Latent Trait Measurement Models for Other (not Binary) Responses

Today's Topics:

- Review of generalized models for categorical outcomes
- Ordered Categories → Graded Response
- Maybe Ordered Categories → Partial Credit
- Unordered Categories → Nominal Response
- Count Outcomes → Poisson and Negative Binomial
- Too many Zeros Outcomes → Zero-Inflated/Hurdle/Two-Part

Too Logit to Quit… https://www.youtube.com/watch?v=Yc43R8QnCXY

- The **logit** is also the basis for many other generalized models for predicting categorical outcomes

- Next we'll see how C possible response categories can be predicted using $C - 1$ binary "submodels" that involve carving up the categories in different ways, in which each binary submodel uses a logit link to predict its outcome

- Types of categorical outcomes:
  - Definitely ordered categories: "**cumulative logit**"
  - Maybe ordered categories: "**adjacent category logit**"
  - Definitely NOT ordered categories: "**generalized logit**"

Logit-Based Models for *C* Ordinal Categories

- Known as "**cumulative logit**" or "**proportional odds**" model in generalized models; will be known as "graded response model" in IRT
  - e.g., LINK=CLOGIT, DIST=MULT in SAS GLIMMIX (for non-latent-trait models)
- Models the probability of **lower vs. higher** cumulative categories via $C - 1$ submodels (e.g., if $C = 4$ possible responses of $c = 0,1,2,3$):

$\underbrace{\mathbf{0} \text{ vs. } \mathbf{1, 2,3}}_{\text{Submodel}_1}$ $\underbrace{\mathbf{0,1} \text{ vs. } \mathbf{2,3}}_{\text{Submodel}_2}$ $\underbrace{\mathbf{0,1,2} \text{ vs. } \mathbf{3}}_{\text{Submodel}_3}$

I've named these submodels based on what they predict, but programs will use their own names in the output.

- In SAS, what the binary submodels predict depends on whether the model is predicting **DOWN ($y_i = 0$, the default)** or **UP ($y_i = 1$)** **cumulatively**
- **Example predicting UP in an empty model (subscripts=parm,submodel)**
- Submodel 1: $\text{Logit}(y_i > 0) = \beta_{01}$ → $p(y_i > 0) = \exp(\beta_{01})/[1 + \exp(\beta_{01})]$
- Submodel 2: $\text{Logit}(y_i > 1) = \beta_{02}$ → $p(y_i > 1) = \exp(\beta_{02})/[1 + \exp(\beta_{02})]$
- Submodel 3: $\text{Logit}(y_i > 2) = \beta_{03}$ → $p(y_i > 2) = \exp(\beta_{03})/[1 + \exp(\beta_{03})]$

Logit-Based Models for C Ordinal Categories

- Models the probability of **lower vs. higher** cumulative categories via $C-1$ submodels (e.g., if $C=4$ possible responses of $c=0,1,2,3$):

$$\underbrace{\textbf{0}\text{ vs. }\textbf{1,2,3}}_{\text{Submodel}_1} \qquad \underbrace{\textbf{0,1}\text{ vs. }\textbf{2,3}}_{\text{Submodel}_2} \qquad \underbrace{\textbf{0,1,2}\text{ vs. }\textbf{3}}_{\text{Submodel}_3}$$

→ Prob_1 → Prob_2 → Prob_3

$$\boxed{\begin{array}{l} \text{Logit}(\mathrm{y_i} > 2) = \beta_{03} \\ \rightarrow p(\mathrm{y_i} > 2) = \dfrac{\exp(\beta_{03})}{1+\exp(\beta_{03})} \end{array}}$$

- What the binary submodels predict depends on whether the model is predicting **DOWN ($\mathbf{y_i = 0}$, the default)** or **UP ($\mathbf{y_i = 1}$) cumulatively**
  - Either way, the model predicts the middle category responses ***indirectly***

- **Example if predicting UP with an empty model:**
  - Probability of 0 = $1 - \text{Prob}_1$
Probability of 1 = $\text{Prob}_1 - \text{Prob}_2$
Probability of 2 = $\text{Prob}_2 - \text{Prob}_3$
Probability of 3 = $\text{Prob}_3 - 0$

The cumulative submodels that create these probabilities are each estimated using **all the data** (good, especially for categories not chosen often), but **assume order in doing so** (may be bad or ok, depending on your response format).

Logit-Based Models for C Categories

- Uses **multinomial distribution for residuals**, whose PDF for $C = 4$ categories of $c = 0,1,2,3$, an observed $y_i = c$, and indicators I if $c = y_i$

Only p_{ic} for the response $y_i = c$ gets used

$$f(\mathrm{y_i} = \mathrm{c}) = p_{\mathrm{i0}}^{\mathrm{I[y_i=0]}} p_{\mathrm{i1}}^{\mathrm{I[y_i=1]}} p_{\mathrm{i2}}^{\mathrm{I[y_i=2]}} p_{\mathrm{i3}}^{\mathrm{I[y_i=3]}}$$

  - Maximum likelihood is then used to find the most likely parameters in the model to predict the probability of each response through the (usually logit) link function; the probabilities sum to 1: $\sum_{\mathrm{c=1}}^{\mathrm{C}} p_{\mathrm{ic}} = 1$

- Other models for categorical data that use the multinomial:
  - Adjacent category logit (partial credit): Models the probability of **each next highest** category via $C - 1$ submodels (e.g., if $C = 4$):

    0 vs. **1** **1** vs. **2** **2** vs. **3**

  - Baseline category logit (nominal): Models the probability of **reference vs. other** category via $C - 1$ submodels (e.g., if $C = 4$ and $0 = \mathrm{ref}$):

    0 vs. **1** **0** vs. **2** **0** vs. **3**

In **nominal** models, all parameters are estimated **separately** per submodel

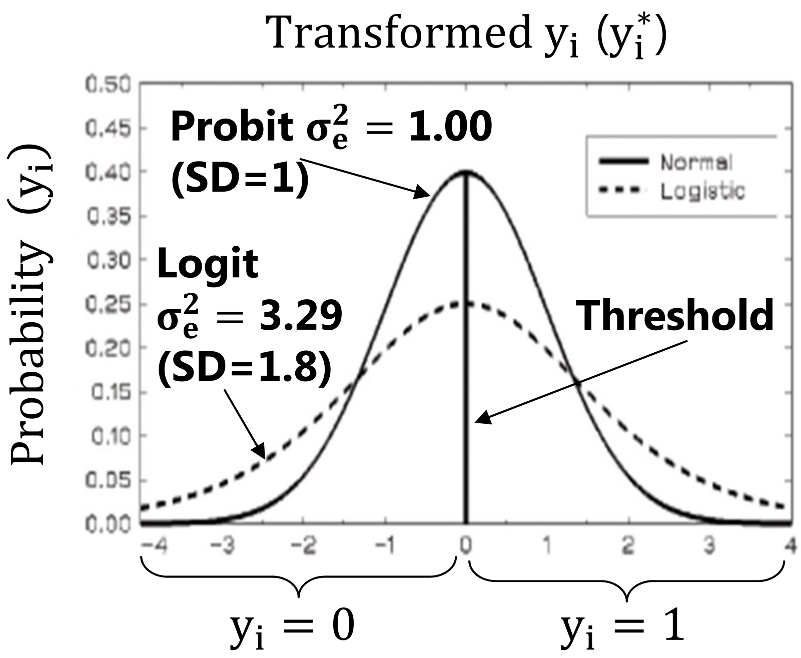
Polytomous Items = Categorical Outcomes

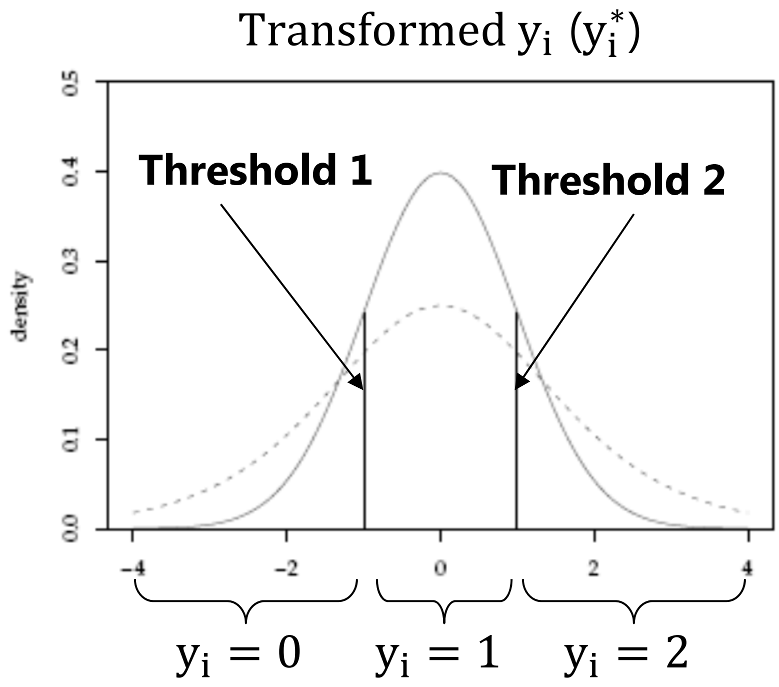
- **Polytomous = more than 2 options**
- Polytomous models are not numbered like binary models, but instead get called different names (acronyms)
  - Most have a 1-PL vs. 2-PL version that also go by different names
  - Within each, different constraints on what to do with multiple options

- **Three main kinds of polytomous models:**
  - Response options are ordered for sure → Cumulative Logit
    - Graded Response or Modified Graded Response Model
  - Response options may be ordered→ Adjacent Category Logit
    - (Generalized) Partial Credit Model or Rating Scale Model
  - No way are these response options ordered → Baseline Category Logit
    - Nominal Response Model: Notably, this can be constrained to fit combo of partial credit and nominal responses (e.g., for "not applicable" responses)

The Threshold Concept

$$\mathbf{y_i^*} = \textbf{\textit{threshold}} + \textbf{rest of your model} + \mathbf{e_i}$$





- Each categorical variable is **really** the chopped-up version of a pretend underlying continuous variable ($\mathbf{y}^*$) with mean = 0 (variance = 1.00 or 3.29)
- Polytomous models will differ in how they make use of multiple thresholds per item in which C = # categories, so **# thresholds per item = *k* = C−1**

Graded Response Model for Ordinal Categories

- Ideal for items with clear underlying response continuum (e.g., Likert)
- **# response options don't have to be the same across items**
- GRM is an "indirect" or "difference" model
  - Compute difference between models to get probability of each response
- Estimate 1 $\mathrm{a_i}$ per item and $k = \mathrm{C} - 1$ difficulties (4 options → 3 difficulties)
- Models the probability of **lower vs. higher** cumulative categories via k submodels (e.g., if $C = 4$ possible responses of $c = 0{,}1{,}2{,}3$):

$$\underbrace{\textbf{0}\text{ vs. }\textbf{1, 2,3}}_{\text{Submodel}_1} \qquad \underbrace{\textbf{0,1}\text{ vs. }\textbf{2,3}}_{\text{Submodel}_2} \qquad \underbrace{\textbf{0,1,2}\text{ vs. }\textbf{3}}_{\text{Submodel}_3}$$

Mplus will refer to these thresholds using $1, $2, $3 after the item name

- Each submodel is estimated using all the data cumulatively (assumes order)
- As with binary items, Mplus estimates the IFA model directly (loadings and thresholds), but unlike binary items, it won't do the IRT conversion into discriminations and difficulties (but you can write NEW parameters or see my spreadsheet for help)

Example Graded Response Model (GRM)

IRT version of the GRM: Note that a_i is the same across submodels…

- **0** vs. **1,2,3**: $\text{Logit}(y_{is} > 0) = a_i(\theta_s - b_{1i})$ → $p1(y_{is} > 0) = \frac{\exp(a_i(\theta_s - b_{1i}))}{1+\exp(a_i(\theta_s - b_{1i}))}$
- **0,1** vs. **2,3**: $\text{Logit}(y_{is} > 1) = a_i(\theta_s - b_{2i})$ → $p2(y_{is} > 1) = \frac{\exp(a_i(\theta_s - b_{2i}))}{1+\exp(a_i(\theta_s - b_{2i}))}$
- **0,1,2** vs. **3**: $\text{Logit}(y_{is} > 2) = a_i(\theta_s - b_{3i})$ → $p3(y_{is} > 2) = \frac{\exp(a_i(\theta_s - b_{3i}))}{1+\exp(a_i(\theta_s - b_{3i}))}$

IFA version of the GRM—what is directly estimated in Mplus:

- **0** vs. **1,2,3**: $\text{Logit}(y_{is} > 0) = -\tau_{1i} + \lambda_i F_s$ → $p1(y_{is} > 0) = \frac{\exp(-\tau_{1i}+\lambda_i F_s)}{1+\exp(-\tau_{1i}+\lambda_i F_s)}$
- **0,1** vs. **2,3**: $\text{Logit}(y_{is} > 1) = -\tau_{2i} + \lambda_i F_s$ → $p2(y_{is} > 1) = \frac{\exp(-\tau_{2i}+\lambda_i F_s)}{1+\exp(-\tau_{2i}+\lambda_i F_s)}$
- **0,1,2** vs. **3**: $\text{Logit}(y_{is} > 2) = -\tau_{3i} + \lambda_i F_s$ → $p3(y_{is} > 2) = \frac{\exp(-\tau_{3i}+\lambda_i F_s)}{1+\exp(-\tau_{3i}+\lambda_i F_s)}$

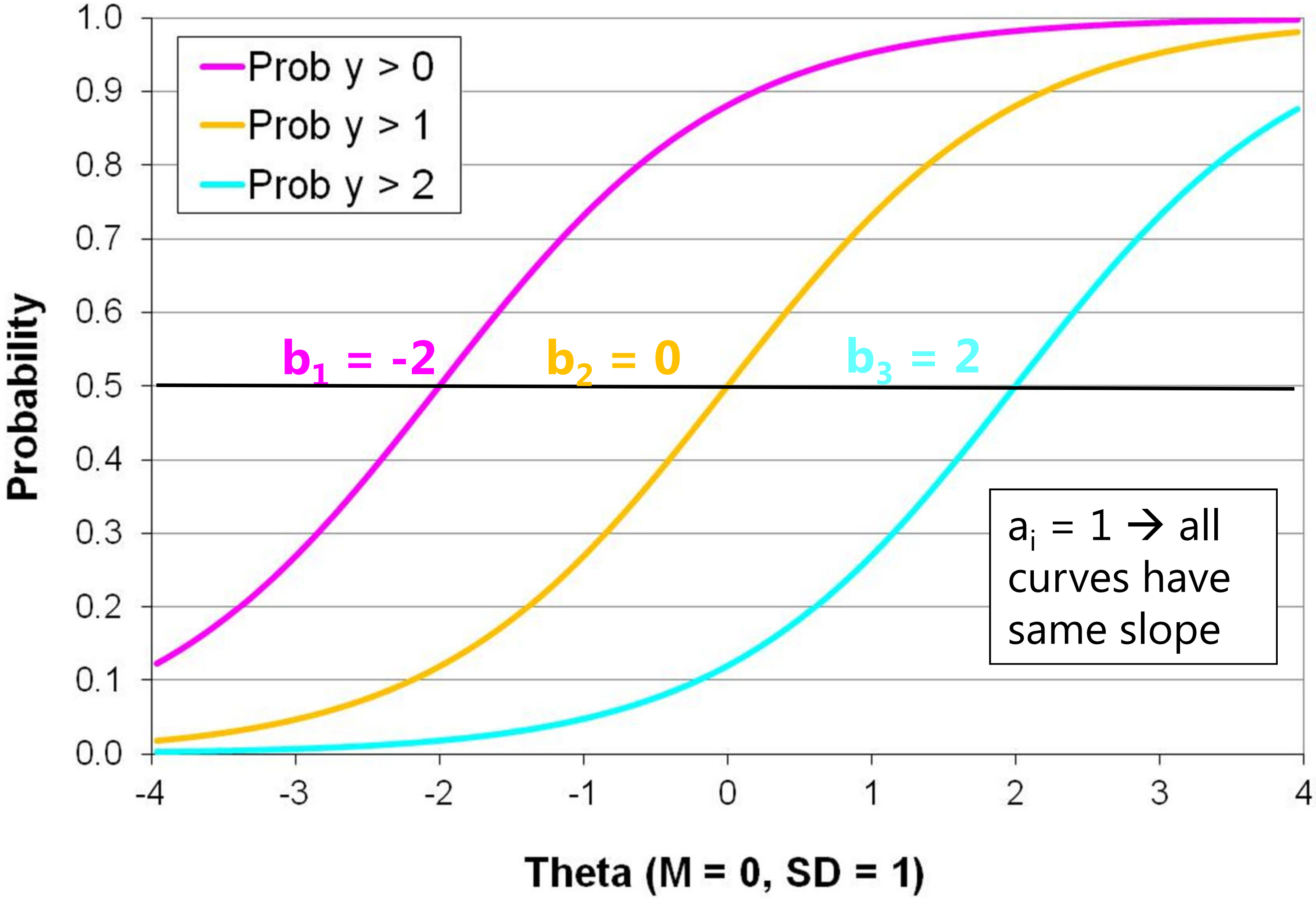
GRM indirectly predicts probability of each category

- $p(y_{is} = 0) = 1 - p1$
  $p(y_{is} = 1) = p1 - p2$
  $p(y_{is} = 2) = p2 - p3$
  $p(y_{is} = 3) = p3 - 0$

IRT b_{ki} = trait level needed for a 50% probability (logit =0) of the **higher binary category**

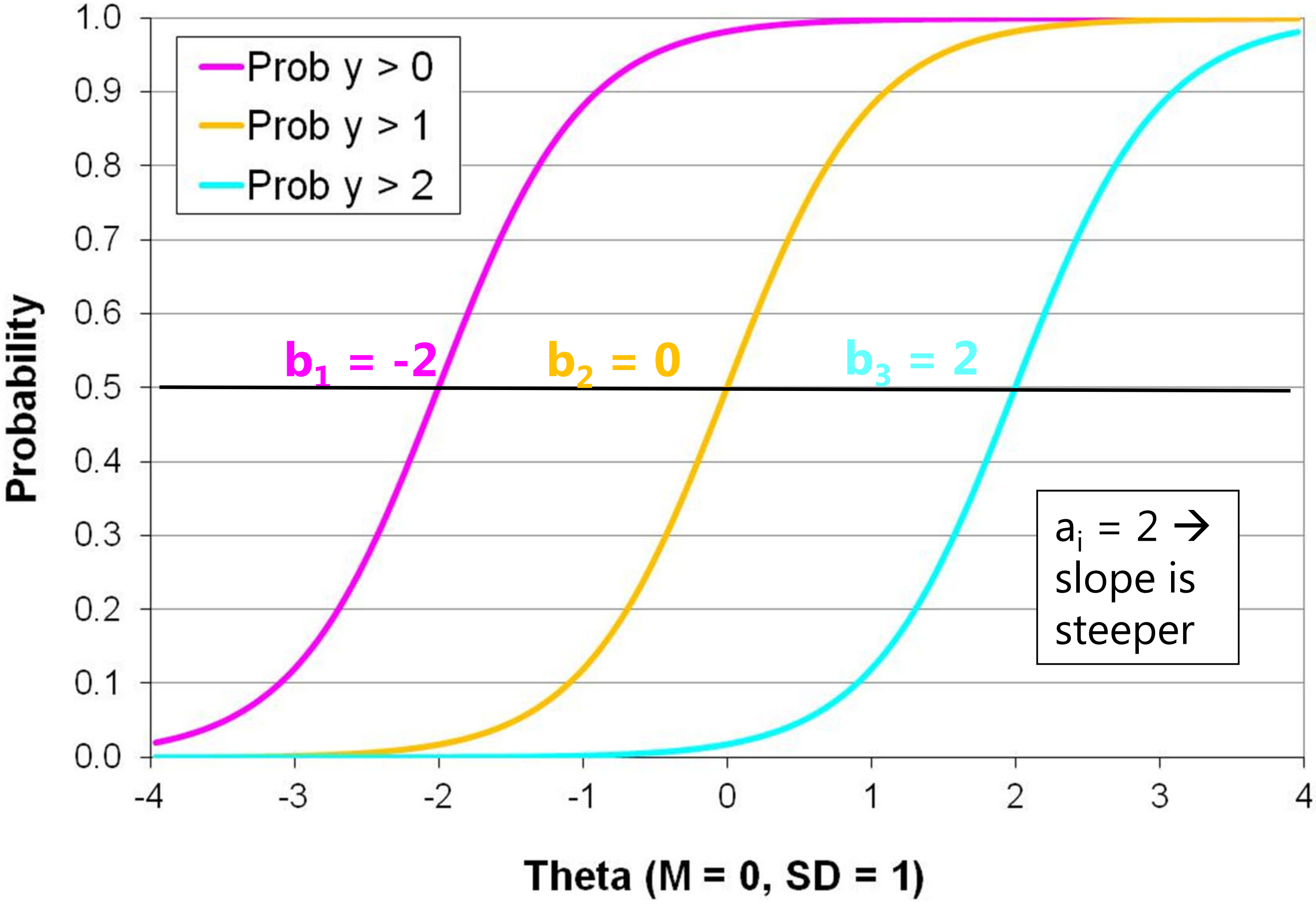
IFA τ_{ki} = logit of the probability of the **lower binary category** when Factor = 0

Cumulative Item Response Curves: GRM for 4-Category (0123) Item, a_i=1



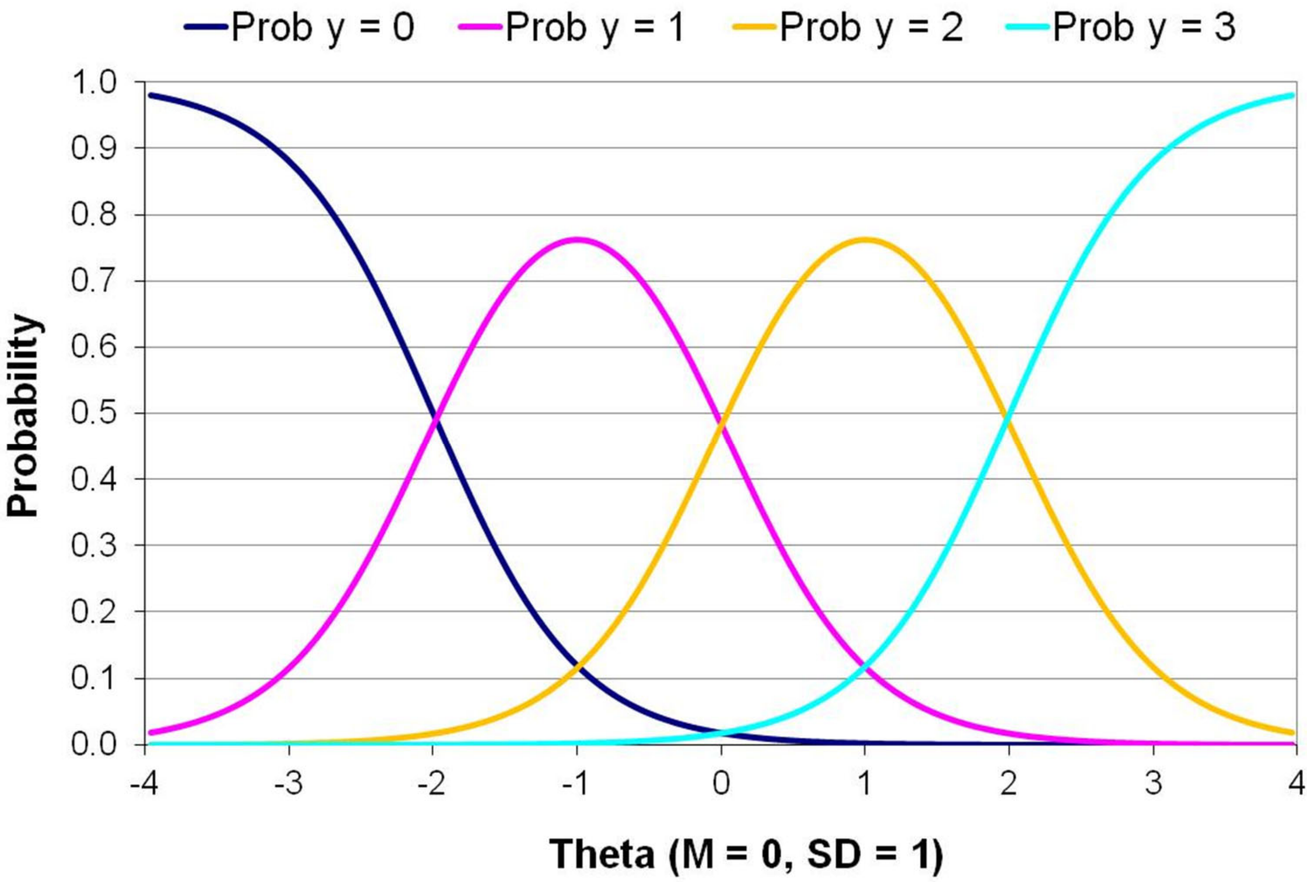
a_i = 1 → all curves have same slope

Cumulative Item Response Curves: GRM for 4-Category (0123) Item, a_i=**2**



a_i = 2 → slope is steeper

Category Response Curves: GRM for 4-Category (0123) Item, a_i=2

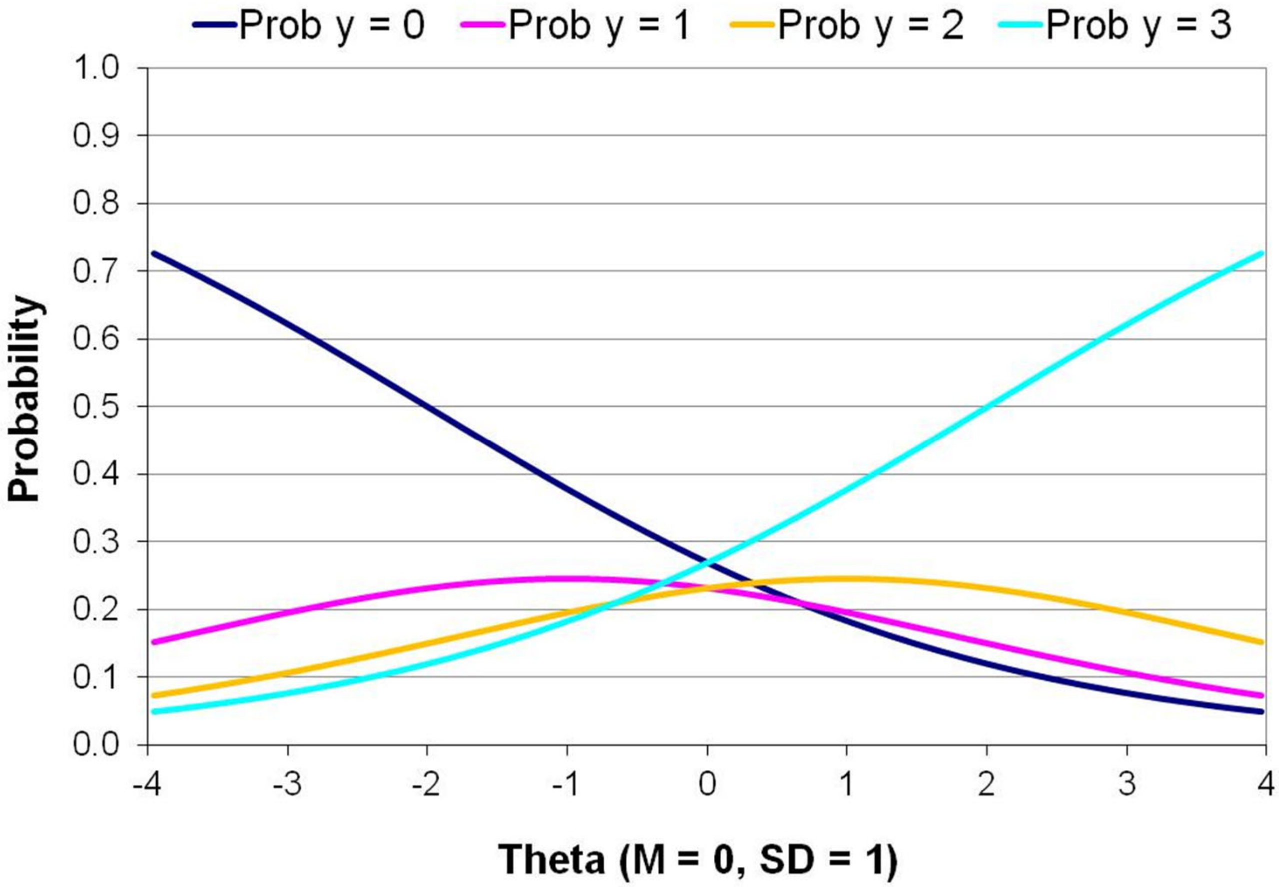


GRM indirectly predicts the probability of each category response across Theta:

$$
\begin{aligned}
p(\mathrm{y}_{\mathrm{is}} = 0) &= \ \ 1 - p1 \\
p(\mathrm{y}_{\mathrm{is}} = 1) &= p1 - p2 \\
p(\mathrm{y}_{\mathrm{is}} = 2) &= p2 - p3 \\
p(\mathrm{y}_{\mathrm{is}} = 3) &= p3 - 0
\end{aligned}
$$

Mplus will make this type of Plot for you under the title “Item Characteristic Curve”

Category Response Curves: GRM for 4-Category (0123) Item, a_i=.5

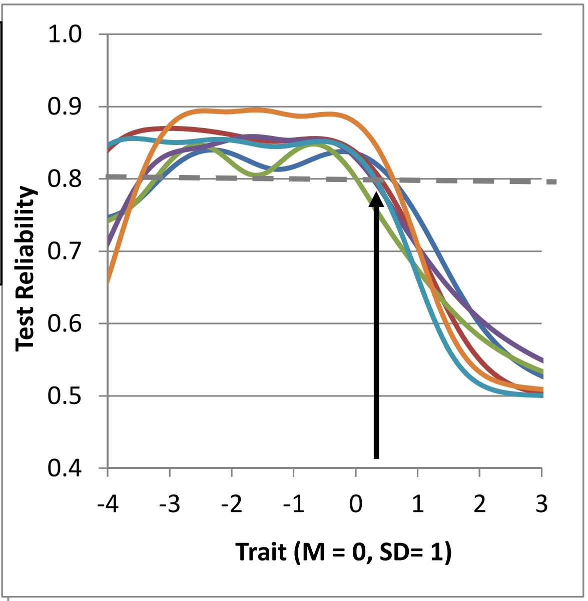


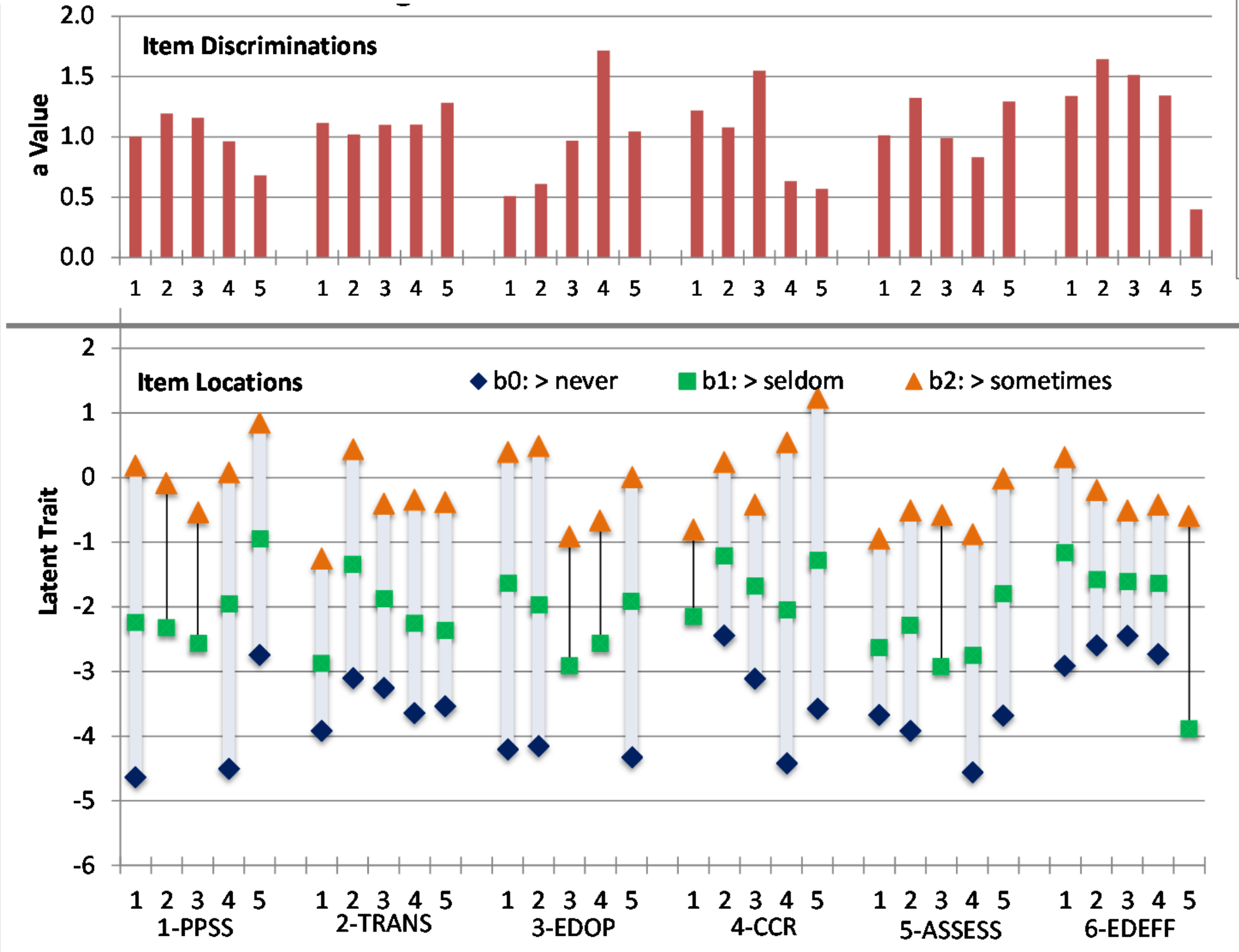
This is exactly what you do NOT want to see.

Although they are ordered, the middle categories are worthless (not differentiated).

Top Panel: Red bars display item discriminations.

Bottom Panel: Item locations for 4 response options: *Never*, *Seldom*, *Sometimes*, and *Often*





The location of y > *sometimes* ranges from ~0 to .5 across items, creating lower reliability for traits>.5

Being able to select a higher option (e.g., *almost always*) would improve reliability for high-trait respondents.

"Modified" ("Rating Scale") Graded Response Model (MGRM)

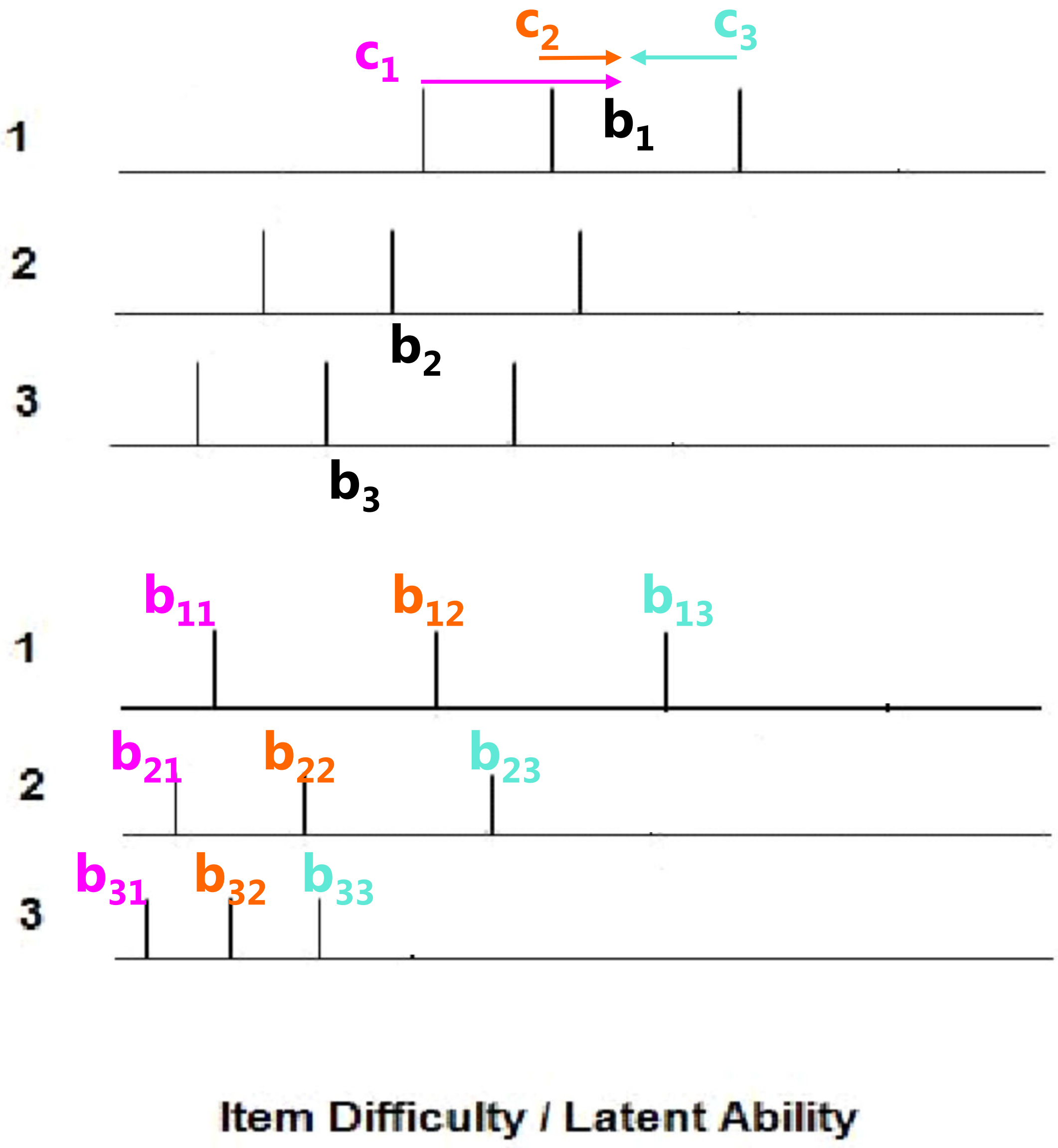
- More parsimonious version of GRM for items with **same response options**
- In GRM, *k**#items difficulties + 1 discrimination are estimated per item
- In MGRM, each item gets own slope and own overall 'location' parameter, but the differences between categories around that location are constrained to be equal across items (get a "c" shift for each threshold)
  - So, different a_i and b_i per item, but same c_1, c_2, and c_3 across items (**one c = 0**)
  - Not directly available within Mplus, but can be done using threshold constraints

- **0** vs. **1,2,3**: $\text{Logit}(y_{is} > 0) = a_i(\theta_s - b_i + c_1) \rightarrow p1(y_{is} > 0) = \frac{\exp(a_i(\theta_s - b_i + c_1))}{1 + \exp(a_i(\theta_s - b_i + c_1))}$

- **0,1** vs. **2,3**: $\text{Logit}(y_{is} > 1) = a_i(\theta_s - b_i + c_2) \rightarrow p2(y_{is} > 1) = \frac{\exp(a_i(\theta_s - b_i + c_2))}{1 + \exp(a_i(\theta_s - b_i + c_2))}$

- **0,1,2** vs. **3**: $\text{Logit}(y_{is} > 2) = a_i(\theta_s - b_i + c_3) \rightarrow p3(y_{is} > 2) = \frac{\exp(a_i(\theta_s - b_i + c_3))}{1 + \exp(a_i(\theta_s - b_i + c_3))}$

Modified vs. Regular GRM



Modified GRM → **k "shift" c terms:** All category distances are same across items, and each item gets its own "$\mathbf{b_i}$" overall location

Original GRM → **k locations per item**: All category distances differ across items

Summary of Models for Ordered Categorical Responses

Some of these in Mplus via "CATEGORICAL ARE"	**Difficulty Per Item Only (category distances equal)**	**Difficulty Per Category Per Item**
Equal discrimination across items (1-PLish)?	(possible, but no special name)	(possible, but no special name)
Unequal discriminations (2-PLish)?	"Modified GRM" or "Rating Scale GRM" (same response options)	"Graded Response Model" "Cumulative Logit"

- GRM and Modified GRM are reliable models for ordered categorical data
  - Commonly used in real-world testing; most stable to use in practice
  - Least data demand because all data get used in estimating each b_{ki}
  - Only *major* deviations from the model will end up causing problems such that categories should be allowed to be unordered (perhaps bimodal distributions)

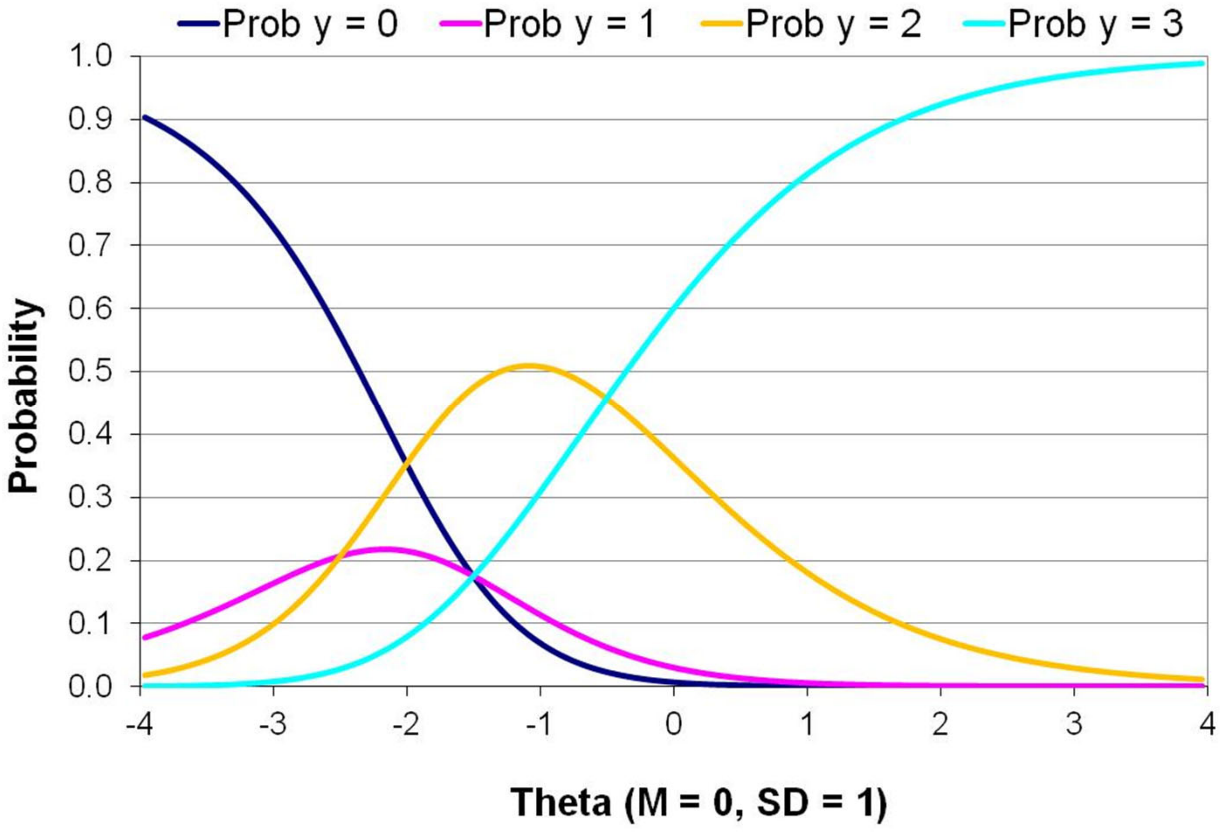
(Generalized) Partial Credit Model (PCM)

- When you want to **test** an assumption of an ordered underlying continuum
- **# response options doesn't have to be same across items**, but there is no guarantee that every category will be most likely at some point
- Is a "direct, divide-by-total" model (prob of each category is given directly)
- Estimate *k* difficulty-like $\boldsymbol{\delta_{ki}}$ **"step" parameters**, which are the Theta values at which the next category becomes **more likely** (not necessarily 50%)
- Models the probability of **adjacent categories** ("adjacent category logit")
  - Divide item into a series of binary items, but without order constraints beyond adjacent categories because it only uses those 2 categories ($c = 0{,}1{,}2{,}3$):

IRT for PCM (now in Mplus): Is "generalized" if $\mathbf{a_i}$ is used instead of $\mathbf{a}$

- **If 0** or **1**: $\text{Logit}(y_{is} = 1) = a_i(\theta_s - \delta_{1i}) \rightarrow p1(y_{is} = 1) = \frac{\exp(a_i(\theta_s - \delta_{1i}))}{1+\exp(a_i(\theta_s - \delta_{1i}))}$
- **If 1** or **2**: $\text{Logit}(y_{is} = 2) = a_i(\theta_s - \delta_{2i}) \rightarrow p2(y_{is} = 2) = \frac{\exp(a_i(\theta_s - \delta_{2i}))}{1+\exp(a_i(\theta_s - \delta_{2i}))}$
- **If 2** or **3**: $\text{Logit}(y_{is} = 3) = a_i(\theta_s - \delta_{3i}) \rightarrow p3(y_{is} = 3) = \frac{\exp(a_i(\theta_s - \delta_{3i}))}{1+\exp(a_i(\theta_s - \delta_{3i}))}$

Category Response Curves: PCM for 4-Category (0123) Item, a_i=1

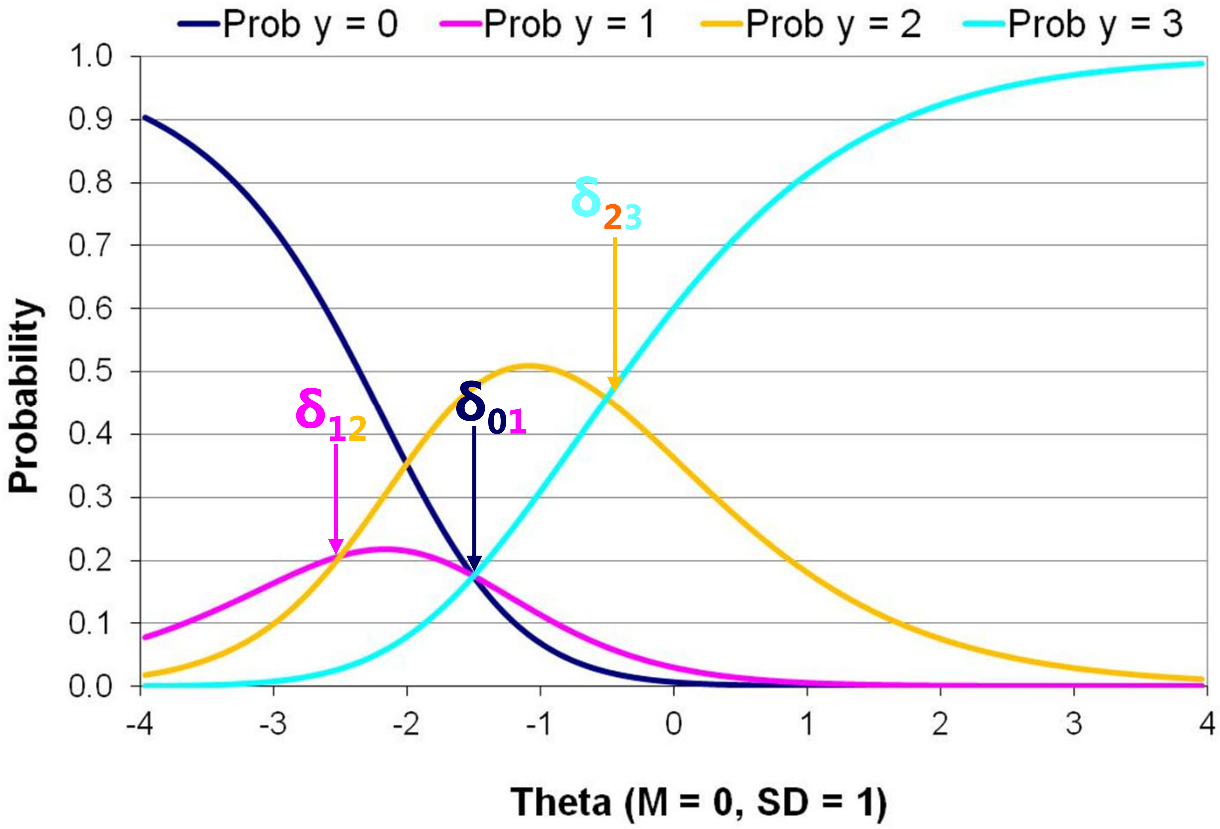


PCM directly predicts the probability of each category response across Theta:

$$p(\mathrm{y_{is}} = 0) = 1 - p1$$
$$p(\mathrm{y_{is}} = 1) = p1$$
$$p(\mathrm{y_{is}} = 2) = p2$$
$$p(\mathrm{y_{is}} = 3) = p3$$

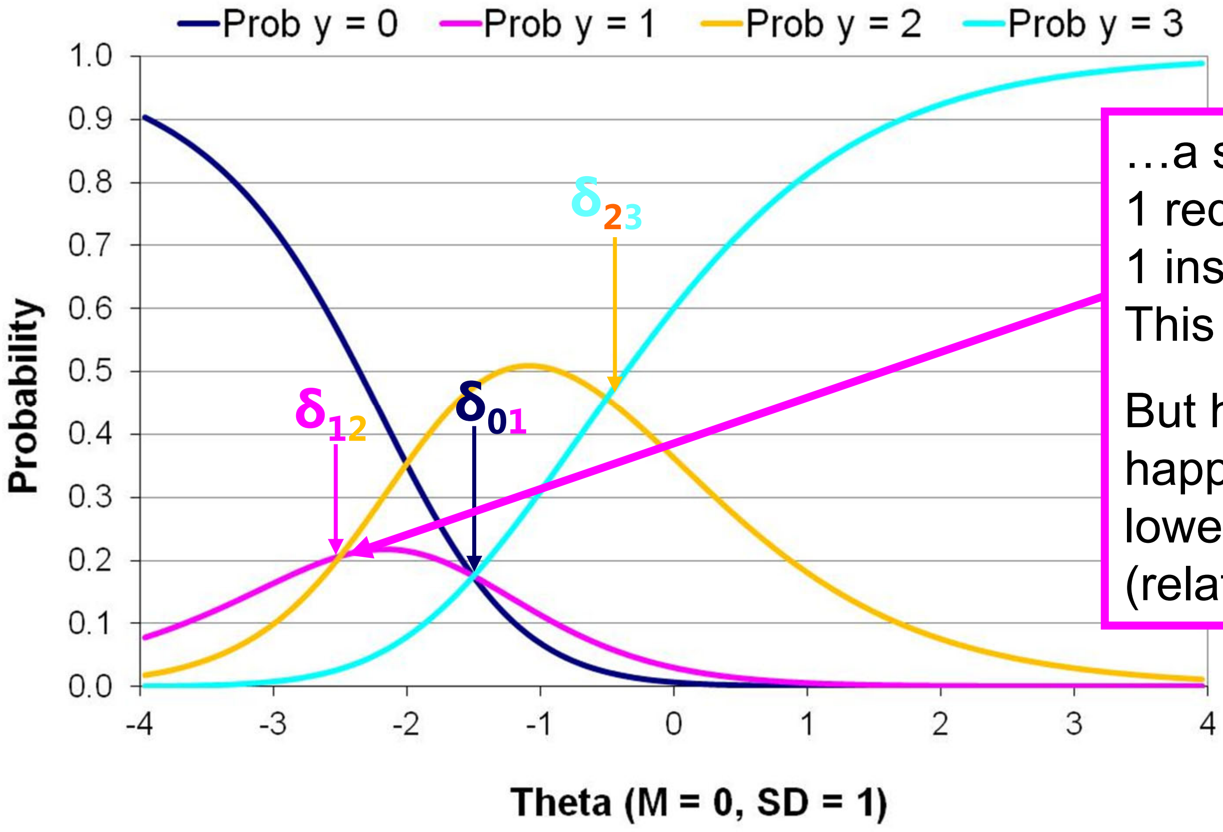
These curves look similar to the GRM, but the location parameters are interpreted differently because they are NOT cumulative, they are only adjacent…

Category Response Curves: PCM for 4-Category (0123) Item, a_i=1



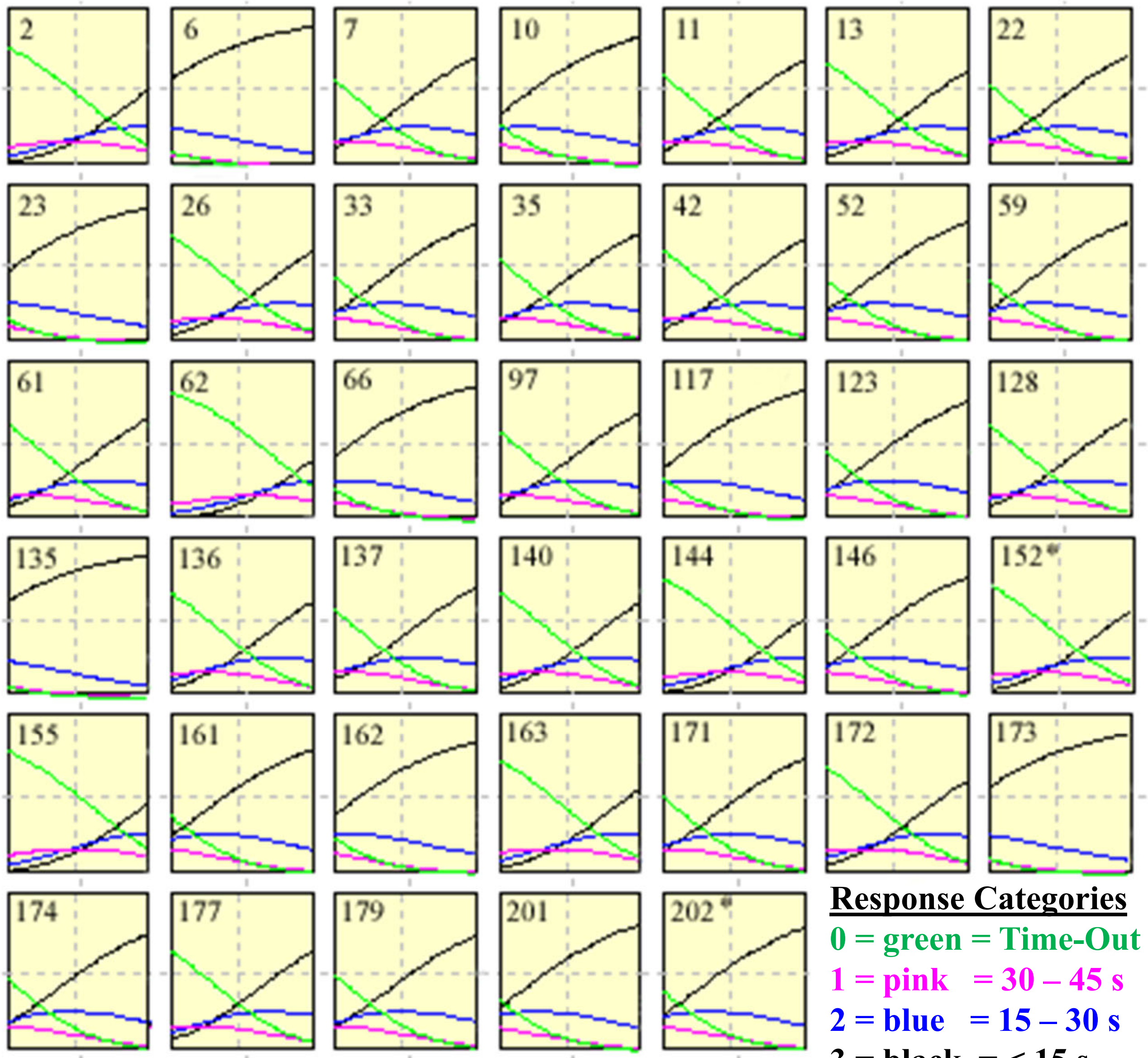
The $\boldsymbol{\delta_{ki}}$ terms are the location where the next category response becomes more likely (not necessarily 50%).

Category Response Curves: PCM for 4-Category (0123) Item, a_i=1



…a score of 2 instead of 1 requires less Theta than 1 instead of 0 … This is called a 'reversal'

But here, this likely happens because of a lower frequency of y = 1 (relative to y = 0)



Response Categories

0 = green = Time-Out

1 = pink = 30 – 45 s

2 = blue = 15 – 30 s

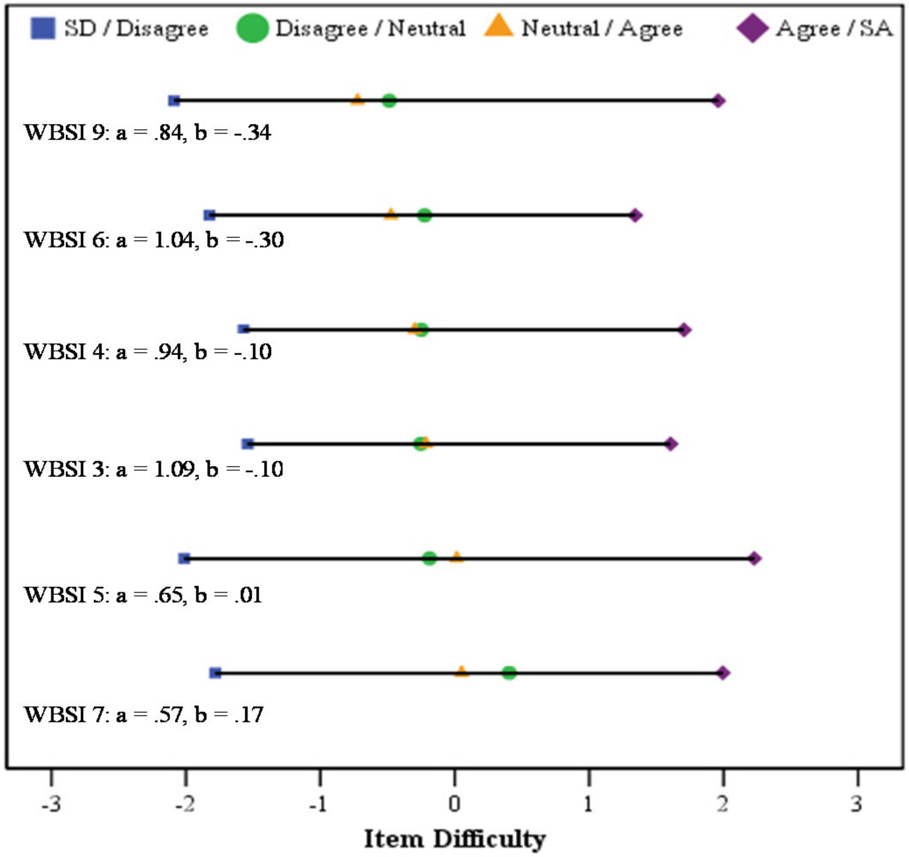
3 = black = < 15 s

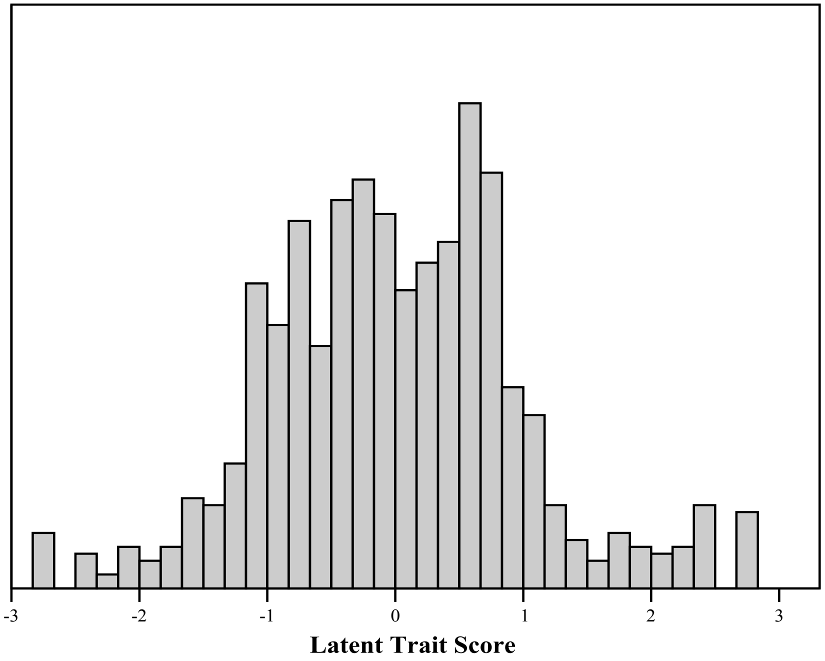
*Misfit ($p < .05$)

More of what you don’t want to see… category response curves from a PCM where reversals are a plenty…

…and the middle categories are fairly useless.

Generalized Partial Credit Model Example: General Intrusive Thoughts (5 options)

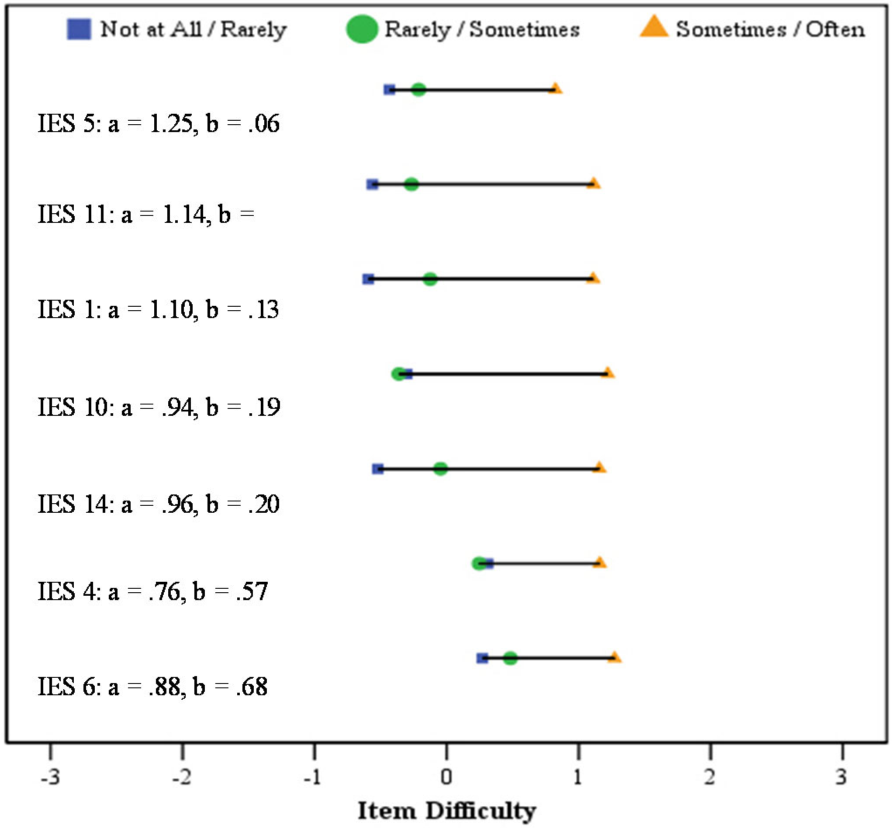




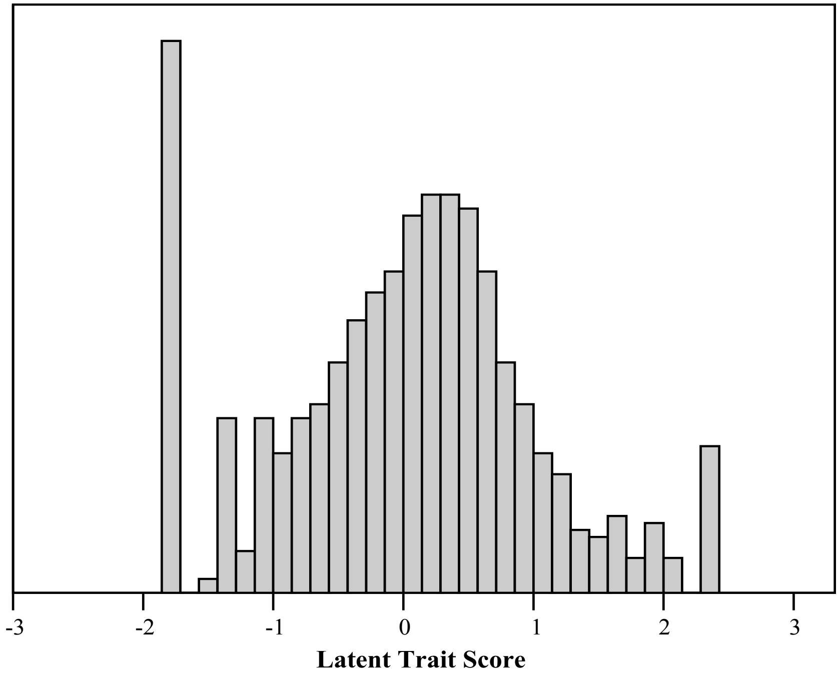
Note that the 4 thresholds **cover a wide range** of the latent trait, and what the distribution of Theta looks like as a result...

But the middle 3 categories are used infrequently and/or are not differentiable

Generalized Partial Credit Model Example: Event-Specific Intrusive Thoughts (4 options)



Note that the 3 thresholds **do not cover a wide range** of the latent trait, and what the distribution of theta looks like as a result…



Rating Scale Model (RSM)

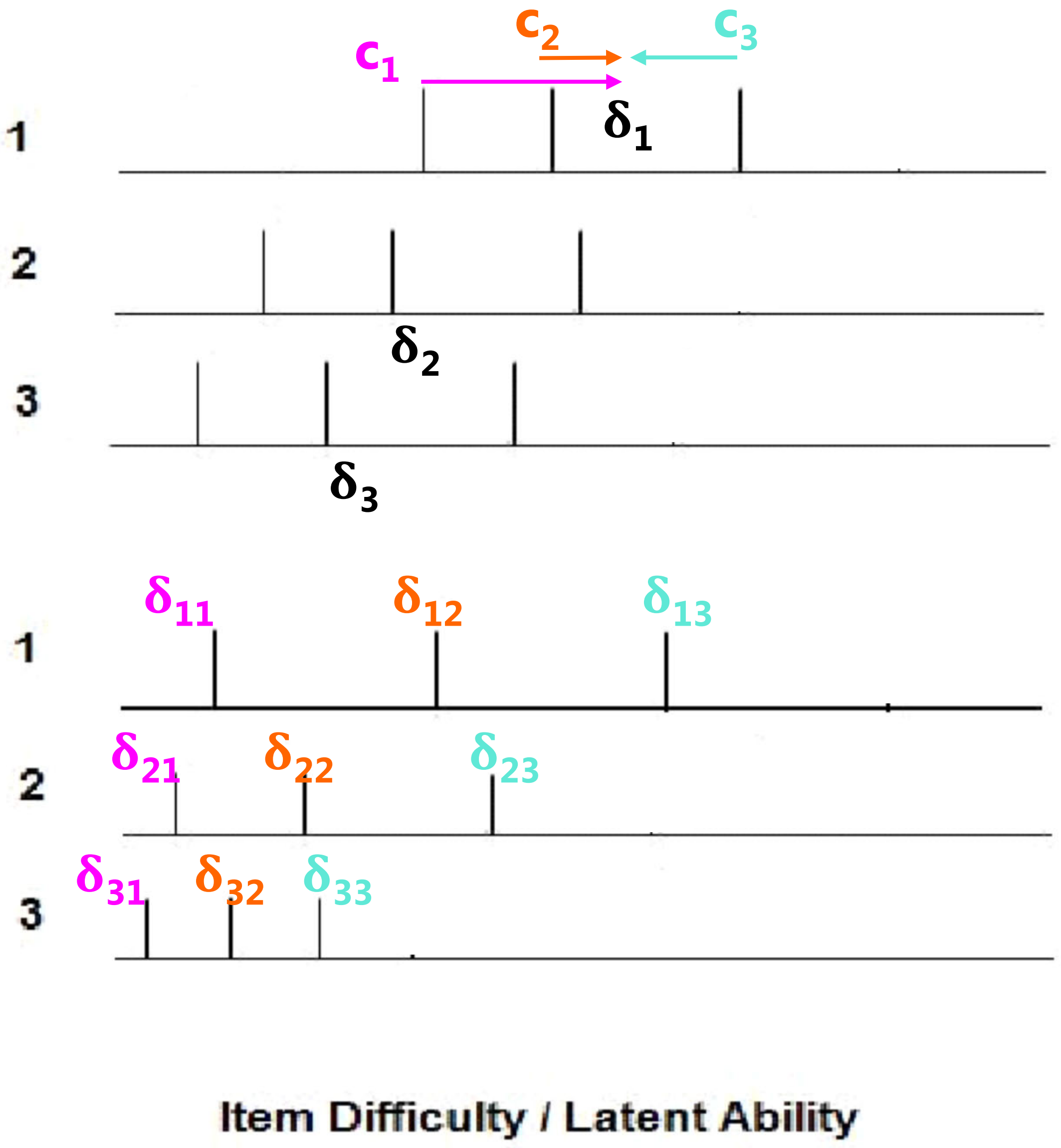
- More parsimonious version of PCM for items with **same response format**
- In PCM, *k**#items step parameters +1 discrimination are estimated per item
- In RSM, each item gets own slope and own overall 'location' parameter, but the differences between categories around that location are constrained to be equal across items (get a "c" shift for each threshold; **one c = 0**)
  - So, different b_i (and possibly a_i) per item, but same c_1, c_2, and c_3 across items
  - Also not available within Mplus

- **If 0 or 1**: $\text{Logit}(y_{is}=1)=a(\theta_s-\delta_i+c_1) \rightarrow p1(y_{is}=1)=\frac{\exp(a(\theta_s-\delta_i+c_1))}{1+\exp(a(\theta_s-\delta_i+c_1))}$

- **If 1 or 2**: $\text{Logit}(y_{is}=2)=a(\theta_s-\delta_i+c_1) \rightarrow p2(y_{is}=2)=\frac{\exp(a(\theta_s-\delta_i+c_2))}{1+\exp(a(\theta_s-\delta_i+c_2))}$

- **If 2 or 3**: $\text{Logit}(y_{is}=3)=a(\theta_s-\delta_i+c_1) \rightarrow p3(y_{is}=3)=\frac{\exp(a(\theta_s-\delta_i+c_3))}{1+\exp(a(\theta_s-\delta_i+c_3))}$

Rating Scale vs. Regular PCM



Rating Scale→
k "shift" c terms: All category distances are same across items, and each item gets its own "δ_i" overall location

Original PCM →
***k* locations per item**: All category distances differ across items

Summary of Models for Maybe-Ordered Categorical Responses

Not directly available in Mplus	**Difficulty Per Item Only (category distances equal)**	**Difficulty Per Category Per Item**
Equal discrimination across items (1-PLish)?	"Rating Scale PCM"	"Partial Credit Model"
Unequal discriminations (2-PLish)?	"Generalized Rating Scale PCM"?? (same response options)	"Generalized PCM" "Adjacent Category Logit"

- Partial Credit Models test the assumption of ordered categories
  - This can be useful for item screening, but perhaps not for actual analysis
- These models have additional data demands relative to GRM
  - Only data from that threshold get used (i.e., for 1 vs. 2, 0 and 3 don't contribute)
  - So larger sample sizes are needed to identify all model parameters
  - Sometimes categories have to be consolidated to get the model to not blow up

Nominal Response Model for Unordered Categories

- Ideal for items with no ordering of any kind (e.g., favorite color)
- **# response options don't have to be the same across items**
- NRM is a "direct" model → prob of each category given directly
- Estimate k a_i per item and k "intercepts" per item
- Models the probability of **each** category relative to a baseline category (sort of like dummy-coding the outcome variable, here baseline = 0):

$$\underbrace{0 \text{ vs. } 1}_{\text{Submodel}_1} \qquad \underbrace{0 \text{ vs. } 2}_{\text{Submodel}_2} \qquad \underbrace{0 \text{ vs. } 3}_{\text{Submodel}_3}$$

Mplus will refer to these "intercepts" using #1, #2, #3 after the item name

- Available in Mplus with NOMINAL ARE option (estimated as an IFA model)
- Can be useful to examine utility of distractors in multiple choice tests
- Can also be constrained to become the GPCM for Likert-scale responses and stay nominal for unordered responses (like "not applicable")

Example Nominal Response Model (NRM)

IRT version of the NRM: Note that $\mathrm{a_i}$ is NOT the same across submodels…

- **If 0** or **1**: $\mathrm{Logit}(y_{is} = 1) = a_{1i}(\theta_s - b_{1i}) \rightarrow p1(y_{is} = 1) = \frac{\exp(a_{1i}(\theta_s + b_{1i}))}{1+\exp(a_{1i}(\theta_s + b_{1i}))}$
- **If 0** or **2**: $\mathrm{Logit}(y_{is} = 2) = a_{2i}(\theta_s - b_{2i}) \rightarrow p2(y_{is} = 2) = \frac{\exp(a_{2i}(\theta_s + b_{2i}))}{1+\exp(a_{2_i}(\theta_s + b_{2i}))}$
- **If 0** or **3**: $\mathrm{Logit}(y_{is} = 3) = a_{3i}(\theta_s - b_{3i}) \rightarrow p3(y_{is} = 3) = \frac{\exp(a_{3i}(\theta_s + b_{3i}))}{1+\exp(a_{3i}(\theta_s + b_{3i}))}$

IFA version of the GRM—what is estimated in Mplus (with intercepts):

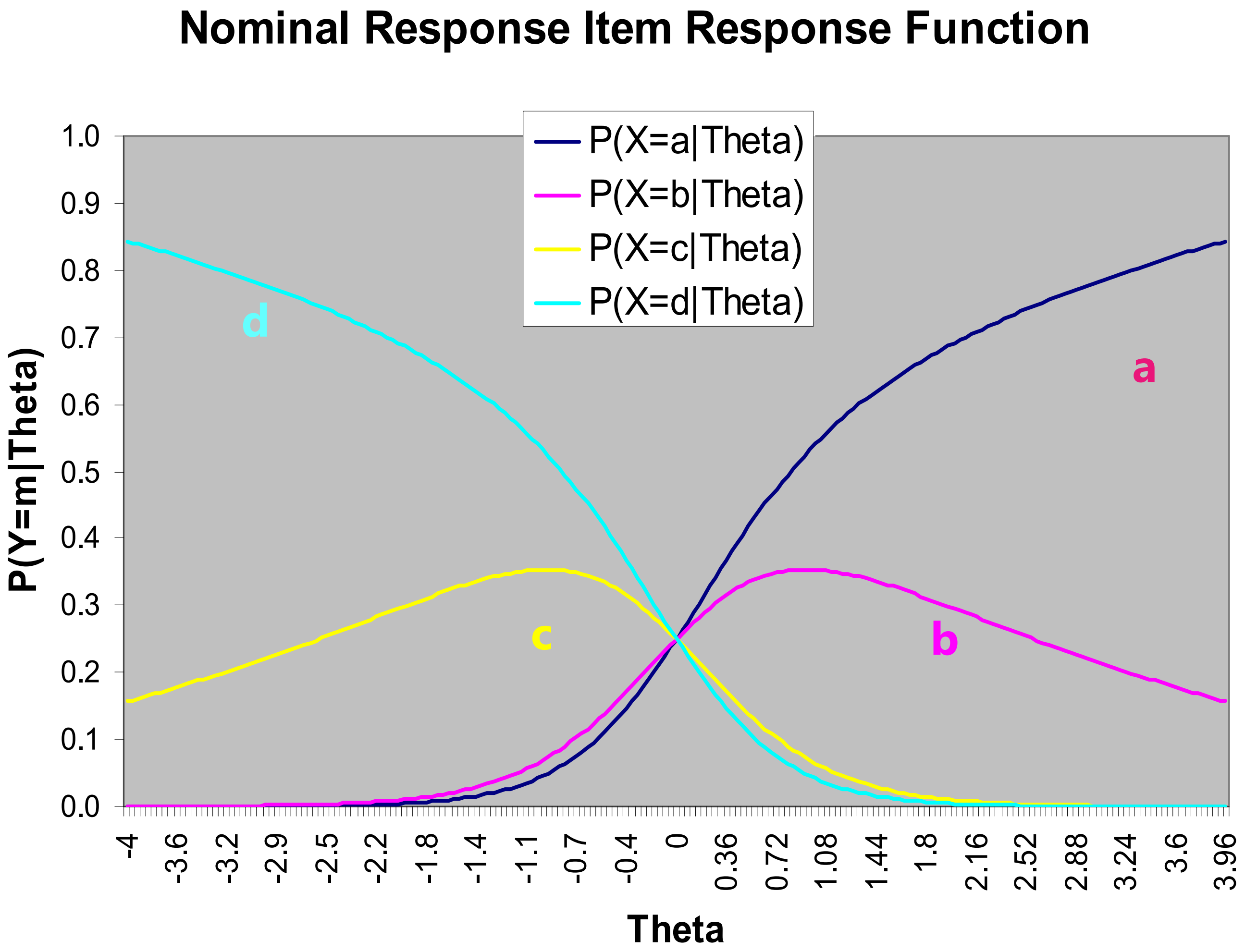
- **If 0** or **1**: $\mathrm{Logit}(y_{is} = 1) = -\tau_{1i} + \lambda_{1i}F_s \rightarrow p1(y_{is} = 1) = \frac{\exp(\mu_{1i} + \lambda_{1_i}F_s)}{1+\exp(\mu_{1i} + \lambda_{1_i}F_s)}$
- **If 0** or **2**: $\mathrm{Logit}(y_{is} = 2) = -\tau_{2i} + \lambda_{2i}F_s \rightarrow p2(y_{is} = 2) = \frac{\exp(\mu_{2i} + \lambda_{2_i}F_s)}{1+\exp(\mu_{2i} + \lambda_{2_i}F_s)}$
- **If 0** or **3**: $\mathrm{Logit}(y_{is} = 3) = -\tau_{3i} + \lambda_{3i}F_s \rightarrow p3(y_{is} = 3) = \frac{\exp(\mu_{3i} + \lambda_{3_i}F_s)}{1+\exp(\mu_{3i} + \lambda_{3_i}F_s)}$

GRM directly predicts probability of each category

- $p(y_{is} = 0) = 1 - p1 + p2 + p3$
  $p(y_{is} = 1) = p1$
  $p(y_{is} = 2) = p2$
  $p(y_{is} = 3) = p3$

IRT $\mathbf{b_{ki}}$ = trait level needed for a 50% prob (logit =0) of the **lower binary category**
IFA $\boldsymbol{\mu_{ki}}$ = logit of the probability of the **higher binary category** when Factor = 0

Category Response Curves (NRM for 5-Category Item)



Example Analysis of Multiple Choice Distractors:

People low in Theta are most likely to pick **d**, but **c** is their second choice

People high in Theta are most likely to pick **a**, but **b** is their second choice

Summary: Polytomous Models

Many kinds of polytomous IRT/IFA models…

- Some **assume order** of response options…
  - **Graded Response Model Family** → "cumulative logit model"
    - Models cumulative change in categories using all data for each submodel
- Some allow you to **test order** of response options…
  - **(Generalized) Partial Credit Model Family** → "adjacent category logit model"
    - Models adjacent category thresholds only, so they allow you to see reversals (empirical mis-ordering of your response options with respect to Theta)
    - (G)PCM useful for identifying separability and adequacy of categories
- Some assume **no order** of response options…
  - **Nominal Model** → "baseline category logit model"
    - Useful to examine probability of each response option or semi-ordered options
    - Can be hard to estimate all parameters (especially separate α or λ per submodel)

Models for Count Outcomes

- Counts: non-negative integer unbounded responses
  - e.g., how many cigarettes did you smoke this week?
  - Traditionally uses natural log link so that predicted outcomes stay ≥ 0

- $\mathbf{g(\bullet)}$ $\mathrm{Log}\big(\mathrm{E}(\mathrm{y_i})\big) = \mathrm{Log}(\mu_\mathrm{i}) = \mathrm{model}$ → predicts mean of $\mathrm{y_i}$
- $\mathbf{g^{-1}(\bullet)}$ $\mathrm{E}(\mathrm{y_i}) = \exp(\mathrm{model})$ → to un-log it, use $\exp(\mathrm{model})$

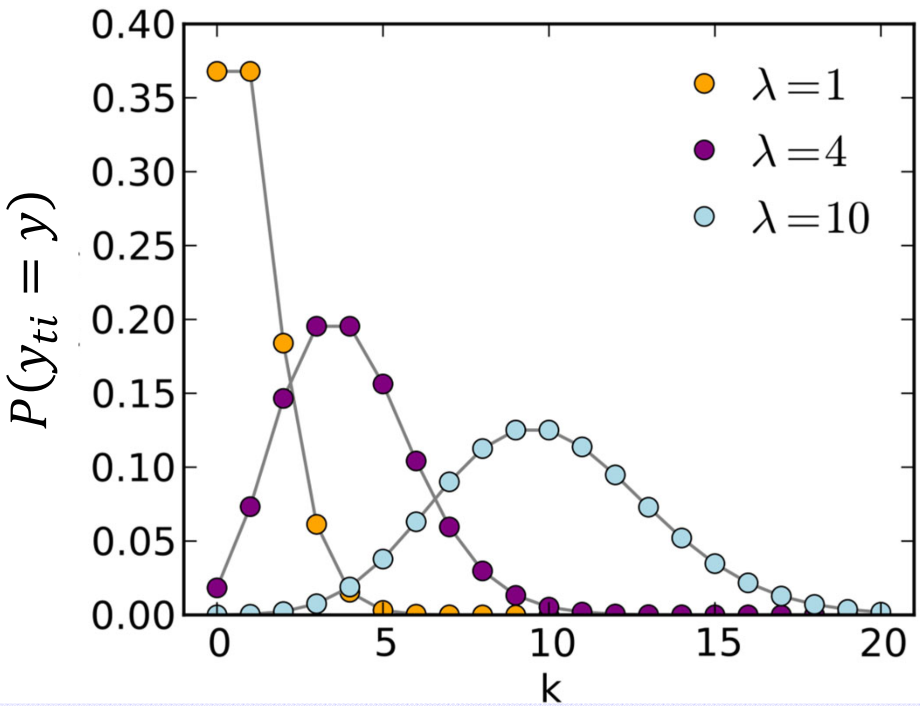
- IFA model (no IRT analog): $\mathbf{Log(y_{is}) = \mu_i + \lambda_i F_s + (e_{is})}$
  - Model has intercepts and loadings, just predicting log(y_{is}) instead
  - In Mplus, identify outcomes as COUNT = in VARIABLE: section

- What about an estimated residual variance? It depends…

Poisson Conditional Distribution

- Poisson distribution has one parameter, λ, which is **both its mean and its variance** (so λ = mean = variance in Poisson)
- $f(\mathrm{y_i}|\lambda) = \mathrm{Prob}(\mathrm{y_i} = \mathrm{y}) = \frac{\lambda^{\mathrm{y}} * \exp(-\lambda)}{\mathrm{y}!}$
- PDF: $\mathrm{Prob}(\mathrm{y_i} = \mathrm{y}|\beta_0, \beta_1, \beta_2) = \frac{\mu_i^{\mathrm{y}} * \exp(-\mu_i)}{\mathrm{y}!}$

y! is factorial of y



The dots indicate that only integer values are observed.

Distributions with a small expected value (mean or λ) are predicted to have a lot of 0's.

Once $\lambda > 6$ or so, the shape of the distribution is close to a that of a normal distribution.

3 potential problems for Poisson…

- The standard Poisson distribution is rarely sufficient, though
- **Problem #1: When mean ≠ variance**
  - If variance < mean, this leads to "under-dispersion" (not that likely)
  - If variance > mean, this leads to "over-dispersion" (happens frequently)
- **Problem #2: When there are *no* 0 values**
  - Some 0 values are expected from count models, but in some contexts $y_i > 0$ always (but subtracting 1 won't fix it; need to adjust the model)
- **Problem #3: When there are *too many* 0 values**
  - Some 0 values are expected from the Poisson and Negative Binomial models already, but many times there are even more 0 values observed than that
  - To fix it, there are two main options, depending on what you do to the 0's
- Each of these problems requires a model adjustment to fix it…

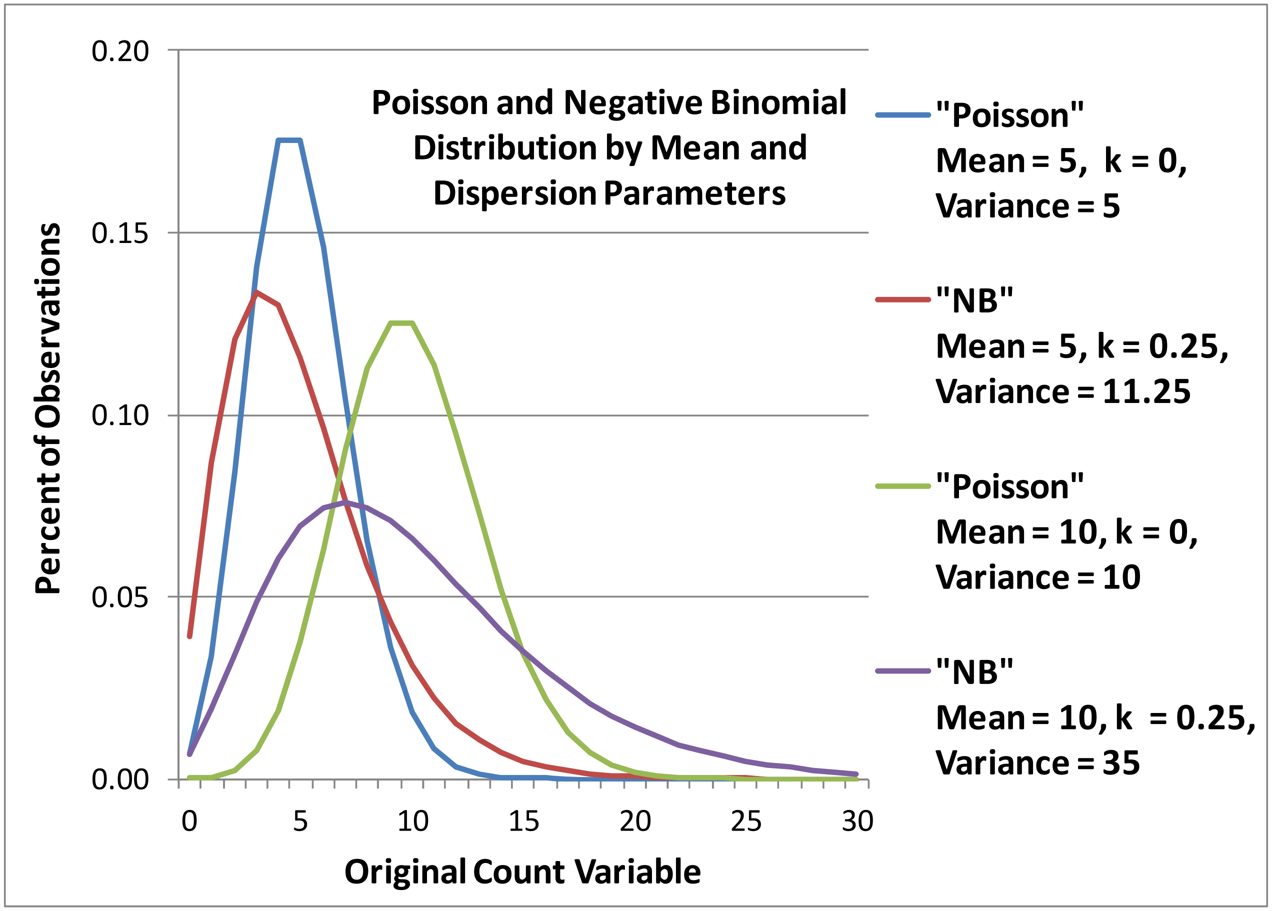
Problem #1: Variance > mean = over-dispersion

- To fix it, we must add a parameter that allows the variance to exceed the mean → a Negative Binomial (NB) distribution
  - Says residuals are a mixture of Poisson and gamma distributions, such that λ itself is a random variable with a gamma distribution
  - So expected mean is still given by λ, but the variance will be > Poisson

- IFA Model: $\mathbf{Log}(\mathbf{y}_{\mathrm{is}}) = \boldsymbol{\mu}_{\mathbf{i}} + \boldsymbol{\lambda}_{\mathrm{i}}\mathbf{F}_{\boldsymbol{s}} + \mathbf{e}_{\mathbf{is}}^{\mathbf{G}}$
  - NB has a k dispersion parameter, such that: $\mathrm{Var}(\mathrm{y}_{\mathrm{is}}) = k(1 + k\alpha)$
  - Poisson is nested within negative binomial (can do −2ΔLL test of α ≠ 0)

- In Mplus, specify which residual distribution you want:
  - COUNT = y1 (p) y2 (nb); → y1 is Poisson; y2 is negative binomial

Negative Binomial (NB) = “Stretchy” Poisson…



$\text{Mean} = \lambda$
$\text{Dispersion} = k$
$\text{Var}(\text{y}_{\text{is}}) = k(1 + k\alpha)$

A Negative Binomial model can be useful for count residuals with extra skewness, but otherwise follow a Poisson distribution.

- Because its k dispersion parameter is fixed to 0, the Poisson model is nested within the Negative Binomial model—to test improvement in fit:
- Is $-2\left(LL_{Poisson} - LL_{NegBin}\right) > 3.84$ for $df = 1$? Then $p < .05$, keep NB

Problem #2: There are no 0 values

- "**Zero-Altered**" or "**Zero-Truncated**" Poisson or Negative Binomial: ZAP/ZANB or ZTP/ZTNB (used in hurdle models)
  - Is usual NB distribution, just not allowing any 0 values
  - In Mplus, COUNT = var1 (nbt); → Negative Binomial Truncated
  - It does NOT work to just subtract 1 and use a usual count distribution

- Poisson PDF was: $\mathrm{Prob}(y_i = y|\mu_i\ ) = \frac{\mu_i^y * \exp(-\mu_i)}{y!}$
- Zero-Truncated Poisson PDF is:
  - $\mathrm{Prob}\big(y_i = y|\mu_{i,}y_i > 0\big) = \frac{\mu_i^y * \exp(-\mu_i)}{y![1-\exp(-\mu_i)]}$
  - $\mathrm{Prob}(y_i = 0) = \exp(-\mu_i)$, so $\mathrm{Prob}(y_i > 0) = 1 - \exp(-\mu_i)$
  - Divides by probability of non-0 outcomes so probability still sums to 1

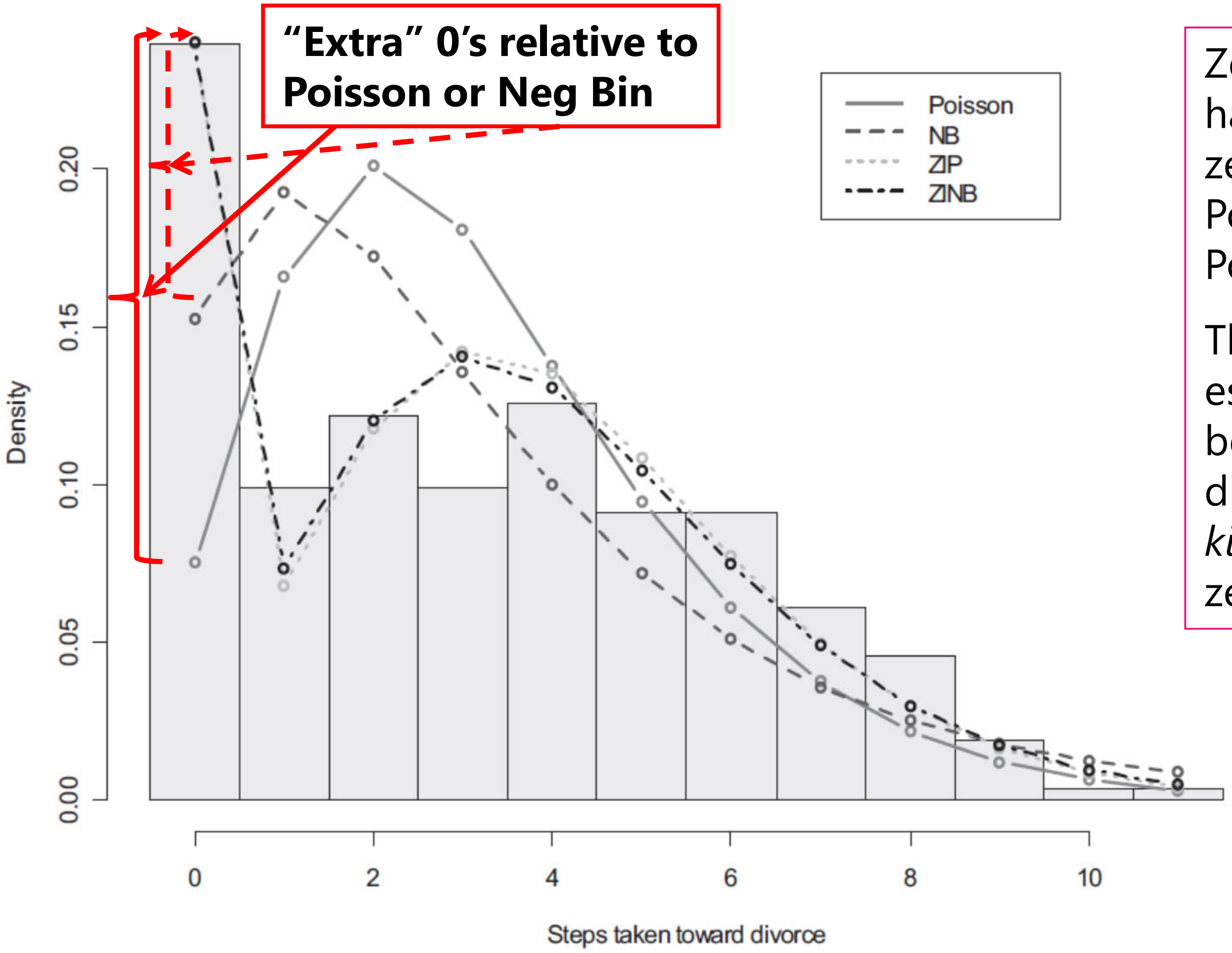
Problem #3: Too many 0 values, Option #1

- "**Zero-Inflated**" Poisson (pi) or Negative Binomial (nbi); available within Mplus using COUNT =
  - Distinguishes **two kinds of 0 values**: **expected** and **inflated** ("structural") through a mixture of distributions (Bernoulli + Poisson/NB)
  - Creates two submodels to predict "if ***extra*** 0" and "if not, how much"?
    - Does not readily map onto most hypotheses (in my opinion)
    - But a ZIP example would look like this… (ZINB would add *k* dispersion, too)

- Submodel 1: $\mathrm{Logit}(y_{\mathrm{is}} = \mathrm{extra}\ 0) = -\tau_{\mathrm{i}1} + \lambda_{\mathrm{i}1}\mathrm{F}_{s1}$
  - Predict being an <u>extra 0</u> using Link = Logit, Distribution = Bernoulli
  - Don't have to specify a factor model for this part, can simply allow a threshold that says your data have extra 0 values relative to the usual count distribution

- Submodel 2: $\mathrm{Log}(y_{\mathrm{is}}) = \mu_{\mathrm{i}2} + \lambda_{\mathrm{i}2}\mathrm{F}_{s2}$
  - Predict rest of counts (including 0's expected from count distribution)

Example of Zero-Inflated Outcomes



"Extra" 0's relative to Poisson or Neg Bin

Figure 1. Histogram of Marital Status Inventory with predicted probabilities from regressions. NB = negative binomial; ZIP = zero-inflated Poisson; ZINB = zero-inflated negative binomial.

Zero-inflated distributions have extra "structural zeros" not expected from Poisson or NB ("stretched Poisson") distributions.

This can be tricky to estimate and interpret because the model distinguishes between *kinds of zeros* rather than zero or not...

Image borrowed from Atkins & Gallop, 2007

Problem #3: Too many 0 values, Option #2

- "**Hurdle**" models for Poisson or Negative Binomial
  - PH (not in Mplus) or NBH: Explicitly **separates 0 from non-0 values** through a mixture of distributions (Bernoulli + Zero-Altered Poisson/NB)
  - Creates two submodels to predict "if any 0" and "if not 0, how much"?
    - Easier to think about in terms of prediction (in my opinion)

- Submodel 1: $\mathrm{Logit}(y_{is} = 0) = -\tau_{i1} + \lambda_{i1}F_{s1}$
  - Predict being <u>any 0</u> using Link = Logit, Distribution = Bernoulli
  - Don't have to specify predictors for this part, can simply allow it to exist

- Submodel 2: $\mathrm{Log}(y_{is}|y_{is} > 0) = \mu_{i2} + \lambda_{i2}F_{s2}$
  - Predict rest of positive counts via Link = Log, Distribution = ZAP/ZANB

- These models are for discrete outcomes, but there is an analogous version for continuous outcomes

Two-Part Models for Continuous Outcomes

- A two-part model is an analog to hurdle models for zero-inflated count outcomes (and could be used with count outcomes, too)
  - Explicitly **separates 0 from non-0 values** through a mixture of distributions (Bernoulli + Normal or LogNormal) → usually much easier to explain!

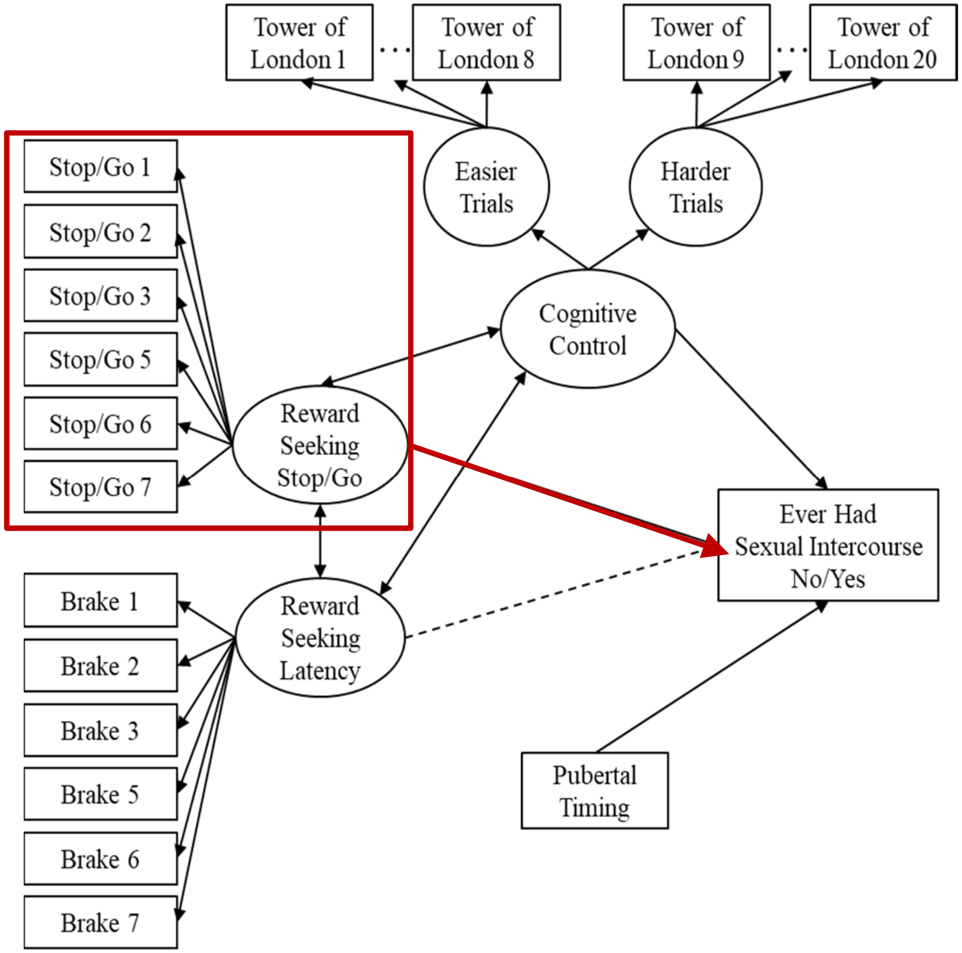
- Submodel 1: $\mathrm{Logit}(y_i > 0) = -\tau_{i1} + \lambda_{i1}F_{s1}$ → predict being NOT 0

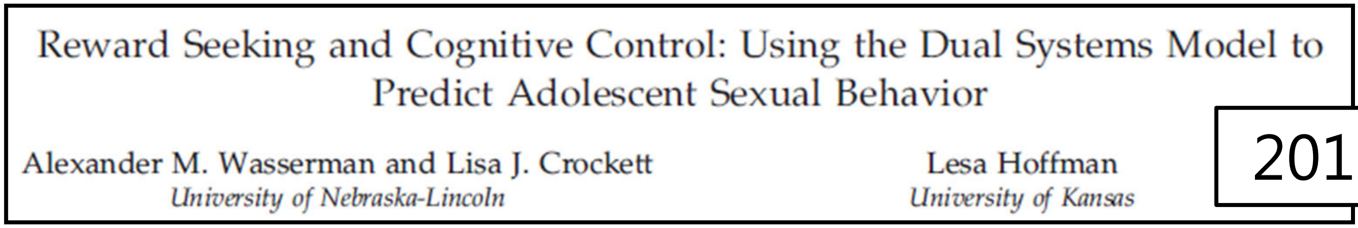
- Submodel 2: $(y_i | y_i > 0) = \mu_{i2} + \lambda_{i2}F_{s2}$ → predicts non-0 using normal or lognormal residuals

  - Two-part model uses Mplus DATA TWOPART: command
    - NAMES ARE y1-y4; → list outcomes to be split into 2 parts
    - CUTPOINT IS 0; → where to split observed outcomes
    - BINARY ARE b1-b4; → create names for “0 or not” part
    - CONTINUOUS ARE c1-c4; → create names for “how much” part
    - TRANSFORM IS LOG; → transformation of continuous part
    - 0 or not: predicted by logit of being NOT 0 (“something” is the 1)
    - How much: predicted by transformed normal distribution (like log)

Two-Part Measurement Model Example

- Two-part model for a sensation-seeking task: 7 driving trials in which light turns yellow…
- Two outcomes:
  - Go through light?
  - Latency to break if not going through?
- Created two latent factors, each of which then predicted other outcomes




Reward Seeking and Cognitive Control: Using the Dual Systems Model to Predict Adolescent Sexual Behavior

Alexander M. Wasserman and Lisa J. Crockett
University of Nebraska-Lincoln

Lesa Hoffman
University of Kansas

2017

Pile of 0's Taxonomy

- What kind of **amount** do you want to predict?
  - Discrete: Count → Poisson, Stretchy Count → Negative Binomial
  - Continuous: Normal, Log-Normal, Gamma (not in Mplus)

- What kind of **If 0** do you want to predict?
  - Discrete: Extra 0 beyond predicted by amount?
→ Zero-inflated Poisson or Zero-inflated Negative Binomial
  - Discrete: Any 0 at all?
→ Hurdle Poisson or Hurdle Negative Binomial
  - Continuous: Any 0 at all?
→ Two-Part with Continuous Amount (see above)
  - Note: Given the same amount distribution, these alternative ways of predicting 0 will result in the same empty model fit

Wrapping Up…

- When fitting latent factor models (or when just predicting observed outcomes from observed predictors instead), you have many options to fit non-normal distributions
  - **CFA:** Continuous outcomes with normal residuals, X → Y is linear
    - If residuals may not be normal *but a linear X→ Y relationship is still plausible*, you can use MLR estimation instead of ML to control for that
  - **IRT and IFA:** Categorical or ordinal outcomes with Bernoulli/multinomial residuals, X → transformed Y is linear; X → original Y is nonlinear
    - Full information MML traditionally paired with IRT version of model; limited information WLSMV traditionally paired with IFA version of model instead
  - **Count family:** Non-negative integer outcomes, X → Log(Y) is linear
    - Residuals can be Poisson (where mean = variance) or negative binomial (where variance > mean); either can be zero-inflated or zero-truncated
    - Hurdle or two-part may be more direct way to predict/interpret excess zeros (predict zero or not and how much rather than two kinds of zeros)