

Latent Trait Measurement Models for Binary Responses: IRT and IFA

- Topics:
 - **The Big Picture of Latent Trait Measurement Models**
 - 1, 2, 3, and 4 Parameter IRT (and Rasch) Models
 - Item and Test Information (for Indexing Reliability)
 - From Item Response Theory Models to Item Factor Models
 - Model Estimation, Comparison, and Evaluation

The Big Picture of CTT

- **CTT** predicts the total: $Y_{total_s} = TrueScore_s + error_s$
 - Items are assumed exchangeable, and their properties are not part of the model for creating a latent trait estimate (as total)
 - **Because the latent trait estimate IS the sum score**, it can be problematic to make comparisons across different test forms
 - Item difficulty = mean of item (is sample-dependent)
 - Item discrimination = item-remainder correlation (is sample-dependent)
 - Estimates of reliability assume (without testing) unidimensionality and tau-equivalence (alpha) or parallel items (Spearman-Brown)
 - Measurement error is (usually) assumed *constant* across the trait level
- How do you make your test better?
 - Get more items. What kind of items? More.

The Big Picture of CFA

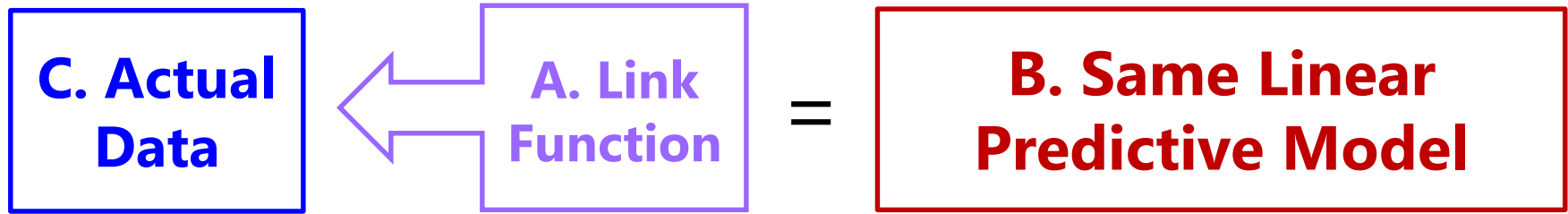
- **CFA predicts the ITEM response:** $y_{is} = \mu_i + \lambda_i F_s + e_{is}$
 - **Linear regression relating continuous item response to latent predictor F_s**
 - Both items AND subjects matter in predicting responses
 - Item difficulty = intercept μ_i (in theory, sample independent)
 - Item discrimination = factor loading λ_i (in theory, sample independent)
 - The goal of the factor is to predict the observed covariances among items, so **factors represent testable assumptions** about the pattern of item covariance
 - Items should be unrelated after controlling for factors → local independence
 - If not, error covariances can be added to hold unexpected multidimensionality
- **Because individual item responses are included:**
 - Items can vary in discrimination (→ Omega reliability) and difficulty
 - To make your test better, you need more BETTER items...
 - With higher standardized factor loadings → with greater information = $\lambda_i^2 / \text{Var}(e_i)$
- Measurement error is still assumed constant across the latent trait (one value)

From CFA to IRT and IFA...

Outcome Type → Model Family Name	Observed Predictor X	Latent Predictor X
Continuous outcomes → <i>"General Linear Model"</i>	"Linear" Regression	Confirmatory Factor Models
Discrete/categorical outcomes → <i>"Generalized Linear Model"</i>	Logistic/Probit/ Multinomial Regression	Item Response Theory and Item Factor Analysis

- Basis of Item Response Theory (IRT) and Item Factor Analysis (IFA) lies in models for discrete outcomes, which are called "**generalized**" linear models
- Thus, IRT and IFA will be easier to understand after reviewing concepts from generalized linear models...

3 Parts of Generalized Linear Models



- A. **Link Function**: How the bounded conditional mean to be predicted (e.g., probability) is transformed to become unbounded so that the model can predict it linearly
- B. **Same Linear Model**: how fixed effects of predictors (factors) combine additively to predict the link-transformed (unbounded) conditional mean outcome
- C. **Conditional Distribution**: how the outcome(s) should be distributed after controlling for the factor(s) given the outcome's sample space (possible values)

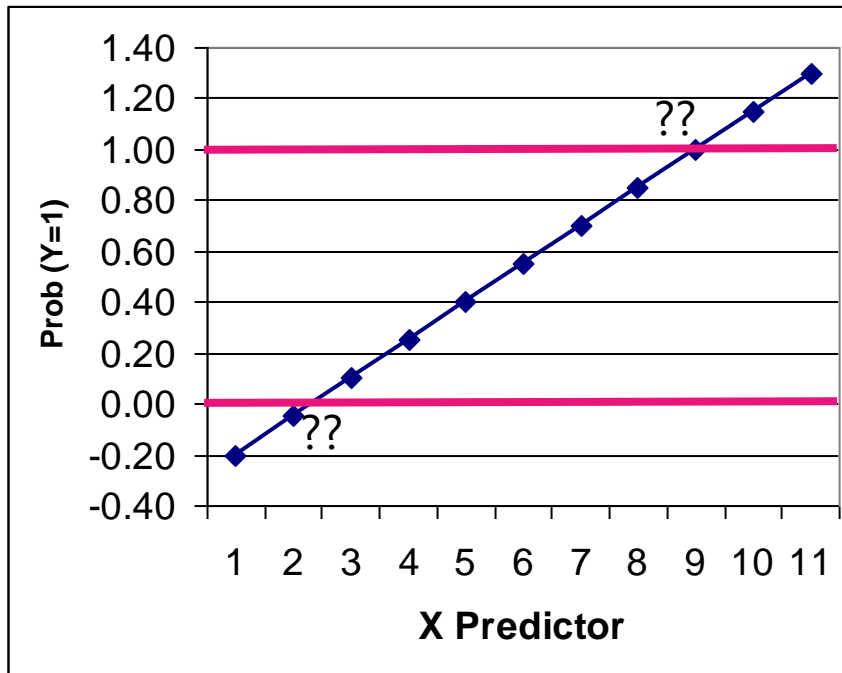
Here's how it works for binary outcomes

- Let's say we have a single binary (0 or 1) outcome... y_i ($i=person$)
- The mean of a binary outcome is the proportion of 1 values
 - So given each person's predictor values, the model tries to predict the **conditional mean**: the **probability of having a 1**: $p(y_i = 1)$
 - The conditional mean has more possible values than the outcome
 - General linear model: $p(y_i = 1) = \beta_0 + \beta_1(x1_i) + \beta_2(x2_i) + e_i$
 - β_0 = expected probability of $y_i = 1$ when all predictors = 0
 - β 's = expected change in $p(y_i = 1)$ for a one-unit Δ in predictor
 - e_i = difference between observed and predicted binary values
 - Model becomes $y_i = (\text{predicted probability of 1}) + e_i$
 - **What could possibly go wrong?**

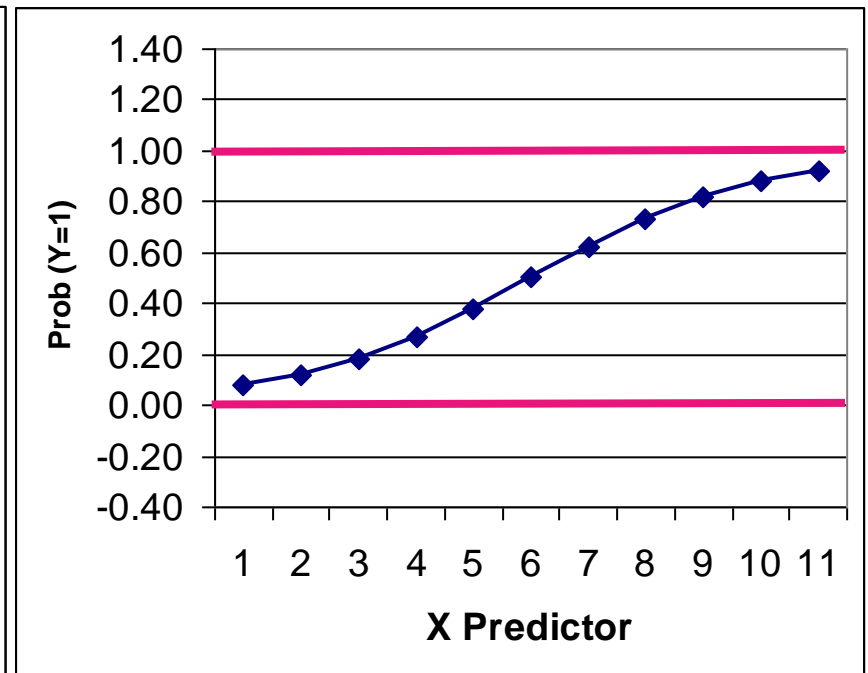
Normal GLM for Binary Outcomes?

- Problem #1: A **linear** relationship between X and Y???
- Probability of a 1 is bounded between 0 and 1, but predicted probabilities from a linear model aren't going to be bounded
- Linear relationship needs to shut off at ends → be nonlinear

We have this...

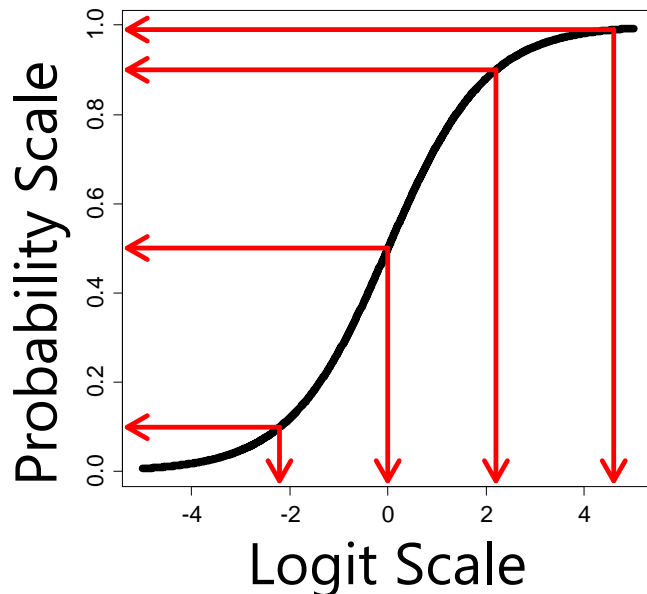


But we need this...



Generalized Models for Binary Outcomes

- Solution to #1: Rather than predicting $p(y_i = 1)$ directly, we must transform it into an unbounded variable with a **link function**:
 - Transform **probability** into **odds**: $\frac{p_i}{1-p_i} = \frac{\text{prob}(y_i=1)}{\text{prob}(y_i=0)}$
 - If $p(y_i = 1) = .7$ then Odds(1) = 2.33; Odds(0) = 0.429
 - But odds scale is skewed, asymmetric, and ranges from 0 to $+\infty$ → Not helpful
 - Take **natural log of odds** → called “**logit**” link: $\text{Log} \left[\frac{p_i}{1-p_i} \right]$
 - If $p(y_i = 1) = .7$, then Logit(1) = 0.846; Logit(0) = -0.846
 - Logit scale is now symmetric about 0, range is $\pm\infty$ → DING

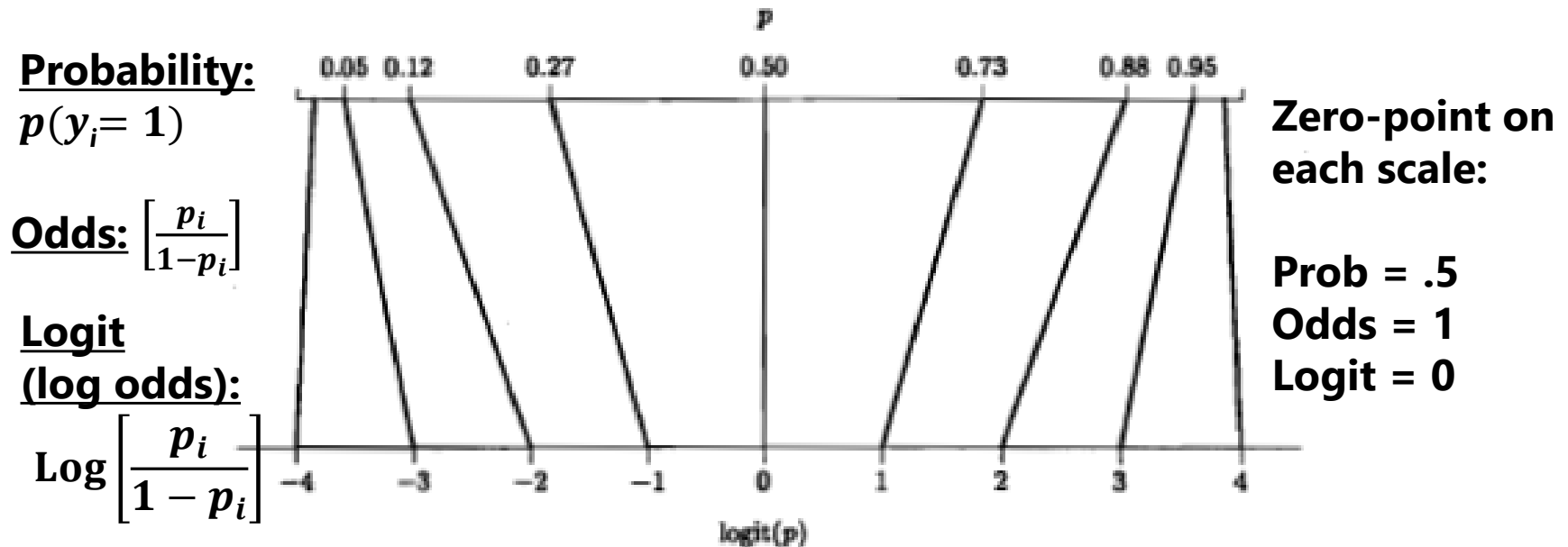


Probability	Logit
0.99	4.6
0.90	2.2
0.50	0.0
0.10	-2.2

Can you guess what $p(.01)$ would be on the logit scale?

Solution to #1: Probability into Logits

- **A Logit link is a nonlinear transformation of probability:**
 - Equal intervals in logits are NOT equal intervals of probability
 - Logits range from $\pm\infty$ and are symmetric about prob = .5 (logit = 0)
 - Now we can use a linear model \rightarrow The model will be **linear with respect to the predicted logit**, which translates into a nonlinear prediction with respect to probability \rightarrow **the outcome conditional mean shuts off at 0 or 1 as needed**



Normal GLM for Binary Outcomes?

- General linear model: $p(y_i = 1) = \beta_0 + \beta_1(x1_i) + \beta_2(x2_i) + e_i$
- If y_i is binary, then e_i can only be 2 things: $e_i = y_i - \hat{y}_i$
 - If $y_i = 0$ then $e_i = (0 - \text{predicted probability})$
 - If $y_i = 1$ then $e_i = (1 - \text{predicted probability})$
- Problem #2a: So the residuals can't be normally distributed
- Problem #2b: The residual variance can't be constant over \hat{y}_i as in GLM because the **mean and variance are dependent**
 - Variance of binary variable: $Var(y_i) = p_i (1 - p_i)$

Mean and Variance of a Binary Variable

Mean (p_i)	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	1.0
Variance	.0	.09	.16	.21	.24	.25	.24	.21	.16	.09	.0

Solution to #2: Bernoulli Distribution

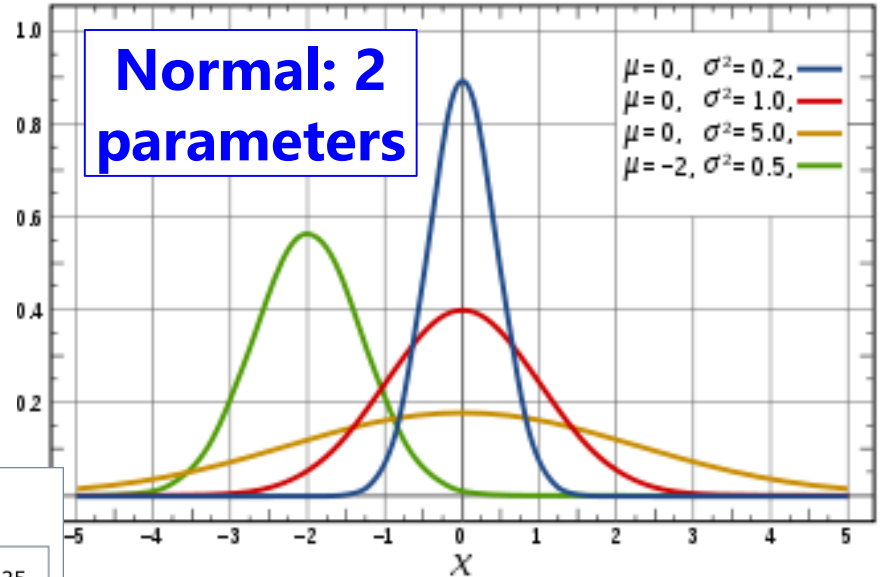
- Rather than using a **normal conditional distribution** for the outcome, we will use a **Bernoulli conditional distribution**

Univariate Normal PDF:

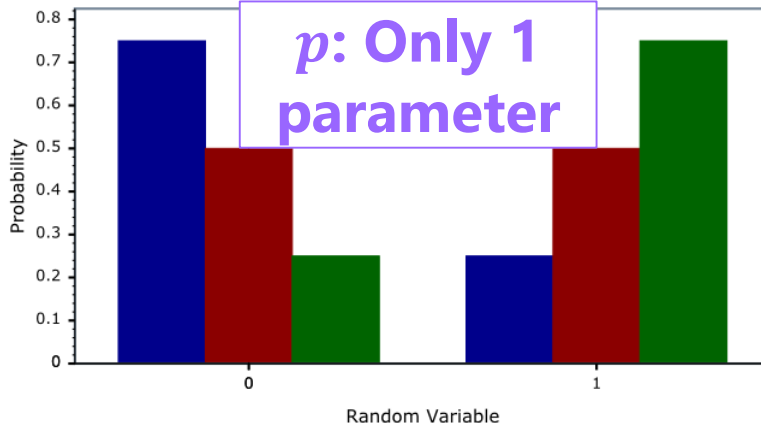
$$f(y_i) = \frac{1}{\sqrt{2\pi\sigma_e^2}} * \exp\left[-\frac{1}{2} * \frac{(y_i - \mu)^2}{\sigma_e^2}\right]$$

Likelihood (y_i)

Normal: 2 parameters



Bernoulli Distribution PDF



Bernoulli PDF:

$$f(y_i) = (p_i)^{y_i} (1 - p_i)^{1-y_i}$$

= $p(1)$ if $y_i=1$,
= $p(0)$ if $y_i=0$

Predicted Binary Outcomes

- **Logit:** $\text{Log} \left[\frac{p(y_i=1)}{1-p(y_i=1)} \right] = \beta_0 + \beta_1(x1_i) + \beta_2(x2_i)$ ← **g(·) link**

- Predictor effects are linear and additive like in GLM, but β = change in **logit** per one-unit change in predictor

- **Odds:** $\left[\frac{p(y_i=1)}{1-p(y_i=1)} \right] = \exp(\beta_0 + \beta_1 x1_i + \beta_2 x2_i)$

- **Probability:** $p(y_i = 1) = \frac{\exp(\beta_0 + \beta_1 x1_i + \beta_2 x2_i)}{1 + \exp(\beta_0 + \beta_1 x1_i + \beta_2 x2_i)}$ ← **g⁻¹(·) inverse link**

or equivalently $p(y_i = 1) = \frac{1}{1 + \exp[-1(\beta_0 + \beta_1 x1_i + \beta_2 x2_i)]}$

- Foreshadowing: IRT models are usually described using the probability version, whereas IFA models use the logit version

Converting Across the 3 Scales

- e.g., for $\text{Log} \left[\frac{p(y_i=1)}{1-p(y_i=1)} \right] = \hat{y}_i = \beta_0 + \beta_1(x1_i) + \beta_2(x2_i)$

Direction	Conditional Mean	Slope for $x1_i$	Slope for $x2_i$
Using logits to predict probability (i.e., the "link"):	\hat{y}_i	β_1	β_2
From logits to odds or odds ratios for effect sizes:	Odds: $\exp(\hat{y}_i)$	Odds ratio: $\exp(\beta_1)$	Odds ratio: $\exp(\beta_2)$
From logits to probability (i.e., the "inverse link"):	$\frac{\exp(\hat{y}_i)}{1 + \exp(\hat{y}_i)}$	Doesn't make any sense!	Doesn't make any sense!

- You can unlogit the model-predicted conditional mean all the way back into probability to express predicted outcomes, but you can only unlogit the slopes back into odds ratios (not all the way back to probability)

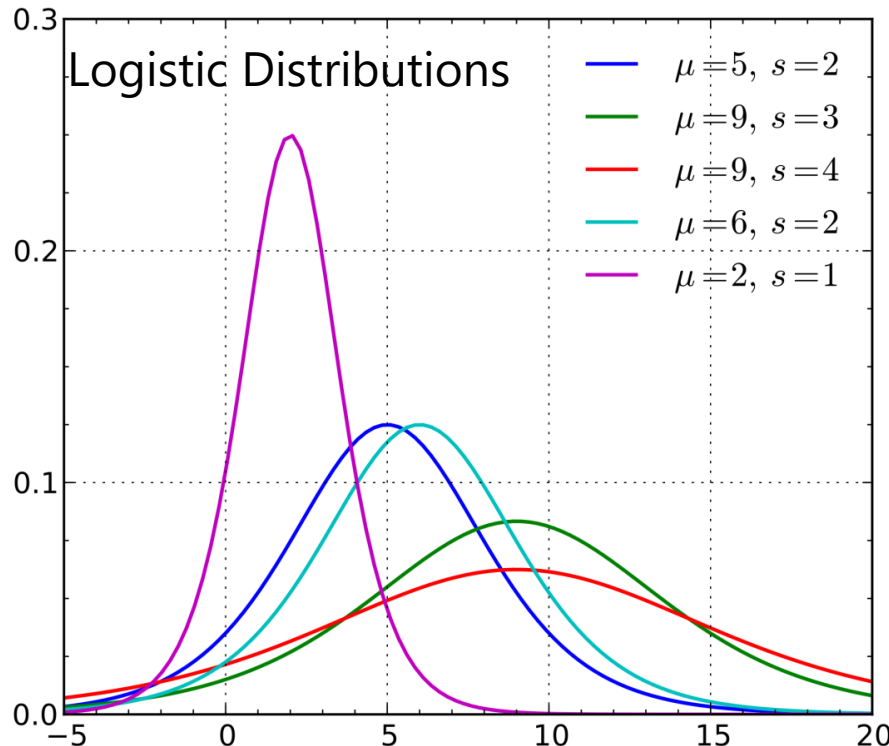
“Latent Responses” for Binary Data

- This model is sometimes expressed by calling the $\text{logit}(y_i)$ an underlying continuous (“latent”) response of y_i^* instead:

Empty Model: $y_i^* = -\textit{threshold} + e_i$

$\textit{threshold} = \beta_0 * -1$ is given in Mplus, not the intercept

- In which $y_i = 1$ if $(y_i^* > \textit{threshold})$, or $y_i = 0$ if $(y_i^* \leq \textit{threshold})$



So **when predicting y_i^*** , then $e_i \sim \text{Logistic}(0, \sigma_e^2 = \mathbf{3.29})$

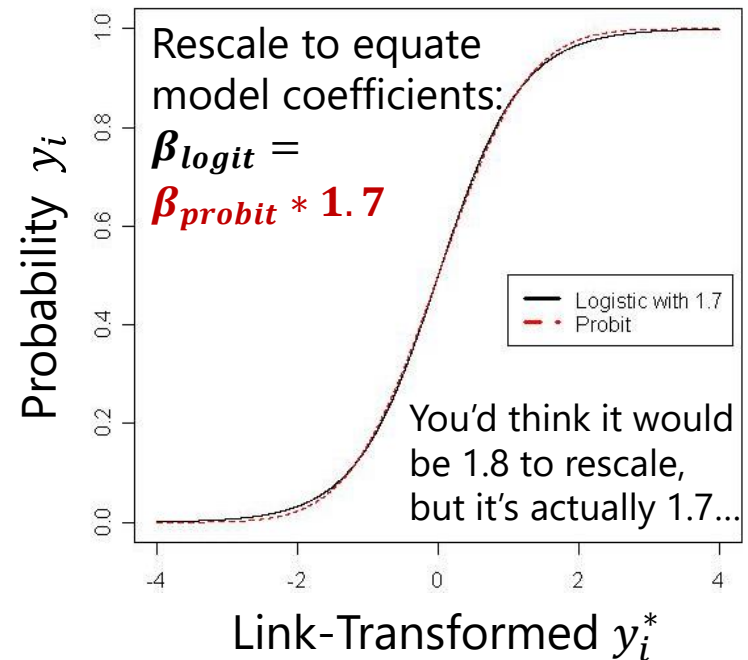
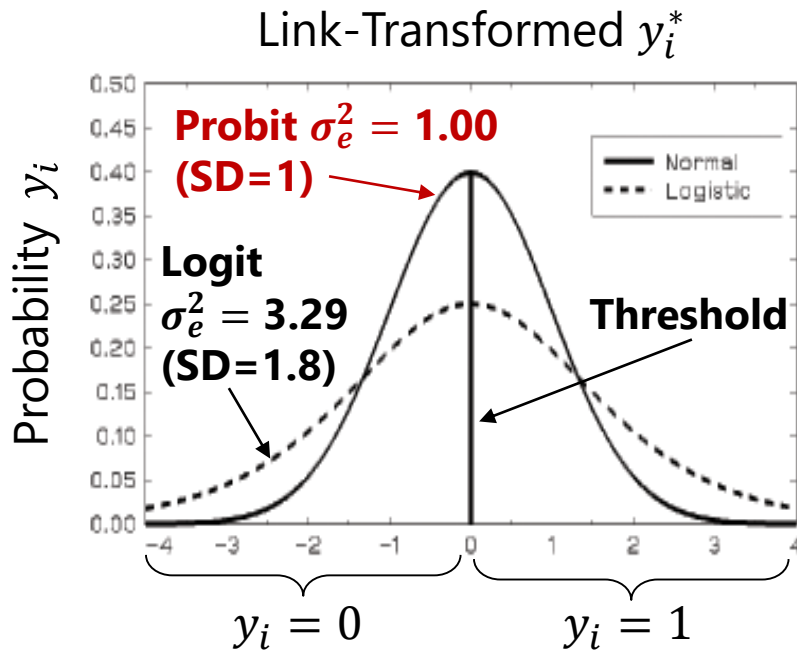
Logistic Distribution:

Mean = μ , Variance = $\frac{\pi^2}{3} s^2$,
where s = scale factor that allows for “over-dispersion” (must be fixed to 1 for binary responses for identification)

Other Link Functions for Binary Data

- The idea that a “latent” continuous variable underlies an observed binary response also appears in a **Probit Regression** model:
 - A **probit** link, such that now your model predicts a different transformed y_i :
$$\text{Probit}(y_i = 1) = \Phi^{-1}[p(y_i = 1)] = \text{linear predictor} \leftarrow \boxed{\text{g}(\cdot) \text{ link}}$$
 - Where Φ = standard normal cumulative distribution function, so the transformed y_i is the **z-score** that corresponds to the value of standard normal curve **below** which conditional mean probability is found (requires integration to inverse link)
 - Same Bernoulli distribution for the conditional binary outcomes, in which residual variance cannot be separately estimated (so no e_i in the model)
 - Model scale: Probit can also predict “latent” response: $y_i^* = -\text{threshold} + e_i$
 - But Probit says $e_i \sim \text{Normal}(\mathbf{0}, \sigma_e^2 = \mathbf{1.00})$, whereas logit $\sigma_e^2 = \frac{\pi^2}{3} = \mathbf{3.29}$
 - So given this difference in variance, probit coefficients are on a different scale than logit coefficients, and so their estimates won’t match... however...

Probit vs. Logit: Should you care? Pry not.



- Other fun facts about probit:
 - Probit = “ogive” in the Item Response Theory (IRT) world
 - Probit has no odds ratios (because it’s not based on odds)
 - Will be the only option in IFA models using limited-information estimation
- Both logit and probit assume **symmetry** of probability curve, but there are other *asymmetric* options as well: (comp) log-log

How IRT/IFA are the same as CFA

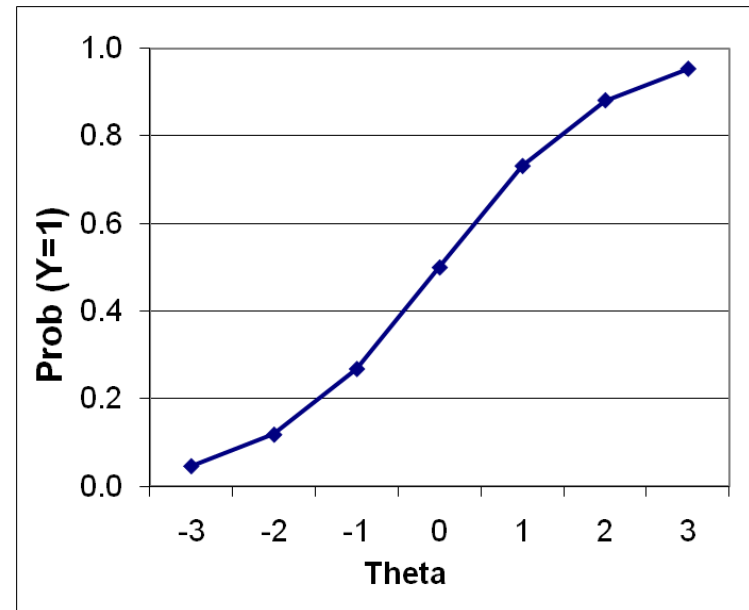
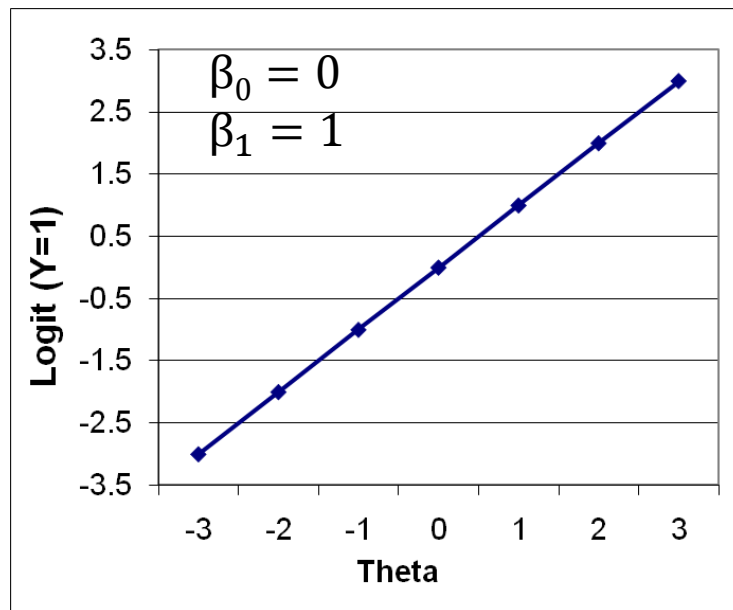
- **NOW BACK TO YOUR REGULARLY SCHEDULED MEASUREMENT CLASS**
- **IRT/IFA** = confirmatory measurement model in which latent traits are the model predictors (so you decide which items measure which traits)
 - Like CFA, **both items and persons matter** because their properties are included in the measurement model (item difficulty, item discrimination; person F)
 - Item discrimination is the same in IRT and IFA, but these models differ in how location of the item on the trait is indexed (item difficulties versus thresholds)
- After controlling for a person's latent trait score (F is now called **Theta**), the item responses should be uncorrelated (also called local independence)
 - The **ONLY** reason item responses are correlated is a unidimensional Theta
 - Otherwise, we can fit confirmatory multidimensional factor models instead, and then responses are independent after controlling for ALL Thetas
 - As in CFA, can be violated by other types unaccounted for multidimensionality or dependency (e.g., method factors, common stem "testlets")
 - Error covariances must be specified using method factors when using ML estimation

How IRT/IFA are *different from* CFA

- IRT/IFA uses the same family of **link functions** (transformations) as in generalized models, it's just that the predictor isn't measured directly
 - IRT/IFA = logistic/probit regression instead of linear regression
 - Predictor = Latent factor/trait in IRT/IFA = "Theta" and its slopes are still supposed to predict the covariance across item responses, just like in CFA
- **IRT/IFA specifies a nonlinear relationship between binary, ordinal, or nominal item responses and the latent trait (now called "Theta")**
 - Probability is bounded between 0 and 1, so the effect (slope) of Theta must be nonlinear, so it will shut off at the extremes of Theta (as an S-shaped curve)
 - Errors cannot have constant variance across Theta or be normally distributed
 - Full-information estimators use logit ($\sigma_{e^*}^2 = 3.29$) or probit ($\sigma_{e^*}^2 = 1.00$) link functions, but limited-information estimators only have probit ($\sigma_{e^*}^2 = 1.00$)
 - Logit = 1.7*Probit, so it's pretty much the same result either way
 - Probit in IRT models is called "ogive" (as discussed in Embretson & Reise)

Nonlinearity in IRT/IFA

- The relationship between Theta and the probability of response=1 is “**nonlinear**” → a **monotonic s-shaped logistic curve** whose shape and location are dictated by the estimated item parameters
 - **Linear** prediction of the **logit**, **nonlinear** prediction of **probability**



- It may be that other kinds of non-linear relationships could be more appropriate and thus fit better → These are “non-parametric” IRT models

Item Response Theory (IRT) “vs” Item Factor Analysis (IFA) Models

Mplus can do ALL of these model/estimator combinations:	Model form with discrimination and difficulty parameters	Model form with loadings and thresholds
Full-information estimation via Maximum Likelihood (“Marginal ML”) → uses <i>original item responses</i>	“IRT” (Mplus gives only for binary responses)	“?” (Mplus gives for all models)
Limited-information estimation via Weighted Least Squares (“WLSMV”) → uses <i>item response summary</i>	“?” (Mplus gives only for binary responses)	“IFA” (Mplus gives for all models)

- CFA assumes normally distributed, continuous item responses, but “**CFA** models for **categorical** responses” are actually **IRT** and **IFA** models
- These different names are used to reflect the combination of how the model is specified and how it is estimated, but it’s the same core model

Model Format in IRT and IFA

- Item Factor Analysis (IFA) models look very similar to CFA, but Item Response Theory (IRT) models look very different
- Partly due to predicting logits/probits (IFA) vs. probability (IRT):
 - **Logit:** $\text{Log} \left[\frac{p(y_i=1)}{1-p(y_i=1)} \right] = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i}$
 - **Probability:** $p(y_i = 1) = \frac{\exp(\beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i})}{1 + \exp(\beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i})}$
- Partly due to different model formats (stay tuned)
- These two model forms are just re-arrangements of each other, but historically have been estimated using different methods (full vs. limited information) and for different purposes
- Mplus provides both kinds of output for binary data, but only IFA output for categorical data (we will calculate IRT version)
- We'll start with IRT for binary responses, then move to IFA ...

Latent Trait Measurement Models for Binary Responses: IRT and IFA

- Topics:
 - The Big Picture of Latent Trait Measurement Models
 - **1, 2, 3, and 4 Parameter IRT (and Rasch) Models**
 - Item and Test Information (for Indexing Reliability)
 - From Item Response Theory Models to Item Factor Models
 - Model Estimation, Comparison, and Evaluation

Simplest IRT Model: One-Parameter Logistic (1-PL or Rasch) Model for Binary Responses (0/1)

- 1PL model is written in different, but equivalent ways (Embretson & Reise):

- **Logit:** $\text{Log} \left[\frac{p(y_{is}=1)}{1-p(y_{is}=1)} \right] = \theta_s - b_i$

y_{is} is 0 or 1 response to item i for subject s

- **Probability:** $p(y_{is} = 1) = \frac{\exp(\theta_s - b_i)}{1 + \exp(\theta_s - b_i)}$

- $\theta_s = \mathbf{subject\ trait} \rightarrow$ most likely latent trait score (**Theta**, a random effect) for subject s given the pattern of item responses
 - $b_i = \mathbf{"item\ difficulty"}$ \rightarrow **location** on latent trait (estimated as fixed effect) (like an intercept, but it's actually "difficulty" now!)
- Probability of $y_{is} = 1$ depends on person trait (theta) vs. item difficulty:
 - If trait > difficulty, then logit > 0, probability > .50
 - If difficulty > trait, then logit < 0, probability < .50

Fundamentals of IRT

- **Back in CTT**, scores only have meaning relative to the persons in the same sample, and thus **sample norms** are needed to interpret a person's score
 - "I got a 12. Is that good?"
"Well, that puts you into the 90th percentile."
"Great!"
 - "I got a 12. Is that good?"
"Well, that puts you into the 10th percentile."
"Doh!"
 - Same score in both cases, but different reference group!
- **In IRT**, the properties of items and persons are placed along the same underlying latent continuum= "**conjoint scaling**"
 - This concept can be illustrated using **construct maps** that order both persons in terms of ability and items in terms of difficulty...

A Real-World Construct Map Example

A Latent Continuum of 80s Pop Culture Knowledge

Person Side

Item Side

Me

Home of Alf

·
My Brother

·
WHAM

·
Undergraduates

·
Duckie

·
Average Adult

·
Cosby Kids

·
My Mom

·
Mickey

Theta θ_s = Item difficulty level at which one has a 50% probability of $y_{is} = 1$

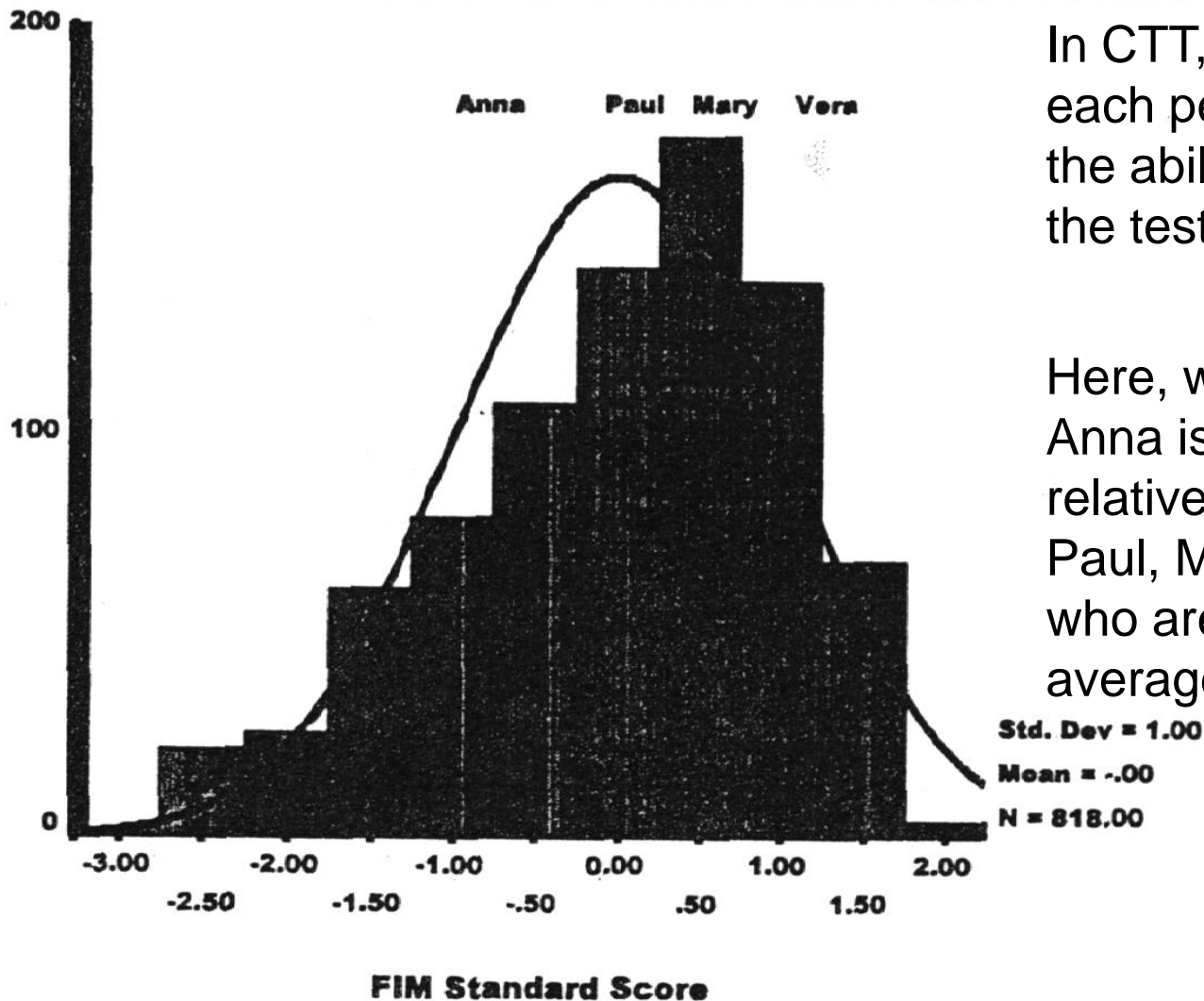
Trait Theta θ_s is interpreted relative to item locations, not group norms

Persons are ordered by Theta ability/severity

Items are located in order by difficulty/severity

Person Theta and Item Difficulty share the same latent metric

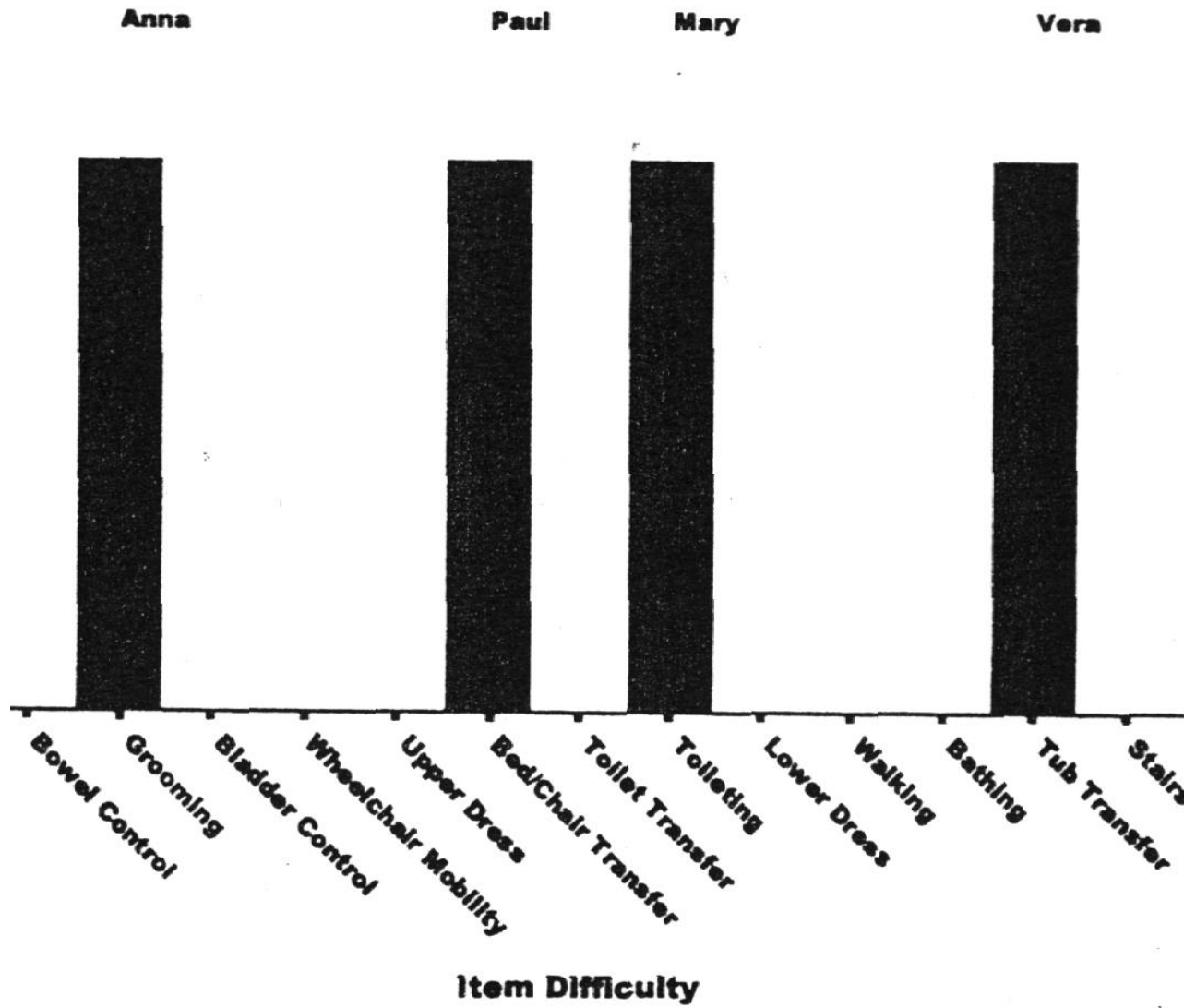
Norm-Referenced Measurement in CTT



In CTT, the ability level of each person is relative to the abilities of the rest of the test sample.

Here, we would say that Anna is functioning relatively worse than Paul, Mary, and Vera, who are each above average (which is 0).

Item-Referenced Measurement in IRT



Each person's Theta score reflects the level of activity they can do on their own **50% of the time.**

The model predicts the probability of accomplishing each task given Theta.

Interpretation of Theta

- **Theta estimates are 'sample-free' and 'scale-free'**
 - Theta estimate does not depend on who took the test with you
 - Theta estimate does not depend on which items were on the test
 - After calibrating all items to the same metric, can get a person's location on latent ability metric regardless of which *particular* items were given
- However: although the Theta estimate does not depend on the particular items, its ***standard error*** does
 - Extreme Thetas without many items of comparable difficulty will not be estimated that well → large SE (flat likelihood)
 - Likewise, items of extreme difficulty without many persons of comparable ability will not be estimated that well → large SE

Another version of the 1PL (Rasch) Model

➤ **Logit:** $\text{Log} \left[\frac{p(y_{is}=1)}{1-p(y_{is}=1)} \right] = a(\theta_s - b_i)$

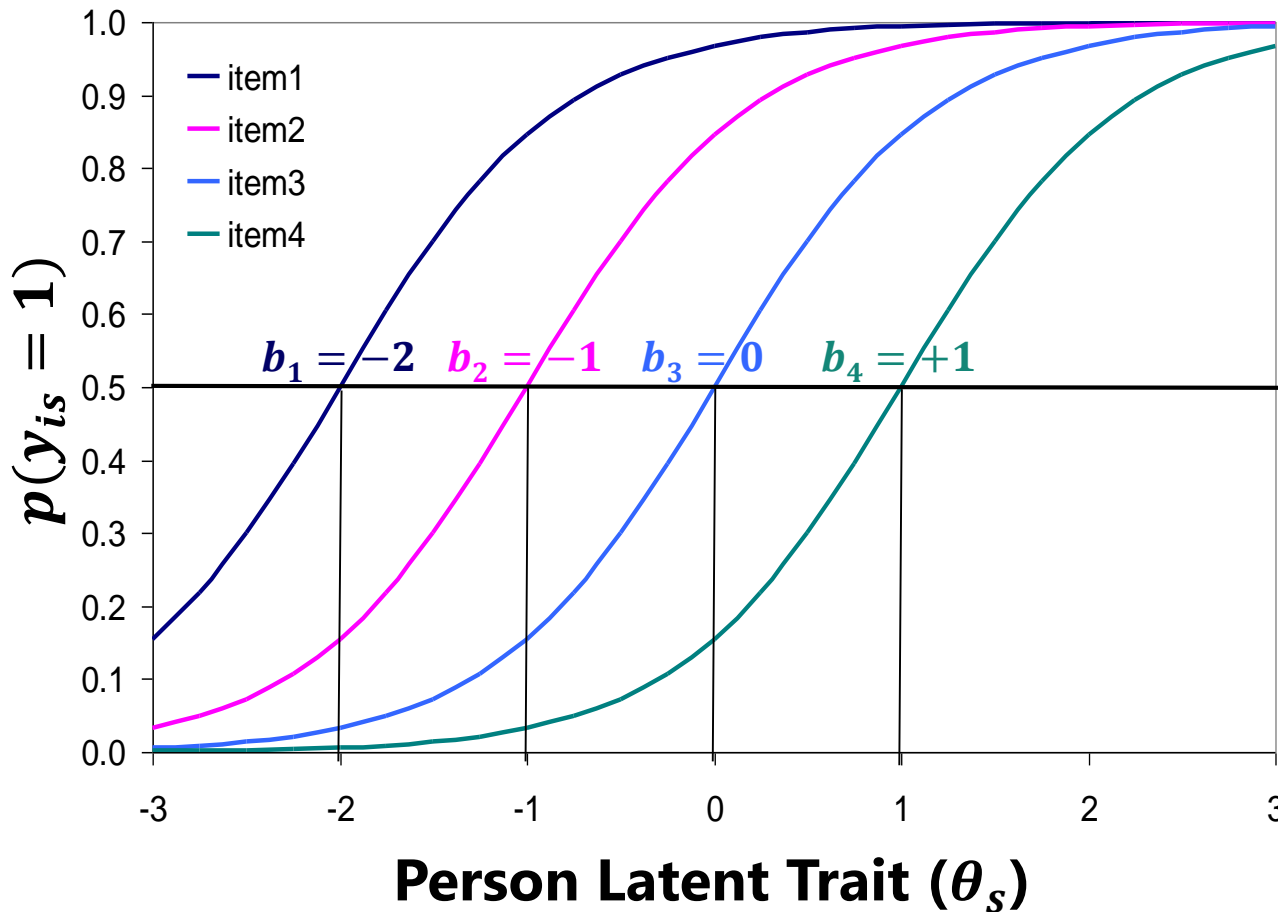
➤ **Probability:** $p(y_{is} = 1) = \frac{\exp[a(\theta_s - b_i)]}{1 + \exp[a(\theta_s - b_i)]}$

Relative to those in a logit-link model, parameters from a **probit-link** (ogive) model will be smaller by a factor of 1.7

- **a = "item discrimination"** = relation of item to latent trait = slope of the s-shape curve as it crosses probability = .50 (its max slope)
- The 1-PL model has " **a** " and not " **a_i** " – that's because **a** is assumed constant across items (and thus, the 1 parameter that is estimated for each item is still difficulty **b_i**)
- If using the probit link function, the predicted outcome is the z-score for the area to the left under the normal curve for that predicted probability
- Previously Mplus factored out **1.7** next to the **a** so that the model parameters would be comparable regardless of using a probit or logit link, but the 1.7 is now embedded in the parameters instead

1-PL (Rasch) Model Predictions

Item Characteristic Curves - 1-PL (Rasch) Model



b_i = **item difficulty**
location on latent trait at which probability = .50

a = **discrimination**
slope at prob = .50, (logit = 0, which is point of inflection)

Note: **equal a terms** means curves will never cross \rightarrow this idea is called "Specific Objectivity"

Can you guess what's next?

2-Parameter Logistic Model (2PL)

- The 1-PL (Rasch) model assumes tau-equivalence → equal discrimination
- Thus, the 2-PL frees this constraint by changing " a " to " a_i " (as fixed effect):

➤ **Logit:** $\text{Log} \left[\frac{p(y_{is}=1)}{1-p(y_{is}=1)} \right] = a_i(\theta_s - b_i)$

➤ **Probability:** $p(y_{is} = 1) = \frac{\exp[a_i(\theta_s - b_i)]}{1 + \exp[a_i(\theta_s - b_i)]}$

Relative to those in a logit-link model, parameters from a **probit-link** (ogive) model will be smaller by a factor of 1.7

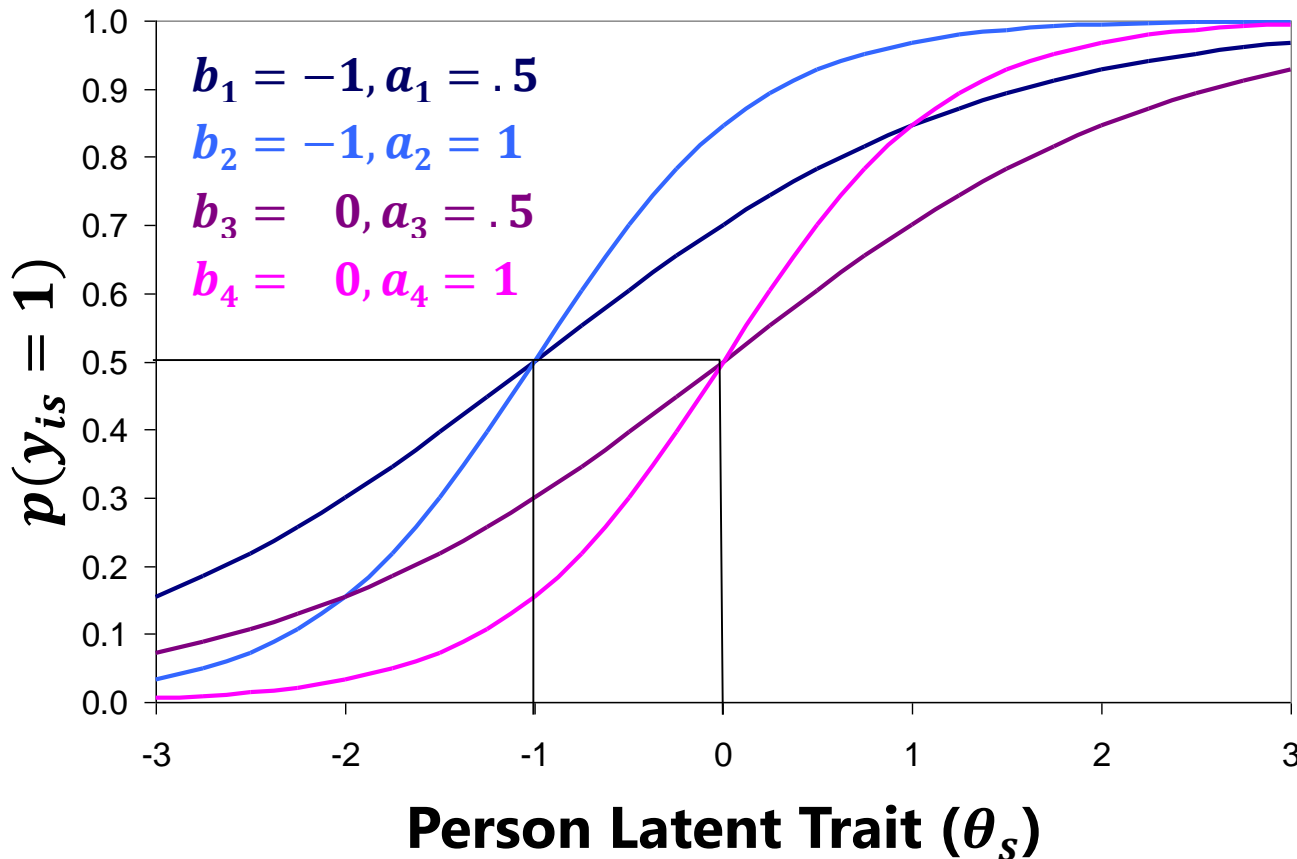
- a_i = "**item discrimination**" = relation of **each item** to latent trait = slope of the s-shape curve at probability = .50 (its max slope)
- b_i is still item difficulty (location where probability = .50)
- Note that a_i is a **linear** slope for theta θ predicting the **logit of $y_{is} = 1$** but a **nonlinear** slope for theta θ predicting the **probability of $y_{is} = 1$**

“IRT” vs. “Rasch”

- According to most **IRT** people, a “Rasch” model is just an IRT model with item discrimination a_i held equal across items (a tau-equivalent model)
 - Rasch = 1-PL where b_i item difficulty is the only item parameter
 - Slope = discrimination a_i = strength of relation of item to latent trait (theta θ_s)
 - *“Items may not be equally ‘good’, so why not just let their slopes vary?”*
- According to most **Rasch** people, the 2PL and rest of IRT is voo-doo
 - Rasch models have specific properties that are lost once you allow the item curves to cross (by using **item-varying a_i**) → **“Loss of Specific Objectivity”**
 - Under the Rasch model, persons are ordered the same in terms of predicted responses regardless of which item difficulty location you’re looking at
 - Under the Rasch model, items are ordered the same in terms of predicted responses regardless of what level of person theta you’re looking at
 - **a_i item discrimination represents a theta*item interaction** → the item curves cross, so the ordering of persons or items is no longer invariant, and this is “bad”
 - *“Items should not vary in discrimination if you know your construct.”*

Item Characteristic Curves: 2PL Model

b_i = **difficulty** = location on latent trait at which $p_i = .50$ (or logit = 0)
 a_i = **discrimination** slope at $p_i = .50$ (at the point of curve inflection)



Note: **unequal a_i**
→ curves cross
→ violates Specific Objectivity

At Theta $\theta_s = -1$:
Items 3 and 4 are harder than 1 and 2
→ lower prob of 1

At Theta $\theta_s = +2$:
Item 1 is now harder than Item 4 →
lower prob of 1

“IRT” vs. “Rasch”: What Goes into Theta

- In Rasch models, **the sum score is a “sufficient statistic”** for Theta (θ_s)
 - For example, given 5 items ordered in difficulty from easiest to hardest, each of these response patterns where **3/5 are correct** would yield the **same estimate of Theta**:

1 1 1 0 0 (most consistent)

0 1 1 1 0

0 0 1 1 1

1 0 1 0 1 (???)

.... (and so forth)

- In 2-parameter models, items with higher discrimination (a_i) **count more** towards Theta (and SE will be lower for instruments with higher a_i items)
 - It not only matters **how many** items you got correct, but **which ones**
 - Rasch people don't like this idea, because then the ordering of persons on Theta is dependent on the item properties

Yet Another Model for Binary Responses: 3-Parameter Logistic Model (3PL)

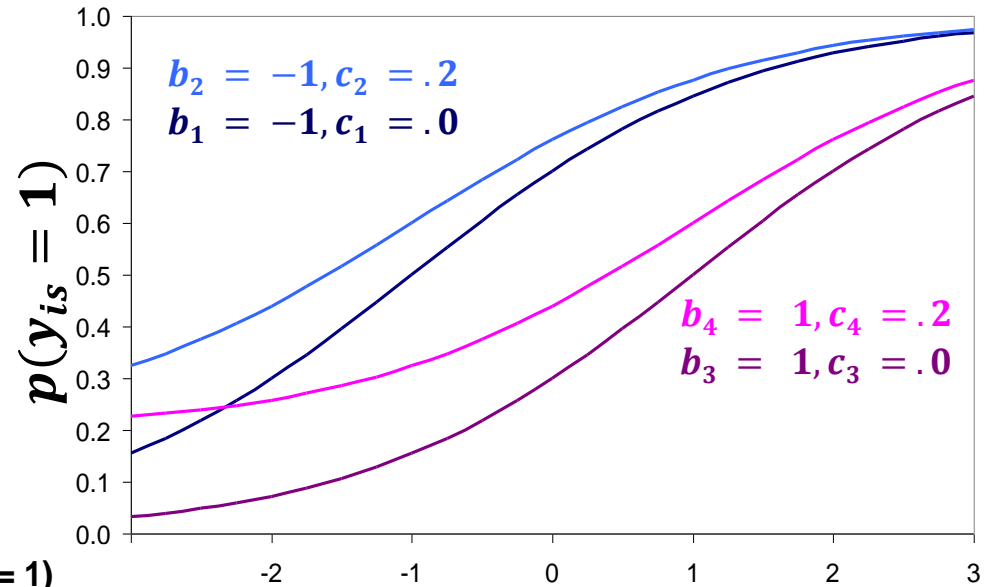
$$p(y_{is} = 1) = c_i + (1 - c_i) \left[\frac{\exp[a_i(\theta_s - b_i)]}{1 + \exp[a_i(\theta_s - b_i)]} \right]$$

- b_i = item difficulty \rightarrow location (still a fixed effect)
 - Higher values \rightarrow more difficult items (lower probability of a 1)
- a_i = item discrimination \rightarrow slope at b_i (still a fixed effect)
 - Higher values = more discriminating items = better items *at its location*
- c_i = item lower asymptote \rightarrow "guessing" (where $c_i > 0$; is a fixed effect)
 - Lower bound of probability of 1 independent of Theta
 - e.g., would be around .25 given 4 equally guess-able multiple-choice responses
 - Could estimate a common c across items as an alternative (but is not typically done)
- Probability starts at guessing c_i then depends on Theta θ_s , a_i , and b_i
 - 3-PL model is available in and after Mplus 7.4; c_i is labeled as \$2
 - Require 1000s of people because c_i parameters are hard to estimate—you must have enough low theta people to determine what the probability of guessing is likely to be

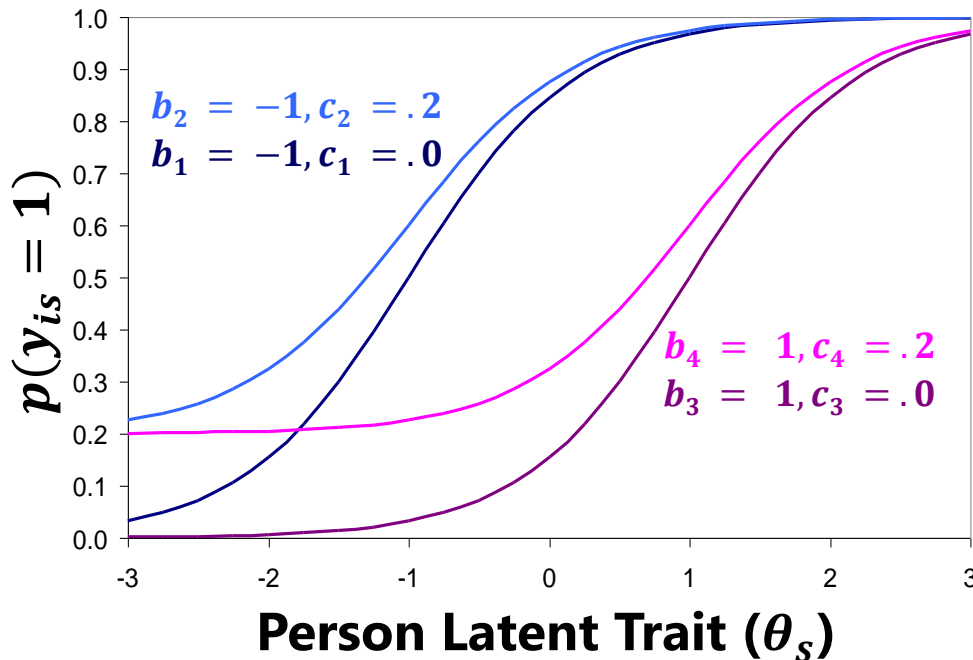
Top: Items with lower discrimination ($a_i = .5$)

Below: Items with higher discrimination ($a_i = 1$)

Item Characteristic Curves - 3-PL Model ($a = .5$)



Item Characteristic Curves - 3-PL Model ($a = 1$)



Person Latent Trait (θ_s)

Note that item difficulty b_i values are no longer where prob = .50 \rightarrow the expected prob at b_i is increased by the lower asymptote c_i parameter

One Last Model for Binary Responses: 4-Parameter Logistic Model (4PL)

$$p(y_{is} = 1) = c_i + (d_i - c_i) \left[\frac{\exp[a_i(\theta_s - b_i)]}{1 + \exp[a_i(\theta_s - b_i)]} \right]$$

- b_i = item difficulty \rightarrow location
- a_i = item discrimination \rightarrow slope
- c_i = item lower asymptote \rightarrow "guessing"
- d_i = item upper asymptote \rightarrow "**carelessness**" (so $d_i < 1$)
 - Maximum probability to be achieved independent of trait (Theta θ_s)
 - Could be carelessness or unwillingness to endorse the item no matter what
- Probability starts at "guessing" c_i , tops out at "carelessness" d_i , then in between depends on Theta θ_s , a_i , and b_i
 - 4-PL model is in or after Mplus 7.4; c_i and d_i are labeled as \$2 and \$3
 - But good luck estimating it! May need to use a common c and d instead

All item parameters
remain fixed effects

Latent Trait Measurement Models for Binary Responses: IRT and IFA

- Topics:
 - The Big Picture of Latent Trait Measurement Models
 - 1, 2, 3, and 4 Parameter IRT (and Rasch) Models
 - **Item and Test Information (for Indexing Reliability)**
 - From Item Response Theory Models to Item Factor Models
 - Model Estimation, Comparison, and Evaluation

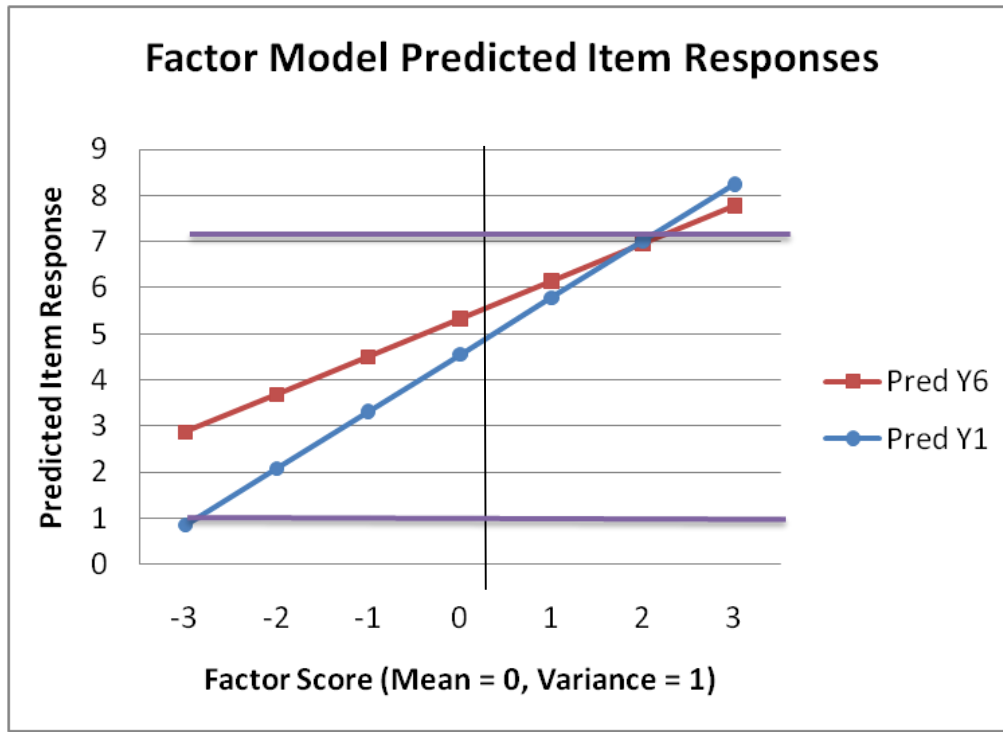
Anchoring: Model Identification in IRT

- As in CFA, we have a latent trait (a pretend variable) without a scale, so we need to give Theta θ_s a scale (it needs a mean and a variance)
 - This is called “**anchoring**” in IRT → CFA calls it “**model identification**”
 - As in CFA, there are 2 equivalent options: *Anchor by Persons* or *Anchor by Items*
- **Anchor by persons:** Fix Theta θ_s mean = 0 and Theta θ_s variance = 1
 - This is the “z-score” approach to model identification used in CFA
 - All item difficulties b_i and item discriminations a_i are then estimated
- **Anchor by items:** Fix one item difficulty $b_i = 0$ and one item $a_i = 1$
 - “Marker item” approach to model identification
 - Mean and variance of Theta θ_s are estimated instead
 - Fixing mean of item difficulty = 0 is another way (more common in Europe)
- Big picture: as in CFA, the numerical scale doesn’t matter, all that matters is that persons and items are on the same scale → “conjoint scaling”

Information: Reliability in IRT Models

- **“Information”** \approx reliability \rightarrow measurement precision
- In **CFA models** (continuous y_{is}), item-specific “information” is rarely referred to, because standardized loadings cover it:
 - How good is my item \rightarrow how much information is in it?
 - How much of its variance is “true” (shared with the factor) relative to how much of its variance is “error”?
 - **Information = unstandardized loading² / error variance**
 - Note that information will be **constant** across trait level in CFA
 - Items with a greater proportion of true variance are better, the end
 - So the instrument’s “information function” **is FLAT** across trait level
 - How do I make my test better?
 - **More items with more information** (with stronger factor loadings)
 - Sum of information across items = **Test information function**
 - Test information function will also be flat across trait level in CFA

Item Information in CFA Models



$$y_{6s} = 5.32 + 0.82(F_s) + e_{6s}$$
$$e_{6s}^2 = 1.67$$

$$y_{1s} = 4.55 + 1.23(F_s) + e_{1s}$$
$$e_{1s}^2 = 1.53$$

$$\text{Info } y_6 = 0.82^2 / 1.67 = .401$$

$$\text{Info } y_1 = 1.23^2 / 1.53 = .998$$

$$\text{Std } y_{6s} = 3.48 + 0.54(F_s) + e_{6s}$$

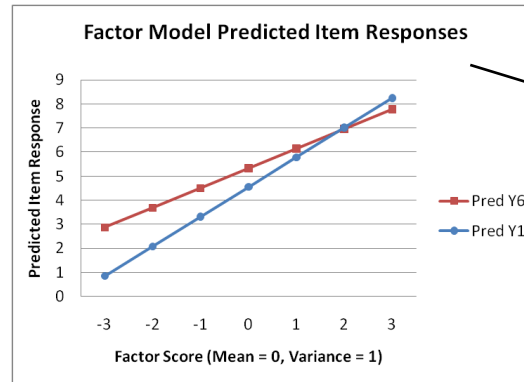
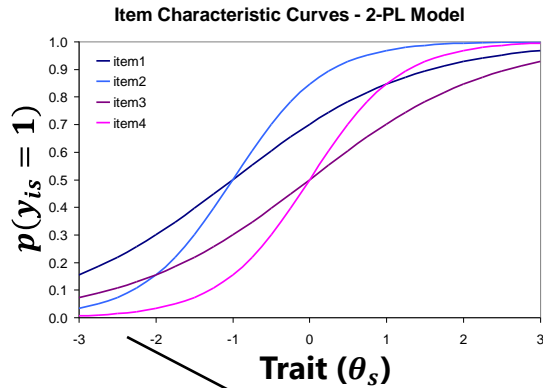
$$\text{Std } y_{1s} = 2.60 + 0.71(F_s) + e_{1s}$$

- CFA has a **linear slope (factor loading)** → predicts the same increase in the y_{is} item response for a one-unit change in F_s (all across levels of F_s)
- y_1 **has more information** than y_6 (and a higher standardized factor loading), so y_1 is better than y_6 , period (for all possible factor scores)

Test Information in IRT Models

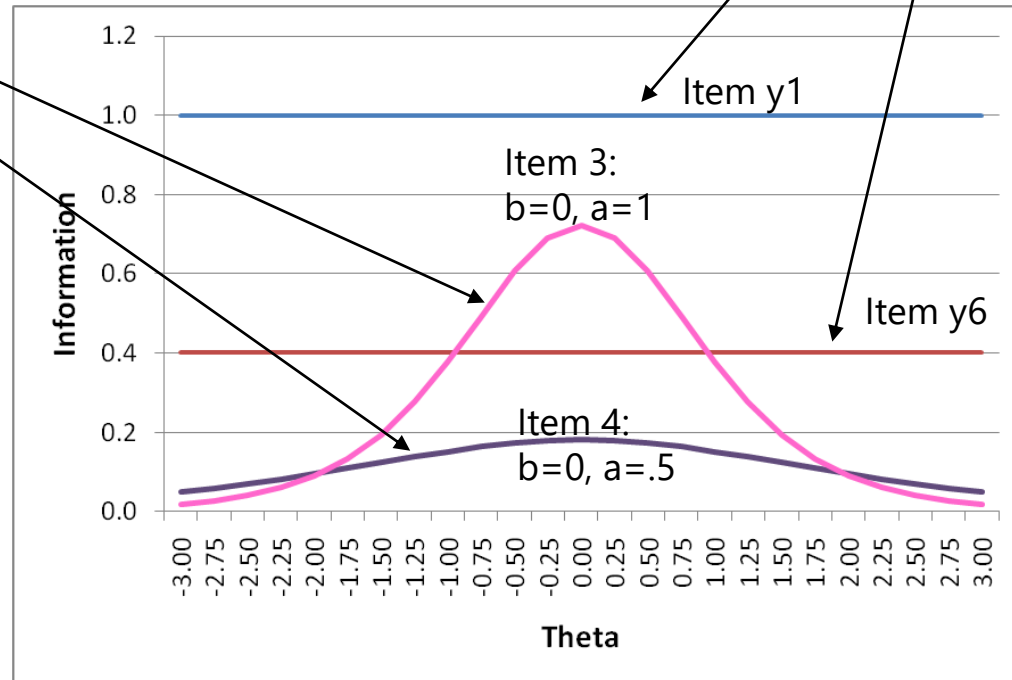
- Test information can be converted to reliability as follows:
 - **Reliability = information / (information + 1)**
 - **Information of 4 converts to reliability of .80**
 - **information of 9 converts to reliability of .90**
- This formula comes from classical test theory:
 - Reliability = true var / (true var + error var)
 - Reliability = 1 / (1 + error var), where error var = 1/info
 - Reliability = 1 / 1 + (1/info) → info / (info + 1)
- An analog of overall model-based reliability (omega) could be computed by summing reliabilities for each possible theta, weighted by the number of people at each level of Theta, but (to me) that's missing the point...
- Because the slopes relating Theta to the probability of an item response are non-linear, this means that **reliability must VARY over Theta**

Item Information in CFA vs. IRT



CFA Item Information Functions

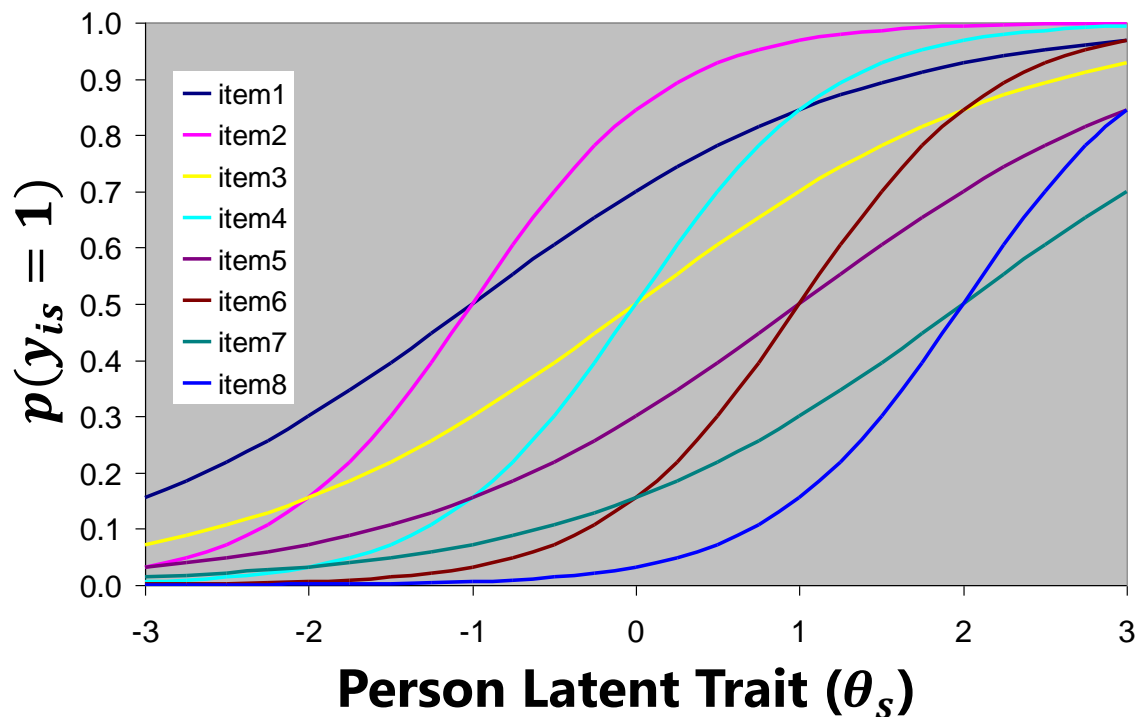
IRT Item Information Functions



Effects of Item Parameters on Item Characteristic Curves

Item	1	2	3	4	5	6	7	8
a discrimination	0.5	1.0	0.5	1.0	0.5	1.0	0.5	1.0
b difficulty	-1.0	-1.0	0.0	0.0	1.0	1.0	2.0	2.0

Item Characteristic Curves



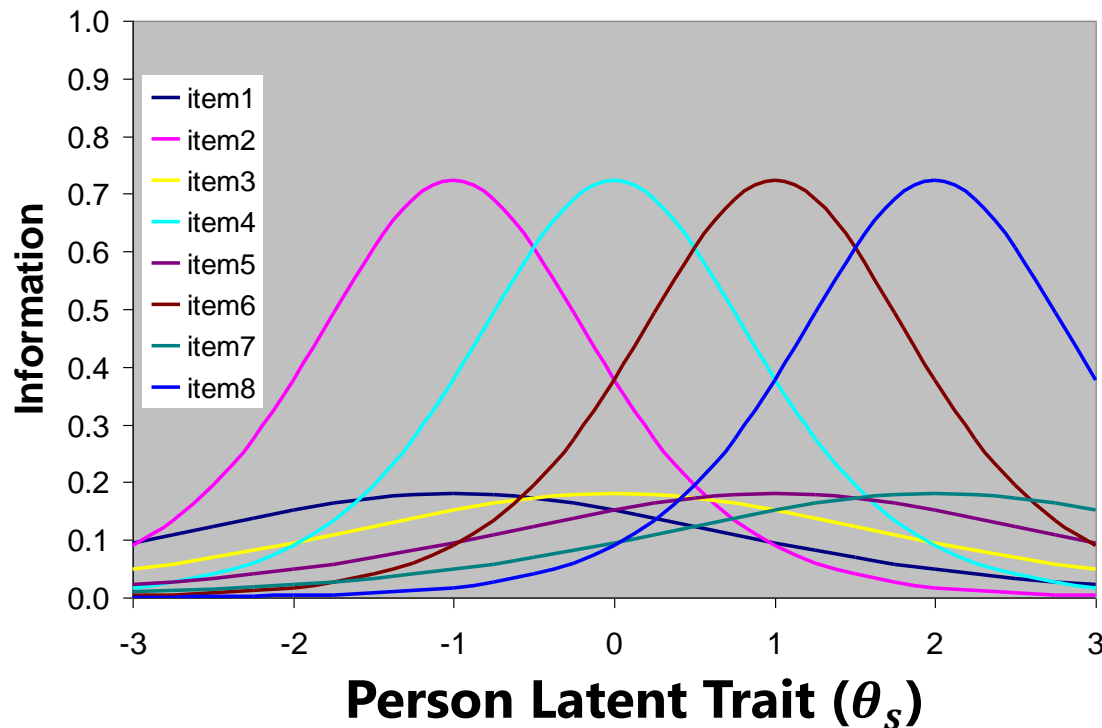
An important result of the non-linear slopes in an IRT model is that the **slope stops working** (so reliability decreases) as you move away from the item difficulty location.

In the **CFA** model with linear slopes, **the slope never stops working** (at least in theory).

Effects of Item Parameters on Item Information Curves

Item	1	2	3	4	5	6	7	8
a discrimination	0.5	1.0	0.5	1.0	0.5	1.0	0.5	1.0
b difficulty	-1.0	-1.0	0.0	0.0	1.0	1.0	2.0	2.0

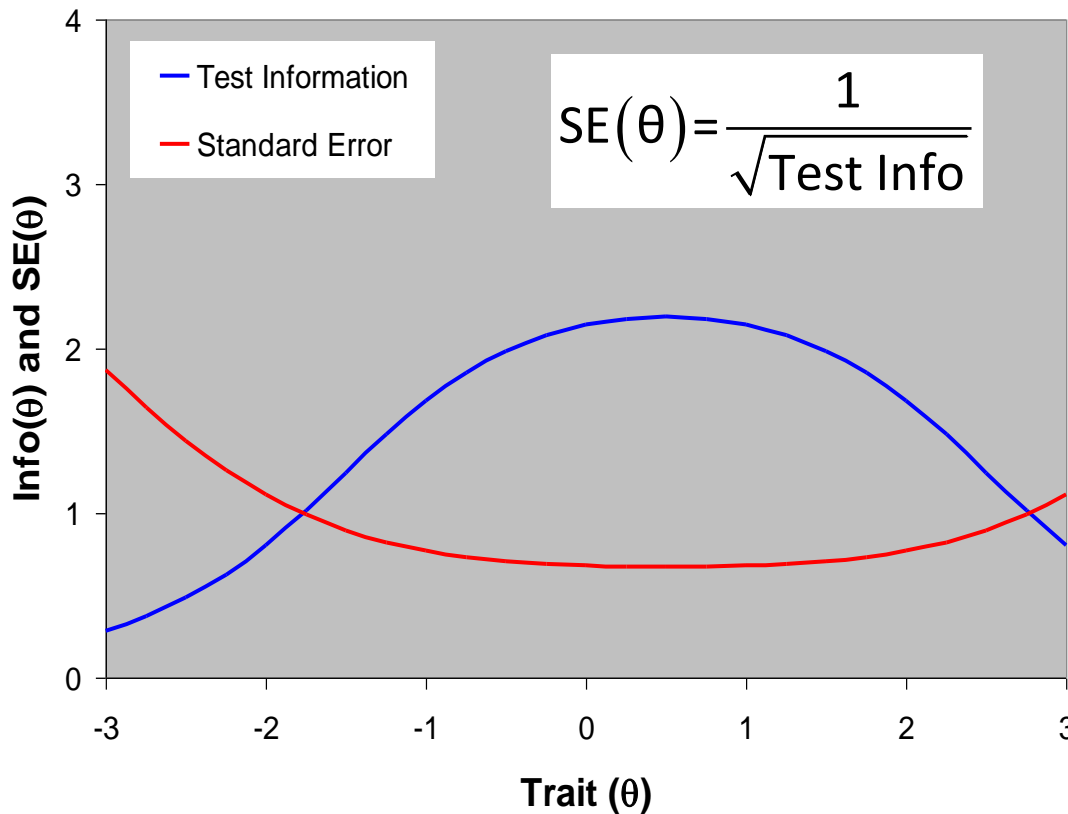
Item Information Functions



Information (reliability) is maximized around the item difficulty location.

Items with greater a_i item discrimination values have greater absolute information.

Test Information (and SE) by Theta



If you sum all the item information curves, you get a **test information** curve that describes how reliable your set of items is over the range of the trait (Theta).

Test Information is very useful to know—it can tell you where the holes are in your measurement precision, it and guides you in adding/removing items.

There is no single 'ideal' test information function—only what is optimal for **your** purposes of measurement. Here are a few examples....

Another Example of (Not-So-Good) Test Information

But test info only gets up to ~2...
(Uh oh!)

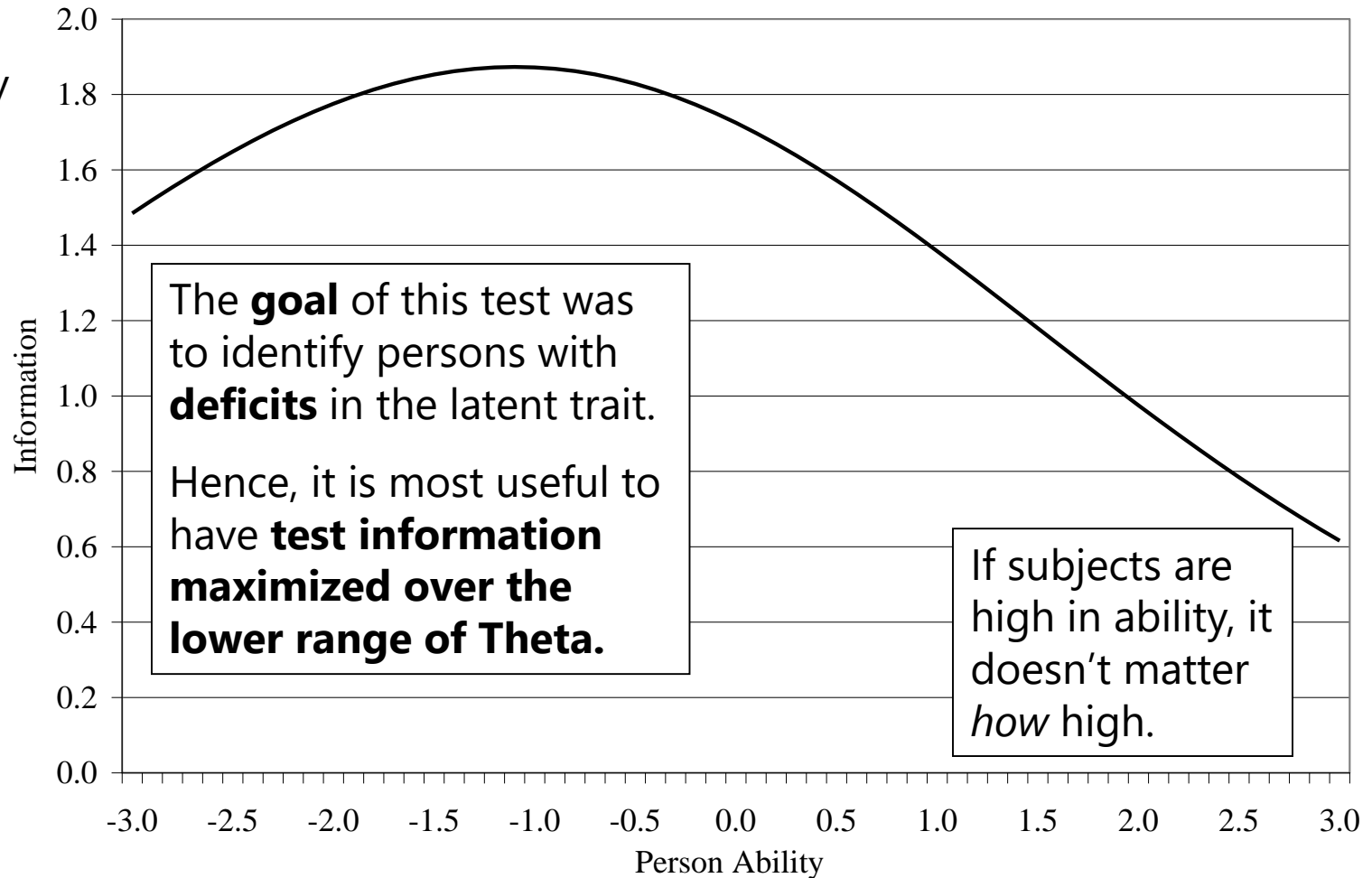
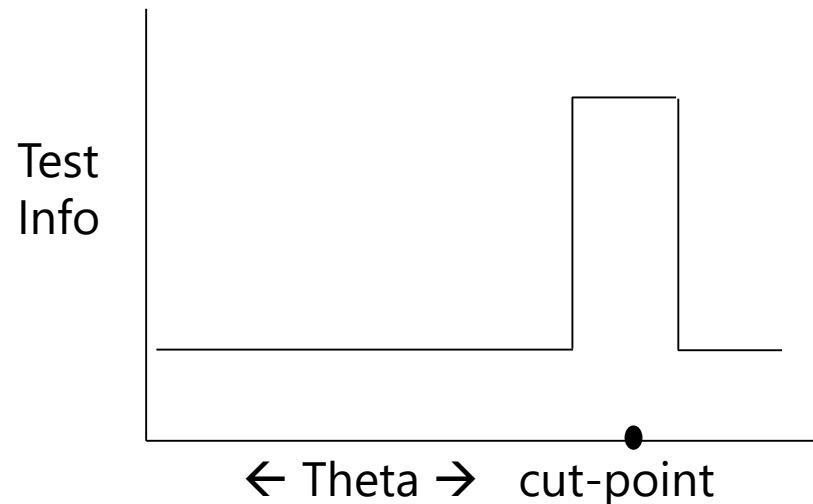


Image from my dissertation!

Other Shapes of Test Information

- If the goal is to measure a trait across persons equally well, and you expect people to be normally distributed, then your best bet is to create a test with information highest in the middle (where most people are likely to be)
- If your goal is to identify individuals below or above a cut-point, however, your **test information function** should ideally look more like this:
 - You'd want to **maximize sensitivity around the cut-point region**, and otherwise not waste time measuring people well who are nowhere near the cut-point
 - If **classifying people** is the goal of measurement, however, you might be better off with a different family of latent trait models in which Theta is already categorical instead
→ "**Diagnostic Classification Models**"



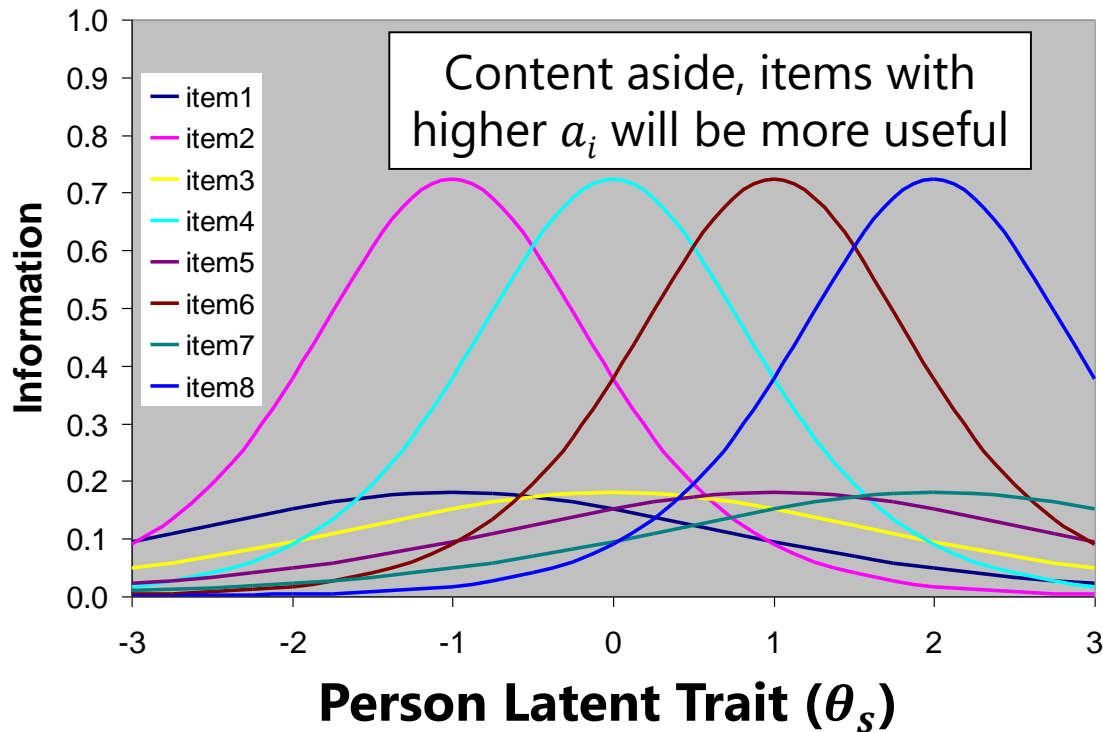
How to Improve Your Test

- In CTT, because item properties are not part of the model, items are seen as exchangeable, and more items is better
 - Thus, *any* new item is *equally* better for the model
- In CFA and IRT, more items is still better...
 - **In CFA, the question is “how much better”?**
 - This depends on the standardized loading; intercepts are not important
 - Specifies a **linear relationship** between theta and the item responses, so “for whom” isn’t relevant—a better item is **better for everyone equally**
 - **In IRT, the question is “how much better, and for whom?”**
 - Depends on the discrimination (a_i slope) and the difficulty (b_i location), respectively (difficulties are important, and are always estimated)
 - Because of the **nonlinear relationship** between theta and the item responses, items are **only useful for thetas in the middle of their S-curves**

Effects of Item Parameters on Item Information Curves

Item	1	2	3	4	5	6	7	8
a discrimination	0.5	1.0	0.5	1.0	0.5	1.0	0.5	1.0
b difficulty	-1.0	-1.0	0.0	0.0	1.0	1.0	2.0	2.0

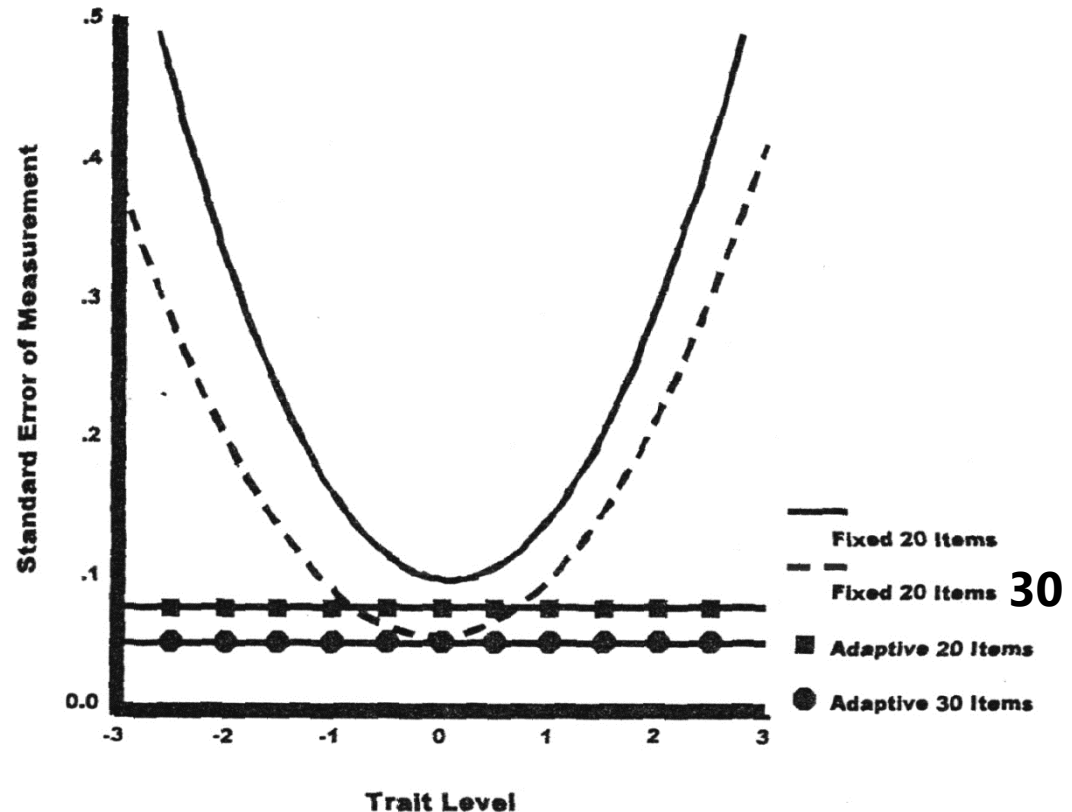
Item Information Functions



In addition to a_i item discrimination, though, you want to make sure you are covering the range of difficulty where you want to measure people best.

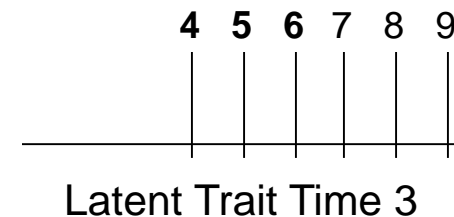
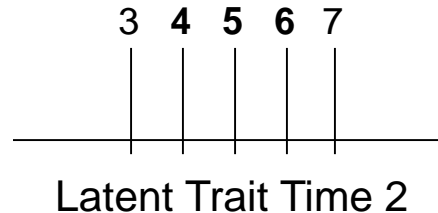
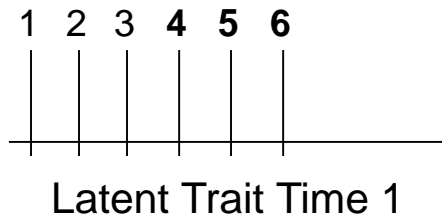
IRT and Adaptive Testing: *Fewer Items Can Actually Be Better*

- In a normal distribution of the latent trait and a comparable distribution of item difficulty, **extreme people are usually measured less well** (higher SE).
- For fixed-item tests, more items is generally better, but one can get the same precision of measurement with fewer items by using **adaptive tests with items of targeted levels of difficulty**. Different forms across person are given to maximize efficiency.

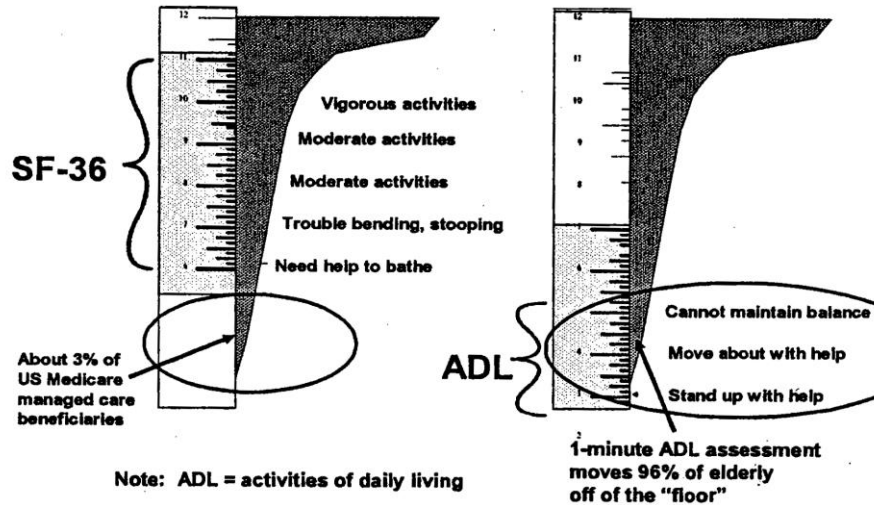


IRT (and CFA) Help Measure Change AND Maintain Sensitivity across Samples

- **Theta is scaled and interpreted relative to the items**, not relative to the other persons in the sample (is item difficulty at $prob = .50$)
 - This means that given a set of pre-calibrated “linking” items, you can administer different forms of a test and still get comparable Thetas
 - Must have some “**linking items**” → common set of items across forms
 - Although this property is helpful when dealing with “accidental” alternative forms (e.g., changed response options, dropped items), linking items can be used advantageously as well
 - Here, **we grow a test over time** within a sample (i.e., using “vertical scaling”):



Combining Measures Increases the Range & Lowers the Physical Function “Floor”



Example: Items from Many Forms Define the Physical Functioning (“Ruler”)



Source: Health Assessment Lab (HAL)

Linking Thetas across Tests

SF-36: measure of *higher* physical functioning

ADL: measure of *lower* physical functioning

Don't choose: Administer a core set of linking items from both tests to a single sample

Linking items then form a common metric

- More precision than single test
- Allows for comparisons across groups or studies

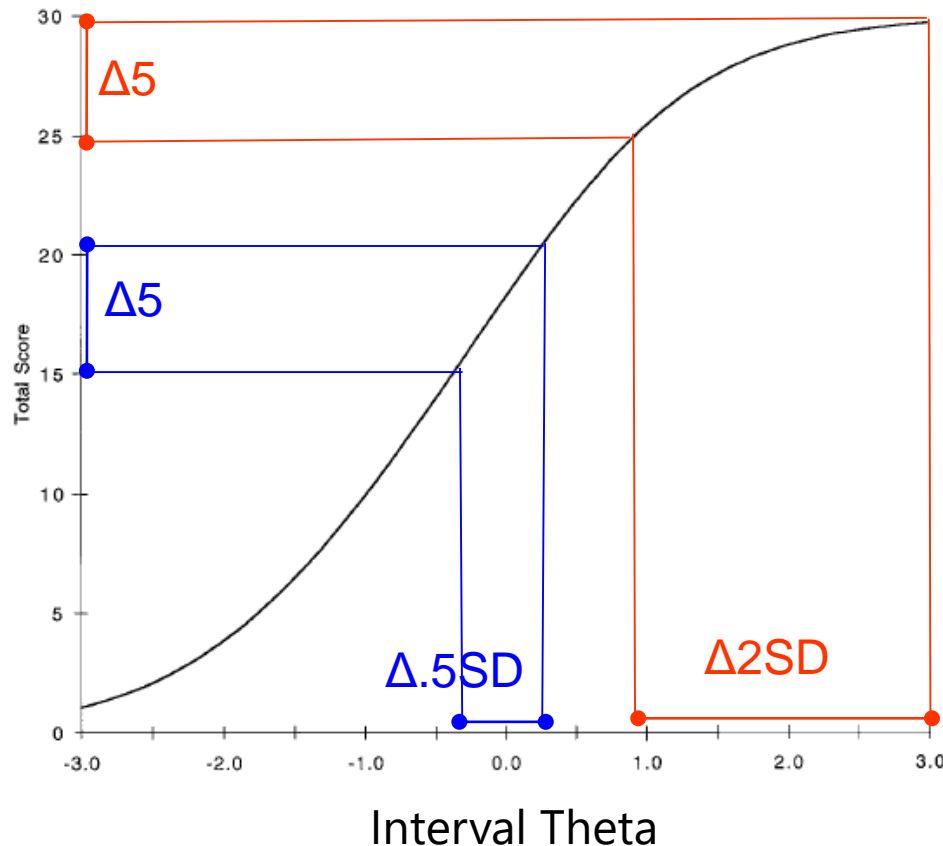
See Mungas & Reed (2000) for an example of linking over forms

Another Benefit of IRT (and CFA)

- **IRT**: If the model fits, the scale of **Theta is linear/interval**
 - Supports mathematical operations that assume interval measurement
 - Same ordering of persons as in raw scores, but the distances between persons are likely to be different, especially at the ends
- **CTT: Sum scores** have an **ordinal** relationship to the latent trait at best
 - Does not support operations that assume interval measurement, which can bias tests of mean differences, regression slopes, etc.
 - Spurious interactions can result in tests of mean differences if groups differ in how well they are measured (i.e., floor and ceiling effects)
- Bottom line: Measurement matters for testing everyday hypotheses, NOT just when fitting measurement models for specific issues

Example from Mungas & Reed (2000)

Test Curve for MMSE Total



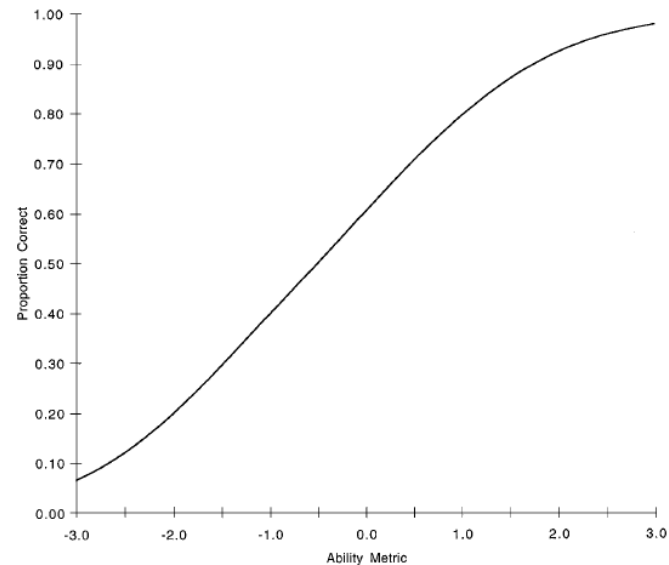
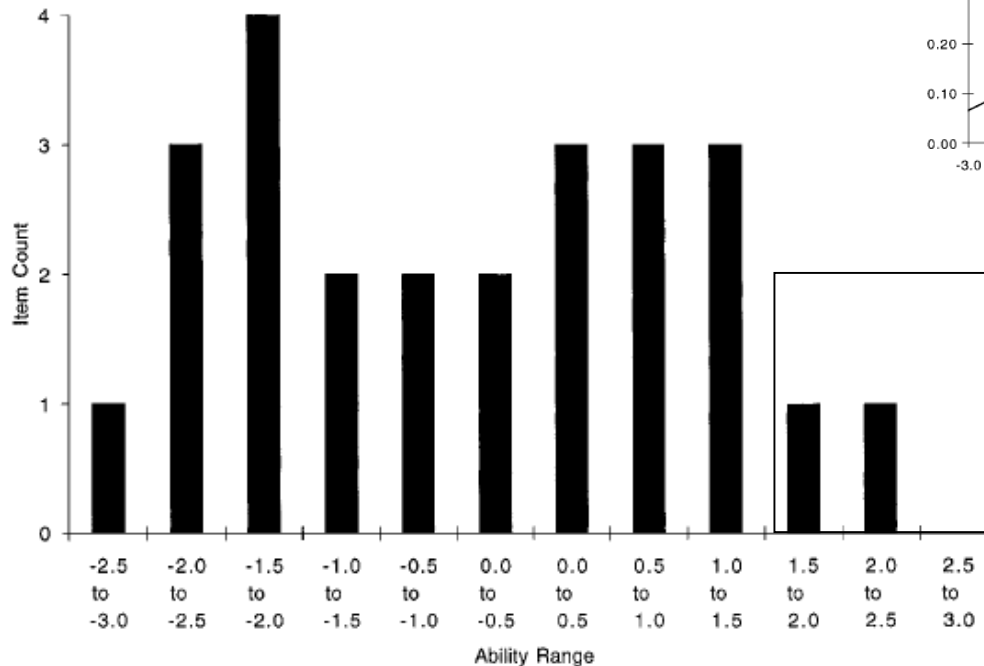
The bottom and top of the MMSE total score (ordinal) are "squished" relative to the latent trait scale (interval).

This means that one-unit changes along the MMSE total do not really have the same meaning across the latent trait, which makes many kinds of comparisons problematic.

Example from Mungas & Reed (2000)

Right: They combined 3 tests to get better measurement, as shown in the test curve →

Below: Items at each trait location contribute to scale's capacity to differentiate persons in ability at each point in the continuum.



There is a hole near the top, which explains the flattening of the curve (less information there).

Latent Trait Measurement Models for Binary Responses: IRT and IFA

- Topics:
 - The Big Picture of Latent Trait Measurement Models
 - 1, 2, 3, and 4 Parameter IRT (and Rasch) Models
 - Item and Test Information (for Indexing Reliability)
 - **From Item Response Theory (IRT) Models to (IFA) Item Factor Models**
 - Model Estimation, Comparison, and Evaluation

Relating Item Factor Analysis (IFA) to Item Response Models (IRT)

- CFA is to linear regression as IRT is to logistic regression, right?
- **Linear regression model** and **CFA model** (for continuous responses):

$$y_{is} = \beta_{0i} + \beta_{1i}x_s + e_{is} \quad y_{is} = \mu_i + \lambda_i F_s + e_{is}$$

- **Logistic regression model** (for 0/1 responses, so there is no e_{is} residual):

$$\text{Log} \left[\frac{p(y_{is}=1)}{1-p(y_{is}=1)} \right] = \beta_{0i} + \beta_{1i}x_s$$

- **2-PL IRT model** (for 0/1 responses, so there is no e_{is} residual):

$$\text{Log} \left[\frac{p(y_{is}=1)}{1-p(y_{is}=1)} \right] = a_i(\theta_s - b_i)$$

Why does this IRT model look so different than the CFA model?
Here's how these all relate...

Relating IFA and IRT

- Linear regression model and

$$y_{is} = \beta_{0i} + \beta_{1i}x_s + e_{is}$$

- CFA model:

$$y_{is} = \mu_i + \lambda_i F_s + e_{is}$$

- Binary regression models and

$$\text{Logit}[p(y_{is} = 1)] = \beta_{0i} + \beta_{1i}x_s$$

$$\text{Probit}[p(y_{is} = 1)] = \beta_{0i} + \beta_{1i}x_s$$

- Our new IFA models:

$$\text{Logit}[p(y_{is} = 1)] = -\tau_i + \lambda_i F_s$$

$$\text{Probit}[p(y_{is} = 1)] = -\tau_i + \lambda_i F_s$$

- IRT models:

2PL: $\text{Logit}[p(y_{is} = 1)] = a_i(\theta_s - b_i)$

Ogive: $\text{Probit}[p(y_{is} = 1)] = a_i(\theta_s - b_i)$

Logit to Probability:

$$\text{prob} = \frac{\exp(\text{logit})}{1 + \exp(\text{logit})}$$

- In CFA, item loading λ_i = “**discrimination**” and item intercept μ_i = “**difficulty**”, but difficulty was backwards (easier or less severe items had higher means)...
- In IFA for binary items within Mplus, the **intercept** μ_i (which was really “**easiness**”) becomes a “**threshold**” τ_i that really IS “**difficulty**”: $\mu_i = -\tau_i$
 - this provides continuity of direction with the IRT difficulty values
- The IRT and IFA models get re-arranged into each other as follows...

From IFA to IRT

IFA with “easiness” **intercept** μ_i : **Logit or Probit** $y_{is} = \mu_i + \lambda_i F_s$ $\mu_i = -\tau_i$

IFA with “difficulty” **threshold** τ_i : **Logit or Probit** $y_{is} = -\tau_i + \lambda_i F_s$

IFA model with “difficulty” thresholds can be written as a **2-PL IRT Model**:

IRT model:

$$\text{Logit or Probit } y_{is} = a_i(\theta_s - b_i) = \underbrace{-a_i b_i}_{\tau_i} + \underbrace{a_i \theta_s}_{\lambda_i}$$

IFA model:

a_i = discrimination
 b_i = difficulty
 $\theta_s = F_s$ latent trait

Convert IFA to IRT:

$$a_i = \lambda_i * \sqrt{\text{Theta Variance}}$$

$$b_i = \frac{\tau_i - (\lambda_i * \text{Theta Mean})}{\lambda_i * \sqrt{\text{Theta Variance}}}$$

Convert IRT to IFA:

$$\lambda_i = \frac{a_i}{\sqrt{\text{Theta Variance}}}$$

$$\tau_i = a_i b_i + \frac{a_i * \text{Theta Mean}}{\sqrt{\text{Theta Variance}}}$$

Note: These formulas rescale a_i and b_i so that theta M=0, VAR=1

If you don't want to rescale theta, use M=0 and VAR=1 to keep your current scale

Thus, IFA = IRT

IRT:

$$\text{Logit or Probit } y_{is} = a_i(\theta_s - b_i) = \underbrace{-a_i b_i}_{\tau_i} + \underbrace{a_i \theta_s}_{\lambda_i}$$

IFA:

-
- An item factor model for binary outcomes is the same as a two-parameter IRT model, so you can keep both camps happy:
 - IFA loadings $\lambda_i \rightarrow$ 2-PL IRT discriminations a_i
 - IFA thresholds $\tau_i = -\mu_i \rightarrow$ 2-PL IRT difficulties b_i
-
- CFA/SEM crowd? Call it **Logit or Probit** $y_{is} = -\tau_i + \lambda_i F_s$
 - “I did IFA” \rightarrow Report item “factor loadings” λ_i and “thresholds” τ_i
 - See also “CFA for categorical data” as sufficiently synonymous
 - IRT crowd? Call it **Logit or Probit** $y_{is} = a_i(\theta_s - b_i)$
 - “I did IRT” \rightarrow Report item “discriminations” a_i and “item difficulties” b_i

3 Kinds of Output in Mplus

- **IFA unstandardized solution:**

- **Item threshold** τ_i = expected logit/probit of $y_{is} = 0$ when $\theta_s = 0$
- **Item loading** λ_i = Δ in logit/probit of $y_{is} = 1$ for a 1-unit Δ in θ_s (Theta)
- Item residual variance not estimated, but is 3.29 in logit or 1.00 in probit for y_{is}^*

- **IFA standardized solution:**

- Variance of logit/probit (of $y_{is} = 1$) $\rightarrow (\lambda_i^2 * \text{Theta Variance}) + (3.29 \text{ or } 1)$
- **std** τ_i = unstd λ_i / SD(Logit or Probit Y) \rightarrow not usually interpreted
- **std** λ_i = unstd λ_i * SD(Theta) / SD(Logit or Probit Y)
 \rightarrow correlation of logit/probit of item response with Theta

The IFA solution **cannot** be used to compute Omega.

- **IRT solution (only one type; only given for binary items):**

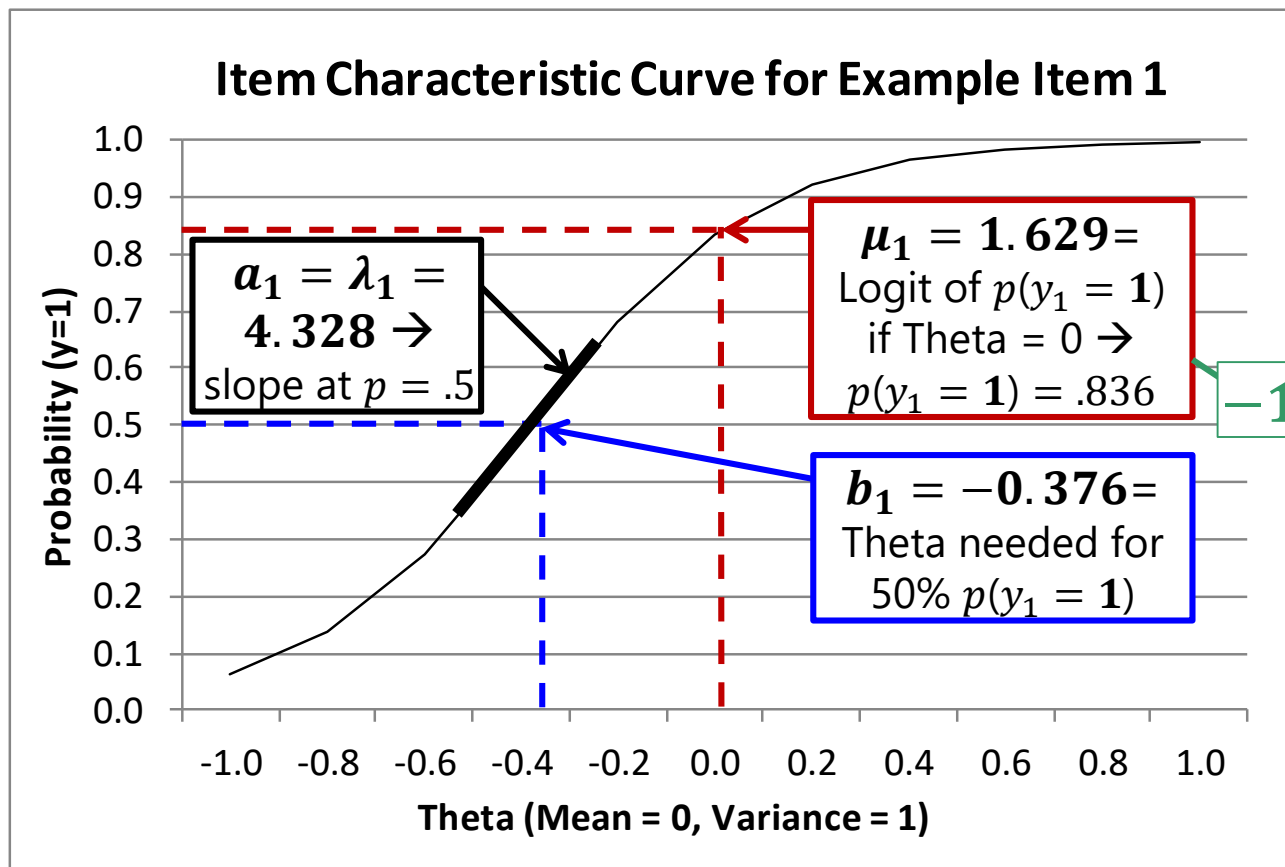
- b_i = Theta at which $\text{prob}(y_{is} = 1) = .50$ or logit or probit = 0
- a_i = Δ in logit or probit of $y_{is} = 1$ for a 1-unit Δ in θ_s (Theta)
= slope of item characteristic curve at b_i item difficulty location

Item Parameter Interpretations

IFA model with **loading** and “easiness” **intercept** μ_i : **Logit** $y_{is} = \mu_i + \lambda_i F_s$

IFA model with **loading** and “difficulty” **threshold** τ_i : **Logit** $y_{is} = -\tau_i + \lambda_i F_s$

2-PL IRT model with **discrimination** and **difficulty**: **Logit** $y_{is} = a_i(\theta_s - b_i)$



From IRT to IFA:

$$\lambda_i = a_i$$

$$\tau_i = a_i b_i$$

$-1 *$

$$\tau_1 = -1.629 =$$
 Logit of $p(y_1 = 0)$ if Theta = 0 \rightarrow $p(y_1 = 0) = .164$

Item Parameter Interpretation

IFA model with **loading** and “easiness” **intercept** μ_i : **Logit** $y_{is} = \mu_i + \lambda_i F_s$

IFA model with **loading** and “difficulty” **threshold** τ_i : **Logit** $y_{is} = -\tau_i + \lambda_i F_s$

2-PL IRT model with **discrimination** and **difficulty**: **Logit** $y_{is} = a_i(\theta_s - b_i)$

- IFA and IRT item slope parameters are interpreted similarly:
 - IFA loading $\lambda_i = \Delta$ in logit/probit of $y_{is} = 1$ for a 1-unit Δ in Theta
 - IRT discrimination $a_i =$ slope of ICC at prob=.50 (logit/probit = 0)
- IFA and IRT item location parameters are interpreted differently:
 - **IFA intercept** $\mu_i =$ logit/probit of $y_{is} = 1$ when **Theta = 0**
 - **IFA threshold** $\tau_i =$ logit/probit of $y_{is} = 0$ when **Theta = 0**
 - **IRT difficulty** $b_i =$ **amount of Theta needed** for logit/probit of $y_{is} = 1$
 - So b_i difficulty values are more interpretable as measures of **location**

CFA vs. IRT/IFA vs. ???

- CFA assumes continuous, normally distributed item responses
 - Robust ML can be used to adjust fit statistics and parameter SEs for non-normality, but it's still a **linear model** for the Factor predicting y_{is}
 - A linear model may not be plausible for Likert item responses (i.e., the model-predicted responses may extend beyond the possible response options for possible Factor levels)
- IRT/IFA is for categorical, binomial/multinomial item responses
 - **Linear model between Theta and logit/probit(y_{is}) instead**
 - Because Likert item responses are bounded and only ordinal, not interval, IRT/IFA should probably be used for this kind of data
 - CFA may not be too far off given ≥ 5 normally distributed responses, but then you can't see how useful your answer choices are (stay tuned)
- For non-normal but continuous (not categorical) responses, other latent trait measurement models are available (stay tuned)

Summary: Binary IRT/IFA Models

- IRT/IFA are a family of models that specify the relationship between the latent trait ("Theta") and a link-transformation of probability of $y_{is} = 1$
 - **Linear** relationship between Theta and **Logit or Probit** of $y_{is} = 1$
→ **nonlinear** relationship between Theta and **Probability** of $y_{is} = 1$
- The form of the relationship depends on:
 - At least the location on the latent trait (given by difficulty b_i or threshold τ_i)
 - Perhaps the strength of relationship may vary across items (given by a_i or λ_i)
 - If not, its a "1-PL" or "Rasch model" → assumes tau-equivalence
 - Also maybe lower and upper asymptotes (c_i and d_i) → but good luck with that!
- Because the slopes are non-linear, this implies that **reliability** (now called "test information") **must vary** across levels of theta
 - So items are not just "good" or "bad", but are "good" or "bad" for whom?
- **Now what about model fit??? Let's talk estimation first...**

Latent Trait Measurement Models for Binary Responses: IRT and IFA

- Topics:
 - The Big Picture of Latent Trait Measurement Models
 - 1, 2, 3, and 4 Parameter IRT (and Rasch) Models
 - Item and Test Information (for Indexing Reliability)
 - From Item Response Theory Models to Item Factor Models
 - **Model Estimation, Comparison, and Evaluation**

What all do we have to estimate?

- For example, a 7-item binary test and a 2-PL model, (assuming we fix the Theta distribution to have mean=0 and variance=1):
 - 7 item discriminations (a_i) and 7 item difficulties (b_i) = 14 parameters
- **Item parameters** are **FIXED effects** → specific item inference
 - Missing data can lead to different numbers of total items across persons
- What about the all the individual person **Thetas**?
 - The individual factor scores are not part of the model—in other words, Theta scores are modeled as **RANDOM effects** (= U's in MLM, btw)
 - Thus, our inference is about the distribution of the latent traits in the population of persons, which we assume to be multivariate normal
 - i.e., we care about the **Theta means, variances, and covariances** in the sample, but **not** about the Theta estimates for each **individual** per se

Estimation: Items, then People

3 full-information item estimation methods:

- **“Full-information”** → uses individual item responses
- 3 methods differ with respect to how they handle unknown person thetas
- First, two less-used and older methods:
 - **“Conditional” ML** → *Theta? We don't need no stinking theta...*
 - Uses total score as “Theta” (so can't include people with all 0's or all 1's)
 - Thus, is only possible within Rasch models (where the total is sufficient for theta)
 - If the Rasch model holds, estimators are consistent and efficient and can be treated like true likelihood values (i.e., can be used in model comparisons)
 - **“Joint” ML** → *Um, can we just pretend the thetas are fixed effects?*
 - Iterates back and forth between persons and items (each as fixed effects) until item parameters don't change much—then calls it done (i.e., converged)
 - Many disadvantages: estimators are biased, inconsistent, with too small SEs and likelihoods that can't be used in model comparisons
 - More persons → more parameters to estimate, too → so bad gets even worse

Marginal ML Estimation (with Numeric Integration)

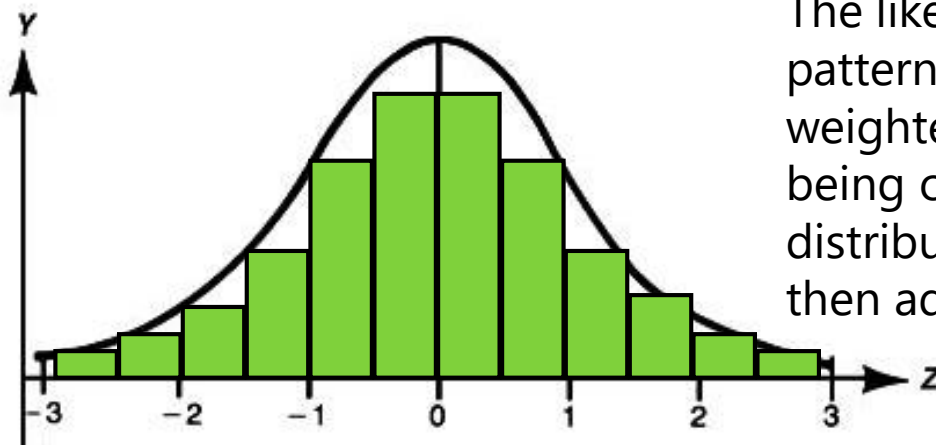
- Gold standard of estimation (and used in Mplus)
 - This is the same idea of multivariate height, just using a different distribution than multivariate normal for the log-likelihood function
- Relies on two assumptions of **independence**:
 - Item responses are independent after controlling for Theta: “local”
 - This means that the joint probability (likelihood) of two item responses is just the probability of each multiplied together
 - Persons are independent (no clustering or nesting)
 - You can add random effects to handle dependence, but then the assumption is “independent after controlling for random effects”
- Doesn't assume it knows the individual thetas, but it does assume that the *distribution* of theta(s) is (multivariate) normal

Marginal ML via Numeric Integration

- Step 1: Select starting values for all item parameters (e.g., using CTT values)
- Step 2: Compute the **likelihood for each person** given by the *current* parameter values (using start values or updated values later on)
 - IRT model gives probability of response given item parameters and Theta
 - To get likelihood per person, take each predicted item probability and plug them into: **Likelihood (all responses) = Product over items of: $p^y(1-p)^{1-y}$**
 - But we don't have Theta yet! No worries: computing the likelihood for each set of possible parameters requires *removing* the individual Thetas from the model equation—by **integrating** across the possible Theta values for each person
 - Integration is done by "Gaussian Quadrature" → summing up rectangles that approximate the integral (the area under the curve) for each person
- **Step 3:** Decide if you have the right answers, which occurs when the sum of the log-likelihoods changes very little across iterations (i.e., it converges)
- **Step 4:** If you aren't converged, choose new parameters values
 - Newton-Rhapson or Fisher Scoring (calculus), EM algorithm (Thetas = missing data)

“Marginal” ML Estimation

- More on Step 2: Divide the Theta distribution into rectangles
 - “**Gaussian Quadrature**” (# rectangles = # “**quadrature points**”)
 - Divide the whole distribution into rectangles, and then take the most likely section for each person and rectangle that more specifically
 - This is “**adaptive quadrature**” and is computationally more demanding, but gives more accurate results with fewer rectangles (Mplus uses 15)



The likelihood of each person’s response pattern at each Theta rectangle is then weighted by that rectangle’s probability of being observed (as given by the normal distribution). The weighted likelihoods are then added together across all rectangles. → ta da! “**numeric integration**”

- Unfortunately, each additional Theta or Factor adds another dimension of integration (so 2 factors = 15×15 rectangles to try at each iteration)

Example of Numeric Integration

- Start values for item parameters (here with $a = 1$):
 - Item 1: mean = .73 \rightarrow logit = +1, so starting $b_1 = -1$
 - Item 2: mean = .27 \rightarrow logit = -1, so starting $b_2 = +1$
- Compute per-person likelihood using item parameters and possible Thetas (-2,0,2) with IRT model: $\text{logit}(y_{is} = 1) = a(\theta - b_i)$

	Theta = -2	Logit	IF y=1 Prob	IF y=0 1-Prob	Likelihood if both y=1	Theta prob	Theta width	Product per Theta
Item 1 b = -1	(-2 - -1)	-1	0.27	0.73	0.0127548	0.05	2	0.001275
Item 2 b = +1	(-2 - 1)	-3	0.05	0.95				
	Theta = 0	Logit	Prob	1-Prob				
Item 1 b = -1	(0 - -1)	1	0.73	0.27	0.1966119	0.40	2	0.15729
Item 2 b = +1	(0 - 1)	-1	0.27	0.73				
	Theta = +2	Logit	Prob	1-Prob				
Item 1 b = -1	(2 - -1)	3	0.95	0.05	0.6963875	0.05	2	0.069639
Item 2 b = +1	(2 - 1)	1	0.73	0.27				

Overall Likelihood (Sum of Products over All Thetas): 0.228204

(then multiply over all people)

(repeat with new values of item parameters until find highest overall likelihood)

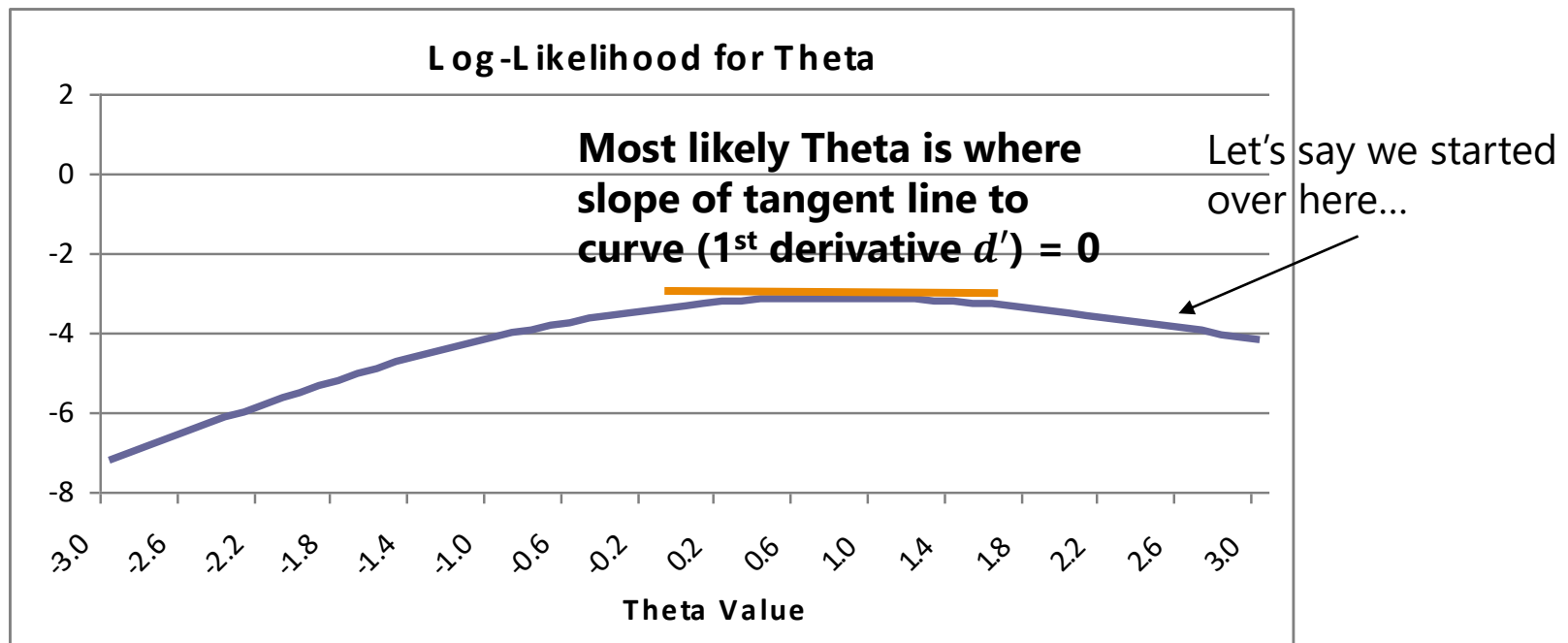
Once we have the item parameters, we need some Thetas...

- Let's say we are searching for Theta given observed responses to 5 items with known difficulty values, so we try out 2 possible Thetas
 - **Step 1:** Compute $\text{prob}(Y)$ using IRT model given each possible Theta
 - $b_1 = -2, \theta_s = -1$: $\text{Logit}(y_{is} = 1) = (-1 - -2) = 1$, so $p(y_{is} = 1) = .73$
 - $b_5 = 2, \theta_s = -1$: $\text{Logit}(y_{is} = 1) = (-1 - 2) = -3$, so $p(y_{is} = 1) = .05 \rightarrow p(y_{is} = 0) = .95$
 - **Step 2:** Multiple item probabilities together \rightarrow product = "likelihood"
 - Products get small really fast, but can take the log, and then add them instead
 - **Step 3:** See which Theta has the highest likelihood (here, +2)
 - More quadrature points
 \rightarrow better estimate of Theta
 - **Step 4:** Because people are independent, we can multiply all their response likelihoods together and solve all at once

Item	b	Y	Term	Value if...	
				$\theta = -1$	$\theta = +2$
1	-2	1	p	0.73	0.98
2	-1	1	p	0.50	0.95
3	0	1	p	0.27	0.88
4	1	1	p	0.12	0.73
5	2	0	1-p	0.95	0.50
Product of values:				0.01	0.30

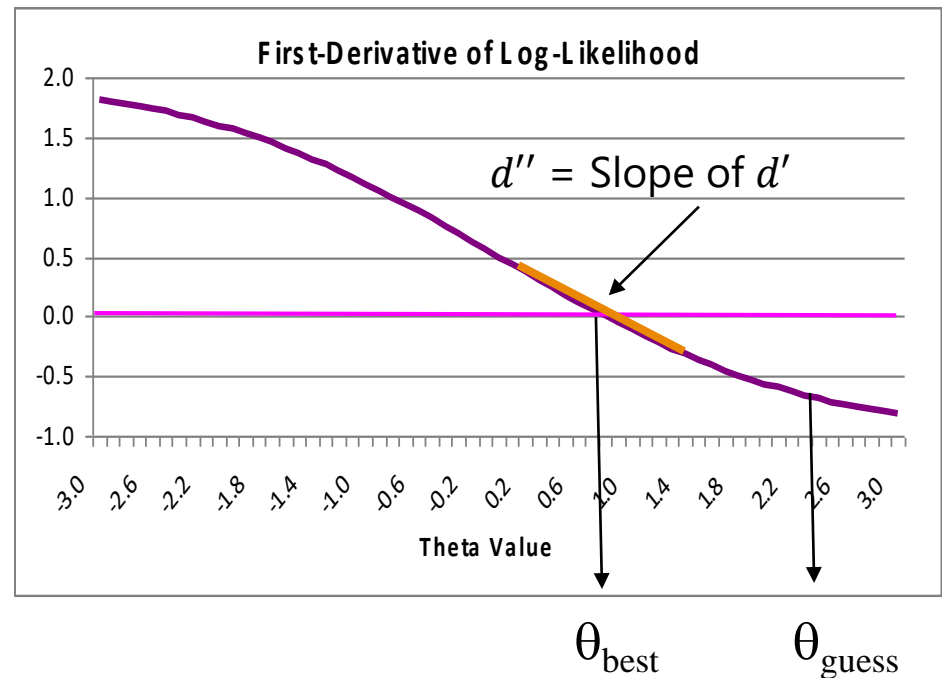
Theta Estimation via Newton Raphson

- We could calculate the likelihood over wide range of Thetas for each person and plot those likelihood values to see where the peak is...
 - But we have lives to lead, so we can solve it mathematically instead by finding where the slope of the likelihood function (the 1st derivative, d') = 0 (its peak)
- Step 1: Start with a guess of Theta, **calculate 1st derivative d'** at that point
 - Are we there ($d' = 0$) yet? Positive d' = too low; negative d' = too high



Theta Estimation via Newton Raphson

- Step 2: **Calculate the 2nd derivative** (slope of slope, d'') at current theta guess
 - Tells us **how far off we are**, which is used to figure out how much to adjust by
 - d'' will always be negative as we approach top, but d' can be positive or negative
- Calculate new guess of Theta: $\theta_{new} = \theta_{old} - (d'/d'')$
 - If $(d'/d'') < 0 \rightarrow$ Theta increases
 - If $(d'/d'') > 0 \rightarrow$ Theta decreases
 - If $(d'/d'') = 0$ then at the peak!
- **2nd derivative d'' also tells you how good of a peak you have**
 - Need to know where your best Theta is (at $d' = 0$), as well as how precise it is (from d'')
 - If the function is flat, d'' will be smallish
 - **Want large d'' because $1/\text{SQRT}(d'') = \text{Theta's SE}$**



Theta Estimation: ML with Help

- ML is used to come up with the most likely Theta given observed item response pattern and the item parameters...
 - ...but can't estimate Theta if item responses are all 0's or all 1's!
- **Prior distributions** to the rescue (Bayes, *cough cough*)!
 - Multiply likelihood function for Theta with prior distribution (usually we assume multivariate normal)
 - Contribution of the prior is minimized with increasing items, but allows us to get Thetas for all 0 or all 1 response patterns
- Note the implication of this for what Theta really is for each person:
 - **THETA IS A DISTRIBUTION, NOT A VALUE!**
 - Although we can find the most likely value, we can't ignore its probabilistic nature or how good of an estimate it is (how peaked the LL function is)
 - SE is constant for CFA factor scores, but SE is NOT constant for IRT Thetas
 - **THIS IS WHY YOU SHOULD AVOID OUTPUTTING THETAS**

Theta Estimation: 3 Methods

- **ML:** Maximum Likelihood Scoring
 - Uses just item parameters to come up with Thetas
 - Can't estimate Theta if none or all are answered correctly
- **MAP:** Maximum a Posteriori Scoring
 - Combine ML estimate with a continuous normal prior distribution
 - Theta estimate is mode of combined posterior distribution
 - Theta will be regressed toward mean if reliability is low
 - Is used in Mplus WLSMV
- **EAP:** Expected A Posteriori Scoring
 - Combine ML estimate with a 'rectangled' normal prior distribution
 - Theta estimate is mean of combined posterior distribution
 - Is used in Mplus ML for CFA or IRT/IFA (and is best version)

Model Comparisons in IRT: Relative Model Fit via $-2\Delta LL$ Tests

- **Nested models** can be compared with the same $-2\Delta LL$ tests we used in CFA → without the “robust” part of ML, so they get simpler (scale factor=1)
 - Step 1: Calculate $-2\Delta LL = -2(LL_{\text{fewer}} - LL_{\text{more}})$
 - Step 2: Calculate $\Delta df = df_{\text{more}} - df_{\text{fewer}}$ (given as “# free parms”)
 - Compare $-2\Delta LL$ with $df = \Delta df$ to χ^2 critical values (or excel CHIDIST)
 - Add 1 parameter? $-2\Delta LL(1) > 3.84$, add 2: $-2\Delta LL(2) > 5.99\dots$
- If **adding** parameters, model fit can be **better** or **not better**
- If **removing** parameters, model fit can be **worse** or **not worse**
- **AIC and BIC** values (from $-2LL$) can be used to compare non-nested models (given same sample of people and items), **smaller is better**
- No trustable absolute global fit measures available via full information ML for IRT → categorical data can't be summarized by just a covariance matrix

Local Model Fit Using ML IRT

- IRT programs (but not Mplus) provide “item fit” and “person fit” statistics
 - Item fit: Predicted vs. observed ICCs—how well do they match?
Or via inferential tests (Bock Chi-Square Index or BILOG version)
 - Person fit “Z” based on predicted vs. observed response patterns
 - Many require the use of outputted thetas, which makes them problematic
- **Using ML in Mplus:** Local item fit available with **TECH10** output
 - **Univariate item fits:** How well did the model reproduce the observed response proportions? (Not likely to have problems here)
 - **Bivariate item fits:** Contingency tables for pairs of responses → Get χ^2 value for each pair of items for their remaining dependency after controlling for Theta(s)
- Bivariate item fit is the basis of the newest absolute fit statistics discussed by Maydeu-Olivares (2015): M_2 (analogous to χ^2 test), $RMSEA_2$, and $SRMR_2$
 - Not currently provided in Mplus; not yet standard practice across areas

What Goes Wrong for Absolute (Global) Model Fit using ML...

- **ML is a full-information estimator, and it is now trying to reproduce the observed item response pattern, not a covariance matrix!**
- Model DF is based on FULL response pattern: $\#responses^{\#items}$
 - $DF = \# \text{ possible observed patterns} - \# \text{ parameters} - 1$
 - So, for an example of 24 binary items in a 1-PL Model:
 - $\text{Max DF} = 2^{24} - \#a_i - \#b_i - 1 = 16,777,216 - 1 - 24 - 1 = \mathbf{16,777,190!}$
 - If some cells aren't observed (Mplus deletes them from the χ^2 calculation), then DF may be $<$ Max DF, and thus χ^2 won't have the right distribution
- Pearson χ^2 based on classic formula: $(\text{observed} - \text{expected})^2 / \text{expected}$
 - Good luck finding enough people to fill up all possible patterns!
 - Other χ^2 given in output is "Likelihood Ratio" χ^2 , calculated differently
 - Linda Muthén suggests "if these don't match, they should not be used"
 - **χ^2 generally won't work well for assessing absolute global fit in IRT**

Summary: ML for IRT Models

- Full-information Marginal ML with numeric integration for IRT models tries to find the item parameters that are most likely *given the observed item response pattern* → IFA or IRT parameters on logit or probit scales
- Because of the integration (rectangling Theta) required at each step of estimation, it may not be feasible to use ML for IRT models in small samples or for many factors at once (too many rectangles simultaneously)
 - This where MCMC estimation can be a more practical strategy
- IRT using ML does not have agreed-upon measures of absolute global fit
 - Categorical item responses cannot be summarized by just a covariance matrix anymore, but by all possible response patterns instead
 - Usually there are not enough people to fill up all possible response patterns, so there's no valid basis for an absolute fit comparison
 - Nested models (on same items) can still have relative fit compared via $-2\Delta LL$
- There is another game in town for IRT in Mplus, however...

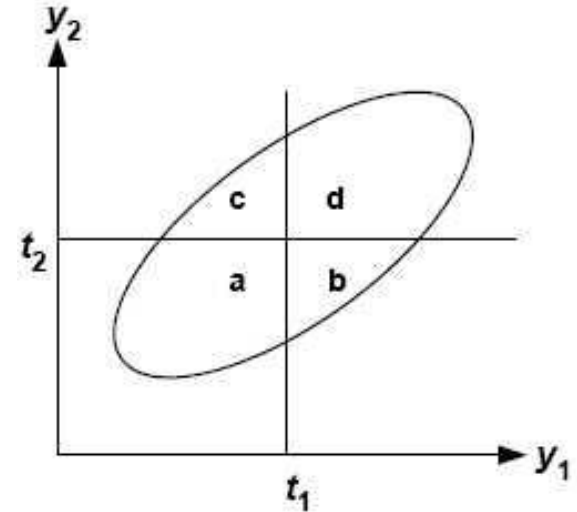
Another Alternative: WLSMV

- **WLSMV**: “Weighted Least Square parameter estimates use a diagonal weight matrix and a Mean- and Variance-adjusted χ^2 test”
 - Called “diagonally-weighted least squares” by non-Mplus people
- Translation: **WLSMV** is a **limited-information** estimator that uses a different summary of responses instead → a **“linked” covariance matrix**
- Fit can then be assessed in regular CFA ways, because what is trying to be reproduced is again a **type of covariance matrix**
 - Instead of the *full item response pattern* (as in ML)
 - We can then get the typical measures of absolute fit as in CFA
- Normally CFA uses the *observed* covariance matrix of the items...
 - But correlations among binary items will be less than 1 any time p differs from .5, so the covariances will be restricted as well...
 - What if we could fit a covariance matrix on the logit or probits instead???

WLSMV Estimation

Data	$y_2 = 0$	$y_2 = 1$
$y_1 = 0$	a	c
$y_1 = 1$	b	d

Use the observed cell proportions as the area under the curve of each section of the bivariate normal distribution to determine what the correlation would be →



- WLSMV first estimates correlation matrix of underlying item responses (probit scale)
 - For binary responses → “tetrachoric correlation matrix”
 - For ordinal (polytomous) responses → “polychoric correlation matrix”
- The model then tries to find item parameters to predict this new correlation matrix
- The diagonal W “weight” part then tries to emphasize reproducing latent variable correlations that are relatively well-determined more than those that aren’t
 - The full weight matrix is of order $z \times z$, where z is number of elements to estimate
 - The “diagonal” part means it only uses the *preciseness of the estimates themselves*, not the covariances among the “preciseness-es” (much easier, and not a whole lot of info lost)
- The “MV” corrects the χ^2 test for bias arising from this weighting process

More about WLSMV Estimation

- Works much faster than ML when you have small samples or many factors to estimate (because no rectangling is required)
- Does assume missing data are **missing completely at random**, whereas ML assumes only *missing at random* (conditionally random)
- Because a saturated covariance matrix is used as the input data, we get absolute fit indices as in CFA, but they should be interpreted with caution
 - Fewer people → less well-estimated “saturated” matrix to start from
 - More skewness, fewer categories → easier to get falsely good model fit
- Model parameters will be on the **probit scale** instead of logit scale
 - Unlike full-information ML, in which you can choose logit or probit, though
- Two item variance scalings in Mplus via the **PARAMETERIZATION** option on the **ANALYSIS** command, where a 1 is needed for identification
 - **“Delta”** (default): variance (Y^*) = factor + error = 1 = “marginal parameterization”
 - **“Theta”**: error variance = 1 instead = “conditional parameterization”
 - **WE WILL USE THIS ONE TO HELP SIMPLIFY IRT CONVERSIONS**

Model Comparisons with WLSMV using DIFFTEST in Mplus

- Not the same process! Model DF is NOT calculated in usual way, and model fit is not compared in the usual way
 - Absolute χ^2 model fit values are meaningless—they are not comparable!
 - Difference in model χ^2 are not distributed as χ^2
- Here's how you do nested model comparisons in WLSMV:
 - Step 1: Estimate model with *more* parameters, adding this command:
 - `SAVEDATA: DIFFTEST=more.dat;` → Saves needed derivatives to file
 - Step 2: Estimate model with *fewer* parameters, adding this command:
 - `ANALYSIS: DIFFTEST=fewer.dat;` → Uses those derivatives to do $\Delta\chi^2$ test
 - Step 2 model output will have a new χ^2 difference test in it that you can use, with df difference to compare to a χ^2 distribution

Assessing Local Model Fit

- **The need to check local model fit is the same in IRT/IFA as in CFA**
- **Using ML:** Local item fit in Mplus available with **TECH10** option
 - **Univariate item fits:** How well did the model reproduce the observed response frequencies? (Not likely to have problems here if each item has own location)
 - **Bivariate item fits:** Contingency tables for pairs of responses → Get χ^2 value for each pair of items for their remaining dependency after controlling for Theta(s)
- **Under WLSMV:** Residual correlation matrix (i.e., model–data discrepancy) via the RESIDUAL option on OUTPUT statement (just like in CFA)
 - Predicted and residual (left-over) item correlations given in *correlation* metric
 - Look for large residual correlations in absolute value (but no significance tests)
 - Will be MUCH easier to do for many items than bivariate fit in ML

Error Covariances in IRT/IFA

- Additional relationships between items can be included:
 - Via **error covariances** (the same as in CFA) when using **WLSMV** because the model is being estimated on the tetrachoric/polychoric correlation matrix (so the error of the underlying probit can covary, even if item error or total variances = 1 for identification)
 - Error covariances are not allowed when using maximum likelihood
 - Instead, you can specify "**method factors**" (in WLSMV or ML), also known as a "bifactor model" (which can also be used in CFA models)
- Here is an example using WLSMV to demonstrate both ways:

```
! Primary factor/theta
Trait BY item1-item5*;
[Trait@0]; Trait@1;
! Error covariance
item2 WITH item3*;
```

```
! Primary factor/theta
Trait BY item1-item5*;
[Trait@0]; Trait@1;
! Uncorrelated factor to
  create error covariance
ErrFact BY item2@1 item3@1;
[ErrFact@0]; ErrFact*;
ErrFact WITH Trait@0;
```


Error Covariances in IRT/IFA

```
! Primary factor/theta
Trait BY item1-item5*;
[Trait@0]; Trait@1;
! Uncorrelated factor to
  create error covariance
ErrFact BY item2@1 item3@1;
[ErrFact@0]; ErrFact*;
ErrFact WITH Trait@0;
```

For models with many method factors, add the **ANALYSIS:** option **MODEL=NOCOVARIANCES** to make all factors **uncorrelated** by default (instead of correlated by default as usual)

TRAIT	BY				
ITEM1		0.994	0.078	12.724	0.000
ITEM2		2.138	0.148	14.459	0.000
ITEM3		1.823	0.125	14.527	0.000
ITEM4		1.106	0.090	12.311	0.000
ITEM5		0.232	0.045	5.200	0.000
ERRFACT	BY				
ITEM2		1.000	0.000	999.000	999.000
ITEM3		1.000	0.000	999.000	999.000
ERRFACT	WITH				
TRAIT		0.000	0.000	999.000	999.000
Variances					
TRAIT		1.000	0.000	999.000	999.000
ERRFACT		1.996	0.314	6.357	0.000

To create a negative error covariance, fix the ErrFact loadings to 1 and -1 instead.

The variance of ErrFact is the positive error covariance between items 2 and 3.

IRT/IFA Model Estimation: Summary

- Full-information Marginal ML estimation with numeric integration provides:
 - '**Best guess**' at to the value of each item parameter (and person theta if you ask for it)
 - **SE** that conveys the uncertainty of that prediction
- The '**best guesses**' for the model parameters do not depend on the sample:
 - Item estimates do not depend on the particular individuals that provided responses
 - Person estimates do not depend on the particular items that were administered
 - Thus, model parameter estimates are sample-invariant
- The **SEs** for those model parameters DO depend on the sample
 - Item parameters will be estimated less precisely **where** there are fewer individuals
 - Person parameters will be estimated less precisely **where** there are fewer items
- **WLSMV** in Mplus is a limited-estimation approach for IFA or IRT models
 - Uses an estimated tetrachoric correlation matrix as input for the factor analysis
 - Works better for many factors than ML (but can be less trustworthy overall)