2007

User's Manual: WinGen2

Header: User's Manual: WinGen

User's Manual for *WinGen*:

Windows Software that Generates IRT Model Parameters and Item Responses[1]

Kyung T. Han and Ronald K. Hambleton

University of Massachusetts Amherst

---

[1] Center for Educational Assessment Research Report No. 642. Amherst, MA: University of Massachusetts, Center for Educational Assessment.

Contents

I. Introduction

Item response theory (IRT) is a popular and valuable framework for modeling educational and psychological test data, due to attractive properties such as the invariance of item and examinee parameter values (when IRT models can be found that fit the test data) and item parameters and examinee parameters being reported on a common scale. However, IRT models often rely on or are based on two strict assumptions, unidimensionality and the mathematical form of the item characteristic functions, and violations of these assumptions can produce serious negative consequences to the measurement process. One way of evaluating the impact of violating these two critical assumptions, as well as studying factors such as the impact of choice of models, examinee sample sizes, the shape of ability distributions, and test length, and many other factors, is via Monte Carlo simulation studies.

The essential part of Monte Carlo studies is simulating examinee item response data given true model parameter values. A number of computer programs for simulating IRT data have been developed since the early 1970s. However, most of them were developed in the DOS environment (e.g., DATAGEN, Hambleton & Rovinelli, 1973, GENIRV, Baker, 1989, and RESGEN, Muraki, 1992). As a result, these programs are limited today because of inherent problems in DOS: (1) slow performance speed (16-bit), (2) limited usable system resources, (3) incompatibility with recent 32-bit Windows-based OSs, and (4) not a user-friendly interface. Some recent computer simulation programs, such as WINIRT (Fang & Johanson, 2005) and PARDSIM (Yoes, 1997), were developed for Windows-based OSs, but they provide only a few options for generating data from IRT models. Therefore, the primary goal of this project was to develop computer software that could simulate IRT data using many IRT models and with many

important options to facilitate the design and implementation of computer simulation studies in the field of psychometric methods.

A computer program, called WinGen (Han, 2006) was developed to generate dichotomous and polytomous item response data for several IRT models and for many conditions that arise in practice. WinGen has the following useful features:

1. *WinGen supports various IRT models.*

There are several kinds of unidimensional and multidimensional IRT models that are widely used today: (1) dichotomous IRT models with one, two, and three parameters, (2) non-parametric models, (3) polytomous IRT models such as the partial credit model, generalized partial credit model, graded response model, rating scale model, and nominal response model, and, (4) multidimensional compensatory models. WinGen supports these IRT models and allows users to have a mixture of more than one IRT model in a set of items (for example, data for the first 10 items might be generated from the two-parameter logistic model, the next 30 items might be generated from the three-parameter logistic model, and with the last 10 items, data might be generated by the graded response model). Such a data set might be useful when simulating state test data, for example.

2. *WinGen generates IRT model parameter values from various distributions for realistic data.*

WinGen can generate sets of item parameters and sets of examinee ability parameters to essentially create realistic item response data from various kinds of distributions. Most programs that simulate IRT data only support normal and/or uniform distributions. Although uniform and normal distributions may be theoretically ideal or easy to understand, these distributions are not likely to be the only distributions seen in the real world or of interest to researchers. With

WinGen a user can choose a normal, uniform, or beta distribution (which can be defined to allow for the generation of skewed distributions, skewed from highly positive to highly negative) for the examinee parameter in a model, and a normal, uniform, beta, or lognormal distribution for item parameters so that the user can conduct a research study with more realistic IRT data sets, or even unrealistic parameters, if that is the interest of the researcher.. WinGen provides a graphic function of plotting a histogram of examinee ability parameters so that a user can make sure the distribution of the generated parameters is what he/she wanted. This graphing feature is especially helpful when the goal is to produce skewed distributions. Also, WinGen gives a user substantial control over item and examinee parameters by allowing a user to enter user specified parameter sets and/or to set SEED values for generating model parameters.

3. *WinGen provides an intuitive and user-friendly interface.*

By taking advantage of running on a Windows platform, WinGen provides a virtually point-and-click solution for simulating IRT data. There are three stages for simulating data: (1) generating or reading in ability parameter values, (2) generating or reading in item parameter values, and (3) simulating item response data. All three stages are shown on a one-stop interface screen, so a user can reduce the number of possible mistakes by being able to check the simulated data at each stage. Also, the intuitive interface of WinGen would be helpful for teaching IRT to those who are not familiar with the theory. Once item parameters and/or ability parameters are generated, WinGen immediately provides IRT plots such as item characteristic curves (ICC), test characteristic curves (TCC), item information curves (IFC), and test information curves (TIC) so that a user can make sure the generated parameters fit his/her research purposes. See Figure 1 for an example of the interface.

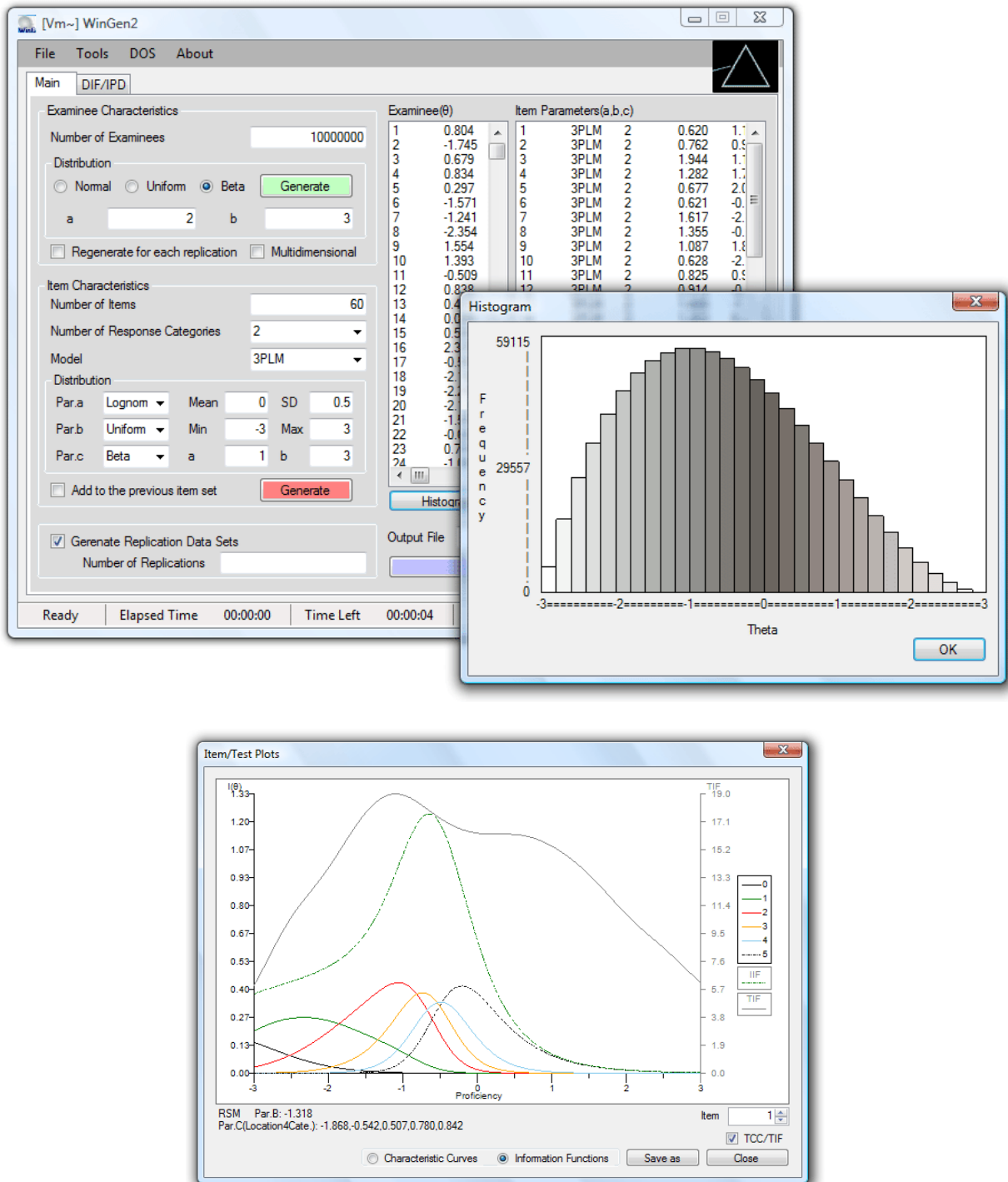Figure 1.1 *WinGen provides an intuitive and user-friendly interface.*

Figure 1.1 *WinGen provides an intuitive and user-friendly interface (continued).*
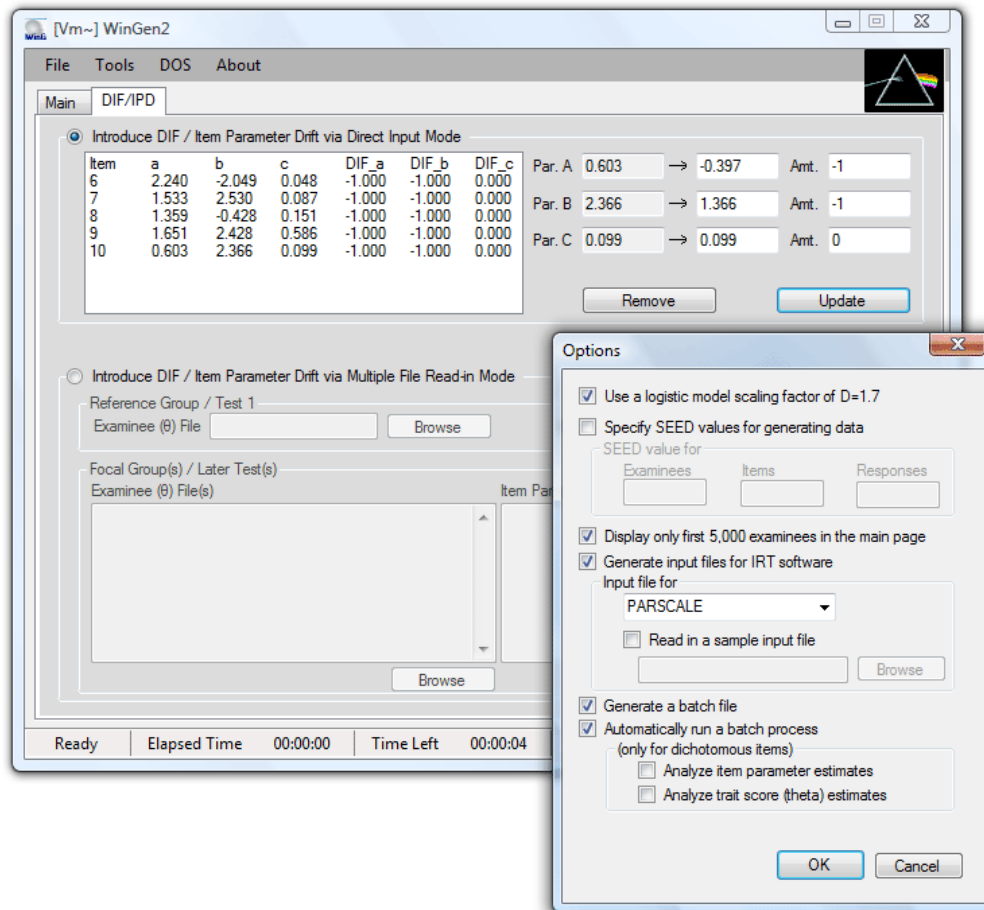
4. *WinGen is based on the most recent computer environment.*

WinGen was developed on *Microsoft .NET frameworks 2.0*, the most recent computer software development environment. The program can run on either 32bit Windows series (ex., Windows XP) or 64bit Windows series (ex., Windows vista) and is optimized to use system resources efficiently so that the program can handle a bigger data in a shorter time. This software can also easily be converted to web-based software.

5. *WinGen provides a powerful research tool.*

Several powerful tools are provided for various research purposes in WinGen. Replication data can be simulated up to 1,000,000 sets, and syntax files for other IRT programs, such as PARSCALE (Muraki & Bock, 2003), BILOG-MG (Zimowski, Muraki, Mislevy, & Bock, 2003), and MULTILOG (Thissen, 2003), are automatically generated (of course, a user can specify the syntax of the software by reading in a sample syntax file). Batch files can also be generated for handling multiple calibrations in a cue. WinGen provides a dialog input to introduce differential item functioning (DIF) or item parameter drift in the simulated data. With multiple file read-in option in WinGen, a user can have multiple groups of examinees and multiple sets of items/tests. WinGen can also read-in output files (parameter estimates) from other IRT programs and automatically analyze DIF by computing RMSD, MAD, and BIAS. Table 1 summarizes the features of WinGen and several other simulation software available.

Educational Importance

Although many IRT model simulation software programs have been developed, most of them were designed for intentionally narrow purposes since they were not publicly shared programs. WinGen is more general and more user friendly than many of the IRT simulation programs available today. WinGen is easy to use with an intuitive, user-friendly interface while providing very strong research tools and performance at the same time. Thus, WinGen should be useful for researchers who want to do simulation research, and students who are eager to learn more about IRT.

Table 1.1. *Comparison of IRT simulation programs.*

	DATAGEN	GENIRV	RESGEN	WINIRT	PARDSIM	WinGen2
Authors	Hambleton & Rovinelli(1973)	Baker(1989)	Muraki(1992)	Fang & Johanson(2005)	Yoes(1997)	Han (2007)
Examinees	4,000	4,000	1,000 (default)	4,000	N/A	100,000,000*
Items	400	100	100 (default)	400	N/A	100,000,000*
OS	DOS	DOS	DOS	Windows	Windows (16,32bit)	Windows(32,64bit)
IRT Models	1.Dichotomous	1.Dichotomous 2.Polytomous	1.Dichotomous 2.Polytomous 3.Multidimensional	1.Dichotomous	1.Dichotomous	1.Dichotomous 2.Polytomous 3.Nonparametric 4.Multidimensional
Distribution	1.Uniform 2.Normal	1.Uniform 2.Normal	1.Uniform 2.Normal 3.Lognormal 4.Gamma	1.Uniform 2.Normal	1.Uniform 2.Normal	1.Uniform 2.Normal 3.Lognormal 4.Two-Parameter Beta
Options	1.Replicating		1.Replicating 2.Multiple Blocks, subtests, groups			1.Replicating 2.Introducing DIF/Drift 3.Communicating with other IRT software 4.Analyzing outputs of other IRT software 5. Plotting ICF, TCC, IIF, TIF, and histogram of theta scores 6. Multiple tests, groups.

* the maximum number of examinees and items could be more or less than 100,000,000; it mainly depends on the available system resources of the computer on which WinGen runs.
Note: partially adopted from Harwell, Stone, Hsu, and Kirisci (1996, p.117)

II. Item Response Models Used within *WinGen*

Unidimensional IRT Models for Dichotomous Responses

Item response theory emerged as early as the 1940s though the popularity came much later in the 1970s. As can be recognized by the name, IRT models consider examinee behavior at the item level, not at the test level. Modeling at the item level creates much more flexibility for applications to test development, study of differential item functioning, computer-adaptive testing, score reporting, etc. Early IRT models were developed to handle dichotomous responses (i.e., binary responses; for example, 0 (incorrect) and 1 (correct)) but today, models are available to handle just about all types of educational and psychological data (see, van der Linden & Hambleton, 1997).

Two of the fundamental assumptions with IRT models are *unidimensionality* and *local independence*. The assumption of unidimensionality means that a set of items and/or a test measure(s) only one latent trait (θ), and local independence refers to the assumption that there is no statistical relationship between examinees' responses to the pairs of items in a test, once the primary trait measured by the test is removed. The two assumptions are really just different ways to say the same thing about the data. The third main assumption concerns the modeling of the relationship between the trait measured by the test and item responses. What follows are various models that make different assumptions about that relationship.

Normal Ogive Model

The normal ogive model was the first IRT model for measuring psychological and/or educational latent traits (Ferguson, 1942; Lawley, 1943; Mosier, 1940, 1941; Richardson, 1936). The normal ogive model was refined later by Lord and Novick (1968). In the model, an item characteristic curve (ICC) is derived from the cumulative density function (CDF) of a normal distribution. A mathematical expression of the normal ogive model is as follows:

$$P_i(\theta) = \int_{-\infty}^{a_i(\theta - b_i)} \frac{1}{\sqrt{2\pi}} e^{\frac{-z^2}{2}} dz, \tag{1}$$

where

$P_i(\theta)$ is the probability of a randomly chosen examinee at ability level θ answering item i correctly,

a_i is the discrimination parameter of item i,

b_i is the difficulty parameter of item i, and

z is a standardized score of the examinee involving trait score, and the two item parameters.

One-Parameter Logistic Model (1PLM – a.k.a. Rasch Model)

A mathematician in Denmark, George Rasch, came up with a different approach to IRT in the 1950s (see, Rasch, 1960). He used a logistic function to derive an ICC instead of the normal ogive function (though at the time he expressed his model differently), and his model contributed to simplifying the normal ogive model and the complexity of computation, though at the time, he appeared to be unaware of the earlier work on the topic of item response theory. In the Rasch model, the probability of a randomly chosen examinee at an ability level θ obtaining a correct answer on item i can be expressed as

$$P_i(\theta) = \frac{1}{1 + e^{-D(\theta - b_i)}}, \tag{2}$$

where e is an exponential constant whose value is about 2.718, and D is a scaling factor whose value is 1.7. The choice of this value for D, produces near equivalent values and interpretations between the item parameters in the normal ogive and two-parameter logistic models. Today, it is common to simply set D=1.0 since the normal ogive model is rarely used in practice and so preserving consistent of interpretations between the models is not important. But it is important to know when either studying item parameter estimates or generating them, that the value of D in the model be considered. It is still common, especially with the two and three parameter logistic models to retain D in the model with a value of 1.7. When D=1.0 (it is common to say that the model parameters are placed on what is referred to as the "logistic metric") and with D=1.7 (it is common to say that the model parameters are placed on what is referred to as the "normal metric".)

Two-Parameter Logistic Model (2PLM)

Two-parameter logistic model (2PLM) is a generalization of the 1PLM. Instead of having a fixed discrimination of '1' across all items as in 1PLM, in the 2PLM, each item has its own discrimination parameter. Thus, the model is mathematically expressed as

$$P_i(\theta) = \frac{1}{1 + e^{-Da_i(\theta - b_i)}}. \tag{3}$$

Three-Parameter Logistic Model (3PLM)

The three-parameter logistic model (3PLM) allows an ICC to have non-zero lower asymptotes. This model is more suitable for response data with those items in which examinees at the extremely low proficiency level may get the items correctly by chance; for example, a multiple choice item. In this model

$$P_i(\theta) = c_i + (1 - c_i) \frac{1}{1 + e^{-Da_i(\theta - b_i)}}, \tag{4}$$

where c_i represents the probability that examinees at extremely low levels of the trait answer item i correctly. This third item parameter, c_i, is often called either the *pseudo-chance-level parameter* or the *guessing parameter*, although 'pseudo-chance-level parameter' is theoretically more appropriate (Hambleton, Swaminathan, & Rogers, 1991). The 2PLM is a special case of 3PLM when $c=0$, and 1PLM is a special case of 2PLM when $a=1$.

Nonparametric Item Response Model

ICCs are always characterized by a single function in IRT models with parameters. However, assuming a single function for ICCs may not be appropriate to represent response data in some cases. Nonparametric item response models, in which a variety of shapes of ICCs are allowed, were developed in 1950s, even before parametric item response models were introduced.

Ramsay (1991) proposed a *kernel smoothing* approach for nonparametric item response models. In a kernel smoothing approach, *P(θ)* is estimated at *g*-th evaluation point, q_g, by a local averaging procedure. Thus,

$$\widehat{P}_i(\theta) = \sum_{g=1}^{G} w_g(\theta) Y_i^{(g)},$$ (5)

when

$$w_g(\theta) = \frac{K[(q_g - \theta)/h]}{\sum_k K[(q_k - \theta)/h]}.$$ (6)

So,

$$\widehat{P}_i(\theta) = \frac{\sum_g K\left[\dfrac{q_g - \theta}{h}\right] Y_i^{(g)}}{\sum_g K\left[\dfrac{q_g - \theta}{h}\right]},$$ (7)

where

h is the bandwidth parameter, which controls bias and sampling variance, and

K(u) is one of the kernel functions (Ramsay, 1991, p.617):

(a) *K(u)=0.5, |u|≤1*, and 0 otherwise, for uniform,

(b) *K(u)=0.75(1-u²)*, *|u|≤1*, and 0 otherwise, for quadratic, and

(c) *K(u)=exp(-u²/2)* for Gaussian.

Nonparametric item response models may not be as practically useful for operational uses as parametric models because nonparametric item response models do not provide informative, interpretable item parameters (for example, difficulty parameters), and it is hard to equate tests under nonparametric models. However, nonparametric models are frequently used for research

purposes such as evaluating model fit for parametric models since nonparametric models produce item characteristic fuctions that are very close to the observed data.

<div align="center">

Unidimensional IRT Models for Analyzing Polytomous Responses

</div>

In dichotomous item response models, the only type of response data is binary (i.e., 0 or 1). However, in some test situations, responses can be of more than two categories. For example, a questionnaire asking attitude, using Likert-scale items, may result in 5 categorical responses (strongly disagree, disagree, neutral, agree, and strongly agree, which can be coded from 0 to 4). Sometimes polytomous responses are dichotomized to be handled within dichotomous item response models, but it is very inappropriate in most cases because dichotomizing polytomous responses changes the nature of the scale of the measure and, as a result, validity of the measure could be seriously threatened.

Several item response models were developed to enable uses of polytomous responses within an IRT framework. Many of polytomous item response models are basically generalizations of the dichotomous item response models.

Partial Credit Model (PCM)

The partial credit model is an extension of the 1PLM (a.k.a., Rasch model) (Masters; 1982, 1987, 1988a, 1988b). Equation (2) for the 1PLM above can be rewritten as

$$P_i(\theta) = \frac{1}{1 + e^{-D(\theta - b_i)}} = \frac{\exp(D(\theta - b_i))}{1 + \exp(D(\theta - b_i))} = \frac{P_{i1}(\theta)}{P_{i0}(\theta) + P_{i1}(\theta)}, \quad (8)$$

where $P_{i1}(\theta)$ is the probability of a randomly chosen examinee, whose proficiency level is θ, scoring 1 on item i, and $P_{i0}(\theta)$ is the probability of a randomly chosen examinee, whose

proficiency level is θ, scoring 0 on item i. Thus, the probability of a person at θ, scoring x over x-1 can be computed as

$$\frac{P_{ix}(\theta)}{P_{ix-1}(\theta) + P_{ix}(\theta)} = \frac{\exp(D(\theta - b_{ix}))}{1 + \exp(D(\theta - b_{ix}))} \quad , \quad x = 1, 2, \ldots, m_i, \tag{9}$$

where $P_{ix}(\theta)$ and $P_{ix-1}(\theta)$ refer to the probabilities of an examinee at θ, scoring x and x-1, respectively. It should be noted that the number of item difficulty parameters are, now, m_i (one less than the number of response categories) in Equation (9). The probability of a randomly chosen examinee, who is at θ, scoring x on item i can be expressed as

$$P_{ix}(\theta) = \frac{\exp \sum_{k=0}^{x} (D(\theta - b_{ik}))}{\sum_{h}^{m_i} \exp \sum_{k=0}^{h} (D(\theta - b_{ik}))} \quad , \quad x = 1, 2, \ldots, m_i. \tag{10}$$

The function of Equation (10) is often called the *score category response function* (SCRF).

Generalized Partial Credit Model (GPCM)

The generalized partial credit model (Muraki, 1992) is a generalization of the PCM with a parameter for item discrimination added to the model. Muraki (1992) expressed the model mathematically as following:

$$P_{ix}(\theta) = \frac{\exp \sum_{k=0}^{x} (Z_{ik}(\theta))}{\sum_{h}^{m_i} \exp \sum_{k=0}^{h} (Z_{ik}(\theta))} \quad , \tag{11}$$

where

$$Z_{ik}(\theta) = Da_i(\theta - b_i + d_{ix}), \tag{12}$$

where d_{ix} is the relative difficulty of score category x of item i. Although Muraki (1992) followed the same way of parameterization for item and score category difficulty as Andrich's (1978)

rating scale model, the item difficulty parameters for each score category can be simply rewritten

as

$$b_{ix} = b_i - d_{ix},$$ (13)

and so is Equation (12),

$$Z_{ik}(\theta) = Da_i(\theta - b_{ix}).$$ (14)

The only difference between the PCM and GPCM is the additional discrimination parameters for

each item (a_i). (In *WinGen*, the parameterization of Equation (14) is used.)

Rating Scale Model (RSM)

There are two different approaches to the rating scale model. Andersen's (1977, 1983) proposed a response function, in which the values of the category scores are directly used as a part of the function:

$$P_{ix}(\theta) = \frac{e^{w_i\theta - a_{ix}}}{\sum_{x=1}^{m} e^{w_x\theta - a_{ix}}},$$ (15)

where $w_1, w_2, ..., w_m$ are the category scores, which prescribe how the m response categories are

scored, and a_{ih} are item parameters connected with the items and categories. An important

assumption of this model is that the category scores are equidistant.

Another form of the RSM was proposed by Andrich (1978a, 1978b), which can be seen

as a modification of PCM. In Andrich's RSM, item response functions are computed via

$$P_{ix}(\theta) = \frac{\exp\sum_{j=0}^{x}(\theta - (b_i + d_{ix}))}{\sum_{x=0}^{m}(\exp\sum_{j=0}^{x}(\theta - (b_i + d_{ij})))}.$$ (16)

where d_{ix} is the relative difficulty of score category x of item i. Andrich's RSM assumes that the category scores are fixed across all items in a testlet, and RSM should not be used if the scale of category scores varies across items in a testlet. (In WinGen, Andrich's RSM (Equation (16)) is used, and d- parameters are renamed to c- in the program interface only for an operational reason.)

Graded Response Model (GRM)

The graded response model was introduced by Samejima (1969, 1972, 1995) to handle ordered polytomous categories such as letter grading, A, B, C, D, and F and polytomous responses to attitudinal statements (such as a Likert scale). The model is expressed as

$$P_{ix}^*(\theta) = \frac{\exp(Da_i(\theta - b_{ix}))}{1 + \exp(Da_i(\theta - b_{ik}))},$$
(17)

where $P_{ix}^*(\theta)$ is the probability of an randomly chosen examinee with proficiency of θ scoring x or above on item i. This function is called the *cumulative category response function* (CCRF). Probability of each score category can be given by

$$P_{ix}(\theta) = P_{ix}^*(\theta) - P_{ix+1}^*(\theta).$$
(18)

Thus, the score category response function (SCRF) of the GRM can be expressed as

$$P_{ix}(\theta) = \frac{\exp[-Da_i(\theta - b_{ix+1})] - \exp[-Da_i(\theta - b_{ix})]}{[1 + \exp[-Da_i(\theta - b_{ix})]][1 + \exp[-Da_i(\theta - b_{ix+1})]]}.$$
(19)

Unlike the PCM and GPCM, the interpretation of item parameters of the GRM should be based on the CCRF, not on the SCRF. Within the GRM, a value of b-parameter for each response category indicates where a probability that a randomly chosen examinee, whose proficiency level (θ) is exactly same as the value of b-parameter, scores x or higher is 50% on the CCRF.

Although the statistical approaches to category response functions are totally different between the GRM and GPCM (Equation (11)), and so are the interpretation of item parameters, the SCRFs from the two models are usually very close to each other.

Nominal Response Model (NRM)

The nominal response model (also called the *Nominal Categories Model*) was introduced by Bock (1972). Unlike the other polytomous IRT models introduced above, polytomous responses in NRM are unordered (or at least not assumed to be ordered). Even though responses are often coded numerically (for example, $0,1,2,\ldots,m$), the values of the responses do not represent some sort of scores on items, but just nominal indications for response categories. Some applications of the NRM are found in uses with multiple choice items. The category function of NRM can be expressed as

$$P_{ix}(\theta) = \frac{e^{z_{ix}}}{\sum_{k=1}^{m_i} e^{z_{ik}}}, \qquad (20)$$

where

$$z_{ix} = a_{ix}\theta + c_{ix}. \qquad (21)$$

In Equation (21), a_{ix} and c_{ix} are called the slope and intercept parameters, respectively, and they are related with item discrimination and location. The sum of *a-* parameters and the sum of *c-* parameters across response categories are constrained to be zero. *a-* and *c-* parameters in Equation (21) can be rewritten as

$$z_{ix} = a_{ix}(\theta + b_{ix}), \qquad (22)$$

where

$$b_{ix} = \frac{a_{ix}}{c_{ix}}. \tag{23}$$

Thus, Equation (20) can be rewritten as

$$P_{ix}(\theta) = \frac{e^{a_{ix}(\theta - b_{ix})}}{\sum_{k=1}^{m_i} e^{a_{ix}(\theta - b_{ix})}}, \tag{24}$$

in which the *a*- and *b*- parameters can be interpreted as slope and location of items like other IRT models except no apriori ordering (WinGen uses the NRM in the form of Equation (24)).

Multidimensional IRT Model for Dichotomous Responses

Multidimensional Compensatory Three-Parameter Logistic Model (MC3PLM)

Whether it is intended or not, when an item or a set of items measures more than one latent trait, the assumption of unidimensionality is violated. The violation of the unidimensionality assumption may cause a systematic bias in the measurement process even though unidimensional IRT models are known to be somewhat robust against this violation.

Reckase (1985) came up with a multidimensional IRT model which can be seen as an extension of the unidimensional 3PL model:

$$P_i(\theta) = c_i + (1 - c_i)\frac{e^{(a'_i \theta + b_i)}}{1 + e^{(a'_i \theta + b_i)}}, \tag{25}$$

where

$P_i(\theta)$ is the probability of a correct response on test item *i* for a randomly chosen examinee whose proficiency is θ,

a_i is a vector of parameters related to the discriminating power of the test item,

b_i is a parameter related to the difficulty of the test item (but, NOT the difficulty itself),

c_i is a pseudo-chance level parameter, and

θ is a vector of trait scores for the examinee on the dimensions.

Equation (25) is very similar to the 3PLM (Equation (4)) except the fact that a_i and θ are vectors of the parameters for each dimension. A vector of a-parameters in Equation (25) can be tranformed to
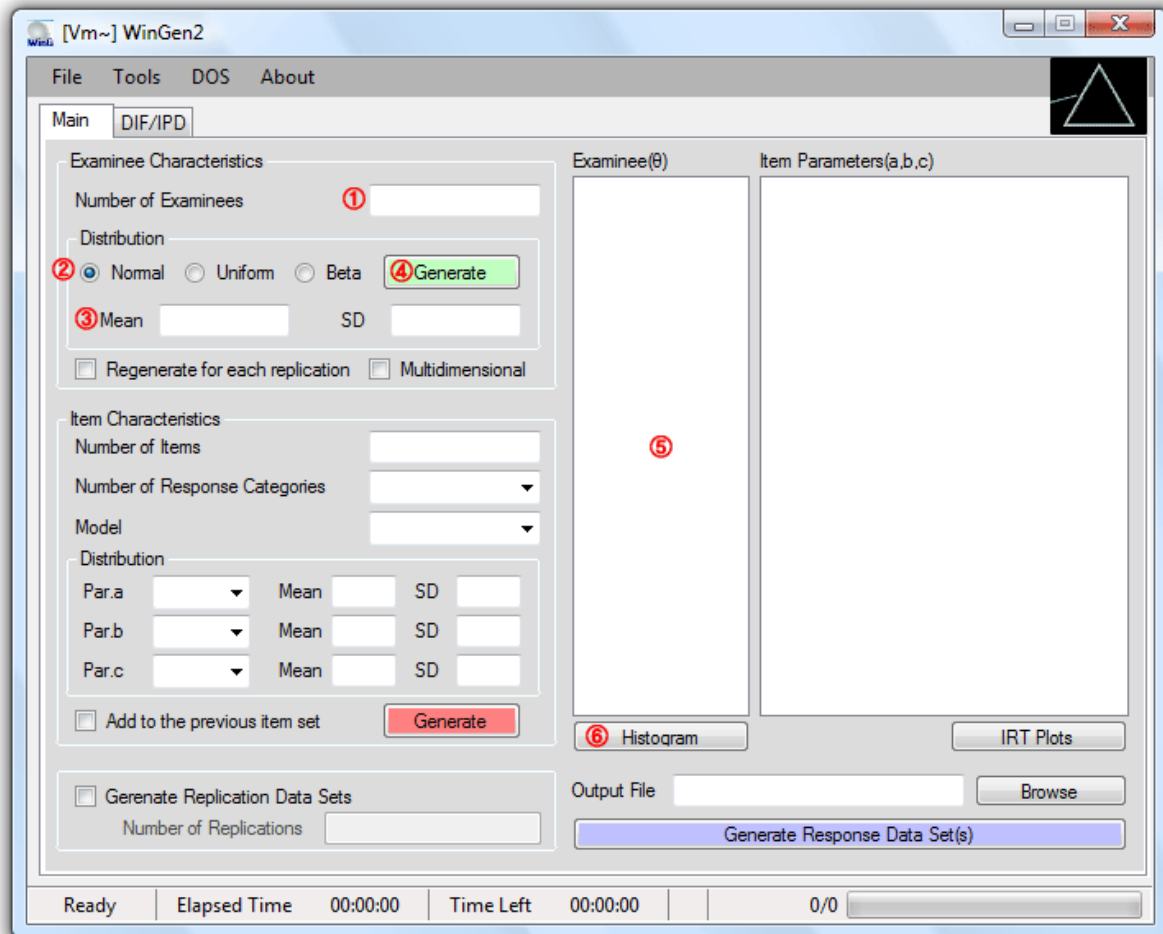
$$\text{MDISC}_i = \sqrt{\sum_{k=1}^{p} a_{ik}^2} \,, \tag{26}$$

where MDISC_i is the discrimination of the item i for the best combination of abilities and p is the number of dimensions. Also, a value that is equivalent in interpretation to the unidimensional b-parameter can be given by

$$\text{MDIFF}_i = \frac{-b_i}{MDISC_i} \,. \tag{27}$$

III. Quick Start

Generating Examinee Data (Step 1)

① Specify the number of examinees

② Select type of score distribution

③ Specify mean and standard deviation for a normal score distribution

　or, specify minimum and maximum value for a uniform score distribution

　or, specify a and b parameters for a beta score distribution

④ Click on a green 'Generate' button

⑤ Generated examinee theta scores should be shown in the box. The data set can be saved at

　'File > Save > Examinee Data'

⑥ Distribution of examinee thetas can be shown by clicking on the 'Histogram' button.

An example of generating examinee data where 50,000 examinees were drawn from a beta distribution (a=2, b=4) is shown in Figure 3.1. The use of beta distributions makes it easy to simulate skewed score distributions.

Figure 3.1 *Example of Generating Examinee Data*
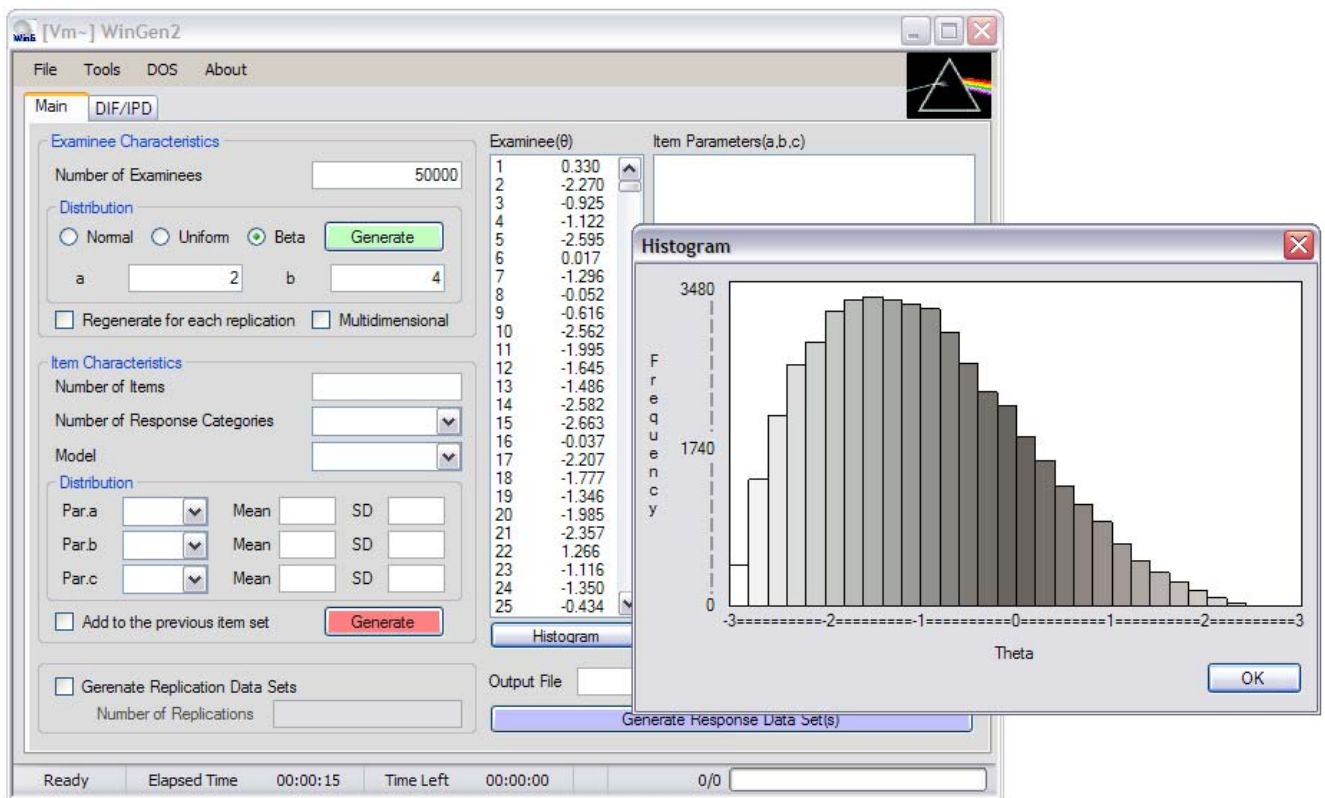
Generating Item Data (Step 2)
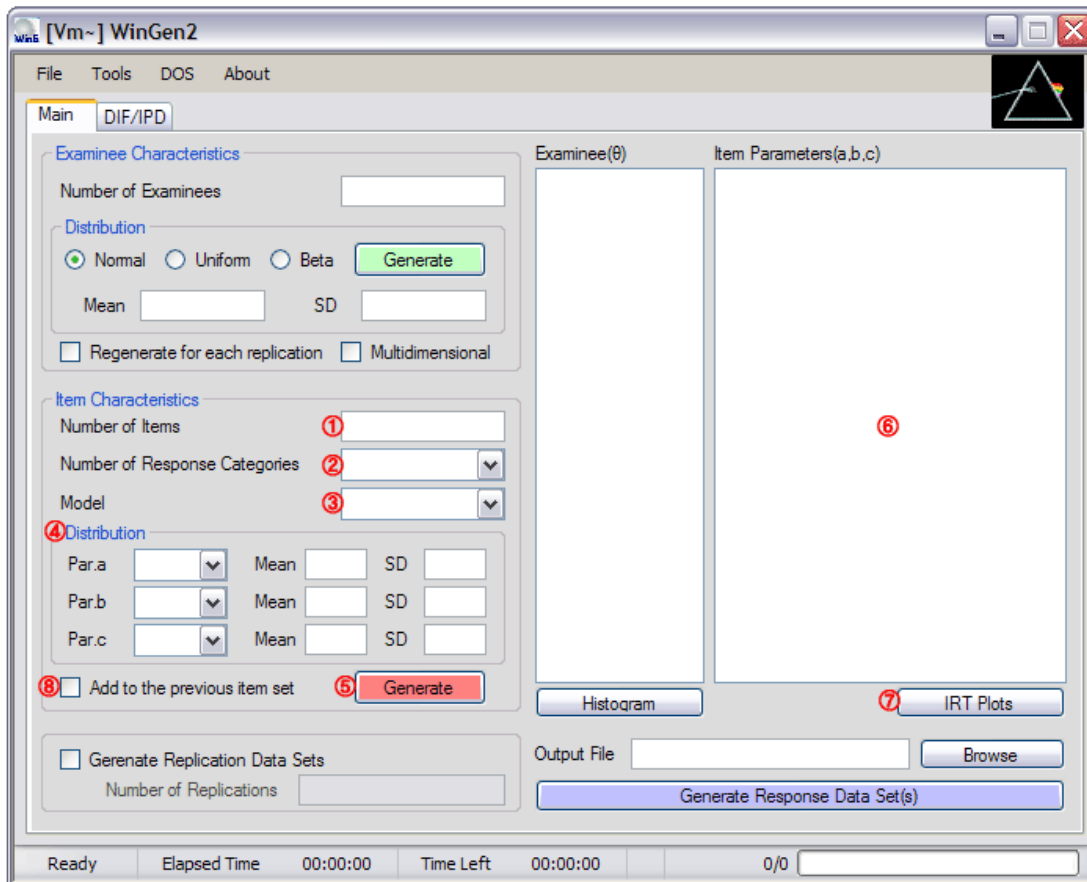
① Specify the number of items

② Specify the number of response categories. Select '2' for dichotomous response data.

③ Select an IRT model

④ Select distribution of item parameters and specify properties of the distributions

⑤ Click on a red 'Generate' button

⑥ Generated item parameter data should be shown in the box. The data set can be saved at 'File -> Save -> Item Parameter Data'

⑦ Item Characteristic Curves (ICCs), Test Characteristic Curve (TCC), Item Information Function Curves (IIF), and Test Information Function Curve (TIF) are provided by clicking on the 'IRT Plots' button

⑧ Mark the check box, 'Add to the previous item set', and repeat steps (① to ⑦) if another
set of items (or items of different IRT models) needs to be added to a previous set of
items. This option is particularly useful when a mixed format test form is to be simulated.

An example of generating item data, where 15 dichotomous items based on 3PLM and 5
polytomous items based on GRM (5 categories) are generated, is shown in Figure 3.2.

Figure 3.2 *Example of Generating Item Data*
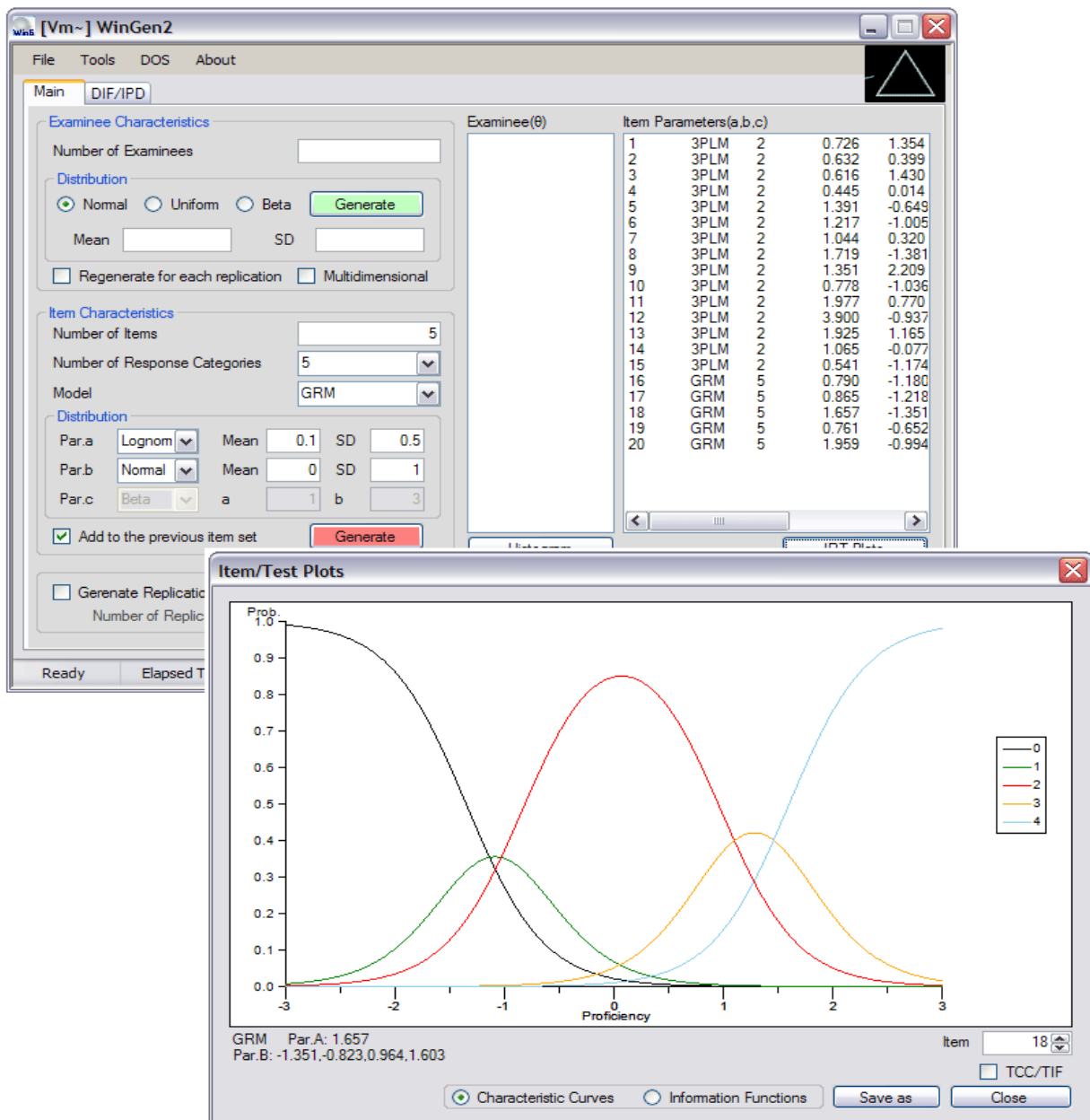
Generating Item Response Data (Step 3)
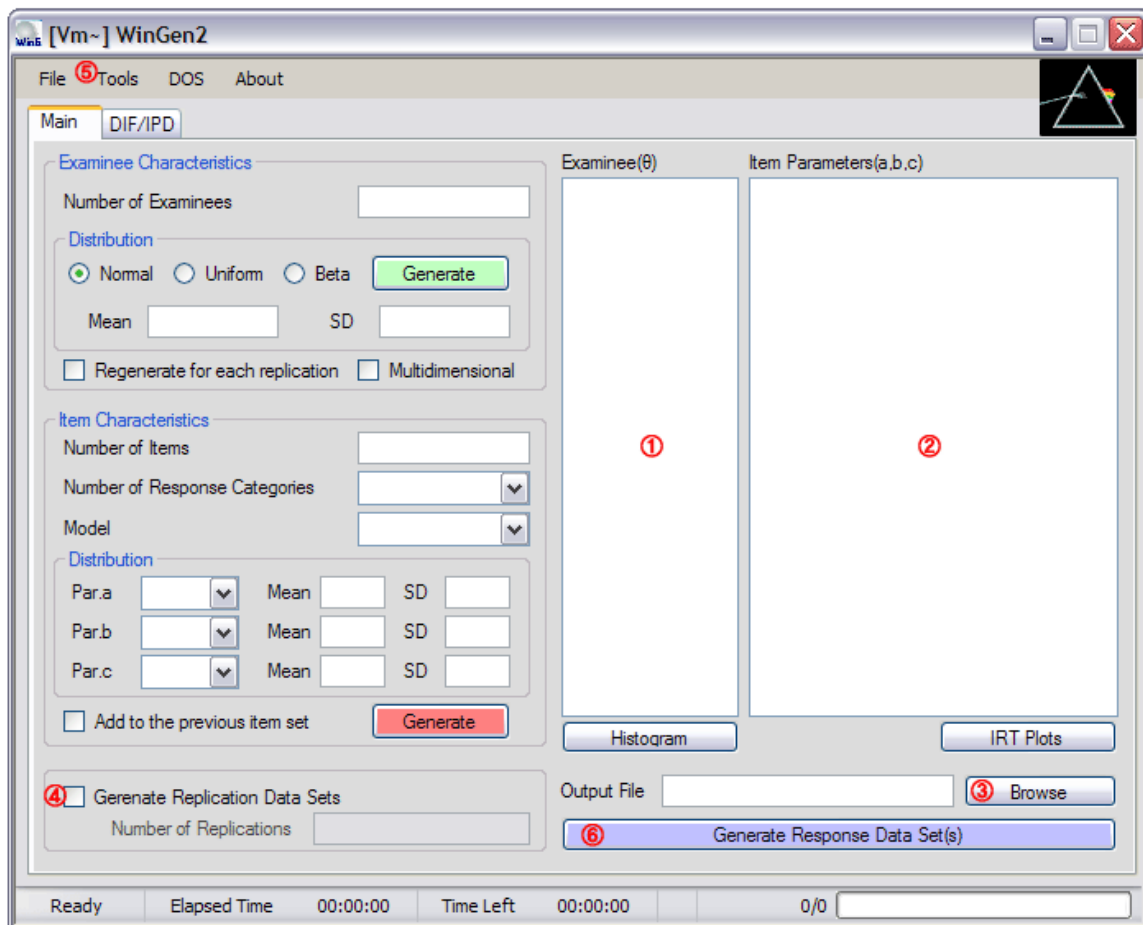
① Make sure examinee data generated in Step1 or a data set is opened with the pull-down menu, 'File > Open > Examinee Data'

② Make sure item parameter data generated in Step2 or a data set is opened with the pull-down menu, 'File > Open > Item Parameter Data'

③ Specify the name of the response data file by clicking on 'Browse'

④ Check the check box and specify the number of replications if replication data are desired

⑤ Adjust options at 'Tools -> Options'

⑥ Click on the blue 'Generate Response Data Set(s)' button. The data set should be seen by opening up the file with the extension of '.wgr'

An example of generating item data, where 15 dichotomous items based on the 3PLM and 5 polytomous items based on the GRM (5 categories) are generated, is shown in Figure 3.3.

Figure 3.3 *Example of Simulating Response Data*
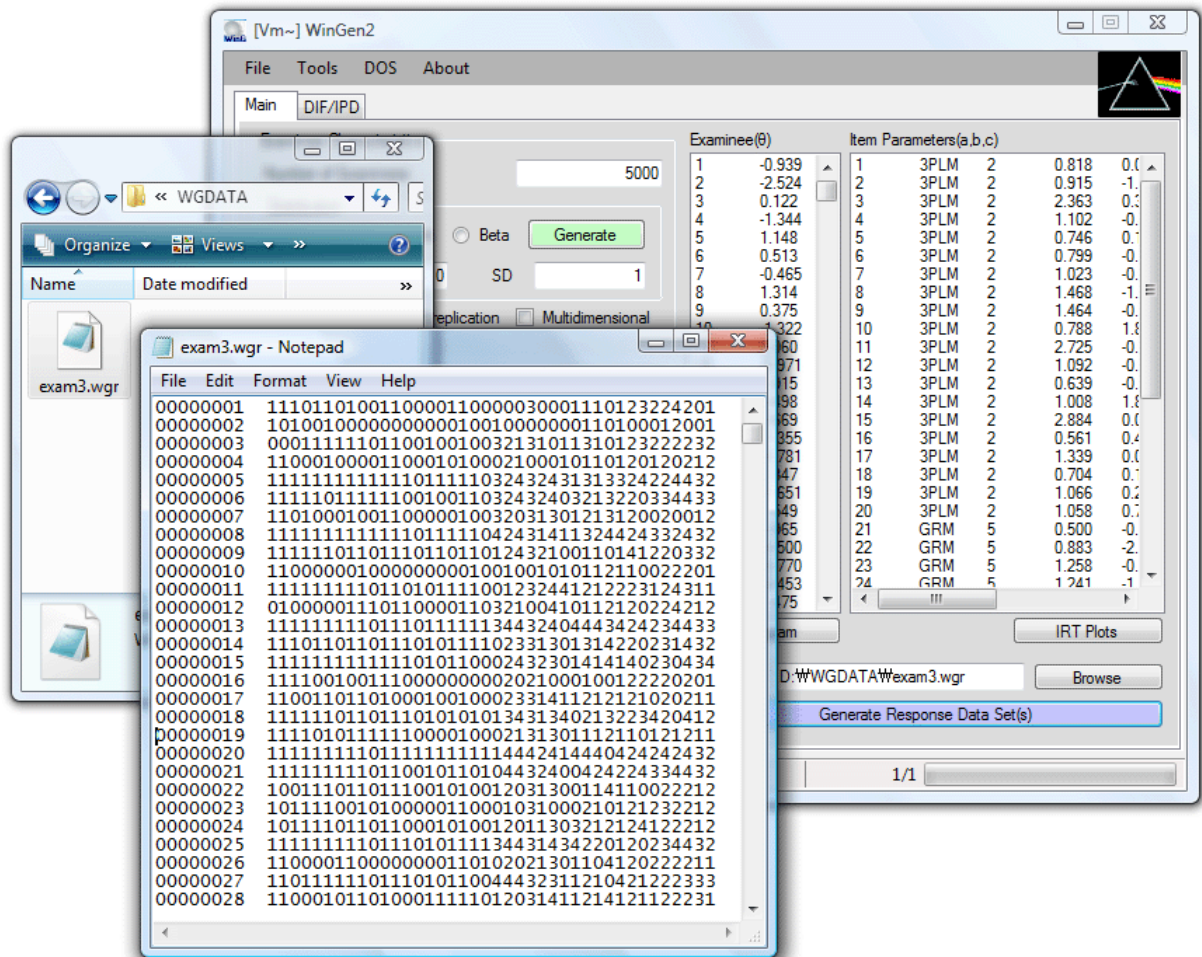
Options
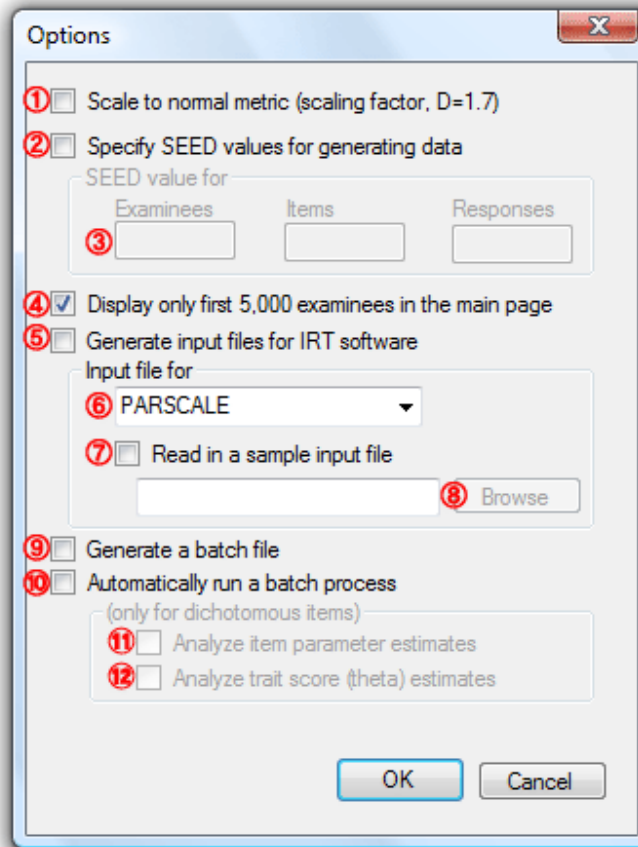
① 'Scale to normal metric (scaling factor, D=1.7)' (default: unchecked). By checking this option, WinGen generates item response data on the same scale as the normal-ogive model. Even when the normal ogive model though is of little interest, it is common to report scores on the "normal metric." This is less true when the polytomous response models are used.

② 'Specify SEED values for generating data' (default: unchecked). If it is unchecked, WinGen will set SEED values according to a system clock.

③ If Option ② is checked, three SEED values have to be specified for examinee data, item parameter data, and item response data, respectively. Specifying SEED values enables users to have control over generating random numbers.

④ 'Display only the first 5,000 examinees in the main page' (default: checked). WinGen displays only first 5,000 examinees in the main page if this option is checked. Undisplayed examinees are still in the memory and will be used for plotting a histogram and for

simulating response data. All examinees can be displayed by unchecking this option, but it may significantly slow down the processing time.

⑤ 'Generate input files for estimating program' (default: checked). This option allows WinGen automatically generating syntax files for estimating software, such as PALSCALE, Bilog-MG, and MULTILOG.

⑥ If Option ④ is checked, an estimating program has to be selected. Automated syntaxes for PARSCALE and for Bilog-MG are written to estimate parameters based on the 3PLM, and automated syntaxes for MULTILOG are written to estimate parameters based on the 2PLM.

⑦ 'Read in a sample input file' (default: unchecked). If it is checked, users can import a sample syntax file for an estimating program and make WinGen generate automated syntax based on the sample file imported. It must be noted that there should be one statement/command per line in a sample file.

⑧ If Option ⑥ is checked, a sample input file for an estimating program has to be specified. The sample input file specified should be for the estimating program chosen at Option ⑤.

⑨ 'Generate a batch file' (default: checked). If it is checked, WinGen automatically generates a batch file by which users or WinGen can run estimating programs by one cue.

⑩ 'Automatically run a batch process' (default: unchecked). If it is checked, WinGen will automatically run a batch file to run estimating programs after simulating data.

⑪ 'Analyze item parameter estimates' (default: unchecked). If it is checked, WinGen will import item parameter estimates from the estimating program chosen above and calculate the correlation, RMSE(RMSD), MAD, and BIAS. The statistics computed are stored in '*.wgz' files.

⑫ 'Analyze trait (theta) estimates' (default: unchecked). If it is checked, WinGen will import examinee score estimates from the estimating program chosen above and calculate the correlation, RMSE(RMSD), MAD, and BIAS. The statistics computed are stored in '*.wgz' files.

VI. WinGen Files

File Extensions

WinGen uses and produces several kinds of input and output files. Unique extensions are assigned on files according to their purposes. The files associated with WinGen are summarized in Table 4.1.

Table 4.1 *Extensions of WinGen Files*

Extension	Description	Type
*.wgc	WinGen cue file for executing sets of syntax files	Input only
*.wgd	WinGen descriptive statistics for simulated data (examinee, item)	Output only
*.wge	WinGen data file for Examinees	Input and Output
*.wgi	WinGen data file for Item Parameters	Input and Output
*.wgr	WinGen data file for Generated Responses	Output only
*.wgs	WinGen syntax file	Input only
*.wgz	WinGen analysis summary file	Output only
*.blm	Bilog-MG syntax file	Input and Output
*.psl	PALSCALE syntax file	Input and Output
*.mlg	MULTILOG syntax file	Input and Output
*.par	Item parameter data file for Bilog-MG, PALSCALE, or MULTILOG	Output only
*.sco	Examinee score data file for Bilog-MG, PALSCALE, or MULTILOG	Output only

WinGen File Formats

All input and output files of WinGen are ASC text file format, which can be opened and edited by Notepad, TextPad, MS Excel, SPSS, SAS, so on.
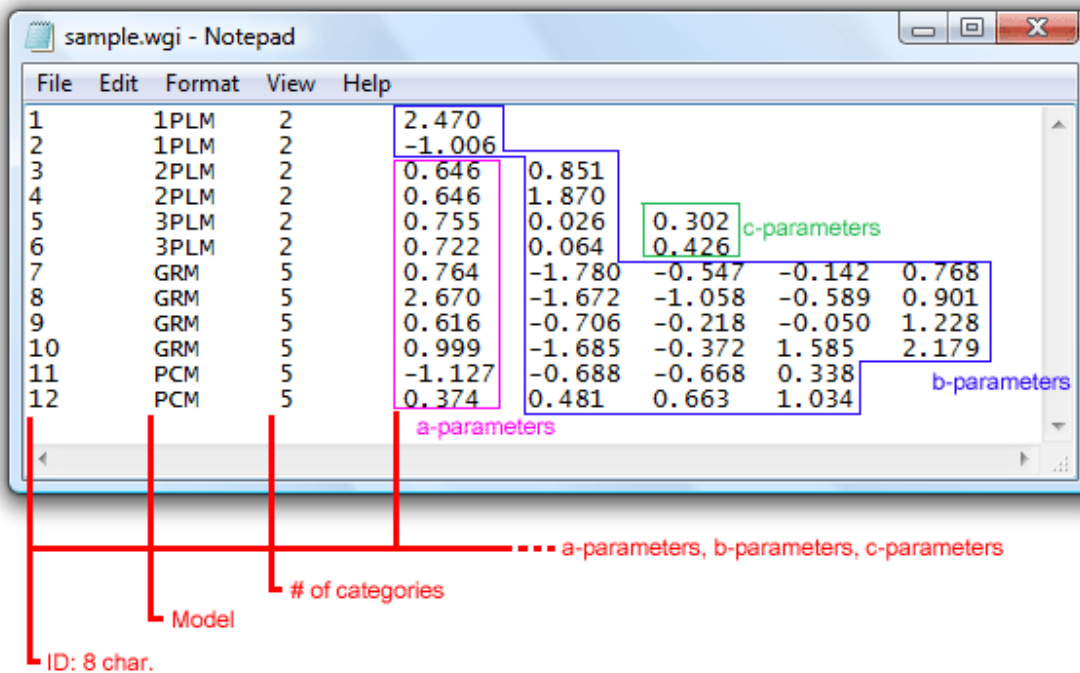
(a) WinGen Examinee data file (*.wge) – 'tab-delimited'

Format: [ID][theta's]
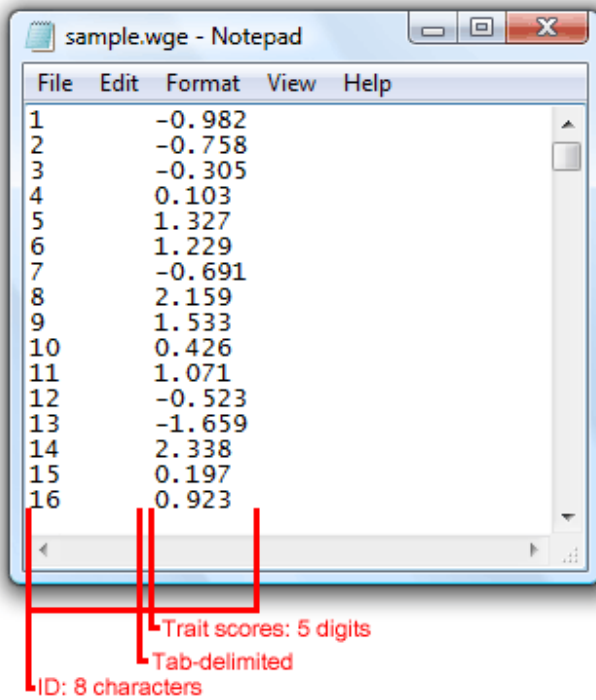
(b) WinGen Item parameter data file (*.wgi) – 'tab-delimited'

Format: [Item#][Model][# of categories][a-parameters][b-parameters][c-parameters]

Models: 1PLM, 2PLM, 3PLM, NON, GRM, PCM, GPCM, NRM, RSM, and/or MC3PLM

1PLM: One Parameter Logistic Model

2PLM: Two Parameter Logistic Model

3PLM: Three Parameter Logistic Model

GRM: Graded Response Model

PCM: Partial Credit Model

GPCM: General Partial Credit Model

NRM: Nominal Response Model

RSM: Rating Scale Model

MC3PLM: Multidimensional Compensatory Three Parameter Logistic Model

NON: Non-Parametric Model

NOTE: When the non-parametric model is used, another format should be used as following:

[Item#]['NON'][# of evaluation points between -3.0 and 3.0 on the theta scale][list of probabilities for each evaluation point]

(c) WinGen item Response data file (*.wgr) – 'space-delimited'

Format: [Examinee ID][2 spaces][Response string]
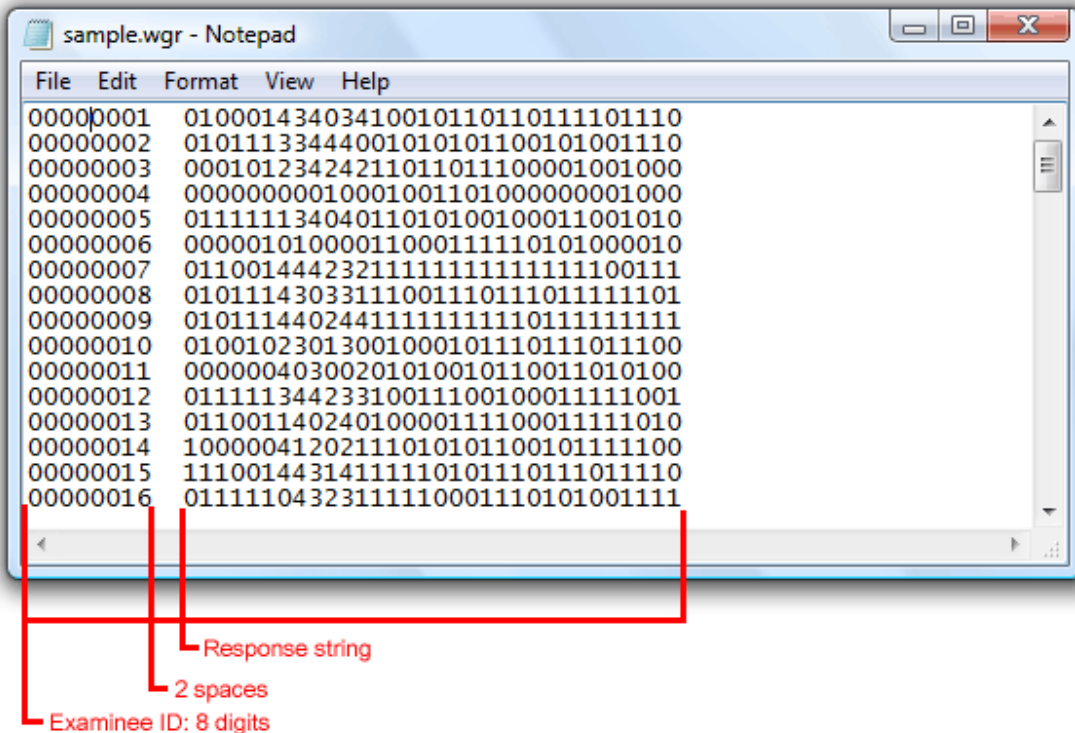
Naming Rules for Simulated Response Data Files

When a user generates more than one replication of the data and/or generates DIF/IPD data, WinGen automatically modifies the response data file name to organize a set of response data files generated. The naming rules WinGen uses are summarized in Table 4.2.

Table 4.2 *Naming Rules for Response Data Files*

| Modified File Name	Explanation
filename	a single data set
*filename*_1, *filename*_2, …, *filename*_n	*n* sets of replicated data
*filename*_r	a single data set for a reference group in DIF/IPD study
*filename*_f	a single data set for a focal group in DIF/IPD
*filename*_r_1, *filename*_r_2, …, *filename*_r_n	*n* sets of replicated data for a reference group in DIF/IPD study
*filename*_f1_1, *filename*_f1_2, *filename*_f2_1, *filename*_f2_2, …, *filename_fm_n*	*n* sets of replicated data for *m*-th focal group

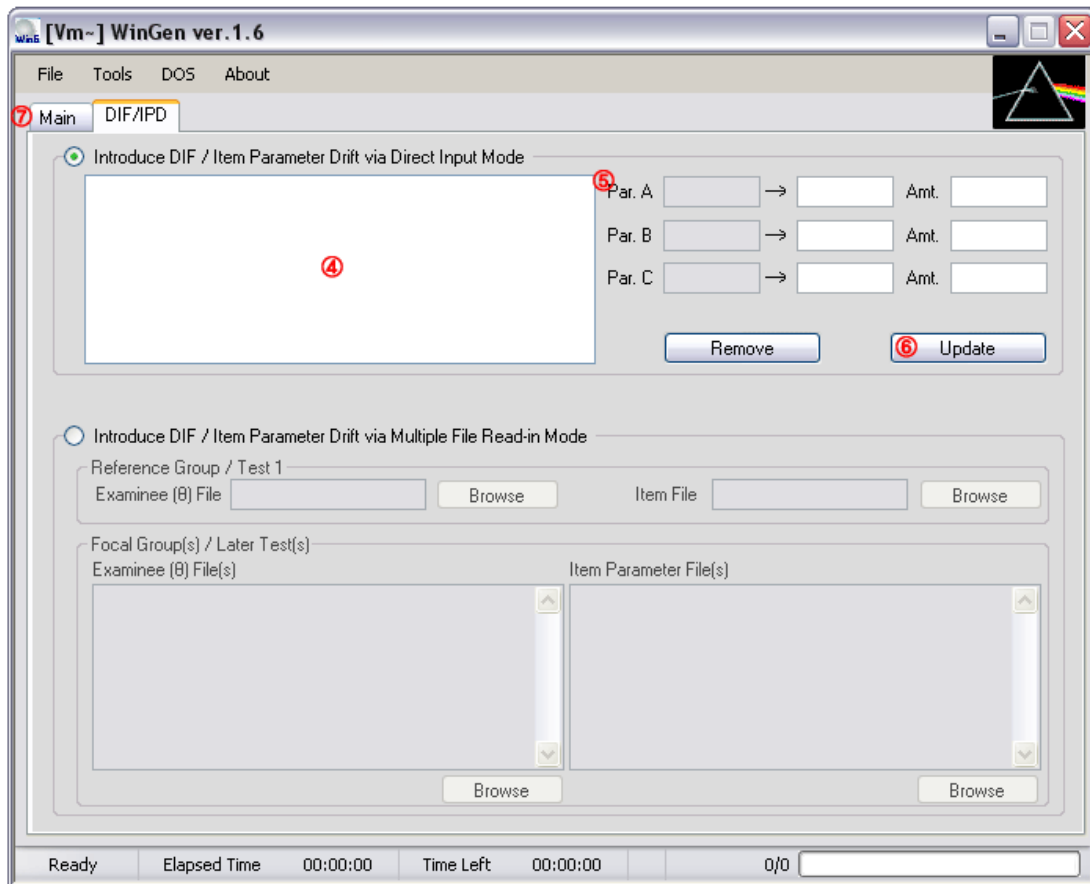* the file extension for response data files, '*.wgr', is skipped in the table

V. Advanced Uses of WinGen

Generating Response Data with DIF / IPD

There are two ways for generating response data with DIF/IPD: (a) direct input mode, and (b) multiple file read-in mode.

(a) Direct Input Mode
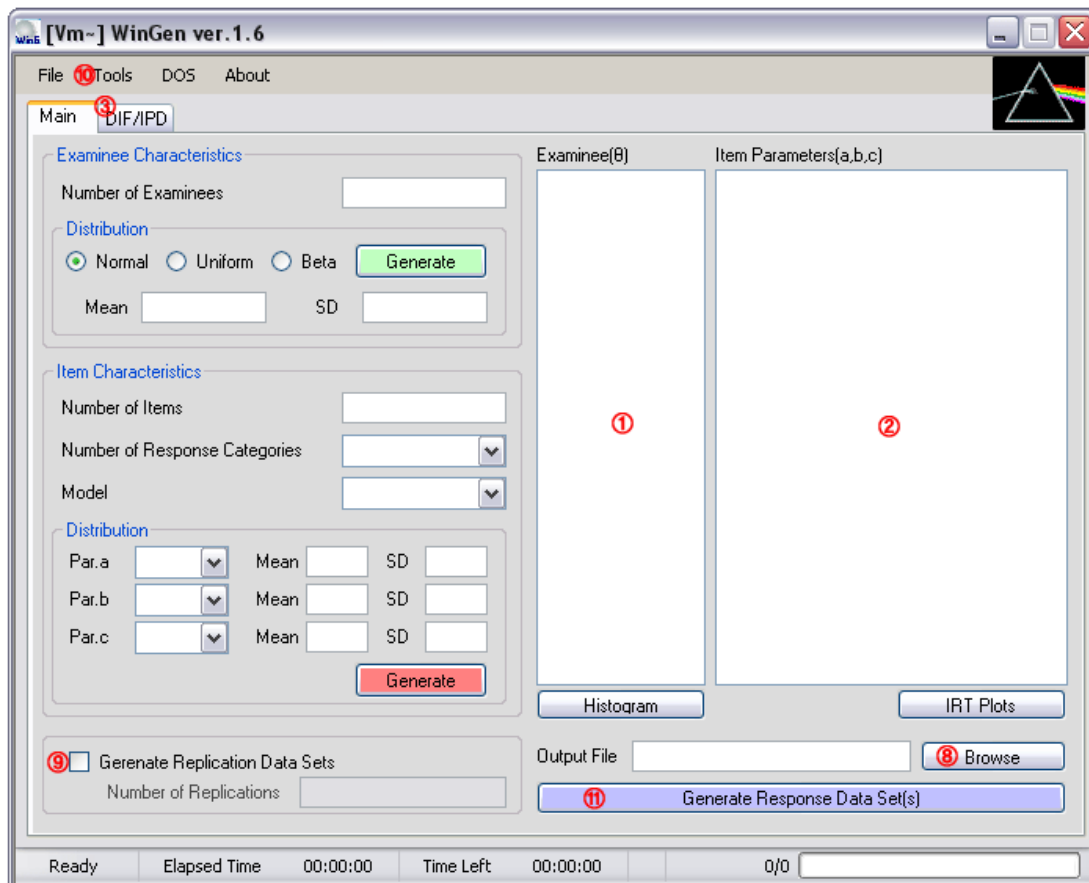
NOTE: Direct Input Mode supports only the 1PLM, 2PLM, and 3PLM. In order to introduce DIF/IPD on the items with other models, Multiple File Input Mode should be used.

① Make sure examinee data generated in Step1 or a data set is opened at

   'File>Open>Examinee Data'

② Make sure item parameter data generated in Step2 or a data set is opened at

   'File>Open>Item Parameter Data'

  Then, select items that are to be with DIF/IPD

③ Click on 'DIF/IPD' tab

④ The items selected at ③ should be shown in the box. Select an item to introduce DIF/IPD on the item

⑤ The original parameters of the item selected should be in the first column. Specify either item parameter with DIF/IPD in the second column or the amount of DIF/IPD in the third column

⑥ Click on 'Update' button. The amount of DIF/IPD should be reflected In the box of ③.
Go on the process of ③, ④, and ⑤ for all the items to which you want to introduce DIF/IPD

⑦ Click on 'Main' tab

⑧ Specify the name of response data file by clicking on 'Browse'

⑨ Check the check box and specify the number of replications if replicated studies are desired

⑩ Adjust Options at 'Tools>Options'

⑪ Click on the blue 'Generate Response Data Set(s)' button. The data set should be seen by opening up the file with the extension of '.wgr'

(b) Multiple File Read-in Mode
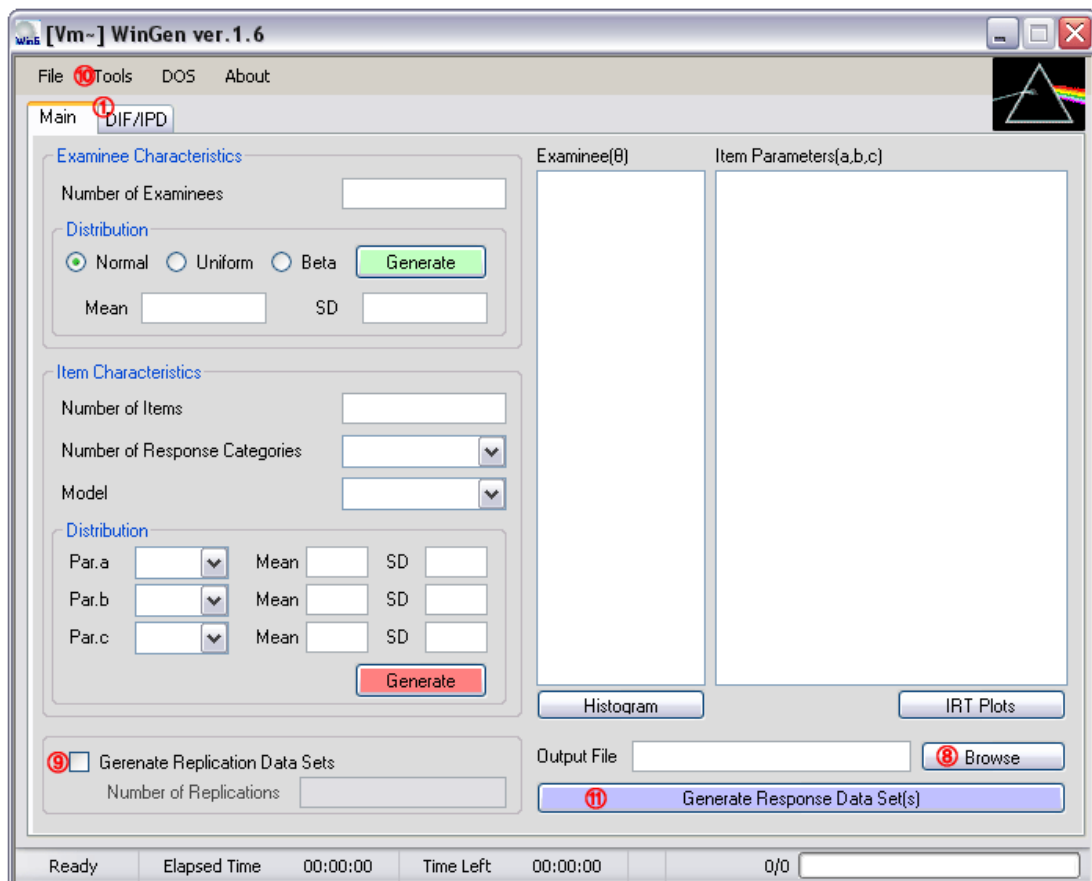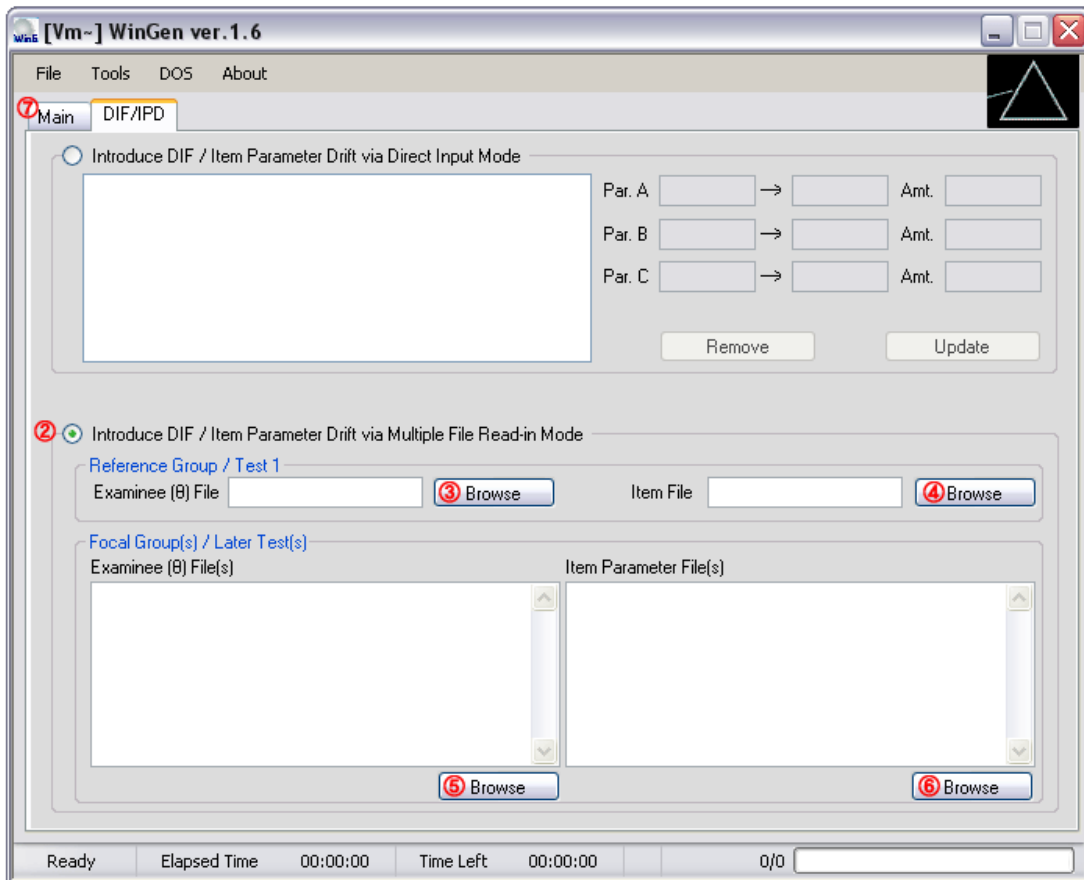
① Click on 'DIF/PD' tab

② Click on the radio button in the bottom ('Introduce DIF/IPD via Multiple File Read-in Mode)

③ Click on 'Browse' button and specify an examinee data file for a reference group

④ Click on 'Browse' button and specify an item parameter data file for a reference group

⑤ Click on 'Browser' button and specify an examinee data file for all focal groups

Or, specify examinee data files exactly as many as the number of focal groups

⑥ Specify item parameter data files exactly as many as the number of focal groups

⑦ Click on 'Main' tab

⑧ Specify the name of response data file by clicking on 'Browse'

⑨ Check the box and specify the number of replications if replicated studies are desired

⑩ Adjust Options at 'Tools > Options'

⑪ Click on the blue 'Generate Response Data Set(s)' button. The data set should be seen by opening up the file with the extension of '.wgr'

Generating Multidimensional Item Response Data
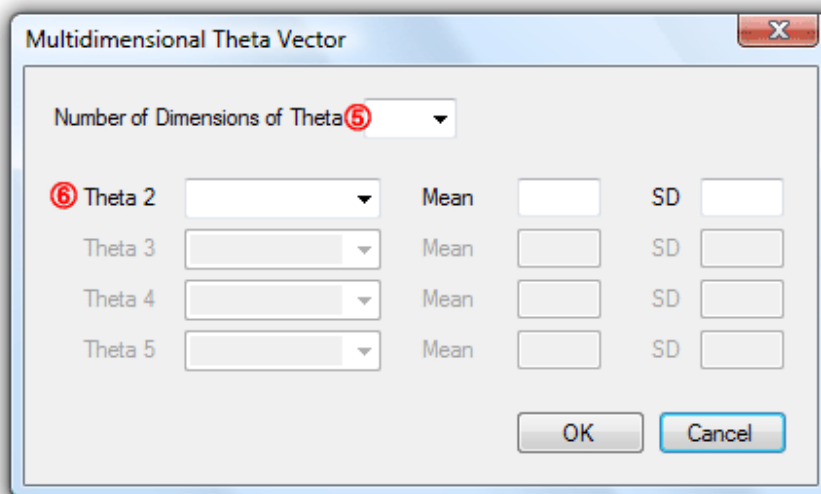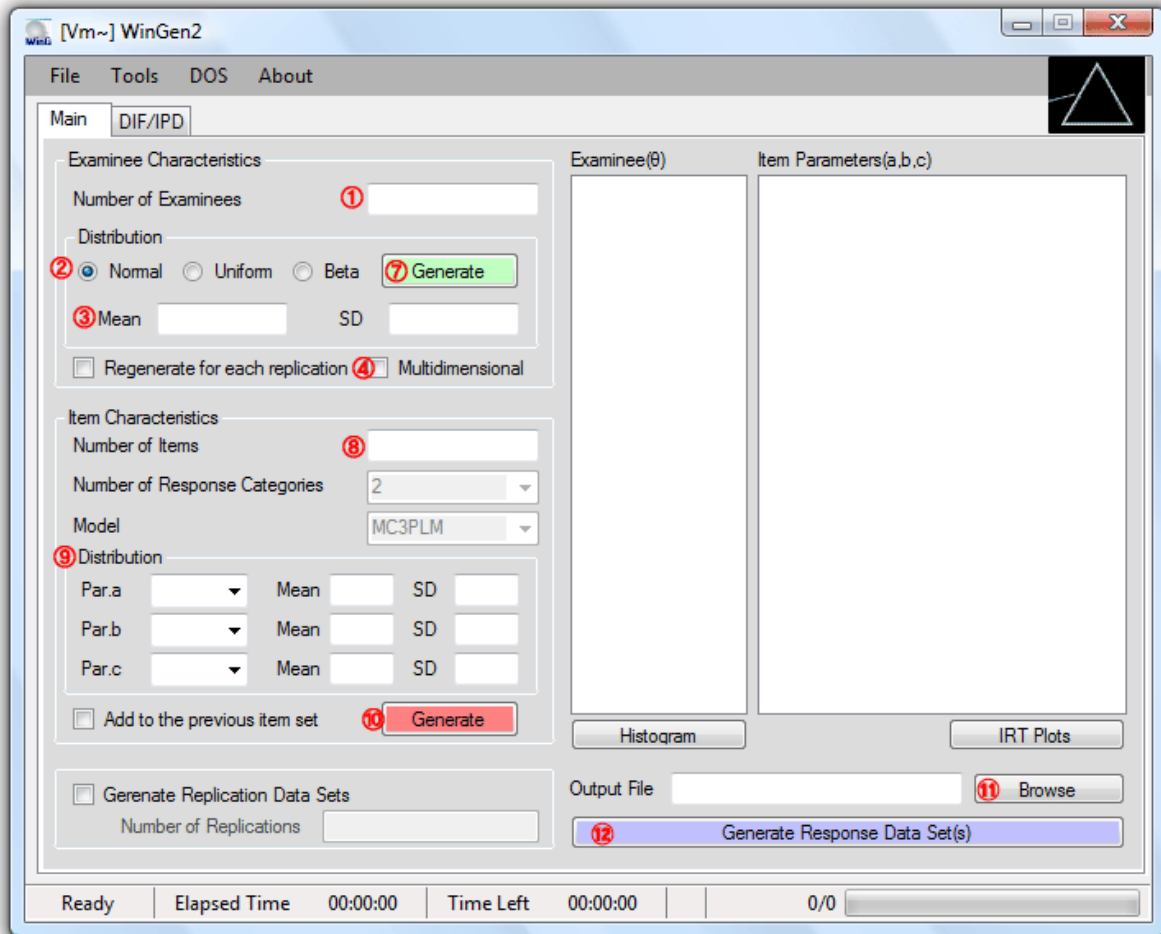
① Specify the number of examinees

② Select type of score distribution

③ Specify mean and standard deviation for the normal distribution

　　or, specify minimum and maximum value for a uniform distribution

　　or, specify a and b parameters for a beta distribution

④ Click on the checkbox, 'Multidimensional'. A Dialog box will pop up.

⑤ Select the number of dimensions (up to 5)

⑥ Select type of distribution and specify the characteristics of the distribution for each

　　dimension. Dimensions 2 to 5 can be set to be correlated with the dimension 1 by

　　selecting 'Correlated'. Click on 'OK' button.

⑧ Specify the number of items. The number of response categories is automatically set to '2'

　　since in this version of WinGen only dichotomous response data can be simulated. The

　　model is automatically set to 'MC3PLM'.

⑨ Select distribution of item parameters and specify properties of the distributions

⑩ Click on the red 'Generate' button

⑪ Specify the name of the response data file by clicking on 'Browse'

⑫ Click on the blue 'Generate Response Data Set(s)' button. The data set can be seen by

　　opening up the file with the extension of '.wgr'

Using Syntax Files and Cue Files

　　　　A user can use a syntax file to run WinGen instead of point-and-click on the graphic interface. Syntax files for WinGen can be composited using any kind of text editor software such as 'Notepad' or 'TextPad'.

Table 5.1 *Commands / Options Used in the WinGen Syntax File*

Line / Description	Commends / Options
Line1. Examinee Data Characteristics	[*# of examinees*], [*type of distribution*],[*mean/min/a*],[*SD/max/b*],['replicate' *or nothing*]

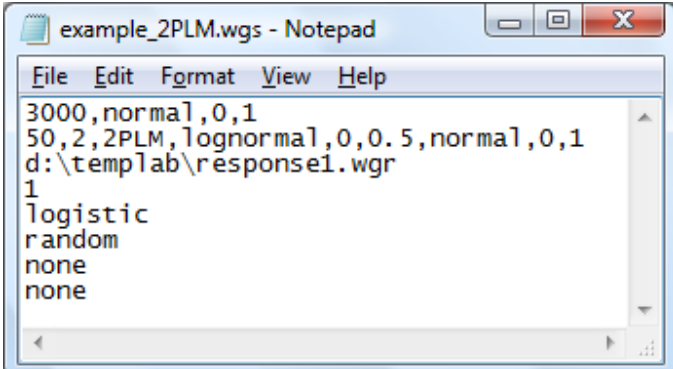|	
	['skip']
	['*file*'],[*full filename of the examinee data (*.wge) including the path (the name of the folder)*]

* 'skip' option lets WinGen skip generating examinees' data, and WinGen will use examinees' data which currently is loaded in the examinees' theta box for simulating the response data.	['MIRT'],[*# of examinees*],[*# of dimensions*],['replicate' *or nothing*]
	  Followed by
	  [*distribution for the first dim.*],[*mean/min/a*],[*SD/max/b*]
	  [*distribution for the second dim.*],[*mean/min/a*],[*SD/max/b*]
	      ……

Line2. Item Data Characteristics * 'skip' option lets WinGen skip generating item parameters, and WinGen will use item parameter data which currently is loaded in the item parameter box for simulating the response data.	[*# of items*],[*# of response categories*],[*model*],[*distribution for item parameter*],[*mean/min/a*],[*SD/max/b*],….
	['skip']
	['file'],[*full filename of the item data (*.wgi) including path the (the name of the folder)*]
Line3. Name of Response Data File	[*full name of the response data (*.wgr) including the path where the simulated response data will be saved*]
Line4. Replication	['replication'],[*# of replications if you chose* 'replication']
	['1']
Line5. Scale (Normal or Logistic)	['logistic']
	['normal']
Line6. Seed Value	['random']
	[*SEED for examinees*],[*SEED for items*],[*SEED for responses*]
Line7. Programs for Estimating Parameters * 'auto' option can be used only for dichotomous response data sets.	['none']
	['Bilog-MG' or 'Parscale' or 'Multilog'],['auto']
	['Bilog-MG' or 'Parscale' or 'Multilog'],[*full filename including the path of the sample syntax file for the program*]
Line8. A Batch to Run Programs for Estimating Parameters * the option 'batch', 'run', 'item', and 'score' trigger the options of	['none']
	['batch']
	['batch'],['run']

⑧, ⑨, ⑩, and ⑪ in the Option dialog. See Section 4. 'Options' above for details.	['batch'],['run'],['item']
	['batch'],['run'],['item'],['score']

* The commands/options italicized are the places where your inputs need to be placed.

** The commands/options in quotation marks are the actual commands/options to be input except the quotation marks.

*** Each line of syntax is delimited by a comma (','). There should be no space before and after the comma.
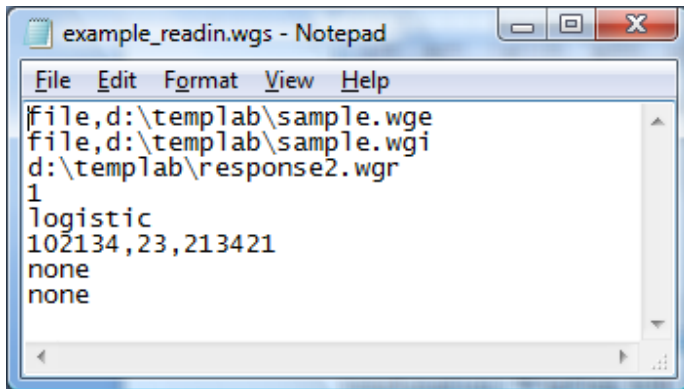
Example #1: 3,000 examinees and 50 items with 2PLM

WinGen will generate 3000 theta scores from a normal distribution with a mean of 0 and a standard deviation of 1. Then, WinGen will generate 50 items with 2PLM ('a' parameters from a lognormal distribution with a mean of 0 and a SD of 0.5, and. 'b' parameters from a normal distribution with a mean of 0 and a SD of 1). WinGen will store simulated response data into *'d:\templap\responses1.wgr'* WinGen does not replicate data (i.e., just simulate data one time). The response data will be simulated based on the logistic model scale (D=1.0). SEED values will be randomly chosen. WinGen will not generate syntax files for the programs for estimating parameters and WinGen will not generate a batch file, according to the specifications of the user.
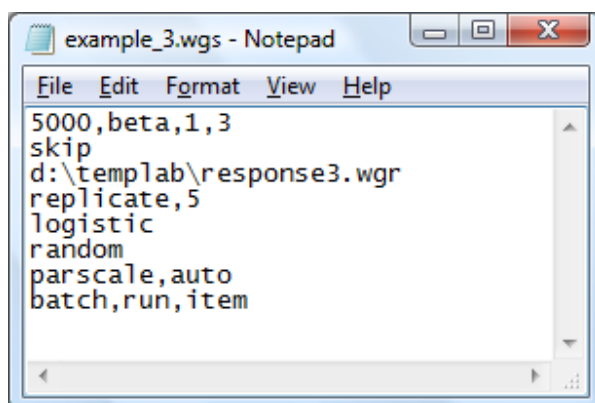
Example #2: Reading-in examinee and item data from files

WinGen will read-in examinees' data from the file named '*d:\templab\sample.wge*' Then, WinGen will read-in item parameter data from '*d:\templab\sample.wge*' WinGen will store simulated response data into '*d:\templap\responses2.wgr*' WinGen does not replicate data (i.e., just simulate data one time). The response data will be simulated based on the logistic model scale (D=1.0). SEED values will be 102134, 23,213421 for examinees, items, and responses, respectively although SEED values for examinees and items will not be used since the user had WinGen read-in from files . WinGen will not generate syntax files for the programs for estimating parameters and WinGen will not generate a batch file, according to the specifications of the user.

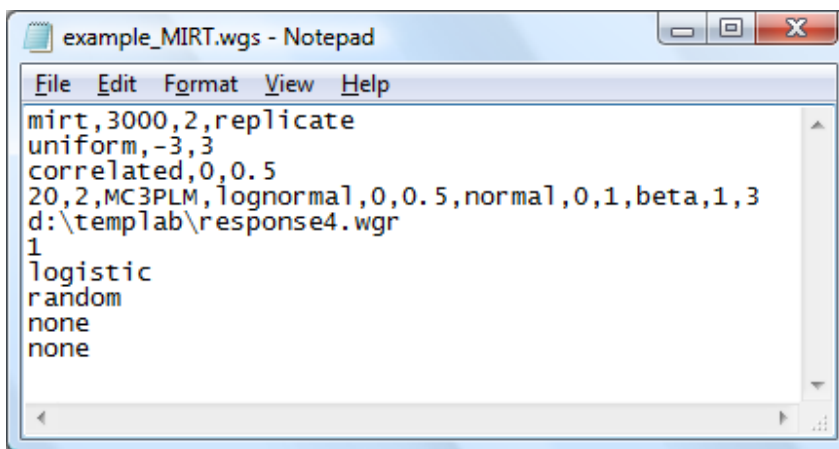Example #3: Generating examinees, using existing items in memory, replicating, and estimating

WinGen will generate 5000 examinee theta scores from a beta distribution with the distribution parameters of 1 and 3. Then, WinGen will read-in item parameter data from '*d:\templab\sample.wge*' WinGen will store simulated response data into '*d:\templap\responses3.wgr*' (actually, the response files will be saved with the name of

'reponse3_1.wgr', 'reponse3_2.wgr', 'reponse3_3.wgr', 'reponse3_4.wgr', and 'reponse3_5.wgr' since you have a syntax for replicating this study 5 times). The response data will be simulated based on normal metric scale (D=1.7). SEED values will be randomly chosen. WinGen will generate syntax files for PARSCALE automatically (not from a sample code), and will generate a batch file to run PARSCALE for the 5 replications. PARSCALE will estimate only the item parameters.

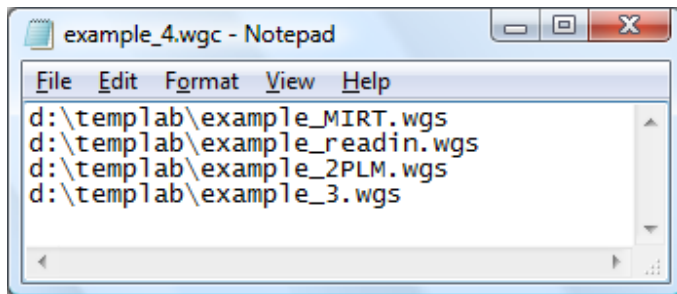Example #3: Simulating multidimensional IRT data (2 dimensions)

WinGen will generate 3000 examinees' theta for two dimensions. The first theta distribution will be drawn from a uniform distribution between –3 and 3. The vector of theta for the second dimension will be correlated with the first dimension of theta. Then, 20 items with MC3PLM will be generated (a-parameters from a lognormal distribution, b-parameters from a normal distribution, and c-parameters from a beta distribution). WinGen will store simulated response data into *'d:\templap\responses4.wgr'*. WinGen will not replicate this study. The response data will be simulated based on logistic model scale (D=1.0). SEED values will be randomly chosen. WinGen will not generate syntax files for the programs for estimating parameters. WinGen will not generate a batch file.

Cue File

A cue file is kind of a batch file with which WinGen runs multiple syntax files. It is simply a list of full file names of the syntax files. A cue file can be executed at "File>Open and Run> Cue File" on the menu bar.

Example of a Cue File


```
example_4.wgc - Notepad
File  Edit  Format  View  Help
d:\templab\example_MIRT.wgs
d:\templab\example_readin.wgs
d:\templab\example_2PLM.wgs
d:\templab\example_3.wgs
```

References

Baker, F. B. (1989). *GENIRV: A program to generate item response vectors* (Unpublished manuscript). Madison, WI: University of Wisconsin, Laboratory of Experimental Design.

Fang, H., & Johanson, G. (2005, April). W*INIRT: A Windows-based item response theory data generator with an equating and differential item functioning simulation guide*. Paper presented at the meeting of the American Educational Research Association, Montreal, Canada.

Hambleton, R. K., & Rovinelli, R. (1973). *A Fortran IV program for generating examinee response data from logistic test models. Behavioral Science, 17, 73-74. (Software was revised, September 1990)*

Hambleton, R. K., Swaminathan, H., & Rogers, H. J. (1991). *Fundamentals of item response theory.* Newbury Park, CA: Sage.

Han, K. T. (2007). *WinGen2: Windows software that generates IRT parameters and item responses* [computer program]. Amherst, MA: University of Massachusetts, Center for Educational Assessment. Retrieved May 13, 2007, from http://www.umass.edu/remp/software/wingen/

Harwell, M. R., Stone, C. A., Hsu, T.-C., & Kirisci, L. (1996). Monte Carlo studies in item response theory. *Applied Psychological Measurement, 20(2),* 101-125.

Microsoft. (2002). *Microsoft Windows XP* [computer program]. Redmond, WA: Author.

Microsoft. (2005). *Microsoft .NET framework version 2.0* [computer program]. Redmond, WA: Author.

Muraki, E. (1992). *RESGEN: Item response generator* [computer program]. Princeton, NJ: Educational Testing Service.

Muraki, E., & Bock, R. D. (2003). *PARSCALE 4: IRT item analysis and test scoring for rating-scale data* [computer program]. Chicago, IL: Scientific Software.

Thissen, D. (2003). *MULTILOG* 7: *Multiple categorical item analysis and test scoring using item response theory* [computer program]. Chicago, IL: Scientific Software.

Yoes, M. (1997). *PARDSIM* [computer program]. St. Paul, MN: Assessment Systems Corporation.

van der Linden, W. J., & Hambleton, R. K., (Eds.). (1997). *Handbook of modern item response theory*. New York: Springer-Verlag.

Zimowski, M. F., Muraki, E., Mislevy, R. J., & Bock, R. D. (2003). *BILOG-MG 3* [computer program]. Chicago, IL: Scientific Software.

Version: May 13, 2007