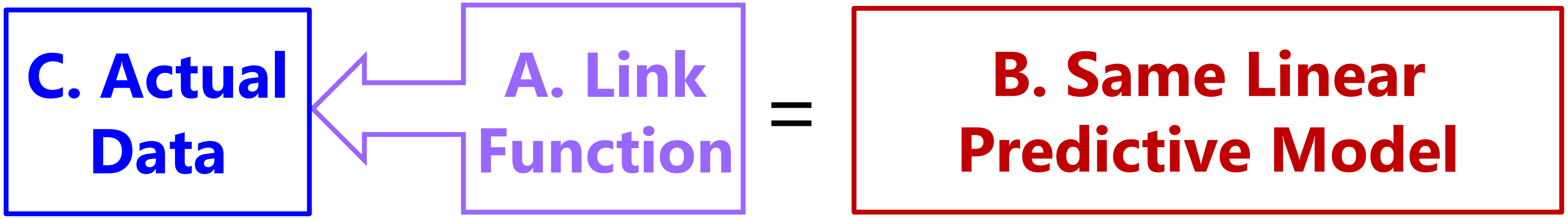
Multivariate Models (via Univariate Software)

- Topics:
  - Taxonomy of multivariate dependency: balanced vs. unbalanced
  - Multivariate models: in univariate vs. "truly multivariate" software
  - **R** matrix choices for residual variance and covariance
  - Adding dependency indirectly via random intercept variance
  - Fixed effects parameterization choices:
    direct slopes vs. differences in slopes

3 Parts of Generalized Linear Models



C. Actual Data ← **A. Link Function** = **B. Same Linear Predictive Model**

A. **Link Function**: Transformation of conditional mean used to keep *predicted outcomes* within bounds of the possible values

B. **Same Linear Predictor**: How the model fixed effects linearly predict the *link-transformed* conditional mean of the outcome
  - Btw, I call fixed effects the "**model for the means**" more generally

C. **Conditional Distribution**: How the outcome residuals should be distributed given the possible values of the outcome

- **Now we need to consider how the model needs to adapt when residuals are correlated → capture "dependency"**
  - Btw, I call this idea the "**model for the variance**" more generally

Types of Multivariate Dependency

- **Dependency** (*aka*, "**residual correlation**") arises whenever **multiple outcomes are collected from the same sampling unit**, for example:
  - A single outcome across repeated occasions, under multiple conditions, or multiple measures from the same person ("repeated measures" data)
  - Multiple persons from the same pair ("dyadic" data)
  - Multiple persons from the same group ("clustered" data)

- A **key distinction** in guiding modeling options is whether the multiple outcomes are "**balanced**"—**structured the same** for every sampling unit
  - **Balanced:** all persons have the same *potential* occasions, conditions, or measures (where *potential* allows missing values) from a common set
  - **Unbalanced:** no common set (e.g., observed occasions differ across persons; number of persons within a cluster differs across clusters)

- We will not cover **unbalanced** outcomes in this class—they will be covered instead in classes focused on **multilevel models** (*aka*, mixed-effects models, PSQF 6271, PSQF 6272) involving random intercepts and slopes

Estimating (Balanced) Multivariate Models

- Multivariate models can be estimated by **"tricking" univariate software** for general(*ized*) linear models (e.g., SAS MIXED, STATA MIXED, R GLS) **if each variable is either a predictor OR an outcome**, but not both, such as when:
  - You want to examine **mean differences** across the outcomes (e.g., over time or across conditions, as in traditional Repeated Measures ANOVA)
  - You want to test **differences in the slopes of predictors** across outcomes (as in traditional multivariate analysis of variance, or MANOVA)
  - A **big downside** for general*ized* models is that some parameters are **forced equal** across outcomes (e.g., categorical submodel intercepts, over-dispersion scale factors)

- Multivariate models will need to be estimated in **"truly" multivariate software** instead (i.e., as **path analysis models** or structural equation models) if some **variables are both predictors and outcomes**, such as in mediation
  - e.g., mediation = X → M → Y, in which M is both an outcome of X and a predictor of Y at once, which involves regressions instead of covariances between outcomes
  - Path analysis *can* be more flexible for generalized models (options differ by program)

- For both types of analyses, we will use **likelihood estimation instead of ordinary least squares**, so that cases with missing outcomes are not removed from the model (for what happens with missing predictors, stay tuned)

Back to General Linear Models…

- Regardless of software, **relations among outcomes from the same sampling unit** can be specified in one of two ways:
  - **Directly** → is only possible for models with normal residuals (GLM)
    - Linear predictor will only include fixed effects, like usual, because residual dependency is captured directly via residual covariances
  - **Indirectly** → is the only option using true likelihood estimation using non-normal conditional distributions (i.e., general*ized* linear models)
    - Add (latent) random intercept to the linear predictor to capture residual dependency (so the usual conditional distributions can still be used)

- To understand the difference, we first need to describe models for <u>independent observations</u> using new vocabulary—fun with matrices!
  - Let's start with this general linear model: $\boldsymbol{y_i} = \boldsymbol{\beta_0} + \boldsymbol{\beta_1}(\boldsymbol{x_i}) + \boldsymbol{e_i}$
    - In this "scalar" notation, the assumed independence is hidden…
    - What follows is the "**direct**" way of including relations among outcomes (we will see the "**indirect**" way at work shortly, stay tuned…)

Example: $N = 6$ persons, $n = 1$ outcome

- GLM in scalar form: $\boldsymbol{y_i} = \boldsymbol{\beta_0} + \boldsymbol{\beta_1}(\boldsymbol{x_i}) + \boldsymbol{e_i}$ with binary $\boldsymbol{x_i}$ = 0 or 1
- GLM using matrices with $k = 2$ fixed effects: $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{E}$

$$\begin{array}{cccccc} \mathbf{Y} & & \mathbf{X} & \boldsymbol{\beta} & & \mathbf{E} \\ \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} & = & \begin{bmatrix} 1 & x_1 \\ 1 & x_2 \\ 1 & x_3 \\ 1 & x_4 \\ 1 & x_5 \\ 1 & x_6 \end{bmatrix} & \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} & + & \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix} \end{array}$$

$$= \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \end{bmatrix} = \begin{bmatrix} \beta_0 1 + \beta_1 x_1 \\ \beta_0 1 + \beta_1 x_2 \\ \beta_0 1 + \beta_1 x_3 \\ \beta_0 1 + \beta_1 x_4 \\ \beta_0 1 + \beta_1 x_5 \\ \beta_0 1 + \beta_1 x_6 \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix}$$

$\mathbf{Y} = N * 1$ outcome vector
$\mathbf{X} = N * k$ "design" matrix for predictors that have fixed effects
$\boldsymbol{\beta} = k * 1$ fixed effects vector
$\mathbf{E} = N * 1$ residual vector

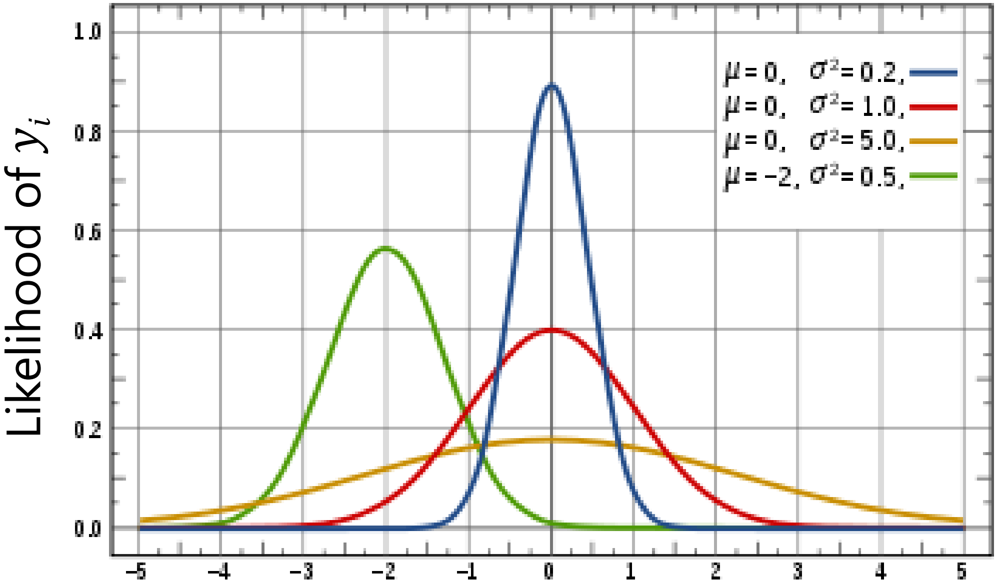
where (by default) $\boldsymbol{e_i} \sim \mathrm{Normal}(0, \boldsymbol{\sigma_e^2})$

Across all 6 persons, **variance–covariance "R"** matrix of the $\boldsymbol{e_i}$ residuals is of type "**independent**" (or "**VC**" = variance components in SAS)

$$\begin{bmatrix} \sigma_e^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_e^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_e^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_e^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_e^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_e^2 \end{bmatrix}$$

Off-diagonal 0 values → independent residuals
$\boldsymbol{\sigma_e^2}$ → all persons share common residual variance

Review: Univariate Normal PDF



Univariate Normal PDF:

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

Sum over persons of log of $f(y_i)$ = Model Log-Likelihood → Model Fit

- This PDF tells us how **likely** (i.e., **tall**) any value of y_i is given two things:
  - Conditional mean $\hat{y}_i$ (labeled as μ in left plot)
  - Residual variance σ_e^2
- We can see this work using the NORMDIST function in excel!
  - Easiest for **empty** model:

$$y_i = \beta_0 + e_i$$

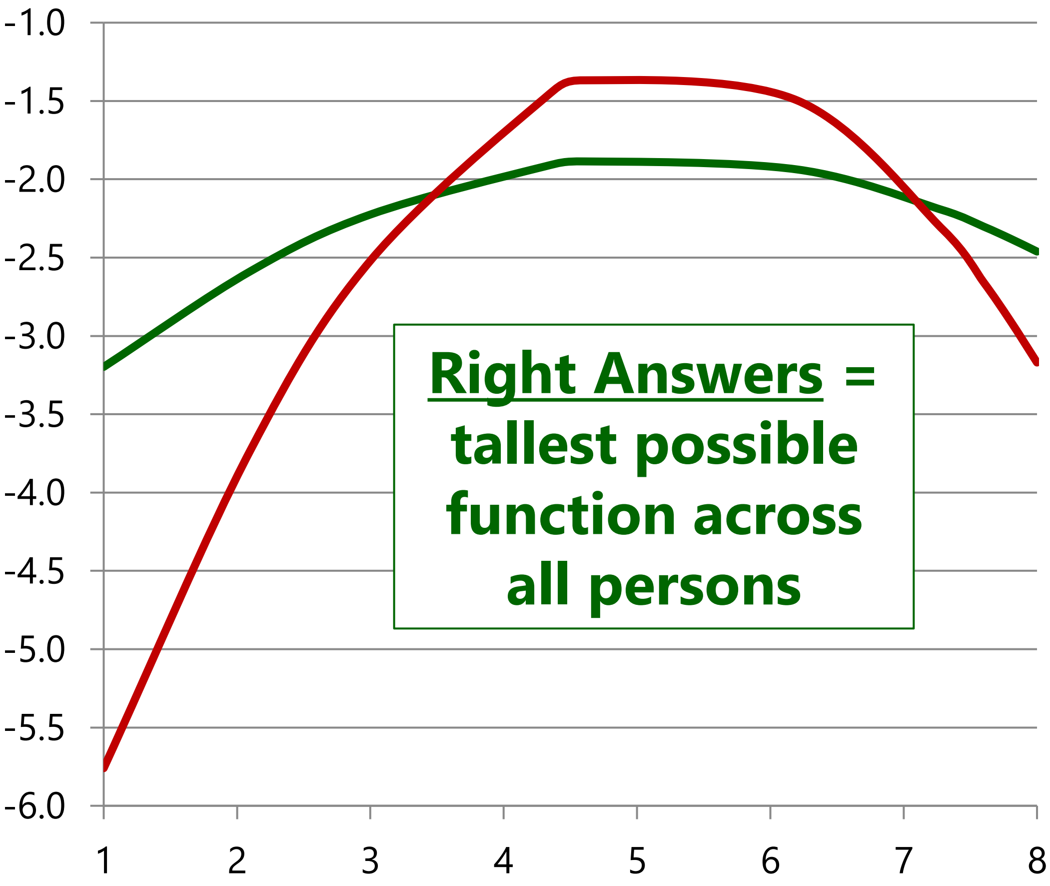
- We can check our math via software using ML!

Image borrowed from: https://en.wikipedia.org/wiki/Normal_distribution

Univariate ML via Excel "NORMDIST"

Key idea: Normal Distribution formula → data height

	Right	Wrong
Mean	5.19	5.24
Variance	6.56	2.00
Outcome	**Log(Height)**	**Log(Height)**
1.0	-3.20	-5.76
2.1	-2.59	-3.73
3.0	-2.22	-2.52
4.3	-1.92	-1.49
4.6	-1.89	-1.37
6.2	-1.94	-1.50
7.3	-2.20	-2.33
7.6	-2.30	-2.66
7.8	-2.38	-2.90
8.0	-2.46	-3.17
SUM = Model LL = taller is better		
	-23.09	-27.42



Right Answers = tallest possible function across all persons

Tricking Univariate into Multivariate…

- What if the 6 observations were **2 outcomes (e.g., occasion T1 and T2 here) each from 3 persons** instead? We need a new model that builds in some covariance for residuals from the same person *(and maybe a different amount of residual variance for each outcome as well, stay tuned)*
- If the outcomes are in separate columns, then **to use univariate software, we need to "stack" the separate outcomes** into a single column (i.e., go from a "wide" to a "long" data structure), like this:

"Wide" Structure:
1 row per person

Person i	**yT1**	**yT2**
1	y_{11}	y_{12}
2	y_{21}	y_{22}
3	y_{31}	y_{32}

"**Reshape**" wide into long data →

"Long" Structure:
1 row per outcome

Person i	**Time t**	x_{it}	y_{it}
1	1	0	y_{11}
1	2	1	y_{12}
2	1	0	y_{21}
2	2	1	y_{22}
3	1	0	y_{31}
3	2	1	y_{32}

In the "long" structure we add a time column to index which occasion is included in each row

We also need to make a centered version of the time index, x_{it} , to include as a predictor

Multivariate: $N = 3$ persons, $n = 2$ outcomes

- Multiv. GLM: $y_{it} = \beta_{00} + \beta_{01}(x_{it}) + e_{it}$ with binary x_i = 0 or 1 requires a **per-person model** in matrices: $\mathbf{Y_i} = \mathbf{X_i}\boldsymbol{\beta} + \mathbf{E_i}$

$$\begin{array}{cccccc} \mathbf{Y_i} & & \mathbf{X_i} & \boldsymbol{\beta} & & \mathbf{E_i} \\ \begin{bmatrix} y_{i1} \\ y_{i2} \end{bmatrix} & = & \begin{bmatrix} 1 & x_{i1} \\ 1 & x_{i2} \end{bmatrix} & \begin{bmatrix} \beta_{00} \\ \beta_{01} \end{bmatrix} & + & \begin{bmatrix} e_{i1} \\ e_{i2} \end{bmatrix} \end{array}$$

where $e_i \sim \mathrm{MVN}(0, R_i)$, and **2*2** $\mathbf{R_i}$ is chosen to predict **3 unique terms**:

MVN = multivariate normal

$$\mathbf{R_i} = \begin{bmatrix} \sigma_1^2 & \sigma_{1,2} \\ \sigma_{1,2} & \sigma_2^2 \end{bmatrix}$$

(same matrix pattern would be repeated including whichever **complete rows** each person has)

Across all 3 persons, the **combined residual variance–covariance** matrix $\mathbf{R}$ has a "**block diagonal**" structure with three 2*2 per-person pockets of variances (σ^2) and covariance (σ):

$$\begin{bmatrix} \sigma_1^2 & \sigma_{1,2} & 0 & 0 & 0 & 0 \\ \sigma_{1,2} & \sigma_2^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_1^2 & \sigma_{1,2} & 0 & 0 \\ 0 & 0 & \sigma_{1,2} & \sigma_2^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_1^2 & \sigma_{1,2} \\ 0 & 0 & 0 & 0 & \sigma_{1,2} & \sigma_2^2 \end{bmatrix}$$

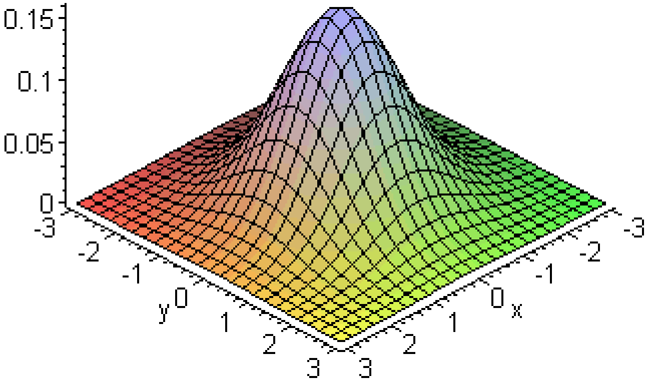
Off-block-diagonal 0 values → no residual covariances *across persons*

Same symbols → all persons share common residual variances and covariance

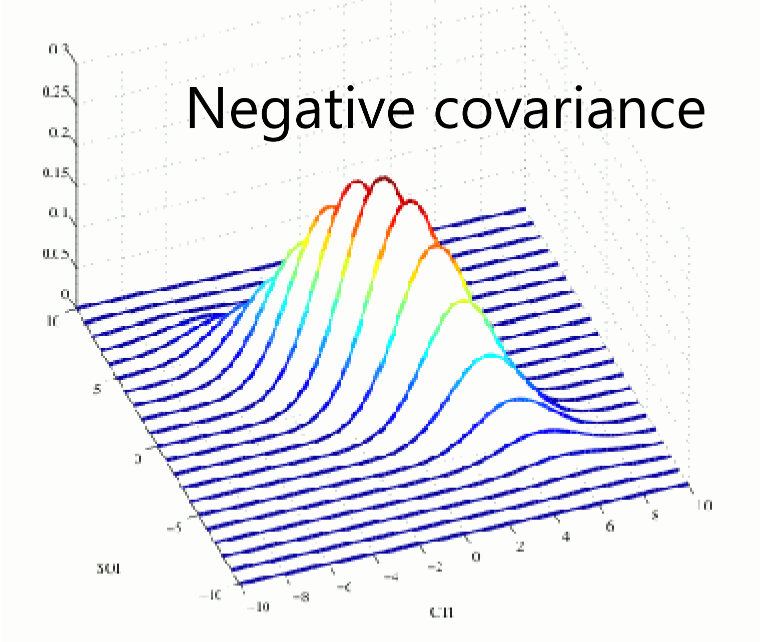
Welcome to Multivariate Normal!

- Same principle as univariate normal, but **LL is calculated for each person's SET of outcomes** (then LL is summed over persons)
- Model parameters to be found include parameters that predict **EACH outcome's residual variance and all residual covariances**
- So each outcome's likelihood height has its own dimension, but the joint shape depends on the covariance between outcomes:

No covariance



Negative covariance



Multivariate Normal for $\boldsymbol{Y}_i$ (height for all n outcomes for person i)

Univariate Normal PDF: $f(y_i) = \left(2\pi\sigma_e^2\right)^{-1/2} * \exp\left[-\frac{1}{2} * \left(y_i - y_i\right)\left(\sigma_e^2\right)^{-1}\left(y_i - y_i\right)\right]$

Multivariate Normal PDF: $f(\mathbf{Y}_i) = \left(2\pi\right)^{-n/2} * \left|\mathbf{R}_i\right|^{-1/2} * \exp\left[-\frac{1}{2} * \left(\mathbf{Y}_i - \mathbf{X}_i\boldsymbol{\beta}\right)^T \left(\mathbf{R}_i\right)^{-1}\left(\mathbf{Y}_i - \mathbf{X}_i\boldsymbol{\beta}\right)\right]$

- In MVN, the single $\hat{y}_i$ per person is replaced with $\mathbf{X}_i\boldsymbol{\beta}$ from the model
- The model also gives $\mathbf{R}_i$ → the predicted residual variance–covariance matrix across outcomes (built using the same terms for each person here)
- Uses $|\mathbf{R}_i|$ = determinant of $\mathbf{R}_i$ = summary of *non-redundant* info
  - Reflects sum of variances across outcomes controlling for covariances
- $(\mathbf{R}_i)^{-1}$ → matrix inverse → like dividing (so can't be 0 or negative)
  - $(\mathbf{R}_i)^{-1}$ must be "positive definite", which in practice means no 0 residual variances and no out-of-bounds residual correlations between outcomes
  - Otherwise, programs use "generalized inverse" → questionable results

What about Missing Data? Bad news…

- In univariate software using likelihood estimation: **each <u>row</u> must be complete** (otherwise, software drops it from model)
  - So whole people are not removed if they are missing some outcomes
  - (Untestable) assumption is then "**missing at random**" which means random after taking into account the person's other rows—said differently, the shape of each person's likelihood function would stay the same given complete or incomplete cases
- Whole people will be removed if they are **missing a person-level predictor**, which then implies the predictor is **missing completely at random** (what it sounds like)
  - Two solutions: **multiple imputation**, or **treat the predictor as an outcome** in a "truly" multivariate model—both require making distributional assumptions for the predictor with missing values… stay tuned

"Long" Structure:
1 row per <u>outcome</u>

Person i	**Time t**	x_{it}	y_{it}
1	1	0	y_{11}
1	2	1	y_{12}
2	1	0	y_{21}
2	2	1	y_{22}
3	1	0	y_{31}
3	2	1	y_{32}

For now, we will pre-select our sample for complete observations to keep the sample (and all model fit statistics) comparable across models

Multivariate: $N = 2$ persons, 3 outcomes

- Multiv. GLM: $\boldsymbol{y_{it}} = \boldsymbol{\beta_{00}} + \boldsymbol{\beta_{01}}(\boldsymbol{x1_{it}}) + \boldsymbol{\beta_{02}}(\boldsymbol{x2_{it}}) + \boldsymbol{e_{it}}$ with 2 binary predictors **per-person model**: $\mathbf{Y_i} = \mathbf{X_i}\boldsymbol{\beta} + \mathbf{E_i}$

$$\begin{array}{cccccc} \mathbf{Y_i} & & \mathbf{X_i} & \boldsymbol{\beta} & & \mathbf{E_i} \\ \begin{bmatrix} y_{t1} \\ y_{t2} \\ y_{t3} \end{bmatrix} & = & \begin{bmatrix} 1 & x1_{t1} & x2_{t1} \\ 1 & x1_{t2} & x2_{t2} \\ 1 & x1_{t3} & x2_{t3} \end{bmatrix} & \begin{bmatrix} \beta_{00} \\ \beta_{01} \\ \beta_{02} \end{bmatrix} & + & \begin{bmatrix} e_{t1} \\ e_{t2} \\ e_{t3} \end{bmatrix} \end{array}$$

where $\boldsymbol{e_{it}} \sim \mathrm{MVN}(0, \mathbf{R})$, and **3*3** $\mathbf{R_i}$ is chosen to predict **6 unique terms**:

$$\mathbf{R_i} = \begin{bmatrix} \sigma_1^2 & \sigma_{1,2} & \sigma_{1,3} \\ \sigma_{1,2} & \sigma_2^2 & \sigma_{2,3} \\ \sigma_{1,3} & \sigma_{2,3} & \sigma_3^2 \end{bmatrix}$$

(same matrix pattern would be repeated including whichever **complete rows** each person has)

Across both persons, the **combined residual variance–covariance** matrix $\mathbf{R}$ has a "**block diagonal**" structure with two 3*3 per-person pockets of variances (σ^2) and covariances (σ):

$$\begin{bmatrix} \sigma_1^2 & \sigma_{1,2} & \sigma_{1,3} & 0 & 0 & 0 \\ \sigma_{1,2} & \sigma_2^2 & \sigma_{2,3} & 0 & 0 & 0 \\ \sigma_{1,3} & \sigma_{2,3} & \sigma_3^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_1^2 & \sigma_{1,2} & \sigma_{1,3} \\ 0 & 0 & 0 & \sigma_{1,2} & \sigma_2^2 & \sigma_{2,3} \\ 0 & 0 & 0 & \sigma_{1,3} & \sigma_{2,3} & \sigma_3^2 \end{bmatrix}$$

Off-block-diagonal 0 values → no residual covariances *across persons*
Same symbols → all persons share common residual variances and covariances

What should the *R* Matrix Look Like?

- **Goal: re-create all variances and covariances in $\mathbf{R_i}$ (to form big $\mathbf{R}$ for all)**
  - The "**direct**" way uses only different **R** patterns ("**R**-side" models, as opposed to "**G**-side" models with random intercepts, stay tuned)
- **Next are 3 "direct" choices relevant for unordered multiple outcomes** (btw, there are way more choices for outcomes ordered in time or space)
  - SAS MIXED: `REPEATED DVindex /TYPE=?? SUBJECT=PersonID R RCORR;`
  - SAS GLIMMIX: `RANDOM DVindex /TYPE=?? SUBJECT=PersonID RESIDUAL;`
  - Stata MIXED: Goes into option `residuals(??, t(DVindex))`
    - Not possible in STATA GLM or MEGLM (as far as I know)
  - R GLS (within LME package): `gls(correlation= , weights= )`
- **The 3 choices for *R* patterns we will use differ in 2 respects**:
  - Are the residual variances ($\boldsymbol{\sigma_e^2}$) the same across outcomes?
    - **If so**, then are residual **covariances** ($\boldsymbol{\sigma}$) are also the same across outcome pairs (remember: covariance is unstandardized correlation)?
    - **If not**, might residual **correlations** ($\boldsymbol{r}$) still be the same across outcome pairs (because covariances will differ whenever variances differ)?

Option 1 for **R** Patterns: The Answer Key

- **Option 1 is an "unstructured" R matrix:** all variances and covariances estimated separately (i.e., **non-constant, outcome-specific dependency**)
  - **UN** is a *description*, not a prediction, so it will fit best (i.e., as tallest LL)
  - Requires parameters = $\frac{n*(n+1)}{2}$ for n outcomes (so is hard to estimate past 5ish outcomes in smaller samples)
  - Left: $\mathbf{R_{cov}}$ is a covariance matrix; $\mathbf{R_{cor}}$ is a correlation matrix

$$\mathbf{R}_{\text{cov}} = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_2^2 & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & \sigma_3^2 \end{bmatrix} \quad \mathbf{R}_{\text{cor}} = \begin{bmatrix} 1 & \frac{\sigma_{12}}{\sigma_1\sigma_2} & \frac{\sigma_{13}}{\sigma_1\sigma_3} \\ \frac{\sigma_{12}}{\sigma_1\sigma_2} & 1 & \frac{\sigma_{23}}{\sigma_2\sigma_3} \\ \frac{\sigma_{13}}{\sigma_1\sigma_3} & \frac{\sigma_{23}}{\sigma_2\sigma_3} & 1 \end{bmatrix}$$

- Btw, **an unstructured R matrix** is also used in a "multivariate (MANOVA) model" or the "multivariate approach" to repeated measures (RM) ANOVA
  - Why the difference? When people say "RM ANOVA" or "MANOVA" they are often referring to the use of **OLS** instead of **REML (or ML) estimation**
    - **Same model**, but **OLS uses only complete outcomes per person** (so outcomes are assumed *missing completely at random*), but likelihood estimation uses all possible outcomes (assumed *missing at random* instead)
    - Btw, generalized least squares (GLS) is how fixed effects are computed after using REML or ML to search for the most likely variance model parameters

Two More Choices for Patterns of R

- **Option 2 is a "compound symmetry heterogeneous" R** matrix: separate variances, but covariances are created using a common correlation ("CSH"):
  - Uses **$n + 1$ total parameters**; given different outcome residual variances, all outcome residuals are correlated to the same extent (i.e., constant dependency)

$$\mathbf{R}_{cov} = \begin{bmatrix} \sigma_1^2 & CSH\sigma_1\sigma_2 & CSH\sigma_1\sigma_3 \\ CSH\sigma_2\sigma_1 & \sigma_2^2 & CSH\sigma_2\sigma_3 \\ CSH\sigma_3\sigma_1 & CSH\sigma_3\sigma_2 & \sigma_3^2 \end{bmatrix} \quad \mathbf{R}_{cor} = \begin{bmatrix} 1 & CSH & CSH \\ CSH & 1 & CSH \\ CSH & CSH & 1 \end{bmatrix}$$

CSH is still not available within STATA MIXED ☹

- **Option 3 is a "compound symmetry" R** matrix: equal residual variances and equal residual covariances (so only **2 parameters** no matter how many outcomes)
  - All **dependency is constant across outcomes** and is caused by person mean diffs: the "CS" parameter
  - Also known as the "univariate" approach to RM ANOVA (if using least squares) and equal to "random intercept only" model (the "indirect" way of capturing dependency)

$$\mathbf{R}_{cov} = \begin{bmatrix} CS+\sigma_e^2 & CS & CS \\ CS & CS+\sigma_e^2 & CS \\ CS & CS & CS+\sigma_e^2 \end{bmatrix}$$

$$\mathbf{R}_{cor} = \begin{bmatrix} 1 & \frac{CS}{CS+\sigma_e^2} & \frac{CS}{CS+\sigma_e^2} \\ \frac{CS}{CS+\sigma_e^2} & 1 & \frac{CS}{CS+\sigma_e^2} \\ \frac{CS}{CS+\sigma_e^2} & \frac{CS}{CS+\sigma_e^2} & 1 \end{bmatrix}$$

How to Choose among **R** Matrices

- Use **likelihood ratio tests (LRT)**: treat difference in −2LL as regular χ^2 with DF = # parameters different (also, smallest AIC and BIC win)
  - Independent (equal residual variances, no residual covariances) is nested in all others
    - CS fit better than Indep? There's residual covariance (dependency) across outcomes
  - CS is nested in CSH, which are both nested in UN (= recreation of the data)
    - CSH fit better than CS? Then residual variances need to differ by outcome
    - UN fit better than CSH? Then residual correlations need to differ across outcome pairs

- Goal: find a **simpler** model that **fits not worse than UN**
  - UN will always fit best by −2LL because it is trying to create the complete data results (assuming missing at random, so may differ from descriptive statistics for the data)
  - Why not just use UN always? For many outcomes at once it may not always be estimable, and using a simpler model that fits not worse can lead to greater power (because more unnecessary parameters → less power)

- Btw, **R** matrix residual variances and covariances can also be allowed to differ across groups (see Example 4a from 2020 class); test if that helps with LRTs
  - And in univariate models, residual variance can differ by predictors, too!

Assessing Relative Model Fit, In General

- **Model for the Means (linear predictor of fixed effects)** → which fixed effects of predictors are included in the model
  - Because fixed effects of predictors are unbounded, you can always use **univariate or multivariate Wald tests** to see if they contribute to the model (using denominator DF, depending on software availability)
  - Could use LRTs, but only for models estimated with maximum likelihood (not residual maximum likelihood, a better choice for normal residuals)

- **Model for the Variance** → what the pattern of variance and covariance of residuals from the same sampling unit should be
  - **DOES** require assessment of relative model fit **using LRTs**: Because variances cannot be negative, you should not use Wald test *p*-values (i.e., that show up in MIXED output next to the variance estimate)
  - Conditional distributions can only be compared using LRTs (usually with a mixture χ^2) or information criteria (AIC, BIC) if they are nested
    - e.g., Poisson and Negative Binomial differ by "stretchy k"; binomial and beta-binomial differ by "stretchy ϕ"; zero-inflation models add an intercept in another submodel that predicts the logit of being an extra 0

Introducing **G** & **R** → **V** (Person as Unit)

- e.g., For three outcomes per person, a **Compound Symmetry** $\mathbf{R_i}$ matrix for one person would have this pattern:

$$\mathbf{R}_i = \begin{bmatrix} CS+\sigma_e^2 & CS & CS \\ CS & CS+\sigma_e^2 & CS \\ CS & CS & CS+\sigma_e^2 \end{bmatrix}$$

  - Shown below is how the CS pattern can be produced an equivalent way, in which "CS" is the same thing as "random intercept variance" ($\boldsymbol{\tau_{U_0}^2}$) which is distinguished from "residual variance" ($\boldsymbol{\sigma_e^2}$)

Random effect source(s) of person dependency are moved to $\mathbf{G}$ Matrix (dimensions are NOT person-specific)

Remaining within-person variance and covariance is in $\mathbf{R_i}$ matrix (dimensions are person-specific)

Total (Marginal) Residual Variance-Covariance Matrix is called $\mathbf{V_i}$ Matrix (dimensions are person-specific)

$$\mathbf{G} = \begin{bmatrix} \tau_{U_0}^2 \end{bmatrix} \quad \text{"+"} \quad \mathbf{R}_i = \begin{bmatrix} \sigma_e^2 & 0 & 0 \\ 0 & \sigma_e^2 & 0 \\ 0 & 0 & \sigma_e^2 \end{bmatrix} \quad \text{"="} \quad \mathbf{V}_i = \begin{bmatrix} \tau_{U_0}^2+\sigma_e^2 & \tau_{U_0}^2 & \tau_{U_0}^2 \\ \tau_{U_0}^2 & \tau_{U_0}^2+\sigma_e^2 & \tau_{U_0}^2 \\ \tau_{U_0}^2 & \tau_{U_0}^2 & \tau_{U_0}^2+\sigma_e^2 \end{bmatrix}$$

So $CS = \tau_{U_0}^2$, but what is a "U" anyway ????

Univariate vs. Multivariate Models "Indirect" Dependency via U_{i0}

- **Univariate** General Linear Model (used for 1 outcome):

$$y_i = [\beta_0 + \beta_1(x_i) + \cdots] + e_i$$

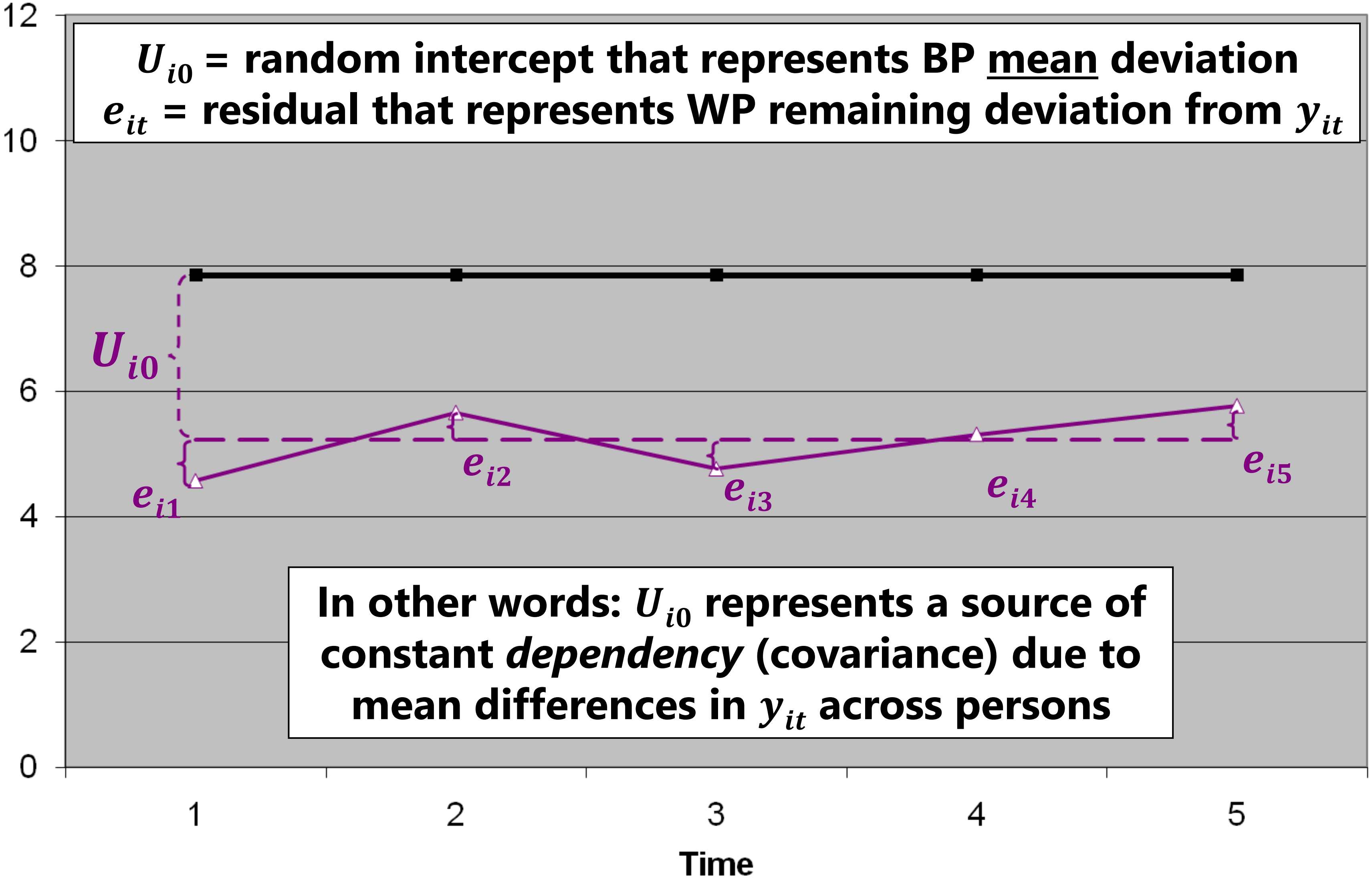
  - β Fixed effects → create conditional mean from predictors
  - e_i = person-specific residual deviation from predicted y_i

- **Multivariate** General Linear Model (for >1 outcomes):

$$y_{it} = [\beta_{00} + \beta_{10}(x_i) + \cdots] + U_{i0} + e_{it}$$

  - β Fixed effects → create conditional mean from predictors
  - U_{i0} = random intercept = person deviation from predicted y_{it} mean
  - e_{it} = outcome-specific residual deviation from predicted by y_{it} + U_{i0}

Example "Error" in a Multivariate GLM



U_{i0} = random intercept that represents BP <u>mean</u> deviation
e_{it} = residual that represents WP remaining deviation from y_{it}

In other words: U_{i0} represents a source of constant *dependency* (covariance) due to mean differences in y_{it} across persons

Indirect Dependency via a Random Intercept

- A scalar **example GLM** with $n = 3$ outcomes (A, B, and C):

$$\boldsymbol{y_{it}} = \boldsymbol{\beta_{00}} + \boldsymbol{\beta_{01}}(\boldsymbol{dvA_{it}}) + \boldsymbol{\beta_{02}}(\boldsymbol{dvB_{it}}) + \boldsymbol{U_{i0}} + \boldsymbol{e_{it}}$$

- In matrix notation, this becomes $\mathbf{Y_i} = \mathbf{X_i}\boldsymbol{\beta} + \mathbf{Z_i}\mathbf{U_i} + \mathbf{E_i}$

$$\begin{array}{ccccccccc} \mathbf{Y_i} & & \mathbf{X_i} & \boldsymbol{\beta} & & \mathbf{Z_i} & \mathbf{U_i} & & \mathbf{E_i} \\ \begin{bmatrix} y_{i1} \\ y_{i2} \\ y_{i3} \end{bmatrix} & = & \begin{bmatrix} 1 & dvA_{i1} & dvB_{i1} \\ 1 & dvA_{i2} & dvB_{i2} \\ 1 & dvA_{i3} & dvB_{i3} \end{bmatrix} & \begin{bmatrix} \beta_{00} \\ \beta_{01} \\ \beta_{02} \end{bmatrix} & + & \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} & [U_{i0}] & + & \begin{bmatrix} e_{i1} \\ e_{i2} \\ e_{i3} \end{bmatrix} \end{array}$$

$\mathbf{Y_i} = n * 1$ outcome vector
$\mathbf{X_i} = n * k$ matrix for predictors that have <u>fixed</u> effects
$\boldsymbol{\beta} = k * 1$ fixed effects vector
$\mathbf{Z_i} = n * u$ matrix for predictors that have <u>random</u> effects

n = # outcomes for person i
k = # model fixed effects
u = # model random effects

$\mathbf{U_i} = u * 1$ random effects vector
$\mathbf{E_i} = n * 1$ residual vector

Predicted V in Multivariate GLM: Total Variance and Covariance across $n = 3$ Outcomes for Person i

$$\boxed{\mathbf{V}_i = \mathbf{Z}_i * \mathbf{G}_i * \mathbf{Z}_i^{T} + \mathbf{R}_i}$$

$$\mathbf{V}_i = \begin{bmatrix}1\\1\\1\end{bmatrix}\begin{bmatrix}\tau^2_{U_0}\end{bmatrix}\begin{bmatrix}1 & 1 & 1\end{bmatrix}+\begin{bmatrix}\sigma^2_e & 0 & 0\\ 0 & \sigma^2_e & 0\\ 0 & 0 & \sigma^2_e\end{bmatrix}$$

$$\mathbf{V}_i = \begin{bmatrix}\tau^2_{U_0}+\sigma^2_e & \tau^2_{U_0} & \tau^2_{U_0}\\ \tau^2_{U_0} & \tau^2_{U_0}+\sigma^2_e & \tau^2_{U_0}\\ \tau^2_{U_0} & \tau^2_{U_0} & \tau^2_{U_0}+\sigma^2_e\end{bmatrix}$$

$$\mathbf{V}_i = \mathbf{R}_i = \begin{bmatrix}\mathrm{CS}+\sigma^2_e & \mathrm{CS} & \mathrm{CS}\\ \mathrm{CS} & \mathrm{CS}+\sigma^2_e & \mathrm{CS}\\ \mathrm{CS} & \mathrm{CS} & \mathrm{CS}+\sigma^2_e\end{bmatrix}$$

Same result: compound symmetry, either indirectly (**G** & **R**→ **V**) or directly (CS for **R**)

$\mathbf{Z}_i$ = $n\ x\ u$ values of **predictors with random effects**, so can differ per person (u = 1: intercept)

$\mathbf{Z}_i^{T}$ = $u\ x\ n$ values of predictors with random effects (just $\mathbf{Z}_i$ transposed)

$\mathbf{G}_i$ = $u\ x\ u$ estimated **random effects variances and covariances**, so will be the same for all persons (τ^2_{U0} = intercept variance)

$\mathbf{R}_i$ = $n\ x\ n$ **outcome-specific residual variances and covariances**, so will be same for all persons (here, just diagonal σ^2_e, although it's possible to add heterogeneous variances and/or covariances)

Distribution Terminology for MVN

- Scalar: $y_{it} = \beta_{00} + \beta_{01}(dvA_{it}) + \beta_{02}(dvB_{it}) + U_{i0} + e_{it}$
  Matrix: $\mathbf{Y_i} = \mathbf{X_i \beta} + \mathbf{Z_i U_i} + \mathbf{E_i}$

$\hat{\mathbf{Y}}_\mathbf{i} = \mathbf{X_i \beta}$ where $\hat{\mathbf{Y}}_\mathbf{i}$ **is the conditional Mean** created by **fixed effects** in the model for means

Model for the Variance creates $\mathbf{V_i}$

$$\mathbf{V}_\mathrm{i} = \mathbf{Z}_\mathrm{i} * \mathbf{G} * \mathbf{Z}_\mathrm{i}^\mathrm{T} + \mathbf{R}_\mathrm{i}$$

$$\mathbf{V}_\mathrm{i} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} \tau^2_{\mathrm{U}_0} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} \sigma^2_\mathrm{e} & 0 & 0 \\ 0 & \sigma^2_\mathrm{e} & 0 \\ 0 & 0 & \sigma^2_\mathrm{e} \end{bmatrix}$$

- This model says the "**marginal**" distribution of the total column of $\mathbf{Y}$ outcomes is: $\mathbf{Y} \sim \mathrm{N}(\mathbf{X\beta}, \mathbf{V})$
- This model says the "**conditional**" distribution of the total column of $\mathbf{Y}$ outcomes is: $\mathbf{Y|U} \sim \mathrm{N}(\mathbf{X\beta} + \mathbf{ZU}, \mathbf{R})$
  - Conditional = after controlling for fixed AND random effects
  - Marginal and conditional "general" linear models both have same normal distribution (which makes ML estimation relatively straightforward)

Fewer Options for Generalized Models

- Conditional distribution in multivariate general linear models: $\mathbf{Y}|\mathbf{U} \sim \mathrm{N}(\mathbf{X}\boldsymbol{\beta} + \mathbf{ZU}, \mathbf{R})$
- But $\mathbf{R}$ and choices for its patterns doesn't exist for generalized model variants (when using true maximum likelihood, at least)
  - No separately estimated constant residual variance (e.g., in Bernoulli, multinomial, Poisson, or binomial) means no directly estimated residual covariances are possible for multivariate models in any software
  - Univariate software (e.g., SAS GLIMMIX, STATA/R GLM) does not have separate "stretchy" terms by outcome for negative binomial, beta-binomial, or gamma (and no separate submodel intercepts for ordinal outcomes)
- So to maintain independent observations in the conditional distribution, all multivariate outcome relationships must be modeled indirectly in the linear predictor using regressions among outcomes OR random effects
  - In tricking univariate software into multivariate, can use random effects only
  - In software for path analysis or structural equation models (SEM), can use regressions between outcomes OR random effects
- Estimation becomes harder because random effects must be integrated out of the likelihood (i.e., via adaptive Gaussian quadrature or Bayes)

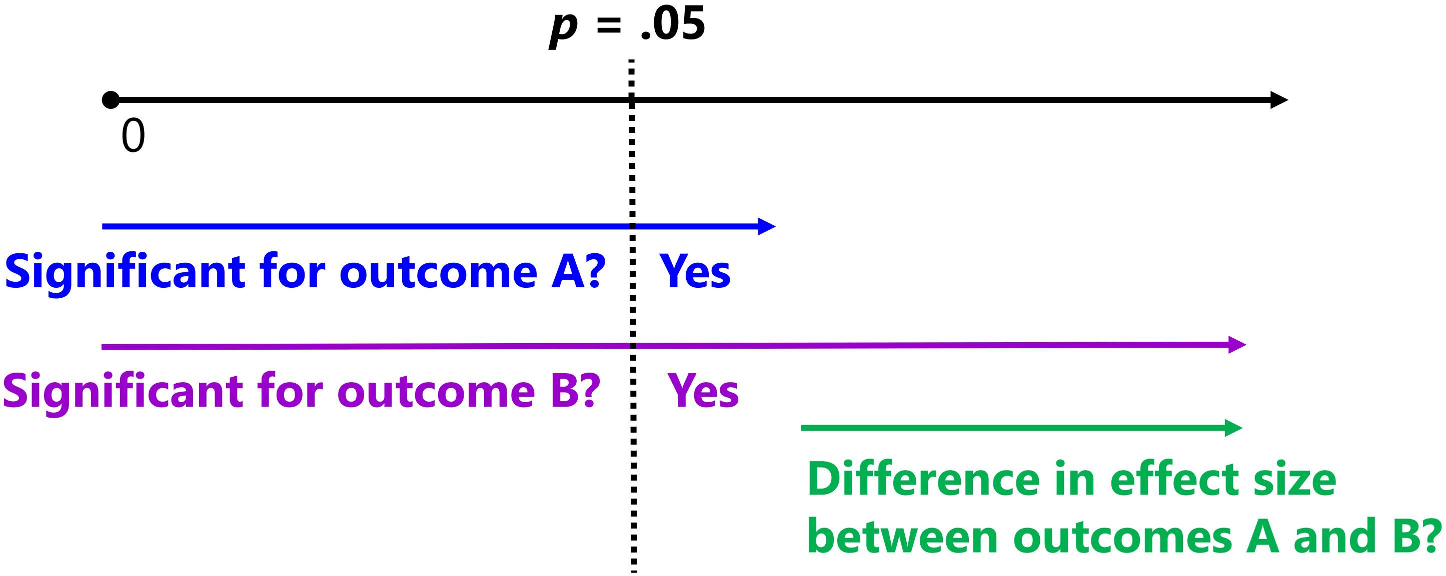
Use of Multivariate Models → Strategies:

- **Repeated measures** designs (e.g., sampling over occasions or conditions) readily lend themselves to multivariate models
  - Goal is to examine **mean differences** across the per-person outcomes (i.e., as in traditional RM ANOVA but using REML or ML for missing data)
  - Usual "**general intercept**" fixed effects strategy is likely most useful

- Multivariate models are also the optimal way to predict multiple outcomes (DVs)—simultaneously rather than in separate models
  - Examine **differences in predictor slopes** across outcomes (and then maybe constrain slopes that are similar in size to be equal for greater power)
    - Outcomes should be transformed to common scale (such as by z-scoring them) if not *similarly interpretable* already (e.g., such as variants of same scale)
    - If so, a "**DV-specific intercepts**" fixed effects strategy may be more useful
  - Predicting each outcome is better than predicting difference scores or to "controlling for time 1" ANCOVA ("residualized change", see here for why)

Differences in Effect Size across Outcomes

Scenario 1: Fixed slope is significant for both outcomes:



Absolute Value of Effect Size

p = .05

0

Significant for outcome A? Yes

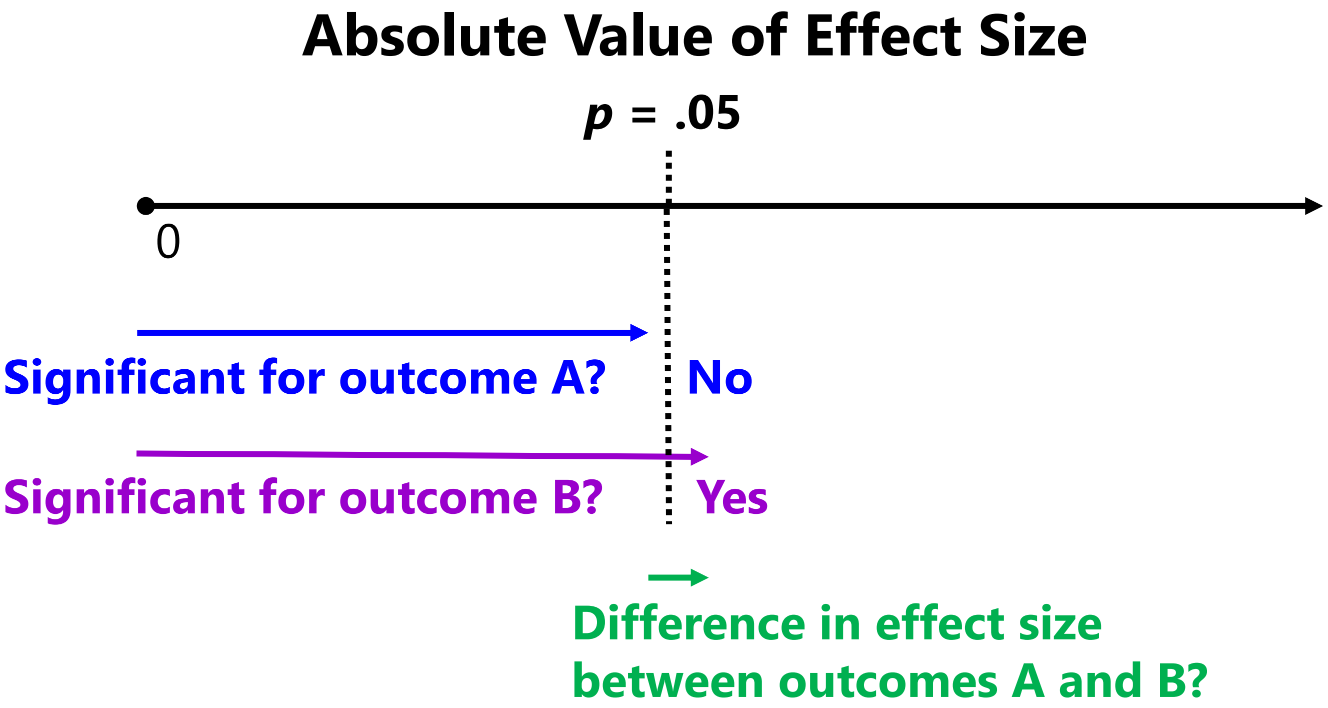
Significant for outcome B? Yes

Difference in effect size between outcomes A and B?

Just because a predictor slope is **significant for both outcomes** does not mean it has the **same magnitude** of relationship with both outcomes!

Differences in Effect Size across Outcomes

Scenario 2: Fixed slope is significant for outcome B only:



Absolute Value of Effect Size

***p* = .05**

0

Significant for outcome A? **No**

Significant for outcome B? **Yes**

Difference in effect size between outcomes A and B?

Also, just because a predictor slope is **non-significant for one outcome but significant for another outcome** does not mean it has **different magnitudes** of relationships across outcomes!

Fixed Effects Parameterizations

- Here are **2 ways of fitting the same model** predicting y_{id} (a "long" stacked outcome indexed by a categorical variable DV=A, B, or C) from a general intercept (i.e., a single column of 1s), a person-level predictor x_i, and 3 dummy-coded predictors: *dvA*, *dvB*, *dvC*:
  - If **DV=A**, then dvA=**1**, dvB=0, dvC=0
  - If **DV=B**, then dvA=0, dvB=**1**, dvC=0
  - If **DV=C**, then dvA=0, dvB=0, dvC=**1**

1. "**General intercept**": provides fixed effects for a reference DV and DV differences in fixed effects relative to the reference DV
   - So *fixed effects for non-reference DV* are found as linear combinations
2. "**DV-specific intercepts**": provides effects separately by DV
   - So *DV differences in fixed effects* are found as linear combinations
   - This is always how path models are parameterized (stay tuned)

Outcome "DV" as a Categorical Predictor

- DV-specific dummy codes can be replaced by a categorical DV predictor
  - SAS: put in on the CLASS statement; STATA: use i. prefix; R: use "factor"
- For a predictor with C categories, the program automatically then creates $\boldsymbol{C}$ **new internal variables**, for example "DV" with $C = 3$:

DV	**_IsA**	**_IsB**	**_IsC**
A	1	0	0
B	0	1	0
C	0	0	1

Default reference category by program:

- **SAS** (and SPSS): **highest/last** is reference
- STATA and R: **lowest/first** is reference
  - Can easily change reference category, e.g. in STATA, last = ref → ib(last).DV

- It then determines how many of these internal variables are needed to create C means—if using an intercept, then it's $C - 1$; without the intercept, is C
- It enters them until it hits that criterion—the one left out is your reference; if all C internal variables are included, then each is a custom intercept
- When referring the categorical predictor (e.g., ESTIMATE, LINCOM, CONTRAST), you must tell it what to do with EACH of these internal variables [e.g., 1 0 0]

"General Intercept" Parameterization

- **Empty Model:** $\boldsymbol{\hat{y}_{id} = \beta_{00} + \beta_{01}(dvA_{id}) + \beta_{02}(dvB_{id})}$
  - β_{00} = intercept for DV=C (i.e., when dvA=0 and dvB=0)
  - β_{01} = mean difference for DV=C vs. DV=A
  - β_{02} = mean difference for DV=C vs. DV=B
  - $\beta_{02} - \beta_{01}$ = mean difference for DV=A vs. DV=B
  - $\beta_{00} + \beta_{01}$ = intercept for DV=A
  - $\beta_{00} + \beta_{02}$ = intercept for DV=B

With the general intercept (=1 for all), the dvA and dvB variables capture typical mean differences (just between outcomes instead of predictor groups)

- **Add $\boldsymbol{x_i}$ predictor:** $\boldsymbol{+\beta_{10}(x_i) + \beta_{11}(dvA_{id})(x_i) + \beta_{12}(dvB_{id})(x_i)}$
  - β_{10} = x_i slope for DV=C (i.e., when x*dvA=0 and x*dvB=0)
  - β_{11} = difference in x_i slope for DV=C vs. DV=A
  - β_{12} = difference in x_i slope for DV=C vs. DV=B
  - $\beta_{12} - \beta_{11}$ = difference in x_i slope for DV=A vs. DV=B
  - $\beta_{10} + \beta_{11}$ = x_i slope for DV=A
  - $\beta_{10} + \beta_{12}$ = x_i slope for DV=B

“General Intercept” Parameterization

- Including a general intercept is convenient when differences between occasions or conditions are of most interest (i.e., then those difference scores are captured directly by fixed effects)
- Here are 2 equivalent versions of this model using **SAS MIXED**:

```
CLASS PersonID DV;
MODEL y = dvA dvB x dvA*x dvB*x
            / SOLUTION DDFM=Satterthwaite;
REPEATED DV / R RCORR TYPE=?? SUBJECT=PersonID;
```

```
CLASS PersonID DV;
MODEL y = DV x DV*x
            / SOLUTION DDFM=Satterthwaite;
REPEATED DV / R RCORR TYPE=?? SUBJECT=PersonID;
```

Note SAS REPEATED and STATA RESIDUALS **R** matrices stay the same across syntax variants

- Here are 2 equivalent versions of this model using **STATA MIXED**:

```
mixed y c.dvA c.dvB c.x c.dvA#c.x c.dvB#c.x,    ///
        || personid: , noconstant variance reml ///
        dfmethod(satterthwaite) residuals(??,t(DV))
```

```
mixed y ib(last).DV c.x ib(last).DV#c.x,        ///
        || personid: , noconstant variance reml ///
        dfmethod(satterthwaite)residuals(??,t(DV))
```

I used ib(last) to make C the reference DV (as in SAS)

“General Intercept” Parameterization

- Including a general intercept is convenient when differences between occasions or conditions are of most interest (i.e., then those difference scores are captured directly by fixed effects)
- Here are 2 equivalent versions of this model using **R GLS** (from NLME package, which allows customizable **R** matrices), in which DVc = recoded DV so that reference is DV = C:

  -
    ```
    Model = gls(data=dataname, method="REML",
                model=y~1 +dvA +dvB +x +dvA:x +dvB:x,
                correlation=??, weights=??)
    ```

  -
    ```
    Model = gls(data=dataname, method="REML",
                model=y~1 +factor(DVc) +x + factor(DVc):x,
                correlation=??, weights=??)
    ```

Note that **R** matrix (with covariances controlled by **`correlation`** and variances controlled by **`weights`**) would stays the same across syntax variants

“DV-Specific Intercepts” Parameterization

- **Empty Model:** $\hat{\boldsymbol{y}}_{\boldsymbol{id}} = \underline{\boldsymbol{\beta_{00}}(\boldsymbol{dvC_{id}})} + \boldsymbol{\beta_{01}}(\boldsymbol{dvA_{id}}) + \boldsymbol{\beta_{02}}(\boldsymbol{dvB_{id}})$
  - β_{00} = intercept for DV=C (i.e., when dvA=0 and dvB=0)
  - β_{01} = intercept for DV=A (i.e., when dvB=0 and dvC=0)
  - β_{02} = intercept for DV=B (i.e., when dvA=0 and dvC=0)
  - $\beta_{01} - \beta_{00}$ = mean difference for DV=C vs. DV=A
  - $\beta_{02} - \beta_{00}$ = mean difference for DV=C vs. DV=B
  - $\beta_{02} - \beta_{01}$ = mean difference for DV=A vs. DV=B

Without the general intercept (=1 for all), the dvA, dvB, and dvC dummy variables act like “switches” that turn on fixed effects for its DV

- **Add $\boldsymbol{x_i}$ predictor:** $+\, \underline{\boldsymbol{\beta_{10}}(\boldsymbol{x_i})(\boldsymbol{dvC_{di}})} + \boldsymbol{\beta_{11}}(\boldsymbol{x_i})(\boldsymbol{dvA_{id}}) + \boldsymbol{\beta_{12}}(\boldsymbol{x_i})(\boldsymbol{dvB_{id}})$
  - β_{10} = x_i slope for DV=C (i.e., when x*dvA=0 and x*dvB=0)
  - β_{11} = x_i slope for DV=A (i.e., when x*dvB=0 and x*dvC=0)
  - β_{12} = x_i slope for DV=B (i.e., when x*dvA=0 and x*dvC=0)
  - $\beta_{11} - \beta_{10}$ = difference in x_i slope for DV=C vs. DV=A
  - $\beta_{12} - \beta_{10}$ = difference in x_i slope for DV=C vs. DV=B
  - $\beta_{12} - \beta_{11}$ = difference in x_i slope for DV=A vs. DV=B

"DV-Specific Intercepts" Parameterization

- Removing the general intercept is convenient when fixed effects per DV are of most interest or when not all DVs get all fixed effects
- Here are 2 equivalent versions of this model using **SAS MIXED**:

```
CLASS PersonID DV;
MODEL y = dvC dvA dvB dvC*x dvA*x dvB*x
            / NOINT SOLUTION DDFM=Satterthwaite;
REPEATED DV / R RCORR TYPE=?? SUBJECT=PersonID;
```

```
CLASS PersonID DV;
MODEL y = DV DV*x
            / NOINT SOLUTION DDFM=Satterthwaite;
REPEATED DV / R RCORR TYPE=?? SUBJECT=PersonID;
```

Note SAS REPEATED and STATA RESIDUALS stay the same; in STATA you could also use ib(last) to make C the reference DV (same as in SAS)

- Here are 2 equivalent versions of this model using **STATA MIXED**:

```
mixed y c.dvC c.dvA c.dvB c.DVc#c.x c.dvA#c.x c.dvB#c.x, ///
        noconstant  || personid: , noconstant variance   ///
        reml dfmethod(satterthwaite) residuals(??,t(DVc))
```

```
mixed y ib(last).DV ib(last).DV#c.x,                     ///
        noconstant  || personid: , noconstant variance   ///
        reml dfmethod(satterthwaite) residuals(??,t(DVc))
```

“DV-Specific Intercepts” Parameterization

- Removing the general intercept is convenient when fixed effects per DV are of most interest or when not all DVs get all fixed effects
- Here are 2 equivalent versions of this model using **R GLS** (from NLME package, which allows customizable R matrices) in which DVc = recoded DV so that reference is DV = C:

```
➢ Model = gls(data=dataname, method="REML",
                 model=y~0 +dvC +dvA +dvB +dvC:x +dvA:x +dvB:x,
                 correlation=??, weights=??)
```

```
➢ Model = gls(data=dataname, method="REML",
                 model=y~0 +factor(DVc) + factor(DVc):x,
                 correlation=??, weights=??)
```

Note that **R** matrix (with covariances controlled by `correlation` and variances controlled by `weights`) would stays the same across syntax variants

Specifying Fixed Effects: Caveats

- Btw, to constrain the x_i slope to be equal across DVs, remove its DV-interaction terms—just enter x_i as a main effect (either version)
- You *can* mix-and-match parameterizations—but do so *carefully*!
  - e.g., here is how to retain separate effects of $x1_i$ for DVs A, B, and C, but have a slope of $x2_i$ only for DV=A and DV=B
  - **General intercept version:**

$$\begin{aligned}\hat{y}_{id} = \beta_{00} + \beta_{01}(dvA_{id}) + \beta_{02}(dvB_{id}) \\ + \beta_{10}(x1_i) + \beta_{11}(dvA_{id})(x1_i) + \beta_{12}(dvB_{id})(x1_i) \\ + \beta_{21}(dvA_{id})(x2_i) + \beta_{22}(dvB_{id})(x2_i)\end{aligned}$$

  - **DV-specific intercept version:**

$$\begin{aligned}\hat{y}_{id} = \beta_{00}(dvC_{id}) + \beta_{01}(dvA_{id}) + \beta_{02}(dvB_{id}) \\ + \beta_{10}(dvC_{id})(x1_i) + \beta_{11}(dvA_{id})(x1_i) + \beta_{12}(dvB_{id})(x1_i) \\ + \beta_{21}(dvA_{id})(x2_i) + \beta_{22}(dvB_{id})(x2_i)\end{aligned}$$

  - Either way, β_{21} and β_{22} give the effect of $x2_i$ for DV=A and DV=B (but it *looks* logically inconsistent in the general intercept version)

Wrapping Up…

- When each sampling unit has >1outcome → **multivariate models**
  - We need to add terms that capture **dependency (correlated residuals)**, this semester for balanced designs (i.e., same *potential* outcomes for all units)
  - For plausibly normal outcomes, **dependency can be modeled directly**: we can allow same or different residual variances and covariances across outcomes (in a **person-specific R matrix** of type UN, CSH, or CS)
  - We can use **likelihood ratio tests** (treat $-2\Delta LL$ as χ^2) to compare nested models to decide which fits least worse to protect our fixed effect SEs
  - For **other outcome types**, dependency **must** be modeled indirectly by including **random effects** (which means more challenging estimation)

- For convenience, **fixed effects** can be specified in 2 different ways
  - Single general intercept → DV terms reflect DV **differences**
  - Multiple DV-specific intercepts → DV terms are **switches** for own effects

- Univariate software for multivariate generalized linear models is less flexible than "truly" multivariate software—**so onto path analysis models**!!