Chapter 1

Appendix C

Roy (2009) presents two nested models that specify the condition of equality as required, with a third nested model for an additional test. There three formulations share the same structure, and can be specified by making slight alterations of the code for the Reference Model.

1.0.1 Model Fit 1

This is a simple model with no interactions. There is a fixed effect for each method and a random effect for each subject.

$$y_{ijk} = \beta_j + b_i + \epsilon_{ijk}, \qquad i = 1, \dots, 2, j = 1, \dots, 85, k = 1, \dots, 3$$

$$b_i \sim \mathcal{N}(0, \sigma^2_b), \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

```
> Ref.Fit = lme(y ~ meth-1, data = dat,   #Symm , Symm#
+     random = list(item=pdSymm(~ meth-1)),
+     weights=varIdent(form=~1|meth),
+     correlation = corSymm(form=~1 | item/repl),
+     method="ML")
```

1.1 Test 1

The first of Roy's candidate model can be implemented using the following code;

```
ref.nlme = lme(BP ~ method-1, data = dat,
random =  list(subject=pdSymm(~ method-1)),
weights=varIdent(form=~1|method),
correlation = corSymm(form=~1 | subject/obs), method="ML")
```

For the blood pressure data used in Roy (2009), all four candidate models are implemented by slight variations of this piece of code, specifying either `pdSymm` or `pdCompSymm` in the second line, and either `corSymm` or `corCompSymm` in the fourth line.

1.1.1 Nested Model (Between-Item Variability)

```
> NMB.fit  = lme(y ~ meth-1, data = dat,   #CS , Symm#
+     random = list(item=pdCompSymm(~ meth-1)),
+     correlation = corSymm(form=~1 | item/repl),
+     method="ML")
```

For example, the first test model '`test1.nlme`' is implemented with the same code as `ref.nlme`, except for the term `pdCompSymm` in the second line, rather than `pdSymm`.

```
test1.nlme = lme(BP ~ method-1, data = dat,
random = list(subject=pdCompSymm(~ method-1)),
weights = varIdent(form=~1|method),
correlation = corSymm(form=~1 | subject/obs), method="ML")
```

1.1.2 Roy’s Reference Model

Conventionally LME models can be tested using Likelihood Ratio Tests, wherein a reference model is compared to a nested model.

Roy (2009) presents two nested models that specify the condition of equality as required, with a third nested model for an additional test. There three formulations share the same structure, and can be specified by making slight alterations of the code for the Reference Model.

1.2 Roy’s Candidate Models

```
> Ref.Fit = lme(y ~ meth-1, data = dat,   #Symm , Symm#
+     random = list(item=pdSymm(~ meth-1)),
+     weights=varIdent(form=~1|meth),
+     correlation = corSymm(form=~1 | item/repl),
+     method="ML")
```

1.3 Roy’s Candidate Models

```
> Ref.Fit = lme(y ~ meth-1, data = dat,   #Symm , Symm#
+     random = list(item=pdSymm(~ meth-1)),
+     weights=varIdent(form=~1|meth),
+     correlation = corSymm(form=˜1 | item/repl),
+     method="ML")
```

Roy(2009) presents two nested models that specify the condition of equality as required, with a third nested model for an additional test. There three formulations share the same structure, and can be specified by making slight alterations of the code for the Reference Model.

Roy(2009) presents two nested models that specify the condition of equality as required, with a third nested model for an additional test. There three formulations share the same structure, and can be specified by making slight alterations of the code for the Reference Model. Nested Model (Between-Item Variability)

```
> NMB.fit  = lme(y ˜ meth-1, data = dat,   #CS , Symm#
+     random = list(item=pdCompSymm(˜ meth-1)),
+     correlation = corSymm(form=˜1 | item/repl),
+     method="ML")
```

1.3.1 Nested Model (Within item Variability)

```
> NMW.fit = lme(y ˜ meth-1, data = dat,   #Symm , CS#
+     random = list(item=pdSymm(˜ meth-1)),
+     weights=varIdent(form=˜1|meth),
+     correlation = corCompSymm(form=˜1 | item/repl),
+     method="ML")
```

1.3.2 Nested Model (Overall Variability)

Additionally there is a third nested model, that can be used to test overall variability, substantively a a joint test for between-item and within-item variability. The motivation for including such a test in the suite is not clear, although it does circumvent the need for multiple comparison procedures in certain circumstances, hence providing a simplified procedure for non-statisticians.

```
> NMO.fit = lme(y ~ meth-1, data = dat,   #CS , CS#
+     random = list(item=pdCompSymm(~ meth-1)),
+     correlation = corCompSymm(form=~1 | item/repl),
+     method="ML")
```

1.3.3 Model Fit 1

This is a simple model with no interactions. There is a fixed effect for each method and a random effect for each subject.

$$y_{ijk} = \beta_j + b_i + \epsilon_{ijk}, \qquad i = 1, \ldots, 2, j = 1, \ldots, 85, k = 1, \ldots, 3$$

$$b_i \sim \mathcal{N}(0, \sigma_b^2), \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

```
> Ref.Fit = lme(y ~ meth-1, data = dat,   #Symm , Symm#
+     random = list(item=pdSymm(~ meth-1)),
+     weights=varIdent(form=~1|meth),
+     correlation = corSymm(form=~1 | item/repl),
+     method="ML")
```

```
Linear mixed-effects model fit by REML

Data: dat

Log-restricted-likelihood: -2155.853

Fixed: BP ~ method

(Intercept)     methodS

127.40784    15.61961


Random effects:

Formula: ~1 | subject

(Intercept) Residual

StdDev:    29.39085 12.44454


Number of Observations: 510

Number of Groups: 85
```

The following output was obtained.

```
Linear mixed-effects model fit by REML

Data: dat

Log-restricted-likelihood: -2047.714

Fixed: BP ~ method

(Intercept)     methodS

127.40784    15.61961


Random effects:

Formula: ~1 | subject

(Intercept)

StdDev:    28.28452


Formula: ~1 | method %in% subject
```

```
(Intercept) Residual
StdDev:    12.61562 7.763666


Number of Observations: 510
Number of Groups:
subject method %in% subject
85                 170
```

Using this R implementation for other data sets requires that the data set is structured appropriately (i.e. each case of observation records the index, response, method and replicate). Once formatted properly, implementation is simply a case of re-writing the first line of code, and computing the four candidate models accordingly.

A likelihood ratio test is perform to determine which model is more suitable. To perform this test, simply use the anova command with the names of the candidate models as arguments. The following piece of code implement the first of Roy's variability tests.

```
> anova(MCS1,MCS2)
Model df     AIC    BIC  logLik   Test L.Ratio p-value
MCS1     1  8 4077.5 4111.3 -2030.7
MCS2     2  7 4075.6 4105.3 -2030.8 1 vs 2 0.15291  0.6958
>
```

The fixed effects estimates are the same for all four candidate models. The inter-method bias can be easily determined by inspecting a summary of any model. The summary presents estimates for all of the important parameters, but not the complete variance-covariance matrices (although some simple R functions can be written to overcome this). The variance estimates for the random effects for MCS2 is presented below.

```
Random effects:
```

```
Formula: ~method - 1 | subject

Structure: Compound Symmetry

StdDev Corr

methodJ  30.765

methodS  30.765 0.829

Residual  6.115
```

Similarly, for computing the limits of agreement the standard deviation of the differences is not explicitly given. Again, A simple R function can be written to calculate the limits of agreement directly.

1.3.4 Variability test 1

This is a test on whether both methods A and B have the same between-subject variability or not.

$$H_0 : \; d_A = d_B \tag{1.1}$$

$$H_A : \; d_A \neq d_B \tag{1.2}$$

When implemented using R, this test is facilitated by constructing a model specifying a symmetric form for D (i.e. the alternative model) and comparing it with a model that has compound symmetric form for D (i.e. the null model). For this test $\hat{\boldsymbol{\Sigma}}$ has a symmetric form for both models, and will be the same for both.

1.4 Model Fit 2

Nested Model (Between-Item Variability)

```
> NMB.fit  = lme(y ~ meth-1, data = dat,   #CS , Symm#
+     random = list(item=pdCompSymm(~ meth-1)),
+     correlation = corSymm(form=~1 | item/repl),
+     method="ML")
```

This is a simple model, this time with an interaction effect. There is a fixed effect for each method. This model has random effects at two levels b_i for the subject, and another, b_{ij}, for the respective method within each subject.

$$y_{ijk} = \beta_j + b_i + b_{ij} + \epsilon_{ijk}, \qquad i = 1, \dots, 2, j = 1, \dots, 85, k = 1, \dots, 3$$

$$b_i \sim \mathcal{N}(0, \sigma_1^2), \qquad b_{ij} \sim \mathcal{N}(0, \sigma_2^2), \qquad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$$

In this model, the random interaction terms all have the same variance σ_2^2. These terms are assumed to be independent of each other, even within the same subject.

```
Linear mixed-effects model fit by REML
Data: dat
Log-restricted-likelihood: -2047.714
Fixed: BP ~ method
(Intercept)     methodS
127.40784    15.61961

Random effects:
Formula: ~1 | subject
(Intercept)
StdDev:    28.28452

Formula: ~1 | method %in% subject
(Intercept) Residual
StdDev:    12.61562 7.763666

Number of Observations: 510
Number of Groups:
subject method %in% subject
85                  170
```

1.4.1 Model Fit 3

This model is a more general model, compared to 'model fit 2'. This model treats the random interactions for each subject as a vector and allows the variance-covariance matrix for that vector to be estimated from the set of all positive-definite matrices. $\boldsymbol{y_i}$ is the entire response vector for the ith subject. $\boldsymbol{X_i}$ and $\boldsymbol{Z_i}$ are the fixed- and random-effects design matrices respectively.

$$\boldsymbol{y_i} = \boldsymbol{X_i}\boldsymbol{\beta} + \boldsymbol{Z_i}\boldsymbol{b_i} + \boldsymbol{\epsilon_i}, \qquad i = 1, \ldots, 85$$

$$\boldsymbol{Z_i} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Psi}), \qquad \boldsymbol{\epsilon_i} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\sigma^2 \Lambda})$$

Bibliography

Roy, A. (2009). An application of the linear mixed effects model to ass the agreement between two methods with replicated observations. *Journal of Biopharmaceutical Statistics 19*, 150–173.