Future Directions in Software for Mixed-Effects Models:

Version 3.0 of NLME

José C. Pinheiro

Bell Laboratories, Lucent Technologies

and

Douglas M. Bates

Department of Statistics, UW–Madison

Outline

- Current implementation: NLME 2.1
- groupedData class and methods
- Linear Mixed-Effects Models: lme
- Nonlinear Mixed-Effects Models: nlme
- Model extensions:
  - reStruct and pdMat classes
  - corStruct classes
  - varFunc classes
- Multilevel LME models
- Conclusions and future developments

NLME 2.1

- Available with versions 3.4 and higher of S-PLUS
- Fits linear (lme) and nonlinear (nlme) mixed-effects models
- Different covariance structures for random effects (e.g. unstructured, diagonal)
- Correlation structures (e.g. AR1) and variance functions (e.g. power of fitted response) for within-group error
- Primarily intended for single grouping factor, though possible, in some cases, to extend to multilevel grouping
- Customization possible, but difficult

Grouped data

- Observations can be divided into disjoint groups according to one or more classification variables
- Examples include longitudinal data, repeated measures data, and split-plot designs.
- Observation within the same group are frequently correlated.
- Data with a single grouping factor are the rule in medical studies, but multiply grouped data are common in industrial and agricultural applications and in sociological studies
- Displaying, summarizing, and modeling grouped data may be challenging.

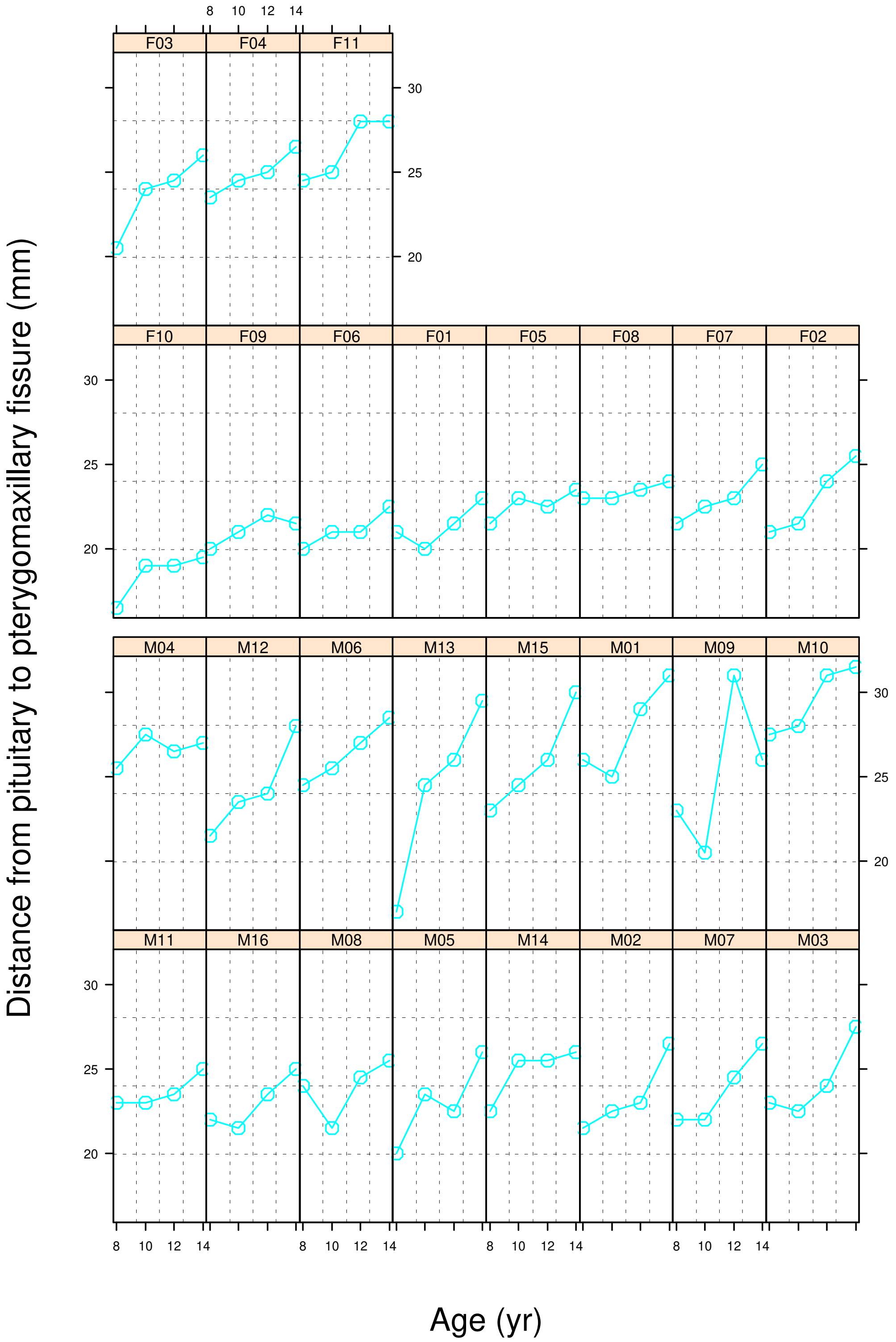
Some examples of grouped data

Orthodont – orthodontic growth data

Distance (in millimeters) between the pituitary and the pterygomaxillary fissure measured every two years for 27 children (16 males, 11 females) from age 8 until age 14.

- balanced data
- linear growth with time
- similar patterns across subjects, but with different intercepts and slopes
- single level of grouping: subject

Orthodont grouped data

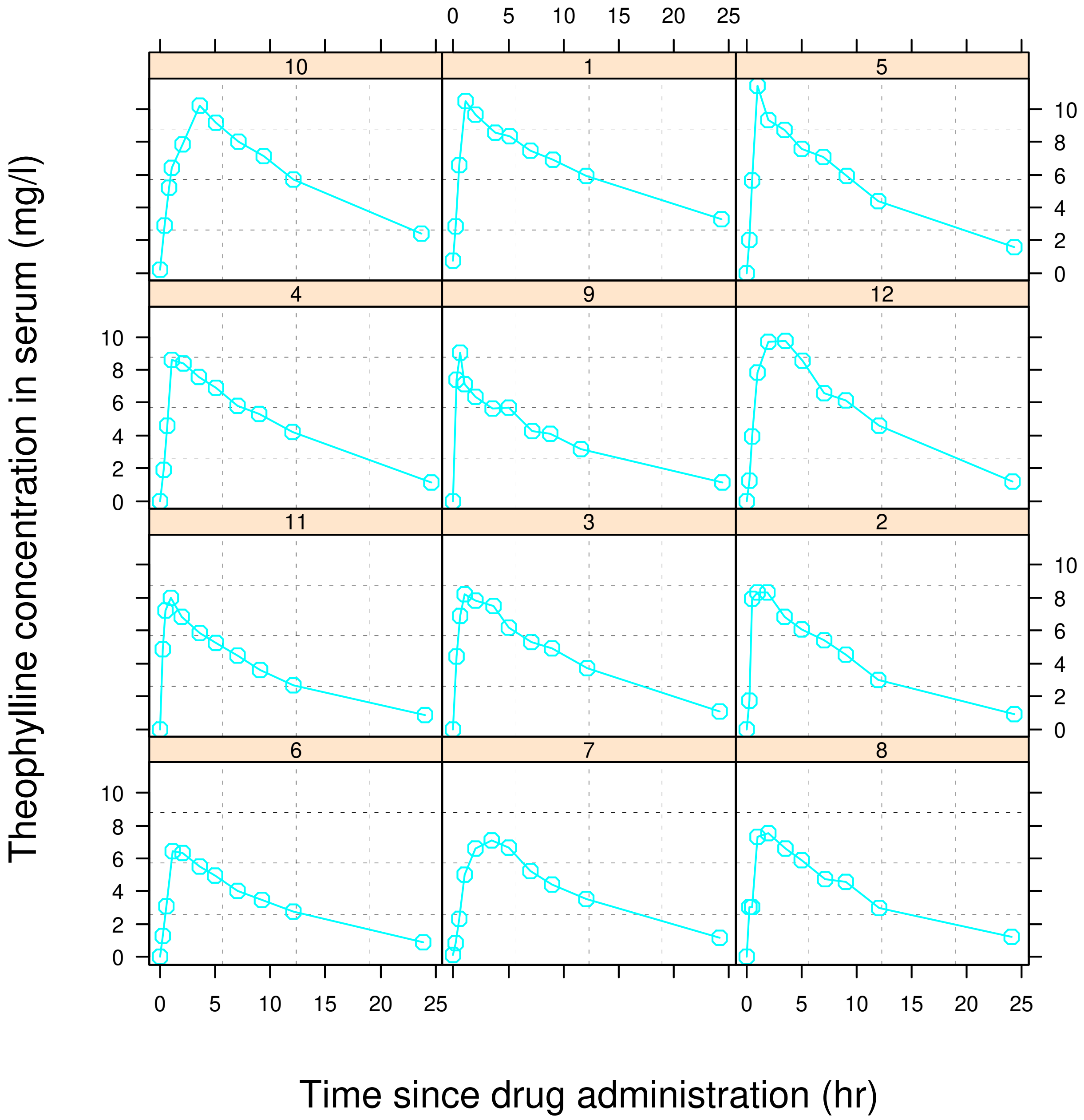


Some examples: Theoph data

Theophylline kinetics: Twelve subjects were given oral doses of Theophylline then serum concentrations (in mg/L) were measured at 11 time points over the next 25 hours.

- typical laboratory study with few subjects and several, (nearly) balanced observations
- Nonlinear relationship between concentration and time
- Similar patterns across subjects, with different parameters
- Single grouping factor: subject

Theoph grouped data

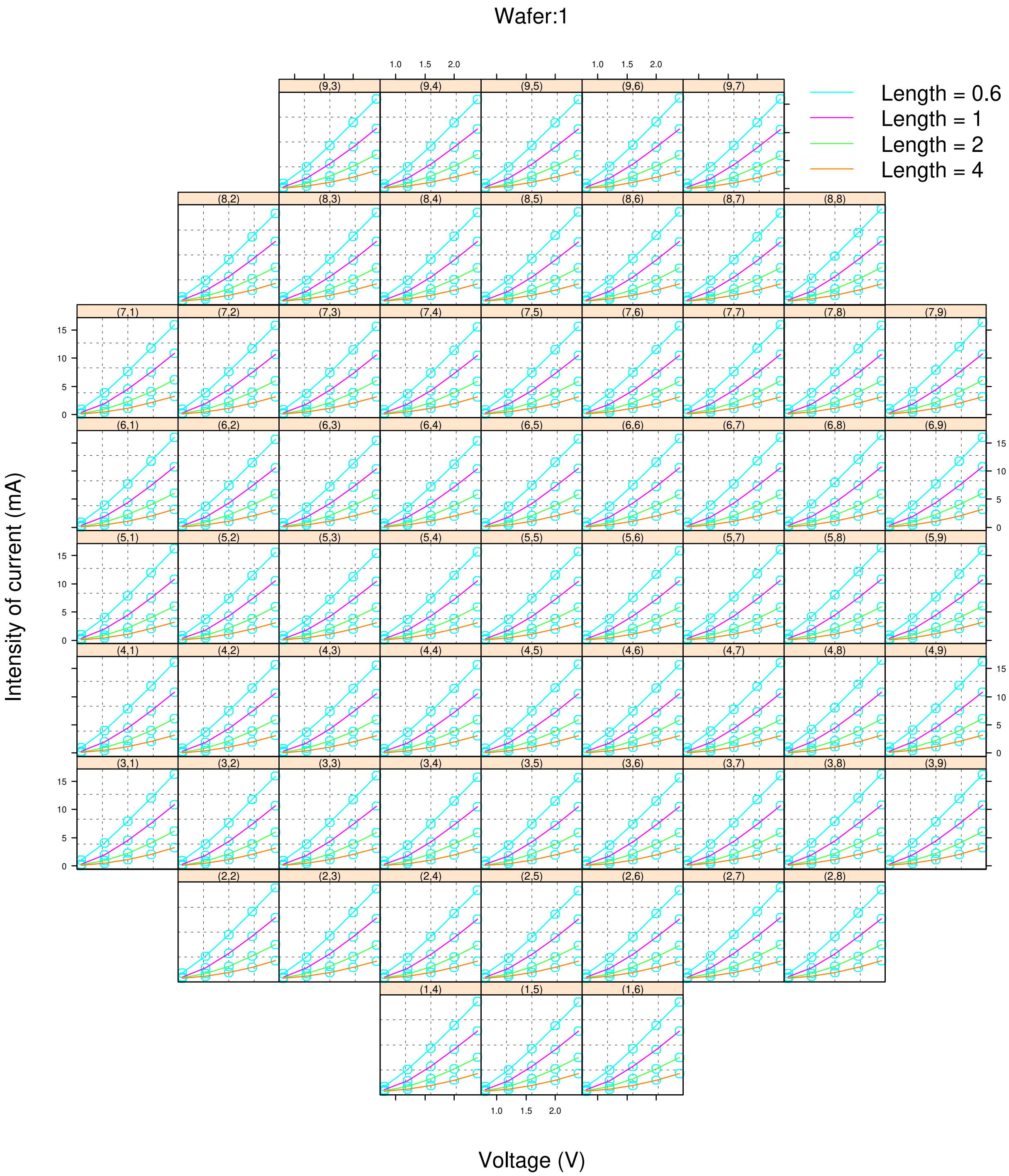


Some examples: Wafer data

Device variability in IC manufacturing

- Experiment to determine the effect of channel length on variability of analog MOS circuits
- Intensity of current (in mA) at five ascending voltages (in Volts) were collected on n-channel devices of four different width/length ratios: 80μm/0.6μm, 80/1, 80/2, and 80/4.
- Measurements were made on 6 wafers, each subdivided into 16 sites containing four devices, one from each geometry.
- Three nested levels of grouping: Wafer, Site within Wafer, and Device within Site within Wafer

Wafer grouped data



A groupedData class

- Common and versatile way of organizing data in S-PLUS is as data.frame objects: rectangular arrays with rows corresponding to observations and columns to variables.

- groupedData objects extend data.frames by incorporating a formula that designates special roles for the variables

- Most important special roles are: the response variable, the primary covariate, and the grouping factors.

- Other factors and covariates may also be assigned special roles

Creating groupedData objects

- A function to create an object of a given class is called a constructor for that class
- The groupedData constructor has two required arguments:
  - a display formula specifying the special roles for the variables of the form:
    `response ~ primary|groups`
    for single nesting, or
    `response ~ primary|g1/g2/../gN`
    for multiple nesting
  - a data.frame representing the data
- Optional arguments include: labels, units, outer, and inner

Orthodont groupedData

Data available in a file *Ortho.dat*

```
   distance age Subject Sex
     26.0     8     M01     M
     25.0    10     M01     M
     . . .
     28.0    12     F11     F
     28.0    14     F11     F
```

groupedData object constructed with

```
> Orthodont <- groupedData(formula = distance  ~ age | Subject,
+   data = read.table("Ortho.dat", header = T),
+   outer = ~ Sex,
+   labels = list(x = "Age",
+       y = "Distance from pituitary to pterygomaxillary fissure"),
+   units = list(x = "(yr)", y = "(mm)"))
> Orthodont
Grouped Data: distance ~ age | Subject
    distance age Subject Sex
  1      26.0   8      M01   M
  2      25.0  10      M01   M
. . .
> class(Orthodont)
[1] "groupedData"    "data.frame"
```

Special roles of variables in display formula:

- response is distance
- primary covariate is age
- grouping variable is Subject

Theoph groupedData

Data available in a data frame *Theoph*

```
> Theoph
   Subject  Weight Dose  time conc
1        1  >= 155 4.02  0.00 0.74
2        1  >= 155 4.02  0.25 2.84
. . .
131     12   < 155 5.30 12.05 4.57
132     12   < 155 5.30 24.15 1.17
```

groupedData object constructed with

```
> Theoph <-
+   groupedData(formula = conc ~ time | Subject,
+      data = Theoph,
+      labels = list(x = "Time since drug administration",
+                    y = "Theophylline concentration in serum"),
+      units = list(x = "(hr)", y = "(mg/l)"),
+      outer = ~ Weight)
```

Special roles of variables in display formula:

- response is conc
- primary covariate is time
- grouping variable is Subject

Wafer groupedData

Data available in data.frame `Wafer`

```
> Wafer
   Wafer  Site Device Length voltage  current
 1      1      1       1     0.6      0.8  0.92158
 2      1      1       1     0.6      1.2  3.91040
 3      1      1       1     0.6      1.6  7.69520
. . .
```

groupedData object constructed with

```
> Wafer <-
+   groupedData(current ~ voltage | Wafer/Site/Device,
+      data = Wafer,
+      labels = list(x = "Voltage", y = "Intensity of current"),
+      units = list(x = "(V)", y = "(mA)"),
+      outer = list(Device = ~ Length))
```

Special roles of variables in display formula:

- response is current
- primary covariate is voltage
- grouping structure is Wafer/Site/Device

Displaying groupedData objects

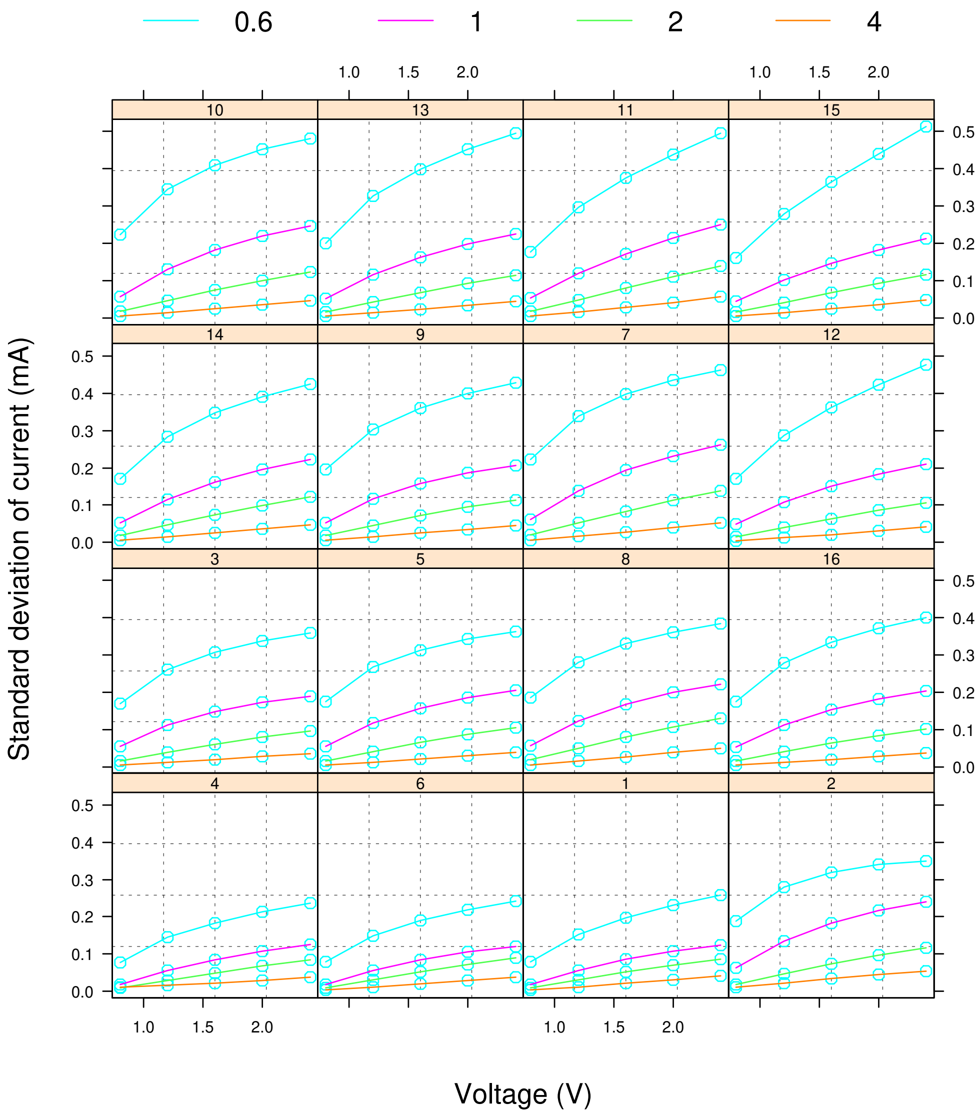
- Plot methods based on the Trellis library.
- In the single-nesting case:
  - display and Trellis formulas coincide
  - By default, uses one panel per group.
- In the multi-nesting case, may need to specify:
  - the display level
  - the collapsing level
  - the collapsing function

Displaying the Wafer Data

display and collapsing levels = `Wafer` and

collapsing function = standard deviation

```
> plot(Wafer, display = 1, collapse = 1, inner = ~ Length,
+       FUN = function(x) sqrt(var(x)))
```



A Linear Mixed-Effects Model

Proposed by Laird and Ware (1982). In the single grouping case, the response vector $\boldsymbol{y}_i$ for the ith group is represented as

$$\boldsymbol{y}_i = \boldsymbol{X}_i\boldsymbol{\beta} + \boldsymbol{Z}_i\boldsymbol{b}_i + \boldsymbol{\epsilon}_i,\ i = 1, \ldots, M$$

where $\boldsymbol{\beta}$ is the fixed effects vector, $\boldsymbol{b}_i$ is the random effects vector, $\boldsymbol{X}_i$ and $\boldsymbol{Z}_i$ are known fixed effects and random effects regression matrices, and $\boldsymbol{\epsilon}_i$ is the within-group error vector.

The assumptions of the LME model are:

- The $\boldsymbol{b}_i$ are independently distributed as $\mathcal{N}\left(\boldsymbol{0}, \sigma^2\boldsymbol{D}\right)$
- The $\boldsymbol{\epsilon}_i$ are independently distributed as $\mathcal{N}\left(\boldsymbol{0}, \sigma^2\boldsymbol{\Lambda}_i\right)$
- The $\boldsymbol{b}_i$ and the $\boldsymbol{\epsilon}_i$ are independent.

LME model (cont.'d)

The LME model allows within-group correlations because:

- observations within the same group share the same random effects $\boldsymbol{b}_i$;
- the within-group covariance matrix $\boldsymbol{\Lambda}_i$ allows general structures, such as AR(1), MA(1), etc.

These two capabilities for modeling the within-group correlation may compete with each other in the model specification.

The lme function, class, and methods

- Maximum likelihood and restricted maximum likelihood
- Syntax similar to lm
  `lme(fixed, data, random)`
- Unstructured and structured $\boldsymbol{D}$: reStruct and pdMat
- Correlation structures for $\boldsymbol{\Lambda}_i$: corStruct
- Variance Functions for $\boldsymbol{\Lambda}_i$: varFunc
- Methods for displaying, plotting, predicting and comparing fits

Analysis of the Orthodont Data

A linear model seems adequate and both intercept and slope appear to vary with individual.

$$y_{ij} = (\beta_0 + b_{0i}) + (\beta_1 + b_{1i})\,\mathsf{age}_j + \epsilon_{ij}$$

Fitting this model with lme (version 3.0):

```
> Orth.lme1 <- lme(fixed = distance ~ age, data = Orthodont)
> Orth.lme1
Call:
  Fixed: distance ~ age
 Random:  ~ age
  Groups:  ~ Subject
   Data: Orthodont
Random effects variance-covariance
 Structure: Unstructured, Matrix Logarithm parameterization
 Standard Deviations:
 (Intercept)       age
    2.194103 0.2149245
 Correlation:
    (Intercept)
age -0.5814881
 Within-Group Standard Error: 1.31004
Fixed Effects Estimate(s):
  (Intercept)       age
    16.76111 0.6601852
Number of Observations: 108
Number of Groups: 27
```

Analysis of the Orthodont Data (cont.'d)

Fitted objects inherit from class lme, for which several methods are available.

The update method can be used to test if there are differences in growth pattern between boys and girls.

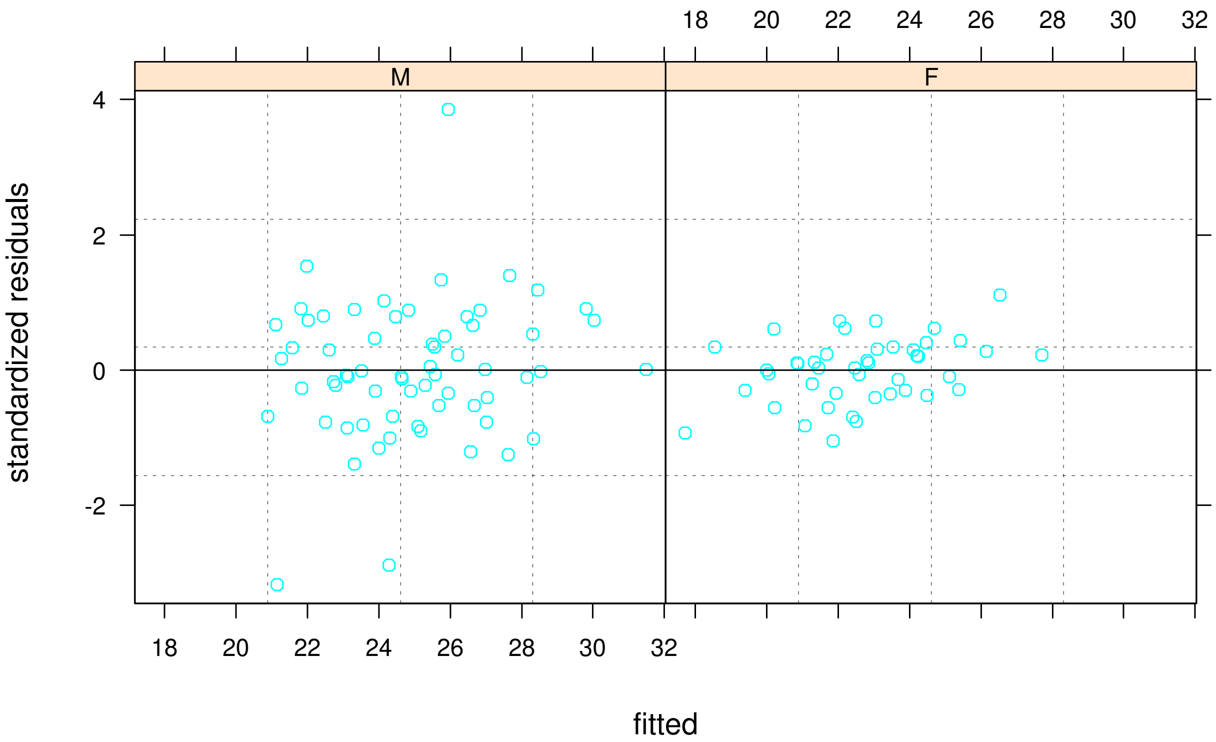
```
> Orth.lme2 <- update(Orth.lme1, distance ~ Sex * age)
> summary(Orth.lme2)
. . .
Fixed Effects Estimate(s):
                  Value Approx. Std.Error     z-value
(Intercept)  16.8566761        0.76775405  21.9558285
        Sex   0.5160511        0.76775405   0.6721568
        age   0.6319602        0.06482534   9.7486609
    Sex:age  -0.1524148        0.06482534  -2.3511605
. . .
```

There doesn't seem to be gender differences in intercept, but the growth rate appears to be higher among boys.

Analysis of the Orthodont Data (cont.'d)

The adequacy of the fit can be examined using the plot method:

```
> plot(Orth.lme2, resid(., type = "s") ~ fitted(.) | Sex)
```



There appears to be more variability among boys, than among girls.

A Non-linear Mixed-Effects Model

Proposed by Lindstrom and Bates (1990). In the single grouping variable case, the jth response of the ith individual is represented as

$$y_{ij} = f\left(\boldsymbol{\phi}_{ij}, \boldsymbol{x}_{ij}\right) + \epsilon_{ij}$$

- model function f relating response y_{ij} to covariates $\boldsymbol{x}_{ij}$ is nonlinear in parameters $\boldsymbol{\phi}_{ij}$.
- $\boldsymbol{\phi}_{ij}$ can include common parameters (fixed effects) $\boldsymbol{\beta}$ and random effects $\boldsymbol{b}_i$. We set

$$\boldsymbol{\phi}_{ij} = \boldsymbol{A}_{ij}\boldsymbol{\beta} + \boldsymbol{B}_{ij}\boldsymbol{b}_i$$

  where $\boldsymbol{A}_{ij}, \boldsymbol{B}_{ij}$ are design matrices.

NLME Model (cont.'d)

- Distributional assumptions

$$\boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{\Lambda}_i) \quad \boldsymbol{b}_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 \boldsymbol{D})$$

  $\boldsymbol{\epsilon}_i$ and $\boldsymbol{b}_i$ independent.

- In general the interest is in population characteristics $\boldsymbol{\beta}$, the fixed effects, $\boldsymbol{D}$, the scaled covariance matrix of the random effects, $\boldsymbol{\Lambda}_i$, the scaled within-group covariance matrix, and σ^2, the within-group variance. Estimates of the $\boldsymbol{\phi}_{ij}$ may also be of interest.

The nlme function, class, and methods

- Approximate maximum likelihood and restricted maximum likelihood
- Syntax similar to nls

  `nlme(object, fixed, data, random, start)`
- Unstructured and structured $\boldsymbol{D}$: reStruct and pdMat
- Correlation structures for $\boldsymbol{\Lambda}_i$: corStruct
- Variance functions for $\boldsymbol{\Lambda}_i$: varFunc
- Methods for displaying, plotting, predicting, updating, and comparing fits

Analysis of the Theoph Data

Theoretical model: first order compartment model with absorption in a peripheral compartment. Concentration C_t at time t is modeled as

$$C_t = \frac{\Delta K k_a}{\text{Cl}(k_a - K)} \left[\exp\left(-Kt\right) - \exp\left(-k_a t\right)\right]$$

where Δ is the initial dose, k_a is the absorption rate constant, K is the elimination rate constant, and Cl is the clearance.

To ensure positivity of the rate constants and the clearance, the model is reparameterized using the logarithms of these quantities.

Analysis of Theoph Data (cont.'d)

Fitting the model with nlme (version 3.0):

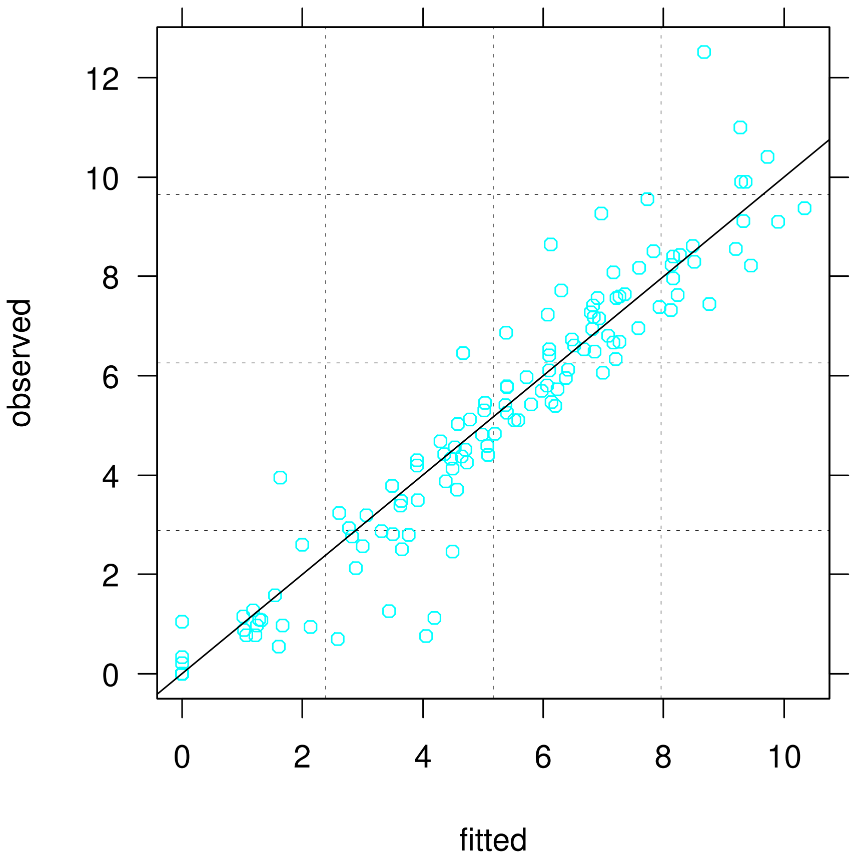
```
> Theoph.nlme1 <- nlme(conc ~ FOC(Dose, time, lCl, lka, lK),
+                     fixed = lCl + lka + lK ~ 1,
+                     data = Theoph,
+                     random = lCl + lka ~ 1,
+                     start = list(fixed = c(-3, 0.5, -2)))

> Theoph.nlme1
. . .
Random effects variance-covariance
 Structure: Unstructured, Matrix Logarithm parameterization
 Standard Deviations:
       lCl       lka
 0.1671167 0.6409361
 Correlation:
           lCl
lka 0.02296011

 Within-Group Standard Error: 0.7095636

Fixed Effects Estimate(s):
        lCl       lka         lK
 -3.227083 0.4653843 -2.454484

Number of Observations: 132
Number of Groups: 12
```

Initial values for fixed effects and choice of random components based on individual fits.

Analysis of Theoph Data (cont.'d)

We can evaluate the the fit with the plot method

```
> plot(Theoph.nlme1, conc ~ fitted(.))
```

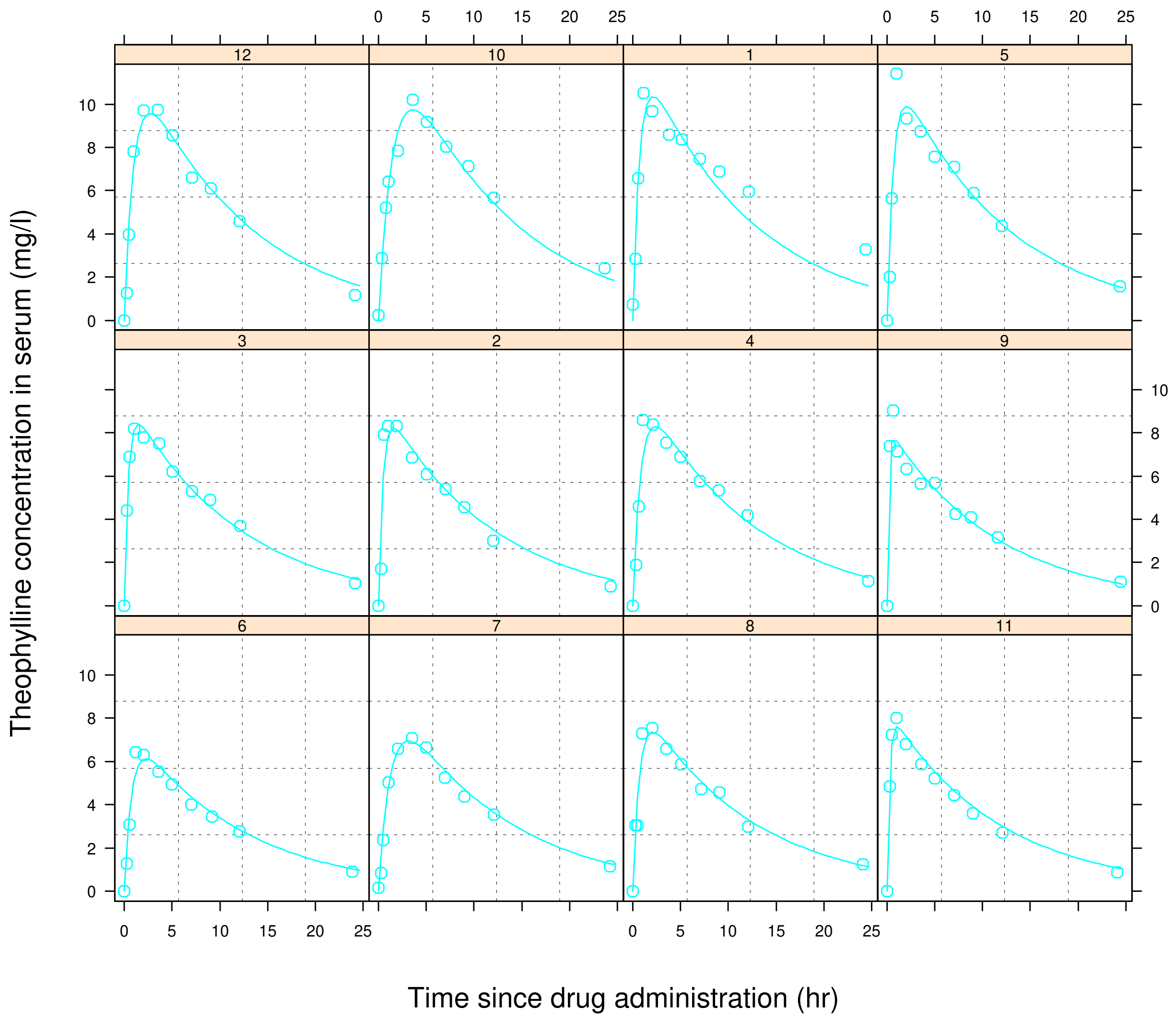


The fitted model explains the individual Theophylline concentrations reasonably well. There are no apparent violations of the assumptions in the NLME model.

Analysis of Theoph Data (cont.'d)

We can get a better idea of goodness-of-fit with the augPred (augmented predictions) method.

```
> plot(augpred(Theoph.nlme1))
```



Random effects structures

A reStruct object includes the following:

- a formula, or list of formulas in the multilevel case, defining the random effects model and grouping variable(s)
  $\sim$ `x1 + x2` $|$ `g`
- a parameterization for the covariance structure, defined through a pdMat class (default is *unstructured* with matrix logarithm parameterization)
- values for the unconstrained parameters defining the covariance structure
- methods for coef, coef $\Leftarrow$, and logLik

pdMat: positive definite matrices

- Highly customizable classes for representing positive definite matrices
- User must define a constructor and methods for coef and coef $\Leftarrow$
- Internal representation: unconstrained coefficients
- Available classes:
  - Unconstrained: pdChol, pdLogChol, pdMatrixLog, pdSpher, pdGivens
  - pdDiag
  - pdIdent
  - pdCompSymm
  - pdBlocked

pdMat: example

Unstructured with log-Choleski parameterization

```
> pdObj1 <- pdLogChol(~ age * Sex, data = Orthodont)

> pdObj1
Uninitialized positive definite matrix structure of class pdLogChol.

> Names(pdObj1)
[1] "(Intercept)" "age"         "Sex"         "age:Sex"

> Dim(pdObj1)
[1] 4 4

> coef(pdObj1) <- double(10)

> pdObj1
Variance matrix structure of class pdLogChol representing
            (Intercept) age Sex age:Sex
(Intercept)           1   0   0       0
        age           0   1   0       0
        Sex           0   0   1       0
    age:Sex           0   0   0       1

> coef(pdObj1)     # by default, upper triangular part
[1] 1 0 1 0 0 1 0 0 0 1

> coef(pdObj1, unconst = T)
[1] 0 0 0 0 0 0 0 0 0 0
```

reStruct: example

In the analysis of the Theoph data, low estimated correlation between lCl and lka random effects suggests a diagonal covariance structure

```
> Theoph.nlme2 <- update(Theoph.nlme1,
+   random = pdDiag(lCl + lka ~ 1))

> Theoph.nlme2
. . .
Random effects variance-covariance
 Structure: Diagonal
 Standard Deviations:
       lKa       lCl
 0.6469824 0.1668781

 Within-Group Standard Error: 0.708896

Fixed Effects Estimate(s):
        lKe       lKa     lCl
 -2.454663 0.4658023 -3.2272
. . .

> anova(Theoph.nlme1, Theoph.nlme2)
             Model df    AIC    BIC  logLik   Test Lik.Ratio P-value
Theoph.nlme1     1  7 368.04 388.22 -177.02
Theoph.nlme2     2  6 366.04 383.34 -177.02 1 vs. 2 0.0016545 0.96755
```

Correlation structure classes

A corStruct object includes the following:

- a formula representing covariate(s) and, optionally, grouping variables

  $\sim$ `time | group`

- values for the unconstrained parameters defining the correlation structure

- values for the covariate(s), possibly split by groups

- methods for coef, coef $\leftarrow$, logLik, and recalc

- available classes: corAR1, corCAR1, corARMA, corCompSymm, and corHF

- used in lme and nlme, but easy to incorporate in other functions, such as lm, aov, and nls

corStruct: example

Theoph example: concentration is measured over time, serial correlation may be present. Fitting and testing an CAR(1) model.

```
> Theoph.nlme3 <- update(Theoph.nlme2,
+     corStruct = corCAR1(formula = ~ time))

> Theoph.nlme3
. . .
 Standard Deviations:
       lKa       lCl
 0.6411898 0.1669475

 Serial Correlation Structure: Continuous AR(1)
 Parameter estimate(s):
         Phi
 1.247175e-08

 Within-Group Standard Error: 0.7096584

Fixed Effects Estimate(s):
        lKe       lKa       lCl
 -2.454665 0.4654057 -3.227212
. . .
> anova(Theoph.nlme2, Theoph.nlme3)
             Model df    AIC    BIC  logLik    Test Lik.Ratio P-value
Theoph.nlme2     1  6 366.04 383.34 -177.02
Theoph.nlme3     2  7 368.03 388.21 -177.02 1 vs. 2 0.0066783 0.93487
```

Variance function classes

A varFunc object includes the following:

- a formula representing covariate(s) and, optionally, grouping variables
- values for the unconstrained parameters defining the variance function
- values for variables used in calculating covariate(s)
- methods for coef, coef $\Leftarrow$, and varWeights
- available classes: varIdent, varPower, varExpon, varConstPower, and varComb
- used in lme and nlme, but easy to incorporate in other functions, such as lm, aov, and nls

varFunc: example

Orthodont example: variance appears to be larger amongst boys. Testing with varIdent variance function

```
> Orth.lme3 <- update(Orth.lme2,
+   varFunc = varIdent(formula = ~ 1 | Sex))

> Orth.lme3
. . .

 Variance Function: Different standard errors per stratum
 Parameter estimates:
 M        F
 1 0.411338

 Within-Group Standard Error: 1.621528
. . .

> anova(Orth.lme2, Orth.lme3)
          df    AIC    BIC  logLik    Test Lik.Ratio   P-value
Orth.lme2  8 443.81 465.26 -213.90
Orth.lme3  9 424.04 448.18 -203.02 1 vs. 2    21.764 3.084e-06
```

varFunc: another example

Pharmacokinetics data: variability usually increases with response. Testing varConstPower variance function in Theoph example

```
> Theoph.nlme4 <- update(Theoph.nlme2,
+     varFunc = varConstPower(~ fitted(.)))

> Theoph.nlme4
. . .
 Standard Deviations:
      lKa       lCl
 0.639121 0.1701568

 Variance Function: Constant plus power of fitted value
 Parameter estimates:
    Const.     Power
 0.2260263 0.2492105

 Within-Group Standard Error: 0.4792242
. . .

> anova(Theoph.nlme2, Theoph.nlme4)
             Model df    AIC    BIC  logLik   Test Lik.Ratio    P-value
Theoph.nlme2     1  6 366.04 383.34 -177.02
Theoph.nlme4     2  8 355.14 378.20 -169.57 1 vs. 2    14.905 0.00057994
```

Multilevel LME models

- Generalization of the Laird and Ware model for data of multiple, nested levels grouping.
- In the two-grouping-level case, the n_{ij}-dimensional vector of responses for the jth level-2 unit nested within the ith level-1 unit can be written

$$\boldsymbol{y}_{ij} = \boldsymbol{X}_{ij}\boldsymbol{\beta} + \boldsymbol{Z}_{i,j}\boldsymbol{b}_i + \boldsymbol{Z}_{ij}\boldsymbol{b}_{ij} + \boldsymbol{\epsilon}_{ij}$$

$$\boldsymbol{b}_i \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{D}_1), \quad \boldsymbol{b}_{ij} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{D}_2)$$

$$\boldsymbol{\epsilon}_{ij} \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{\Lambda}_{ij})$$

- Can be generalized to any number of nested levels
- lme function in NLME 3.0 handles multilevel models

Analysis of the Wafer data

- Three levels of nesting: Wafer/Site/Device
- Variability depends on channel length
- Fairly large data set:
  $16 \times 67 \times 4 \times 5 = 21440$ observations
- Model building strategy: study each length separately
  - Grouping structure simplifies to Wafer/Site
  - 5360 observations per data set
  - Response function has the same shape for all lengths
  - Variability is approximately constant for the same length

Wafer data analysis (cont.'d)

- illustrate analysis steps with 0.6μm data

```
> Waf0.6 <- Wafer[Wafer$Length == 0.6,]
```

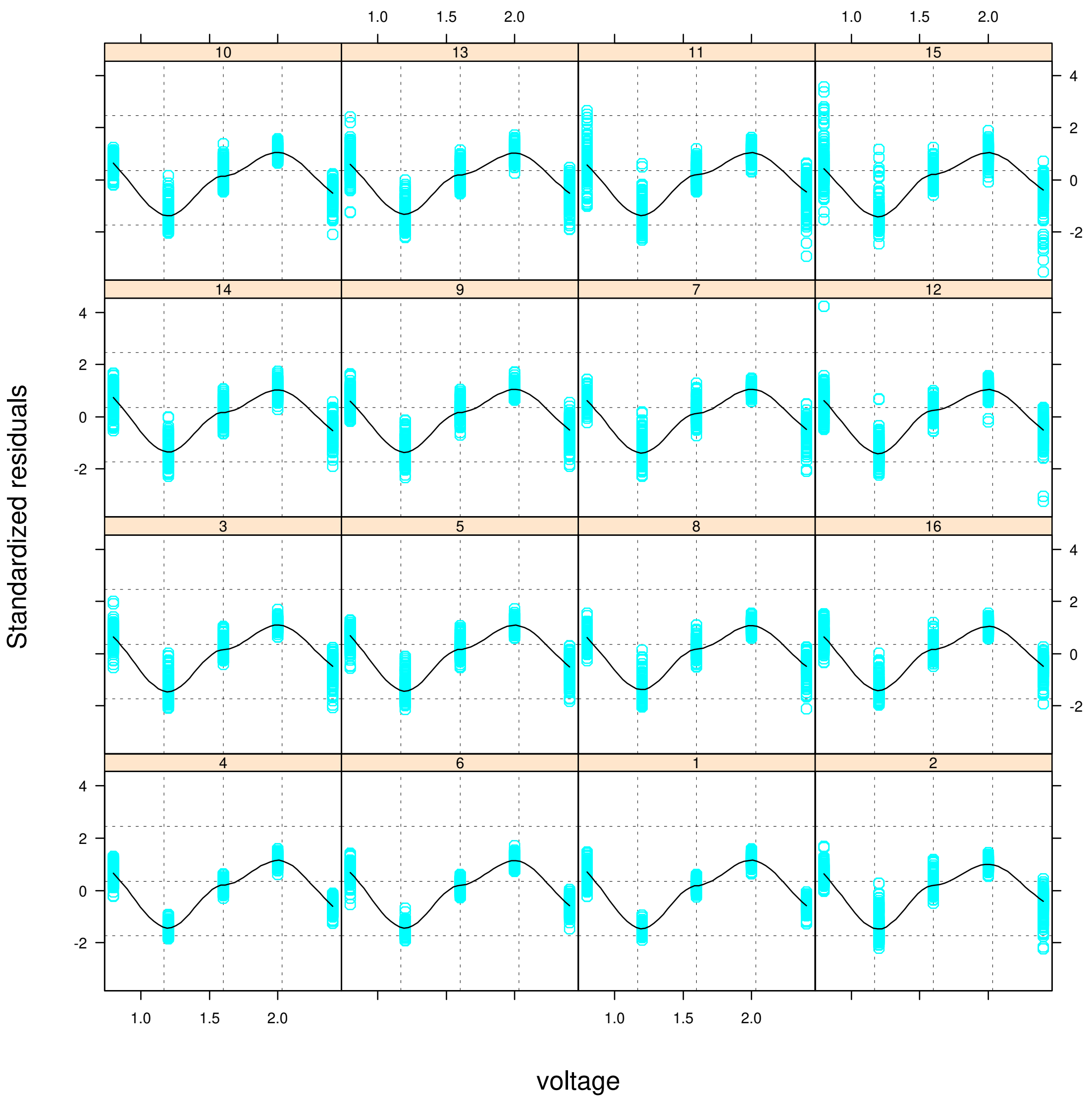
- second order polynomial seems a reasonable empirical model
- preliminary model building analysis indicate that
  - all three parameters vary with Wafer
  - intercept and linear term vary with Site

lme fit of Wafer data

```
> Waf0.6.lme1 <- lme(current ~ voltage + I(voltage^2),
+   data = Waf0.6,
+   random = list(Wafer = voltage + I(voltage^2), Site = voltage))
> Waf0.6.lme1
. . .
 Level 1: Wafer
 Formula: ~ voltage + I(voltage^2)
 Structure: Unstructured, Matrix Logarithm parametrization
 Standard Deviations:
 (Intercept)       voltage   I(voltage^2)
   0.1592957     0.3652543     0.04496319
 Correlation:
            (Intercept)     voltage
     voltage -0.9510162
I(voltage^2)  0.5555595  -0.7853729

 Level 2: Site within Wafer
 Formula: ~ voltage
 Structure: Unstructured, Matrix Logarithm parametrization
 Standard Deviations:
 (Intercept)      voltage
   0.0812176     0.146277
 Correlation:
      (Intercept)
voltage 0.8768002

Within-Group Standard Error: 0.1075564

Fixed Effects Estimate(s):
  (Intercept)      voltage I(voltage^2)
   -4.373939      5.782387     1.162791
. . .
```

Checking the lme fit

```
> plot(Waf0.6.lme1, resid(., type = "s") ~ voltage | Wafer)
# included loess curve in panels to enhance visualization
```



Checking the lme fit (cont.'d)

- Residual plots inidicate a periodic pattern, similar for all Wafers

- Best approach: use nlme model to estimate parameters related to periodic behavior

- To keep using lme model we assumed the period known (non-linear LS estimate = 1.37 V). Fixed effects component of the model:

```
current ~ voltage + I(voltage^2) +
           cos(2 * pi * voltage / 1.37) +
           sin(2 * pi * voltage / 1.37)
```

- After a few model building steps, included random effects sin and cos terms at the Wafer level

Checking the lme fit (cont.'d)

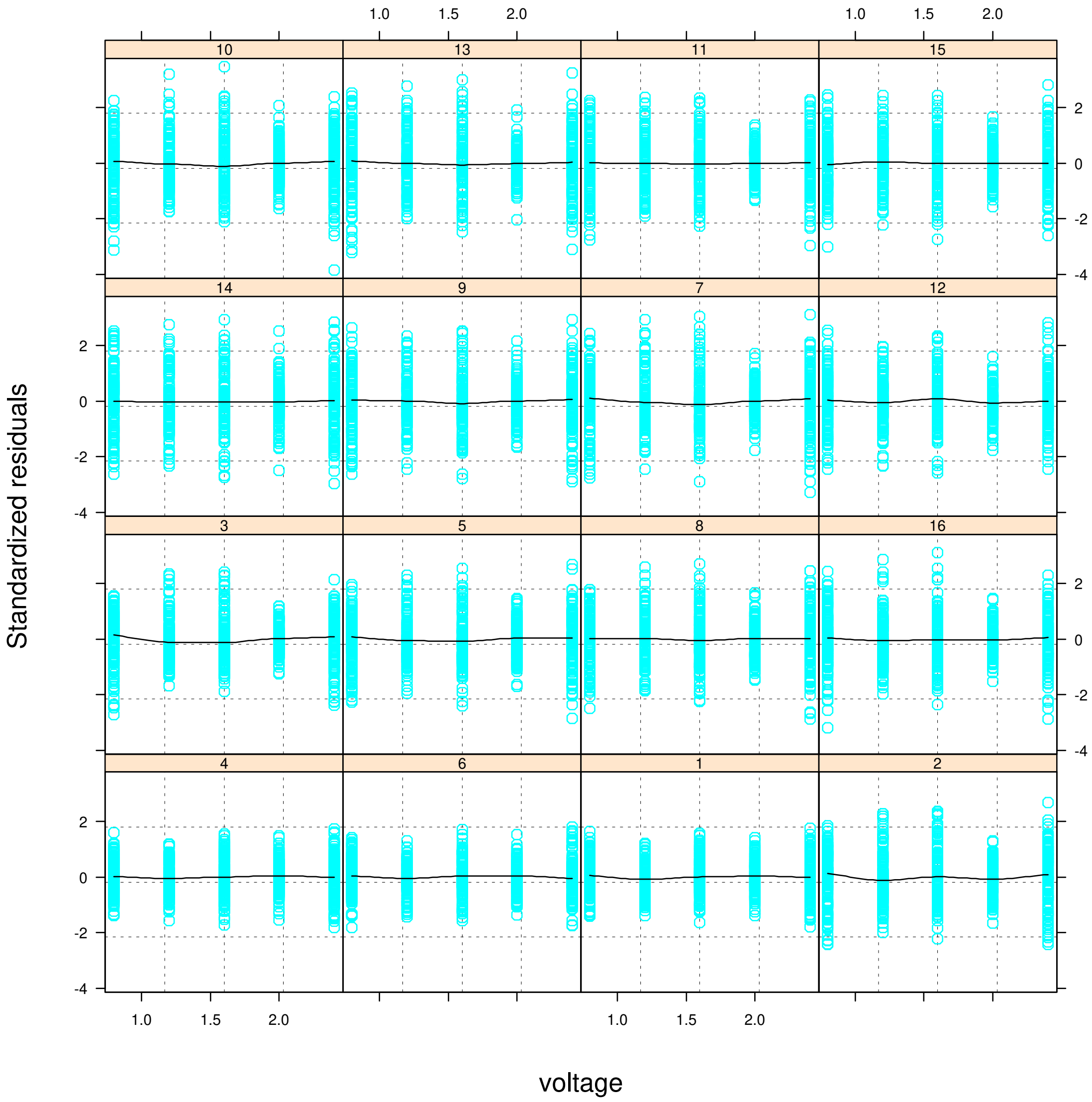
- These random effects were independent, with same variance and independent of the other Wafer random effects:

```
Wafer = pdBlocked(list(~ voltage + I(voltage^2),
   varIdent(~ cos(2 * pi * voltage / 1.37) +
            sin(2 * pi * voltage / 1.37) - 1)))
```

Checking the lme fit (cont.'d)

Corresponding fit in object Waf0.6.lme3

```
> anova(Waf0.6.lme1, Waf0.6,lme2)
            df    AIC    BIC logLik   Test Lik.Ratio P-value
Waf0.6.lme1 13 -12303 -12208   6165
Waf0.6.lme2 16 -34135 -34019  17083 1 vs. 2     21838       0
```

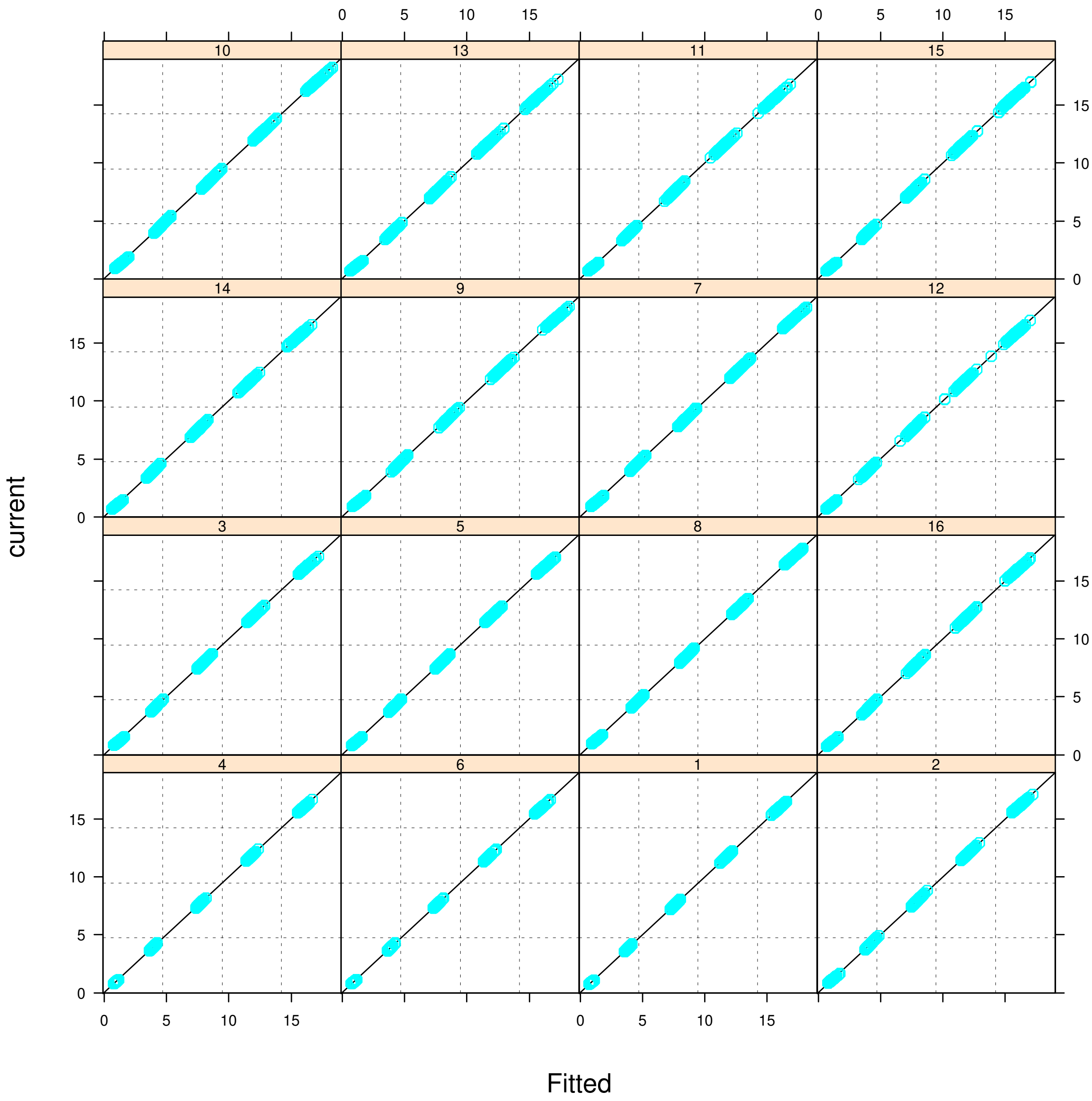


Checking the lme fit (cont.'d)

```
> Waf0.6.lme2
. . .
 Level 1: Wafer
 Composite Structure: Blocked

 Block 1: (Intercept), voltage, I(voltage^2)
 Standard Deviations:
 (Intercept)      voltage   I(voltage^2)
   0.1664647     0.3765405     0.04812175
 Correlation:
            (Intercept)     voltage
     voltage -0.9552292
I(voltage^2)  0.5996369  -0.8095647

 Block 2: cos((2 * pi * voltage)/1.37), sin((2 * pi * voltage)/1.37)
 Standard Deviations:
 cos((2 * pi * voltage)/1.37) sin((2 * pi * voltage)/1.37)
             0.003952008               0.003952008

 Level 2: Site within Wafer
 Standard Deviations:
 (Intercept)      voltage
   0.1271616     0.157153
 Correlation:
      (Intercept)
voltage 0.2523868

Within-Group Standard Error: 0.02995001

Fixed Effects Estimate(s):
  (Intercept)      voltage I(voltage^2) cos((2 * pi * voltage)/1.37)
   -4.173613      5.506798     1.249767                  -0.09589532
 sin((2 * pi * voltage)/1.37)
                0.09977845
```

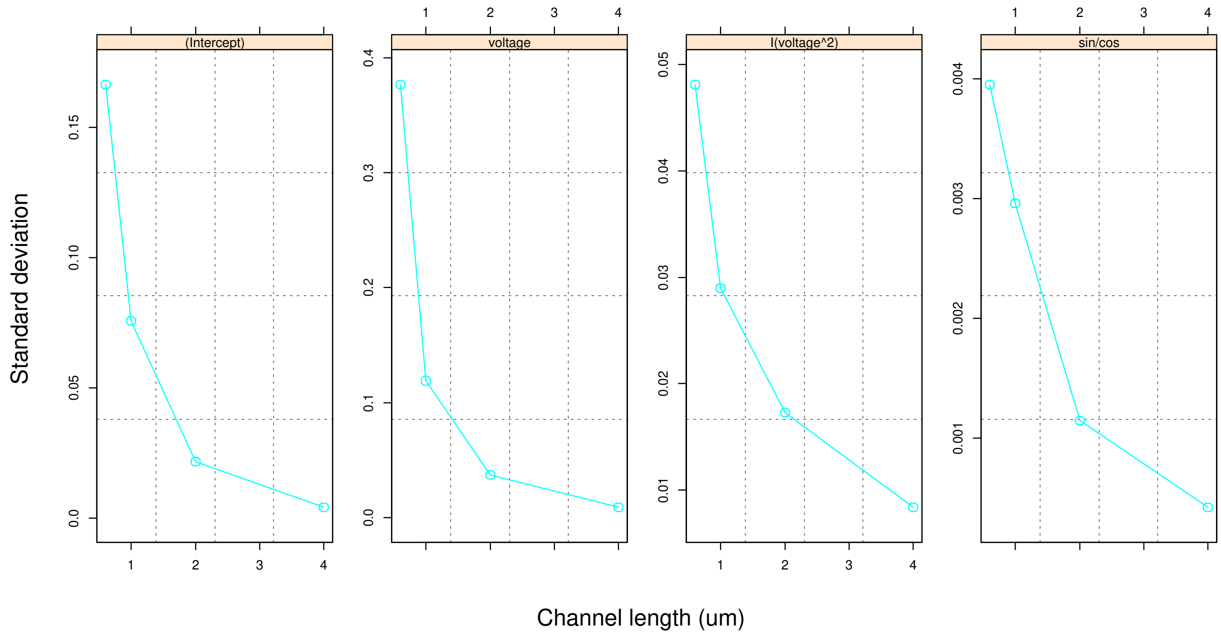
Checking the lme fit (cont.'d)

```
> plot(Waf0.6.lme2, current ~ fitted(.) | Wafer)
```

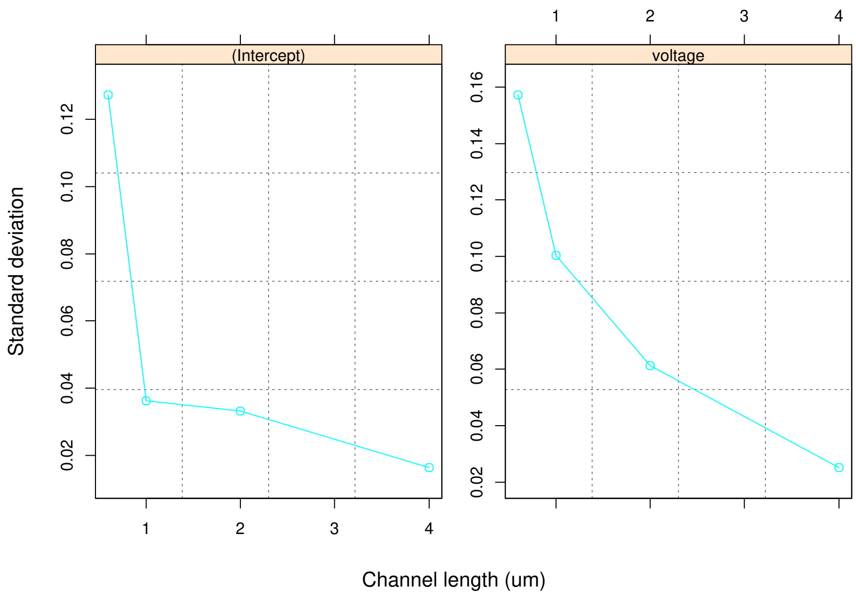


Comparing the lme fits by length

Wafer standard deviations

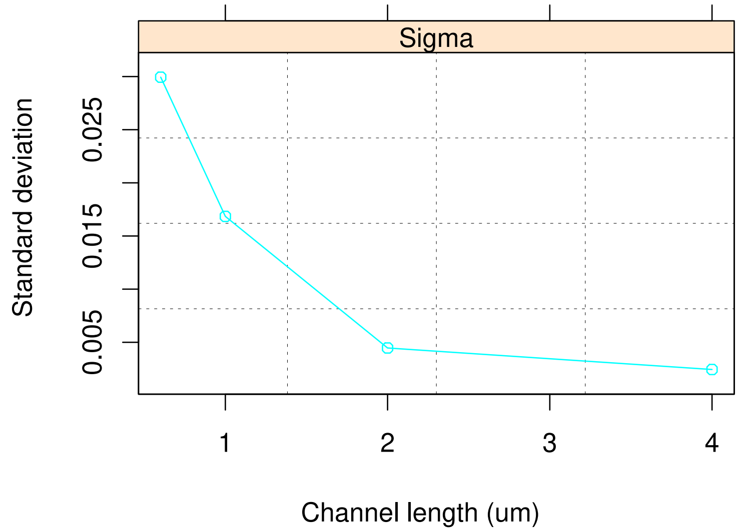


Site within Wafer standard deviations



Comparing lme fits (cont.'d)

Within-device standard deviations



- Variability decreases with channel length for all random effects and the within-device error
- Decrease patterns are similar, suggesting a common model form for the variance components
- Periodic component is observed for all lengths, with similar period but decreasing amplitude

Conclusions and future directions

- NLME 3.0 is described and documented in the book Mixed-Effects Models in S-PLUS by Bates & Pinheiro, to be published in 1998
- beta version of the LME part of NLME 3.0 to be released until end of 1997
- under development:
  - nlme algorithms, especially multilevel nlme
  - methods for prediction and inference (confidence intervals and hypothesis testing)

Conclusions (cont.'d)

- for the "not-near" future
  - alternative approximations for log-likelihood in nlme
  - generalized linear mixed-effects models: glme ?