

S-043/Stat-151
Analysis for Clustered and Longitudinal Data
(Multilevel & Longitudinal Models)

Lecture 6.3:
Fitting models with complex
residual error structure

Office hours

Now sign up for 20 minute slots

1 with instructor per week max

1 with TF per week max

TFs generously have many, many slots. But they also need to graduate.

PLEASE come to one instructor slot at some point at least.





Sign up for final
project presentation
slots ASAP

Road Map

Setup: What we have been doing so far, presented in a manner we can build on it.

The math: Different ways we can represent residual structure

Seeing it in action: We will fit the different models to the same data, and look at the results to understand what they look like.

This is about building models to fit the residual patterns and variation in the data.



The Setup:
Why today
is
happening

The data: National Youth Survey (NYS)

This data comes from a survey in which the same students were asked yearly about their acceptance of 9 “deviant” behaviors (such as smoking marijuana, stealing, etc.).

The study began in 1976

Key variables:

- ★ “Attitude towards deviation” (our outcome)
- ★ Gender, minority status, family income

The concern: complex temporal dependencies

We have longitudinal data, measured at specific **waves**

We can fit a **linear growth model**, but worry that our within-student residuals are **not independent, identically distributed**.

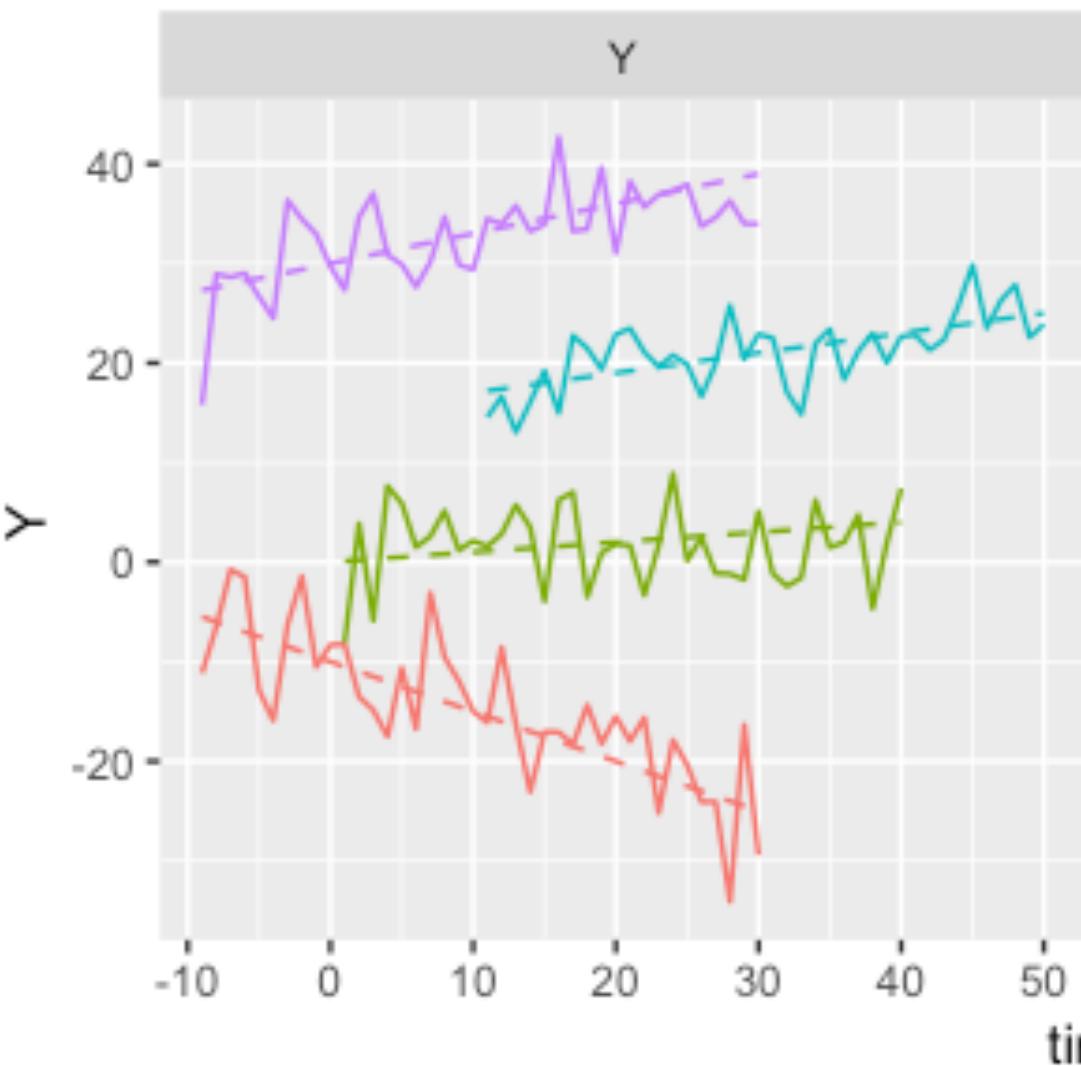
I.e., in our many small worlds we **do not have OLS within world**.

We need to fix that to get good standard errors. (E.g., same worry as cluster robust SEs from last lecture.)

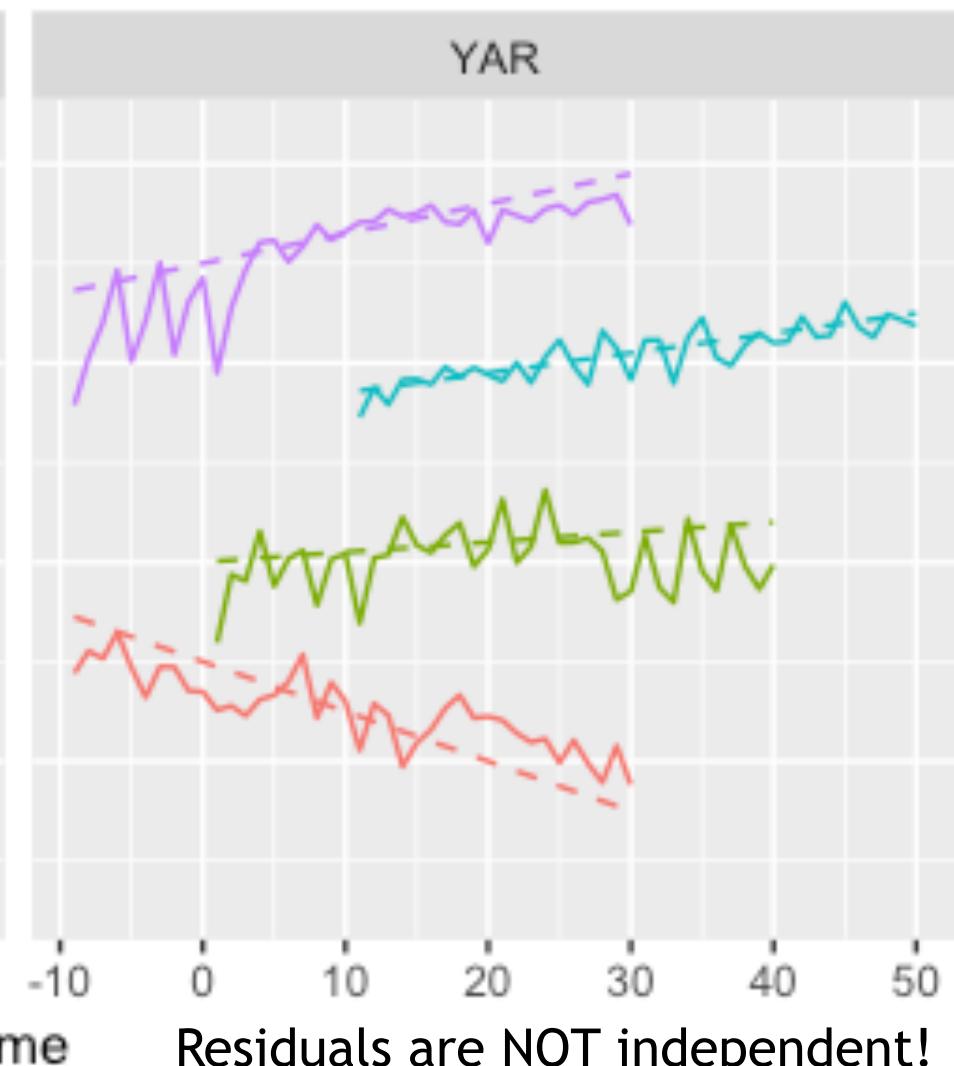
Autoregression as a visual

Four students, 40 time points each

i.i.d. residuals



autoregressive residuals



Residuals are NOT independent!

What we've done so far...

In everything so far, every **residual** has the same variance and 0 correlation with any other **residual** (in the same group).

This is a *diagonal* variance-covariance matrix (in a given group).

It's our default choice when it comes to most MLMs.

BUT, **longitudinal data** can be different, if measurements are taken at the same point for each respondent.

Our Unconditional Linear Growth Model

$$Y_{ti} = \pi_{0i} + \pi_{1i}a_{ti} + \epsilon_{ti}$$

Notation from
R&B pg 163-164

$$\epsilon_{ti} \sim N(0, \sigma^2)$$

Our Covariance Matrix

$$\pi_{0i} = \beta_{00} + r_{0i}$$

$$M = \begin{bmatrix} \tau_{00} & \\ \tau_{10} & \tau_{11} \end{bmatrix}$$

$$\pi_{1i} = \beta_{10} + r_{1i}$$

$$(r_{0i}, r_{1i}) \sim N(\mathbf{0}, M)$$

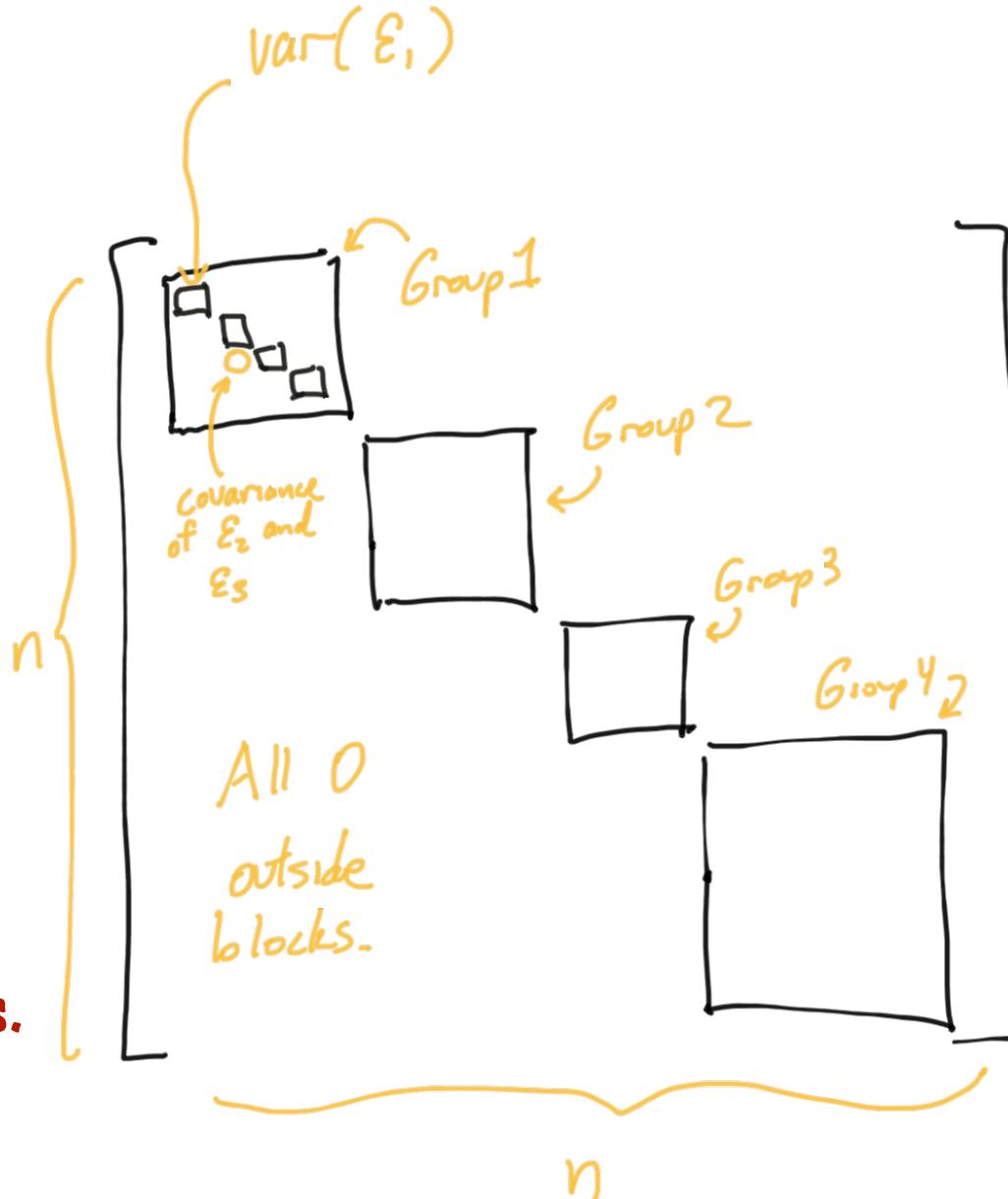
We use M to denote
the whole matrix.

- ★ a_{ti} , *age*, is the actual time of the t^{th} observation of child i (this allows for different measurement occasions).
- ★ The above is a classic random-slope model, nothing more, nothing less.
- ★ a_{ti} in this case is “time elapsed from first measure”, which gives a specific interpretation to the intercept.

Overall, all our residuals are block diagonal, with each block being the same structure

Our variance-covariance of residuals

$$\Sigma =$$



Each group (student) has the same structure for the residuals.

We have 4 clusters.
Cluster 1 has 4 units in it.

Two ways forward

Method 1: *overall* residuals $u_{ti} = r_i + \epsilon_{ti}, t = 1, \dots, 5$

- ★ Think of the units (students) as independent draws of residuals, and estimate the student-level matrix

Method 2: *within-student* residuals $\epsilon_{ti}, t = 1, \dots, 5$

- ★ Have a multilevel model, but then model the residuals on top of the multilevel model.

Modeling $u_{ti} = r_i + \epsilon_{ti}, t = 1, \dots, 5$



A very flexible specification

We could assume for each respondent i , that the residuals are like this:

$$\begin{pmatrix} u_{i1} \\ u_{i2} \\ u_{i3} \\ u_{i4} \\ u_{i5} \end{pmatrix} \sim N \left[\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tau_{11} & \tau_{12} & \tau_{13} & \tau_{14} & \tau_{15} \\ \tau_{21} & \tau_{22} & \tau_{23} & \tau_{24} & \tau_{25} \\ \tau_{31} & \tau_{32} & \tau_{33} & \tau_{34} & \tau_{35} \\ \tau_{41} & \tau_{42} & \tau_{43} & \tau_{44} & \tau_{45} \\ \tau_{51} & \tau_{52} & \tau_{53} & \tau_{54} & \tau_{55} \end{pmatrix} \right]$$

15 parameters

I.e., any possible variance-covariance matrix for the residuals.

We need the **same matrix across respondents** because we need to see multiple examples of that matrix to estimate the various parameters. (This is **DIFFERENT** from cluster-robust, where we average (poorly estimated) individual but different matrices)

This is kind of like a homoskedasticity assumption **for the groups**.

Autoregressive Error Structure

The residuals that are closer together are more correlated

Our matrix has 2 parameters defining our overall structure

$$\begin{pmatrix} u_{i1} \\ u_{i2} \\ u_{i3} \\ u_{i4} \\ u_{i5} \end{pmatrix} \sim N \left[\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma^2 & \sigma^2\rho & \sigma^2\rho^2 & \sigma^2\rho^3 & \sigma^2\rho^4 \\ \sigma^2 & \sigma^2 & \sigma^2\rho & \sigma^2\rho^2 & \sigma^2\rho^3 \\ . & \sigma^2 & \sigma^2\rho & \sigma^2\rho^2 & \sigma^2\rho^3 \\ . & . & \sigma^2 & \sigma^2\rho & \sigma^2\rho^2 \\ . & . & . & \sigma^2 & \sigma^2\rho \\ . & . & . & . & \sigma^2 \end{pmatrix} \right]$$

We can express the covariance between any two residuals as

$$cov(\epsilon_j, \epsilon_k) = \sigma^2 \rho^{|j-k|}$$

Much less flexible than unconstrained, but simple parameters that tell us about structure.

Marginal error and error within group

$$Y_{ij} = \beta_0 + \beta_1 X_{ij} + u_j + \epsilon_{ij}$$

In the above random intercept model we have *two* error terms, one for the group, and one for the residual.

If we think of this as an *overall* error term, we have for the residuals of a group:

$$\Sigma = \begin{pmatrix} \sigma^2 + \tau & & & & \\ \tau & \sigma^2 + \tau & & & \\ \tau & \tau & \sigma^2 + \tau & & \\ \tau & \tau & \tau & \sigma^2 + \tau & \\ \tau & \tau & \tau & \tau & \sigma^2 + \tau \end{pmatrix}$$

Modeling within-student residuals

$$\epsilon_{ti}, t = 1, \dots, 5$$

on top of our multilevel model



Mind-blowing extension: MLM along with structure on the residuals

Consider breaking our matrix into two pieces, the multilevel piece (M) and the residual piece:

$$\Sigma_i = M + \sigma^2 I_5$$

variance-covariance for overall residuals of group i **multilevel piece** **“left-over” piece (here, the simplest, OLS)**

Our residual piece is plain old OLS i.i.d. residuals.

We can now put structure (e.g., heteroskedasticity or autoregressive structure) on the second piece

Default: Vanilla OLS residuals

This is how we can think of 5 i.i.d. residuals, all normal and independent:

$$\begin{pmatrix} \epsilon_{1i} \\ \epsilon_{2i} \\ \epsilon_{3i} \\ \epsilon_{4i} \\ \epsilon_{5i} \end{pmatrix} \sim N \left[\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma^2 & & & & \\ 0 & \sigma^2 & & & \\ 0 & 0 & \sigma^2 & & \\ 0 & 0 & 0 & \sigma^2 & \\ 0 & 0 & 0 & 0 & \sigma^2 \end{pmatrix} \right]$$

Our *overall* residuals $u_{ti} = r_i + \epsilon_{ti}$ are then this thing plus our single (shared) r_i for the student

This one allows each time point to have its own variance (but everything is still independent)

$$\begin{pmatrix} \epsilon_{1i} \\ \epsilon_{2i} \\ \epsilon_{3i} \\ \epsilon_{4i} \\ \epsilon_{5i} \end{pmatrix} \sim N \left[\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_1^2 & & & & \\ 0 & \sigma_2^2 & & & \\ 0 & 0 & \sigma_3^2 & & \\ 0 & 0 & 0 & \sigma_4^2 & \\ 0 & 0 & 0 & 0 & \sigma_5^2 \end{pmatrix} \right]$$

We have heteroskedasticity
across the time points.

So how do we fit these?

We use R, of course.

But we can't use lmer()...



R HAS DIFFERENT PACKAGES

Two R packages: nlme and lme4

R has multiple packages. Two are of interest:

lme4 (with the `lmer()` command)

- ★ This is the package we have been using.
- ★ Nice, clean specification of models.
- ★ Deals with non-Gaussian outcomes (logistic, count, etc.).

nlme (with the `lme()` and `gls()` commands)

- ★ Also does multilevel models.
- ★ **Easier access to specifying fancy correlation structure.**
- ★ Less current, more fussy, bit of a pain.
- ★ Probably your core package if you are a repeated measures/longitudinal sort of person.

What is Generalized Least Squares (GLS)

Generalized Least Squares is when you fit a normal regression model, but have a specific idea of what the residual structure looks like.

In R, you use `gls()`, and do a `lm()` style formula followed with specification of the correlation structure.

This is not the world of multilevel models

You can express multilevel type structure in the residual specification, but this framework doesn't "think this way"

Generalized Linear Model (GLM)
is not the same as
Generalized Least Squares (GLS)

Logistic regression, etc

Weird residual structure



An example of different structure fit to the same data

Raudenbush & Brky pg 190



Comparing several models

R&B walk through the following:

- ★ Random intercept (compound symmetry)
- ★ Random slope
- ★ Random slope with heteroskedastic residual error by time
- ★ Autoregressive structure with no random intercepts
- ★ Fully unconstrained covariance matrix on the 5 residuals (assuming i.i.d. youth)



Compound Symmetry with gls()

```
> modelCompSymm = gls(ATTIT ~ AGE,  
+                      data=nys1,  
+                      correlation=corCompSymm(form=~AGE | ID) )  
> summary(modelCompSymm)
```

Generalized least squares fit by REML
Model: ATTIT ~ AGE

Correlation Structure: Compound symmetry

Formula: ~AGE | ID

Parameter estimate(s) :

Rho

0.51

Coefficients:

	Value	Std. Error	t-value	p-value
(Intercept)	-0.51	0.054	-9.5	0
AGE	0.06	0.004	16.2	0

Correlation:

(Intr)

AGE -0.97

Standardized residuals:

Min	Q1	Med	Q3	Max
-1.771	-0.771	-0.064	0.712	3.384

Residual standard error: 0.26

Degrees of freedom: 1079 total; 1077 residual

No discussion of random effects.
Instead we talk about what the covariance matrix looks like

Note we are looking at residuals around the mean model now.





The correlation matrix for our model

```
> getVarCov(modelCompSymm, type="marginal",  
           individual=3)
```

Marginal variance covariance matrix

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0.066	0.034	0.034	0.034	0.034
[2,]	0.034	0.066	0.034	0.034	0.034
[3,]	0.034	0.034	0.066	0.034	0.034
[4,]	0.034	0.034	0.034	0.066	0.034
[5,]	0.034	0.034	0.034	0.034	0.066

Standard Deviations: 0.26 0.26 0.26 0.26 0.26

These are the square root of the diagonal

$$\begin{pmatrix} \sigma^2 + \tau & & & & \\ \tau & \sigma^2 + \tau & & & \\ \tau & \tau & \sigma^2 + \tau & & \\ \tau & \tau & \tau & \sigma^2 + \tau & \\ \tau & \tau & \tau & \tau & \sigma^2 + \tau \end{pmatrix}$$

Note: This ACTUALLY MATCHES pg 195 in R&B (I am stunned)



Which individual to get our covariance matrix for (all would be the same in our case)



AR(1) (only) model

```
> modelAR1 = gls(ATTIT ~ AGE,  
+                   data=nys1,  
+                   correlation=corAR1(form=~AGE | ID) )  
>  
>
```

```
> summary(modelAR1)
```

Generalized least squares fit by REML
Model: ATTIT ~ AGE

Correlation Structure: ARMA(1,0)

Formula: ~AGE | ID

Parameter estimate(s) :

Phil
0.62

Coefficients:

	Value	Std.Error	t-value	p-value
(Intercept)	-0.45	0.075	-6	0
AGE	0.06	0.006	11	0

Correlation:

(Intr)
AGE -0.99

Standardized residuals:

Min	Q1	Med	Q3	Max
-1.750	-0.811	-0.033	0.748	3.404

Residual standard error: 0.26

Degrees of freedom: 1079 total; 1077 residual



AR(1) covariance matrix

```
> getVarCov(modelAR1,type="marginal", individual=7)
Marginal variance covariance matrix
 [,1] [,2] [,3] [,4] [,5]
[1,] 0.0656 0.040 0.025 0.015 0.0094
[2,] 0.0404 0.066 0.040 0.025 0.0153
[3,] 0.0249 0.040 0.066 0.040 0.0249
[4,] 0.0153 0.025 0.040 0.066 0.0404
[5,] 0.0094 0.015 0.025 0.040 0.0656
Standard Deviations: 0.26 0.26 0.26 0.26 0.26
>
```

$$\begin{pmatrix} \sigma^2 & & & & \\ \rho\sigma^2 & \sigma^2 & & & \\ \rho^2\sigma^2 & \rho\sigma & \sigma^2 & & \\ \rho^3\sigma^2 & \rho^2\sigma^2 & \rho\sigma^2 & \sigma^2 & \\ \rho^4\sigma^2 & \rho^3\sigma^2 & \rho^2\sigma^2 & \rho\sigma^2 & \sigma^2 \end{pmatrix}$$

This strip is for two apart

Some rounding but notice diagonal striping

Nearly same as in R&B



Random Slopes

```
> modelRS = lme(ATTIT ~ AGE,  
+                 data=nys1,  
+                 random=~AGE | ID )
```

```
> summary(modelRS)
```

Linear mixed-effects model fit by REML

Random effects:

Formula: ~AGE | ID

Structure: General positive-definite, Log-Cholesky parametrization
StdDev Corr

(Intercept)	0.51	(Intr)
AGE	0.05	-0.98
Residual	0.16	

Fixed effects: ATTIT ~ AGE

	Value	Std.Error	DF	t-value	p-value
(Intercept)	-0.51	0.058	839	-8.8	0
AGE	0.06	0.005	839	13.1	0

Correlation:

(Intr)	
AGE	-0.98

Standardized Within-Group Residuals:

Min	Q1	Med	Q3	Max
-2.879	-0.560	-0.075	0.575	3.456

Number of Observations: 1079

Number of Groups: 239

We can think of “random slopes” as specifying our residual correlation matrix and not think of it directly in multilevel terms.



The correlation matrix for our model

```
> getVarCov(modelRS, type="marginal", individual=3)
ID 3
Marginal variance covariance matrix
      1     2     3     4     5
1 0.040 0.016 0.019 0.021 0.024
2 0.016 0.048 0.026 0.032 0.037
3 0.019 0.026 0.061 0.042 0.050
4 0.021 0.032 0.042 0.079 0.063
5 0.024 0.037 0.050 0.063 0.102
Standard Deviations: 0.2 0.22 0.25 0.28 0.32
```

See how the standard deviations climb? This is because the intercept is at the early time point. As the slopes are different, overall outcomes diverge.



Random Slopes with Heteroskedasticity

```
> modelRS = lme(ATTIT ~ AGE,  
+                 data=nys1,  
+                 random=~AGE|ID,  
+                 weights=varIdent(form=~1|agefac) )
```

```
> summary(modelRS)
```

Linear mixed-effects model fit by REML

Random effects:

Formula: ~AGE | ID

Structure: General positive-definite, Log-Cholesky parametrization
StdDev Corr

(Intercept)	0.577	(Intr)
AGE	0.054	-0.98
Residual	0.141	

Variance function:

Structure: Different standard deviations per stratum

Formula: ~1 | agefac

Parameter estimates:

11	12	13	14	15
1.00	1.20	1.31	1.13	0.98

Fixed effects: ATTIT ~ AGE

	Value	Std.Error	DF	t-value	p-value
(Intercept)	-0.49	0.057	839	-8.6	0
AGE	0.06	0.005	839	13.1	0

Correlation:

(Intr)	
AGE	-0.98



And our covariance

```
> getVarCov(modelRS, type="marginal", individual=3)
ID 9
Marginal variance covariance matrix
      1     2     3     4     5
1 0.049 0.017 0.019 0.021 0.022
2 0.017 0.056 0.026 0.031 0.036
3 0.019 0.026 0.068 0.042 0.049
4 0.021 0.031 0.042 0.081 0.063
5 0.022 0.036 0.049 0.063 0.111
  Standard Deviations: 0.22 0.24 0.26 0.28 0.33
>
```

Now our residual variances along the diagonal are allowed to vary beyond what the random slope model says.



Fully unrestricted

```
> modelUnrestricted = gls(ATTIT ~ AGE,  
+                         data=nys1,  
+                         correlation=corSymm(form=~1|ID),  
+                         weights=varIdent(form=~1|agefac) )
```

Unrestricted covariances

Correlation Structure: General

Formula: ~1 | ID

Parameter estimate(s) :

Correlation:

	1	2	3	4
2	0.46			
3	0.37	0.51		
4	0.44	0.44	0.66	
5	0.47	0.44	0.60	0.76

Variance function:

Structure: Different standard deviations per stratum

Formula: ~1 | agefac

Parameter estimates:

	11	12	13	14	15
1.0	1.1	1.4	1.5	1.6	

Coefficients:

	Value	Std. Error	t-value	p-value
(Intercept)	-0.46	0.055	-8.3	0
AGE	0.06	0.005	13.0	0

Correlation:

	(Intr)
AGE	-0.98

Unrestricted covariances

Heteroskedasticity by time (wave)



Our variance-covariance matrix

```
source('getVarCov2.R')
```

```
getVarCovFixedGls(modelUnrestricted,  
                    type="marginal", individual=3)
```

Marginal variance covariance matrix

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	0.036	0.019	0.019	0.024	0.026
[2,]	0.019	0.045	0.029	0.027	0.028
[3,]	0.019	0.029	0.072	0.052	0.048
[4,]	0.024	0.027	0.052	0.084	0.066
[5,]	0.026	0.028	0.048	0.066	0.088

Standard Deviations: 0.19 0.21 0.27 0.29 0.3

**Jameson (student of mine, past TF) found a bug
in the original code and fixed it. So you should
source this file and use the revised method.**

Comments

- ★ Type “?corClasses” in R for the different types of residual variance-covariances matrices lme can estimate.
- ★ lme doesn’t do non-normal outcomes.
- ★ See
<http://rpsychologist.com/r-guide-longitudinal-lme-lmer>
for some further examples, including alternative polynomial growth model and piecewise growth model specifications.

Recap:
Why might we want these
fancy error structures?



Reason 1: Heteroskedasticity is a thing and we should allow it

HS&B has public and catholic schools.

We might imagine that there is greater variation of students in public schools.

This can raise two concerns:

- ★ Ignoring this makes our model misspecified. We need to fix that to get better inference elsewhere.
- ★ Knowing and characterizing this is a research question in its own right.

Heteroscedasticity at Level 1

Extend our two-level model as:

$$(R_{ij} \mid TX = 0) \sim \mathcal{N}(0, \sigma_0^2)$$

$$(R_{ij} \mid TX = 1) \sim \mathcal{N}(0, \sigma_1^2)$$

lme is your only option here.

The code is:

```
lme(y ~ time * tx,  
     random = ~ time | subjects,  
     weights = varIdent(form= ~ 1 | tx), data=data)
```

Why do this? Think about HSEB and the Public vs Catholic schools. We might believe heterogeneity is larger in the public schools



What is the final model here?

Reason 2: Even after good modeling, we might not believe our residuals are independent

For longitudinal data, our canonical quadratic growth model says:

$$Y_{it} = \pi_{0i} + \pi_{1i}(a_{it} - L) + \pi_{2i}(a_{it} - L)^2 + \epsilon_{it}$$

Assumptions:

- ★ Given an individual's random effects, their residuals are independent of each other
- ★ The residuals are independent of the random intercept and slope
- ★ The residuals are homoskedastic Also can be sketchy.
Homoskedasticity

This one is
a killer!

How to go about Layering Models

Our recommendation:

- ★ Fit your growth curve model
- ★ Layer on more complex entanglements of the residual structure
 - Think of this complexity as a sensitivity check to protect your inference

An alternate approach

- ★ Cluster-robust standard errors
- ★ This approach primarily assumes independence of the clusters from each other.
- ★ Weaker assumptions, but no characterization of individual variation or individual prediction.

Another alternate approach: a hybrid

- ★ First model your multilevel model
- ★ Use cluster-robust standard errors using the associated residuals (this is what STATA's "robust" option does).

Do I need to care?

Check in:
<http://cs179.org/lec63>

This lecture is a first window into tools you may need if you are fitting autoregressive models.

Another option: cluster-robust standard errors ON TOP OF your multilevel model (i.e., use sandwich style estimators from prior lecture).

These advanced specifications make multilevel modeling more robust, i.e., better standard errors.

With longitudinal data in particular, this can be somewhat necessary. But ignoring it is generally relatively safe.

Appendix



Implementing Compound Symmetry with lme() (instead of the gls method from prior)

```
> modelRE = lme(ATTIT ~ AGE,  
+                 data=nys1,  
+                 random=~1|ID )  
>  
> summary(modelRE)
```

Linear mixed-effects model fit by REML

Random effects:

Formula: ~1 | ID
(Intercept) Residual
StdDev: 0.18 0.18

Fixed effects: ATTIT ~ AGE

	Value	Std.Error	DF	t-value	p-value
(Intercept)	-0.51	0.054	839	-9.5	0
AGE	0.06	0.004	839	16.2	0

Correlation:

	(Intr)
AGE	-0.97

Standardized Within-Group Residuals:

Min	Q1	Med	Q3	Max
-2.905	-0.644	-0.014	0.604	3.269

Number of Observations: 1079

Number of Groups: 239

**Within-group residuals: i.e.,
AFTER we take out random
group effects.**



Comparing the two methods: we are indeed fitting the same model

```
> modelCompSymm$logLik  
[1] 106  
> modelRE$logLik  
[1] 106  
  
> summary(modelCompSymm)$AIC  
[1] -205  
> summary(modelRE)$AIC  
[1] -205
```

All we are showing here is
the gls and the lme calls
are doing the same thing.

It is just the syntax that is
different.

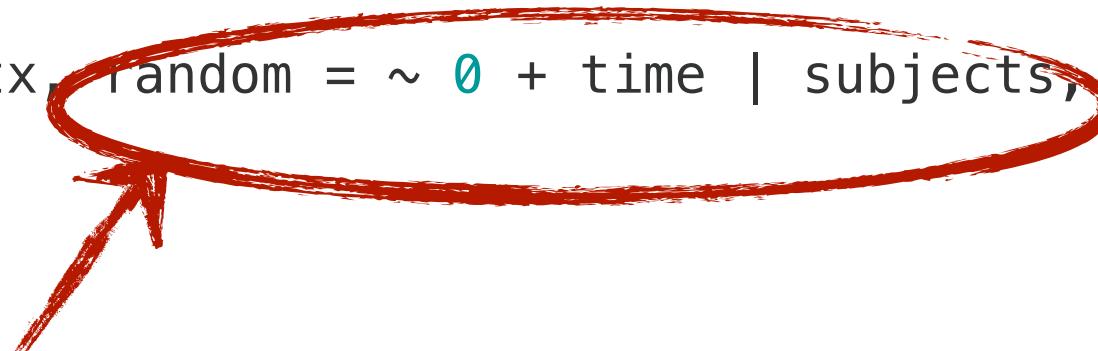
Related but Different Syntax in our two packages

lme4

```
lmer(y ~ time * tx + ( 0 + time | subjects), data=data)
```

nlme

```
lme(y ~ time * tx, random = ~ 0 + time | subjects, data=data)
```



lme() declares random effects separately from the fixed effects (while lmer() puts them all together).

And what is the model?