperature or humidity.

Question : CLIMATE CONTROL

Find whether each control influences temperature and humidity by changing the sliders. You can start again by clicking RESET.

Draw lines in the diagram on the right to show what each control influences.

To draw a line, click on a control and then click on either Temperature or Humidity. You can remove any line by clicking on it.

FIGURE 1.
Interface of the climate control item in PISA 2012.

In this paper, we present a statistical model describing such detailed response processes of a group of respondents.

2. Model Description

In this section, we describe a latent hidden Markov model (LHMM) for response process data. We use $Y = (Y_1, \ldots, Y_T)$ to denote a generic response process (action sequence) of an item. For $t = 1, \ldots, T$, the t-th element Y_t of Y is a random variable denoting the t-th action taken in the process and it takes value from the action set $\mathcal{A}$ containing M possible actions of the item. For simplicity, the actions in $\mathcal{A}$ are represented by $1, \ldots, M$. We use $y = (y_1, \ldots, y_T)$ to represent a realization of Y. Given a sequence $x = (x_1, \ldots, x_T)$ and indices $i < j$, we use $x_{i:j}$ to denote the subsequence of x consisting of $x_i, x_{i+1}, \ldots, x_j$. In the following, we first describe the idea of using HMMs as a framework for modeling response processes in Sect. 2.1. Then, in Sect. 2.2, latent trait variables are introduced to HMMs to account for individual differences in response processes.

2.1. Hidden Markov Models

Under an HMM, the evolution of a response process Y is determined by a hidden state sequence $S = (S_1, \ldots, S_T)$ that evolves according to a first-order Markov model. Each element in S indicates the hidden state of the step. It takes value from the set $\mathcal{S} = \{1, 2, \ldots, K\}$ containing K possible hidden states. In HMMs, the evolution of the action sequence and hidden states is characterized by three groups of parameters, a K-dimensional initial state probability vector $\pi = (\pi_1, \ldots, \pi_K)$, a $K \times K$ state transition probability matrix $P = (p_{kl})$ describing the evolution of the hidden states, and a $K \times M$ state-action probability matrix $Q = (q_{kj})$ describing the action distributions under different hidden states. Let $s = (s_1, \ldots, s_T)$ be a realization of S. In the initial step, the hidden state S_1 follows the initial state probability distribution π with

$$P(S_1 = s_1) = \pi_{s_1}. \tag{1}$$

Given $S_1 = s_1$, the distribution of the initial action Y_1 is determined by s_1 and Q as

$$P(Y_1 = y_1 \mid S_1 = s_1) = q_{s_1, y_1}. \tag{2}$$

For $t = 2, \ldots, T$, given all previous hidden states $S_{1:(t-1)}$ and all previous actions $Y_{1:(t-1)}$, the distribution of hidden state S_t depends only on S_{t-1} as

$$P(S_t = s_t \mid S_{1:(t-1)} = s_{1:(t-1)}, Y_{1:(t-1)} = y_{1:(t-1)}) = P(S_t = s_t \mid S_{t-1} = s_{t-1}) = p_{s_{t-1}, s_t}. \tag{3}$$

Given all previous actions $Y_{1:(t-1)}$ and all hidden states up to step t, action Y_t is determined by S_t only as

$$P(Y_t = y_t \mid Y_{1:(t-1)} = y_{1:(t-1)}, S_{1:t} = s_{1:t}) = P(Y_t = y_t \mid S_t = s_t) = q_{s_t, y_t}. \tag{4}$$

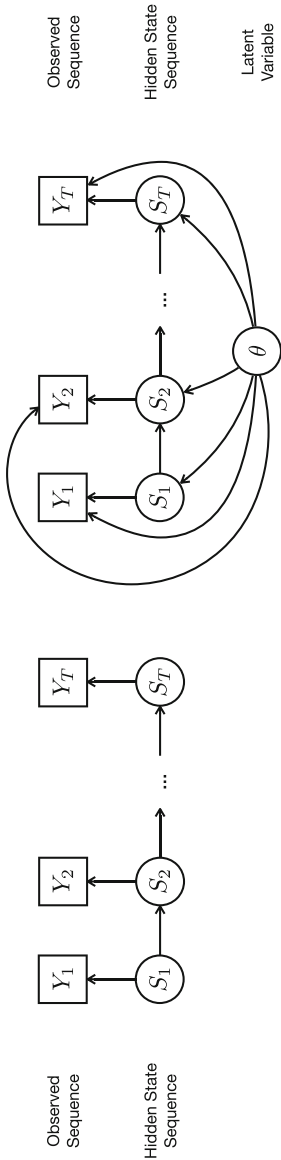The structure of HMMs is demonstrated in the left panel of Fig. 2.

FIGURE 2.
Structure of HMM (left) and LHMM (right).

The structure formed by the observed sequence (action sequence) and the hidden state sequence in HMMs resembles problem-solving processes. Solving a complex task often involves completing a few simpler subtasks or going through several problem-solving stages. For example, solving the CC item described in Sect. 1 could involve two stages. After the item starts, students may enter an exploration stage in which they explore the interface and try to understand the effects of clicking buttons and sliding control bars. Once they become familiar with the interface, an efficient problem-solving stage may start. The role of problem-solving stages or subtasks in response processes is similar to that of the hidden states in HMMs. In HMMs, the hidden states evolve according to a Markov model and decide the action distribution at each step. In a response process, the stage or subtask changes as the process progresses, and how actions are taken varies across different stages or subtasks. In the efficient problem-solving stage of the CC item, students are more likely to take actions that are directly related to identifying the climate variable associated with each control bar. For example, they may conduct experiments with only one bar moved away from the default position, making actions such as "−1_0_0," "0_2_0" and "0_0_1" more likely be taken. Such pattern may not exist in the exploration stage. With the analogy between hidden states and problem-solving stages, the state transition probability matrix P describes how respondents move between the stages, and the state-action probability matrix Q describes how actions are used differently in different stages.

Using HMMs for modeling response processes also enables us to incorporate long-term dependence among actions in a response process. Actions in a response process are not taken independently. The action that a respondent would take at a given step could depend on not only the action taken in the step immediately before the current one, but also the actions taken several steps back. Although the strong conditional independence assumptions (3) and (4) imposed on the hidden states and actions limit the complexity of the dependence structure that HMMs could characterize, we will see later in Sect. 4 that such simplified structure can reasonably describe response processes in our case studies and that the hidden states can indeed be interpreted as problem-solving stages.

2.2. Latent Hidden Markov Models

The above HMM representation of action sequences does not explicitly account for individual differences in problem-solving. If two respondents took the same actions in steps 1 to t, then, under HMMs, the distributions of the action at step $t + 1$ would be the same for the two respondents. The difference in the actual actions taken at this step is purely explained by randomness. However, respondents usually behave differently in solving problems. For instance, in the CC item, students with more advanced problem-solving skills may have a higher chance to enter the efficient problem-solving stage. They may also start the response processes in the efficient problem-solving stage without going through the exploration stage. These differences lead to different initial probability vectors, state transition probability matrices, and action emission probability matrices for different respondents in the framework of HMMs. To incorporate such individual variation in problem-solving, we introduce an additional latent variable in HMM, producing a latent HMM (LHMM), which is described in detail below.

Let θ denote a uni-dimensional variable describing a latent trait of a respondent. Throughout this paper, we assume θ follows the standard normal distribution for simplicity. Under LHMM, given θ, the response process follows the HMM whose parameters π, P, and Q are further parametrized as follows. For initial state probabilities,

$$\pi_k(\theta) = P(S_1 = k \mid \theta) = \begin{cases} \frac{1}{1+\sum_{k'=2}^{K} \exp(\tau_{k'}\theta+\mu_{k'})} & \text{for } k = 1; \\ \frac{\exp(\tau_k\theta+\mu_k)}{1+\sum_{k'=2}^{K} \exp(\tau_{k'}\theta+\mu_{k'})} & \text{for } k = 2, \ldots, K. \end{cases} \quad (5)$$

For $k = 1, \ldots, K$, the probability of a hidden state jumping from state k to state l is

$$
p_{kl}(\theta) = P(S_{t+1} = l \mid S_t = k, \theta) = \begin{cases} \frac{1}{1+\sum_{l'=2}^{K} \exp(a_{kl'}\theta + b_{kl'})} & \text{for } l = 1; \\[2mm] \frac{\exp(a_{kl}\theta + b_{kl})}{1+\sum_{l'=2}^{K} \exp(a_{kl'}\theta + b_{kl'})} & \text{for } l = 2, \ldots, K; \end{cases} \tag{6}
$$

and the probability of taking action j in state k is

$$
q_{kj}(\theta) = P(Y_t = j \mid S_t = k, \theta) = \begin{cases} \frac{1}{1+\sum_{j'=2}^{M} \exp(c_{kj'}\theta + d_{kj'})} & \text{for } j = 1; \\[2mm] \frac{\exp(c_{kj}\theta + d_{kj})}{1+\sum_{j'=2}^{M} \exp(c_{kj'}\theta + d_{kj'})} & \text{for } j = 2, \ldots, M. \end{cases} \tag{7}
$$

In other words, given θ, the initial hidden states S_1, other hidden states S_t given the previous hidden states S_{t-1}, and the action Y_t given the current state S_t all follow multinomial logistic models (MLM; McCullagh & Nelder, 2018) with θ as the covariate and state 1 or action 1 as the baseline category. The structure of the model is depicted in the right panel of Fig. 2.

In (5), (6) and (7), τ_k, μ_k, a_{kl}, b_{kl}, c_{kj}, d_{kj} are real-valued parameters. We write $\boldsymbol{\mu} = (\mu_2, \ldots, \mu_K)$, $\boldsymbol{\tau} = (\tau_2, \ldots, \tau_K)$, $\boldsymbol{A} = (a_{kl})$, $\boldsymbol{B} = (b_{kl})$, $\boldsymbol{C} = (c_{kj})$, and $\boldsymbol{D} = (d_{kj})$ as compact notation of the parameters where $\boldsymbol{A}$ and $\boldsymbol{B}$ are $K \times (K-1)$ matrices and $\boldsymbol{C}$ and $\boldsymbol{D}$ are $K \times (M-1)$ matrices. In addition, we also use $\boldsymbol{\eta}$ to denote the vector collecting all parameters ($\boldsymbol{\mu}$, $\boldsymbol{\tau}$, $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, and $\boldsymbol{D}$) in LHMM.

Under the model for the state transition probability in (6), given $S_t = k$, the log odds of jumping to state l versus state 1 at step $t + 1$ is $a_{kl}\theta + b_{kl}$. Parameter b_{kl} is the log odds when $\theta = 0$. Parameter a_{kl} controls how sensitive the log odds are to the change of θ. If $a_{kl} > 0$, a larger θ leads to higher odds of jumping to state l against state 1. If $a_{kl} < 0$, a larger θ leads to lower odds. In the CC item, if θ represents students' problem-solving skill proficiency, we may expect the odds of jumping from the exploration state to the efficient problem-solving state against staying in the exploration state to increase as θ increases. If the exploration stage is the baseline state (State 1), then a_{12} is expected to be positive. If $a_{kl} = 0$ for $l = 2, \ldots, K$, then the state transition distribution for state k does not depend on θ and is completely determined by $b_{kl}, l = 2, \ldots, K$. Parameters $\boldsymbol{\tau}$ and $\boldsymbol{\mu}$ for modeling the initial state probabilities in (5) and parameters $\boldsymbol{c}$ and $\boldsymbol{d}$ for modeling the state-action probabilities in (7) can be interpreted similarly.

3. Statistical Inference

In this section, we describe how to estimate the parameters $\boldsymbol{\eta}$, latent trait θ, and hidden states $\boldsymbol{S}$ in LHMM when a set of response processes from different respondents is available. We distinguish response processes and hidden state sequences from different respondents through superscripts. For example, $\boldsymbol{Y}^{(i)} = (Y_1^{(i)}, \ldots, Y_{T_i}^{(i)})$ is the response process of respondent i and $\boldsymbol{S}^{(i)} = (s_1^{(i)}, \ldots, s_{T_i}^{(i)})$ is the corresponding (unobserved) hidden state sequence. The latent trait of respondent i is denoted by θ_i. The set $\mathcal{Y}_n = \{\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(n)}\}$ collects n observed response processes from n respondents. In the following, we first give the likelihood function for LHMMs in Sect. 3.1 and then describe how to obtain the marginalized maximum likelihood estimator of the model parameters in Sect. 3.2. Latent trait estimation and hidden state estimation are then discussed in Sects. 3.3 and 3.4, respectively.

3.1. Likelihood Function

Since both θ_i and the hidden state sequence $S^{(i)}$ in LHMMs are unobservable, the (marginalized) likelihood function of η given the i-th observed response process $Y^{(i)} = y^{(i)}$ is

$$L_i(\eta) = P\left(Y^{(i)} = y^{(i)} \mid \eta\right) = \int \sum_{s^{(i)}} P\left(Y^{(i)} = y^{(i)}, S^{(i)} = s^{(i)}, \theta_i \mid \eta\right) d\theta_i$$

$$= \int \phi(\theta_i) \sum_{s^{(i)}} P\left(Y^{(i)} = y^{(i)}, S^{(i)} = s^{(i)} \mid \theta_i, \eta\right) d\theta_i, \tag{8}$$

where the summation is over all possible realization $s^{(i)}$ of $S^{(i)}$ and $\phi(\theta) = \frac{1}{\sqrt{2\pi}}e^{-\theta^2/2}$ is the probability density function of the standard normal distribution.

Given θ_i and η, the action sequence $Y^{(i)}$ and the associated hidden state sequence $S^{(i)}$ follow the HMM with the elements of parameters π, P, and Q being $\pi_k(\theta_i)$, $p_{kl}(\theta_i)$, and $q_{kj}(\theta_i)$ defined in (5), (6), and (7). Thus, according to assumptions (1) – (4),

$$P(Y^{(i)} = y^{(i)}, S^{(i)} = s^{(i)} \mid \theta_i, \eta) = \pi^{(i)}_{s_1^{(i)}} q^{(i)}_{s_1^{(i)}, y_1^{(i)}} \prod_{t=2}^{T_i} p^{(i)}_{s_{t-1}^{(i)}, s_t^{(i)}} q^{(i)}_{s_t^{(i)}, y_t^{(i)}}, \tag{9}$$

where $\pi_k^{(i)}$, $p_{kl}^{(i)}$ and $q_{kj}^{(i)}$ are short-hand notation for $\pi_k(\theta_i)$, $p_{kl}(\theta_i)$ and $q_{kj}(\theta_i)$, respectively. Combining (8) and (9), we obtain the (marginalized) likelihood function for a set of response processes $\mathcal{Y}_n$ under LHMM as

$$L(\eta \mid \mathcal{Y}_n) = \prod_{i=1}^n L_i(\eta) = \prod_{i=1}^n \left\{ \int \phi(\theta_i) \sum_{s^{(i)}} \pi^{(i)}_{s_1^{(i)}} q^{(i)}_{s_1^{(i)}, y_1^{(i)}} \prod_{t=2}^{T_i} p^{(i)}_{s_{t-1}^{(i)}, s_t^{(i)}} q^{(i)}_{s_t^{(i)}, y_t^{(i)}} d\theta_i \right\}. \tag{10}$$

The main difficulty in evaluation $L(\eta \mid \mathcal{Y}_n)$ at a given η is to compute the summation over $s^{(i)}$ and the integration over θ_i. Since the number of terms involved in the summation in $L_i(\eta)$ is of order T_i^K, it is computationally burdensome to calculate the summation directly even for a sequence with a moderate length T_i and a small number of hidden states K. A dynamic programming algorithm called forward–backward algorithm (Rabiner & Juang, 1986) has been designed to compute the likelihood function for HMMs. It thus can be used to efficiently compute the summation over $s^{(i)}$ in (10) for a given θ_i. For the integration, although the integral does not have a closed-form expression, it can be computed numerically using Gauss–Hermite quadrature. The details of evaluating the likelihood function $L(\eta \mid \mathcal{Y}_n)$ are provided in Appendix A.

3.2. Parameter Estimation

We estimate the parameter vector η in LHMMs by its maximum likelihood estimator (MLE)

$$\hat{\eta} = \underset{\eta}{\operatorname{argmax}} \, L(\eta \mid \mathcal{Y}_n) = \underset{\eta}{\operatorname{argmax}} \, \log L(\eta \mid \mathcal{Y}_n). \tag{11}$$

Since the likelihood function $L(\eta \mid \mathcal{Y}_n)$ is differentiable and its gradient is computable, gradient-based optimization algorithms can be used to maximize the log-likelihood function $\log L(\eta \mid \mathcal{Y}_n)$. We choose the BFGS algorithm (Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970)

to obtain $\hat{\boldsymbol{\eta}}$ because of its fast convergence rate. Although the gradient of $L(\boldsymbol{\eta} \mid \mathcal{Y}_n)$ can be numerically computed from $L(\boldsymbol{\eta} \mid \mathcal{Y}_n)$, supplying the exact gradient to the BFGS algorithm often leads to faster computation. The expressions of the gradient of $L(\boldsymbol{\eta} \mid \mathcal{Y}_n)$ and how they can be evaluated using dynamic programming and Gauss–Hermite quadrature are given in Appendix B.

Expectation–Maximization (EM) algorithms (Dempster et al., 1977) are often used to compute MLEs for latent variable models such as HMMs. An EM algorithm can be designed for LHMMs, but unlike the EM algorithm for HMM, the optimization problem in the maximization step for LHMMs does not have a closed-form solution, so BFGS or other numerical optimization algorithms are still needed. In addition, numerical integration and dynamic programming are also needed in the expectation step as in the evaluation of the likelihood function. Therefore, EM algorithms do not bring significant convenience in computing the MLE for LHHMs, and we directly maximize the log-likelihood function using the BFGS algorithm instead.

3.3. Latent Trait Estimation

Given the estimated parameters $\hat{\boldsymbol{\eta}}$, the latent trait θ_i is often estimated by the maximum a posteriori (MAP) estimator or the expected a posteriori (EAP) estimator. We adopt the EAP estimator for estimating θ_i because it can be easily computed from the intermediate results of computing $\hat{\boldsymbol{\eta}}$, whereas computing the MAP estimator requires optimizing the posterior density function of θ_i for each i. More specifically, the EAP estimator of θ_i is

$$\hat{\theta}_i = E(\theta_i | \hat{\boldsymbol{\eta}}, \boldsymbol{y}^{(i)}) = \frac{\int \theta_i \sum_{\boldsymbol{s}^{(i)}} P(\boldsymbol{Y}^{(i)} = \boldsymbol{y}^{(i)}, \boldsymbol{S}^{(i)} = \boldsymbol{s}^{(i)} \mid \theta_i, \hat{\boldsymbol{\eta}}) \phi(\theta_i) d\theta_i}{\int \sum_{\boldsymbol{s}^{(i)}} P(\boldsymbol{Y}^{(i)} = \boldsymbol{y}^{(i)}, \boldsymbol{S}^{(i)} = \boldsymbol{s}^{(i)} \mid \theta_i, \hat{\boldsymbol{\eta}}) \phi(\theta_i) d\theta_i}. \tag{12}$$

The denominator in (12) is $L_i(\hat{\boldsymbol{\eta}})$. Its value is already computed in the final iteration of the optimization algorithm for obtaining $\hat{\boldsymbol{\eta}}$. Also, the numerator can be calculated using Gauss–Hermite quadrature. The components needed in the calculation (e.g., quadrature points, weights, and function values at the quadrature points) can be recycled from computing $L_i(\hat{\boldsymbol{\eta}})$ since the integrand in the numerator is the integrand in the denominator multiplied by θ_i.

3.4. Hidden State Estimation

Given the estimated parameters $\hat{\boldsymbol{\eta}}$ and latent trait $\hat{\theta}_i$, we estimate $\boldsymbol{s}^{(i)}$ by the most probable hidden state sequence for the action sequence $\boldsymbol{y}^{(i)}$:

$$\hat{\boldsymbol{s}}^{(i)} = \underset{\boldsymbol{s}^{(i)}}{\operatorname{argmax}} \, P(\boldsymbol{Y}^{(i)} = \boldsymbol{y}^{(i)}, \boldsymbol{S}^{(i)} = \boldsymbol{s}^{(i)} \mid \hat{\theta}_i, \hat{\boldsymbol{\eta}}). \tag{13}$$

As the number of possible hidden state sequences is exponentially large, directly maximizing the probability is computationally expensive. Instead, in light of the connection between HMM and LHMM, we obtain $\hat{\boldsymbol{s}}^{(i)}$ through the Viterbi algorithm (Viterbi, 1967), a dynamic programing algorithm for finding the most probable hidden state sequence for HMM. Details of the algorithm are provided in Appendix C.

We implement the algorithms mentioned in Sects. 3.1, 3.2, 3.3, and 3.4 in R (R Core Team, 2023). The Gaussian quadrature points and weights are computed using the R package statmod (Giner et al., 2023). The function optim() is used to maximize the marginalized likelihood function for LHMMs. The evaluation of the marginalized likelihood function is implemented using Rcpp (Eddelbuettel & François, 2011) to improve the computational speed.

4. Case Studies

In this section, we demonstrate how LHMM helps understand respondents' problem-solving processes through two case studies of response processes from PISA 2012. We consider data from two items, the CC item described in Sect. 1 and the TICKET item to be introduced in Sect. 4.2. The two items represent two important types of interactive problem-solving items: MicroDYN system and finite state automata OECD (2014, Chapter 1).

4.1. Climate Control Item

In this section, we present the analysis of the CC response processes of 350 students from the USA. The lengths of the response processes range from 3 to 128, with the average being 21. More than 90% of the response processes contain fewer than 50 actions. As we described in Sect. 1, the item originally has 126 distinct actions. In this case study, we simplify the 125 experiment setting actions so that each action only reflects which bars are placed at a nonzero position at the time of clicking the APPLY button. The simplification leads to nine distinct actions in total: "RESET," "None," "Top," "Middle," "Bottom," "Top_Middle," "Top_Bottom," "Middle_Bottom," and "All."

The CC item has been well-studied in the literature (Chen et al., 2019a; Greiff et al., 2015; Xu et al., 2020). An efficient way of solving the item is to adopt the Varying-One-Thing-At-a-Time (VOTAT) strategy. The action sequence "Top, RESET, Bottom, RESET, Middle" is an example of response processes adopting this strategy. In this case study, with the help of the proposed LHMM, we examine how students' problem-solving processes vary in terms of adopting the VOTAT strategy. We fit the LHMM with $K = 2$. The choice of K is made based on our understanding of the problem-solving stages involved in the item (Sect. 2.1). Also, the CC item belongs to MicroDYN systems, which typically contains two phases: the knowledge acquisition phase in which students collect information about the required task, and the knowledge application phase in which students solve the required task by applying the acquired knowledge (Herborn et al., 2017). Our choice seems also reasonable from this perspective. We will discuss more about data-driven selection of K in Sect. 4.3. For simplicity, we ignore the individual difference in the initial state probability vector π by setting $\tau = 0$. As a comparison to LHMM, we also fit the HMM with two hidden states to the response processes. The results are presented below.

4.1.1. Comparison Between LHMM and HMM Fits
Since HMM is a special case of LHMM with τ, a, and c being zeros, we compare the goodness of fit of the two models using the likelihood ratio test. The results ($\chi^2(18) = 2394.0$, p-value < 0.001) show that the proposed LHMM provided a better fit than the standard HMM. Also, the Bayesian Information Criterion (BIC) values of HMM and LHMM are 25020.7 and 22786.5, respectively. The results again support incorporating the latent variable θ to explain the heterogeneity of response processes. We focus on the results from LHMM in the rest of the sections. The estimated parameters are provided in Table 3 in Appendix D.

4.1.2. Latent Trait Interpretation
In Fig. 3, we present $\hat{\theta}$ and its relationship with students' binary item responses. The middle panel displays boxplots of $\hat{\theta}$ grouped by students' binary item responses indicating whether the item was answered correctly (1) or incorrectly (0). It shows that students who successfully solved the item tend to have a higher $\hat{\theta}$. In addition, the ROC curve of classifying the item response by $\hat{\theta}$ is plotted in the right panel of Fig. 3. The area under the curve (AUC) is 0.709. These results suggest that the latent trait may be related to students' problem-solving skills. Further examination of the response processes shows that the VOTAT strategy is often used in the response processes with large $\hat{\theta}$, while the response processes with small $\hat{\theta}$ often lack this feature. Table 1 presents a few examples of response processes with top or bottom 5%
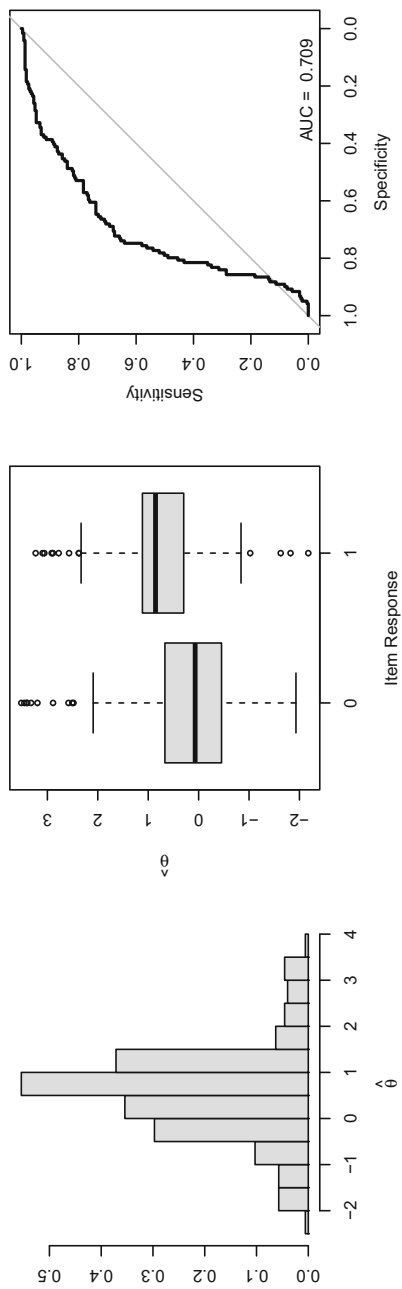
FIGURE 3.

Left: histogram of $\hat{\theta}$ in the CC item. Middle: boxplots of $\hat{\theta}$ grouped by binary item responses. Right: ROC curve of classifying binary item responses using $\hat{\theta}$.

TABLE 1.
Examples of response processes with top or bottom 5% of $\hat{\theta}$.

$\hat{\theta}$	Response Process
2.91	None, RESET, Middle, Middle, RESET, Middle, RESET, Top, None, RESET, Bottom
2.57	None, RESET, Top, RESET, Middle, RESET, Bottom, RESET, None, RESET, Bottom
-1.93	Top_Bottom, Top_Middle_Bottom
-1.82	Top_Bottom, None, Top_Bottom, RESET, Top_Bottom

Note: For ease of presentation, the exhibited response processes are shortened by removing consecutively repeating actions.
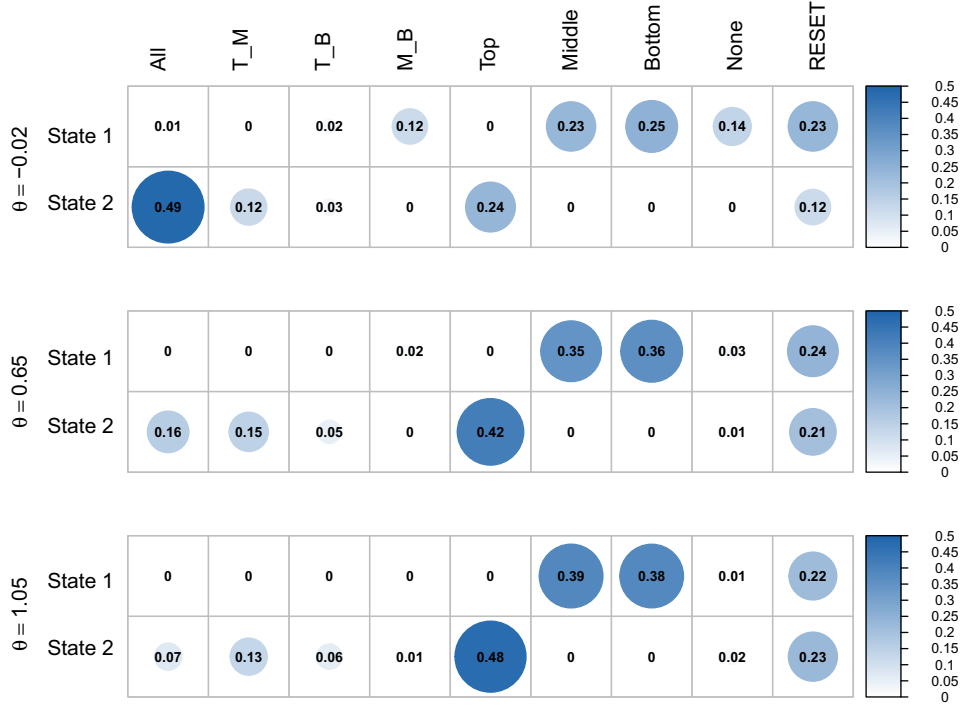

FIGURE 4.
State-action probability matrices at the quartiles of $\hat{\theta}$ in the CC item. The column names T_M, T_B, and M_B stand for actions "Top_Middle," "Top_Bottom," and "Middle_Bottom," respectively.

of $\hat{\theta}$. Since the ability of using the VOTAT strategy is closely related to problem-solving skills, it is not unreasonable to interpret the latent trait θ as students' problem-solving proficiency.

4.1.3. Hidden State Interpretation In Fig. 4, we plot the state-action probability matrices at the quartiles of $\hat{\theta}$ to check the connection between the two hidden states and the problem-solving processes of the CC item. As shown in the figure, actions "Middle," "Bottom," and "RESET" are often used in State 1 but rarely used in State 2. These actions are often used in the VOTAT strategy to isolate the effect of a control bar. In contrast, the actions involving placing multiple bars at nonzero positions (e.g., "All" and "Top_Middle") often have small or near zero probabilities in State 1 and higher probabilities in State 2. These patterns suggest that students often explore the item interface and try to figure out how to solve the problem in State 2 and apply the efficient
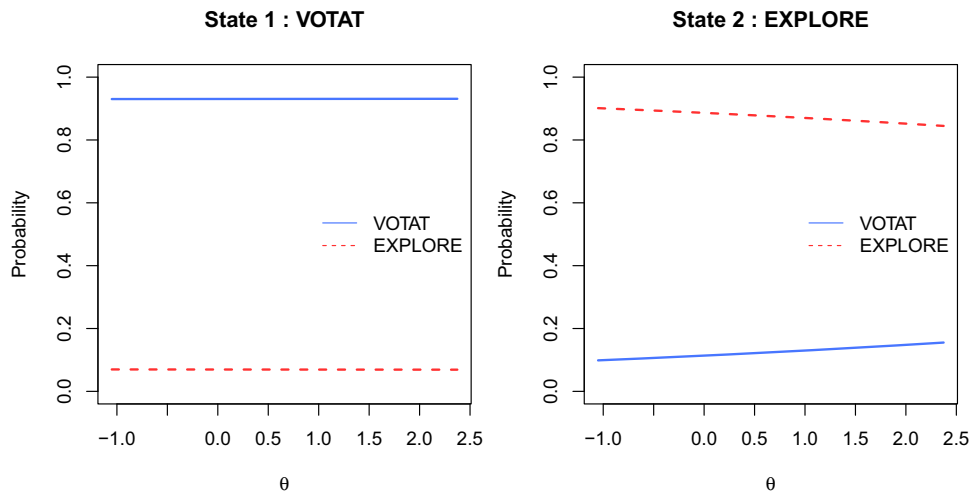
FIGURE 5.
State transition probability curves of the CC item.

VOTAT strategy to solve the problem in State 1. We label States 2 and 1 as the EXPLORE and VOTAT States, respectively.

The probabilities associated with action "Top" under the two states seem to conflict with the interpretation. Intuitively, action "Top" plays a similar role as actions "Middle" and "Bottom." It should be a critical action in applying the VOTAT strategy. However, in the estimated state-action probability matrices, "Top" has a much higher probability in the EXPLORE State than in the VOTAT State. A possible reason for this counterintuitive pattern is that "Top" is also used when students explore the interface since the top bar is often the first control bar one would move. The fitted LHMM tends to assign the same hidden states to "Top" as those for "All," "Top_Middle," and other actions that students used when exploring the interface.

4.1.4. Difference in Response Processes Across Students   We investigate the variation in response processes by examining the state transition and the state-action probability curves shown in Figs. 5 and 6. Regarding the state transition probabilities, in both states, the probability of staying in the same state is high, indicating students often stay in one state for a few steps and then jump to the other state. As θ increases, the probability of staying in the EXPLORE State decreases, while the probability of staying in the VOTAT State has no notable change. These results suggest that students with higher problem-solving proficiency often take fewer actions in the EXPLORE State and reach the efficient VOTAT State faster. Once in the VOTAT State, all students are very likely to stay in the efficient problem-solving state regardless of their problem-solving proficiency.

In terms of state-action probability, students' behaviors vary with θ in both states. In the EXPLORE State, students with a higher θ tend to use diverse actions to explore the interface while those with a lower θ often place all three bars at nonzero positions without exploring other patterns or using the RESET button. In the VOTAT State, the behavior patterns for students with θ between 0.5 and 2.5 do not vary greatly. They mainly use actions "Middle," "Bottom," and "RESET" to apply the VOTAT strategy. On the other hand, students with a lower θ are more likely to take actions "Middle_Bottom," "Top_Bottom," and "None." By examining the action sequences and corresponding estimated hidden state sequences of these students, we found that some of these students seem to use a different way to implement the VOTAT strategy. Instead of placing only one bar at a nonzero position at a time, they place one additional bar at a nonzero position at
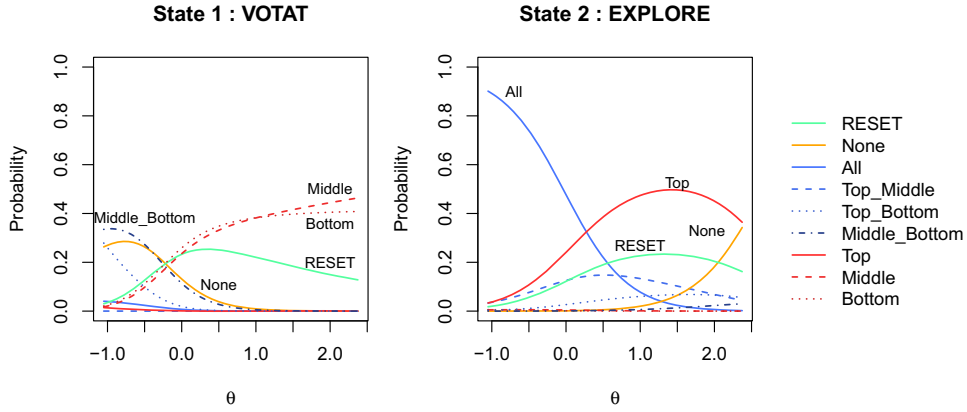
FIGURE 6.
State-action probability curves of the CC item.

a time. Inspecting the change in the temperature and humidity readings under the two settings provides information on the climate variable controlled by the additional bar. Action sequence "Middle, Middle_Bottom, Top_Middle_Bottom, Middle_Bottom, Bottom, Top_Bottom" is an example of the response processes performing VOTAT in this way. Students can identify the climate variable associated with the bottom bar by comparing the readings under settings "Middle" and "Middle_Bottom." Similarly, the readings from "Middle_Bottom" and "Bottom" provide information about the middle bar. The results from LHMM can help us distinguish the two different ways of implementing the VOTAT strategy to some extent.

4.2. TICKET Item

In the section, we focus on the response processes from the TICKET item. The item requires students to purchase a full fare country train ticket with two individual trips through an automated ticketing machine. The interactive screen of the ticketing machine, along with operating instructions, is included in the item interface. Figure 7 presents screenshots of different pages shown on the ticketing machine screen and how the screen changes in response to button clicks. The upper left screenshot gives the initial page. The same operating instructions are shown in the left panel of each page. They are omitted in the screenshots of subsequent pages to save space. The flow of the pages is marked by the arrows in the figure. When solving the item, a student needs to choose the train network (country trains or city subway), fare type (full fare or concession), and pricing basis (daily or individual trips) for the tickets in sequence. The student will also be asked to choose the number of trips if individual trip tickets are chosen. Tickets are purchased after clicking the BUY button on the final page. If the CANCEL button is clicked on any page, then the screen will return to the initial page with all previous choices cleared. The TICKET item involves 13 distinct actions, each corresponding to clicking a button in the interface. The descriptions of these actions are provided in Table 2.

Our dataset contains the response processes of 417 students from the USA. The process length ranges from 4 to 32, with a mean of 6.58 and a standard deviation of 3.71. The action counts and proportions are presented in the last column of Table 2. Among the 417 students, 272 (65.2%) answered the item correctly.

As described above, the TICKET item requires students to make three major choices regarding the network, fare type, and pricing basis of the ticket to be bought. Choosing each aspect of the ticket can be seen as a subtask of the original task. However, the transitions of the subtasks are

FIGURE 7.
Interface of the TICKET item in PISA 2012.

TABLE 2.
Actions in the TICKET item.

Action	Description	Count (Proportion)
country_train	Click COUNTRY TRAINS button	398 (14.5%)
city_subway	Click CITY SUBWAY button	173 (6.3%)
full_fare	Click FULL FARE button	412 (15.0%)
concession	Click CONCESSION button	120 (4.3%)
daily	Click DAILY button	107 (3.9%)
individual	Click INDIVIDUAL button	399 (14.6%)
trip_1	Choose tickets for one individual trip	54 (2.0%)
trip_2	Choose tickets for two individual trips	391 (14.3%)
trip_3	Choose tickets for three individual trips	38 (1.4%)
trip_4	Choose tickets for four individual trips	16 (0.6%)
trip_5	Choose tickets for five individual trips	26 (0.9%)
Buy	Click BUY button to purchase the chosen tickets	403 (14.7%)
Cancel	Click CANCEL button	205 (7.5%)

completely determined by the action in the current step, clearly violating the model assumptions of LHMM. Despite this, we still fit LHMM with three hidden states ($K = 3$) to examine whether LHMM can recover the subtask structure under model misspecification. Because of the design of the interface, all students start the item by choosing the train network, so we ignore the individual difference in the initial state probability vector π by setting $\tau = 0$. The HMM with three hidden states is also fitted to the data for comparison.

4.2.1. Comparison Between LHMM and HMM Fits   Similar to the analysis of the CC item, we use the likelihood ratio test and BIC to compare the model fits of LHMM and HMM. The likelihood ratio test gives test statistic $\chi^2(42) = 872.0$ and p-value< .001. The BIC values of HMM and LHMM are 8663.8 and 8124.3, respectively. These results indicate that LHMM is more appropriate to describe the response processes than HMM. In the remaining parts of the section, we examine the results from LHMM in detail. The estimated parameters are provided in Table 4 in Appendix D.

4.2.2. Latent Trait Interpretation   The left panel of Fig. 8 presents the histogram of the estimated latent traits $\hat{\theta}$ for the 417 students. A tall bar on the high end of $\hat{\theta}$ stands out in the graph. It exists because a significant proportion of students have identical action sequences "country_trains, full_fare, individual, trip_2, Buy," leading to identical $\hat{\theta}$ values for these students.

   The boxplots in the middle panel show that students who answered the item correctly tend to have a smaller $\hat{\theta}$, and those who answered the item incorrectly tend to have a larger $\hat{\theta}$. The ROC curve in the right panel of Fig. 8 has AUC 0.899, which is close to 1. Based on these results, we interpret the latent trait as students' problem-solving skills that the TICKET item is designed to assess. With this interpretation, it is evident that students with the action sequence "country_trains, full_fare, individual, trip_2, Buy" should have the highest $\hat{\theta}$ as this action sequence corresponds to the most succinct way of answering the item correctly.

4.2.3. Connection Between Hidden States and Subtasks   To examine how the hidden states in LHMM are connected with problem-solving subtasks, we present in Fig. 9 the state-action probability matrices at the quartiles of $\hat{\theta}$. In State 1, actions "country_train" and "city_subway"
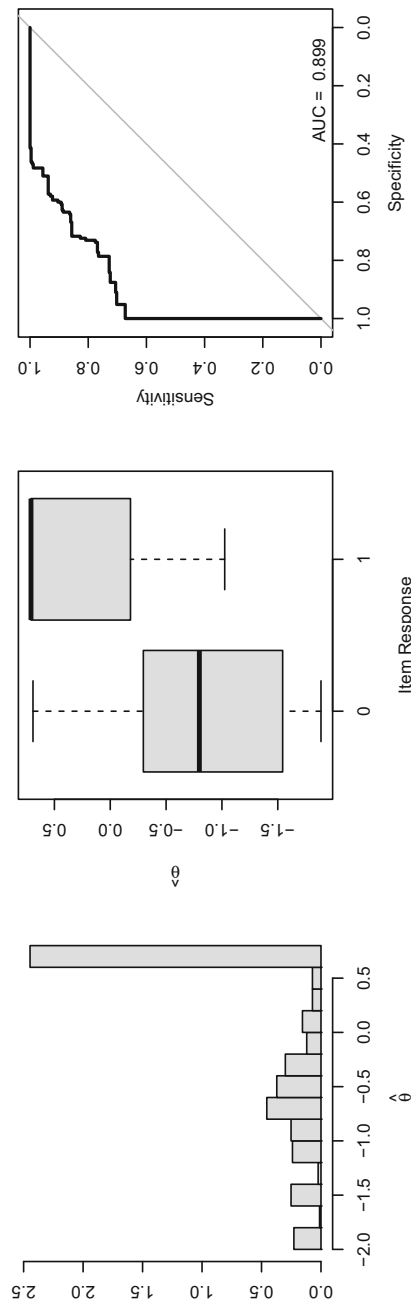
FIGURE 8.

Left: histogram of $\hat{\theta}$ in the TICKET item. Middle: boxplots of $\hat{\theta}$ grouped by binary item responses. Right: ROC curve of classifying binary item responses using $\hat{\theta}$.
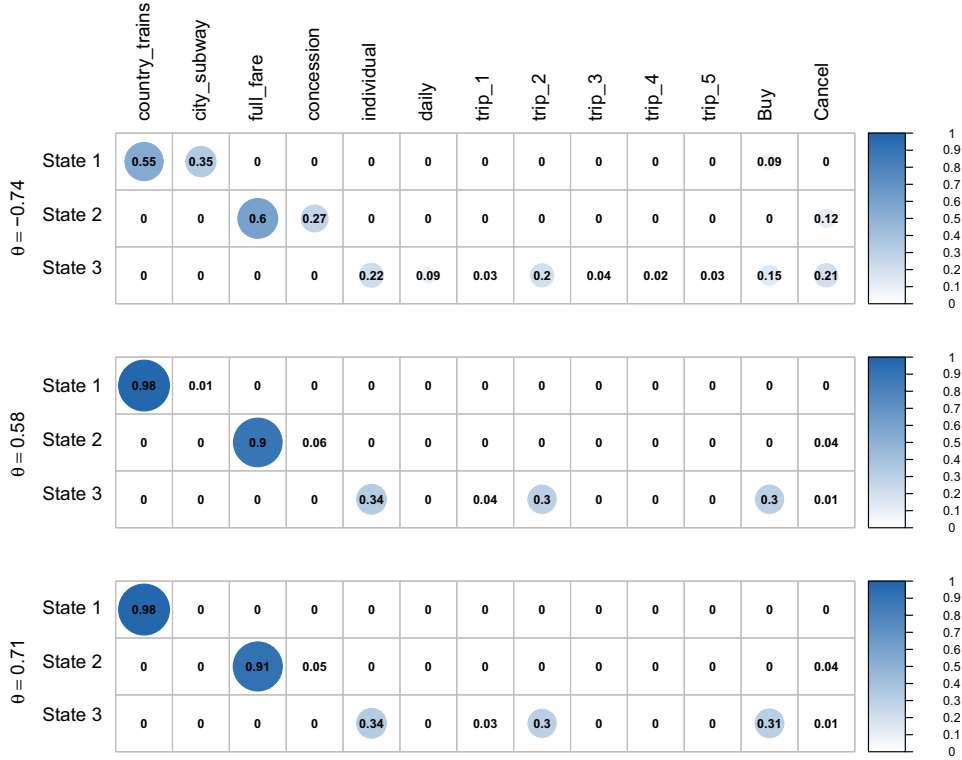
FIGURE 9.
State-action probability matrices at the quartiles of $\hat{\theta}$ in the TICKET item.

are taken with high probabilities. Since the two actions are the only possible choices for the train network, State 1 is related to the subtask of choosing the train network. Similarly, State 2 is related to the subtask of choosing fare type as actions "full_fare" and "concession" are taken with high probabilities in this state. Finally, in State 3, the high probability actions involve choosing between daily and individual trip tickets and the number of individual trips. Hence, the state is related to the subtask of choosing the pricing basis (and related details) of the tickets. Although LHMM is not provided with the information on how the required task should be solved, the hidden states in the fitted model can be linked to the subtasks of solving the required task. This suggests that LHMM is able to capture the structure of students' response processes of the TICKET item. For ease of reference, we label the three states as the Network state, the Fare Type state, and the Pricing state, respectively.

4.2.4. Difference in Response Processes Across Students With the interpretations of the latent trait and hidden states in mind, we are now ready to examine how the response processes vary across students. Figure 10 shows the curves of state-action probabilities as functions of the latent trait θ. In the Network state, as θ increases, the probability of taking the correct action ("country_train") increases while the probability of taking the incorrect action ("city_subway") decreases. Similar patterns are also observed in the Fare Type and Pricing states. Since θ is interpreted as students' problem-solving proficiency, the results agree with the intuition that students who are better at problem-solving will more likely make the correct choices.

Figure 11 exhibits the curves of state transition probabilities. According to the figure, when in the Network state (left panel), the students will almost surely jump to the Fare Type state in
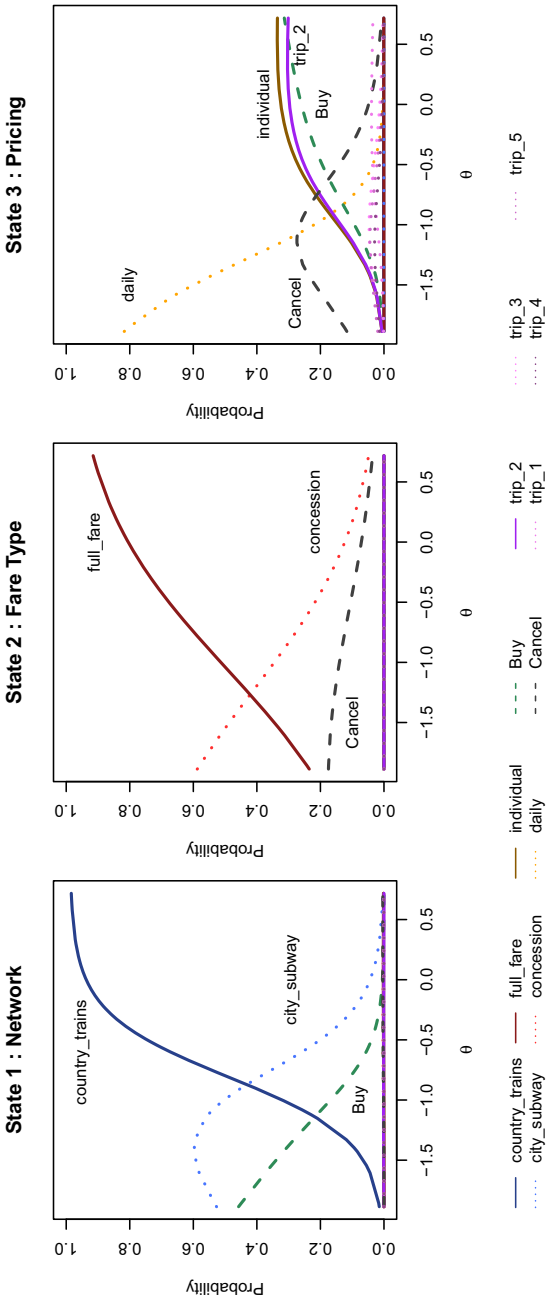
FIGURE 10.
State-action probability curves of the TICKET item.

the next step regardless of students' problem-solving proficiency θ. If the current state is the Fare Type state (middle panel), the next state is either the Pricing state or the Network state. Among the two states, the Pricing state takes about 90% of the probability, and the percentage increases slightly as θ increases. If the current state is the Pricing state (right panel), the state either stays at the Pricing state (for selecting the number of individual trips) or returns to the Network state. Students with higher θ values are more likely to stay in the Pricing state. The transition from Fare Type or Pricing to Network is a result of clicking the CANCEL button to correct previous choices or explore the interface more. Students with better problem-solving skills are less likely to make mistakes and need less exploration. Hence, they are less likely to restart the selection process and more likely to remain on the track of choosing the network, fare type, and pricing basis to complete the required task.

Note that the state transition patterns obtained from LHMM are consistent with how the screen pages are switched in the interface although such information was not utilized in model fitting. It again suggests that LHMM is able to characterize the structure of the response processes of the TICKET item. The resemblance between state transitions and page changes further confirms the links between hidden states and the subtasks.

4.3. Selection of K

Fitting LHMMs requires a pre-specified number of hidden states K. In the case studies, we set $K = 2$ in the CC item and $K = 3$ in the TICKET item based on our understanding of the required tasks in the two items. For both items, we tried $K = 2, 3, 4, 5, 6$ and found that the chosen K produces the most interpretable results. Data-driven methods for setting K in an HMM are available. Chapter 15.6 of Cappé et al. (2005) gives a penalized maximum likelihood method for selecting K. The resulting criterion for selection is similar to BIC but with a heavier penalty on the model complexity. For the CC item, the modified BIC values for the LHMM with $K = 2, 3, 4, 5, 6$ are 22820.11, 22724.54, 22298.61, 23597.99, 24391.41 with $K = 4$ corresponding to the smallest value. However, the values for $K = 2, 3, 4$ do not vary very much. Our choice $K = 2$ also seems reasonable in terms of the criterion. For the TICKET item, the modified BIC values are 9402.99, 8586.84, 9169.67, 9695.15, and 10094.50. According to this criterion, $K = 3$ is selected, matching our choice. Although strong consistency has been established for the modified BIC, we do not recommend to select K solely based on the criterion. Whether the obtained results are interpretable is also an important factor. Finally, as far as the authors' knowledge, the ordinary BIC has not been theoretically justified for selecting K in HMMs or LHMMs as the model with a smaller K is located on the boundary instead of the interior of the parameter space of the model with a larger K.

5. Simulation Studies

In this section, we demonstrate the performance of LHMM through simulation studies. We investigate 1) the performance of estimating the parameters, latent traits, and hidden states in LHMM and 2) how well the proposed model can be distinguished from the ordinary HMM.

5.1. Settings

For the first aim, we generate action sequences from the LHMM described in Sect. 2.2 with three hidden states ($K = 3$) and ten actions ($M = 10$). The parameters of the LHMM are presented in Table 5 in Appendix E. These parameters are chosen so that the resulting probability curves (solid lines in Fig. 14) resemble those estimated from the TICKET item. Datasets with different combinations of sample size n and average sequence length L are generated to investigate how
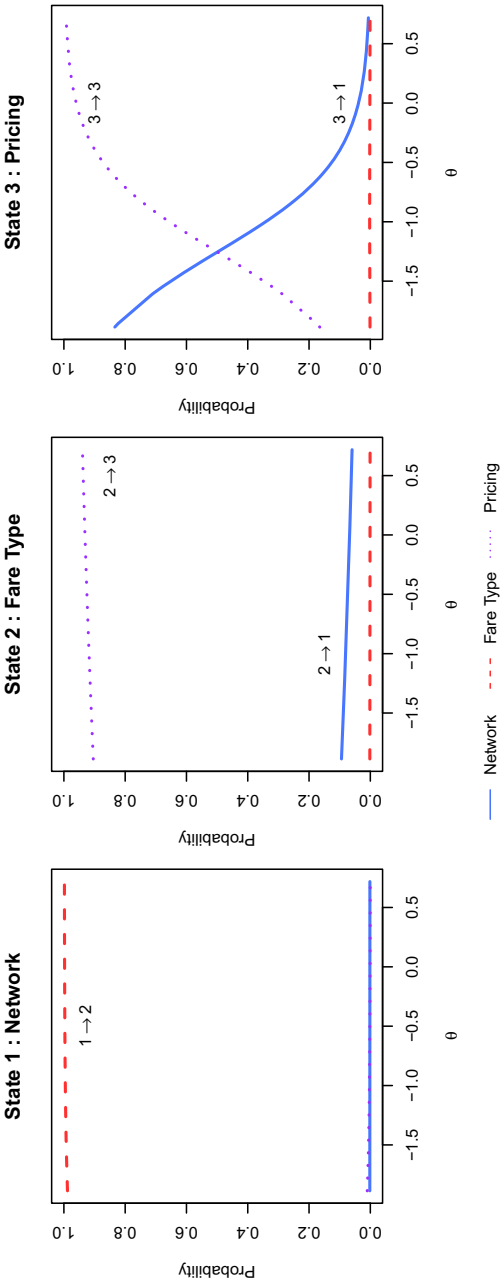
FIGURE 11.
State transition probability curves of the TICKET item.

the estimation performance changes with the two quantities. Two choices of n, 100 and 500, are considered to represent small and large sample size scenarios. Two choices of L, 10 and 50, are considered to represent short and long sequences scenarios. Fifty datasets are generated for each combination of n and L. To generate a sequence, we first generate θ_i from the standard normal distribution and the sequence length T_i from the Poisson distribution with mean L. Then, the hidden states and actions in the sequence are generated sequentially from the LHMM. Note that the sequences in a generated dataset vary in length and action composition.

For each dataset, we fit the proposed LHMM using the algorithms described in Sect. 3. Since the parameters in LHMM are identifiable up to a permutation of hidden states and sign change of parameters $\boldsymbol{A}$ and $\boldsymbol{C}$, the following post-estimation processing is conducted to match the estimated latent traits and parameters with the corresponding true values before evaluating the estimation performance. First, to align the estimated and the true latent traits, we multiply $\hat{\theta}$ by -1 if the correlation between $\hat{\theta}$ and θ is negative. Then, the estimated latent traits $\hat{\theta}$ are standardized to have mean zero and standard deviation one. The estimated parameters are also rescaled and shifted accordingly. Furthermore, for each possible permutation of the hidden states, we record the deviation of the estimated probability curves computed using the permuted parameters from the true curves. The deviation measure will be described shortly. The permutation producing the smallest deviation measurement is then chosen to obtain the final permuted parameters and hidden states for evaluation.

For each dataset, we measure the discrepancy between the estimated and true state transition probability curves using root-mean-squared error (RMSE):

$$\mathrm{RMSE}(\boldsymbol{P}) = \sqrt{\frac{1}{nK^2}\sum_{k=1}^{K}\sum_{l=1}^{K}\sum_{i=1}^{n}(\hat{p}_{kl}(\theta_i) - p_{kl}(\theta_i))^2},$$

where θ_i is the true latent trait of the i-th sequence in the dataset, $p_{kl}(\theta)$ is the true probability curve of transition from state k to state l, and $\hat{p}_{kl}(\theta)$ is the corresponding estimated curve. The RMSEs for evaluating the estimated state-action probability curves and the initial state probability distribution are defined analogously. We use the Pearson correlation between the estimated and true latent traits to evaluate the accuracy of latent trait estimation. The accuracy of hidden state estimation is evaluated by computing the proportion of the estimated hidden states that match the true ones.

For the purpose of comparing the proposed model and the ordinary HMM, we also fit the HMM with $K = 3$ to each of the datasets generated previously. In addition, we generate datasets from a HMM with $K = 3$ and fit both LHMM and HMM to each datasets. The model parameters are set as their counterparts in the LHMM with $\theta = 0$. Same as the settings for generating datasets from LHMM, we consider sample size $n = 100, 500$ and average sequence length $L = 10, 50$. Fifty datasets are generated for each combination of n and L. The two fitted models for a given dataset are compared using BIC.

5.2. Results

5.2.1. Comparison of LHMM and HMM　　Figure 12 presents the histograms of the difference between the BIC values of LHMM and HMM in different scenarios. The top and bottom rows correspond to results for datasets generated from LHMM and HMM, respectively. According to the figure, BIC can correctly choose between LHMM and HMM most of the time. Several mistakes are made in the case of $n = 100$ and $L = 10$ when the true model is LHMM and in the case of $n = 500$ and $L = 10$ when the true model is HMM. Overall, short sequences may bring difficulty in distinguishing between LHMM and HMM using BIC.
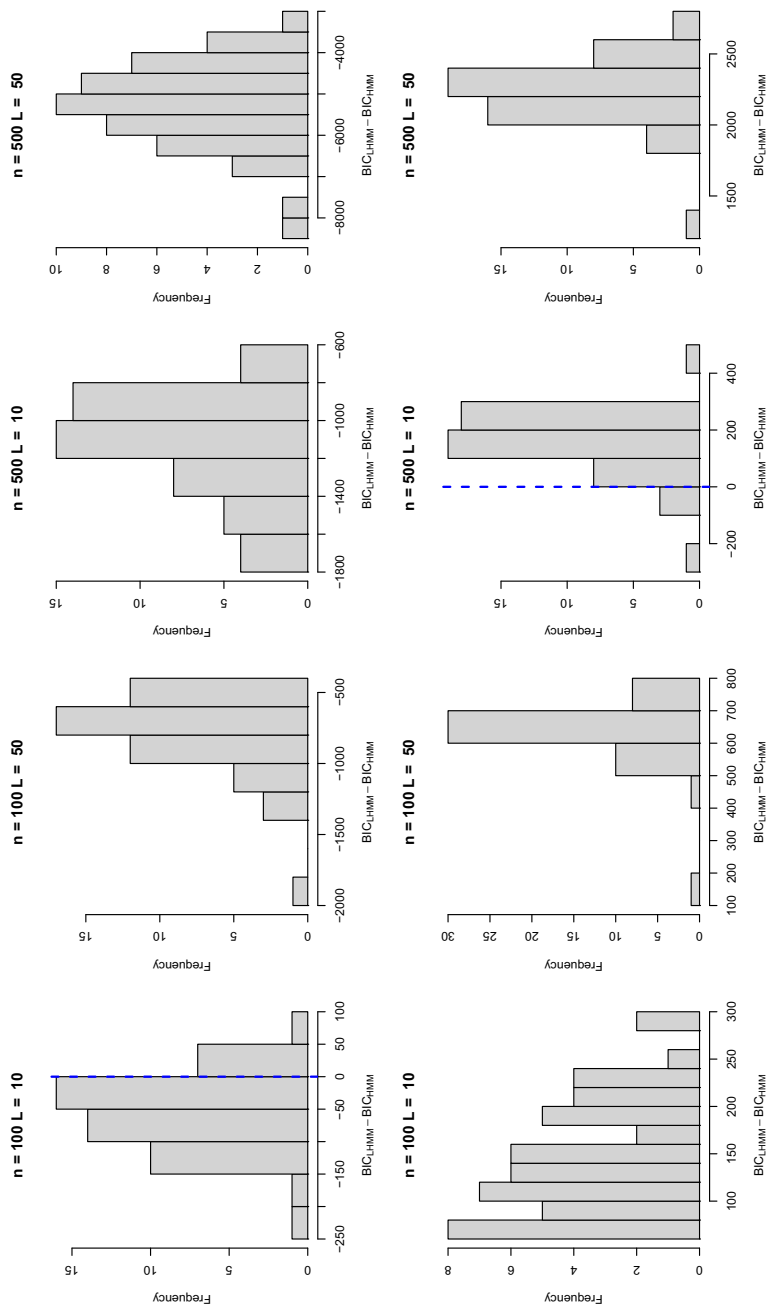
FIGURE 12.
Histograms of the difference between the BIC of LHMM and HMM for datasets generated from LHMM (top) and HMM (bottom).

5.2.2. Parameter Estimation   The RMSEs of the estimated probabilities are given in Fig. 13. In general, the performance of estimating the state transition and state-action probabilities improves as n or L increases. However, increasing sequence length does not reduce the RMSE of the estimated initial state probabilities since only the first action in each sequence provides information on π. In Fig. 13, the medians of the RMSE for each component are usually below 0.1, indicating reasonable overall estimation performance. We present the estimated probability curves from two randomly selected datasets in Fig. 14. One dataset is selected from the small-n-small-L scenario ($n = 100$, $L = 10$), and the other one is selected from the large-n-large-L scenario ($n = 500$, $L = 50$). It is clear that the probability curves are estimated more accurately in the latter scenario. Although the estimated curves in the small-n-small-L scenario can deviate from the true curves by a large amount, the overall trends often still resemble the true ones.

5.2.3. Latent Trait and Hidden States Estimation   Figure 15 presents the results of estimating the latent variables in LHMM. Overall, both latent traits and hidden states are estimated reasonably well. In all scenarios, the median Pearson correlations between estimated and true latent traits are above 0.85, and the median accuracies of estimated hidden states are above 0.8. However, large variation in estimation performance is seen in scenarios with short sequences. Note that both latent traits and hidden states are sequence-specific. While increasing sequence length provides more information for these latent variables, increasing sample size does not directly provide more information. As a result, in Fig. 15, we do not see significant improvement in the estimation performance when n is increased from 100 to 500.

6. Summary and Discussion

Process data contain rich information on respondents' problem-solving behaviors that is not available in traditional item responses. This paper proposes an LHMM for characterizing response processes and understanding the heterogeneity of problem-solving behaviors across respondents. Under the proposed model, a response process follows an HMM given the respondent's latent trait. The parameters in HMM are further parametrized using the latent trait to account for individual differences in solving problems. The structure of HMMs is analogous to problem-solving processes with the hidden states interpreted as problem-solving subtasks. The latent-trait-dependent state transition probability matrix P and state-action probability matrix Q describe how respondents differ in arranging and completing subtasks, respectively.

In LHMM, the latent trait is introduced as a quantity that summarizes the differences in respondents' behavior patterns in a parsimonious and abstract way. Examining the estimated latent trait $\hat{\theta}$ can help distinguish the behavior patterns of respondents in different groups. For example, in the CC item, after comparing $\hat{\theta}$ with respondents' age, one may observe that younger respondents have significantly smaller $\hat{\theta}$ and thus are less likely to reach the efficient VOTAT state. Identifying the differences can provide guidance on designing more targeted and even individualized interventions.

Although the proposed model does not impose concrete meaning on the latent trait, it is possible to find meaningful interpretations of the latent trait for at least some items. In the case studies, we interpret the latent trait as respondents' problem-solving skill by observing the close relationship between $\hat{\theta}$ and the binary item responses. In general, interpreting the latent trait requires knowledge about the item design. One can first infer the meaning of θ by examining how the state transition probability P and the state-action probability matrix Q change with θ. The conjecture can then be verified by comparing the latent traits with appropriate quantities (e.g., binary item response in our case).
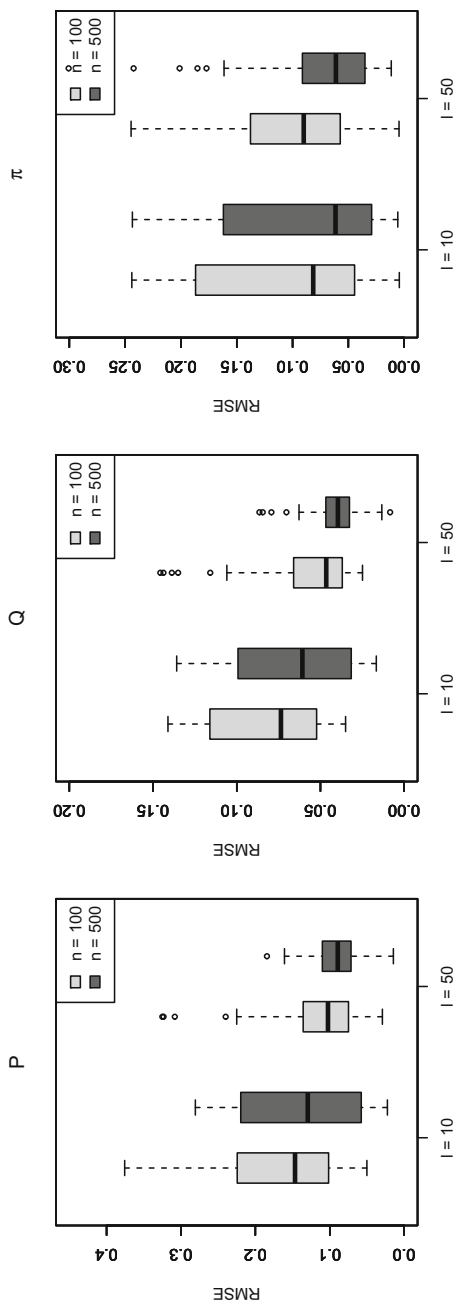
FIGURE 13.
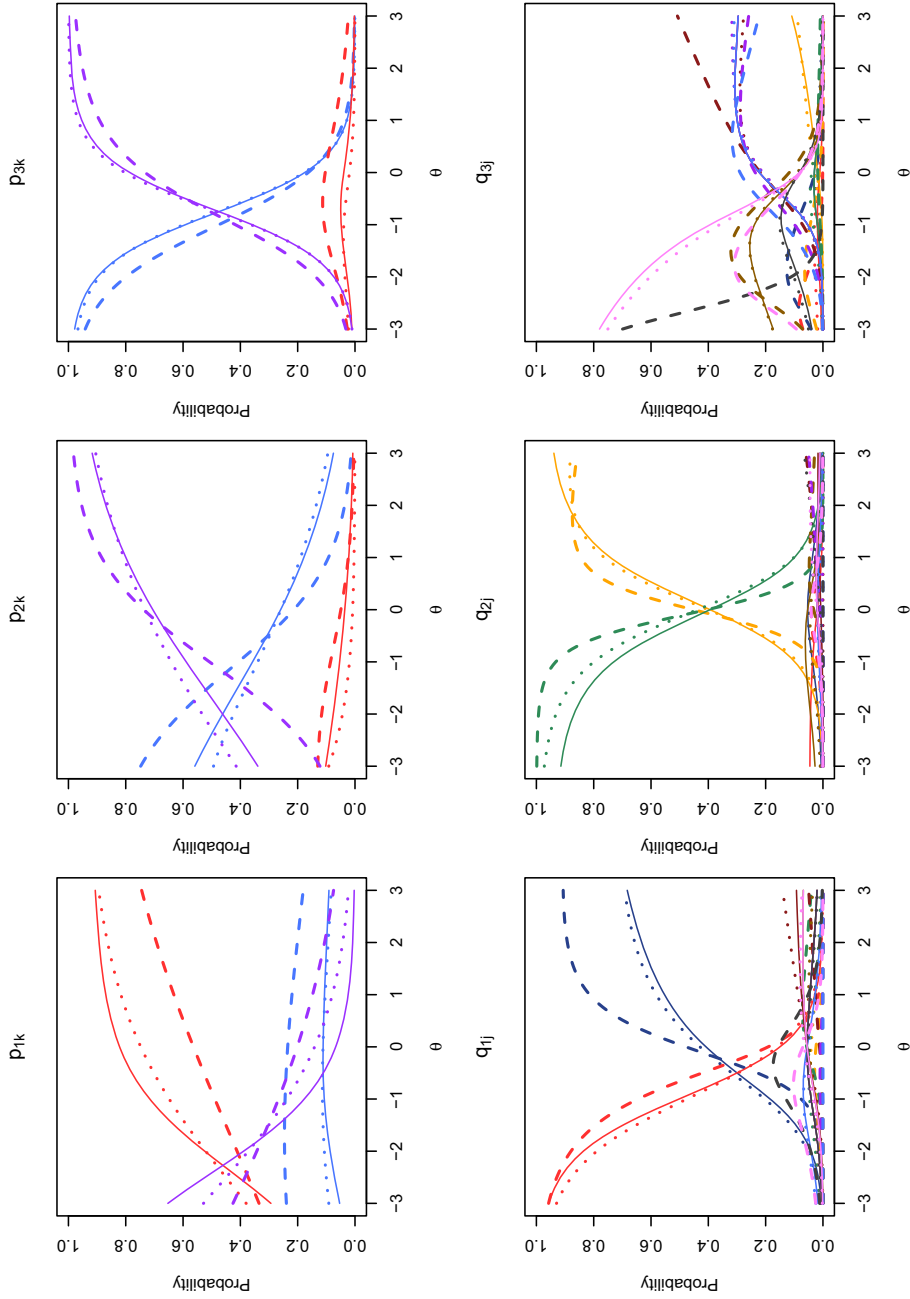Boxplots of the RMSEs of estimated probabilities.

FIGURE 14.
True (solid lines) and estimated state transition (top row) and state-action probability curves (bottom row) for scenarios $n = 100$, $L = 10$ (dashed lines) and $n = 500$, $L = 50$ (dotted lines). Different elements in a probability distribution are distinguished by colors.
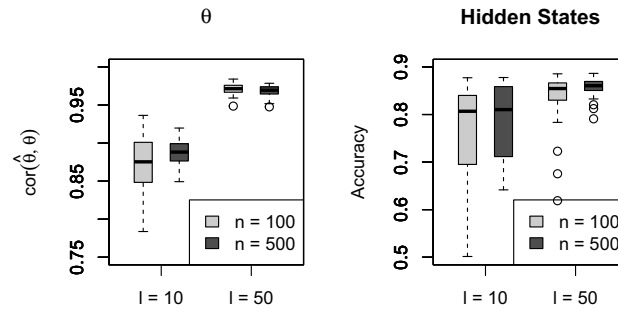
FIGURE 15.
Boxplots of evaluation measures of estimated latent variables.

One important feature of the proposed model is the use of HMM as the basic framework for modeling response processes. This feature allows us to connect latent traits to problem-solving subtasks. Latent variable models with simpler structures may not possess this feature although they can still describe the heterogeneity of problem-solving processes. For example, one could consider a set of n-grams (He & von Davier, 2016) of actions. Item response theory (IRT) models can be used to describe the binary matrix recording whether each n-gram appears in each response process or not. We considered a 2PL IRT approach for analyzing the TICKET and CC items. Unigrams and bigrams of actions were used to form the data matrix. The estimated latent traits have a moderate to strong correlation with those obtained from LHMM (0.87 for the TICKET item and 0.47 for the CC item), indicating that similar levels of heterogeneity are summarized in the two models. For the TICKET item, the results from the IRT model show that, as θ increases, the probability of using "country_train" increases and the probability of using "subway" decreases, but the results do not reveal that the two actions are the two options for determining the train network. Similarly, for the CC item, we are not able to tell from the IRT model results that students with a higher θ tend to reach the efficient VOTAT State faster.

Under the proposed model, the response processes of respondents with the same θ are assumed to evolve according to the same stochastic process. This assumption does not deprive the model's ability to describe the behavior diversity among these respondents. In fact, the diversity is charac- terized by the dispersion of the state transition and the state-action probability distributions. For example, if a state-action distribution is concentrated on a single action, different respondents are very likely to take the same action under the given hidden state. The actions taken by different respondents will be more diverse if more actions are assigned significant probability or if the probability is more evenly distributed among actions. According to the state-action distributions in Fig. 9, in State 1, respondents' behaviors vary more at $\theta = -0.74$ than those at $\theta = 0.71$ since three actions have non-trivial probabilities at the smaller θ, while a single action "county_train" takes almost all probability mass at the larger θ. Similar patterns are observed in other states. Quantitatively, Shannon entropy (Cover & Thomas, 2006) can be used to measure the dis- persion or the uncertainty of a distribution, with a higher value indicating more variation. The Shannon entropies of the state-action probability distributions in the three hidden states are 0.34, 0.38, and 0.23 at $\theta = -0.74$ and reduced to 0.02, 0.16, and 0.01 at $\theta = 0.71$. These results are consonant with existing empirical results showing that the behavior patterns of respondents with lower problem-solving proficiency have more variation (Eichmann et al., 2020; He et al., 2019; Ulitzsch et al., 2022a,b).

The proposed model and our data analyses have several limitations, suggesting directions for further investigation. In the case studies, we interpret the latent trait as problem-solving proficiency because of its close relationship with binary item responses. Ideally, the interpretation should be

further validated through carefully designed experiments or reliable measurement results. Such analyses will also help gauge the possibility of using LHMM as a measurement model. Also, we currently interpret the hidden states based on the patterns in state-action probability matrices and our understanding of the required tasks. More systematic and less subjective methods for interpreting hidden states would be beneficial for applying the proposed model in practice.

When analyzing the CC item, we grouped multiple recorded actions into one action. The simplification may cause information loss and obscure certain patterns in students' behaviors. Other ways of grouping the actions may lead to results different from those shown in our case study. Analyzing the response processes without simplifying the action set can avoid the problem but will be more computationally intensive.

Because of the complexity of human behaviors, it is almost certain that LHMM is not the underlying model generating real response processes. Nonetheless, the overly simplified model is helpful for understanding respondents' problem-solving behaviors as we showed in the case studies. It is interesting to see if more suitable models can be designed to capture more complicated patterns without sacrificing interpretability and computational efficiency too much.

Like many other studies on process data, we focus on analyzing individual items and ignore potential connections between them. Joint modeling of response processes from multiple items can be statistically more efficient than single-item analyses. It can also help us understand how the behavior patterns in different items are related. Although joint analysis using LHMM is conceptually straightforward, it could be computationally challenging due to the increased number of parameters. In addition, there are several model choices to consider, such as whether respondents' behaviors in different items should be affected by the same latent trait. As it stands, the proposed LHMM is not flexible enough to accommodate such needs. As we discussed below, several extensions can be considered to increase the flexibility of the model and better support joint modeling of response processes from different items.

The proposed model involves a single latent trait in the initial state probability vector π, state transition probability matrix P, and state-action probability Q. In complex problem-solving items, it is very likely that multidimensional latent traits should be used for characterizing the heterogeneity of response processes. It is also possible that P and Q are affected by different latent traits. LHMMs involving multidimensional latent traits are straightforward to formulate. However, the MLE of the model parameter may be difficult to obtain since numerical integration in higher dimensions may be computationally expensive and inaccurate. Developing more computationally efficient methods for statistical inference is essential for such models. Jointly estimating model parameters and latent traits may be a direction to take as this method has demonstrated computational advantages in multidimensional item response theory models Chen et al. (2019b).

Currently, no constraint is imposed on the state transitions in our model. Transitions could occur between any two states. In practice, constraints exist on state transitions. For example, in the TICKET item, the design of the item interface determines that no transition should occur from the Network state to the Pricing state or from the Pricing state to the Fare Type state. In the CC item, it is natural to assume that students will not go back to an inefficient problem-solving state once they figure out the efficient strategy, which prohibits the transition from the VOTAT State to the EXPLORE State. The fitted LHMM allows such transitions, although only with tiny probabilities. Imposing constraints on state transitions reduces the dimension of parameter space and thus improves the stability of the model fit. Developing data-driven methods to detect such structure in the state transition probability matrix is an interesting future direction.

As a final remark, we would like to point out that the proposed LHMM may not be suitable for response processes from all problem-solving items despite its good performance in characterizing the response structure of the TICKET item and the CC item. Also, if the required task does not have a multi-subtask pattern, one can still fit an LHMM to the response processes, but the hidden states may not have a clear interpretation.

Appendix A LHMM Likelihood Computation

The likelihood for a set of response processes $\mathcal{Y}_n$ following an LHMM is

$$L(\boldsymbol{\eta} \mid \mathcal{Y}_n) = \prod_{i=1}^{n} P(\boldsymbol{Y}^{(i)} = \boldsymbol{y}^{(i)} \mid \boldsymbol{\eta}) = \prod_{i=1}^{n} \left\{ \int \phi(\theta_i) P(\boldsymbol{Y}^{(i)} = \boldsymbol{y}^{(i)} \mid \boldsymbol{\eta}, \theta_i) d\theta_i \right\}.$$

We demonstrate here how to compute $L_i\left(\boldsymbol{\eta} \mid \boldsymbol{y}^{(i)}\right) = P\left(\boldsymbol{Y}^{(i)} = \boldsymbol{y}^{(i)} \mid \boldsymbol{\eta}\right)$. For notation simplicity, the superscripts and the subscripts denoting different respondents are suppressed hereafter. We explain first how to compute $f(\boldsymbol{\eta}, \theta) = P(\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{\eta}, \theta)$ given $(\boldsymbol{\eta}, \theta)$ and then how to numerically integrate $\phi(\theta) f(\boldsymbol{\eta}, \theta)$ with respect to θ to obtain $L(\boldsymbol{\eta} \mid \boldsymbol{y})$.

For $k = 1, \ldots, K$ and $t = 1, \ldots, T$, define the forward probability

$$\alpha_t(k \mid \theta) = P(\boldsymbol{Y}_{1:t} = \boldsymbol{y}_{1:t}, S_t = k \mid \boldsymbol{\eta}, \theta). \tag{A1}$$

Given $\boldsymbol{\eta}$ and θ, we can obtain $f(\boldsymbol{\eta}, \theta)$ from the forward probabilities $\alpha_T(k \mid \theta)$ since $f(\boldsymbol{\eta}, \theta) = \sum_{k=1}^{K} \alpha_T(k \mid \theta)$. According to HMM assumptions (1–4), it is easy to verify $\alpha_1(k \mid \theta) = \pi_k(\theta) q_{k, y_1}(\theta)$ and

$$\alpha_t(k \mid \theta) = \sum_{l=1}^{K} \alpha_{t-1}(l \mid \theta) p_{lk}(\theta) q_{k, y_t}(\theta), \ t = 2, \ldots, T, \tag{A2}$$

where $\pi_k(\theta)$, $p_{kl}(\theta)$, and $q_{kj}(\theta)$ are defined in (5–7). Therefore, $\alpha_T(k \mid \theta)$ can be computed by first calculating $\alpha_1(k \mid \theta)$ and then applying (A2) recursively.

Besides the forward probabilities, one can also define the backward probability

$$\beta_t(k \mid \theta) = P(\boldsymbol{Y}_{(t+1):T} = \boldsymbol{y}_{(t+1):T} \mid S_t = k, \boldsymbol{\eta}, \theta), \ k = 1, \ldots, K, \ t = 1, \ldots, T-1. \tag{A3}$$

Letting $\beta_T(k \mid \theta) = 1$, then we have the recursive relation

$$\beta_t(k \mid \theta) = \sum_{l=1}^{K} p_{kl}(\theta) q_{l, y_{t+1}}(\theta) \beta_{t+1}(l \mid \theta). \tag{A4}$$

Although computing $f(\boldsymbol{\eta}, \theta)$ does not require the backward probabilities, we still compute them when evaluating the likelihood because they, together with the forward probabilities, are essential components for computing the derivatives of the likelihood function. See Appendix B for details.

Given that $f(\boldsymbol{\eta}, \theta)$ is computable, we can approximate

$$L(\boldsymbol{\eta} \mid \boldsymbol{y}) = \int \phi(\theta) f(\boldsymbol{\eta}, \theta) d\theta = \frac{1}{\sqrt{\pi}} \int e^{-x^2} f(\boldsymbol{\eta}, \sqrt{2x}) dx$$

using Gaussian–Hermite quadrature by $\frac{1}{\sqrt{\pi}} \sum_{u=1}^{U} w_u f(\boldsymbol{\eta}, \sqrt{2} x_u)$ where $x_1, \ldots, x_U$ are U quadrature points and $w_1, \ldots, w_U$ are the associated weights. The quadrature points and the corresponding weights for a given U can be computed based on the Hermite polynomials. We use the function gauss.quad in the R package statmod for this aim.

The algorithm for computing the likelihood function for LHMM is summarized in Algorithm 1.

Algorithm 1. (LHMM likelihood computation) The likelihood function $L(\boldsymbol{\eta} \mid \boldsymbol{y})$ for a response process $\boldsymbol{y}$ following LHMM is computed in the following steps.

1. Obtain Gaussian–Hermite quadrature points $x_1, \ldots, x_U$ and the associated weights $w_1, \ldots, w_U$.
2. For $u = 1, \ldots, U$, compute $f(\boldsymbol{\eta}, \sqrt{2}x_u)$ as follows.
   (a) Compute $\alpha_1(k \mid \sqrt{2}x_u) = \pi_k(\sqrt{2}x_u)q_{k,y_1}(\sqrt{2}x_u)$ and set $\beta_T(k \mid \sqrt{2}x_u) = 1$ for $k = 1, \ldots, K$.
   (b) For $t = 2, \ldots, T$ and $k = 1, \ldots, K$, compute

$$\alpha_t(k \mid \sqrt{2}x_u) = \sum_{l=1}^{K} \alpha_{t-1}(l \mid \sqrt{2}x_u) p_{lk}(\sqrt{2}x_u) q_{k,y_t}(\sqrt{2}x_u)$$

   and

$$\beta_{T-t+1}(k \mid \sqrt{2}x_u) = \sum_{l=1}^{K} p_{kl}(\sqrt{2}x_u) q_{l,y_{T-t+2}}(\sqrt{2}x_u) \beta_{T-t+2}(l \mid \sqrt{2}x_u).$$

   (c) Compute $f(\boldsymbol{\eta}, \sqrt{2}x_u) = \sum_{k=1}^{K} \alpha_T(k \mid \sqrt{2}x_u)$.
3. Compute $L(\boldsymbol{\eta} \mid \boldsymbol{y}) = \frac{1}{\sqrt{\pi}} \sum_{u=1}^{U} w_u f(\boldsymbol{\eta}, \sqrt{2}x_u)$.

Appendix B Gradient of LHMM Log-Likelihood Function

For a given element η in $\boldsymbol{\eta}$,

$$\frac{\partial \log L(\boldsymbol{\eta})}{\partial \eta} = \sum_{i=1}^{n} \frac{1}{L_i(\boldsymbol{\eta} \mid \boldsymbol{y}^{(i)})} \frac{\partial L_i(\boldsymbol{\eta} \mid \boldsymbol{y}^{(i)})}{\partial \eta}.$$

The algorithm for calculating $L_i(\boldsymbol{\eta} \mid \boldsymbol{y}^{(i)})$ is presented in Appendix A. We explain here how to compute $\frac{\partial L_i(\boldsymbol{\eta} \mid \boldsymbol{y}^{(i)})}{\partial \eta}$. The superscripts and the subscripts denoting different respondents are suppressed hereafter to simplify notation. Let $f(\boldsymbol{\eta}, \theta) = P(\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{\eta}, \theta)$. Then

$$\frac{\partial L(\boldsymbol{\eta} \mid \boldsymbol{y})}{\partial \eta} = \int \phi(\theta) \frac{\partial f(\boldsymbol{\eta}, \theta)}{\partial \eta} d\theta. \tag{A5}$$

If $\frac{\partial f(\boldsymbol{\eta}, \theta)}{\partial \eta}$ is computable given $(\boldsymbol{\eta}, \theta)$, then the integral on the right-hand side of (A5) can be approximated using Gaussian–Hermite quadrature similarly as in computing the likelihood function. In the remaining part, we focus on deriving $\frac{\partial f(\boldsymbol{\eta}, \theta)}{\partial \eta}$. In the following calculations, the initial state probability π_k, the state transition probabilities p_{kl}, and the state-action probabilities q_{kj} all depend on θ as defined in (5–7). To simplify notation, we do not explicitly write them as functions of θ.

First, consider taking derivative of f with respect to π_k, p_{kl}, and q_{kj}. Define $\boldsymbol{\alpha}_t = (\alpha_t(1), \ldots, \alpha_t(K))^\top$ and $\boldsymbol{\beta}_t = (\beta_t(1), \ldots, \beta_t(K))^\top$ where $\alpha_t(k)$ and $\beta_t(k)$ are the forward

and backward probabilities defined in (A1) and (A3), respectively. Then, the relationship in (A2) and (A4) can be expressed compactly as

$$\boldsymbol{\alpha}_t = \boldsymbol{\alpha}_{t-1}^\top \boldsymbol{P} \tilde{\boldsymbol{Q}}_t, \text{ and } \boldsymbol{\beta}_t = \boldsymbol{P} \tilde{\boldsymbol{Q}}_{t+1} \boldsymbol{\beta}_{t+1},$$

where $\boldsymbol{P}$ is the state transition probability matrix and $\tilde{\boldsymbol{Q}}_t = \mathrm{diag}\{q_{1,y_t}, \ldots, q_{K,y_t}\}$. Recursively applying the above relationship, we get

$$\boldsymbol{\alpha}_t = \boldsymbol{\pi}^\top \tilde{\boldsymbol{Q}}_1 \boldsymbol{P} \tilde{\boldsymbol{Q}}_2 \cdots \boldsymbol{P} \tilde{\boldsymbol{Q}}_t \text{ and } \boldsymbol{\beta}_t = \boldsymbol{P} \tilde{\boldsymbol{Q}}_{t+1} \cdots \boldsymbol{P} \tilde{\boldsymbol{Q}}_T \mathbf{1},$$

where $\mathbf{1}$ is a column vector of K ones. Let x denote a generic element of $\boldsymbol{\pi}$, $\boldsymbol{P}$ or $\boldsymbol{Q}$. Then,

$$\frac{\partial f}{\partial x} = \frac{\partial \boldsymbol{\pi}^\top \tilde{\boldsymbol{Q}}_1}{\partial x} \boldsymbol{P} \tilde{\boldsymbol{Q}}_2 \cdots \boldsymbol{P} \tilde{\boldsymbol{Q}}_T \mathbf{1} + \sum_{t=1}^{T-1} \boldsymbol{\alpha}_t^\top \frac{\partial \boldsymbol{P} \tilde{\boldsymbol{Q}}_{t+1}}{\partial x} \boldsymbol{\beta}_{t+1}.$$

Replacing x with π_k, p_{kl}, and q_{kj} and simplifying the expression, we obtain

$$\begin{aligned}
\frac{\partial f}{\partial \pi_k} &= q_{k,y_1} \beta_1(k), \ k = 1, \ldots, K \\
\frac{\partial f}{\partial p_{kl}} &= \sum_{t=1}^{T-1} \alpha_t(k) \beta_{t+1}(l) q_{l,y_{t+1}}, \ k, l = 1, \ldots, K \\
\frac{\partial f}{\partial q_{kj}} &= \sum_{t:y_t=j} \alpha_t(k) \beta_t(k) / q_{kj}, \ k = 1, \ldots, K, \ j = 1, \ldots, M.
\end{aligned} \tag{A6}$$

According to the chain rule,

$$\begin{aligned}
\frac{\partial f}{\partial \mu_k} &= \sum_{k'=1}^{K} \frac{\partial f}{\partial \pi_{k'}} \frac{\partial \pi_{k'}}{\partial \mu_k} = \pi_k \left(\frac{\partial f}{\partial \pi_k} - \sum_{k'=1}^{K} \frac{\partial f}{\partial \pi_{k'}} \pi_{k'} \right), \ \frac{\partial f}{\partial \tau_k} = \sum_{k'=1}^{K} \frac{\partial f}{\partial \pi_{k'}} \frac{\partial \pi_{k'}}{\partial \tau_k} = \theta \frac{\partial f}{\partial \mu_k}, \\
\frac{\partial f}{\partial b_{kl}} &= \sum_{l'=1}^{K} \frac{\partial f}{\partial p_{kl'}} \frac{\partial p_{kl'}}{\partial b_{kl}} = p_{kl} \left(\frac{\partial f}{\partial p_{kl}} - \sum_{l'=1}^{K} \frac{\partial f}{\partial p_{kl'}} p_{kl'} \right), \ \frac{\partial f}{\partial a_{kl}} = \sum_{l'=1}^{K} \frac{\partial f}{\partial p_{kl'}} \frac{\partial p_{kl'}}{\partial a_{kl}} = \theta \frac{\partial f}{\partial b_{kl}}, \\
\frac{\partial f}{\partial d_{kj}} &= \sum_{j'=1}^{M} \frac{\partial f}{\partial q_{kj'}} \frac{\partial q_{kj'}}{\partial d_{kj}} = q_{kj} \left(\frac{\partial f}{\partial q_{kj}} - \sum_{j'=1}^{K} \frac{\partial f}{\partial q_{kj'}} q_{kj'} \right), \ \frac{\partial f}{\partial c_{kj}} = \sum_{j'=1}^{M} \frac{\partial f}{\partial q_{kj'}} \frac{\partial q_{kj'}}{\partial c_{kj}} = \theta \frac{\partial f}{\partial d_{kj}}.
\end{aligned} \tag{A7}$$

Combining (A6) and (A7) gives $\frac{\partial f}{\partial \eta}$ for $\eta = \tau_k, \mu_k, a_{kl}, b_{kl}, c_{kj}, d_{kj}$.

Appendix C Viterbi Algorithm

Let $\boldsymbol{y}$ be a sequence following the LHMM with parameters $\boldsymbol{\eta}$ and latent trait θ. The most probable hidden state sequence $\hat{\boldsymbol{s}}$ can be found using the Viterbi algorithm. For $k = 1, \ldots, K$ and $t = 2, \ldots, T$, define

$$v_t(k) = \max_{s_{1:(t-1)}} P(\mathbf{Y}_{1:t} = \mathbf{y}_{1:t}, \mathbf{S}_{1:(t-1)} = s_{1:(t-1)}, S_t = k \mid \theta, \boldsymbol{\eta}).$$

According to HMM assumptions (1)–(4), we have the recursive relation

$$v_t(k) = \max_{l=1,\dots,K} v_{t-1}(l) p_{lk}(\theta) q_{k,y_t}(\theta),$$

where $v_1(k) = \pi_k(\theta) q_{k,y_1}(\theta)$. Let

$$u_t(k) = \operatorname*{argmax}_{l=1,\dots,K} v_{t-1}(l) p_{lk}(\theta) q_{k,y_t}(\theta).$$

After computing $v_t(k)$ and $u_t(k)$ for $k = 1, \dots, K$ and $t = 2, \dots, T$ sequentially, the most probable hidden state sequence can be obtained by backtracing:

$$\hat{s}_T = \operatorname*{argmax}_{k=1,\dots,K} v_T(k), \quad \hat{s}_t = \operatorname*{argmax}_{k=1,\dots,K} u_{t+1}(k), \quad \text{for } t = T-1, \dots, 1. \tag{A8}$$

The algorithm is summarized in Algorithm 2.

Algorithm 2. (Viterbi Algorithm) The most probable hidden state sequence $\hat{s}$ for a response process $\mathbf{y}$ following the LHMM with latent trait θ is obtained in the following steps.

1. For $k = 1, \dots, K$, compute $v_1(k) = \pi_k(\theta) q_{k,y_1}(\theta)$.
2. For $t = 2, \dots, T$,

   (a) Compute $w_t(l, k) = v_{t-1}(l) p_{lk}(\theta) q_{k,y_t}(\theta)$ for $k, l = 1, \dots, K$;
   (b) Record $v_t(k) = \max_l w_t(l, k)$ and $u_t(k) = \operatorname*{argmax}_l w_t(l, k)$ for $k = 1, \dots, K$.

3. Obtain $\hat{s}$ by backtracing:

   (a) $\hat{s}_T = \operatorname{argmax}_k v_T(k)$;
   (b) For $t = T - 1, \dots, 1$, set $\hat{s}_t = \operatorname{argmax}_k u_{t+1}(k)$.

Appendix D Estimated LHMM Parameters in Case Studies

Tables 3 and 4 present the LHMM parameter estimates for the CC item and the TICKET item, respectively.

TABLE 3.
Estimated LHMM parameters for the CC item.

| | A | | | B | | π |
	State 1	State 2		State 1	State 2	
State 1	0.00	0.00		0.00	−2.59	0.13
State 2	0.00	−0.15		0.00	2.05	0.87

| | C | | | | | | | | |
	All	T_M	T_B	M_B	Top	Middle	Bottom	None	RESET
State 1	0.00	0.23	−0.91	0.60	−1.05	4.12	4.02	0.97	3.58
State 2	0.00	1.87	2.63	3.79	2.49	0.47	−0.71	4.75	2.43

| | D | | | | | | | | |
	All	T_M	T_B	M_B	Top	Middle	Bottom	None	RESET
State 1	0.00	−5.80	0.97	2.75	−2.13	3.48	3.57	2.90	3.47
State 2	0.00	−1.34	−2.88	−6.26	−0.68	−4.66	−5.74	−6.11	−1.36

For ease of comparison, a column of zeros is prepended to A, B, C, and D. In C and D, T_M, T_B, and M_B stand for actions "Top_Middle," "Top_Bottom," and "Middle_Bottom," respectively.

TABLE 4.
Estimated LHMM parameters for the TICKET item.

| | A | | | B | | | τ | μ | π |
	State 1	State 2	State 3	State 1	State 2	State 3			
State 1	0.00	0.07	−1.28	0.00	6.52	−0.63	0.00	0.00	0.990
State 2	0.00	−0.06	0.19	0.00	−4.17	2.62	0.00	−7.33	0.001
State 3	0.00	1.82	2.56	0.00	−3.23	3.21	0.00	−4.75	0.009

| | C | | | | | | | | | | | |
	trains	subway	full	conc.	indv.	daily	trip_1	trip_2	trip_3	trip_4	trip_5	Buy	Cancel
State 1	0.00	−3.52	−0.34	−0.77	−0.04	−0.21	−0.77	−1.92	−0.58	−0.28	−0.12	−4.60	−0.04
State 2	0.00	−0.85	0.76	−0.72	0.95	−0.61	0.43	0.07	0.41	0.32	0.77	−0.98	−0.37
State 3	0.00	−0.90	1.81	−0.04	1.08	−3.84	0.88	1.06	−0.98	−1.54	−1.48	1.30	−1.43

| | D | | | | | | | | | | | |
	trains	subway	full	conc.	indv.	daily	trip_1	trip_2	trip_3	trip_4	trip_5	Buy	Cancel
State 1	0.00	−3.07	−6.24	−5.93	−6.58	−6.35	−6.95	−7.49	−6.91	−7.27	−7.04	−5.24	−5.57
State 2	0.00	−3.00	8.12	6.24	−4.27	0.11	−2.29	−2.77	−2.89	−2.92	−3.23	−0.52	5.69
State 3	0.00	−1.92	−0.92	0.03	5.85	1.35	3.74	5.76	2.66	1.58	2.06	5.63	3.94

For ease of comparison, a column of zeros is prepended to A, B, C, and D. In C and D, trains, subway, full, conc., and indv. stand for actions "country_trains," "city_subway," "full_fare," "concession," and "individual," respectively.

Appendix E True Parameters in Simulation Studies

Table 5 presents the parameters of LHMM for generating the action sequences in the simulation study. The values are chosen so that the resulting state transition and state-action probability curves are similar to those obtained in the TICKET item.

TABLE 5.
Parameters used for generating action sequences in the simulation study.

	τ	μ	A $l=2$	$l=3$		B $l=2$	$l=3$
$k=1$			0.1	-1.0		2.0	-0.5
$k=2$	0.0	-2.0	-0.1	0.5		-2.0	1.0
$k=3$	0.0	-1.0	1.0	2.0		-1.5	1.5

	C								
	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$	$j=9$	$j=10$
$k=1$	-2.0	-0.5	-0.5	0.0	-0.1	-0.5	-1.0	-0.5	-0.1
$k=2$	-1.0	1.0	-1.0	1.0	-0.5	0.5	0.0	0.5	0.5
$k=3$	-1.0	1.5	0.0	1.0	-1.5	1.0	1.0	-1.0	-2.0

	D								
	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$	$j=7$	$j=8$	$j=9$	$j=10$
$k=1$	-1.0	-2.0	-2.0	-2.0	-2.0	-2.0	-2.0	-2.0	-2.0
$k=2$	-1.0	2.0	2.0	-2.0	0.0	-1.0	-1.0	-1.0	-1.0
$k=3$	-1.0	-0.5	0.0	2.0	1.0	2.0	2.0	1.0	1.0

Data availability

The dataset analyzed in the current study are available at https://www.oecd.org/pisa/pisaproducts/database-cbapisa2012.htm

Declarations

Conflict of interest The author has no conflicts of interest to declare that are relevant to the content of this article.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

Binkley, M., Erstad, O., Herman, J., Raizen, S., Ripley, M., Miller-Ricci, M., & Rumble, M. (2012). Defining twenty-first century skills. In *Assessment and teaching of 21st century skills* (pp. 17–66). Springer.

Broyden, C. G. (1970). The convergence of a class of double-rank minimization algorithms 1. General considerations. *IMA Journal of Applied Mathematics, 6*(1), 76–90.

Cappé, O., Moulines, E., & Ryden, T. (2005). *Inference in hidden Markov models*. Springer. https://doi.org/10.1007/0-387-28982-8

Chen, Y. (2020). A continuous-time dynamic choice measurement model for problem-solving process data. *Psychometrika, 85*(4), 1052–1075.

Chen, Y., Li, X., Liu, J., & Ying, Z. (2019a). Statistical analysis of complex problem-solving process data: An event history analysis approach. *Frontiers in Psychology, 10*, 486.

Chen, Y., Li, X., & Zhang, S. (2019b). Joint maximum likelihood estimation for high-dimensional exploratory item factor analysis. *Psychometrika, 84*(1), 124–146.

Cover, T. M., & Thomas, J. A. (2006). *Elements of information theory* (2nd ed.). Wiley.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society Series B, 39*(1), 1–22.

Eddelbuettel, D., & François, R. (2011). Rcpp: Seamless r and c++ integration. *Journal of Statistical Software, 40*, 1–18.

Eichmann, B., Greiff, S., Naumann, J., Brandhuber, L., & Goldhammer, F. (2020). Exploring behavioural patterns during complex problem-solving. *Journal of Computer Assisted Learning, 36*(6), 933–956.

Fletcher, R. (1970). A new approach to variable metric algorithms. *The Computer Journal, 13*(3), 317–322.

Giner, G., Chen, L., Hu, Y., Dunn, P., Phipson, B., & Chen, Y. (2023). statmod: Statistical modeling [Computer software manual]. Retrieved from https://cran.r-project.org/package=statmod

Goldfarb, D. (1970). A family of variable-metric methods derived by variational means. *Mathematics of Computation, 24*(109), 23–26.

Greiff, S., Niepel, C., Scherer, R., & Martin, R. (2016). Understanding students' performance in a computer-based assessment of complex problem solving: An analysis of behavioral data from computer-generated log files. *Computers in Human Behavior, 61*, 36–46. https://doi.org/10.1016/j.chb.2016.02.095

Greiff, S., Wüstenberg, S., & Avvisati, F. (2015). Computer-generated log-file analyses as a window into students' minds? A showcase study based on the PISA 2012 assessment of problem solving. *Computers & Education, 91*, 92–105.

Han, Y., Liu, H., & Ji, F. (2021). A sequential response model for analyzing process data on technology-based problem-solving tasks. *Multivariate Behavioral Research, 57*, 960.

He, Q., & von Davier, M. (2016). Analyzing process data from problem-solving items with n-grams: Insights from a computer-based large-scale assessment. In Y. Rosen, S. Ferrara, & M. Mosharraf (Eds.), Handbook of research on technology tools for real-world skill development (pp. 749-776). Information Science Reference. https://doi.org/10.4018/978-1-4666-9441-5.ch029

He, Q., Liao, D., & Jiao, H. (2019). Clustering behavioral patterns using process data in PIAAC problem-solving items. In *Theoretical and practical advances in computer-based educational measurement* (pp. 189-212). Springer.

Herborn, K., Mustafić, M., & Greiff, S. (2017). Mapping an experiment-based assessment of collaborative behavior onto collaborative problem solving in PISA 2015: A cluster analysis approach for collaborator profiles. *Journal of Educational Measurement, 54*(1), 103–122.

Liang, K., Tu, D., & Cai, Y. (2022). Using process data to improve classification accuracy of cognitive diagnosis model. *Multivariate Behavioral Research*. https://doi.org/10.1080/00273171.2022.2157788

Lord, F. M. (1980). *Applications of item response theory to practical testing problems*. Routledge.

McCullagh, P., & Nelder, J. (2018). *Generalized linear models*. Routledge.

OECD. (2014). *PISA 2012 results: Creative problem solving: Students' skills in tackling real-life problems* (Vol. 5). OECD Publishing. https://doi.org/10.1787/9789264208070-en

R Core Team. (2023). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from https://www.R-project.org/

Rabiner, L., & Juang, B. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine, 3*(1), 4–16.

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

Shanno, D. F. (1970). Conditioning of quasi-Newton methods for function minimization. *Mathematics of Computation, 24*(111), 647–656.

Stadler, M., Fischer, F., & Greiff, S. (2019). Taking a closer look: An exploratory analysis of successful and unsuccessful strategy use in complex problems. *Frontiers in Psychology, 10*, 777.

Tang, X., Wang, Z., He, Q., Liu, J., & Ying, Z. (2020). Automatic feature construction for process data using multidimensional scaling. *Psychometrika, 85*, 378–397.

Tang, X., Wang, Z., Liu, J., & Ying, Z. (2021). An exploration of process data by action sequence autoencoder. *British Journal of Mathematical and Statistical Psychology, 74*, 1–33.

Ulitzsch, E., He, Q., & Pohl, S. (2022a). Using sequence mining techniques for understanding incorrect behavioral patterns on interactive tasks. *Journal of Educational and Behavioral Statistics, 47*(1), 3–35.

Ulitzsch, E., Ulitzsch, V., He, Q., & Lüdtke, O. (2022b). A machine learning-based procedure for leveraging clickstream data to investigate early predictability of failure on interactive tasks. *Behavior Research Methods, 55*, 1392.

Viterbi, A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE transactions on Information Theory, 13*(2), 260–269.

von Davier, M., Khorramdel, L., He, Q., Shin, H. J., & Chen, H. (2019). Developments in psychometric population models for technology-based large-scale assessments: An overview of challenges and opportunities. *Journal of Educational and Behavioral Statistics, 44*(6), 671–705.

Wang, Z., Tang, X., Liu, J., & Ying, Z. (2022). Subtask analysis of process data through a predictive model. *British Journal of Mathematical and Statistical Psychology*. https://doi.org/10.1111/bmsp.12290

Xiao, Y., He, Q., Veldkamp, B., & Liu, H. (2021). Exploring latent states of problem-solving competence using hidden Markov model on process data. *Journal of Computer Assisted Learning, 37*(5), 1232–1247.

Xu, H., Fang, G., & Ying, Z. (2020). A latent topic model with Markov transition for process data. *British Journal of Mathematical and Statistical Psychology, 73*(3), 474–505.

Zhang, S., Wang, Z., Qi, J., Liu, J., & Ying, Z. (2023). *Accurate assessment via process data. Psychometric, 88*, 76–97. https://doi.org/10.1007/s11336-022-09880-8

Zhan, P., & Qiao, X. (2022). Diagnostic classification analysis of problem-solving competence using process data: An item expansion method. *Psychometrika, 87*, 1529.