

# FlexCCT: Software for Continuous CCT

Stephen L. France

University of Wisconsin - Milwaukee  
Sheldon B. Lubar School of Business  
P. O. Box 742  
1 414 229 4596  
france@uwm.edu

Mahyar Vaghefi

University of Wisconsin - Milwaukee  
Sheldon B. Lubar School of Business  
P. O. Box 742  
1 414 229 4235  
mahyar@uwm.edu

William H. Batchelder

University of California, Irvine  
2129 Social Sciences Plaza A  
Mail Code: 5100  
1 949 824 7271  
whbatc@uci.edu

## ABSTRACT

In this paper, we describe a software package called FlexCCT for analyzing numerical ratings data. FlexCCT is implemented in MATLAB and incorporates a range of different cultural consensus theory (CCT) models. We describe the standalone GUI version of FlexCCT. We give an illustrative example, showing how FlexCCT can be used to analyze and interpret essay rating data.

## Keywords

CCT, maximum likelihood, optimization, essay rating

## 1. INTRODUCTION

Cultural consensus theory (CCT) is a methodology used to analyze cultural values or “truth”. CCT has several educational data analysis/data mining applications. CCT can be used to analyze educational essay/question ratings data to i) evaluate rater competency, ii) evaluate rater bias, iii) calculate accurate competency weighted ratings, and iv) evaluate the easiness/difficulty of rating individual answers. The CCT results can be used to evaluate a set of essay ratings and then recommend actions, for example retraining certain raters or using rater competency to determine the number of raters assigned to rating tasks. CCT can also be used as part of the rating/grading process; for example, the item easiness CCT models can be used to assign additional raters to answers that are deemed difficult to rate.

We present FlexCCT, a software package for implementing maximum likelihood CCT. We do not give axiomatic or mathematical descriptions of the class of CCT models described in this paper. These can be found in [1,3,4]. We give an intuitive description of several of the CCT models implemented in the FlexCCT software, concentrating on CCT models for continuous data. We summarize the model features and describe the software implementation of the models. We then describe work that uses CCT to analyze essay grading data.

## 2. THE CCT MODELS

Consider a situation where there are  $n$  subjects or raters. Each rater assigns a score to each of  $m$  questions. Each question could be a quality rating, an estimation of quantity, or any other type of question that may elicit a numerical response. There is no a-priori known correct answer to any of the questions, which is why CCT has taken the moniker of “test theory without an answer key” [2].

Let  $\mathbf{X}$  be an  $n$  rater  $\times$   $m$  item matrix of item ratings or scores. A simple method of calculating a  $1 \times m$  latent answer vector  $\mathbf{z}$  would be to calculate the average score for each item across all raters. However, this ignores the fact that some raters may be more competent than other raters and that some raters may be “biased” to giving lower or higher scores. The basic CCT models are based upon a set of axioms [1,2,3,4]. In this paper, we do not describe these axioms formally, but we give some basic intuition.

User competency is defined as a measure of inverse error variance. For each rater  $i$ , the rater competence  $d_i \in \mathbb{R}^+$  is defined as the inverse error variance, so that  $d_i^{-1} = \sigma^2(\epsilon_{ik})$ . The maximum likelihood function for the basic CCT model is given in (1).

$$L(\mathbf{d}, \mathbf{z} | \mathbf{X}) = \prod_{k=1}^m \prod_{i=1}^n \sqrt{(d_i/2\pi)} e^{-(d_i(x_{ik} - z_k)^2/2)} \quad (1)$$

Bias operationalizes the tendency of raters to consistently rate either lower or higher than the latent answer key values. Either additive bias or multiplicative bias can be incorporated into the basic model. For additive bias, the inner likelihood term  $ILT = (x_{ik} - z_k)$  is replaced by  $(x_{ik} - b_i - z_k)$  and for multiplicative bias it is replaced by  $(x_{ik} - b_i z_k)$ . Only one type of bias can be included in the model, as including both additive and multiplicative biases over parameterizes the model [4]. Three models for “item easiness”, described in [4], are incorporated into FlexCCT. The first model is an error variance model, where  $\sigma^2(\epsilon_{ik})$  is split into rater components and answer components. The second model incorporates a multiplicative easiness scaling factor, so that for each combination of rater  $i$  and item  $j$ , the competency is scaled by an easiness parameter  $\beta_j$ , so that  $d_i$  is replaced by  $d_i \beta_j$ . The third model is similar to the second model, except that the easiness parameter is additive, so that  $d_i$  is replaced by  $d_i + \beta_j$ .

## 3. SOFTWARE DESCRIPTION

FlexCCT consists of a set of MATLAB functions and a compiled, standalone GUI version of the software. The GUI consists of a single input screen, where the user configures the software parameters and an output screen, which displays results from the CCT model optimization. The output screen has an option to save the output parameter values. The input screen is given in Figure 1 and a description of the associated options is given in Table 1.

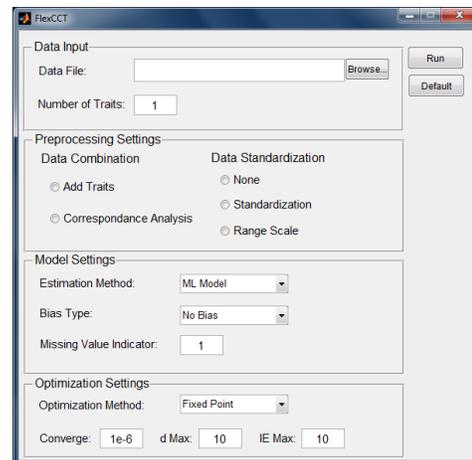


Figure 1: Input Screen

**Table 1. Input Parameters**

Name	Options
Data File	A csv file containing the values of <b>X</b> . Rows correspond to raters and columns correspond to items. The data should not contain a header.
Data Combination	<b>Add Traits:</b> If more than one trait/attribute then add the trait values together. <b>Correspondence analysis:</b> Calculates a single continuous trait from multiple qualitative traits.
Data Standardization	<b>None:</b> Use raw data. <b>Standardize:</b> Subtract column $\mu$ and divide by column $\sigma$ . <b>Range Scale:</b> Divide by column range.
Estimation Method	<b>Simple Average:</b> $d_i = 1$ for all raters and $z_k$ is the average value of $x_{ik}$ . <b>Factor Analysis:</b> Utilizes a minimum residual factor analysis as per classical CCT [5]. <b>ML Model:</b> Basic maximum likelihood model from (1). <b>IE Error Variance:</b> Item easiness error variance model. <b>IE Multiply:</b> Item easiness where $d_{ij} = d_i \beta_j$ . <b>IE Add:</b> Item easiness where $d_{ij} = d_i + \beta_j$ .
Bias Type	<b>No Bias:</b> (see section 2). <b>Additive Bias:</b> (see section 2). <b>Multiplicative Bias:</b> (see section 2).
Optimization Method	<b>Fixed point:</b> Fixed point estimation. <b>Two Stage Fixed Point.</b> The values of <b>z</b> and <b>d</b> are estimated first, followed by other parameters. <b>Derivative Free:</b> Standard MATLAB routine. <b>Gradient:</b> MATLAB Gradient descent optimization, utilizing first order derivatives
Converge	Converge criteria for optimization, (default = 1e-6).
MissingVal	Indicator for a missing value, defaults to -1
Max d	Upper bound for <b>d</b> . Prevents a single rater having dominant competency (default = 10)
Max IE	Upper bound for item easiness $\beta$ (default = 10).

When the “Run” button is pressed and the model optimization is completed, the user is presented with an output screen, which displays a summary of the model output. This summary includes the maximum log-likelihood, the algorithm run time, and values for all of the output parameters. The output button allows users to save the output parameter values to a csv file. In the csv file, each set of parameters (e.g. **z**, **d**, **b**,  $\beta$ ) is assigned to a column in the file. Row vectors are transposed. The file output parameters are summarized in Table 2.

**Table 2. Output Parameters**

Name	Options
<b>z</b>	An $1 \times m$ latent answer vector.
<b>d</b>	An $n \times 1$ vector of competencies.
<b>b</b>	An $n \times 1$ vector of biases.
$\beta$	An $1 \times m$ vector of item easiness parameters
<b>pll (1-3)</b>	Three $1 \times m$ vectors of partial log likelihoods corresponding to values calculated with floor( <b>z</b> ), <b>.z</b> , and floor( <b>z</b> )+1.

## 4. EDUCATIONAL EXAMPLE

In [4], a detailed example is given to show how CCT can be used in essay (or more general) rating applications. A subset of 50 essays was taken from a set of high school essays. The prompt for the essays was to describe a situation involving laughter. A grading rubric was defined and each essay was graded on 6 attributes, with each attribute having a range from 1-6. An overall continuous score was calculated using two approaches. In the first approach, the assumptions of classical test theory were used and the scores for each attribute were added together to give a total score in the range of 6–36. In the second approach, multiple correspondence analysis was used to explicitly scale the multiple ordinal attribute scales into one continuous scale. The essays were graded by 2 expert graders and 10 student graders. Each student grader was given 30 minutes for training and 3 minutes to grade each essay.

Some overall conclusions reached in [4] are that CCT provides useful measures of rater competency, rater bias, and item easiness/difficulty. CCT can be used to help train and evaluate raters and to identify essays where accurate evaluation is difficult. The CCT competencies can be used to produce competency weighted averages of essay ratings. In the essay rating data analysis, incorporating bias gave improved model fit and additive bias gave better model fit than multiplicative bias. Likewise, the multiplicative item easiness model gave better model fit than the additive item easiness model.

## 5. CONCLUSIONS AND FUTURE WORK

The current version of FlexCCT (1.0.0) provides a flexible framework for implementing CCT and can be used to analyze a wide range of ratings/questionnaire data. FlexCCT is implemented as a set of MATLAB functions. There is a standalone GUI version of the software, which does not require a MATLAB license and provides a wrapper for a set of continuous CCT models. For future versions of FlexCCT, we plan to incorporate clusterwise CCT, which simultaneously assigns raters to clusters/cultures and calculates the CCT model for each culture.

## 6. ACKNOWLEDGMENTS

The third author acknowledges the support of a grant from the Army Research Office (ARO) and a fellowship from the Oak Ridge Institute for Science and Education (ORISE).

## 7. REFERENCES

- [1] Batchelder, W. H. and Romney, A. K. New results in test theory without an answer key. In Roskam, E. E. ed. *Mathematical Psychology in Progress*. Springer-Verlag, Heidelberg, Germany, 1989, 229-248.
- [2] Batchelder, W. and Romney, A. 1988. Test theory without an answer key. *Psychometrika*, 53, 1 (Mar. 1988), 71-92.
- [3] France, S. L. and Batchelder, W. H. 2012. *Unsupervised Consensus Analysis for On-line Review and Questionnaire Data*. Working paper, UC Irvine.
- [4] France, S.L. and Batchelder, W. H. 2013. *A Maximum Likelihood Item Difficulty Model for Consensus Analysis*. Working paper, UC Irvine.
- [5] Romney, A. K., Weller, S. C. and Batchelder, W. H. 1986. Culture as Consensus: A Theory of Culture and Informant Accuracy. *American Anthropologist*, 88, 2 (Oct. 1986), 313-338