Online Appendix A. *modelavgIRT* R function

modelavgIRT R function Description:

This function reads in person scores (i.e., EAP scores) and their standard errors from the validation sample, and information criteria values (BIC, AIC) from the calibration sample from each of a set of candidate IRT models and outputs model-averaged person scores and standard errors (see manuscript Equations 4 and 5).

modelavgIRT R function Input:

personscores – A dataset consisting of person scores obtained from each candidate model in the validation sample, with rows denoting person and columns denoting model

personSEs – A dataset consisting of person score standard errors obtained from each candidate model in the validation sample, with rows denoting person and columns denoting model

selectionindex – List of information criteria values (BIC, AIC) for each model, in the order of the columns of *personscores* and *personSEs*

rescale – Logical; if set to TRUE (default), prior to averaging each models' person scores will be rescaled to have mean of 0 and a variance of 1 and standard errors will be rescaled proportionally

modelavgIRT R function Code:

```
modelavgIRT <- function(personscores,personSEs,selectionindex,rescale=TRUE) {
  ##rescale personscores to have mean 0 and var 1
  #rescale personSEs proportionally
  if(rescale==TRUE){
  for(i in seq(ncol(personscores))){
    personscores[,i] <- (personscores[,i] - mean(personscores[,i]))/sd(personscores[,i])
    personSEs[,i] <- personSEs[,i]/sd(personscores[,i])
  }
  }
  ##compute weights
  weights <- c(rep(NA,length(selectionindex)))
  for(i in seq(length(selectionindex))){
    weights[i] <- sum(exp(-.5*selectionindex[1:length(selectionindex)]+.5*selectionindex[i]))^(-1)
  }
  ##compute averaged person scores
  avg.personscore <- matrix(NA,nrow(personscores),1)
  for(i in seq(nrow(personscores))){
    avg.personscore[i,] <- sum(weights*personscores[i,])
  }
  ##compute averaged person SEs
  avg.personSE <- matrix(NA,nrow(personSEs),1)
  for(i in seq(nrow(personSEs))){
    avg.personSE[i,] <- sum(weights*sqrt(personSEs[i,]^2+(personscores[i,]-avg.personscore[i,])^2))
  }
  output <- list(weights,avg.personscore,avg.personSE)
  names(output) <- c("weights","Average person score","Average person SE")
  return(output)
}
```

ONLINE APPENDIX B: Generating parameters for illustration

Online Appendix B. Generating parameters for illustration

The generating model is a 3-parameter logistic (3-PL) bifactor model with two secondary dimensions. The probability of response "1" is given by:

$$P(y_{ji} = 1 | \theta_j, \theta_{jd}) = c_i + \frac{1 - c_i}{1 + \exp[-(\alpha_i \theta_j + \alpha_{id} \theta_{jd} - \beta_i)]},$$

where

y_{ji} is the item response (0 or 1) for item *i* and person *j*;

θ_j is the ability score for person *j* (primary dimension). It is generated from a standard normal distribution. It is the person score of substantive interest in our illustration;

θ_{jd} is the secondary dimension score for person *j*. Each item loads onto one of two secondary dimensions, *d* = 1 or 2. The first 10 items load on *d*=1 and next 10 load on *d*=2. The secondary dimension scores are not of substantive interest in our illustration;

β_i is the item difficulty for item *i*. It is generated from a standard normal distribution;

α_i is the (primary dimension) item discrimination for item *i*. It is generated from a log-normal distribution with $\mu = 0.08$ and $\sigma = 0.3$;

α_{id} is the (secondary dimension) item discrimination for item *i*. It is generated as .378 for all items, which induces an ECV (explained common variance; Reise, Bonifay & Haviland, 2013) for the primary dimension equal to .90, implying very weak secondary dimensions; and

c_i is the guessing parameter (lower asymptote) for item *i*. It is generated as .1 (for all items).

Reference

Reise, S. P., Bonifay, W. E., & Haviland, M. G. (2013). Scoring and modeling psychological measures in the presence of multidimensionality. *Journal of Personality Assessment*, *95*, 129-140.