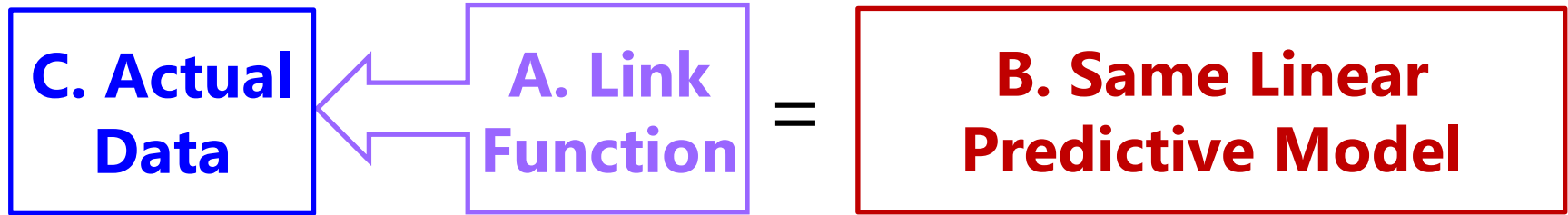


Multivariate Models via Univariate Software

- 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



- A. Link Function: Transformation of conditional mean used to keep *predicted outcomes* within the bounds of the outcome
- 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**”—**is 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 group differs across groups)
- We will not cover **unbalanced** outcomes in this class—they will be covered instead in classes focused on **multilevel models** (*aka*, mixed-effects models, hierarchical linear models) 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**, 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 generalized 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., $X \rightarrow M \rightarrow Y$, in which M is both an outcome of X and a predictor of Y, which involves regressions instead of correlations 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, **multivariate 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., *generalized* 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 independent observations using new vocabulary—fun with matrices!
 - Let's start with this general linear model: $y_i = \beta_0 + \beta_1(x_i) + e_i$
 - 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

- This GLM as scalar: $y_i = \beta_0 + \beta_1(x_i) + e_i$ with binary $x_i = 0$ or 1
- This GLM using matrices with $k = 2$ fixed effects: $Y = X\beta + E$

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

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

$Y = N * 1$ outcome vector

$X = N * k$ "design" matrix for predictors that have fixed effects

$\beta = k * 1$ fixed effects vector

$E = N * 1$ residual vector

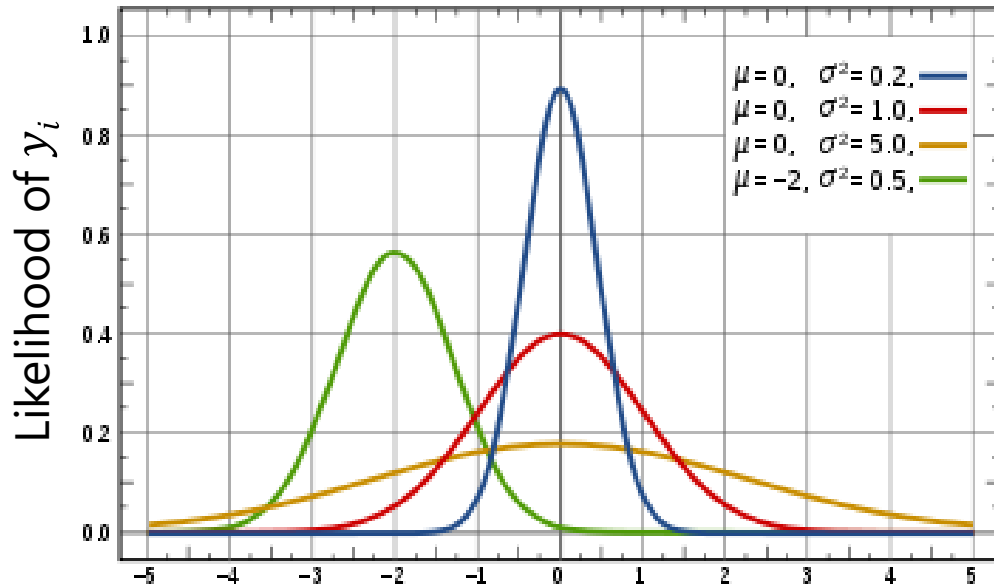
where (by default) $e_i \sim \text{Normal}(0, \sigma_e^2)$

Across all 6 persons, the **combined residual variance-covariance** matrix is "VC":

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

Off-diagonal 0 values \rightarrow independent residuals
 $\sigma_e^2 \rightarrow$ all persons share common residual variance

Review: Univariate Normal PDF



Univariate Normal PDF:

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

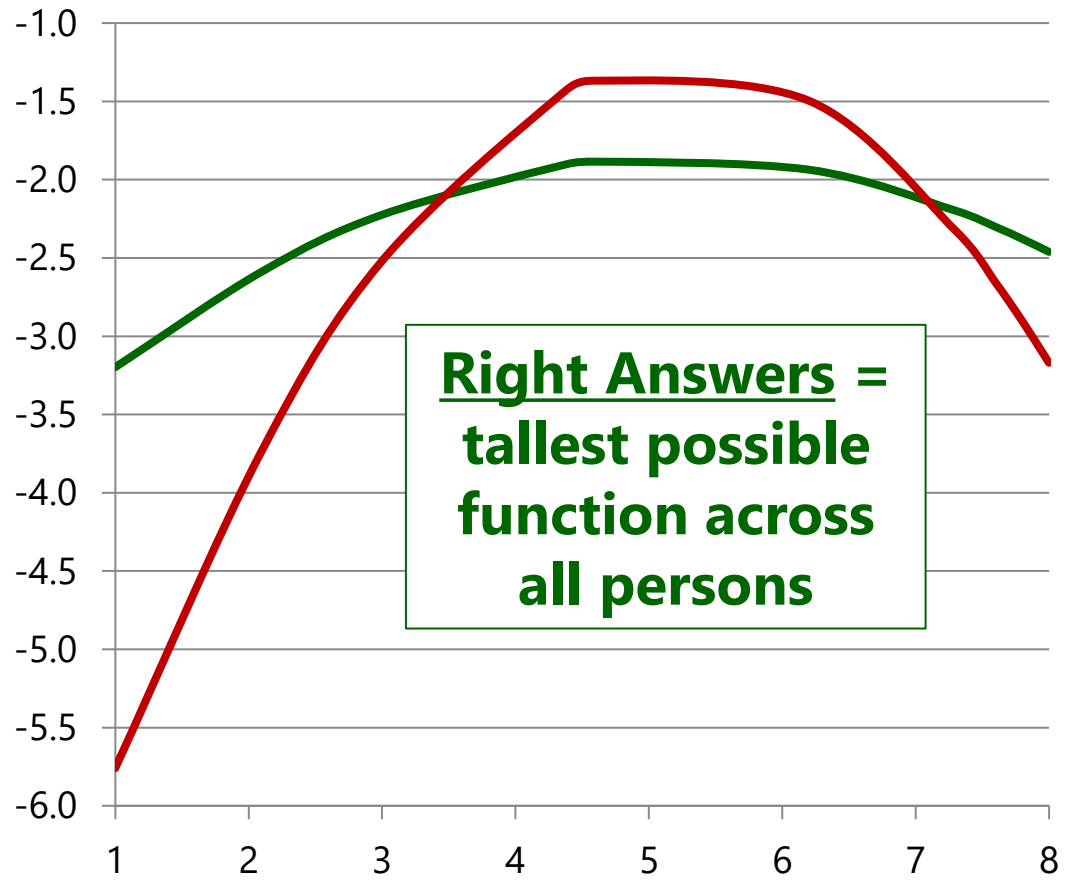
Sum over persons of log of $f(y_i)$ =
Model Log-Likelihood \rightarrow Model Fit

- This PDF tells us how **likely** (i.e., **tall**) any value of y_i is given two things:
 - Conditional mean \hat{y}_i
 - 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!

Univariate ML via Excel “NORMDIST”

Key idea: Normal Distribution formula → data height

Mean	5.19	5.24
Variance	6.56	2.00
	Right	Wrong
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



Tricking Univariate into Multivariate...

- What if the 6 observations were **2 outcomes (e.g., T1 and T2 occasions here) each from 3 persons** instead? We need a new model that builds in per-person residual dependency (and maybe different residual variances across outcomes 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 “long” data structure) like this:

“Wide” Structure:
1 row per person

Person	yT1	yT2
1	y_{1t1}	y_{1t2}
2	y_{2t1}	y_{2t2}
3	y_{3t1}	y_{3t2}

“Long” Structure:
1 row per outcome

Person	Time	x	y
1	1	0	y_{1t1}
1	2	1	y_{1t2}
2	1	0	y_{2t1}
2	2	1	y_{2t2}
3	1	0	y_{3t1}
3	2	1	y_{3t2}

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 , 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: $Y_i = X_i\beta + E_i$

$$Y_i \quad X_i \quad \beta \quad E_i$$

$$\begin{bmatrix} y_{t1} \\ y_{t2} \end{bmatrix} = \begin{bmatrix} 1 & x_{t1} \\ 1 & x_{t2} \end{bmatrix} \begin{bmatrix} \beta_{00} \\ \beta_{01} \end{bmatrix} + \begin{bmatrix} e_{t1} \\ e_{t2} \end{bmatrix}$$

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

Across all 3 persons, the **combined residual variance-covariance** matrix now has a "**block diagonal**" structure with three 2×2 per-person pockets of variances (σ^2) and covariance (σ):

where $e_{it} \sim \text{MVN}(0, R)$, and 2×2 R is chosen to predict **3 unique terms**:

MVN = multivariate normal $R = \begin{bmatrix} \sigma_1^2 & \sigma_{1,2} \\ \sigma_{1,2} & \sigma_2^2 \end{bmatrix}$

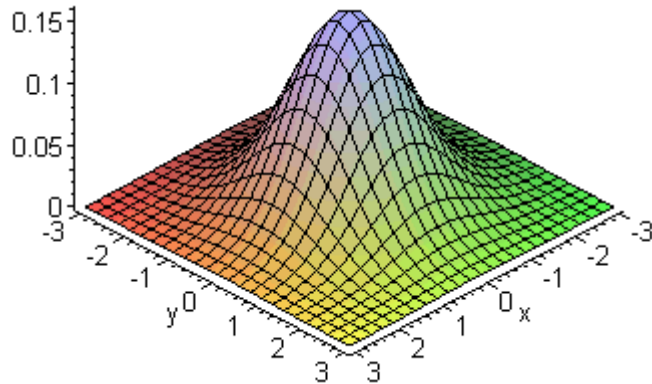
$$\begin{bmatrix} \sigma_1^2 & \sigma_{1,2} & 0 & 0 & 0 & 0 \\ \sigma_{1,2} & \sigma_2^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_1^2 & \sigma_{1,2} & 0 & 0 \\ 0 & 0 & \sigma_{1,2} & \sigma_2^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_1^2 & \sigma_{1,2} \\ 0 & 0 & 0 & 0 & \sigma_{1,2} & \sigma_2^2 \end{bmatrix}$$

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

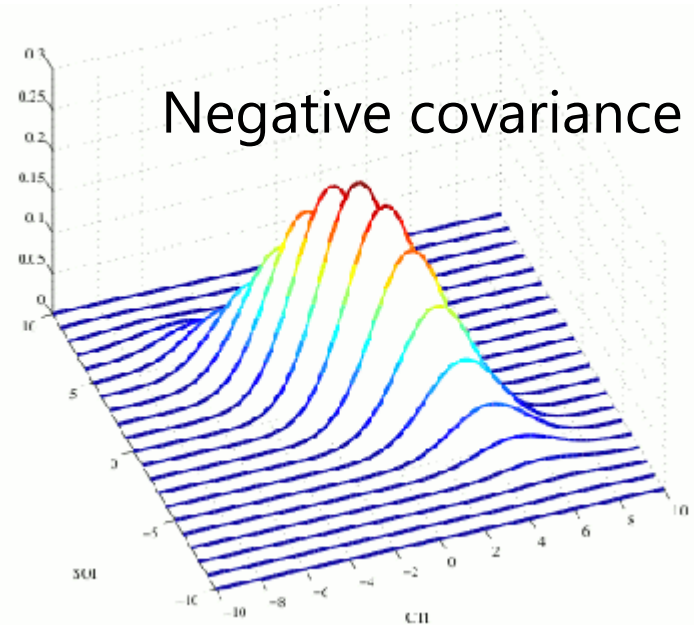
Welcome to Multivariate Normal!

- 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 Y_i

(height for all n outcomes for person i)

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

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

- In MVN, the single \hat{y}_i per person is replaced with $\mathbf{X}_i\boldsymbol{\beta}$ from the model
- The model also gives $\mathbf{R}_i \rightarrow$ 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
- $(\mathbf{R}_i)^{-1} \rightarrow$ matrix inverse \rightarrow 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" \rightarrow questionable results

What about Missing Data? Bad news...

- In univariate software using likelihood estimation: **each row 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 outcome

Person	Time	x	y
1	1	0	y_{1t1}
1	2	1	y_{1t2}
2	1	0	y_{2t1}
2	2	1	y_{2t2}
3	1	0	y_{3t1}
3	2	1	y_{3t2}

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

$$\begin{array}{c|ccc|c} Y_i & X_i & \beta & E_i \\ \hline \begin{bmatrix} y_{t1} \\ y_{t2} \\ y_{t3} \end{bmatrix} & \begin{bmatrix} 1 & x1_{t1} & x2_{t1} \\ 1 & x1_{t2} & x2_{t2} \\ 1 & x1_{t3} & x2_{t3} \end{bmatrix} & \begin{bmatrix} \beta_{00} \\ \beta_{01} \\ \beta_{02} \end{bmatrix} & \begin{bmatrix} e_{t1} \\ e_{t2} \\ e_{t3} \end{bmatrix} \end{array} +$$

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

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

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

Across both persons, the **combined residual variance-covariance** matrix now 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 \rightarrow no residual covariances *across persons*
 Same symbols \rightarrow all persons share common residual variances and covariances

What should the R Matrix Look Like?

- **Goal: predict all unique variances and covariances in R**
 - The “**direct**” way of doing so 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 (σ^2) the same across outcomes?
 - If so, then are residual covariances (σ) are also the same across outcome pairs (remember: covariance is unstandardized correlation)?
 - If not, might residual correlations (r) still be the same across outcome pairs (because covariances will differ if variances differ)?

Option 1 for R Patterns: The Answer Key

- **Option 1 is an “unstructured” R matrix:** all variances and covariances are 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)

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

➤ Left: \mathbf{R}_{cov} is a covariance matrix; \mathbf{R}_{cor} is a correlation matrix

- Btw, **an unstructured R matrix** is also known as 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 arrive 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)

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

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

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

- 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)

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)
 - VC (equal residual variances, no residual covariances) is nested in all others
 - CS fit better than VC? There's residual covariance (dependency) across outcomes
 - CS is nested in CSH, which are both nested in UN (= the data)
 - CSH fit better than CS? Then residual variances need to differ by outcome
 - UN fit better than CSH? Then res. 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 the data)
 - Why not just use UN always? It may not always be estimable, and using a simpler model that fits not worse can lead to greater power (because more unnecessary parameters \rightarrow less power)
- Btw, R matrix residual variances and covariances can also be allowed to differ across groups (see example 5c); 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 (with 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 cannot 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** \rightarrow **V** (Person is Unit)

- e.g., For three outcomes per person, a **Compound Symmetry R** matrix 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 CS can be produced an equivalent way, in which "CS" is the same thing as "random intercept variance" ($\tau_{U_0}^2$) which is distinguished from "residual variance" (σ_e^2)

Random effect source(s) of person dependency are moved to **G** Matrix (dimensions are NOT person-specific)

$$\mathbf{G} = \begin{bmatrix} \tau_{U_0}^2 \end{bmatrix}$$

" + "

Remaining within-person variance and covariance is in **R** matrix (dimensions are person-specific)

$$\mathbf{R}_i = \begin{bmatrix} \sigma_e^2 & 0 & 0 \\ 0 & \sigma_e^2 & 0 \\ 0 & 0 & \sigma_e^2 \end{bmatrix}$$

" = "

Total Predicted Residual Variance-Covariance Matrix is called **V** Matrix (dimensions are person-specific)

$$\mathbf{V}_i = \begin{bmatrix} \tau_{U_0}^2 + \sigma_e^2 & \tau_{U_0}^2 & \tau_{U_0}^2 \\ \tau_{U_0}^2 & \tau_{U_0}^2 + \sigma_e^2 & \tau_{U_0}^2 \\ \tau_{U_0}^2 & \tau_{U_0}^2 & \tau_{U_0}^2 + \sigma_e^2 \end{bmatrix}$$

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):

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

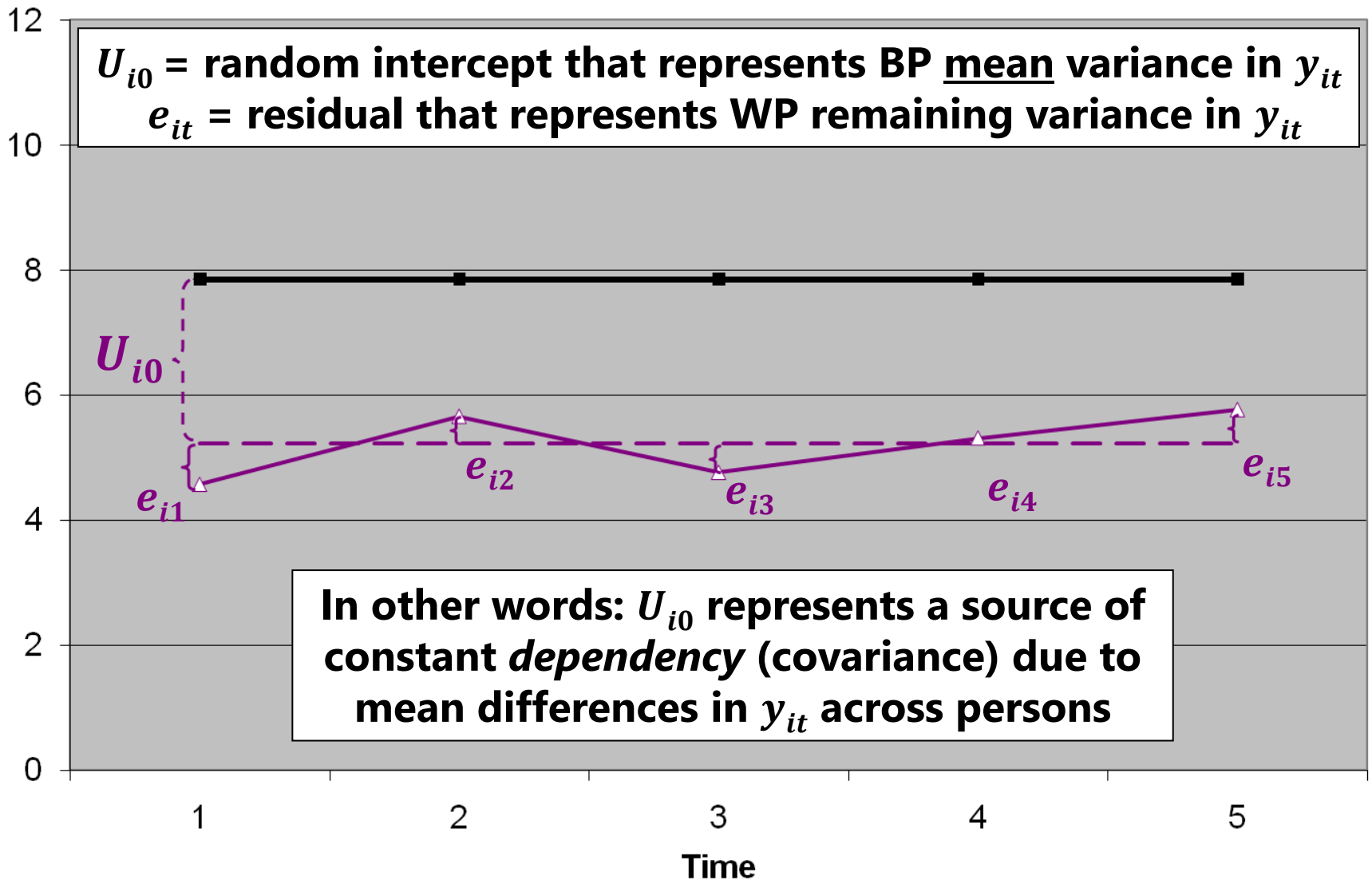
- β Fixed effects → create conditional mean from predictors
- e_i = person-specific residual deviation from predicted y_i

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

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

- 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



Indirect Dependency via a Random Intercept

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

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

- In matrix notation, this becomes $Y_i = X_i\beta + Z_iU_i + E_i$

Y_i	X_i	β	Z_i	U_i	E_i
$\begin{bmatrix} y_{t1} \\ y_{t2} \\ y_{t3} \end{bmatrix}$	$\begin{bmatrix} 1 & dvA_{t1} & dvB_{t1} \\ 1 & dvA_{t2} & dvB_{t2} \\ 1 & dvA_{t3} & dvB_{t3} \end{bmatrix}$	$\begin{bmatrix} \beta_{00} \\ \beta_{01} \\ \beta_{02} \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$	$[U_{i0}]$	$\begin{bmatrix} e_{t1} \\ e_{t2} \\ e_{t3} \end{bmatrix}$

$Y_i = n * 1$ outcome vector
 $X_i = n * k$ matrix for predictors that have fixed effects
 $\beta = k * 1$ fixed effects vector
 $Z_i = n * u$ matrix for predictors that have random effects

$n = \#$ outcomes for person i
 $k = \#$ model fixed effects
 $u = \#$ model random effects
 $U_i = u * 1$ random effects vector
 $E_i = n * 1$ residual vector

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

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

$$\mathbf{V}_i = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} \tau_{U_0}^2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} \sigma_e^2 & 0 & 0 \\ 0 & \sigma_e^2 & 0 \\ 0 & 0 & \sigma_e^2 \end{bmatrix}$$

$$\mathbf{V}_i = \begin{bmatrix} \tau_{U_0}^2 + \sigma_e^2 & \tau_{U_0}^2 & \tau_{U_0}^2 \\ \tau_{U_0}^2 & \tau_{U_0}^2 + \sigma_e^2 & \tau_{U_0}^2 \\ \tau_{U_0}^2 & \tau_{U_0}^2 & \tau_{U_0}^2 + \sigma_e^2 \end{bmatrix}$$

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

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

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

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

$\mathbf{G}_i = u \times u$ estimated **random effects variances and covariances**, so will be the same for all persons ($\tau_{U_0}^2 =$ intercept variance)

$\mathbf{R}_i = n \times n$ **outcome-specific residual variances and covariances**, so will be same for all persons (here, just diagonal σ_e^2 , 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: $Y_i = X_i\beta + Z_iU_i + E_i$

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

Model for the Variance creates V_i

$$V_i = Z_i^* G_i^* Z_i^T + R_i$$

$$V_i = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} \tau_{U_0}^2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \end{bmatrix} + \begin{bmatrix} \sigma_e^2 & 0 & 0 \\ 0 & \sigma_e^2 & 0 \\ 0 & 0 & \sigma_e^2 \end{bmatrix}$$

- This model says the “**marginal**” distribution of the total column of Y outcomes is: $Y \sim N(X\beta, V)$
- This model says the “**conditional**” distribution of the total column of Y outcomes is: $Y|U \sim N(X\beta + ZU, R)$
 - 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

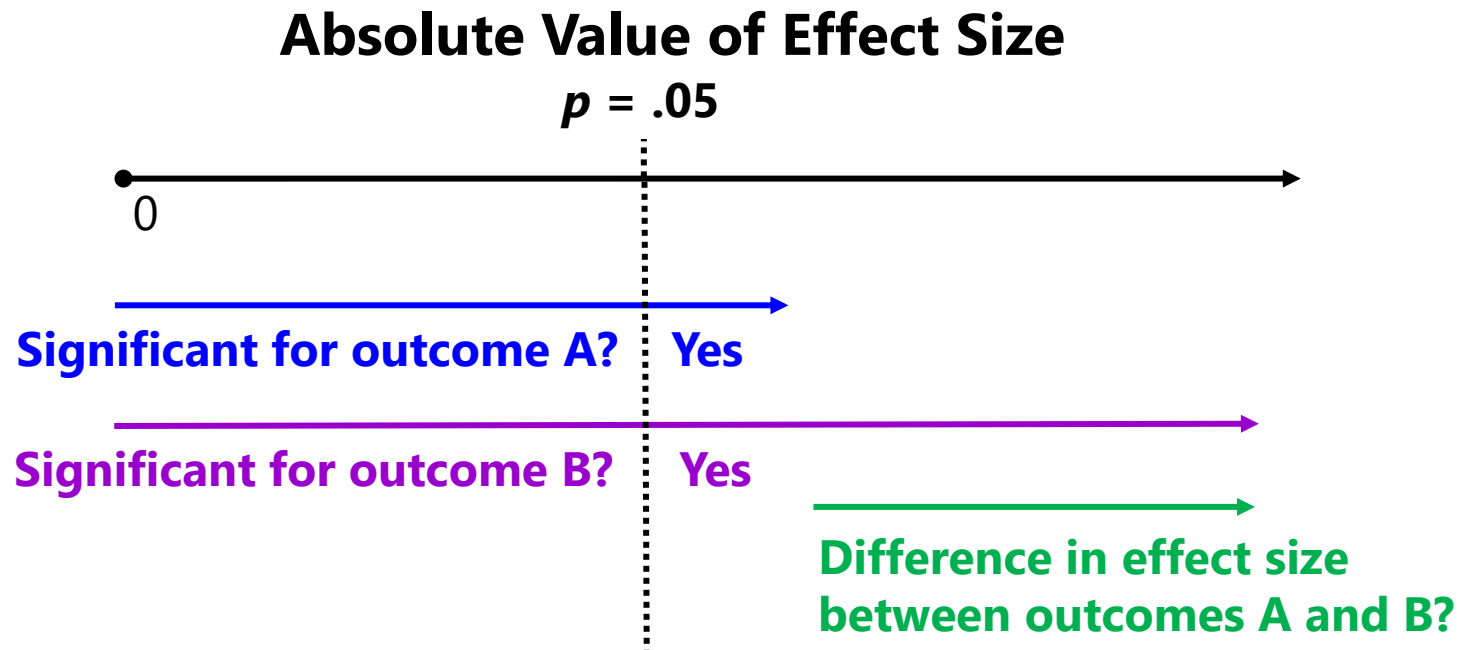
- Conditional distribution in multivariate general linear models:
 $Y|U \sim N(X\beta + ZU, R)$
- But R and choices for its patterns doesn't exist for generalized model variants (when using true maximum likelihood, at least)
 - No separately estimated constant residual variance (e.g., in Bernoulli, multinomial, 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 effects** across outcomes (and then constrain effects 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 a better alternative to predicting difference scores or to “controlling for time 1” ANCOVA (“residualized change”, bleh)

Differences in Effect Size across Outcomes

Scenario 1: Fixed slope is significant for both outcomes:

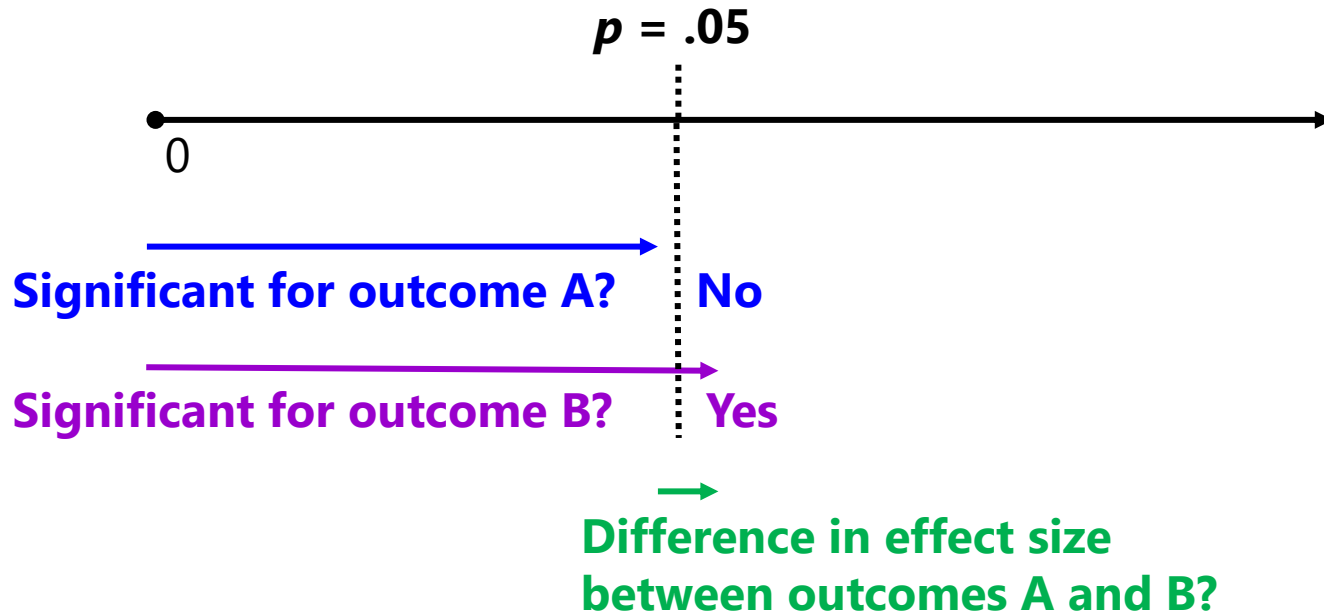


Just because a predictor slope is **significant for both outcomes** does not mean it has the **same magnitude** of relationship with both outcomes!

Differences in Effect Size across Outcomes

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

Absolute Value of Effect Size



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 $DV=A, B, \text{ or } C$) 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 **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:** $\hat{y}_{id} = \beta_{00} + \beta_{01}(dvA_{id}) + \beta_{02}(dvB_{id})$

- 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 act like typical group differences (just between outcomes instead of predictor groups)

- **Add x_i predictor:** $+ \beta_{10}(x_i) + \beta_{11}(dvA_{id})(x_i) + \beta_{12}(dvB_{id})(x_i)$

- 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):

- `Model = gls(data=Example5a, method="REML",
 model=y~1 +dvA +dvB +x +dvA:x +dvB:x,
 correlation=??, weights=??)`
- `Model = gls(data=Example5a, method="REML",
 model=y~1 +factor(DV3) +x + factor(DV3):x,
 correlation=??, weights=??)`

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

“DV-Specific Intercepts” Parameterization

• **Empty Model:** $\hat{y}_{id} = \beta_{00}(dvC_{id}) + \beta_{01}(dvA_{id}) + \beta_{02}(dvB_{id})$

- 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 x_i predictor:** $+ \beta_{10}(x_i)(dvC_{di}) + \beta_{11}(x_i)(dvA_{id}) + \beta_{12}(x_i)(dvB_{id})$

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

“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):
 - `Model = gls(data=Example5a, method="REML",
 model=y~0 +dvC +dvA +dvB +dvC:x +dvA:x +dvB:x,
 correlation=??, weights=??)`
 - `Model = gls(data=Example5a, method="REML",
 model=y~0 +factor(DV3) + factor(DV3):x,
 correlation=??, weights=??)`

Note that R matrix (with covariances controlled by `correlation` and variances controlled by `weights`) would stay 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:**
$$\hat{y}_{id} = \beta_{00} + \beta_{01}(dvA_{id}) + \beta_{02}(dvB_{id}) + \beta_{10}(x1_i) + \beta_{11}(dvA_{id})(x1_i) + \beta_{12}(dvB_{id})(x1_i) + \beta_{21}(dvA_{id})(x2_i) + \beta_{22}(dvB_{id})(x2_i)$$
 - **DV-specific intercept version:**
$$\hat{y}_{id} = \beta_{00}(dvC_{id}) + \beta_{01}(dvA_{id}) + \beta_{02}(dvB_{id}) + \beta_{10}(dvC_{id})(x1_i) + \beta_{11}(dvA_{id})(x1_i) + \beta_{12}(dvB_{id})(x1_i) + \beta_{21}(dvA_{id})(x2_i) + \beta_{22}(dvB_{id})(x2_i)$$
 - 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 >1 outcome \rightarrow **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** ($-2\Delta LL$ 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 \rightarrow DV terms reflect DV **differences**
 - Multiple DV-specific intercepts \rightarrow 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!!**