LATENT TRANSITION ANALYSIS FOR LONGITUDINAL COGNITIVE DIAGNOSIS MODELING

By

YASEMIN KAYA

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2015

To my family

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my advisor, Dr. Walter Leite, for his excellent guidance, patience, and providing me an exceptional atmosphere for doing research. I would also like to thank my committee members, my friends, and my family. I would never have been able to finish my dissertation without their help and support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ABBREVIATIONS

AIC	Akaike information criterion
BIC	Bayesian information criterion
CCR	Correct classification rate
CDM	Cognitive diagnosis models
C-RUM	Compensatory reparameterized unified model
CTR	Correct transition rate
DCM	Diagnostic classification models
DINA	Deterministic input noisy-and-gate model
DINO	Deterministic input noisy-or-gate model
EAP	Posterior probabilities
IRT	Item response theory
LCA	Latent class analysis
LTA	Latent transition analysis
MCCR	Marginal correct classification rate
NIDO	Noisy input deterministic-or-gate model
NIDA	Noisy-input deterministic-and-gate model
NC-RUM	Non-compensatory reparameterized unified model

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

LATENT TRANSITION ANALYSIS FOR LONGITUDINAL COGNITIVE DIAGNOSIS MODELING

By

Yasemin Kaya

August 2015

Chair: Walter Lana Leite
Major: Research and Evaluation Methodology

Cognitive diagnosis models are diagnostic models used to classify respondents into homogenous groups based on multiple categorical latent variables representing the measured cognitive attributes, and applied in education and social sciences. This study aims to develop a longitudinal model for cognitive diagnosis modeling, which can be applied to repeated measurements in order to monitor attribute stability of the individuals, and to account for respondent dependence. A model based on latent transition analysis modeling in cognitive diagnosis has been developed and the use of the proposed model has been evaluated through a Monte Carlo simulation study. The study results indicate that the use of latent transition analysis for longitudinal cognitive diagnosis modeling provides sufficiently correct classification rates.

CHAPTER 1
INTRODUCTION

Cognitive Diagnosis Models (CDMs), alternatively called as Diagnostic classification models (DCMs) (e.g., Tatsuoka, 1983; Nichols, Chipman, & Brennan, 1995; Templin & Henson, 2006), are models used to classify respondents into homogenous groups based on multiple categorical latent variables representing the measured cognitive attributes (Rupp, Templin, & Henson, 2010). CDMs are a group of models used in development and implementation of diagnostic assessments. In other words, CDMs provide classification of respondents into mastery profiles of measured attributes that respondents need to acquire in order to respond correctly to the assessments (Rupp et.al., 2010).

CDMs are commonly used for analyzing observed categorical response data collected in diagnostic assessments in education and social sciences. There are three domains where CDMs can be meaningfully applied (Rupp et al., 2010): clinical diagnosis of psychological disorders, cognitive diagnosis assessment in education, and standards-based assessment in education. Because clinical use of CDMs is not the scope of my study, I will focus on the educational applications of it. In education, CDMs are used to provide exploratory information for instruction and learning. As a secondary aim, they are also used for standardized-based classifications (Rupp et.al., 2010).

Diagnostic modeling has recently received increasing attention of researchers in education because of its potential use for diagnosing individual mental states and weaknesses in learning skills. However, the majority of the research focuses on analyzing a single collection of data, without paying attention to the following stages of students' mastered skills or development of the status of students over time. A cognitive diagnostic model has not been proposed to follow student improvement in skill mastery through time using longitudinal assessments.

Latent transition analysis (LTA) is a longitudinal application of the latent class analysis (LCA) model, which aims to identify whether there is a change between latent classes over time (Collins & Lanza, 2010). While LCA is performed on a set of variables collected at one time to identify latent classes that represent the response patterns in the data, LTA is applied to measurements repeated over time to follow the move between the identified latent classes (Collins & Lanza, 2010). Majority of the CDMs are methodologically classified into two categories: Item Response Theory (IRT) based models that estimate the probability of answering an item correctly through a logit function, and LCA based models that classify respondents into diagnostic classes in which same probability if estimated for people in the same diagnostic class. In order to obtain individual academic growth of students' cognitive ability across consecutive occasions, von Davier and Xu (2011) extended IRT approaches (Anderson, 1985; Embretson, 1991) via the General Diagnostic Model (von Davier, 2005) and illustrated the use of IRT models for measuring change in longitudinal designs. However, as a longitudinal application of LCA, I adapted LTA to cognitive diagnosis analysis in order to explain changes of the attribute profiles of the students over time.

In order to address these issues mentioned above, this study has developed a longitudinal model for cognitive diagnosis modeling, which can be applied to repeated measurements in order to monitor transition of individuals between cognitive classes and to observe gain of necessary attributes throughout time in a curricular setting. My goal was to develop a model based on an innovative use of the LTA model, more generally mixture modeling, in cognitive diagnosis. Through the development of a model combining the LTA and CDMs, I aimed to address within-individual dependence in follow up measurements of learning, as well as moving and staying groups in between cognitive classes.

CHAPTER 2
LITERATURE REVIEW

Overview of Cognitive Diagnosis Models

Cognitive diagnostic assessments in education have been developed to assess cognitive processes behind learning, and provide formative information about examinees. In more technical terms, educational cognitive diagnosis is the assessment that is used to diagnose the cognitive attributes that an examinee needs to master in order to achieve a particular domain, and to classify such examinees to define diagnostic groups based on the attributes they possess (Rupp et.al., 2010).

Cognitive diagnosis provides a detailed evaluation of mastered skills or attributes for an examinee rather than giving a general score. Before using cognitive diagnostic assessments, it is first necessary to identify the latent variables that are measured by a particular assessment. The latent variables measured by cognitive diagnostic assessments are labeled differently in the literature including skills, attributes, and latent traits. In the current study, I used the term *attributes* to refer to latent variables in CDMs.

A q-matrix is an essential piece of cognitive diagnostic models because it defines the connection between items and attributes. More specifically, a q-matrix is an item by attribute table that specifies whether each attribute is measured by each item through binary entries of 1s or 0s (Rupp et.al. 2010). Let q_{ia} symbolize an element of a q-matrix in which *i* and *a* are the number of rows and columns, respectively representing the items and the attributes. A value of 1 represents that item *i* measures attribute *a*.

CDM estimates two item parameters, which are slipping and guessing parameters, and one person parameter, which is α. I define these parameters in detail in a later section.

Different CDM models exist in the literature, which can be classified as being compensatory or non-compensatory. Non-compensatory CDMs assume that lack of one latent variable, which is an attribute in our context, cannot be compensated by another latent variable. Common examples of non-compensatory models are the deterministic-input noisy-and-gate (DINA), the noisy-input deterministic-and-gate (NIDA), and non-compensatory reparameterized unified model (NC-RUM) (Haertel, 1989; de la Torre& Douglas, 2004; Junker & Sijtsma, 2001; DiBello, Stout & Roussos, 1995; Hartz, 2002). On the other hand, compensatory CDMs presume that lack of one latent variable can be compensated by the presence of another latent variable. Examples of commonly used compensatory models are the deterministic input noisy-or-gate model (DINO), the noisy input deterministic-or-gate model (NIDO), and the compensatory reparameterized unified model (C-RUM) (Templin & Henson, 2006; Templin, 2004; Hartz, 2002).

In this section a brief review of the cognitive diagnosis models mentioned above will be provided. Even though I aim to provide notable characteristics of these selected models, my study particularly focuses on DINA and DINO models. I have evaluated the performances of two core CDM models (i.e., DINA and DINO). I have chosen these two models, because of their common use in current methodological and empirical studies (e.g., Henson & Douglas, 2005; de la Torre, 2008; von Davier, 2014).

Non-Compensatory Models

DINA model

The DINA model (Macready &Dayton, 1977; Haertel, 1989; Junker &Sijtsma, 2001; de la Torre & Douglas, 2004), is a non-compensatory CDM because it assumes that lack of one attribute cannot be reimbursed by the existence of another attribute. DINA is also a model works with a conjunctive condensation rule, which means that, to be able to have a high probability of responding an item correctly, an individual needs to master all the attributes required by that item. Lack of one attribute would tremendously drop the probability. The main limitation of DINA model is that it does not make a distinction between the respondents who did not master one or more attributes.

CDM classifies each respondent to one cognitive class in which all the individuals will have the same pattern of attribute mastery. DINA model estimates the probability of a correct response to item *i* for all the respondents in latent class *c* as follows:

$$\pi_{ic} = P(X_{ic} = 1 \mid \xi_{ic}) = (1 - s_i)^{\xi_{ic}} g_i^{(1-\xi_{ic})} \tag{2-1}$$

where π_{ic} is the probability of correct response, X_{ic} is the observed response, ξ_{ic} is the attribute mastery indicator, and s_i and g_i are respectively the slipping and the guessing parameters. If a respondent masters all the required attributes, $\xi_{ic} = 1$, the probability of responding the item correctly is equal to the probability of not slipping for the item, $1 - s_i$. On the other hand, if the respondent fails to master at least one of the required attributes, $\xi_{ic} = 0$, the probability of responding the item correctly is drops to the probability of guessing for the item, g_i. The DINA model order constrains the slipping and guessing parameters: $1 - s_i$ is assumed to be greater than g_i, thus, the probability of responding an item correctly is guaranteed to be always higher for the respondents who mastered all the measured attributes than the respondents who lacked at least one of the measured attributes, regardless of the magnitudes of slipping and guessing parameters (Rupp et.al. 2010).

The attribute mastery indicator is formulated as follows:

$$\xi_{ic} = \prod_{a=1}^{A} \alpha_{ca}^{q_{ia}} \tag{2-2}$$

where *A* is the total number of attributes measured, and q_{ia} indicates whether attribute *a* is measured by item *i.* The possible values that q_{ia} takes are 0 or 1. The other indicator α_{ca} identifies whether the respondent in latent class *c* mastered attribute *a*, which is defined with the values of 0 or 1 as well. Since ξ_{ic} is created through multiplication of each alpha for every measured attribute, lack of a single measured attribute would cause the value of ξ_{ic} to be 0 as well.

The slipping parameter s_i is defined for the probability of responding an item incorrectly for a respondent who has mastered all the required attributes, whereas the guessing parameter g_i defines the probability of responding an item correctly for a respondent who has not mastered at least one required attribute (Rupp et.al. 2010).

$$s_i = P(X_{ic} = 0 \mid \xi_{ic} = 1) \tag{2-3}$$

$$g_i = P(X_{ic} = 1 \mid \xi_{ic} = 0) \tag{2-4}$$

NIDA model

The Noisy Inputs, Deterministic and gate model (NIDA) is a non-compensatory conjunctive CDM model (Maris 1999; Junker & Sijtsma, 2001). Similar to DINA model, NIDA model assumes that lack of one attribute cannot be compensated by another attribute, and master of all attributes is required to get a high probability of responding an item correctly. The probability of a correct response to item *i* in latent class *c* is defined as follows:

$$\pi_{ic} = P(X_{ic} = 1 | \alpha_c) = \prod_{a=1}^{A} [(1 - s_a)^{\alpha_{ca}} g_a^{1-\alpha_{ca}}]^{q_{ia}} \tag{2-5}$$

where X_{ic} is the observed response for item *i* in latent class *c*, α_c is the attribute mastery indicator for class *c*, s_a is the slipping parameter for attribute *a*, g_a is the guessing parameter for attribute *a*, and q_{ia} is the attribute measurement indicator from q-matrix.

Unlike DINA model, NIDA model estimates slipping and guessing parameters at the attribute level, constraining all the items to have the same slipping and guessing parameters. These parameters are formulated as follows:

$$s_a = P(\zeta_{cia} = 0 | \alpha_{ca} = 1) \tag{2-6}$$

$$g_a = P(\zeta_{cia} = 1 | \alpha_{ca} = 0) \tag{2-7}$$

where ζ_{cia} is the observed response indicator (Rupp et.al. 2010).

NC-RUM

The NC-RUM is a CDM in which slipping and guessing parameters are not constrained across items or attributes, unlike DINA and NIDA (DiBello, Stout &Roussos, 1995; Hartz, 2002). The probability of a correct response to item *i* in latent class *c* in NC-RUM model is defined as follows:

$$\pi_{ic} = P(X_{ic} = 1 | \alpha_c) = \pi_i^* \prod_{a=1}^{A} r_{ia}^{*(1-\alpha_{ca}) q_{ia}} \tag{2-8}$$

where X_{ic} is the observed response for item *i* in latent class *c*, α_c is the vector of all attribute mastery indicators for all attributes in latent class *c*, α_{ca} is attribute mastery indicator for attribute *a* in latent class *c*, q_{ia} is the attribute measurement indicator for item *i* and attribute *a*, π_i^* is the baseline probability for item *i* that is calculated through the multiplication of the probabilities of all mastered required attributes, and r_{ia}^* is the penalty probability for item *i* and attribute *a* that is calculated as the ratio of the probability of guessing to the probability of not slipping. The following equations define the baseline probability π_i^* and the penalty probability r_{ia}^* parameters, respectively (Rupp et.al. 2010).

$$\pi_i^* = \prod_{a=1}^{A} \pi_{ia}^{q_{ia}} \tag{2-9}$$

where π_{ia} is the probability of correctly responding item *i* for attribute *a*. And,

$$r_{ia}^* = \frac{r_{ia}}{\pi_{ia}} \tag{2-10}$$

where r_{ia} is the probability of guessing item *i* for attribute *a*.

Compensatory Models

DINO model

The deterministic input, noisy-or-gate model, known as DINO, is a compensatory CDM (Templin & Henson, 2006; Templin, 2004) because it assumes that lack of one measured attribute can be compensated by another attribute. More specifically, mastery of at least one attribute compensates deficit of all the other measured attributes. Similarly to the DINA model, the slipping and the guessing parameters are estimated at the item level. The DINO model works with a disjunctive condensation rule in which the presence of at least one measured attribute guaranties a high probability of endorsing an item (Rupp et.al. 2010).

DINO model estimates the probability of a correct response for item *i* in latent class *c* as follows:

$$\pi_{ic} = P(X_{ic} = 1 | \omega_{ic}) = (1 - s_i)^{\omega_{ic}} g_i^{(1-\omega_{ic})} \tag{2-11}$$

where π_{ic} is the probability of correct response, X_{ic} is the observed response, ω_{ic} is the latent response variable, and s_i and g_i are respectively the slipping and the guessing parameters (Rupp et.al. 2010). The latent response variable ω_{ic} in the DINO model above is defined as follows:

$$\omega_{ic} = 1 - \prod_{a=1}^{A} (1 - \alpha_{ca})^{q_{ia}} \tag{2-12}$$

where q_{ia} specifies whether attribute *a* is measured by item *i.* q_{ia} takes the binary values of 0 or 1. Likewise, α_{ca} indicates whether the respondent in latent class *c* mastered attribute *a*, which is defined with the values of 0 or 1 as well. In case that attribute *a* is not measured by item *i*, q_{ia} would take a value of 0, and consequently the value of $(1 - \alpha_{ca})$ would not matter. On the other hand, if attribute *a* is measured by item *i*, q_{ia} would take a value of 1, and accordingly $(1 - \alpha_{ca})$ counts for the final value that ω_{ic} takes. If the respondent in latent class *c* masters attribute *a*, α_{ca} takes a value of 1, and thereby $(1 - \alpha_{ca})$ would be 0. However, if the respondent in latent class *c* does not master attribute *a*, $(1 - \alpha_{ca})$ ends up being 1. Because the occurrence of $\omega_{ic} = 1$ depends on existence of at least one 0 in the multiplication term, mastering at least one attribute greatly increases the probability of endorsing the item. DINO model is useful when only one attribute is required to be mastered among more than one attribute

(Rupp et.al. 2010). The slipping, s_i, and the guessing, g_i, parameters of the DINO model are defined in the same way as in the DINA model.

NIDO model

The noisy input, deterministic-or-gate model (NIDO) is a compensatory CDM (Templin, 2006), where the probability of correctly responding item *i* in the latent class *c* is defined through an IRT-based logistic regression equation, as follows:

$$\pi_{ic} = P(X_{ic} = 1 | \alpha_c) = \frac{\exp\left(\sum_{a=1}^{A}(\lambda_{.,0,(a)} + \lambda_{.,1,(a)}\alpha_{ca})q_{ia}\right)}{1 + \exp\left(\sum_{a=1}^{A}(\lambda_{.,0,(a)} + \lambda_{.,1,(a)}\alpha_{ca})q_{ia}\right)} \tag{2-13}$$

where $\lambda_{.,0,(a)}$ is the intercept and $\lambda_{.,1,(a)}$ is the slope parameters at the attribute level, α_{ca} is the attribute mastery indicator for attribute *a* in class *c*, and q_{ia} is the attribute measurement indicator for item *i* and attribute *a*. The NIDO model is limited due to its restriction on the parameters across items. All the slope and intercept parameters are defined at the attribute level, and restricted to be equal for all items (Rupp et.al. 2010).

C-RUM

The compensatory reparameterized unified model (C-RUM) is the most flexible compensatory CDM, because it is estimated without any item or attribute constrains (Hartz, 2002; von Davier & Yamamoto, 2004). Similar to the NIDO model, the C-RUM is defined with a logistic regression equation as follows:

$$\pi_{ic} = P(X_{ic} = 1 | \alpha_c) = \frac{\exp\left(\lambda_{i,0} + \sum_{a=1}^{A}\lambda_{i,1,(a)}\alpha_{ca}q_{ia})\right)}{1 + \exp\left(\lambda_{i,0} + \sum_{a=1}^{A}\lambda_{i,1,(a)}\alpha_{ca}q_{ia})\right)} \tag{2-14}$$

where $\lambda_{i,0}$ is the intercept parameter for item *i*, and $\lambda_{i,1,(a)}$ is the slope parameter for item *i* and attribute *a*, α_{ca} is the attribute mastery indicator for attribute *a* in class *c*, and q_{ia} is the attribute measurement indicator for item *i* and attribute *a*. The C-RUM estimates one intercept parameter for every item, and one slope parameter for every item and attribute combination. Thus, C-RUM model overcomes the main limitation of NIDO model, which is that it constrains all the parameters to be equal across items.

Specification of Q-matrix for Cognitive Diagnostic Modeling

The estimation of a q-matrix is made by the educational experts based on their theoretical judgments. Any misspecification of the q-matrix patterns can cause error on classification of examinees to diagnostic groups. In order to minimize the classification error caused by false specification of the q-matrix, and to improve and evaluate q-matrix specification methods, numerous studies have been conducted within the q-matrix specification and validation procedures, and the performance of the CDMs under different q-matrix specification conditions.

Chiu (2013) developed a q-matrix enhancement method and evaluated it through three simulation studies and a real data analysis. DINA and NIDA models were applied in the cognitive diagnosis modeling step. The three simulation studies were applied to evaluate effectiveness of the q-matrix refinement method; effects of the number of misspecified entries on q-matrix; and effect of misspecification type respectively. In the first simulation, the study was designed with varying conditions of sample size (100, 500, 1000), number of items (20, 40, 80), number of attributes (3, 4, 5), CDM models (DINA, NIDA), slipping and guessing parameters (upper bounds: 0.2, 0.3, 04, 0.5), and misspecification of q-entries (10%, 20%). In the second and the third simulation studies, the conditions were fixed at 1000 examinees, 80 items, and 4 attributes. The analysis results showed that the proposed method improved the accuracy of q-matrix estimation. However, the author concluded that evaluation of the q-matrix refinement across DINA and NIDA models might not be valid due to the fact that the slipping and guessing parameters function at the attribute level in the NIDA model, as oppose to the DINA model.

Liu, Xu, and Ying (2012) proposed a model to identify q-matrix and estimate related model parameters. A simulation study was conducted to demonstrate the performance of the proposed model under the DINA model. The number of items and attributes were fixed at 20 and 3, respectively. The attributes were generated from a uniform distribution. The slipping and the guessing parameters were fixed at 0.2. Four different sample sizes were simulated (500, 1000, 2000, 4000), and 100 data sets were generated under each condition. The researchers concluded that the proposed model performed well with large sample sizes. An extension of the estimation procedure was recommended for the DINO model as the dual model of the DINA model.

In his study, Xu (2013) focused on statistical inference of CDM. He conducted number of simulations to demonstrate identifiability, estimation, and hypothesis testing of q-matrices by generating data from the DINA model under different conditions as being one of the most commonly used CDM model. The first set of simulation studies varied based on the availability of true q-matrix information. The estimated and the true q-matrices were compared for the purpose of evaluation. The simulation studies contained 20 items and three attribute levels (3, 4, 5), and the attributes were generated from a uniform distribution. The slipping and guessing parameters were fixed at 0.2.

Four sample sizes were considered (500, 1000, 2000, 4000). The second set of simulation studies were conducted to investigate the statistical inference of q-matrices under the DINA model. Simulations varied based on the availability of real data parameters. In the first simulation of the second set of simulations, the item number was fixed at 20, and the number of attributes varied as 3, 4, and 5. Four different sample sizes were estimated (250, 500, 1000, 2000). The slipping and the guessing parameters were fixed at 0.2. In the second simulation, unlike the first one, the number of items was fixed at 10. Similar to Liu, Xu, and Ying (2012), Xu (2013) recommended the extension of the analysis strategies to the other CDM models, such as the DINO model, for the further research purposes.

Another study, that proposed a mathematical framework to construct q-matrix, was conduct by Xiang (2013). In order to examine the performance the method on q-matrix estimation accuracy, the author conducted two simulation studies that generated the data through the DINA model. The first simulation study generated data with 30 items, 5 attributes, 2000 examinees, and slipping and guessing parameters of 0.1. The second simulation study was conducted based on a real data matrix of a simplified version of fraction subtraction data of Tatsuoka (1990), which was simplified and used by another researcher (de la Torre, 2009b). The reason of using a q-matrix from a real data application was explained as examining the proposed model under a more complicated and unorganized true q-matrix. With DINA model estimation, the data and q-matrix conditions were fixed as 15 items, 5 attributes, 2000 examinees, and 0.1 slipping and guessing parameters. The results were evaluated based on the proportion of correct identification of the q-matrix.

Close (2012) proposed a technique based on principal components analysis to estimate true q-matrices for DINA model, and investigated the feasibility of the technique through a simulation study and three empirical studies. In the simulation study, data were simulated with 4000 examinees, 21 items, and 3 attributes using DINA model. The slipping and the guessing parameters were estimated from a uniform distribution with the lower bounds of 0.02 and 0.05, and upper bounds of 0.05 and 0.25, respectively. The author concluded that use of components analysis could provide improvement on q-matrix estimation for diagnostic items and items measuring limited content domains.

Previous research has proved the proper use of several methods that provide improvements on the correct identification of under different condition settings. Because evaluation of q-matrix specifications are not the scope my research, in this study, it was be assumed that the q-matrix has been correctly specified.

The Latent Class Analysis (LCA) Model

The DINA and DINO models can be viewed as constrained LCA models. These constraints are placed on class probabilities, and the models are structured by specified q-matrices. Thus in this section before extending these CDMs to longitudinal data, unrestricted LCA model has been briefly summarized. In latent variable analyses, latent variables are measured by observed indicators, and the name of the latent variable analysis is determined depending on nature of the latent and observed variables: continuous or categorical. Latent class analysis model is one of the latent variable analyses in which each latent variable and observed indicator is categorical. Categorical latent variables have multinomial distributions. Because of their categorical nature, latent variables are called as latent classes in LCA models. The main purpose of applying an LCA to a set of data is to define a group of latent classes that correspond to existing response patterns, and explain occurrence of each class (Collins & Lanza, 2010). Every individual belongs to only one latent class. Names of the latent classes are given by the researchers depending on the explained theory. The latent class model is:

$$P(Y=y)=\sum_{c=1}^{C}\gamma_c\prod_{j=1}^{J}\prod_{r_j=1}^{R_j}\rho_{j,r_j|c}^{I(y_j=r_j)} \tag{2-15}$$

where $I(y_j = r_j)$ is an indicator function that is equal to 1 when the response is r_j, and equal to 0 otherwise. In LCA, two different parameters are estimated: (a) γ_c is the latent class prevalence, and (b) $\rho_{j,r_j|c}$ is the item-response probability. Latent class prevalences are the probabilities of being in a latent class. Each individual belongs to only one latent class. Therefore, latent class prevalences for whole group of individuals sum to 1.

$$\sum_{c=1}^{C}\gamma_c = 1 \tag{2-16}$$

where γ_c is the probability of being in latent class c (Collins & Lanza, 2010).

Latent classes are interpreted through item-response probabilities. Item-response probabilities in LCA represent probabilities of responding a particular response, “yes” for most of the dichotomous items, to an item conditional on being a member of a latent class. The probability estimates for all the possible responses to each item conditional on a latent class sum to 1.

$$\sum_{r_j=1}^{R_j}\rho_{j,r_j|c} = 1 \tag{2-17}$$

where $\rho_{j,r_j|c}$ is the probability of response r_j to observed variable j, conditional on membership in latent class c, when there are $\boldsymbol{j = 1,...,J}$ observed variable, and observed variable $\boldsymbol{j}$ has $\boldsymbol{r_j = 1,...,R_j}$ response categories. (Collins & Lanza, 2010).

The most fundamental decision made by the researchers in LCA is to decide about the number of classes that represents the data. There are various considerations on deciding the number of classes. The first consideration is parsimony. If model fit and all the other requirements are equally carried by two nested models, the simpler model is preferred in the parsimonious approach. In terms of LCA language, models with less number of classes are preferred against more complex models, in case of equal model fit values. The second consideration is interpretability of the classes based on proposed theory. When number of classes is being chosen, it is researchers' responsibility to make a sensible decision in terms of interpretability and the meaning of the classes (Collins & Lanza, 2010).

There are two types of model fit to be mentioned on LCA analyses: (a) absolute model fit, and (b) relative model fit. Absolute model fit is used to evaluate a latent model in the sense of representing the data sufficiently. Only one model is evaluated regarding the model fit under the theory of absolute model fit. One of the commonly used absolute model fit statistic in LCA is the Agresti's (1990) likelihood-ratio statistic G^2.

$$G^2 = 2\sum_{w=1}^{W} f_w \log\left(\frac{f_w}{\hat{f}_w}\right) \tag{2-18}$$

where W is the total number of all the possible response patterns which is calculated by multiplication of the response options of all the items, f_w is the observed frequency of cell w , and $\hat{f}_w$ is the expected frequency of cell w . Significance of G^2 statistic suggests that the population needs to be represented by more classes than specified in the model. Thus, a model with a higher number of classes is supposed to fit the population of data.

Relative model fit is used to compare two or more models regarding their representation of the data set. The likelihood-ratio difference test is one of the relative model fit indices as long as the models are nested, and one model is a restricted version of the other model. For example, through the likelihood-ratio difference test, a four class model could only be compared with another four class model estimating different number of parameters in LCA. The test is calculated as follows:

$$G_{\Delta}^2 = G_{\text{model}2}^2 - G_{\text{model}1}^2 \tag{2-19}$$

where the model 2 is the simpler model than the model 2. A significant G_{Δ}^2 proves that more complex model fits the data better.

Another relative model fit index is the information criteria. The information criteria indices could be applied for nested or not-nested models. One advantage of the information criteria over the likelihood-ratio difference test is their usefulness for not-nested models. For example, in order to compare a four class model with a three class model in LCA, we cannot use the likelihood-ratio difference test, because the models are not nested in this case. Two of the commonly used information criteria are the Akaike information criterion (AIC; Akaike, 1987), and Bayesian information criterion (BIC; Schwartz, 1978).

$$AIC = G^2 + 2P \tag{2-20}$$

$$BIC = G^2 + [\log(N)]P \tag{2-21}$$

where P is the number of parameters estimated in the model, and N is the sample size. The smaller values for AIC and BIC indicate better fit. In cognitive diagnosis models, determination of the number of classes is not needed, because the q-matrix determines the classes. CDMs are not exploratory models like most applications of LCA. CDMs are more similar to confirmatory latent class analysis models (Finch & Bronk, 2011).

Latent Transition Analysis (LTA) Model

The latent transition analysis model, also known as Mixture Latent Markov Models (Collins & Wugalter, 1992; Hagenaars, 1990; Poulsen, 1982; Van de Pol & Langeheine, 1990), is a latent variable model similar to LCA model. However, while LCA is used with cross-sectional data, the LTA is used with longitudinal data to investigate whether any change occurred in between latent classes across time. The latent transition model with only two time points is:

$$P(Y = y) = \sum_{s_1=1}^{S}\sum_{s_2=1}^{S}\delta_{s_1}\tau_{s_2|s_1}\prod_{t=1}^{2}\prod_{j=1}^{J}\prod_{r_{j,t}=1}^{R_j}\rho_{j,r_{j,t}|s_t}^{I(y_{j,t}=r_{j,t})} \tag{2-22}$$

where $I(y_j = r_j)$ is an indicator function that is equal to 1 when the response is r_j, and equal to 0 otherwise. LTA models estimate three different sets of parameters: (a) δ_{s_t} latent status prevalences, (b) $\rho_{j,r_{j,t}|s_t}$ item-response probabilities, and (c) $\tau_{s_{t+1}|s_t}$ transition probabilities (Collins & Lanza, 2010). Latent status prevalences and item response probabilities are also estimated by LCA models.

Latent Status Prevalences

Latent status prevalences are the counterparts of latent class prevalences in LCA model. Latent status prevalences are estimated for each time point, and one value is estimated for each latent status. Thus, for example, for a model with two time points and four estimated latent statuses, eight prevalences are estimated in total as showing the proportion of people on each latent status. For each time point, latent status prevalences across the latent statuses sum to 1. Hence, by looking at latent status prevalences, we can see increases and decreases on prevalences between time points, but we cannot see patterns of change and how individuals moved between statuses. Based on Collins and Lanza's (2010) notation, δ_{s_t} stands for the probability of membership in latent status *s* at time *t*. Each individual at each time point is a member of only one latent status. This rule comes from the fact that each latent status is mutually exclusive and extensive at each time point.

$$\sum_{s_t=1}^{S} \delta_{s_t} = 1 \tag{2-23}$$

where *S* is number of latent statuses, *T* is number of time points, and δ represent latent status prevalences.

Item-Response Probabilities

Item response probabilities represent probabilities of the individuals being in each latent status who gave the correct or a particular response to each item. In other words, item response probabilities help us label the latent statuses, based on the most common answers to the items we apply. There is an item response probability for each item-status combination. Thus, in total, there are number of items X number of latent statuses cells in total in an item-response probabilities table. The probability estimates of each response alternative to each item sum to 1 for each individual.

$$\sum_{r_{j,t}=1}^{R_j} \rho_{j,r_{j,t}|s_t} = 1 \qquad (2\text{-}24)$$

where J is number of items, $r_{j,t}$ is response category, R_j is number of response categories, s_t is latent status at time t, and ρ is item response probability.

For the sake of model identification, and interpretability of estimates, a restriction is enforced to item-response probability estimation in LTA. Item-response probabilities are assumed to be equal across different time points, which confirms that the meaning of the latent statuses for different time points remains unchanged.

Transition Probabilities

Transition probabilities present the changes that take place between the latent statuses across different time points. Transition probabilities are critical for the current study, because they answer whether any change occurs in between latent classes across time. For a model with two time points, a transition probability table consists of a cross-classification of the number of latent statuses in time 1 by the number of latent statuses in time 2. Given a certain number of time points *T*, a LTA analysis estimates *T*-1 transition probability matrices. For each individual, latent status membership is mutually exclusive and exhausted. In other words, individuals can be a member of only one latent status at each time point, and each row of the transition probability matrix sums to 1.

$$\sum_{s_{t+1}=1}^{S} \tau_{s_{t+1}|s_t} = 1 \qquad (2\text{-}25)$$

where $\tau_{s_{t+1}|s_t}$ is the probability of a transition to latent status s at Time *t+1* conditional on membership in latent status s at Time *t*.

Assuming measurement invariance in LTA allows constraining item-response probabilities to be equal across times, which facilitates interpretation of the probabilities, the meaning of latent statuses will remain the same across different time points. Consequently, in LTA, a single item-response probability matrix is estimated for all the time points, and probabilities are interpreted for by assuming that the meaning of each latent status remains constant. It is also worth to mention that restriction of item-response probabilities simplifies the model estimation due to fewer parameters estimated, and reduces convergence issues (Collins & Lanza, 2010).

Proposed Models

A longitudinal cognitive diagnosis model is an adaptation of latent transition analysis to cognitive diagnosis. The main objective is to propose longitudinal DINA and DINO models to classify examinees with respect to mastery of attributes at multiple measurement waves, and follow individual stability or change in mastery of attributes across time. The proposed models classify examinees into latent statuses through consecutive measurements, and estimate interclass transition probabilities of examinees from time 1 to time 2.

The longitudinal DINA model estimates the probability of obtaining a particular vector of responses as a function of the probabilities of membership in each cognitive class at time 1 (δ_{c_1}), the probabilities of transitioning to a cognitive class at time 2 on cognitive class membership at time 1 ($\tau_{c_2|c_1}$), and the probabilities of observing each response at each time point conditional on cognitive class membership *c* which integrates the DINA model into the equation and combines slipping (s_i) and guessing (g_i) parameters of item *i* as follows.

$$P(Y=y)=\sum_{c_1=1}^{C}\sum_{c_2=1}^{C}\delta_{c_1}\tau_{c_2|c_1}\prod_{t=1}^{2}\prod_{i=1}^{I}\prod_{r_{i,t}=1}^{R_i}[(1-s_i)^{\xi_{ic_t}}g_i^{(1-\xi_{ic_t})}]^{\mathrm{I}(y_{i,t}=r_{i,t})} \tag{2-26}$$

where R_i is the number of response categories of item *i*, s_i and g_i respectively represents slipping and guessing parameters of item *i*, and $\mathrm{I}(y_{j,t}=r_{j,t})$ is an indicator function which is equal to 1 when the response to item *i* at time *t* is r_j, and is equal to 0 otherwise. Another required component of the longitudinal DINA model is *latent response variable* ξ_{ic_t} that is defined for item *i,* and cognitive class *c* at time *t*. The equation of the latent response variable is shown below.

$$\xi_{ic_t}=\prod_{a=1}^{A}\alpha_{c_t a}^{q_{ia}} \tag{2-27}$$

where *A* is the total number of measured attributes, q_{ia} indicates whether attribute *a* is measured by item *i, and* $\alpha_{c_t a}$ identifies whether the respondent in latent class *c* mastered attribute *a*.

Similarly to the longitudinal DINA model, the longitudinal DINO model also estimates the probability of observing a particular vector of responses. The model is formulated as follows.

$$P(Y=y)=\sum_{c_1=1}^{C}\sum_{c_2=1}^{C}\delta_{c_1}\tau_{c_2|c_1}\prod_{t=1}^{2}\prod_{i=1}^{I}\prod_{r_{i,t}=1}^{R_i}[(1-s_i)^{\omega_{ic_t}}g_i^{(1-\omega_{ic_t})}]^{\mathrm{I}(y_{i,t}=r_{i,t})} \tag{2-28}$$

where ω_{ic_t} is the latent response variable of the longitudinal DINO model which differentiate it from the longitudinal DINA model. The latent response variable is defined as shown below.

$$\omega_{ic_t} = 1 - \prod_{a=1}^{A} (1 - \alpha_{c_t a})^{q_{ia}} \tag{2-29}$$

Similarly to the LTA, three types of parameters are produced as a result of longitudinal DINA and DINO model estimations: (a) item-response probabilities, (b) latent status prevalences, and (c) transition probabilities. In addition to these three parameters, posterior probabilities (EAP) are estimated to determine classes for each individual in each time point. For the practical use of longitudinal DINA and DINO models, the parameters of interest are posterior probabilities and transition probabilities. Posterior probabilities include individual information and are used to evaluate each student's performance based on their estimated class group. On the other hand, transition probabilities illustrate the probabilities of remaining in the same class or moving between classes from time 1 to time 2 for the respondents who are in the same cognitive class, which gives us an overall idea of the success of a treatment applied in between two measurements. The objective of this study is to evaluate the performance of the longitudinal DINA and the DINO models in conditions that are typical applications of cognitive diagnostic modeling. The conditions examined reflect manipulations of the correlation between attributes, sample size, and slipping and guessing parameters. The longitudinal DINA and DINO models are also compared across manipulated conditions.

CHAPTER 3
METHODS

Data Generation

To demonstrate the application of the proposed model, a simulation study was implemented with I = 30 items, A = 3 attributes, and N = 500 or 1000 examinees. In order to analyze attribute stability of the examinees between two following hypothetical data collection, two different data sets were simulated for the same examinees as representing the two time points. The data generation parameters for the two time points were simulated independently from each other. Thus, I expected to find similar patterns on the analysis results for the two time points. Any slight difference would be caused by the random variation of the estimated parameters. To be able to show the model on a simple simulation, I kept the number of attributes small, i.e. 3. The q-matrix was generated including items with one, two, and three attribute requirements. Table 3-1 presents the q-matrix for 30 items and 3 attributes, and table 3-2 presents all the possible simulated attribute profiles by showing the items sharing the same attribute profile. As shown in the both tables, 15 out of 30 items (50%) involve a single attribute, 11 items (37%) involve two attributes, and 4 items (13%) involve three attributes.

Two correlation conditions between the attribute pairs were developed. First, the attributes were assumed to be uncorrelated with a 0 correlation, and then a correlation of 0.5 was simulated between the attributes as representing a moderate correlation level. These correlation conditions were specifically selected due to their similar examples in the literature (Henson& Douglas, 2005; Henson, Templin& Douglas, 2007). The proportion of examinees mastering an attribute, p_k , was set to 0.5 for easiness.

Table 3-1. Simulated Q-matrix for 30 items and 3 attributes

Item	Attribute 1	Attribute 2	Attribute 3
1	0	0	1
2	0	1	1
3	0	1	1
4	0	0	1
5	0	0	1
6	1	0	0
7	1	0	1
8	1	0	0
9	0	1	1
10	1	1	0
11	1	0	1
12	0	1	0
13	1	1	1
14	1	1	1
15	1	0	0
16	1	1	0
17	0	1	0
18	0	1	0
19	0	0	1
20	0	1	0
21	1	0	0
22	1	1	1
23	0	0	1
24	0	1	1
25	1	0	1
26	1	1	0
27	1	0	1
28	1	1	0
29	1	0	0
30	1	1	1

1=measuring, 0 = not measuring

Number of possible attribute profiles are defined through the formula of $2^A = 2^3 =$ 8 as *A* representing the number of attributes. Of 8 possible attribute profiles, one is defined as 000, indicating that the item does not measure any of the present attributes. Thus, I did not include the profile of 000 in this simulation study. The other 7 possible profiles were included in the design. The item-to-profile table is presented below at table 3-2. I tried to equally distribute the number of items to each attribute profile.

Table 3-2. Simulated attribute profiles

Item number	Attribute 1	Attribute 2	Attribute 3	Total number of items	Set number
1, 4, 5, 19, 23	0	0	1	5	1
12, 17, 18, 20	0	1	0	4	2
2, 3, 9, 24	0	1	1	4	3
6, 8, 15, 21, 29	1	0	0	5	4
7, 11, 25, 27	1	0	1	4	5
10, 16, 26, 28	1	1	0	4	6
13, 14, 22, 30	1	1	1	4	7

Following previous studies (de la Torre & Douglas, 2004; de la Torre & Young-Sun, 2010; Liu, Xu & Ying, 2012), two sample sizes were simulated: 500 and 1000. True alphas, α_{ca}, which are the binary indicators of whether the examinee in latent class c mastered the measured attribute or not, were simulated from a probit model with a multivariate normal vector with mean of zero and correlation between the attributes at 0.5.

Finally, the slipping and the guessing parameters were varied to represent high and low discriminating items. Two different conditions were simulated for the slipping and guessing parameters: In the first condition, the slipping and guessing parameters were randomly assigned from a uniform distribution with a maximum of 0.2 and a minimum of 0. In the second condition the ranges of the slipping and guessing parameters were determined between 0.2 and 0.4. These conditions have been frequently simulated in previous studies (e.g., Henson& Douglas, 2005; Rupp& Templin, 2008; Liu, Douglas& Henson, 2009; de la Torre& Lee, 2010).

The examinee responses were generated based on longitudinal DINA and DINO models. DINA is the most commonly used and analyzed CDM model by the researchers (e.g., Henson & Douglas, 2005; de la Torre, 2008; von Davier, 2014). For comparison, DINO model was also used in my study as having a close model structure to DINA model. In order to simulate data under the DINA and the DINO models, the R.3.1.0 0 software and *CDM* package were used. The simulated datasets differed for the two models because the latent response variables for the DINA and the DINO models (ξ_{ic_t} and ω_{ic_t}, respectively) were distinctively simulated following the formulas below.

$$\xi_{ic_t} = \prod_{a=1}^{A} \alpha_{c_t a}^{q_{ia}} \tag{3-1}$$

$$\omega_{ic_t} = 1 - \prod_{a=1}^{A} (1 - \alpha_{c_t a})^{q_{ia}} \tag{3-2}$$

In order to investigate the mobility or stability of the student classes from time one to time two in a longitudinal simulation setting, measurements across two times points were simulated. Seven classes were defined in the model for both time points, and constraints applied to the model for each question. Mastery of an attribute was allowed to be acquired or lost over time. Because the data parameters were independently simulated in each time point, estimated transition probabilities were similarly distributed into a seven by seven transition matrix for every data set.

A thousand data sets were generated and analyzed for each condition. In total, 16 conditions were simulated based on a crossed design (2 x 2 x 2 x 2): the number of sample sizes (500 or 1000), the magnitude of correlation between the attributes (0 or 0.5), the size of slipping and guessing parameters (between 0 and 0.2, or 0.2 and 0.4), and the type of CDM (DINA or DINO).

Data Analysis

MPlus software was used to implement CDM analyses by modifying LCA and LTA applications of the software by following the recommendations of Rupp et.al. (2010). The simulated one thousand data sets per condition were analyzed, and the average of the correct classification rates (CCRs) and the marginal correct classification rates (MCCRs) were obtained over a thousand iterations, following the previous research (Li, Cohen, Bottge, & Templin, 2015; de la Torre, J., Hong, Y., & Deng, W., 2010; de la Torre, 2009a). The CCRs for each time point were calculated by comparing the true and the estimated classes of the examinees for a thousand iterations and dividing the number of correctly classified iterations by a thousand. Correct classifications of both time points together were also calculated. The MCCRs for each attribute were individually computed for each time point. MCCRs represent the average correct classification rates of each attribute for each condition over a thousand repeated measurements. The CCRs and the MCCRs for this study were calculated through the following formulas.

$$CCR = \frac{Number\ of\ correctly\ estimated\ classes}{Number\ of\ iterations} \tag{3-3}$$

$$MCCR = \frac{Number\ of\ correctly\ estimated\ classes\ for\ each\ attribute}{Number\ of\ iterations} \tag{3-4}$$

The DINA and the DINO models were compared in terms of the correct classification of the examinees to the latent classes. Finally, convergence rates of each condition over a thousand iterations were presented.

CHAPTER 4
RESULTS

In this chapter, I present results from the simulation study designed to evaluate the performances of the DINA and the DINO models to estimate diagnostic classes of the simulated examinees through repeated measurements. The results of the DINA and the DINO models are separately presented in two sections below.

Simulation Analysis Results for the DINA Model

Table 4-1 presents the marginal correct classification rates (MCCR) for each attribute in time 1 for the DINA model. The two slipping and guessing parameter conditions are coded as *1* representing small slipping and guessing values simulated from a normal distribution with a maximum of 0.2 and a minimum of 0, and *2* representing large slipping and guessing condition which is distributed with a maximum of 0.4 and a minimum of 0.2.

Table 4-1. Marginal correct classification rates for each attribute in time 1 for DINA model

Sample	S and G	Correlation	MCCR in attribute 1	MCCR in attribute 2	MCCR in attribute 3
500	1	0	0.997	0.995	0.997
500	1	0.5	0.997	0.995	0.997
1000	1	0	0.998	0.995	0.997
1000	1	0.5	0.998	0.995	0.997
500	2	0	0.874	0.855	0.873
500	2	0.5	0.893	0.877	0.891
1000	2	0	0.879	0.860	0.876
1000	2	0.5	0.896	0.883	0.896

According to table 4-1, the results indicated that sample size did not have an effect on the MCCRs of any of the attributes when the slipping and guessing parameters are small. Nonetheless, a slight increase of the MCCR was observed as the sample size increased from 500 to 1000 for the large sipping and guessing conditions. Even though I randomly simulated the attribute parameters, the second attribute's MCCR values were lower than both the first and the third attributes. This could only be explained by the random variation of the parameters. Table 4-1 also shows that higher slipping and guessing parameters produced lower MCCR values. While the MCCRs were as high as 0.998 for the first slipping and guessing condition, the highest values were around 0.896 for the second condition. For low slipping and guessing parameters, correlations between the attributes did not affect the MCCRs. However, for higher slipping and guessing parameters, correlation of 0.5 provided higher MCCRs than correlation of 0 did.

Since the parameters of time 1 and time 2 were randomly simulated, the MCCRs for the time 2, which is presented below in table 4-2, were similar to the MCCRs of the time 1. The CCR values were around 0.99 for high slipping and guessing parameters, and ranged from 0.856 to 0.895 for low slipping and guessing parameters.

Table 4-2. Marginal correct classification rates for each attribute in time 2 for DINA model

Sample	S and G	Correlation	MCCR in attribute 1	MCCR in attribute 2	MCCR in attribute 3
500	1	0	0.997	0.995	0.997
500	1	0.5	0.997	0.995	0.997
1000	1	0	0.997	0.995	0.998
1000	1	0.5	0.997	0.995	0.997
500	2	0	0.872	0.856	0.873
500	2	0.5	0.890	0.877	0.892
1000	2	0	0.876	0.859	0.877
1000	2	0.5	0.894	0.882	0.895

Table 4-3 presents the overall correct classification rates (CCR) of each time point, and the correct transition rates (CTR) of the individuals between these two time points. As expected, CCRs showed similar patterns for the two time points because of the random simulation of the parameters. For the low slipping and guessing parameters, obtained CCR values were over 0.98 in both time points, which represents nearly a perfect classification rate. On the other hand, CCRs were unacceptably low, ranging from 0.68 to 0.72, in the conditions where slipping and guessing parameters were between 0.2 and 0.4.

The effect of sample size on the classification rates interacted with the level of slipping and guessing parameters. When the slipping and guessing parameters were between 0 and 0.2, sample size did not show an effect on the CCRs. However, when I increased slipping and guessing parameters to the range of 0.2 to 0.4, I obtained higher CCRs for the data with 1000 sample size than for those with 500 samples. Likewise, an interaction was observed between the effect of correlation on the CCR values and the level of slipping and guessing parameters. For low slipping and guessing parameters, correlations between the attributes did not affect the correct classification of the examinees, however, as slipping and guessing parameters increased, correlation of 0.5 provided higher CCRs than no correlation did.

Table 4-3. Correct classification rates for the two time points and the correct transition rate between the two time points for DINA model

Sample	S and G	Correlation	CCR in time 1	CCR in time 2	CTR between time 1 and time 2
500	1	0	0.990	0.990	0.980
500	1	0.5	0.989	0.989	0.978
1000	1	0	0.990	0.990	0.980
1000	1	0.5	0.990	0.989	0.979
500	2	0	0.683	0.680	0.462
500	2	0.5	0.723	0.721	0.521
1000	2	0	0.692	0.690	0.476
1000	2	0.5	0.733	0.731	0.536

Correct transition rates (CTR) of the examinees from time 1 to time 2 followed a similar pattern as the CCR values of each time point, but the values were distinctively lower than CCRs, as expectedly.

Table 4-4 demonstrates the convergence rates of 1000 data sets. As seen in the table, the convergence rates of the DINA model were over 0.99 for all the conditions. None of the conditions of sample size, slipping and guessing parameters, and correlations made any difference on the convergence rate.

Table 4-4. Convergence rate out of 1000 iterations for each conditional combination for DINA model

Sample	S and G	Correlation	Convergence rate
500	1	0.5	0.997
500	1	0	0.992
500	2	0.5	0.990
500	2	0	0.990
1000	1	0.5	0.997
1000	1	0	0.999
1000	2	0.5	0.996
1000	2	0	0.994

Simulation Analysis Results for the DINO Model

The simulation results for the DINO model were similar to the DINA model. The correct classification rates (CCR) and the marginal correct classification rates (MCCR) were between 0.861 and 0.998. Table 4-5 illustrates the marginal correct classification rates for each attribute in time 1 for the DINO model.

According to table 4-5, the MCCRs of the DINO model for time 1 were at or over the level of 0.85 for each condition. Because each attribute was simulated randomly, as expectedly the results did not differ between the three attributes. Overall results indicated that the effect of sample size interacted with the magnitude of slipping and guessing parameters. This finding corresponds to the results that are obtained from the DINA model.

Table 4-5. Marginal correct classification rates for each attribute in time 1 for DINO model

Sample	S and G	Correlation	MCCR in attribute 1	MCCR in attribute 2	MCCR in attribute 3
500	1	0	0.998	0.995	0.997
500	1	0.5	0.997	0.995	0.997
1000	1	0	0.998	0.995	0.997
1000	1	0.5	0.997	0.995	0.997
500	2	0	0.875	0.857	0.873
500	2	0.5	0.894	0.879	0.892
1000	2	0	0.879	0.861	0.877
1000	2	0.5	0.897	0.882	0.894

For low slipping and guessing condition, MCCRs did not differ between 500 and 1000 sample sizes. However, as the slipping and guessing parameters increased, MCCRs showed slightly higher values for the larger sample sizes. Moreover, the effect of the correlation between the attributes was observed for larger slipping and guessing conditions, but not for the smaller ones. For the conditions with high slipping and guessing parameters, the correlation of 0 between the attributes provided higher MCCR values than the correlation of 0.5 did.

Table 4-6. Marginal correct classification rates for each attribute in time 2 for DINO model

Sample	S and G	Correlation	MCCR in attribute 1	MCCR in attribute 2	MCCR in attribute 3
500	1	0	0.998	0.995	0.997
500	1	0.5	0.997	0.995	0.997
1000	1	0	0.998	0.995	0.998
1000	1	0.5	0.997	0.995	0.997
500	2	0	0.873	0.854	0.873
500	2	0.5	0.892	0.877	0.892
1000	2	0	0.879	0.860	0.878
1000	2	0.5	0.896	0.881	0.895

Table 4-6 presents MCCRs for each attribute in time 2 for the DINO model. The results for time 2 were slightly different from the results of time 1, but still showed similar patterns. Because the parameters were randomly and independently simulated for each time point, this difference could only be explained by randomness. The MCCRs for time 2 were at high levels for the all attributes. The results did not vary between the three attributes, as expected, and between the two sample sizes. In terms of the correlation between the attributes, the results were higher for the correlation of 0.5 than the results for the correlation of 0. The observed smallest MCCR was 0.854.

Table 4-7. Correct classification rates for the two time points and the correct transition rates between the two time points for DINO model

Sample	S and G	Correlation	CCR in time 1	CCR in time 2	CTR between time 1 and time 2
500	1	0	0.990	0.991	0.981
500	1	0.5	0.989	0.990	0.979
1000	1	0	0.990	0.991	0.981
1000	1	0.5	0.990	0.990	0.980
500	2	0	0.685	0.681	0.465
500	2	0.5	0.725	0.722	0.524
1000	2	0	0.693	0.693	0.480
1000	2	0.5	0.732	0.731	0.535

Table 4-7 shows CCRs for the two time points, and CTRs between these two time points for the DINO model. The CCRs of both time points, and the CTRs between these points were similar to the DINA model results. Even though the CCRs for the DINO model differed at some conditions, the results followed the same pattern and were roughly similar for the two time points. Of the other conditions, even though sample size did not show a visible effect on the CCRs, correlation between the attitudes positively affected the CCRs when slipping and guessing parameters were relatively high, which corresponds to the findings of the DINA model presented before. Data sets with 0.5 correlation between the attitudes produced higher CCRs than the data sets with 0 correlation. This effect was not observed when the slipping and the guessing parameters were small. Finally, the CTRs between time 1 and time 2 for the DINO model were equal to or higher than 0.98 level when slipping and guessing parameters were small. However, a significant reduction was observed for the conditions with low slipping and guessing parameters. The values ranged from 0.732 to 0.465, which indicated a poor classification.

Table 4-8. Convergence rates out of 1000 iterations for each conditional combination for DINO model

Sample	S and G	Correlation	Convergence rate
500	1	0.5	0.984
500	1	0	0.982
500	2	0.5	0.973
500	2	0	0.934
1000	1	0.5	0.983
1000	1	0	0.986
1000	2	0.5	0.990
1000	2	0	0.991

Table 4-8 illustrates the convergence rates for each condition out of 1000 iterations for DINO model. Of 1000 iterations, the convergence rate for the DINO model was over 0.93 for all the conditions.

CHAPTER 5
CONCLUSION

Discussion

The use of the proposed longitudinal cognitive diagnosis model with the DINA and the DINO models was evaluated in this study. The results indicated that the use of longitudinal cognitive diagnosis model on the DINA and the DINO model under the simulated conditions provided accurate classifications. Effectiveness of the model was similar under the condition of DINA and DINO modeling. Because of the similarities, in this section, the effective use of the proposed model on the DINA and the DINO models will be evaluated together, and an overall conclusion will be provided.

The study results suggest that size of slipping and guessing parameters is an important factor on providing correct classifications of individuals to diagnostic groups in longitudinal cognitive diagnosis. As the slipping and guessing parameters decrease, correct classification rates visibly increase. The findings of this study ensure that size of slipping and guessing parameters would be problematic for longitudinal DINA and DINO models when they are larger than 0.2. In this study, I aimed to examine the realistic levels of the slipping and guessing conditions by including the values at a range 0 to 0.4. Levels higher than 0.4 are rarely encountered and not desired in real applications. Furthermore, the effects of the sample size and the correlation between the attributes on the correct classification of the individuals to the sub cognitive groups are also affected by the size of the slipping and guessing parameters. When slipping and guessing parameters are low, sample size or correlation between attributes does not have an effect on the classification rates. However, the effects of correlation and sample size appear as slipping and guessing parameters increase.

Correlation between attributes indicates that the attributes are associated with each other, and the mastery of an attribute provides information about the other associated attribute as well. The study findings show that, under small slipping and guessing conditions, existence of a correlation between attributes provides better classification accuracy. This finding agrees with Henson, Templin, and Douglas's (2007) findings from another cognitive diagnosis study. In their study, researchers argue that as the correlation increases, mastery of an attribute implies a greater chance of mastery of the other attributes that are correlated with this attribute. In this study, I only evaluated the effects of a zero and a moderate correlation in order to demonstrate the overall effects of the association between the attributes. The study demonstrated that the existence of a correlation between the attributes helps improve correct classification rates of examinees to the diagnostic groups, however, further research is needed in order to define the smallest problematic correlation level.

For the conditions with high slipping and guessing parameters, sample size of the examinees has a clear effect on the accuracy of classification on the longitudinal cognitive diagnosis. As indicated in the results section, for groups with higher sample sizes, chances of classifying the examinees to correct classes increase. Moreover, existence of a correlation between attributes provides higher correct classification rates than lack of correlation, when slipping and guessing parameters are fairly high. Torre and Douglas (2004) state that high slipping and guessing parameters are indicators of poor fit. High slipping and guessing parameters are caused either by incorrect identification of attributes, or misspecification of the q-matrix. For correct classification and not to obtain misleading results, researchers are recommended to make sure that the attributes and the q matrices are correctly identified by the experts before deciding on the application of a longitudinal cognitive diagnosis model.

The overall conclusion is that the correct classification rates of the longitudinal DINA and the longitudinal DINO models do not differ significantly. Both models work efficiently to classify individuals to cognitive groups when multiple examinations occurred. These are not competing models, but they are appropriate for different situations. In general, the DINA is a model works with a conjunctive condensation rule, and lack of one attribute tremendously drops the probability of responding an item correctly. However, the DINO model works with a disjunctive condensation rule in which the presence of at least one measured attribute guarantees a high probability of endorsing an item. The same rules apply to the longitudinal DINA and DINO models. Thus, researchers' decisions on which model to apply on their data should depend on the nature of the required diagnosis. For instance, if not possessing even only one of the required skills will be enough to identify need of practical significance for intervention, the longitudinal DINA model would be the appropriate model for the analysis.

In a recent study, Li, Cohen, Bottge, and Templin (2015) illustrated the use of latent transition analysis for assessing change in cognitive diagnosis through DINA model on a real data application, and their findings aligned with the results of current study. Their study results supported the effective use of the DINA model with latent transition analysis in educational research. The authors discussed the potential benefits of using the proposed model, so called LTA-DINA, and they claimed that the model is capable for testing different hypotheses about the transition probabilities for certain groups. Additionally, it was stated that the model provided rich information about treatment effects at both individual and group level.

My recommendation to the researchers would be to use the longitudinal cognitive diagnosis models in the cases when small slipping and guessing parameters are predicted. My findings support the use of longitudinal cognitive diagnosis models when the maximum slipping and guessing parameters are expected not to be larger than 0.2. Otherwise, sample size should be relatively large, and correlated attributes are needed. To be able to obtain sufficiently correct classification rates, sample size is recommended not to be smaller than 500 when slipping and guessing parameters are predicted to be larger than 0.2. The study results also show that any existing correlation between attributes would promote correct classification rates of a study

Limitations and Future Research

Some limitations exist in the design and application of this simulation study. First, time 2 data was simulated independent of time 1 data. In a real case scenario in which achievement level is measured, latent classes can be ordered from low achieving level to high achieving level, and students can be allowed to move only to a limited number of classes below and above their initial class level, with the assumption that learning levels of students are not likely to show a tremendous change between the lowest and the highest learning levels across two measurement points. Furthermore, because of the coding and labeling complexity of Mplus program, only three attributes were simulated in this study. In real life scenarios, number of attributes is expected to be higher in some cases. Data with higher number of attributes needs to be evaluated in the future studies. Moreover, simulation can be expanded to more than two time points. In that case, investigating methods and power to detect treatment effects on change in mastery over time can be possible. Finally, the effect of q-matrix and model misspecifications on the longitudinal DINA and the longitudinal DINO models are areas of interest I can suggest for the future research. Additionally, this research can be easily extended to the NC-RUM model.

APPENDIX A

R DATA SIMULATION CODE FOR THE DINA MODEL

```
### Function to simulate DINA data ###

  # create the dina.data.sim function to simulate data
dina.data.sim = function(s.g, corr, items, N, attr,Q)   {
   # download MASS package
        require(MASS)
   # generating the examinees #
   # conditions for slip and guessing parameters. "1" for 0-.2, "2" for .2-.4
        if (s.g==1) {
                min = 0
                max = 0.2
        s.items = runif( items, min, max)
        g.items = runif( items, min, max)
                 }   else   {
                min = 0.2
                max = 0.4
         s.items = runif( items, min, max)
        g.items = runif( items, min, max)   }
   # Alphas are simulated from a probit model with a multivariate normal vector
with
   # mean vector zero and correlations between the skills fixed at .5 (sd = 1)
        sigma = matrix(c(1, corr, corr,corr, 1, corr, corr, corr, 1), 3,3)
        alpha.c1 = as.matrix(mvrnorm(N, mu=rep (0, 3), sigma))  #alpha for time 1
        alpha.c2 = as.matrix(mvrnorm(N, mu=rep (0, 3), sigma))  #alpha for time 2
   # elements of alpha.c are changed to 1 if they are greater than or equal to zk
otherwise they are changed to 0
   # create dichotomous mastery indicators
        alpha.bin1 = ifelse(alpha.c1 >= 0, 1, 0)  #binary alpha for time1
        alpha.bin2 = ifelse(alpha.c2 >= 0, 1, 0)  #binary alpha for time2
   ## generating the item response with the DINA using all the parameters
simulated above ##
  ## for time 1 ##
        xsi1 = matrix(1, nrow= N, ncol= items)
        resp.prob.dina1 = matrix(1, nrow= N, ncol= items)
        for (g in 1:nrow(alpha.bin1)) { # persons binary abilities
        for (h in 1:nrow(Q)) { # the Q-matrix
        for ( a in 1:ncol(Q)) {
                xsi1[g,h]= xsi1[g,h] * (alpha.bin1[g,a] ^ Q[h,a])      # Formula of xsi
is used here
                }
                resp.prob.dina1[g,h] = ((1-s.items[h])^xsi1[g,h])*((g.items[h])^(1-
xsi1[g,h]))
                }}
  # generating a random u matrix of random numbers between 0 and 1
        u1 = matrix((runif(N*items,0,1)), N,items)
```

```
  # Dichotomizing the responses #
        resp.bin.dina1 = ifelse(resp.prob.dina1 >= u1, 1,0)

   ## for time 2 ##
        xsi2 = matrix(1, nrow= N, ncol= items)
        resp.prob.dina2 = matrix(1, nrow= N, ncol= items)
        for (g in 1:nrow(alpha.bin2)) { # persons binary abilities
        for (h in 1:nrow(Q)) { # the Q-matrix
        for ( a in 1:ncol(Q)) {
                xsi2[g,h]= xsi2[g,h] * (alpha.bin2[g,a] ^ Q[h,a])      # Formula of xsi
is used here
                }
                resp.prob.dina2[g,h] = ((1-s.items[h])^xsi2[g,h])*((g.items[h])^(1-
xsi2[g,h]))
                }}
    # generating a random u matrix of random numbers between 0 and 1
        u2 = matrix((runif(N*items,0,1)), N,items)
     # Dichotomizing the responses #
        resp.bin.dina2 = ifelse(resp.prob.dina2 >= u2, 1,0)
  id = c(1:N)
        resp.bin.dina = data.frame(cbind(id, resp.bin.dina1, resp.bin.dina2))
     # save data.dina
        write.table(resp.bin.dina, file="data_dina.CSV", sep=",", quote = F,
row.names = F, col.names = F)
  # save alphas for time 1 and time 2
        alphas = data.frame(cbind(id, alpha.bin1, alpha.bin2))
        return(alphas)
        }       # close function
```

The same parameter simulation pattern was followed for the DINO model. However, the response probabilities were simulated based on the omega parameter, instead of xsi. A simple application of the omega formula is presented below.

```
omega <- matrix(1, nrow= N, ncol= items)
resp.prob.dino <- matrix(1, nrow= N, ncol= items)
for (g in 1: nrow(alpha.bin)) { # persons binary abilities
for (h in 1: nrow(Q)) { # the Q-matrix
                omega[g,h]= 1- (omega[g,h] * ((1-alpha.bin[g,1]) ^ Q[h,1])*((1-
alpha.bin[g,2])^Q[h,2])*((1-alpha.bin[g,3]) ^ Q[h,3]))
                resp.prob.dino[g,h] = (((1-s.items[h])^omega[g,h])*((g.items[h])^(1-
omega[g,h]))))}}
```

APPENDIX B
R DATA ANALYZING CODE

```
## Function to run the dina data ##

#   conditions = c(s.g, corr)
# i is the iteration number, used for records only
        dino.mplus.run = function(DINO.data,mplusinp, mplusout, conditions,i)  {
 # read the intput file
        inputfile = paste(mplusinp,".inp",sep="")
 # Run Mplus
        system(paste("mplus.exe", inputfile, sep=" "), wait = TRUE,
show.output.on.console = F)
 # read the mplus output to verify the outcome of the analysis
        respondent = paste(mplusout,".dat",sep="")
 # read estimated individual classes
        respondents = try(read.table(respondent))
 # Conditional statement to run the model only if all models converged
        if (class(respondents)!="try-error") {
 # read true alphas
        alpha1 = DINO.data[,2:4]
        alpha2 = DINO.data[,5:7]
  # function to call the class numbers for each attribute profile
        BinToDec = function(x)  {
        sum(2^(which(rev(unlist(strsplit(as.character(x), "")) == 1))-1))}
  # create an empty object
        alpha.bin1.class = vector()
        alpha.bin2.class = vector()
  # create a vector with the classes of the individuals
        for (j in 1:N) {    # loop through iterations
                alpha.bin1.class[[j]] = BinToDec(x=alpha1[j,]) +1}
        for (j in 1:N) {    # loop through iterations
                alpha.bin2.class[[j]] = BinToDec(x=alpha2[j,]) +1}

    ##### CCR #####

    # extract the estimated classes in time 1 and time 2
        ctime1 = respondents[,126]
        ctime2 = respondents[,127]
    # compare the true and estimated classes and find the correct classification
rate
        time1 = alpha.bin1.class==ctime1
        time2 = alpha.bin2.class==ctime2
        ccrtime1 = mean(time1)
        ccrtime2 = mean(time2)


    # compare the true and estimated class differences between the two time
```

points and the correct classification rate

```
time.diff = (alpha.bin1.class==ctime1 & alpha.bin2.class==ctime2)
ccrtime.diff = mean(time.diff)

############ attribute CCR for time 1 ##############

# class1 = c(0, 0, 0)
# class2 = c(0, 0, 1)
# class3 = c(0, 1, 0)
# class4 = c(0, 1, 1)
# class5 = c(1, 0, 0)
# class6 = c(1, 0, 1)
# class7 = c(1, 1, 0)
# class8 = c(1, 1, 1)

# for classes 1 to 4 the value is 0
# for classes 5 to 8 the value is 1
require(car) #for "recode" function
# create three binary variables for three attributes indicating if the person mastered that attribute or not
#variables for the estimated classes
c1att1 = recode(ctime1, "c(1,2,3,4)='0'; else='1'")
c1att2 = recode(ctime1, "c(1,2,5,6)='0'; else='1'")
c1att3 = recode(ctime1, "c(1,3,5,7)='0'; else='1'")
# variables for the true classes
alpha1att1 = recode(alpha.bin1.class, "c(1,2,3,4)='0'; else='1'")
alpha1att2 = recode(alpha.bin1.class, "c(1,2,5,6)='0'; else='1'")
alpha1att3 = recode(alpha.bin1.class, "c(1,3,5,7)='0'; else='1'")
# compare the true and estimated classes and find marginal attribute ccr
# for attribute 1
c1att1comp = c1att1==alpha1att1
c1att1ccr = mean(c1att1comp)
# for attribute 2
c1att2comp = c1att2==alpha1att2
c1att2ccr = mean(c1att2comp)
# for attribute 3
c1att3comp = c1att3==alpha1att3
c1att3ccr = mean(c1att3comp)

######attribute CCR for time 2##########

# create three binary variables for three attributes indicating if the person mastered that attribute or not
# variables for the estimated classes
c2att1 = recode(ctime2, "c(1,2,3,4)='0'; else='1'")
c2att2 = recode(ctime2, "c(1,2,5,6)='0'; else='1'")
c2att3 = recode(ctime2, "c(1,3,5,7)='0'; else='1'")
# variables for the true classes
```

```
        alpha2att1 = recode(alpha.bin2.class, "c(1,2,3,4)='0'; else='1'")
        alpha2att2 = recode(alpha.bin2.class, "c(1,2,5,6)='0'; else='1'")
        alpha2att3 = recode(alpha.bin2.class, "c(1,3,5,7)='0'; else='1'")
# compare the true and estimated classes and find marginal attribute ccr
    # for attribute 1
        c2att1comp = c2att1==alpha2att1
        c2att1ccr = mean(c2att1comp)
    # for attribute 2
        c2att2comp = c2att2==alpha2att2
        c2att2ccr = mean(c2att2comp)
    # for attribute 3
        c2att3comp = c2att3==alpha2att3
        c2att3ccr = mean(c2att3comp)
        } else { ccrtime1 = NA
        ccrtime2 = NA
        ccrtime.diff = NA
        c1att1ccr = NA
        c1att2ccr = NA
        c1att3ccr = NA
        c2att1ccr = NA
         c2att2ccr = NA
         c2att3ccr = NA }   # close try function
# organize the data into a table
 results = data.frame(cbind(ccrtime1, ccrtime2, ccrtime.diff, c1att1ccr, c1att2ccr,
c1att3ccr,
            c2att1ccr, c2att2ccr, c2att3ccr))
# add indicators of conditions
        condition.descriptors = c(i,conditions)
        condition.descriptors =
        matrix(condition.descriptors,dim(results)[1],length(condition.descriptors),b
        yrow=T)
        results = data.frame(condition.descriptors, results)
# save results
        write.table(results, file="all_results.csv" , sep=",", row.names = F,
col.names = F, append = T)
}     # close the dina.mplus.run function
```

LIST OF REFERENCES

Agresti, A. (1990). *Categorical Data Analysis* (1st ed.) New York: Wiley.

Andersen, E.B. (1985). Estimating latent correlations between repeated testings. *Psychometrika*, *50*, 3–16.

Chiu, C. Y. (2013). Statistical refinement of the q-matrix in Cognitive Diagnosis. *Applied Psychological Measurement, 37*, 598–618.

Close, C. N. (2012). An exploratory technique for finding the q-matrix for the DINA Model in Cognitive Diagnostic Assessment: Combining theory with data, unpublished doctoral dissertation, University of Minnesota, Minneapolis, MN.

Collins, L.M. & Lanza, S.T. (2010). *Latent class and Latent transition analysis with applications in the social, behavioral, and health sciences*. Hoboken, NJ: John Wiley & Sons Inc.

Collins, L.M., & Wugalter, S.E. (1992). Latent class models for stage-sequential dynamic latent variables. *Multivariate Behavioral Research, 27,* 131–157.

DeCarlo, L. T. (2011). On the analysis of fraction subtraction data: the DINA model, classification, class sizes, and the Q-matrix. *Applied Psychological Measurement, 35*, 8–26.

de la Torre, J., & Douglas, J. A. (2004). Higher-order latent trait models for cognitive diagnosis. *Psychometrika, 69*, 333–353.

de la Torre, J. (2008). An empirically-based method of q-matrix validation for the DINA model: Development and applications. *Journal of Educational Measurement 45*, 343–362.

de la Torre, J. (2009a). A cognitive diagnosis model for cognitively based multiple-choice options. *Applied Psychological Measurement, 33*, 163–183.

de la Torre, J. (2009b). DINA model and parameter estimation: A didactic. *Journal of Educational and Behavioral Statistics, 34*, 115–130.

de la Torre, J., Hong, Y., & Deng, W. (2010). Factors affecting the item parameter estimation and classification accuracy of the DINA model. *Journal of Educational Measurement, 47*, 227–249.

de la Torre, J., & Lee, Y. S. (2010). A note on the invariance of the DINA model parameters. *Journal of Educational Measurement, 47*, 115–127.

de la Torre J., & Lee Y. S. (2013). Evaluating the Wald test for item-level comparison of saturated and reduced models in cognitive diagnosis. *Journal of Educational Measurement, 50,* 355–373.

DiBello, L. V., & Stout, W. (2007). Guest editors' introduction and overview: IRT-based cognitive diagnostic models and related methods. *Journal of Educational Measurement, 44*, 285–291.

DiBello, L. V., Stout, W., & Ruossos, L. A. (1995). Unified cognitive/psychometric diagnostic assessment likelihood-based classification techniques. In: Nichols, P. D., Chipman, S. F., Brennan, R. L. (Eds.), *Cognitively Diagnostic Assessment (*pp.361–389). Hillsdale, NJ: Erlbaum.

Embretson, S.E. (1991). A multidimensional latent trait model for measuring learning and change. *Psychometrika, 56*, 495–515.

Finch, W. H., & Bronk, K. C. (2011). Conducting Confirmatory Latent Class Analysis Using Mplus. *Structural Equation Modeling, 18*, 132–151.

Haertel, E. H. (1989). Using restricted latent class models to map the skill structure of achievement items. *Journal of Educational Measurement, 26,* 333–352.

Hagenaars, J.A. (1990). *Categorical Longitudinal Data - Loglinear Analysis of Panel, Trend and Cohort Data*. Newbury Park: Sage.

Hartz, S. M. (2002). A Bayesian framework for the Unified Model for assessing cognitive abilities: Blending theory with practicality. Unpublished doctoral dissertation, University of Illinois, Champaign, IL.

Henson, R., & Douglas, J. (2005). Test construction for cognitive diagnosis. *Applied Psychological Measurement, 29*, 262–277.

Henson, R., Templin, J., & Douglas, J. (2007). Using efficient model based sum-scores for conducting skills diagnoses. *Journal of Educational Measurement, 44*, 361–376.

Junker, B. W., & Sijtsma, K. (2001). Cognitive assessment models with few assumptions, and connections with nonparametric item response theory. *Applied Psychological Measurement, 25*, 258–272.

Li, F., Cohen, A., Bottge, B., & Templin, J. (2015). A latent transition analysis model for assessing change in cognitive skills. Educational and Psychological Measurement 0013164415591848, first published on June 29, 2015 as doi:10.1177/0013164415591848

Liu, Y., Douglas, J. A., & Henson, R. A. (2009). Testing person fit in cognitive diagnosis. *Applied Psychological Measurement, 33*, 579–598.

Liu, J., Xu, G., & Ying, Z. (2012). Data-driven learning of q-matrix. *Applied Psychological Measurement*. *36*, 548–564.

Macready, G.B., & Dayton, C.M. (1977). Use of probabilistic models in the assessment of mastery. *Journal of Educational Statistics*, *2*, 99–120.

Maris, E. (1999). Estimating multiple classification latent class models. *Psychometrika 64*, 187–212.

Muthen, L. K., & Muthen, B. (2012). Mplus user's guide (7th ed.). Los Angeles, CA: Muthen & Muthen.

Poulsen, C. A. (1982). *Latent Structure Analysis with Choice Modeling Applications.* Aarhus: Aarhus School of Business Administration and Economics.

R Development Core Team. (2012). R: A language and environment for statistical computing. Vienna, Austria: R Foundation for Statistical Computing.

Rupp, A., & Templin, J. (2008a). The effects of q-matrix misspecification on parameter estimates and misclassification rates in the DINA model. *Educational and Psychological Measurement, 68*, 78–98.

Rupp, A., Templin, J., & Henson, R. (2010). *Diagnostic Measurement: Theory, Methods, and Applications*, New York: Guilford.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics 6:* 461–464.

Tatsuoka, K. K. (1983). Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement, 20*, 345–54.

Tatsuoka, K. K. (1990). Toward an integration of item-response theory and cognitive error diagnosis. In N. Frederiksen, R. Glaser, A. Lesgold, &M. Safto (Eds.). Monitoring attribute vector and knowledge acquisition (pp. 453–488). Hillsdale, NJ: Erlbaum.

Templin, J. (2004). Generalized Linear Mixed Proficiency Models for Cognitive Diagnosis, unpublished doctoral dissertation, University of Illinois at Urbana-Champaign, Urbana-Champaign, IL.

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

Templin, J. L., Henson, R. A., & Douglas, J. (2006). General theory and estimation of cognitive diagnosis models: Using Mplus to derive model estimates. Manuscript under review.

Van de Pol, F., & Langeheine, R. (1990). Mixed Markov latent class models. *Sociological Methodology, 20*, 213–247.

von Davier, M., & Yamamoto, K. (2004). A class of models for cognitive diagnosis. Paper presented at the ETS Spearmann Conference, The Inn at Penn, Philadelphia, PA, Invitational Conference, October.

von Davier, M. (2005). A general diagnostic model applied to language testing data (ETS Research Report No. RR-05-16). Princeton, NJ: ETS.

von Davier, M., & Xu, X. (2011). Measuring growth in a longitudinal large-scale assessment with a general latent variable model. *Psychometrica, 76,* 318–336.

von Davier, M. (2014). The DINA model as a constrained general diagnostic model: Two variants of a model equivalency. *British Journal of Mathematical and Statistical Psychology, 67*, 49-71

Xu, G. (2013). Statistical inference for Diagnostic Classification Models, unpublished doctoral dissertation, Columbia University, New York City, NY.

BIOGRAPHICAL SKETCH

Yasemin Kaya was born in 1982, in Turkey. She received the Bachelor of Science degree in elementary science education in 2006 from Pamukkale University, and a Master of Arts in education degree in research and evaluation methodology from the University of Florida in 2010. She received her Ph.D. from the University of Florida in the summer of 2015.