Example 6a: Modeling Change over Time Using Polynomial Trends
(complete data, syntax, and output available for SAS, STATA, and R electronically)

These data for these example models come from Hoffman (2015) chapter 6. We will be examining change in response time (RT) in milliseconds over six practice sessions (balanced time) to a measure of processing speed in a sample of 101 older adults. We will examine the extent to which individual differences in change in RT can be described by polynomial slopes (current Example 6a), piecewise slopes (Example 6b), “truly” nonlinear exponential models (Example 6c using SAS only), and the use of log-transformed time to approximate a (truly nonlinear) exponential model (Example 6d).

SAS Syntax for Data Import and Manipulation:

```
* Define global variable for file location to be replaced in code below;
%LET filesave=C:\Dropbox\21_PSQF7375_Longitudinal\PSQF7375_Longitudinal_Example6;
* Location for SAS files for these models (uses macro variable filesave);
LIBNAME filesave "&filesave.";

* Import chapter 6 stacked data and create time predictor variables;
DATA work.Example6; SET filesave.SAS_Chapter6;
* Center time predictor for polynomial time models;
  time1=session-1; LABEL time1="time1: Session (0=1)";
* Create session dummy codes for testing means model absolute fit in REML;
  IF session=1 THEN s1=1; ELSE s1=0;
  IF session=2 THEN s2=1; ELSE s2=0;
  IF session=3 THEN s3=1; ELSE s3=0;
  IF session=4 THEN s4=1; ELSE s4=0;
  IF session=5 THEN s5=1; ELSE s5=0;
  IF session=6 THEN s6=1; ELSE s6=0;
RUN;
```

STATA Syntax for Data Import and Manipulation:

```
// Define global variable for file location to be replaced in code below
   global filesave "C:\Dropbox\21_PSQF7375_Longitudinal\PSQF7375_Longitudinal_Example6"

// Import chapter 6 stacked data and create time predictor variables
   use "$filesave\STATA_Chapter6.dta", clear

// Center time predictor for polynomial time models (also need to make quadratic version)
   gen time1=session-1
   gen time1sq=time1*time1
   label variable time1   "time1: Session (0=1)"
   label variable time1sq "time1sq: Quadratic Session (0=1)"

// Create session dummy codes for testing means model absolute fit in REML
   gen s1=0
   gen s2=0
   gen s3=0
   gen s4=0
   gen s5=0
   gen s6=0
   replace s1=1 if session==1
   replace s2=1 if session==2
   replace s3=1 if session==3
   replace s4=1 if session==4
   replace s5=1 if session==5
   replace s6=1 if session==6
```

R Syntax for Data Import and Manipulation:

```
# Define variables for working directory and data name
filesave = "C:\\Dropbox/21_PSQF7375_Longitudinal/PSQF7375_Longitudinal_Example6/"
filename = "SAS_Chapter6.sas7bdat"
setwd(dir=filesave)

# Import chapter 6 stacked data with labels
Example6 = read_sas(data_file=paste0(filesave,filename))
```

```
# Convert to data frame as data frame without labels to use for analysis
Example6 = as.data.frame(Example6)
# Sort data by PersonID (needed for correct RCOV matrix)
Example6 = sort_asc(Example6,PersonID,wave)

# Make new variable for session with reference=6 to match other programs
Example6$session6=relevel(factor(Example6$session), ref=6)

# Center time predictor for polynomial time models
Example6$time1=Example6$session-1

# Create session dummy codes for testing means model absolute fit in REML
Example6$s1=0
Example6$s2=0
Example6$s3=0
Example6$s4=0
Example6$s5=0
Example6$s6=0
Example6$s1[which(Example6$session==1)]=1
Example6$s2[which(Example6$session==2)]=1
Example6$s3[which(Example6$session==3)]=1
Example6$s4[which(Example6$session==4)]=1
Example6$s5[which(Example6$session==5)]=1
Example6$s6[which(Example6$session==6)]=1
```

Model 0: Saturated Means, Unstructured Variance Model (the answer key best baseline model)

This provides the ANSWER KEY for both the model for the means (via saturated means) and the model for the variance (via an unstructured **R** matrix of all possible variances and covariances). This model is only is possible to estimate directly (without rounding occasions) in balanced data. The predicted outcome from the (saturated) fixed effects is given by: $\widehat{rt}_{ti} = \beta_0 + \beta_1(s1_{ti}) + \beta_2(s2_{ti}) + \beta_3(s3_{ti}) + \beta_4(s4_{ti}) + \beta_5(s5_{ti})$, but the unstructured model for the variance cannot be easily summarized by scalar notation.

```
TITLE1 "SAS Ch. 6: 0: Saturated Means, Unstructured Variance Model -- TOTAL ANSWER KEY";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = session / SOLUTION DDFM=Satterthwaite;
     REPEATED session / R RCORR TYPE=UN SUBJECT=PersonID;
     LSMEANS session / DIFF=ALL;
RUN; TITLE1;
```

The variable of *session* will be our categorical indicator for time—it will structure the **R** matrix via the REPEATED and CLASS statements given that we have balanced data. Here, *session* also appears on the MODEL statement so that we can estimate all possible mean differences across sessions (i.e., session is a categorical grouping predictor with 5 slopes).

```
display "STATA Ch 6: 0: Saturated Means, Unstructured Variance Model -- TOTAL ANSWER KEY"
mixed rt ib(last).session, || personid: , noconstant variance reml covariance(unstructured) ///
           residuals(unstructured,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estat ic, n(101)                 // AIC and BIC
      estat wcorrelation, covariance // R matrix
      estat wcorrelation             // RCORR matrix
      contrast i.session, small       // Omnibus F-test for session
      margins  i.session,             // Means per session
      margins  i.session, pwcompare(pveffects) df(100) // Mean diffs by session
```

```
print("R Ch 6: 0: Saturated Means, Unstructured Variance Model in GLS -- TOTAL ANSWER KEY")
SatUN = gls(data=Example6, method="REML", model=rt~1+factor(session6),
            correlation=corSymm(form=~as.numeric(session)|PersonID),
            weights=varIdent(form=~1|session))
print("Show results using incorrect DDF, with total leftover variance")
print("Total variance per occasion is created using SD multiplier")
summary(SatUN); summary(SatUN)$sigma^2
print("Show R and RCORR matrices for first person (R is slightly off)")
R=getVarCov(SatUN, individual="101", type="marginal"); R
RCORR=corMatrix(SatUN$modelStruct$corStruct)[[4]]; RCORR
print("session means and pairwise mean differences with incorrect DDF")
emmeans(ref_grid(SatUN), pairwise~session6, adjust="none") # tried mode="df.error"
print("Error when trying to get Satterthwaite DDF, so had to switch to residual DDF")
lsmeans(SatUN, "session6", mode="df.error")
print("F-test p-value now based on Satterthwaite DDF"); anova(SatUN)
```

SAS Output (relevant output only):

Estimated R Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	**301985**	235659	217994	202607	192154	195360
2	235659	**259150**	230217	213232	202092	193268
3	217994	230217	**233368**	205209	196919	188604
4	202607	213232	205209	**217544**	193676	185321
5	192154	202092	196919	193676	**212098**	187840
6	195360	193268	188604	185321	187840	**196733**

This marginal unstructured **R matrix** (R = V here) allows all the variances and covariances to be estimated.

THIS IS THE PATTERN we are trying to duplicate with our **model for the variance**.

Estimated R Correlation Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	1.0000	0.8424	0.8212	0.7905	0.7593	0.8015
2	0.8424	1.0000	0.9361	0.8981	0.8620	0.8559
3	0.8212	0.9361	1.0000	0.9108	0.8851	0.8802
4	0.7905	0.8981	0.9108	1.0000	0.9016	0.8958
5	0.7593	0.8620	0.8851	0.9016	1.0000	0.9196
6	0.8015	0.8559	0.8802	0.8958	0.9196	1.0000

Information Criteria

Neg2LogLike	**Parms**	AIC	AICC	HQIC	BIC	CAIC
8229.8	**21**	8271.8	8273.4	8294.0	8326.7	8347.7

In REML, only parameters in the model for the variance count. −2LL comparisons in REML must have the same fixed effects in the model for the means.

Solution for Fixed Effects

Effect	Session (1-6)	Estimate	Standard Error	DF	t Value	Pr > |t|	
Intercept		1672.14	44.1345	100	37.89	<.0001	Beta0
Session	1	289.76	32.7000	100	8.86	<.0001	Beta1
Session	2	143.04	26.2031	100	5.46	<.0001	Beta2
Session	3	77.8986	22.8842	100	3.40	0.0010	Beta3
Session	4	45.6604	20.7853	100	2.20	0.0303	Beta4
Session	5	35.0397	18.1168	100	1.93	0.0559	Beta5
Session	6	0	.	.	.	.	

Type 3 Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
Session	5	100	16.72	<.0001

What does this F-test tell us?

Least Squares Means

Effect	Session (1-6)	Estimate	Standard Error	DF	t Value	Pr > |t|	
Session	1	1961.89	54.6805	100	35.88	<.0001	B0+B1
Session	2	1815.17	50.6541	100	35.83	<.0001	B0+B2
Session	3	1750.03	48.0684	100	36.41	<.0001	B0+B3
Session	4	1717.80	46.4101	100	37.01	<.0001	B0+B4
Session	5	1707.18	45.8255	100	37.25	<.0001	B0+B5
Session	6	1672.14	44.1345	100	37.89	<.0001	B0

The **saturated means model** allows all session means to be estimated.

THIS IS THE PATTERN we are trying to duplicate with our **model for the means**.

By what other name do you know this “total answer key” model (i.e., when it is estimated using least squares)?

By what other name do you know this “total answer key” model (i.e., when it is estimated in SEM)?

Model 1b: Empty Means, Random Intercept Model (the worst baseline longitudinal model)

Level 1: $y_{ti} = \beta_{0i} + e_{ti}$

Level 2: Intercept: $\beta_{0i} = \gamma_{00} + U_{0i}$

```
TITLE1 "SAS Ch 6: 1b: Empty Means, Random Intercept Model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt =  / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT / G V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R TYPE=VC SUBJECT=PersonID;
     ODS OUTPUT CovParms=CovEmpty; * Save for pseudo-R2;
RUN; TITLE1;
```

```
display "STATA Ch 6: 1b: Empty Means, Random Intercept Model"
mixed rt , || personid: , variance reml covariance(unstructured) ///
           residuals(independent,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estat ic, n(101)                  // AIC and BIC
      estat icc                         // Intraclass correlation
      estat wcorrelation, covariance // V matrix
      estat wcorrelation                // VCORR matrix
```

```
print("R Ch 6: 1b: Empty Means, Random Intercept Model")
EmptyRI = lmer(data=Example6, REML=TRUE, formula=rt~1+(1|PersonID))
print("Show results using Satterthwaite DDF including -2LL as deviance")
summary(EmptyRI, ddf="Satterthwaite"); llikAIC(EmptyRI, chkREML=FALSE)
print("Get ICC"); icc(EmptyRI)
print("Does random intercept improve model fit?")
ranova(EmptyRI, reduce.term=TRUE) # Remove random intercept
```

SAS Output:

Estimated R Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	44900					
2		44900				
3			44900			
4				44900		
5					44900	
6						44900

This **level-1 R matrix VC structure** (equal variance over time, no covariance of any kind) will be used repeatedly as we add fixed and random effects of time to the model.

Estimated G Matrix

Row	Effect	PersonID	Col1
1	Intercept	101	200883

This **level-2 G matrix** (always **unstructured**) currently holding only the random intercept variance will be expanded as we add random effects.

Estimated V Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	**245783**	200883	200883	200883	200883	200883
2	200883	**245783**	200883	200883	200883	200883
3	200883	200883	**245783**	200883	200883	200883
4	200883	200883	200883	**245783**	200883	200883
5	200883	200883	200883	200883	**245783**	200883
6	200883	200883	200883	200883	200883	**245783**

This random intercept only model for the variance predicts a compound symmetry pattern for the marginal **V** matrix (equal variances, equal pairwise covariances over time).

Estimated V Correlation Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	1.0000	0.8173	0.8173	0.8173	0.8173	0.8173
2	0.8173	1.0000	0.8173	0.8173	0.8173	0.8173
3	0.8173	0.8173	1.0000	0.8173	0.8173	0.8173
4	0.8173	0.8173	0.8173	1.0000	0.8173	0.8173
5	0.8173	0.8173	0.8173	0.8173	1.0000	0.8173
6	0.8173	0.8173	0.8173	0.8173	0.8173	1.0000

This random intercept only model for the variance (CS) translates into equal correlation over time as shown in the VCORR matrix (as the ICC).

```
                 Covariance Parameter Estimates
                                    Standard         Z
Cov Parm     Subject     Estimate      Error     Value      Pr > Z
UN(1,1)      PersonID      200883      29471      6.82       <.0001  Level-2 random intercept variance of U0i
Session      PersonID       44900    2825.63     15.89       <.0001  Level-1 residual variance of eti

  Null Model Likelihood Ratio Test
    DF    Chi-Square      Pr > ChiSq
     1        691.74          <.0001
```

This is the LRT of whether we need the random intercept variance: it compares our random intercept model to the nested e-only model (1a in chapter 6). Thus, it is a test of whether the ICC is significantly > 0.

```
                                 Information Criteria
Neg2LogLike       Parms       AIC        AICC        HQIC         BIC        CAIC
     8536.9           2    8540.9      8540.9      8543.0      8546.1      8548.1
```

Only variance model parameters are "counted" as parms using REML.

```
                   Solution for Fixed Effects
                             Standard
Effect        Estimate          Error       DF    t Value     Pr > |t|
Intercept      1770.70        45.4206      100      38.98       <.0001   gamma00
```

This empty model for the means predicts no change on average (RT across six sessions is constant at 1770).

Model 2a: Fixed Linear Time, Random Intercept Model

$$\text{Level 1: } y_{ti} = \beta_{0i} + \beta_{1i}\left(\text{Time}_{ti}\right) + e_{ti}$$

$$\text{Level 2: Intercept: } \beta_{0i} = \gamma_{00} + U_{0i}$$

$$\text{Linear Time: } \beta_{1i} = \gamma_{10}$$

```
TITLE1 "SAS Ch 6: 2a: Fixed Linear Time, Random Intercept Model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT / G V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R TYPE=VC SUBJECT=PersonID;
     ODS OUTPUT CovParms=CovFixLin InfoCrit=FitFixLin; * Save for pseudo-R2 and LRT;
     * Get conditional mean per occasion from value of time predictor;
     ESTIMATE "Intercept at Session=1 Time=0"    intercept 1 time1 0;
     ESTIMATE "Intercept at Session=2 Time=1"    intercept 1 time1 1;
     ESTIMATE "Intercept at Session=3 Time=2"    intercept 1 time1 2;
     ESTIMATE "Intercept at Session=4 Time=3"    intercept 1 time1 3;
     ESTIMATE "Intercept at Session=5 Time=4"    intercept 1 time1 4;
     ESTIMATE "Intercept at Session=6 Time=5"    intercept 1 time1 5;
RUN;
TITLE1 "Calculate pseudo R2 -- variance accounted for by fixed linear time";
%PseudoR2(NCov=2, CovFewer=CovEmpty, CovMore=CovFixLin); TITLE1;
```

```
display "STATA Ch 6: 2a: Fixed Linear Time, Random Intercept Model"
mixed rt c.time1, || personid: , variance reml covariance(unstructured) ///
           residuals(independent,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estat ic, n(101)              // AIC and BIC
      estat wcorrelation, covariance // V matrix
      estat wcorrelation            // VCORR matrix
      // Get conditional mean per occasion from value of time predictor
      lincom _cons*1 + c.time1*0    // Intercept at Session=1 Time=0
      lincom _cons*1 + c.time1*1    // Intercept at Session=2 Time=1
      lincom _cons*1 + c.time1*2    // Intercept at Session=3 Time=2
      lincom _cons*1 + c.time1*3    // Intercept at Session=4 Time=3
      lincom _cons*1 + c.time1*4    // Intercept at Session=5 Time=4
      lincom _cons*1 + c.time1*5    // Intercept at Session=6 Time=5
      estimates store FitFixLin     // Save for LRT
```

```
print("R Ch 6 2a: Fixed Linear Time, Random Intercept Model")
FixLin = lmer(data=Example6, REML=TRUE, formula=rt~1+time1+(1|PersonID))
print("Show results using Satterthwaite DDF including -2LL as deviance")
summary(FixLin, ddf="Satterthwaite"); llikAIC(FixLin, chkREML=FALSE)
```

```
print("Get conditional mean per occasion from value of time predictor")
print("Intercept at Session=1 Time=0"); contest1D(FixLin, ddf="Satterthwaite", L=c(1,0))
print("Intercept at Session=2 Time=1"); contest1D(FixLin, ddf="Satterthwaite", L=c(1,1))
print("Intercept at Session=3 Time=2"); contest1D(FixLin, ddf="Satterthwaite", L=c(1,2))
print("Intercept at Session=4 Time=3"); contest1D(FixLin, ddf="Satterthwaite", L=c(1,3))
print("Intercept at Session=5 Time=4"); contest1D(FixLin, ddf="Satterthwaite", L=c(1,4))
print("Intercept at Session=6 Time=5"); contest1D(FixLin, ddf="Satterthwaite", L=c(1,5))
```

SAS Output:

Estimated R Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	35662					
2		35662				
3			35662			
4				35662		
5					35662	
6						35662

Estimated G Matrix

Row	Effect	PersonID	Col1
1	Intercept	101	202422

Estimated V Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	238084	202422	202422	202422	202422	202422
2	202422	238084	202422	202422	202422	202422
3	202422	202422	238084	202422	202422	202422
4	202422	202422	202422	238084	202422	202422
5	202422	202422	202422	202422	238084	202422
6	202422	202422	202422	202422	202422	238084

The marginal **V** matrix still has a compound symmetry pattern because we have not yet added to the model for the variance (still only a random intercept variance in **G**).

Estimated V Correlation Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	1.0000	0.8502	0.8502	0.8502	0.8502	0.8502
2	0.8502	1.0000	0.8502	0.8502	0.8502	0.8502
3	0.8502	0.8502	1.0000	0.8502	0.8502	0.8502
4	0.8502	0.8502	0.8502	1.0000	0.8502	0.8502
5	0.8502	0.8502	0.8502	0.8502	1.0000	0.8502
6	0.8502	0.8502	0.8502	0.8502	0.8502	1.0000

VCORR now provides a "conditional" ICC after controlling for fixed linear time. If the empty means model ever returns ICC=0, report this conditional ICC instead.

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr > Z	
UN(1,1)	PersonID	202422	29470	6.87	<.0001	Level-2 random intercept variance of U_{0i}
Session	PersonID	35662	2246.48	15.87	<.0001	Level-1 residual variance of e_{ti}

Null Model Likelihood Ratio Test

DF	Chi-Square	Pr > ChiSq
1	787.61	<.0001

This LRT is for the conditional ICC > 0.

Information Criteria

Neg2LogLike	Parms	AIC	AICC	HQIC	BIC	CAIC
8414.7	2	8418.7	8418.7	8420.8	8423.9	8425.9

Only variance model parameters are "counted" as parms using REML.

Solution for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|	
Intercept	1899.63	46.7882	113	40.60	<.0001	gamma00
time1	-51.5719	4.4918	504	-11.48	<.0001	gamma10

Is the fixed linear time slope significant? How do we know?

Estimates

Label	Estimate	Standard Error	DF	t Value	Pr > |t|	int + slope(time)
Intercept at Session=1 Time=0	1899.63	46.7882	113	40.60	<.0001	gamma00 + gamma10(0)
Intercept at Session=2 Time=1	1848.06	45.9176	104	40.25	<.0001	gamma00 + gamma10(1)
Intercept at Session=3 Time=2	1796.49	45.4761	100	39.50	<.0001	gamma00 + gamma10(2)
Intercept at Session=4 Time=3	1744.92	45.4761	100	38.37	<.0001	gamma00 + gamma10(3)
Intercept at Session=5 Time=4	1693.34	45.9176	104	36.88	<.0001	gamma00 + gamma10(4)
Intercept at Session=6 Time=5	1641.77	46.7882	113	35.09	<.0001	gamma00 + gamma10(5)

What kind of average change does this model predict?

What kind of individual differences in change does this model predict?

Calculate pseudo R2 -- variance accounted for by fixed linear time
PsuedoR2 (% Reduction) for CovEmpty vs. CovFixLin

Name	CovParm	Subject	Estimate	StdErr	ZValue	ProbZ	PseudoR2
CovEmpty	UN(1,1)	PersonID	200883	29471	6.82	<.0001	.
CovEmpty	session	PersonID	44900	2825.63	15.89	<.0001	.
CovFixLin	UN(1,1)	PersonID	202422	29470	6.87	<.0001	-0.00766
CovFixLin	session	PersonID	35662	2246.48	15.87	<.0001	0.20575

Which variance did the fixed linear time slope explain?

Why did the other variance increase instead (i.e., have a negative pseudo-R^2 value)?

Model 2b: Random Linear Time Model

$$\text{Level 1: } y_{ti} = \beta_{0i} + \beta_{1i}\left(\text{Time}_{ti}\right) + e_{ti}$$

$$\text{Level 2: Intercept: } \beta_{0i} = \gamma_{00} + U_{0i}$$

$$\text{Linear Time: } \beta_{1i} = \gamma_{10} + U_{1i}$$

```
TITLE1 "SAS Ch 6: 2b: Random Linear Time Model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT time1 / G GCORR V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R TYPE=VC SUBJECT=PersonID;
     ODS OUTPUT CovParms=CovRandLin InfoCrit=FitRandLin; * Save for pseudo-R2 and LRT;
RUN;
TITLE1 "Calculate LRT -- does random linear time slope improve model fit?";
%FitTest(FitFewer=FitFixLin, FitMore=FitRandLin); TITLE1;
```

```
display "STATA Ch 6: 2b: Random Linear Time Model"
mixed rt c.time1, || personid: time1, variance reml covariance(unstructured) ///
          residuals(independent,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estat ic, n(101)                                 // AIC and BIC
      estat recovariance, relevel(personid)            // G matrix
      estat recovariance, relevel(personid) correlation // GCORR matrix
      estat wcorrelation, covariance                   // V matrix
      estat wcorrelation                               // VCORR matrix
      estimates store FitRandLin     // Save for LRT
      lrtest FitRandLin FitFixLin    // Does random linear time slope improve model fit?
```

```
print("R Ch 6 2b: Random Linear Time Model")
RandLin = lmer(data=Example6, REML=TRUE, formula=rt~1+time1+(1+time1|PersonID))
print("Show results using Satterthwaite DDF including -2LL as deviance")
summary(RandLin, ddf="Satterthwaite"); llikAIC(RandLin, chkREML=FALSE)
print("Does random linear time slope improve model fit?")
ranova(RandLin, reduce.term=TRUE) # Remove random slope and covariance
```

SAS Output:

Estimated R Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	27905					
2		27905				
3			27905			
4				27905		
5					27905	
6						27905

Estimated G Matrix

Row	Effect	PersonID	Col1	Col2
1	Intercept	101	253258	-12701
2	time1	101	**-12701**	**2233.83**

Estimated G Correlation Matrix

Row	Effect	PersonID	Col1	Col2
1	Intercept	101	1.0000	-0.5340
2	time1	101	**-0.5340**	1.0000

This **level-2 G matrix** (always **unstructured**) still contains a random intercept variance (1,1), and we have added a random linear time slope variance (2,2), as well as a covariance between the random intercept and random linear time slope for the same person (2,1). The intercept–time slope correlation is shown in GCORR (2,1).

Estimated V Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	**281163**	240557	227856	215155	202455	189754
2	240557	**257995**	219623	209156	198689	188222
3	227856	219623	**239295**	203157	194924	186691
4	215155	209156	203157	**225063**	191158	185159
5	202455	198689	194924	191158	**215298**	183627
6	189754	188222	186691	185159	183627	**210001**

The marginal **V** matrix now predicts that **variance changes quadratically over time** (as a function of time^2).

How the V matrix variances and covariances get calculated in a random linear time model:

$$\mathbf{V}_i \text{ matrix: Variance}\left[y_{time}\right] = \tau^2_{U_0} + \left[(\text{Session}-1)^2\,\tau^2_{U_1}\right] + \left[2(\text{Session}-1)\,\tau_{U_{01}}\right] + \sigma^2_e$$

$$\mathbf{V}_i \text{ matrix: Covariance}\left[y_A, y_B\right] = \tau^2_{U_0} + \left[(A+B)\,\tau_{U_{01}}\right] + \left[(AB)\,\tau^2_{U_1}\right]$$

Estimated V Correlation Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	1.0000	0.8932	0.8784	0.8553	0.8229	0.7809
2	0.8932	1.0000	0.8839	0.8680	0.8430	0.8086
3	0.8784	0.8839	1.0000	0.8754	0.8588	0.8328
4	0.8553	0.8680	0.8754	1.0000	0.8684	0.8517
5	0.8229	0.8430	0.8588	0.8684	1.0000	0.8636
6	0.7809	0.8086	0.8328	0.8517	0.8636	1.0000

The marginal **VCORR** matrix now predicts that the covariance changes over time as well (in a time-dependent pattern).

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr Z	
UN(1,1)	PersonID	253258	37897	6.68	<.0001	Level-2 random intercept variance of U_{0i}
UN(2,1)	PersonID	-12701	3621.98	-3.51	0.0005	Level-2 random intercept-linear covariance
UN(2,2)	PersonID	2233.83	552.92	4.04	<.0001	Level-2 random linear time slope variance of U_{1i}
Session	PersonID	27905	1963.42	14.21	<.0001	Level-1 residual variance of e_{ti}

Null Model Likelihood Ratio Test

DF	Chi-Square	Pr > ChiSq
3	830.20	<.0001

This LRT tells us whether we need the full 2x2 **G** matrix (so it is not helpful).

Information Criteria

Neg2LogLike	**Parms**	AIC	AICC	HQIC	BIC	CAIC
8372.1	**4**	8380.1	8380.2	8384.3	8390.6	8394.6

Only variance model parameters are "counted" as parms using REML.

```
                   Solution for Fixed Effects
                        Standard
Effect       Estimate      Error      DF    t Value    Pr > |t|
Intercept     1899.63    51.4998     100      36.89      <.0001   gamma00
time1        -51.5719     6.1567     100      -8.38      <.0001   gamma10

Calculate LRT -- does random linear time slope improve model fit?
Likelihood Ratio Test for FitFixLin vs. FitRandLin
             Neg2Log
   Name         Like      Parms        AIC        BIC    DevDiff    DFdiff       Pvalue
FitFixLin     8414.7        2       8418.7     8423.9       .          .              .
FitRandLin    8372.1        4       8380.1     8390.6    42.5856       2      5.6579E-10
```

Is the random linear time slope significant? How do we know?

Do we need to compute pseudo-R^2 for this random linear time slope model relative to the previous fixed linear time slope, random intercept model?

<u>95% Random Effects Confidence Intervals that describe the *predicted* range of *individual* random effects:</u>

$$\text{Random Effect 95\% CI} = \text{fixed effect} \pm \left(1.96*\sqrt{\text{Random Variance}}\right)$$

$$\text{Intercept 95\% CI} = \gamma_{00} \pm \left(1.96*\sqrt{\tau^2_{U_0}}\right) \rightarrow 1{,}899.63 \pm \left(1.96*\sqrt{253{,}258}\right) = 913 \text{ to } 2{,}886$$

$$\text{Linear Time Slope 95\% CI} = \gamma_{10} \pm \left(1.96*\sqrt{\tau^2_{U_1}}\right) \rightarrow -51.57 \pm \left(1.96*\sqrt{2{,}233.83}\right) = -145 \text{ to } 42$$

Is it a problem that the CI for the linear time slope overlaps 0? What does this mean?

Model 3a: Fixed Quadratic, Random Linear Time Model

$$\text{Level 1: } y_{ti} = \beta_{0i} + \beta_{1i}\left(\text{Time}_{ti}\right) + \beta_{2i}\left(\text{Time}_{ti}\right)^2 + e_{ti}$$

$$\text{Level 2: Intercept: } \beta_{0i} = \gamma_{00} + U_{0i}$$

$$\text{Linear Time: } \beta_{1i} = \gamma_{10} + U_{1i}$$

$$\text{Quadratic Time: } \beta_{2i} = \gamma_{20}$$

Predicted intercept at any occasion:

$$= \gamma_{00} + \gamma_{10}(\text{Time}_{ti}) + \gamma_{20}(\text{Time}_{ti})^2$$

Instantaneous linear time slope at any occasion:

$$= \gamma_{10} + 2\gamma_{20}(\text{Time}_{ti})$$

```
TITLE1 "SAS Ch 6: 3a: Fixed Quadratic, Random Linear Time Model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 time1*time1 / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT time1 / G GCORR V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R TYPE=VC SUBJECT=PersonID;
     ODS OUTPUT CovParms=CovFixQuad InfoCrit=FitFixQuad; * Save for pseudo-R2 and LRT;
     * Get conditional mean per occasion from values of time predictors;
     ESTIMATE "Intercept at Session=1 Time=0"    intercept 1 time1 0 time1*time1 0;
     ESTIMATE "Intercept at Session=2 Time=1"    intercept 1 time1 1 time1*time1 1;
     ESTIMATE "Intercept at Session=3 Time=2"    intercept 1 time1 2 time1*time1 4;
     ESTIMATE "Intercept at Session=4 Time=3"    intercept 1 time1 3 time1*time1 9;
     ESTIMATE "Intercept at Session=5 Time=4"    intercept 1 time1 4 time1*time1 16;
     ESTIMATE "Intercept at Session=6 Time=5"    intercept 1 time1 5 time1*time1 25;
```

```
     * Get instantaneous linear slope per occasion from 2*value of time predictor;
     ESTIMATE "Linear Slope at Session=1 Time=0" time1 1 time1*time1 0;
     ESTIMATE "Linear Slope at Session=2 Time=1" time1 1 time1*time1 2;
     ESTIMATE "Linear Slope at Session=3 Time=2" time1 1 time1*time1 4;
     ESTIMATE "Linear Slope at Session=4 Time=3" time1 1 time1*time1 6;
     ESTIMATE "Linear Slope at Session=5 Time=4" time1 1 time1*time1 8;
     ESTIMATE "Linear Slope at Session=6 Time=5" time1 1 time1*time1 10;
RUN;
TITLE1 "Calculate pseudo R2 -- variance accounted for by fixed quadratic time";
%PseudoR2(NCov=4, CovFewer=CovRandLin, CovMore=CovFixQuad); TITLE1;
```

Because twice the quadratic slope is how the linear slope changes per unit time, the value for time1 used in estimating the linear slope per session gets multiplied by 2.

```
display "STATA Ch 6: 3a: Fixed Quadratic, Random Linear Time Model"
mixed rt c.time1 c.time1#c.time1, || personid: time1, variance reml covariance(unstructured) ///
           residuals(independent,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estat ic, n(101)                                   // AIC and BIC
      estat recovariance, relevel(personid)              // G matrix
      estat recovariance, relevel(personid) correlation // GCORR matrix
      estat wcorrelation, covariance                     // V matrix
      estat wcorrelation                                 // VCORR matrix
      // Get conditional mean per occasion from values of time predictors
      lincom _cons*1 + c.time1*0 + c.time1#c.time1*0     // Intercept at Session=1 Time=0
      lincom _cons*1 + c.time1*1 + c.time1#c.time1*1     // Intercept at Session=2 Time=1
      lincom _cons*1 + c.time1*2 + c.time1#c.time1*4     // Intercept at Session=3 Time=2
      lincom _cons*1 + c.time1*3 + c.time1#c.time1*9     // Intercept at Session=4 Time=3
      lincom _cons*1 + c.time1*4 + c.time1#c.time1*16    // Intercept at Session=5 Time=4
      lincom _cons*1 + c.time1*5 + c.time1#c.time1*25    // Intercept at Session=6 Time=5
      // Get instantaneous linear slope per occasion from 2*value of time predictor
      lincom c.time1*1 + c.time1#c.time1*0,  small    // Linear Slope at Session=1 Time=0
      lincom c.time1*1 + c.time1#c.time1*2,  small    // Linear Slope at Session=2 Time=1
      lincom c.time1*1 + c.time1#c.time1*4,  small    // Linear Slope at Session=3 Time=2
      lincom c.time1*1 + c.time1#c.time1*6,  small    // Linear Slope at Session=4 Time=3
      lincom c.time1*1 + c.time1#c.time1*8,  small    // Linear Slope at Session=5 Time=4
      lincom c.time1*1 + c.time1#c.time1*10, small    // Linear Slope at Session=6 Time=5
      estimates store FitFixQuad    // Save for LRT
```

```
print("R Ch 6 3a: Fixed Quadratic, Random Linear Time Model")
FixQuad = lmer(data=Example6, REML=TRUE, formula=rt~1+time1+I(time1^2)+(1+time1|PersonID))
print("Show results using Satterthwaite DDF including -2LL as deviance")
summary(FixQuad, ddf="Satterthwaite"); llikAIC(FixQuad, chkREML=FALSE)
print("Get conditional mean per occasion from values of time predictors")
print("Intercept at Session=1 Time=0"); contest1D(FixQuad, ddf="Satterthwaite", L=c(1,0,0))
print("Intercept at Session=2 Time=1"); contest1D(FixQuad, ddf="Satterthwaite", L=c(1,1,1))
print("Intercept at Session=3 Time=2"); contest1D(FixQuad, ddf="Satterthwaite", L=c(1,2,4))
print("Intercept at Session=4 Time=3"); contest1D(FixQuad, ddf="Satterthwaite", L=c(1,3,9))
print("Intercept at Session=5 Time=4"); contest1D(FixQuad, ddf="Satterthwaite", L=c(1,4,16))
print("Intercept at Session=6 Time=5"); contest1D(FixQuad, ddf="Satterthwaite", L=c(1,5,25))
print("Get instantaneous linear slope per occasion from 2*value of time predictor")
print("Linear Slope at Session=1 Time=0"); contest1D(FixQuad, ddf="Satterthwaite", L=c(0,1,0))
print("Linear Slope at Session=2 Time=1"); contest1D(FixQuad, ddf="Satterthwaite", L=c(0,1,2))
print("Linear Slope at Session=3 Time=2"); contest1D(FixQuad, ddf="Satterthwaite", L=c(0,1,4))
print("Linear Slope at Session=4 Time=3"); contest1D(FixQuad, ddf="Satterthwaite", L=c(0,1,6))
print("Linear Slope at Session=5 Time=4"); contest1D(FixQuad, ddf="Satterthwaite", L=c(0,1,8))
print("Linear Slope at Session=6 Time=5"); contest1D(FixQuad, ddf="Satterthwaite", L=c(0,1,10))
```

SAS Output:

Estimated R Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	26176					
2		26176				
3			26176			
4				26176		
5					26176	
6						26176

Estimated G Matrix

Row	Effect	PersonID	Col1	Col2
1	Intercept	101	254164	-12948
2	time1	101	-12948	2332.67

The model for the variance has not changed, so it has the same structure as before (**G** is 2x2).

Estimated G Correlation Matrix

Row	Effect	PersonID	Col1	Col2
1	Intercept	101	1.0000	-0.5318
2	time1	101	-0.5318	1.0000

Estimated V Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	**280339**	241216	228268	215320	202372	189424
2	241216	**256776**	219985	209370	198755	188140
3	228268	219985	**237879**	203420	195138	186855
4	215320	209370	203420	**223646**	191521	185571
5	202372	198755	195138	191521	**214079**	184286
6	189424	188140	186855	185571	184286	**209178**

Estimated V Correlation Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	1.0000	0.8991	0.8839	0.8599	0.8261	0.7822
2	0.8991	1.0000	0.8901	0.8737	0.8477	0.8118
3	0.8839	0.8901	1.0000	0.8819	0.8647	0.8377
4	0.8599	0.8737	0.8819	1.0000	0.8753	0.8580
5	0.8261	0.8477	0.8647	0.8753	1.0000	0.8709
6	0.7822	0.8118	0.8377	0.8580	0.8709	1.0000

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr Z	
UN(1,1)	PersonID	254164	37896	6.71	<.0001	Level-2 random intercept variance of U_{0i}
UN(2,1)	PersonID	-12948	3620.70	-3.58	0.0003	Level-2 random intercept-linear covariance
UN(2,2)	PersonID	2332.67	551.58	4.23	<.0001	Level-2 random linear time slope variance of U_{1i}
Session	PersonID	26176	1844.01	14.20	<.0001	Level-1 residual variance of e_{ti}

Null Model Likelihood Ratio Test

DF	Chi-Square	Pr > ChiSq
3	851.78	<.0001

This LRT tells us whether we need the full 2x2 **G** matrix (so it is not helpful).

Information Criteria

Neg2LogLike	**Parms**	AIC	AICC	HQIC	BIC	CAIC
8341.5	**4**	8349.5	8349.5	8353.7	8359.9	8363.9

Is the fixed quadratic time slope significant? How do we know?

Solution for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|	
Intercept	1945.85	52.2433	106	37.25	<.0001	gamma00
time1	-120.90	14.5415	502	-8.31	<.0001	gamma10
time1*time1	13.8656	2.6348	403	5.26	<.0001	gamma20

Estimates

Label	Estimate	Standard Error	DF	t Value	Pr > |t|	**g = gamma fixed effect**
Intercept at Session=1 Time=0	1945.85	52.2433	106	37.25	<.0001	g00(1)+ g10(0)+ g20(0)
Intercept at Session=2 Time=1	1838.82	48.6084	100	37.83	<.0001	g00(1)+ g10(1)+ g20(1)
Intercept at Session=3 Time=2	1759.51	46.8223	105	37.58	<.0001	g00(1)+ g10(2)+ g20(4)
Intercept at Session=4 Time=3	1707.94	45.2925	105	37.71	<.0001	g00(1)+ g10(3)+ g20(9)
Intercept at Session=5 Time=4	1684.10	44.0458	100	38.24	<.0001	g00(1)+ g10(4)+ g20(16)
Intercept at Session=6 Time=5	1687.99	44.9976	108	37.51	<.0001	g00(1)+ g10(5)+ g20(25)

```
Linear Slope at Session=1 Time=0     -120.90     14.5415   502      -8.31     <.0001  g10(1) + 2*g20(0)
Linear Slope at Session=2 Time=1    -93.1687     10.0191   419      -9.30     <.0001  g10(1) + 2*g20(2)
Linear Slope at Session=3 Time=2    -65.4375      6.6968   139      -9.77     <.0001  g10(1) + 2*g20(4)
Linear Slope at Session=4 Time=3    -37.7062      6.6968   139      -5.63     <.0001  g10(1) + 2*g20(6)
Linear Slope at Session=5 Time=4     -9.9750     10.0191   419      -1.00     0.3200  g10(1) + 2*g20(8)
Linear Slope at Session=6 Time=5     17.7562     14.5415   502       1.22     0.2226  g10(1) + 2*g20(10)
```

What happens to the linear time slope over sessions because of the quadratic time slope (i.e., which of the four kinds of interactions is this quadratic effect of time)?

Why did the fixed linear time slope change from the previous model (−52 versus –120)?

Calculate pseudo R2 -- variance accounted for by fixed quadratic time
PsuedoR2 (% Reduction) for CovRandLin vs. CovFixQuad

Name	CovParm	Subject	Estimate	StdErr	ZValue	ProbZ	PseudoR2
CovRandLin	UN(1,1)	PersonID	253258	37897	6.68	<.0001	.
CovRandLin	UN(2,2)	PersonID	2233.83	552.92	4.04	<.0001	.
CovRandLin	session	PersonID	27905	1963.42	14.21	<.0001	.
CovFixQuad	UN(1,1)	PersonID	254164	37896	6.71	<.0001	-0.003577
CovFixQuad	UN(2,2)	PersonID	2332.67	551.58	4.23	<.0001	-0.044244
CovFixQuad	session	PersonID	26176	1844.01	14.20	<.0001	0.061980

Which variance did the fixed quadratic time slope explain? Why did the other variances increase?

Model 3b: Random Quadratic Time Model

Level 1: $y_{ti} = \beta_{0i} + \beta_{1i}\left(\text{Time}_{ti}\right) + \beta_{2i}\left(\text{Time}_{ti}\right)^2 + e_{ti}$

Level 2: Intercept: $\beta_{0i} = \gamma_{00} + U_{0i}$

Linear Time: $\beta_{1i} = \gamma_{10} + U_{1i}$

Quadratic Time: $\beta_{2i} = \gamma_{20} + U_{2i}$

Predicted intercept at any occasion:
$= \gamma_{00} + \gamma_{10}(\text{Time}_{ti}) + \gamma_{20}(\text{Time}_{ti})^2$

Instantaneous linear time slope at any occasion:
$= \gamma_{10} + 2\gamma_{20}(\text{Time}_{ti})$

```
TITLE1 "SAS Ch 6: 3b: Random Quadratic Time Model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 time1*time1 / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT time1 time1*time1 / G GCORR V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R TYPE=VC SUBJECT=PersonID;
     ODS OUTPUT InfoCrit=FitRandQuad; * Save for LRT;
RUN;
TITLE1 "Calculate LRT -- does random quadratic time slope improve model fit?";
%FitTest(FitFewer=FitFixQuad, FitMore=FitRandQuad); TITLE1;

display "STATA Ch 6: 3b: Random Quadratic Time Model"
mixed rt c.time1 c.time1#c.time1, || personid: time1 time1sq, variance reml covariance(unstructured) ///
          residuals(independent,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estat ic, n(101)                                      // AIC and BIC
      estat recovariance, relevel(personid)                 // G matrix
      estat recovariance, relevel(personid) correlation     // GCORR matrix
      estat wcorrelation, covariance                        // V matrix
      estat wcorrelation                                    // VCORR matrix
      estimates store FitRandQuad     // Save for LRT
      lrtest FitRandQuad FitFixQuad   // Does random quadratic time slope improve fit?

print("R Ch 6 3b: Random Quadratic Time Model -- reports convergence problem")
RandQuad = lmer(data=Example6, REML=TRUE, formula=rt~1+time1+I(time1^2)+(1+time1+I(time1^2)|PersonID))
print("Show results using Satterthwaite DDF including -2LL as deviance")
summary(RandQuad, ddf="Satterthwaite"); llikAIC(RandQuad, chkREML=FALSE)
```

SAS Output:

Estimated R Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	20298					
2		20298				
3			20298			
4				20298		
5					20298	
6						20298

Estimated G Matrix

Row	Effect	PersonID	Col1	Col2	Col3
1	Intercept	101	276206	-35734	3901.96
2	time1	101	-35734	25840	-3903.32
3	time1*time1	101	**3901.96**	**-3903.32**	**634.47**

Estimated G Correlation Matrix

Row	Effect	PersonID	Col1	Col2	Col3
1	Intercept	101	1.0000	-0.4230	0.2948
2	time1	101	-0.4230	1.0000	-0.9640
3	time1*time1	101	**0.2948**	**-0.9640**	1.0000

The **level-2 G matrix** (always **unstructured**) still contains variances for the random intercept (1,1) and random linear time slope (2,2), as well as their covariance (2,1). We have now added a variance for the random quadratic time slope (3,3) and its covariances with the random intercept (3,1) and random linear time slope (3,2) for the same person. **GCORR** provides the corresponding correlations among the random effects.

Estimated V Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	**296504**	244374	220346	204122	195702	195085
2	244374	**251508**	219312	208680	199315	191215
3	220346	219312	**235842**	209043	199808	187840
4	204122	208680	209043	**225508**	197182	184958
5	195702	199315	199808	197182	**211735**	182571
6	195085	191215	187840	184958	182571	**200977**

The marginal **V** matrix now predicts the variances to change in a **quartic pattern (time^4)**.

How the V matrix variances and covariances get calculated in a random quadratic time model:

Predicted Variance at Time *T*:

$$\text{Var}(y_T) = \sigma_e^2 + \tau_{U_0}^2 + 2*T*\tau_{U_{01}} + T^2*\tau_{U_1}^2 + 2*T^2*\tau_{U_{02}} + 2*T^3*\tau_{U_{12}} + T^4*\tau_{U_2}^2$$

Predicted Covariance between Time *A* and *B*:

$$\text{Cov}(y_A, y_B) = \tau_{U_0}^2 + (A+B)*\tau_{U_{01}} + (AB)*\tau_{U_1}^2 + (A^2+B^2)*\tau_{U_{02}} + (AB^2)+(A^2B)*\tau_{U_{12}} + (A^2B^2)*\tau_{U_2}^2$$

Estimated V Correlation Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	1.0000	0.8949	0.8333	0.7894	0.7811	0.7992
2	0.8949	1.0000	0.9005	0.8762	0.8637	0.8505
3	0.8333	0.9005	1.0000	0.9064	0.8941	0.8628
4	0.7894	0.8762	0.9064	1.0000	0.9024	0.8688
5	0.7811	0.8637	0.8941	0.9024	1.0000	0.8850
6	0.7992	0.8505	0.8628	0.8688	0.8850	1.0000

The marginal **VCORR** matrix now predicts that the covariance changes over time in a more complex time-dependent pattern than for random linear time.

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr Z	
UN(1,1)	PersonID	276206	41442	6.66	<.0001	Level-2 random intercept variance of U_{0i}
UN(2,1)	PersonID	-35734	11941	-2.99	0.0028	Level-2 random intercept-linear slope covariance
UN(2,2)	PersonID	25840	5864.41	4.41	<.0001	Level-2 random linear time slope variance of U_{1i}
UN(3,1)	PersonID	3901.96	1949.06	2.00	0.0453	Level-2 random intercept-quad slope covariance
UN(3,2)	PersonID	-3903.32	982.61	-3.97	<.0001	Level-2 random linear-quad slope covariance
UN(3,3)	PersonID	634.47	172.37	3.68	0.0001	Level-2 random quadratic time slope variance of U_{2i}
Session	PersonID	20298	1649.11	12.31	<.0001	Level-1 residual variance of e_{ti}

Null Model Likelihood Ratio Test

DF	Chi-Square	Pr > ChiSq
6	890.51	<.0001

This LRT tells us whether we need the full 3x3 **G** matrix (so it is not helpful).

Information Criteria

Neg2LogLike	Parms	AIC	AICC	HQIC	BIC	CAIC
8302.7	**7**	8316.7	8316.9	8324.2	8335.1	8342.1

Solution for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|	
Intercept	1945.85	53.8497	100	36.13	<.0001	gamma00
time1	-120.90	20.0476	100	-6.03	<.0001	gamma10
time1*time1	13.8656	3.4154	100	4.06	<.0001	gamma20

Calculate LRT -- does random quadratic time slope improve model fit?
Likelihood Ratio Test for FitFixQuad vs. FitRandQuad

Name	Neg2Log Like	Parms	AIC	BIC	DevDiff	DFdiff	Pvalue
FitFixQuad	8341.5	4	8349.5	8359.9	.	.	.
FitRandQuad	8302.7	7	8316.7	8335.1	38.7316	3	1.9784E-8

Is the random quadratic time slope significant? How do we know?

Do we need to compute pseudo-R^2 for this random quadratic time slope model relative to the previous fixed quadratic time slope, random linear time slope model?

<u>95% Random Effect Confidence Intervals that describe the *predicted* range of *individual* random effects:</u>

$$\text{Random Effect 95\% CI} = \text{fixed effect} \pm \left(1.96 * \sqrt{\text{Random Variance}}\right)$$

$$\text{Intercept 95\% CI} = \gamma_{00} \pm \left(1.96 * \sqrt{\tau^2_{U_0}}\right) \rightarrow 1{,}945.9 \pm \left(1.96 * \sqrt{276{,}209}\right) = 916 \text{ to } 2{,}976$$

$$\text{Linear Time Slope 95\% CI} = \gamma_{10} \pm \left(1.96 * \sqrt{\tau^2_{U_1}}\right) \rightarrow -120.9 \pm \left(1.96 * \sqrt{25{,}840}\right) = -436 \text{ to } 194$$

$$\text{Quadratic Time Slope 95\% CI} = \gamma_{20} \pm \left(1.96 * \sqrt{\tau^2_{U_2}}\right) \rightarrow 13.9 \pm \left(1.96 * \sqrt{634}\right) = -36 \text{ to } 63$$

Is it a problem that the CIs for the linear and quadratic time slopes overlap 0? What does this mean?

Bonus: Is there any residual correlation left unmodeled in the level-1 R matrix?

```
TITLE1 "SAS: Test AR1 Residual Correlation in Random Quadratic Time Model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 time1*time1 / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT time1 time1*time1 / G GCORR V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R RCORR TYPE=AR(1) SUBJECT=PersonID;
     ODS OUTPUT InfoCrit=FitRandQuadAR1; * Save for LRT;
RUN;
TITLE1 "Calculate LRT -- does AR1 residual correlation improve model fit?";
%FitTest(FitFewer=FitRandQuad, FitMore=FitRandQuadAR1); TITLE1;
```

```
display "STATA: Test AR1 Residual Correlation in Random Quadratic Time Model"
mixed rt c.time1 c.time1#c.time1, || personid: time1 time1sq, variance reml covariance(unstructured) ///
          residuals(ar1,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estimates store FitRandQuadAR1      // Save for LRT
      lrtest FitRandQuadAR1 FitFixQuad    // Does AR1 residual correlation improve model fit?

print("R: Test AR1 Residual Correlation in Random Quadratic Time Model in LME")
RandQuadAR1 = lme(data=Example6, method="REML", rt~1+time1+I(time1^2),
                  random=~1+time1+I(time1^2)|PersonID,
                  correlation=(corAR1(form=~as.numeric(time1)|PersonID)))
print("Show results using incorrect DDF"); summary(RandQuadAR1)
print("Show G, R, and V matrices for first person")
G=getVarCov(RandQuadAR1, individual="101", type="random.effects"); G
R=getVarCov(RandQuadAR1, individual="101", type="conditional"); R
V=getVarCov(RandQuadAR1, individual="101", type="marginal"); V
print("Does AR1 residual correlation improve fit?")
print("Have to re-run random quadratic time model using lme to get LRT")
RandQuadlme = lme(data=Example6, method="REML", rt~1+time1+I(time1^2),
                  random=~1+time1+I(time1^2)|PersonID)
anova(RandQuadAR1,RandQuadlme) # anova does LRT using LME versions
```

SAS Output (relevant tables only):

Estimated R Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	24534	4197.57	718.16	122.87	21.0220	3.5967
2	4197.57	24534	4197.57	718.16	122.87	21.0220
3	718.16	4197.57	24534	4197.57	718.16	122.87
4	122.87	718.16	4197.57	24534	4197.57	718.16
5	21.0220	122.87	718.16	4197.57	24534	4197.57
6	3.5967	21.0220	122.87	718.16	4197.57	24534

Estimated R Correlation Matrix for PersonID 101

Row	Col1	Col2	Col3	Col4	Col5	Col6
1	1.0000	0.1711	0.02927	0.005008	0.000857	0.000147
2	**0.1711**	1.0000	0.1711	0.02927	0.005008	0.000857
3	0.02927	**0.1711**	1.0000	0.1711	0.02927	0.005008
4	0.005008	0.02927	**0.1711**	1.0000	0.1711	0.02927
5	0.000857	0.005008	0.02927	**0.1711**	1.0000	0.1711
6	0.000147	0.000857	0.005008	0.02927	**0.1711**	1.0000

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error	Z Value	Pr Z	
UN(1,1)	PersonID	272126	41822	6.51	<.0001	
UN(2,1)	PersonID	-35004	12302	-2.85	0.0044	
UN(2,2)	PersonID	24064	6386.03	3.77	<.0001	
UN(3,1)	PersonID	3941.63	1996.01	1.97	0.0483	
UN(3,2)	PersonID	-3554.21	1076.64	-3.30	0.0010	
UN(3,3)	PersonID	547.63	194.97	2.81	0.0025	
AR(1)	**PersonID**	**0.1711**	**0.1286**	**1.33**	**0.1833**	New AR1 correlation in R matrix
Residual		24534	4585.43	5.35	<.0001	

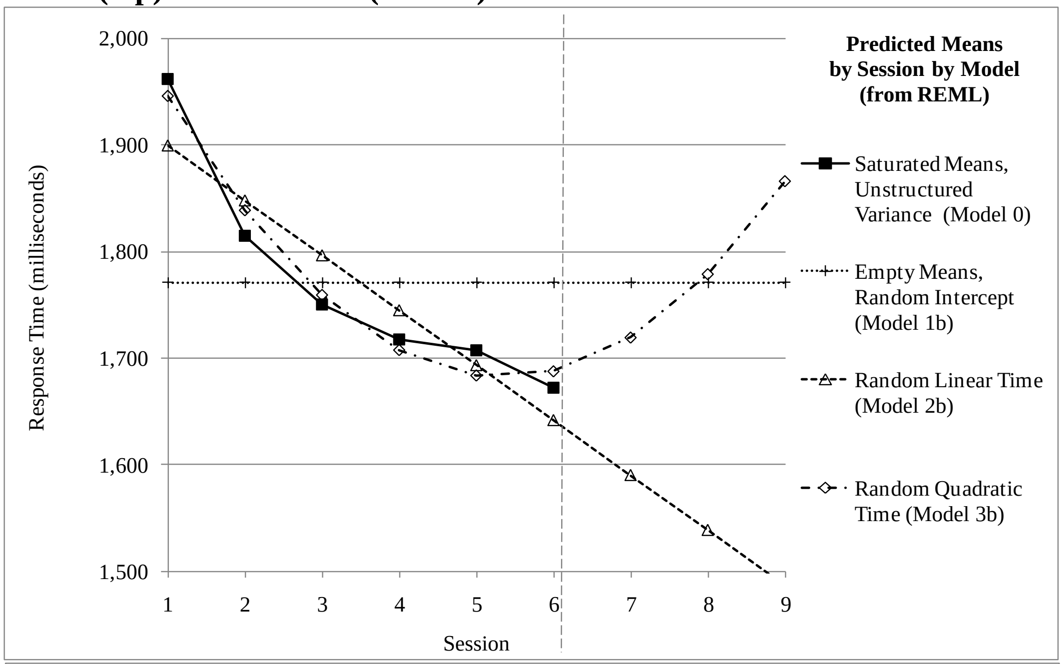
Information Criteria

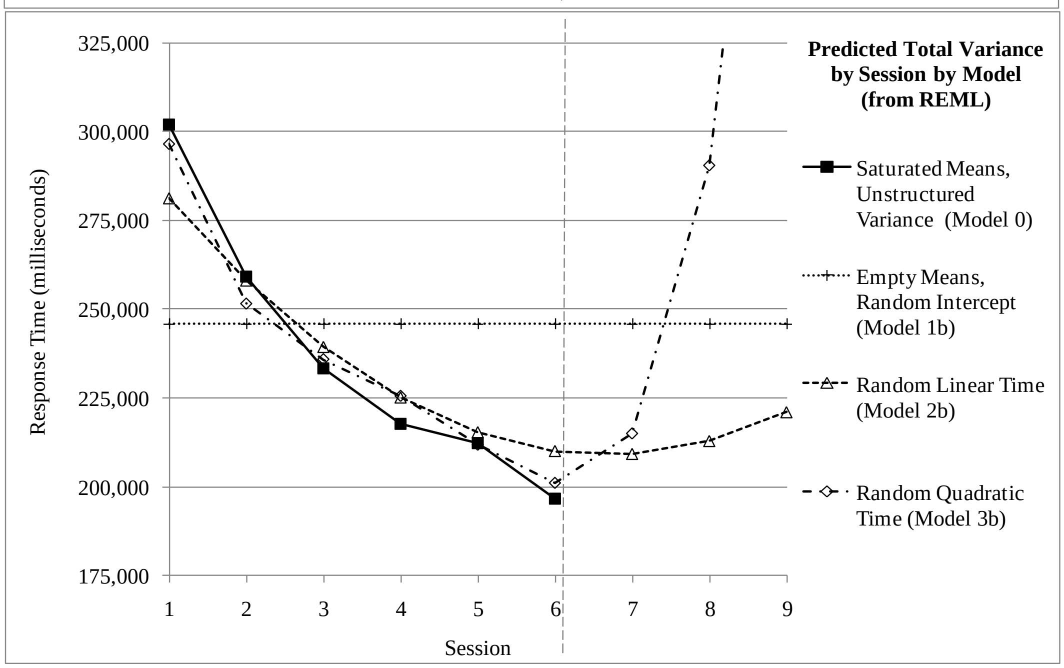
Neg2LogLike	**Parms**	AIC	AICC	HQIC	BIC	CAIC
8300.7	**8**	8316.7	8316.9	8325.1	8337.6	8345.6

Calculate LRT -- does AR1 residual correlation improve model fit?
Likelihood Ratio Test for FitRandQuad vs. FitRandQuadAR1

Name	Neg2Log Like	Parms	AIC	BIC	DevDiff	DFdiff	Pvalue
FitRandQuad	8302.7	7	8316.7	8335.1	.	.	.
FitRandQuadAR1	8300.7	8	8316.7	8337.6	2.07551	1	0.14968

Given that the AR1 R matrix did not improve model fit, I'd say this is as good as it gets for random quadratic time. So how did we do? Let's visually compare model predictions in terms of per-occasion means (top) and variances (bottom)…





Bonus Material: Testing Absolute Fit of Each Side of the Model When Using REML

As shown as Model 0, the saturated means, unstructured variance model is the best-fitting model for each side (means and variances). However, when using REML, we cannot do a model comparison against our random quadratic model, because models cannot differ in their fixed effects for the −2LL (LRT) to be valid. Instead, we can test the absolute fit for each side of the model separately as shown next (This will be included in the second edition of my textbook, in progress).

The absolute fit of the quadratic time model for the means can be tested by mimicking a saturated means model using the *same random quadratic time slopes* (i.e., holding the model for the variance constant):

```
TITLE1 "SAS: Test Absolute Fit of Quadratic Time Means Model (Using Random Quadratic Variance Model)";
TITLE2 "Add 3 session dummy codes to saturate the means model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 time1*time1 s1 s2 s3 / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT time1 time1*time1 / G GCORR V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R TYPE=VC SUBJECT=PersonID;
     CONTRAST "Does fixed quadratic time reproduce saturated means?" s1 1, s2 1, s3 1 / CHISQ;
RUN; TITLE1; TITLE2;

display "STATA: Test Absolute Fit of Quadratic Time Means Model (Using Random Quadratic Variance Model)"
display "Add 3 session dummy codes to saturate the means model"
mixed rt c.time1 c.time1#c.time1 c.s1 c.s2 c.s3, ///
           || personid: time1 time1sq, variance reml covariance(unstructured) ///
           residuals(independent,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      test (c.s1=0)(c.s2=0)(c.s3=0), small  // Does fixed quadratic time reproduce saturated means?

print("R: Test Absolute Fit of Quadratic Time Means Model (Using Random Quadratic Variance Model)")
print("Add 3 session dummy codes to saturate the means model")
QuadMean = lmer(data=Example6, REML=TRUE,
           formula=rt~1+time1+I(time1^2)+s1+s2+s3+(1+time1+I(time1^2)|PersonID))
print("Show results using Satterthwaite DDF including -2LL as deviance")
summary(QuadMean, ddf="Satterthwaite"); llikAIC(QuadMean, chkREML=FALSE)
print("Does fixed quadratic time reproduce saturated means?")
contestMD(QuadMean, ddf="Satterthwaite", L=rbind(c(0,0,0,1,0,0),c(0,0,0,0,1,0),c(0,0,0,0,0,1)))
```

SAS Output (relevant tables only):

Solution for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1603.14	271.78	323	5.90	<.0001
time1	74.8457	138.82	312	0.54	0.5902
time1*time1	-12.2095	17.3760	323	-0.70	0.4828
s1	358.75	267.06	300	1.34	0.1802
s2	149.39	147.22	300	1.01	0.3110
s3	46.0365	62.7733	300	0.73	0.4639

Because there are now **6 fixed effects for the 6 means**, this model is equivalent to **saturated means** (even if the linear and quadratic fixed slopes are largely uninterpretable). In an SEM context, this model would be specified by letting three of the observed occasions' intercepts be estimated (as discrepancies).

The multivariate Wald test using CONTRAST indicates that the 3 extra session contrasts improved model fit (which is bad news here).

Contrasts

Label	Num DF	Den DF	Chi-Square	F Value	Pr > ChiSq	Pr > F
Does fixed quadratic time reproduce saturated means?	3	300	9.07	3.02	0.0284	0.0299

Btw, in this quadratic model, it doesn't matter which three dummy codes are added as fixed effects…

```
TITLE1 "SAS: Test Absolute Fit of Quadratic Time Means Model (Using Random Quadratic Variance Model)";
TITLE2 "Add different 3 session dummy codes to saturate the means model";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 time1*time1 s4 s5 s6 / SOLUTION DDFM=Satterthwaite;
     RANDOM INTERCEPT time1 time1*time1 / G GCORR V VCORR TYPE=UN SUBJECT=PersonID;
     REPEATED session / R TYPE=VC SUBJECT=PersonID;
     CONTRAST "Does fixed quadratic time reproduce saturated means?" s4 1, s5 1, s6 1 / CHISQ;
RUN; TITLE1; TITLE2;

display "STATA: Test Absolute Fit of Quadratic Time Means Model (Using Random Quadratic Variance Model)"
display "Add different 3 session dummy codes to saturate the means model"
mixed rt c.time1 c.time1#c.time1 c.s4 c.s5 c.s6, ///
           || personid: time1 time1sq, variance reml covariance(unstructured) ///
           residuals(independent,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      test (c.s4=0)(c.s5=0)(c.s6=0), small  // Does fixed quadratic time reproduce saturated means?

print("R: Test Absolute Fit of Quadratic Time Means Model (Using Random Quadratic Variance Model)")
print("Add different 3 session dummy codes to saturate the means model")
QuadMean2 = lmer(data=Example6, REML=TRUE,
                formula=rt~1+time1+I(time1^2)+s4+s5+s6+(1+time1+I(time1^2)|PersonID))
print("Show results using Satterthwaite DDF including -2LL as deviance")
summary(QuadMean2, ddf="Satterthwaite"); llikAIC(QuadMean2, chkREML=FALSE)
print("Does fixed quadratic time reproduce saturated means?")
contestMD(QuadMean2, ddf="Satterthwaite", L=rbind(c(0,0,0,1,0,0),c(0,0,0,0,1,0),c(0,0,0,0,0,1)))
```

SAS Output (relevant tables only):

Because there are now **6 fixed effects for the 6 means**, this model is equivalent to **saturated means** (even if the linear and quadratic fixed slopes are largely uninterpretable). In an SEM context, this model would be specified by letting three of the observed occasions' intercepts be estimated (as discrepancies).

The multivariate Wald test using CONTRAST indicates that the 3 extra session contrasts improved model fit (which is bad news here).

Solution for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1961.89	54.1754	102	36.21	<.0001
time1	-187.51	39.2348	400	-4.78	<.0001
time1*time1	40.7916	17.3760	323	2.35	0.0195
s4	-48.6837	62.7733	300	-0.78	0.4386
s5	-157.33	147.22	300	-1.07	0.2861
s6	-371.98	267.06	300	-1.39	0.1647

Contrasts

Label	Num DF	Den DF	Chi-Square	F Value	Pr > ChiSq	Pr > F
Does fixed quadratic time reproduce saturated means?	3	300	9.07	3.02	0.0284	**0.0299**

The absolute fit of the random quadratic time model for the variance can be tested against a UN variance model using the *same fixed quadratic time slopes* (i.e., holding the model for the means constant):

```
TITLE1 "SAS: Test Absolute Fit of Quadratic Time Variance Model (Using Fixed Quadratic Means Model)";
TITLE2 "Change to Unstructured R matrix as variance model answer key";
PROC MIXED DATA=work.Example6 COVTEST NOCLPRINT NAMELEN=100 IC METHOD=REML;
     CLASS PersonID session;
     MODEL rt = time1 time1*time1 / SOLUTION DDFM=Satterthwaite;
     REPEATED session / R RCORR TYPE=UN SUBJECT=PersonID;
     ODS OUTPUT InfoCrit=FitFixQuadUN; * Save for LRT;
RUN; TITLE2;
TITLE1 "Calculate LRT -- does random quadratic time reproduce UN variance model?";
%FitTest(FitFewer=FitRandQuad, FitMore=FitFixQuadUN); TITLE1;

display "STATA: Test Absolute Fit of Quadratic Time Variance Model (Using Fixed Quadratic Means Model)"
display " Change to Unstructured R matrix as variance model answer key"
mixed rt c.time1 c.time1#c.time1, || personid: , noconstant variance reml ///
          residuals(unstructured,t(session)) dfmethod(satterthwaite) dftable(pvalue)
      estimates store FitFixQuadUN     // Save for LRT
      lrtest FitFixQuadUN FitRandQuad  // Does random quadratic time reproduce UN variance model?

print("R: Testing Absolute Fit of Quadratic Time Variance Model (Using Fixed Quadratic Means Model)")
print("Change to Unstructured R matrix as variance model answer key in GLS")
FixQuadUN = gls(data=Example6, method="REML", model=rt~1+time1+I(time1^2),
                correlation=corSymm(form=~as.numeric(session)|PersonID),
                weights=varIdent(form=~1|session))
print("Have to re-run random quadratic time model using LME to get LRT")
RandQuadlme = lme(data=Example6, method="REML", rt~1+time1+I(time1^2),
                  random=~1+time1+I(time1^2)|PersonID)
anova(FixQuadUN,RandQuadlme) # anova does LRT using LME versions
```

SAS Output (relevant table only):

Calculate LRT -- does random quadratic time reproduce UN variance model?
Likelihood Ratio Test for FitRandQuad vs. FitFixQuadUN

Name	Neg2Log Like	Parms	AIC	BIC	DevDiff	DFdiff	Pvalue
FitRandQuad	8302.7	7	8316.7	8335.1	.	.	.
FitFixQuadUN	8267.0	21	8309.0	8363.9	35.7580	14	.001134108

What does this significant LRT result indicate about our random quadratic time model?