Abstract

The development of human-computer interactive items in educational assessments provides opportunities to extract useful process information for problem-solving (Tang et al., 2020). However, the complex, intensive, and noisy nature of process data makes it challenging to model with the traditional psychometric methods (Zhang et al., 2021). Zhu et al. (2016) introduced social network methods to visualize and analyze process data. Nonetheless, research about statistical modeling of process information using social network methods is still limited. This article explored the application of the latent space model (LSM) for analyzing process data in educational assessment. The adjacent matrix of transitions between actions was created based on the weighted and directed network of action sequences and related auxiliary information. Then, the adjacent matrix was modeled with LSM to identify the lower-dimensional latent positions of actions. Three applications based on the results from LSM were introduced: action clustering, error analysis, and performance measurement. The simulation study showed that LSM can cluster actions from the same problem-solving strategy and measure students' performance by comparing their action sequences with the optimal strategy. Finally, we analyzed the empirical data from PISA 2012 as a real case scenario to illustrate how to use LSM.

Keywords: Latent Space Model, Social Network Analysis, Process Data

Latent Space Model for Process Data

Introduction

The innovations of technology-enhanced items in educational assessment, such as simulation-, scenario-, and game-based assessments, collect substantially more information than item responses (Shu et al., 2017; OECD, 2016). For example, the program for international student assessment (PISA) launched a computer-based problem-solving assessment in 2012 and added a human-computer interactive collaborative problem-solving assessment in 2015. The rich process information gathered from these new technology-based tests, such as keystrokes and click flow, provides new opportunities for revealing response patterns, exploring the cognitive structure of problem-solving, and detecting unintended behaviors, and improving measurement precision (Tang et al., 2020; Zhang et al., 2021). Along with these benefits, the significant increased volume, velocity, and variety of process data pose new challenges to materialize the value of these rich data (von Davier et al., 2021). Studies on the log of process data in educational assessment remain relatively scarce (Greiff et al., 2016; He et al., 2019).

Different from cross-sectional item responses, process data is usually saved as action sequences for all tracked actions in high-dimensional and unstandardized formats along with auxiliary information (e.g., timestamp). Mislevy (2019) proposed two basic analytical processes for interpreting and modeling process data. The first process is to extract useful information from the complex and diverse process data in a theory-driven or data-driven approach (Yuan et al., 2019). The second process is the measurement modeling for constructing latent traits of examinees based on the evidence extracted from process data. Some probabilistic measurement models, such as the Markov-IRT model (Shu et al., 2017), the multilevel mixture IRT model (Liu et al., 2018), the

Markova decision process model (LaMar, 2018), continuous-time dynamic choice model (Chen, 2020), and the sequential response model (Han et al., 2021), have been designed for specific problem-solving tasks.

In this study, we focus on the first analytical process. Most practices in large-scale assessment derive indicators from process data based on the predetermined expert rubric (TEL, 2013; OECD, 2016; Care et al., 2017). However, this top-down approach bears a high cost of rulemaking and has the risk of ignoring the non-rule-based operations. On the other hand, the data-driven approach analyzes process data in an exploratory way. In both approaches, one of the main challenges is to distill high-quality and standardized information from the complex action sequences (von Davier et al., 2019). At the item level, there are two main directions for information extraction. The first direction is to aggregate information of actions in action sequences. For example, students' proficiency can be evaluated by comparing the edit distance, such as the Levenshtein distance and the longest common subsequence, between their action sequence and the expert-defined optimal response (Hao et al., 2015; He et al., 2016). Action sequences can also be rescaled, transferred, or embedded into a low-dimensional space using sequence alignment analysis (Hao et al., 2015), multidimensional scaling (MDS; Tang et al., 2020), or neural networks techniques (Tang et al., 2021). The aggregated information of these action sequences can then be treated as the features of students, which has higher accuracy in the prediction of students' performance (Zhang et al., 2021). The second direction is to aggerate information across action sequences to generate features of actions. In this direction, researchers aimed at identifying the role each action plays in problem-solving and how actions interact with each other. For example, natural language processing techniques, such as N-gram and term frequency-inverse document frequency (TF-IDF), have also been applied to identify single actions or sub-sequences (N-grams) that differentiate different proficiency groups, since action sequences can be analogized to strings in natural language (He & von Davier, 2016; Qiao & Jiao, 2018; Stadler et al., 2019; Han et al., 2019). Zhu et al. (2016) also analyzed transitions between actions using social network analysis to extract meaningful action sequences. They computed several local and global network statistics, such as degree centrality, weighted density, and reciprocity, to measure the importance of actions as node. Networks were used to model the eye movement patterns in mathematics problem solving, so that the intercorrelation among eye fixation spots were investigated (Zhu & Feng, 2015).

In this study, we focus on extracting information in the second direction using social network analysis, which analyze the transitions between actions patterns in students' action sequences. Our goal is to analyze the aggregated adjacent matrix of transitions between actions using a statistical modeling framework: latent space model (LSM). LSM is a flexible, expandable, and intuitive method for analyzing the complex network of process data. For analyzing action sequences, it has the potential to describe latent structure of dependencies among actions, measure the effects of auxiliary information, provide supplementary evidence to assess examinee performance, and identify the common problem-solving strategies. The purpose of this study is three-folded: 1) to explore how the complex dependencies among actions, concomitant behavior information, and constraints in the process data can be captured by LSM, 2) to illustrate how the results from LSM can be applied for applications in educational assessment, and 3) to discuss the advantages and limitations of LSM for analyzing process data.

The rest of the paper is organized as follows: we first introduce how the transitions between actions network and LSM can be used to extract the key information from process data. Based on the information extracted from LSM, we propose three applications: action clustering, error analysis, and performance measurement. Then, we examine the LSM and its applications across different process data generation conditions through simulation studies. An empirical study based on one problem-solving item of PISA 2012 is also presented. We conclude with a discussion on the usage of LSM for process data.

Method

From Action sequences to Network

Compared with multivariate data (e.g., testing and survey data), process data is highly unstructured, interdependent, and dynamic. Considering a problem-solving item in which a series of actions is required to be completed, we use $\boldsymbol{A} = \{a_1, \dots, a_N\}$ to denote the set of all possible actions, where N is the number of distinct valid actions. An action sequence of the i-th student is a sequence of actions $\boldsymbol{s_i} = \left(s_1, \dots, s_{L_i}\right)$, where L_i is the length of the sequence.

The structure formed by the actions in action sequences and their interactions could be presented in a network (or called graph) where the nodes represent actions in $\boldsymbol{A}$ and each edge represents a connection between two actions. Variables measured on the nodes (i.e., action attributes) and edges (i.e., transitions between actions attributes) can also be included in the network. The simplest type of adjacent matrix comes in the form of a dichotomous variable, also known as a binary network, which indicates the presence or absence of a connection of interest (e.g., whether the direct transition between two actions is observed). It is also common to find networks in which edges are equipped with weights (e.g., the frequency of transitions between actions) characterizing the corresponding connection between a pair of nodes. The relations between nodes can also be characterized as undirected or directed to indicate any present directionality. In sum, network can be applied for representing the problem-solving processes in various format, which consist of knowledge, concepts, actions, and status as nodes as well as relations, dependencies, and transitions among these nodes as the edge (Zhu, 2021).

A useful way of representing network data is through an adjacent matrix (or called a sociomatrix; Wasserman & Faust, 1994). In cases where there are N possible actions in action sequences, the weighted and directed network of action sequence can be transformed into an $N \times N$ adjacent matrix $\boldsymbol{T} = [t_{ab}]$, with the rows indicating the sending nodes and the columns indicating the receiving nodes. Thus, each element t_{ab} represents the number of times a student takes the b-th action after the a-th action. Since the student's action sequences can be treated as a discrete-time stochastic process (Han et al., 2021), an adjacent matrix of actions can be generated based on the first-order Markov property. When the nodes of "start" and "end" are included, the network can preserve the most information of a sequence of actions by using weighted and directed edges. The network and corresponding adjacent matrix can be defined at different levels. An example of two students with three possible actions (e.g., A, B, and C) is presented in *Figure 1*. We can create an adjacent matrix of response actions for each student. The adjacent matrices at the student level can then be further aggregated into the population level. In this study, all model specifications and data analyses are based on the aggregated adjacent matrix at the population level, since the information from a single action sequence is typically limited and biased.

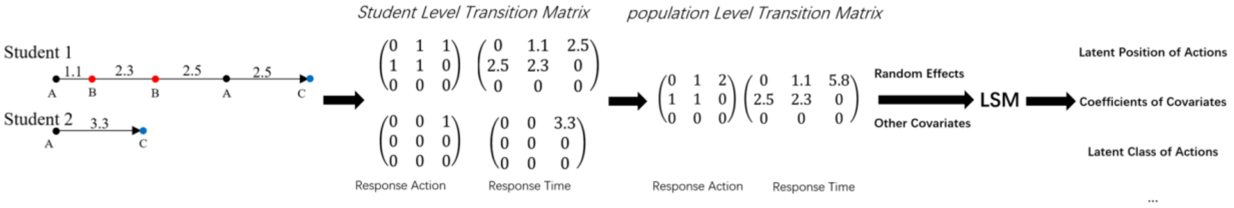


Figure 1. An Example of Extracting Information from Process Data using Latent Space Model

Generally, the auxiliary information about the actions and transitions between actions could be collected from the log of process data, such as the response time (RT). Thus, the formularization of the adjacent matrix can also be customized to capture these features of action sequences. In sum, a network is a useful, flexible, and intuitive tool for preserving complex action sequences. Adjacent matrices further transform the unstandardized information in the network into a standardized format, which captures the main characteristic of interconnectedness and dependencies among actions at different levels.

Latent Space Model for Network

Although descriptive statistics and visualization are informative and effective approaches to summarize the main features of a network, these methods have limitations in making inferences and predicting future networks. Zhu et al. (2016) suggested that statistical modeling techniques in networks should be applied to define probability distributions that account for complex dependencies in the network. Statistical network models aim at identifying the essential stochastic mechanism under which a given network might have arisen, allowing us to test for the significance of predefined features in the network, assess associations between node/edge attributes and the network structure, and impute missing observations (Sosa & Buitrago, 2021). LSM is a representative statistical network model for adjacent matrices.

To model the complex dependencies in a network, LSM assumes that the edges in a network can be modeled as independent random variables, given covariates and latent variables. Specifically, LSM assumes that a node in a network occupies a position as in a low-dimensional latent (Euclidean) space, and distance between two nodes in the latent space explains the likelihood of an edge (Hoff et al., 2002). The general framework of LSM can be expressed as:

$$P(\boldsymbol{T}|\beta, x, Z) = \prod_{(a,b)\in T} P(T_{ab} = t_{ab}|\boldsymbol{\beta}, \boldsymbol{x}, D_{ab}), \tag{1}$$

$$E(T_{ab}|\boldsymbol{\beta},\boldsymbol{x},D_{ab}) = g^{-1}\left(\sum_{k=1}^{P} x_{abk}\beta_k - D_{ab}\right) = g^{-1}\left(\sum_{k=1}^{P} x_{abk}\beta_k - ||\boldsymbol{Z_a} - \boldsymbol{Z_b}||_2\right), \tag{2}$$

$$\boldsymbol{Z_a}, \boldsymbol{Z_b} \sim MVN_d(\boldsymbol{0}, \sigma\boldsymbol{I_d}), \tag{3}$$

where $D_{ab} = ||\boldsymbol{Z_a} - \boldsymbol{Z_b}||_2$ is the pairwise distance between latent positions $\boldsymbol{Z_a}$ for node a and $\boldsymbol{Z_b}$ for node b in a prespecified d-dimensional latent space (i.e., $\boldsymbol{Z_a}, \boldsymbol{Z_a} \in \mathbb{R}^{\boldsymbol{d}}$), which is invariant to rotation. We assume the latent positions follow the multidimensional normal distribution. β_k represents the coefficient for the kth edge covariate x_{abk}. Covariates can also be defined at the edge or node level. Random effects (e.g., the receiver effect or sender effects) can also be included in equation (2), following the framework of generalized linear mixed models (GLMM; McCulloch et al., 2004). To analyze the aggregated adjacent matrix of action sequences, we define the edge as the frequency of transitions between actions and use Poisson regression with logarithm as the link function. The conditional expectation of transitions between actions increases as the distance between the latent position of two nodes decreases along with other covariates.

According to Hoff et al. (2002), a latent structure based on the distance naturally induces homophily, in which the node has closer distances if the transition tendencies in the network are stronger, and the characteristics of nodes are more similar. LSM tries to find a configuration of actions in the latent space such that the interpoint distance models the “affinity” of actions, which indicates how likely two actions will appear together across different action sequences. As an exploratory data analysis tool, LSM provides a parsimonious presentation of complex process data by approximating the high dimensional adjacent matrix with a lower-dimensional latent space. Thus, it is also a model-based embedding technique for dimensionality reduction. The relative positions of actions indicate their degrees of relevance in different problem-solving strategies. For example, actions that belong to the same strategies are more likely to be clustered together in a latent space. The estimated latent positions can be viewed as features of actions in a standardized format, which can be incorporated into traditional psychometrics models.

Incorporating Response Time in LSM

Using RT to infer the nature of cognitive processes has a long history and has been a key element for making inferences about problem-solving processes (De Boeck & Jeon, 2019; Kyllonen & Zu, 2016). However, the associated timing data have mostly been discussed at the item level. For instance, item-level timing data have been used to show how time spent on an item is related to proficiency (Goldhammer et al., 2017; Scherer et al., 2015). Considering timing data at the action level has vast potential to support a more fine-grained assessment of examinee behaviors. (Ulitzsch et al., 2021).

Under the framework of LSM, there are three possible ways to incorporate the information of RT. First, we can build an adjacent matrix of RT between any pair of consecutive actions in the same approach as the adjacent matrix of transitions between actions. Then, the adjacent matrix of RT can be modeled using LSM. Alternatively, RT information can be treated as a covariate to measure the extent of which a transition between actions can be predicted by the features of RT. For example, RT of transitions between actions can be taken as edge covariates. Or, the average RT for each action as the sender (or receiver) can be viewed as node covariates. The flexibility of LSM allows researchers to decide how to incorporate RT information based on different research questions. Following the same idea, other features of action (e.g., type of action as keystrokes or click flow) and transitions between actions can be included in LSM flexibly.

As *Figure 1* indicates, the general approach of using LSM on process data involves first transforming the unstructured action sequence and its auxiliary information into adjacent matrices. Then, adjacent matrices at the student level are aggregated into the population level. Finally, LSM is applied for modeling the adjacent matrix with latent positions of actions and other covariates. Using LSM, we can treat the estimated latent positions as features of actions and estimate the possible fixed or random effects.

Applications

There are two direct results of LSM: latent positions of actions and coefficients of covariates. By extending the model framework and analyzing the results of LSM, we can gain a deeper understanding of process data. In this section, we introduce three possible applications as examples. We provide more explanations and illustrations in the real data example in detail.

Application 1: Action Clustering

Handcock et al. (2007), Krivitsky and Handcock (2008), and Krivitsky et al. (2009) generalize LSM to recreate a model that allows the practitioner to model both transitivity and homophily while simultaneously finding clusters of actors in a model-based fashion. To represent cluster structure in latent space, LSM is extended with an assumption that latent positions of actions are outcomes of a mixture model of normal distributions. For a fixed number of clusters, LSM can be extended with an additional latent structure of the normally distributed subgroups:

$$\boldsymbol{Z_a} \sim \sum_{g=1}^{G} \lambda_g MVN_d\left(\boldsymbol{\mu_g}, \sigma_g^2 \boldsymbol{I_d}\right), \tag{4}$$

where λ_g ($\lambda_g \geq 0$ and $\sum_{g=1}^{G} \lambda_g = 1$) is the probability that the action $\boldsymbol{Z_a}$ belongs to the gth cluster. $\boldsymbol{\mu_g}$ is the mean vector for cluster g. Since the likelihood is invariant to rotations of the latent social space, the covariance matrix is specified independently of the coordinate system. We expect to see actions that belong to different problem-solving strategies arise from different clusters since students tend to generate more transitions between actions among actions of the same cluster. We might improve the overall model fit by assigning actions that are more relevant to the same cluster.

Choosing the number of clusters is a model selection problem. As suggested by Handcock et al. (2007) Bayesian information criterion (BIC) is used to deal with the trade-off between the goodness of fit and the simplicity of the model. Among competing model frameworks with a different number of clusters, we can pick the one with the lowest overall BIC. Different from the standard approach, they applied the conditional posterior model probabilities on the estimate of the latent positions to calculate BIC. In this way, the overall BIC can be approximated by the sum of BIC for the GLMM model (i.e., structure specified in equation (2)) and the BIC for the mixture model (i.e., structure specified in equation (4)).

Application 2: Error Analysis

Following the idea of action clustering, pairwise distances among actions provide a quantitative signal of where errors or confusions appear during the process of problem-solving. Zoanetti (2010) showed the value of process data in providing computer-captured performance evidence of errors, repetitions, and redundancies, in addition to the correctness of a solution. We also find the potential of providing diagnostics of errors in LSM. For example, actions that belong to the optimal strategy but are located near a cluster of incorrect problem-solving strategies are treated as slipping actions because we expect to see students moving from correct strategies to incorrect ones through these actions. Alternatively, they may be signals of a potential design problem, since these actions might be misunderstanding. Similarly, how closely one action is to the 'reset' action also indicates to what extent students are stuck in that step. Detailed information of common problem-solving misconceptions provides useful feedback to the practitioners and researchers on how to improve teaching and modify item design accordingly (Bennett et al., 2007).

Application 3: Performance Measurement

For traditional item formats (e.g., multiple-choice), scoring rubrics are generally straightforward to develop and implement. For simulation and game-based tasks, developing scoring rubrics can be complicated. Compared to a binary score (e.g., correct or incorrect), action sequence provides much more information. For example, Maris and van der Mass (2012) as well as van Rijn and Ali (2018) both designed new scoring rules that include both RT and response accuracy in the framework of IRT.

In many simulation-based tasks, there is an optimal strategy, which refers to the most efficient action sequence to solve a task defined by content experts (He et al., 2019). Students with high ability are expected to have action sequences similar to the optimal strategy since redundant actions are typically associated with struggling or randomly guessing. Hao et al. (2015) found a strong correlation between edit distances and scores obtained from the scoring rubrics of the pump repair task in the National Assessment of Education Progress Technology and Engineering Literacy assessments. Following the same idea of edit distances, we designed a new measurement of performance based on comparing the similarity between the optimal strategy and observed responses in terms of the actions' latent position. To measure the distance between the optimal strategy and observed responses, we used the average linkage:

$$d(\boldsymbol{S_i}, \boldsymbol{S_j}) = \frac{1}{KL}\sum_{i=1}^{K}\sum_{j=1}^{L}\left|\left|\boldsymbol{X_i} - \boldsymbol{Y_j}\right|\right|_2, \tag{6}$$

where $\boldsymbol{X_i}$ $(i = 1, 2, \dots, K)$ is the *K* distinct actions' latent positions from the optimal strategy $(\boldsymbol{S_i})$ and $\boldsymbol{Y_j}$ $(j = 1,2, \dots, L)$ is the L latent positions of actions from a specific action sequence $(\boldsymbol{S_j})$. $\left|\left|\boldsymbol{X_i} - \boldsymbol{Y_j}\right|\right|_2$ represents the Euclidian distance between two actions' latent positions. Average linkage involves observing the distances among all pairs and averaging over all the distances. More actions lead to longer action sequences and more pairs of distances in the average linkage, which is a sign of inefficiency. Systematically incorrect actions also make the average linkage larger since their distance with the actions in the optimal strategy is typically larger. Thus, a small average linkage for an action sequence to the optimal strategy indicates high performance.

Simulation Study

In this section, we demonstrated the proposed feature extraction procedure using simulated data. The goal of this simulation study is to examine the performance and robustness of the LSM in clustering actions and to measure the performance of the action sequences. This simulation study partially followed the design of Tang et al. (2021). We used the R package 'latentnet' (Krivitsky & Handcock, 2008) for LSM. Markov chain Monte Carlo (MCMC) algorithm was applied for parameter estimation with 4000 draws from the posterior distribution, 1000 burn-ins.

Data Generation & Experiment Setting

This simulation study assumed there is an item with ten ($N = 10$) possible actions, which are denoted as $\boldsymbol{A} = \{S, X_1, X_2, X_3, X_4, Y_1, Y_2, Y_3, Y_4, E\}$. All action sequences started with 'S' and end with 'E'. In addition, there were two problem-solving strategies: $\boldsymbol{S_1} = (S, X_1, X_2, X_3, X_4, E)$ and $\boldsymbol{S_2} = (S, Y_1, Y_2, Y_3, Y_4, E)$, with $\boldsymbol{S_1}$ assumed to be the optimal strategy. Students' action sequences were generated from Markov chains, which were determined by the probability adjacent matrix $\boldsymbol{P} = (P_{ab})_{1 \leq a,b \leq N}$ with each element P_{ab} representing a conditional probability of transitions from action a to action b. Given the probability adjacent matrix $\boldsymbol{P}$, we started a sequence with 'S' and generated all subsequent actions according to $\boldsymbol{P}$ until 'E' appeared.

$\widetilde{\boldsymbol{P_1}}$	S	X_1	X_2	X_3	X_4	E
S	0	1	0	0	0	0
X_1	0	0	1	0	0	0
X_2	0	0	0	1	0	0
X_3	0	0	0	0	1	0
X_4	0	0	0	0	0	1
E	0	0	0	0	0	0

$\widetilde{\boldsymbol{P_1}}$+Noise	S	X_1	X_2	X_3	X_4	E
S	0	1	0	0.2	0.3	0.1
X_1	0	0.1	1	0.4	0.1	0.2
X_2	0	0	0.3	1	0.2	0.1
X_3	0	0.2	0.1	0.1	1	0.1
X_4	0	0.1	0.4	0.3	0.2	1
E	0	0	0	0	0	0

$\boldsymbol{P_1}$	S	X_1	X_2	X_3	X_4	E
S	0	0.625	0	0.125	0.188	0.062
X_1	0	0.056	0.556	0.222	0.056	0.111
X_2	0	0	0.187	0.625	0.125	0.063
X_3	0	0.133	0.067	0.067	0.667	0.067
X_4	0	0	0.05	0.2	0.15	0.5
E	0	0	0	0	0	0

Figure 2. Example of Probability Adjacent matrix Generation Process

As shown in *Figure 2*, let ideal matrix $\widetilde{\boldsymbol{P_1}}$ $(\widetilde{\boldsymbol{P_2}})$ denoted the adjacent matrix in probability strictly following the problem-solving strategy $\boldsymbol{S_1}$ $(\boldsymbol{S_2})$ without any noise, which had 1 at the position of transitions between actions in the strategy and 0 at all the other positions. If a student followed the problem-solving strategy $\boldsymbol{S_1}(\boldsymbol{S_2})$, we generated his or her probability adjacent matrix $\boldsymbol{P}$ by adding random noise upon the corresponding ideal matrix $\widetilde{\boldsymbol{P_1}}$ $(\widetilde{\boldsymbol{P_2}})$. Specifically, independent random noises were sampled from the uniform distribution on the interval $[0, R]$ and added to the ideal matrix at each position. The upper bound of random noise (R) was set as a design factor in the simulation study. Then, each element was normalized by dividing its value by the sum of all the elements in its row. This way, students following the same strategies would have similar but distinct action sequences. When modeling the action sequences with LSM, we ignored action ‘S’ and ‘E’ in action sequences since these two actions appeared in all action sequences and would naturally make two clusters of action closer.

We manipulated three design factors in the simulation: the number of students, the proportion of students who follow the optimal strategy, and the upper bound of random noises. There were three levels of sample size: 200, 500, and 1000 students. Proportions of students following the optimal strategy were set to be 30%, 50%, and 70%, respectively. The upper bound of random noise was chosen to be 0.1 and 0.2. Thus, there were 18 conditions in total. We modeled the aggregated adjacent matrix using LSM with two-dimensional latent space and two latent clusters of action without additional covariates.

Results

Table 1 summarized the main results from the simulation study. First, LSM was able to cluster actions into the same group if they belonged to the same problem-solving strategy. If there existed some distinct problem-solving strategy in the process data, the actions within the same strategy tended to interact with each other more frequently than with the actions across different strategies. LSM could detect these strategies based on the aggregated adjacent matrix and group the actions together with high accuracy for all 18 conditions. Second, we compared the average linkage for the two groups of students using different strategies. Based on the Welch two-sample t-tests, students who followed the optimal strategy had a significantly smaller average linkage than students who adopted an alternative strategy. In terms of sample size, the t-test yielded a more significant conclusion when we increased the number of students. For random noise upper bound, the average linkage for students from both the correct and incorrect groups tended to be smaller with more random noises, especially for the incorrect group. Larger random noise gave the students following incorrect strategies a greater chance to take more actions in the optimal strategy. Thus, the difference between correct and incorrect groups tended to be smaller with less significant t statistics. Finally, there was no clear tendency across different conditions when we changed the proportion of students adopting the optimal strategy. The proportion of students adopting the optimal strategy could be interpreted as the item difficulty since a smaller proportion of students adopted the optimal strategy indicated less students had the proficiency to solve the item correctly. Thus, if the item difficulty was within a reasonable range, LSM could distinguish alternative problem-solving strategies from the optimal ones.

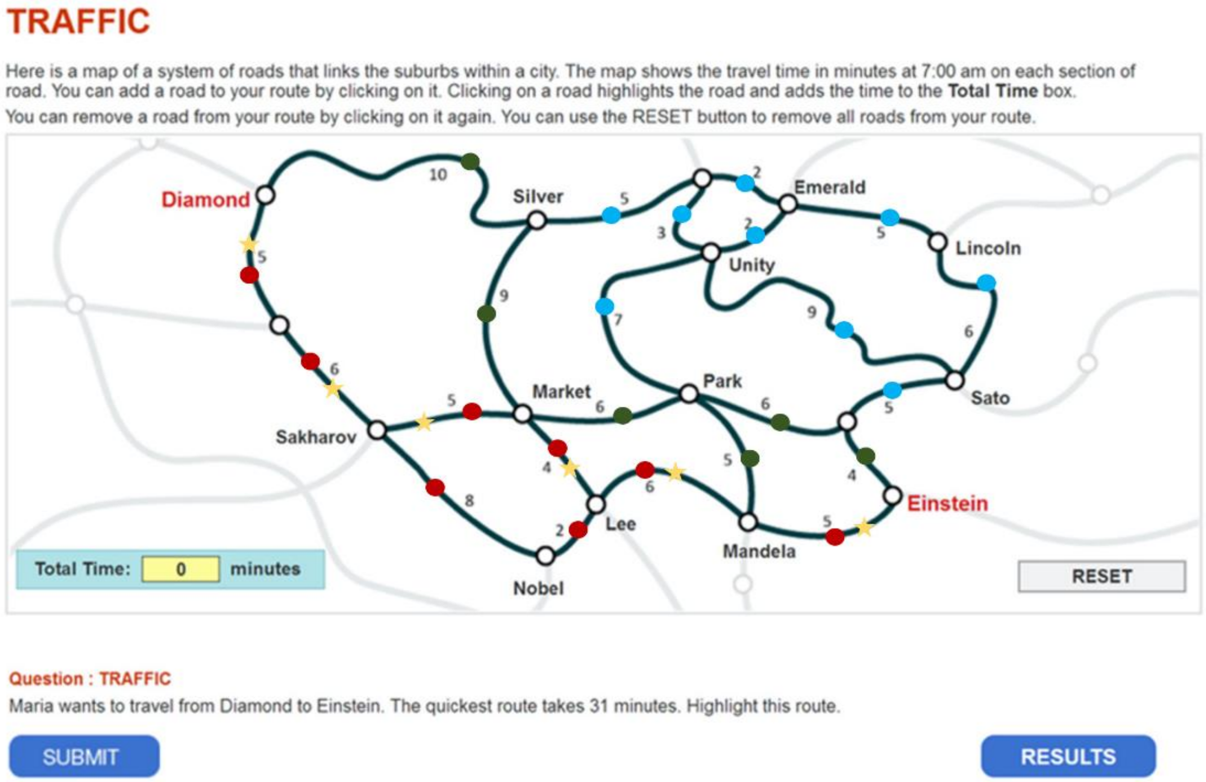
Table 1. Summary of the Simulation Study Results

Number of Students	Random Noise Upper Bound	Proportion of Student Adopting Optimal Strategy	Clustering Accuracy	Average Linkage for the Correct Group	Average Linkage for the incorrect Group	Two-sample t Statistics
200	0.1	0.3	100%	.380	1.109	-16.866
		0.5	100%	.570	1.109	-15.140
		0.7	100%	.650	1.303	-14.996
	0.2	0.3	100%	.333	.670	-10.389
		0.5	100%	.292	.524	-10.611
		0.7	100%	.460	.631	-8.302
500	0.1	0.3	100%	.451	1.162	-24.244
		0.5	100%	.447	1.014	-22.068
		0.7	100%	.555	1.051	-24.988
	0.2	0.3	100%	.312	.630	-16.226
		0.5	100%	.418	.689	-16.233
		0.7	100%	.523	.747	-16.964
1000	0.1	0.3	100%	.398	1.084	-35.373
		0.5	100%	.466	1.082	-36.583
		0.7	100%	.589	1.137	-33.839
	0.2	0.3	100%	.315	.621	-23.171
		0.5	100%	.359	.667	-27.393
		0.7	100%	.412	.665	-22.980

Real Data Analysis

Problem Solving Item and Log Data File

This study illustrates the use of LSM in analyzing the process data through one of the problem-solving tasks in PISA 2012 (Traffic CP007Q8; OECD, 2013). In this task, students were given a map and the travel time on each route. Their goal was to find the quickest route from Diamond to Einstein. Figure 3 shows the interface of this task. The log data contains detailed information about the students' action sequences associated with all click actions and corresponding time, in seconds, since the beginning of the assessment.



TRAFFIC

Here is a map of a system of roads that links the suburbs within a city. The map shows the travel time in minutes at 7:00 am on each section of road. You can add a road to your route by clicking on it. Clicking on a road highlights the road and adds the time to the **Total Time** box.
You can remove a road from your route by clicking on it again. You can use the RESET button to remove all roads from your route.

Total Time: 0 minutes

Question : TRAFFIC
Maria wants to travel from Diamond to Einstein. The quickest route takes 31 minutes. Highlight this route.

Figure 3. Traffic Problem in PISA 2012.
Note that yellow star represents the optimal strategy, red, blue, and green represent three different clusters estimated from LSM.

The study sample was drawn from the PISA 2012 released dataset, which consists of 413 students from the United States. After removing incomplete responses, the sample size for the analysis was 401. Of this group, 75.67% of students answered the item correctly. The average number of recorded click actions in the individual action sequence was 43.88 with a standard deviation of 36.18. The labeled click actions contained not only behaviors on the executable buttons (e.g., road/routes, reset, and submit), but also click behaviors on some inexecutable areas (e.g., map, time minutes, paragraph 1, and city names). In this real data analysis, we only included actions related to the selection of routes into account. The average number of route selection actions in an individual action sequence was 36.55 with a standard deviation of 32.09. The average RT for the action sequence was 102.40 seconds with a standard deviation of 72.69 seconds. The average RT of single-transitions between actions was 2.26 seconds with a standard deviation of 6.21 seconds. After aggregating all the students' transition matrices together, we created an 23 × 23 directed and weighted adjacent matrix. Since we ignored the repeated selection of routes into account, the diagonal of the adjacent matrix was fixed as zero. In this task, the optimal strategy obtained six actions of route selection (*Table 2*). In Figure 3, these actions were marked with the yellow star.

Table 2. Description of the Process Actions

Label	Meaning	In the Optimal Strategy	Average Response Time
P1	Diamond-Nowhere	1	5.37
P2	Diamond-Silver	0	3.64
P3	Emerald-Lincoln	0	1.26
P4	Emerald-Unity	0	1.26
P5	Lee-Mandela	1	1.26
P6	Lincoln-Sato	0	1.48
P7	Mandela-Einstein	1	1.68
P8	Market-Lee	1	1.63
P9	Market-Park	0	1.39
P10	Nobel-Lee	0	1.41
P11	Nowhere-Einstein	0	1.83
P12	Nowhere-Emerald	0	1.31
P13	Nowhere-Sakharov	1	1.61
P14	Nowhere-Unity	0	1.41
P15	Park-Mandela	0	1.62
P16	Park-Nowhere	0	1.46
P17	Sakharov-Market	1	1.51
P18	Sakharov-Nobel	0	1.63
P19	Sato-nowhere	0	1.43
P20	Sliver-Market	0	1.78
P21	Sliver-nowhere	0	1.55
P22	Unity-Park	0	1.49
P23	Unity-Sato	0	1.55

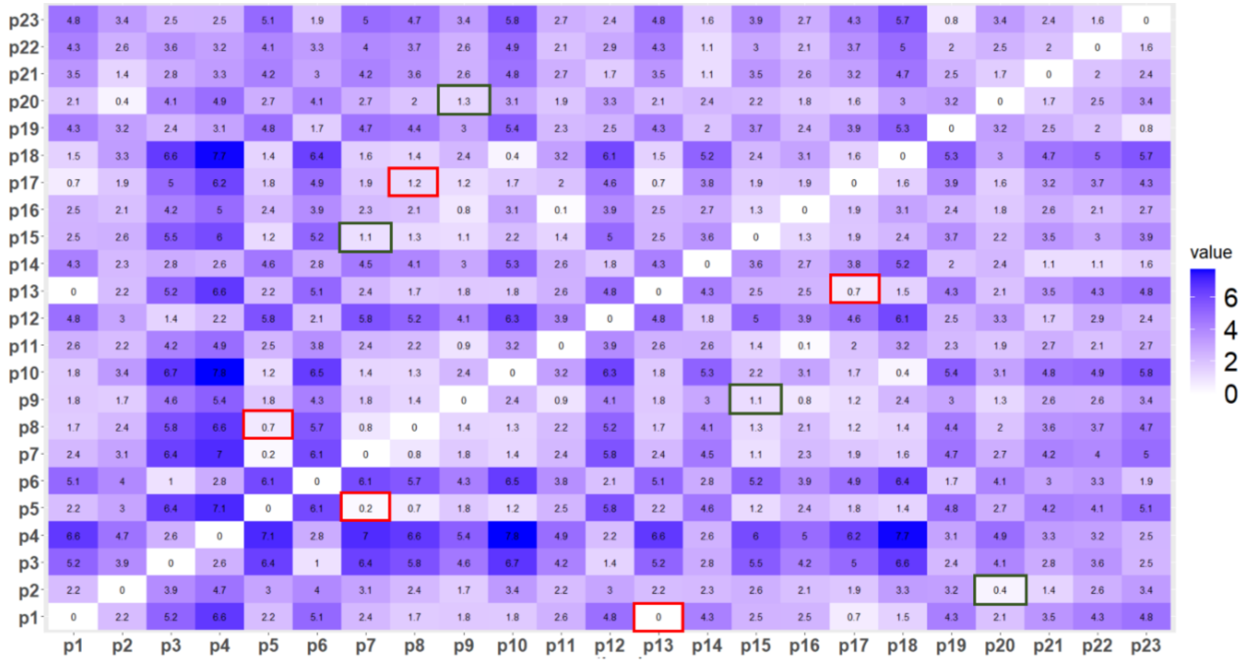
Result

In this study, we fitted the LSM using Poisson regression for the weighted and directed adjacent matrix without self-loops. The average sender and receiver time were also included as the node covariates. By fixing the dimension of latent space as three, we assumed the actions were generated from two distinct clusters since a model with three clusters had the lowest value in overall BIC. The estimated intercept was 4.471 ($p < .05$), which represents the expected number of transitions between any two actions when both the covariate and latent distance were zero. However, the RT for any given transitions between actions could not be zero and the latent distance between two different actions was typically larger than zero. Thus, there was no natural definition of the intercept in LSM. Instead, the intercept represented the overall labor intensity for the item, since we expected to see the transitions between actions across all positions to be larger (i.e., longer action sequences for all students) when the intercept increased. The estimated effect of average sender RT and receiver RT were .005 ($p < .05$) and .008 ($p < .05$), respectively. For one unit change by the average sender (receiver) RT, the log of expected transitions between action's frequency was expected to increase by .005 (.008), given other variables were held constant.

We color-coded three predefined clusters as blue, green, and red in *Figure 3*. Based on the clustering of the actions' latent positions, 9 (39.13%) actions belonged to the blue cluster, 6 (26.08%) actions belonged to the green cluster, and 8 (34.78%) actions belonged to the red cluster. All actions in the optimal strategy belonged to the red cluster. The red cluster also included two actions (i.e., 'Sakharov-Nobel' and 'Nobel-Lee') that were not included in the optimal strategy. These two actions were close to the actions in the optimal strategy, which indicated that students were more likely to make mistakes on these two actions even when they were following the optimal strategy. All actions in the green cluster were also considered slipping actions since they gave other possible routes from Diamond to Einstein that seemed to be comparable with the optimal strategy. As expected, the distance between the centers of the red and green cluster was 1.65, while the distance between centers of the red and blue cluster was 3.48 in the latent space. Generally, the estimated latent position of actions matched well with their position in the *Figure 3* map.

The estimated distance matrix of actions' latent positions was shown in *Figure 4*, with the optimal strategy (in the order of P1-P13-P17-P8-P5-P7) bordered in red. Following the path of optimal strategy in the distance matrix, we could examine where the errors take place. 'Nowhere-Sakharov' (P13) had the shortest distance from 'Diamond-Nowhere' (P1), at zero, meaning most students could be expected to complete these transitions between actions (P1-P13) correctly. Similarly, most students could be expected to successfully reach Market from Sakharov (P17). The most challenging step was choosing the next station after Market correctly, because 'Market-Lee' (P8) might be less likely to be selected than 'Market-Park' (P9), even though it was the right choice. Meanwhile, 'Sakharov-Market' (P17) had a close distance to the starting point 'Diamond-Nowhere' (P1) suggesting that many students might choose to reset after this step. Finally, 'Lee-Mandela' (P5) and 'Mandela-Einstein' (P7) both had a small distance from the last step, which means students were less likely to make mistakes in these two steps. If students took 'Diamond-Silver' (P2) as the first step, then there was another path from Diamond to Einstein (boarded in green in *Figure 4*). Almost all the actions in this alternative path belonged to the green cluster.

Different from the observed frequency in the adjacent matrix, the estimated pairwise distance among actions in latent space followed four axioms of metric space: (1) symmetry: $D_{ab} = D_{ba}$, (2) nonnegativity: $D_{ab} \geq 0$, (3) minimality: $D_{aa} = 0$ (i.e., $D_{ab} = 0$ if and only if $a = b$), and (4) triangle inequality: $D_{ab} + D_{bc} \geq D_{ac}$. For example, if action a and action b were close and action b and action c were close, we expected to see that action a and action c were close in the latent space even if there was no adjacent transition between action a and action c in the observed adjacent matrix. Thus, the distance matrix of actions' latent positions could give a more reliable summary of relationships among actions in action sequences than the adjacent matrix itself. Taking the three-dimension latent position as the features of actions, we could measure the performance of students based on how close the actions were in their action sequence to the actions in optimal strategy. According to the Welch two sample t-test, there was a statistically significant difference in average linkages for the correct and incorrect scored action sequences ($M_1 = 1.659, M_0 = 2.087, SD_1 = 0.485, SD_0 = 0.603, t = 6.379, p < 0.01$). In general, students who gave the correct answers had shorter distances to the optimal strategy and the variation of average linkage was smaller.



	p1	p2	p3	p4	p5	p6	p7	p8	p9	p10	p11	p12	p13	p14	p15	p16	p17	p18	p19	p20	p21	p22	p23
p23	4.8	3.4	2.5	2.5	5.1	1.9	5	4.7	3.4	5.8	2.7	2.4	4.8	1.6	3.9	2.7	4.3	5.7	0.8	3.4	2.4	1.6	0
p22	4.3	2.6	3.6	3.2	4.1	3.3	4	3.7	2.6	4.9	2.1	2.9	4.3	1.1	3	2.1	3.7	5	2	2.5	2	0	1.6
p21	3.5	1.4	2.8	3.3	4.2	3	4.2	3.6	2.6	4.8	2.7	1.7	3.5	1.1	3.5	2.6	3.2	4.7	2.5	1.7	0	2	2.4
p20	2.1	0.4	4.1	4.9	2.7	4.1	2.7	2	1.3	3.1	1.9	3.3	2.1	2.4	2.2	1.8	1.6	3	3.2	0	1.7	2.5	3.4
p19	4.3	3.2	2.4	3.1	4.8	1.7	4.7	4.4	3	5.4	2.3	2.5	4.3	2	3.7	2.4	3.9	5.3	0	3.2	2.5	2	0.8
p18	1.5	3.3	6.6	7.7	1.4	6.4	1.6	1.4	2.4	0.4	3.2	6.1	1.5	5.2	2.4	3.1	1.6	0	5.3	3	4.7	5	5.7
p17	0.7	1.9	5	6.2	1.8	4.9	1.9	1.2	1.2	1.7	2	4.6	0.7	3.8	1.9	1.9	0	1.6	3.9	1.6	3.2	3.7	4.3
p16	2.5	2.1	4.2	5	2.4	3.9	2.3	2.1	0.8	3.1	0.1	3.9	2.5	2.7	1.3	0	1.9	3.1	2.4	1.8	2.6	2.1	2.7
p15	2.5	2.6	5.5	6	1.2	5.2	1.1	1.3	1.1	2.2	1.4	5	2.5	3.6	0	1.3	1.9	2.4	3.7	2.2	3.5	3	3.9
p14	4.3	2.3	2.8	2.6	4.6	2.8	4.5	4.1	3	5.3	2.6	1.8	4.3	0	3.6	2.7	3.8	5.2	2	2.4	1.1	1.1	1.6
p13	0	2.2	5.2	6.6	2.2	5.1	2.4	1.7	1.8	1.8	2.6	4.8	0	4.3	2.5	2.5	0.7	1.5	4.3	2.1	3.5	4.3	4.8
p12	4.8	3	1.4	2.2	5.8	2.1	5.8	5.2	4.1	6.3	3.9	0	4.8	1.8	5	3.9	4.6	6.1	2.5	3.3	1.7	2.9	2.4
p11	2.6	2.2	4.2	4.9	2.5	3.8	2.4	2.2	0.9	3.2	0	3.9	2.6	2.6	1.4	0.1	2	3.2	2.3	1.9	2.7	2.1	2.7
p10	1.8	3.4	6.7	7.8	1.2	6.5	1.4	1.3	2.4	0	3.2	6.3	1.8	5.3	2.2	3.1	1.7	0.4	5.4	3.1	4.8	4.9	5.8
p9	1.8	1.7	4.6	5.4	1.8	4.3	1.8	1.4	0	2.4	0.9	4.1	1.8	3	1.1	0.8	1.2	2.4	3	1.3	2.6	2.6	3.4
p8	1.7	2.4	5.8	6.6	0.7	5.7	0.8	0	1.4	1.3	2.2	5.2	1.7	4.1	1.3	2.1	1.2	1.4	4.4	2	3.6	3.7	4.7
p7	2.4	3.1	6.4	7	0.2	6.1	0	0.8	1.8	1.4	2.4	5.8	2.4	4.5	1.1	2.3	1.9	1.6	4.7	2.7	4.2	4	5
p6	5.1	4	1	2.8	6.1	0	6.1	5.7	4.3	6.5	3.8	2.1	5.1	2.8	5.2	3.9	4.9	6.4	1.7	4.1	3	3.3	1.9
p5	2.2	3	6.4	7.1	0	6.1	0.2	0.7	1.8	1.2	2.5	5.8	2.2	4.6	1.2	2.4	1.8	1.4	4.8	2.7	4.2	4.1	5.1
p4	6.6	4.7	2.6	0	7.1	2.8	7	6.6	5.4	7.8	4.9	2.2	6.6	2.6	6	5	6.2	7.7	3.1	4.9	3.3	3.2	2.5
p3	5.2	3.9	0	2.6	6.4	1	6.4	5.8	4.6	6.7	4.2	1.4	5.2	2.8	5.5	4.2	5	6.6	2.4	4.1	2.8	3.6	2.5
p2	2.2	0	3.9	4.7	3	4	3.1	2.4	1.7	3.4	2.2	3	2.2	2.3	2.6	2.1	1.9	3.3	3.2	0.4	1.4	2.6	3.4
p1	0	2.2	5.2	6.6	2.2	5.1	2.4	1.7	1.8	1.8	2.6	4.8	0	4.3	2.5	2.5	0.7	1.5	4.3	2.1	3.5	4.3	4.8

Figure 4. Distance Matrix of Actions' Latent Position.
Note that the optimal strategy is bordered with red color in the order of P1-P13-P17-P8-P5-P7. A possible alternative strategy is bordered with green color in the order of P2-P20-P9-P15-P7.

Figure 5 illustrated an example of 15 randomly sampled test-takers. The y-axis represented the average linkage of an action sequence to the optimal strategy. The horizontal line represented the theoretical lower bound, which was the average linkage of the optimal strategy to itself. The X-axis indicated the ranking of the test-taker based on their distance to the optimal strategy. The point color represented whether the student was marked as correct or not (i.e., correct marked as grey and incorrectly marked as black). In line with the findings in the *t*-test findings, students who answered the questions correctly yielded a lower average linkage value. As shown, the student with the id "04648" had the smallest average linkage (i.e., 1.12) since he/she only selected the correct routes without taking any unnecessary or incorrect actions. The student with the id "03047" also gave the correct answer but had a distance of 2.94. Taking a close look at their action sequence, we found that this student first tried an incorrect route (P2-P20-P9-P16-P11) in the green cluster. Then, he/she explored actions in the blue clusters (i.e., P6, P3, P14, and P22). Each time he/she chose an incorrect route, he/she deselected the routes back to the starting point rather than using the reset button. When the student finally gave the correct answer, he/she had explored almost all of the routes on the map and had taken 86 actions in total. All the unnecessary actions and incorrect explorations made the average linkage larger. The student with the id “03177” answered the item incorrectly but had an average linkage of 1.22. This student did reach Einstein and all the actions he/she took belong to the red cluster. Since the only action he/she took that was not in the optimal strategy was ‘Sakharov-Nobel’, which was considered a slipping action, he/she received relatively less penalty. In summary, average linkage offered a supplementary measurement of performance beyond binary scoring, which helped to identify those who need more practice even if they were correct and those who failed but were most likely to succeed next time.

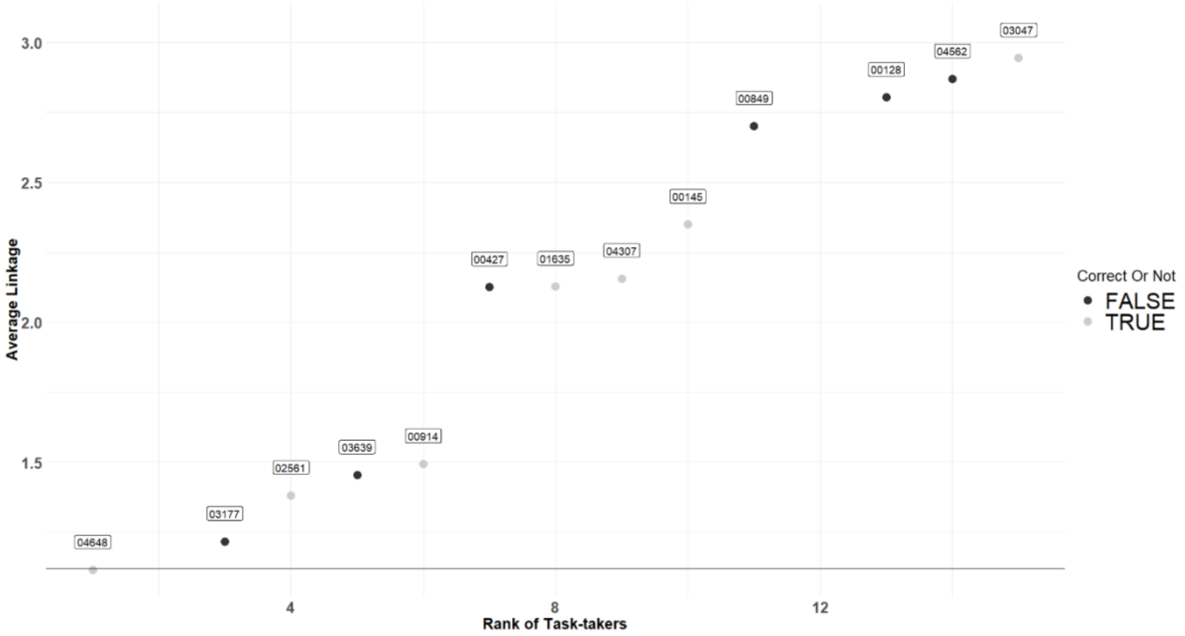


Figure 5. Performance Measurement Based on Average Linkage. The horizontal at bottom refers to the theorical lower bound of average linkage (i.e., 1.12 in this example).

Discussion

In this paper, we introduced the latent space model for analyzing the weighted and directed transitions between actions network of process data, which used low-dimensional geometric projections to represent dependent structures of action in problem-solving strategies. We identified three main benefits of using LSM for process data in educational assessments. First, LSM provides a parsimonious representation of information extracted from educational process data. Model selection methods in LSM provide a method to balance the bias-variance dilemma by comparing the model's structure with different latent space dimensions and setting different number of clusters. As an explanatory method, model selection is based on the estimated configuration for visualization, the interpretation of the model results, and statistical model fit. Second, LSM has the advantages to flexibly incorporate various types of information in the problem-solving process by defining covariates under the framework of GLMM. Moreover, LSM provides a visual and interpretable model-based spatial configuration of the uncertainties in interaction relationships among actions in action sequences. All these benefits of LSM suggest opportunities for deeper analysis of process data.

Multidimensional scaling (Tang et al., 2020) and neural network methods (Tang et al., 2020) generated student features by embedding the action sequences into a fixed dimensional space. Similarly, we created the standardized features of actions by assuming that actions occupy different positions in a low-dimensional latent space. These features in standardized format can be applied to other psychometric implementations. On one hand, the estimated latent position of actions and their clustering structure provide useful information about the latent structure of dependency among actions in the adjacent matrix. On the other hand, the average linkage offers a new measurement of performance. The scoring of simulation and game-based items are primarily governed by the assessment framework. For example, students' performance is scored on both the efficiency and systematicity of their performance in the Well task in NAEP. Efficiency captures the extent to which students are able to take necessary actions only and the speed at which they can respond to specific items. Systematicity captures the extent to which students follow a reasonable and systematic problem-solving routine. To some extent, average linkage takes both dimensions of measurement into account. To compare the distances between two clusters of latent positions, complete linkage, simple linkage, and other matrices could also be explored.

There are several directions to extend the LSM applications in process data. First, methodologies developed in this study can be extended to capture behavioral patterns across multiple items. A major challenge is that features derived from process data are defined at action level and therefore difficult to summarize behaviors across items. He et al. (2021) identified similarities between the students' strategies and the optimal strategy across items, and measured the consistency of these similarities. Following the same idea, we could measure students' average linkages and the consistency of average linkage across items. Secondly, individual adjacent matrices at the individual level were aggregated into the population level in this study. We benefit from focusing on the reinforced information about problem-solving strategies. To model the action sequence at the individual level, the hierarchical latent space model (Sweet et al., 2013) might be used to capture hierarchy at both the individual and population levels. This model accounts for multiple predictors of functional connectivity and individual heterogeneity that manifest over a population. Variational approximations could also be applied to facilitate the application of LSM for larger and more complex networks (Salter-Townshend & Murphy, 2013).

As more research and practice start to explore the next generation of assessment, more studies are needed to further investigate the psychometric issues (e.g., reliability, validity, and scoring) using process data (Bergner & von Davier, 2019). Other data-intensive computational techniques (e.g., educational data mining, dynamic process modeling, and deep learning) make it possible to accurately and fairly interpret results from complex process data. Future research may also explore the similarities and differences between multiple existing methods and identify their advantages in condition-based application. We hope our study can bring some new insights into analyzing process data and have encouraged other researchers and practitioners to use and apply LSM in their work.

References

Bennett, R. E., Persky, H., Weiss, A. R., & Jenkins, F. (2007). *Problem solving in technology-rich environments: A report from the NAEP technology-based assessment project, research and development series, NCES 2007-466*. National Center for Education Statistics.

Bergner, Y., & von Davier, A. A. (2019). Process data in NAEP: Past, present, and future. *Journal of Educational and Behavioral Statistics*, *44*(6), 706–732. https://doi.org/10.3102/1076998618784700

Care, E., Griffin, P. E., & Wilson, M. R. (2017). *Assessment and teaching of 21st century skills: Research and applications*. Springer.

Chen, Y. (2020). A continuous-time dynamic choice measurement model for problem-solving process data. *Psychometrika*, *85*(4), 1052–1075. https://doi.org/10.1007/s11336-020-09734-1

De Boeck, P., & Jeon, M. (2019). An Overview of Models for Response Times and Processes in Cognitive Tests. *Frontiers in Psychology*, *10*. https://doi.org/10.3389/fpsyg.2019.00102

Goldhammer, F., Naumann, J., Rölke, H., Stelter, A., & Tóth, K. (2017). Relating product data to process data from computer-based competency assessment. In D. Leutner, J. Fleischer, J. Grünkorn, & E. Klieme (Eds.), *Competence assessment in education: Research, models and instruments* (pp. 407–425). Springer International Publishing. https://doi.org/10.1007/978-3-319-50030-0_24

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

Han, Y., Liu, H., & Ji, F. (2021). A sequential response model for analyzing process data on technology-based problem-solving tasks. *Multivariate Behavioral Research*, 1–18. https://doi.org/10.1080/00273171.2021.1932403

Han, Z., He, Q., & von Davier, M. (2019). Predictive feature generation and selection using process data from PISA interactive problem-solving items: An application of random forests. *Frontiers in Psychology, 10*. https://doi.org/10.3389/fpsyg.2019.02461

Handcock, M. S., Raftery, R. E., & Tantrum, J. M. (2007). Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series a (Statistics in Society)*, *170*(2), 301–354.

Hao, J., Shu, Z., & von Davier, A. A. (2015). Analyzing process data from game/scenario-based tasks: An edit distance approach. *Journal of Educational Data Mining*, *7*(1), 33–50.

He, Q. F., Borgonovi, F., & Paccagnella, M. (2019). *Using process data to understand adults' problem-solving behaviour in the programme for the international assessment of adult competencies (PIAAC): Identifying generalised patterns across multiple tasks with sequence mining*. OECD Education Working Papers.

He, Q., Borgonovi, F., & Paccagnella, M. (2021). Leveraging process data to assess adults' problem-solving skills: Using sequence mining to identify behavioral patterns across digital tasks. *Computers & Education*, 166, 104170. https://doi.org/10.1016/j.compedu.2021.104170

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*, Volume II (pp. 749–776). Information Science Reference.

Hoff, P. D., Raftery, A. E., & Handcock, M. S. (2002). Latent space approaches to social network analysis. *Journal of the American Statistical Association*, *97*(460), 1090–1098. https://doi.org/10.1198/016214502388618906

Krivitsky, P. N., & Handcock, M. S. (2008). Fitting position latent cluster models for social networks with latentnet. *Journal of Statistical Software*, *24*, 5. https://doi.org/10.18637/jss.v024.i05

Krivitsky, P. N., Handcock, M. S., Raftery, A. E., & Hoff, P. D. (2009). Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social Networks*, *31*(3), 204–213. https://doi.org/10.1016/j.socnet.2009.04.001

Kyllonen, P., & Zu, J. (2016). Use of response time for measuring cognitive ability. *Journal of Intelligence*, *4*(4), 14. https://doi.org/10.3390/jintelligence4040014

LaMar, M. M. (2018). Markov decision process measurement model. *Psychometrika*, *83*(1), 67–88. https://doi.org/10.1007/s11336-017-9570-0

Liu, H., Liu, Y., & Li, M. (2018). Analysis of process data of PISA 2012 computer-based problem solving: Application of the modified multilevel mixture IRT model. *Frontiers in Psychology*, *9*. https://doi.org/10.3389/fpsyg.2018.01372

Maris, G., & van der Maas, H. (2012). Speed-Accuracy response models: Scoring rules based on response time and accuracy. *Psychometrika*, *77*(4), 615–633. https://doi.org/10.1007/s11336-012-9288-y

Mcculloch, C. E., Searle, S. R., & Neuhaus, J. M. (2004). *Generalized, linear, and mixed models*. John Wiley & Sons.

Mislevy, R. J. (2019). Advances in measurement and cognition. *The Annals of the American Academy of Political and Social Science, 683*(1), 164–182.

OECD. (2013). PISA 2012 assessment and analytical framework. In *PISA*. OECD. https://doi.org/10.1787/9789264190511-en

OECD. (2016). *PISA 2015 assessment and analytical framework: Science, reading, mathematic and financial literacy*. OECD Publishing. doi:10.1787/9789264255425-en

Qiao, X., & Jiao, H. (2018). Data mining techniques in analyzing process data: A didactic. *Frontiers in Psychology*, *9*. https://doi.org/10.3389/fpsyg.2018.02231

Salter-Townshend, M., & Murphy, T. B. (2013). Variational Bayesian inference for the latent position cluster model for network data. *Computational Statistics & Data Analysis*, *57*(1), 661–671. https://doi.org/10.1016/j.csda.2012.08.004

Scherer, R., Greiff, S., & Hautamäki, J. (2015). Exploring the relation between time on task and ability in complex problem solving. *Intelligence*, *48*, 37–50. https://doi.org/10.1016/j.intell.2014.10.003

Shu, Z., Bergner, Y., Zhu, M., Hao, J., & von Davier, A. A. (2017). An item response theory analysis of problem-solving processes in scenario-based tasks. *Psychological Test and Assessment Modeling*, *59*(1), 109.

Sosa, J., & Buitrago, L. (2021). A review of latent space models for social networks. *Revista Colombiana de Estadística*, *44*(1), 171–200.

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*. https://doi.org/10.3389/fpsyg.2019.00777

Sweet, T. M., Thomas, A. C., & Junker, B. W. (2013). Hierarchical network models for education research. *Journal of Educational and Behavioral Statistics*, *38*(3), 295–318. https://doi.org/10.3102/1076998612458702

Tang, X., Wang, Z., He, Q., Liu, J., & Ying, Z. (2020). Latent feature extraction for process data via multidimensional scaling. *Psychometrika*, *85*(2), 378–397. https://doi.org/10.1007/s11336-020-09708-3

Tang, X., Wang, Z., Liu, J., & Ying, Z. (2021). An exploratory analysis of the latent structure of process data via action sequence autoencoders. *British Journal of Mathematical and Statistical Psychology*, *74*(1), 1–33. https://doi.org/10.1111/bmsp.12203

TEL. (2013). *Technology and engineering literacy assessments*. https://nces.ed.gov/nationsreportcard/tel/

Ulitzsch, E., He, Q., Ulitzsch, V., Molter, H., Nichterlein, A., Niedermeier, R., & Pohl, S. (2021). Combining clickstream analyses and graph-modeled data clustering for identifying common response processes. *Psychometrika*, *86*(1), 190–214. https://doi.org/10.1007/s11336-020-09743-0

van Rijn, P. W., & Ali, U. S. (2018). A generalized speed–accuracy response model for dichotomous items. *Psychometrika*, *83*(1), 109–131. https://doi.org/10.1007/s11336-017-9590-9 von Davier, A. A., Mislevy, R. J., & Hao, J. (2021). *Computational psychometrics new methodologies for a new generation of digital learning and assessment: With examples in R and python*. Cham, Switzerland Springer.

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. https://doi.org/10.3102/1076998619881789

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications*. Cambridge University Press, Cop.

Yuan, J., Xiao, Y., & Liu, H. (2019). Assessment of collaborative problem solving based on process stream data: A new paradigm for extracting indicators and modeling dyad data. *Frontiers in Psychology*, *10*. https://doi.org/10.3389/fpsyg.2019.00369

Zhang, S., Wang, Z., Qi, J., Liu, J., & Ying, Z. (2021). Accurate assessment via process data. *ArXiv Preprint*. arXiv:2103.15034

Zhu, M. (2021). Social Networks Analysis. In: von Davier, A.A., Mislevy, R.J., Hao, J. (eds) *Computational Psychometrics: New Methodologies for a New Generation of Digital Learning and Assessment*. Methodology of Educational Measurement and Assessment. Springer, Cham. https://doi.org/10.1007/978-3-030-74394-9_13

Zhu, M., & Feng, G. (2015). An exploratory study using social network analysis to model eye movements in mathematics problem solving. *Proceedings of the 5th International Learning Analytics and Knowledge Conference (LAK '15)*, 383–387. http://dx.doi. org/10.1145/2723576.2723591

Zhu, M., Shu, Z., & von Davier, A. A. (2016). Using networks to visualize and analyze process data for educational assessment. *Journal of Educational Measurement*, *53*(2), 190–211. https://doi.org/10.1111/jedm.12107

Zoanetti, N. (2010). Interactive computer based assessment tasks: How problem-solving process data can inform instruction. *Australasian Journal of Educational Technology*, *26*(5). https://doi.org/10.14742/ajet.1053