

simulate_CAT: A Computer Program for Post-Hoc Simulation for Computerized Adaptive Testing*

simulate_CAT: Bilgisayar Ortamında Bireyselleştirilmiş Testler için Post-Hoc Simulasyon Yazılımı

İlker Kalender **

Summary

This paper presents a computer software developed by the author. The software conducts post-hoc simulations for computerized adaptive testing based on real responses of examinees to paper and pencil tests under different parameters that can be defined by user. In this paper, short information is given about post-hoc simulations. After that, the working principle of the software is provided and a sample simulation with required input files is shown. And last, output files are described.

Özet

Bu çalışmada yazar tarafından geliştirilmiş olan bir bilgisayar yazılımı tanıtılmaktadır. Söz konusu yazılım bilgisayar ortamında bireyselleştirilmiş test yaklaşımı için, kullanıcı tarafından tanımlanabilen farklı parametreler altında, bireylerin kağıt kalem testlerine verdikleri yanıtları kullanarak post-hoc simulasyonları yapmaktadır. Çalışmada önce post-hoc simulasyonlar hakkında kısa bir bilgi verilmekte, ardından yazılımın çalışma prensibi ve gerekli dosyalar ile birlikte örnek bir simulasyon gösterilmektedir. Son olarak da, çıktı dosyaları tanıtılmaktadır.

Post-hoc Simulations in computerized adaptive testing

When developing computerized adaptive test (CAT) versions of paper and pencil tests, it is essential to conduct preliminary analyses to see how much CAT provides reduction in number of items and how well standard errors of ability estimations of examinees are. This is done through post-hoc simulations that are based on using examinees' previous response patterns provided for paper and pencil format of a test (Kalender, 2011a). Post-hoc simulations provide a hypothetical testing environment to simulate a test as if examinees are given a CAT-based test. Using real examinee responses provides a better description of examinees' psychometric characteristics (Wang, Bo-Pan, Harris; 1999). Results of the post-hoc simulations yield information as to optimum CAT strategies (such as ability estimation methods, test termination rules, item exposure rates) that can be implemented in real CAT testing conditions (Kalender, 2011b).

A post-hoc simulation approach can be described as follows: (i) when simulation starts for first examinee, an item is selected from the item bank and examinee's responses is checked to that item from the examinee's previous response set in paper and pencil version of the test, (ii) based on that response, an ability estimation is made for examinee and

* The software presented in this paper was developed for author's Ph.D. dissertation entitled *Effects of Different Computerized Adaptive Testing Strategies on Recovery of Ability*.

** Asst. Prof. Dr. İhsan Dođramacı Bilkent University, Graduate School of Education, Ankara-Turkey,
email: kalenderi@bilkent.edu.tr

another item is picked based on predefined item selection rules and examinee's response to that item is checked, (iii) the algorithm is repeated until a predefined test termination rules is hold.

The Software

simulate_CAT is a free, Windows-based computer software to conduct post-hoc simulations for CAT based on dichotomus Item Response Theory models (Kalender, 2012). The software allows users to select two CAT components (Thompson & Weiss, 2011). The first one is ability estimation methods. simulate_CAT can run simulations based on Maximum Likelihood and Bayesian Expected a Posteriori methods. The second component is test termination rules: fixed number of items for each examinee and standard error threshold.

When the former option is selected, all examines are given the same number of items regardless of their standard error of ability estimation. The latter one creates a CAT simulation which runs until standard error of ability goes below a predefined value. simulate_CAT uses Maximum Information approach for item selection.

simulate_CAT requires two input files in text file format. First input file contains responses of examinees. In the file, each line should include an examinee's ID (number of digits for ID can be determined by user) and item responses including IDs in 0 (false) /1 (true) vector.

As an example, format of this file for a group of examinees with an ID of 5 characters and response patterns of 10 is as follows:

ID	Response vector
00001	1010101010
00002	1010101010
00003	1010101010
00004	1010101010
00005	1010101010

Second file includes item parameters. Each line includes an item ID, and parameters of an item calibrated using one of the dichotomous Item Response Theory models (separated with |).

Format of a file including 5 items can be seen below. Item parameters should be ordered in item discrimination (a), item difficulty (b), and pseudo-guessing parameters (c).

For one-parameter logistic model, item discrimination (a) should be defined as 1 and pseudo-guessing parameters (c) as 0 in the item parameter file. Similarly, for two-parameter logistic model, set pseudo-guessing parameter (c) as 0.

Item	ID	a	b	c
	1	1.106	2.595	0.200
	2	0.934	0.761	0.200
	3	0.978	1.160	0.200
	4	1.376	0.161	0.200
	5	0.684	0.816	0.200

Before running post-hoc simulation, item parameters in the second text file (item bank) should be processed by a separate database application, ItemBankBuilder that is delivered with simulate_CAT. ItemBankBuilder creates a database file and transfer all information in the item bank text file to that database.

Conducting a Simulation

User interface of simulate_CAT can be seen in Figure 1.

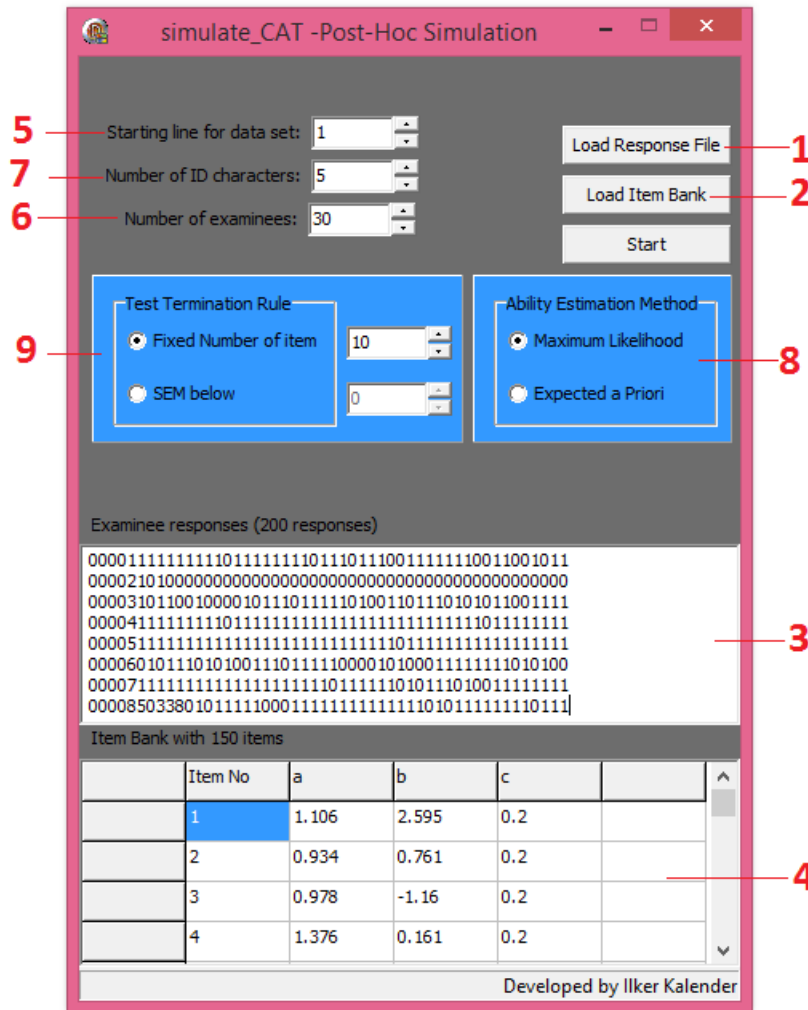


Figure 1. User Interface of simulate_CAT

Using Load Response File button (1), examinees' responses are loaded from the first text file. Load Item Bank (2) button opens a dialog box through which a database file (a file with .DB extension) is selected (User should select Files of Type from the bottom of the dialog box opened to see files with DB extension). Before loading item bank, users should run ItemBankBuilder application to transfer item parameters from a text file into a database file (ItemBankBuilder and its user manual is supplied with simulate_CAT). In the two fields (3 and 4) at the lower part of the user interface, loaded responses and item parameters can be seen.

After that, user can define starting line in response file for post-hoc simulations (5). This feature can be used if user wants to skip some examinees and start at another line.

Number of examinees to include in the simulation can be set using Number of Examinees field (6). Simulation will start at the line set in Starting line for data set field and stops after simulations is run number of examinees set. Number of digits for ID field can also be defined using the interface (7). Prior to running post-hoc simulations, ability estimation method (8) and test termination rule (9) should be selected using the interface. Finally, start button is clicked.

In Figure 1 above, for example, simulation will start with the first examinee with the response file and 30 examinees will be included. Number of ID digits of examinees was defined as 5. Each examinee will be tested using fixed number of items (10) using Maximum Likelihood ability estimation method.

simulate_CAT produces three output files at the same directory where responses and item bank are located. Main output file (Ability.txt) includes a line for each examinee to show examinee ID, number of items given (*#ofItem*), ability estimations (*Theta* and *Theta_Corr* [*original theta is divided by 1 minus square root of SE estimate of the examinees*]), and standard errors for ability estimations (*SE*). Sample output is given below.

ID	#ofItem	Theta	Theta_Corr	SE
00001	10	-1.925	-2.6138	0.5135
00002	10	-2.482	-3.2193	0.4786
00003	10	-1.311	-1.5558	0.3966
00004	10	-1.331	-1.5914	0.4044

The second output file (Details.txt) provides progress for each examinee included in the post-hoc simulations. The third file (Exposure.txt) includes frequency of exposure for each item in the item bank. All data in the output files are tab-separated. The outputs in the three file can be pasted into a data process software such as MS Excel and IBM SPSS to visualize the data.

Availability

The simulate_CAT software was developed using Object Pascal in Delphi. The software does not require installation; it can be directly run from its executable file. But for handling database file, Borland database engine should be installed. This engine was developed and distributed by the company that developed Delphi. In the simulate_CAT package this engine is also included.

There is no limitation for number of examinees and items in the bank rather than computer limitations. Executable file of the simulate_CAT and sample input/output files can be requested via email (kalenderi@bilkent.edu.tr) at no cost. Use of the program is restricted to academic purposes only.

REFERENCES

- Kalender, I. (2011a). *Effects of different computerized adaptive testing strategies on recovery of ability* (Unpublished doctoral dissertation). Middle East Technical University, Ankara.
- Kalender, I. (2011b). Computerized adaptive testing administration for entrance to higher education programs. *Proceedings of the International Higher Education Congress: New Trends and Issues*. (pp. 1641-1646). Istanbul, Turkey: Higher Education Council Publications.
- Kalender, I. (2012). Computerized adaptive testing for student selection to higher education. *Journal of Higher Education*, 2 (1), 13-19.
- Thompson, N. A., & Weiss, D. J. (2011). A framework for the development of computerized adaptive tests. *Practical Assessment, Research, and Evaluation*, 16(1). Weiss, D. J. (1973). The
- Wang, X., Bo-Pan, W. & Harris, V. (1999). *Computerized Adaptive Testing Simulations Using Real Test Taker Responses*. Law School Admission Council Computerized Testing Report. LSAC Research Report Series.