Data Analysis & Graphics Using R – Solutions to Exercises (April 24, 2004)

Preliminaries

```
> library(nlme)
> library(DAAG)
```

The final two sentences of Exercise 1 are challenging! Exercises 1 & 2 should be asterisked.

Exercise 1

Repeat the calculations of Subsection 9.3.5, but omitting results from two vines at random. Here is code that will handle the calculation:

```
n.omit <- 2
take <- rep(T, 48)
take[sample(1:48, n.omit)] <- F
kiwishade.lme <- lme(yield ~ shade, random = ~1 | block/plot,
                     data = kiwishade,subset=take)
VarCorr(kiwishade.lme)[4, 1]  # Plot component of variance
VarCorr(kiwishade.lme)[5, 1]  # Vine component of variance
```

Repeat this calculation five times, for each of n.omit = 2, 4, 6, 8, 10, 12 and 14. Plot (i) the plot component of variance and (ii) the vine component of variance, against number of points omitted. Based on these results, for what value of `n.omit` does the loss of vines begin to compromise results? Which of the two components of variance estimates is more damaged by the loss of observations? Comment on why this is to be expected.

For convenience, we place the central part of the calculation in a function. On slow machines, the code may take a minute or two to run.

```
> trashvine <- function(n.omit = 2) {
+     k <- k + 1
+     n[k] <- n.omit
+     take <- rep(T, 48)
+     take[sample(1:48, n.omit)] <- F
+     kiwishade$take <- take
+     kiwishade.lme <- lme(yield ~ shade, random = ~1 | block/plot,
+         data = kiwishade, subset = take)
+     varp <- as.numeric(VarCorr(kiwishade.lme)[4, 1])
+     varv <- as.numeric(VarCorr(kiwishade.lme)[5, 1])
+     c(varp, varv)
+ }
> varp <- numeric(35)
> varv <- numeric(35)
> n <- numeric(35)
> k <- 0
> data(kiwishade)
> for (n.omit in c(2, 4, 6, 8, 10, 12, 14)) for (i in 1:5) {
+     k <- k + 1
+     vec2 <- trashvine(n.omit = n.omit)
+     n[k] <- n.omit
```

```
+      varp[k] <- vec2[1]
+      varv[k] <- vec2[2]
+ }
```

We plot the results:

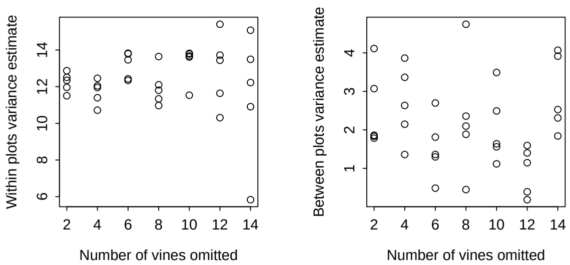

Figure 1: Within, and between plots variance estimates, as functions of the number of vines that were omitted at random

As the number of vines that are omitted increases, the variance estimates can be expected to show greater variability. The fraction omitted may not be large enough for the effect to show clearly. Increasing the number of repeats for each value of `n.omit` would help. The effect should be most evident on the between plot variance. Inaccuracy in estimates of the between plot variance arise both from inaccuracy in the within plot sums of squares and from loss of information at the between plot level.

At best it is possible only to give an approximate d.f. for the between plot estimate of variance (some plots lose more vines than others), which complicates any evaluation that relies on degree of freedom considerations.

Exercise 2
Repeat the previous exercise, but now omitting 1, 2, 3, 4 complete plots at random.

```
> trashplot <- function(n.omit = 2) {
+      k <- k + 1
+      n[k] <- n.omit
+      plotlev <- levels(kiwishade$plot)
+      use.lev <- sample(plotlev, length(plotlev) - n.omit)
+      kiwishade$take <- kiwishade$plot %in% use.lev
+      kiwishade.lme <- lme(yield ~ shade, random = ~1 | block/plot,
+          data = kiwishade, subset = take)
+      varp <- as.numeric(VarCorr(kiwishade.lme)[4, 1])
+      varv <- as.numeric(VarCorr(kiwishade.lme)[5, 1])
+      c(varp, varv)
+ }
> varp <- numeric(20)
> varv <- numeric(20)
> n <- numeric(20)
> k <- 0
> for (n.omit in 1:4) for (i in 1:5) {
+      k <- k + 1
+      vec2 <- trashplot(n.omit = n.omit)
+      n[k] <- n.omit
+      varp[k] <- vec2[1]
```

```
+     varv[k] <- vec2[2]
+ }
```

Again, we plot the results:

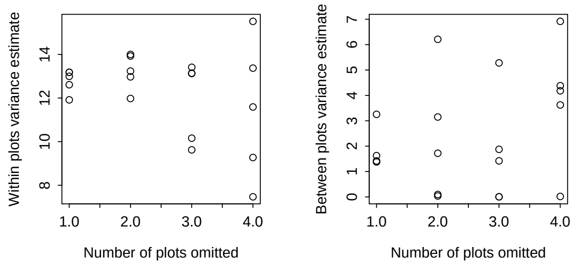

Figure 2: Within, and between plots variance estimates, as functions of the number of whole plots (each consisting of four vines) that were omitted at random.

Omission of a whole plot loses 3 d.f. out of 36 for estimation of within plot effects, and 1 degree of freedom out of 11 for the estimation of between plot effects, i.e., a slightly greater relative loss. The effect on precision will be most obvious where the d.f. are already smallest, i.e., for the between plot variance. The loss of information on complete plots is inherently for serious, for the estimation of the between plot variance, than the loss of partial information (albeit on a greater number of plots) as will often happen in Exercise 1.

Exercise 3
The final sentence has been modified; see the list of Corrections
A time series of length 100 is obtained from an AR(1) model with $\sigma = 1$ and $\alpha = -.5$. What is the standard error of the mean? If the usual $\sigma/\sqrt{n}$ formula were used in constructing a confidence interval for the mean, with σ defined as in Section 9.5.3, would it be too narrow or too wide?

If we know σ, then the usual $\sigma/\sqrt{n}$ formula will give an error that is too narrow; refer back to Subsection 9.5.3 on page 244.

The need to estimate σ raises an additional complication. If σ is estimated by fitting a time series model, e.g., using the function **ar()**, this estimate of σ can be plugged into the formula in Subsection 9.5.3. The note that now follows covers the case where σ^2 is estimated using the formula

$$\hat{\sigma^2} = \frac{\sum(X_i - \bar{X})^2}{n-1}$$

The relevant theoretical results are not given in the text. Their derivation requires a kmowledge of the algebra of expectations.

Note 1: We use the result (proved below)

$$E[(X_i - \mu)^2] = \sigma^2/(1 - \alpha^2) \tag{1}$$

and that

$$E[\sum(X_i - \bar{X})^2] = \frac{1}{1-\alpha^2}(n-1-\alpha)\sigma^2 \simeq \frac{1}{1-\alpha^2}(n-1)\sigma^2 \tag{2}$$

Hence, if the variance is estimated from the usual formula $\hat{\sigma^2} = \frac{\sum(X_i-\bar{X})^2}{n-1}$, the standard error of the mean will be too small by a factor of approximately $\sqrt{\frac{1-\alpha}{1+\alpha}}$.

Note 2: We square both sides of

$$X_t - \mu = \alpha(X_{t-1} - \mu) + \varepsilon_t$$

and take expectations. We have that

$$E[(X_t - \mu)^2] = (1 - \alpha^2)E[(X_t - \mu)^2] + \sigma^2$$

from which the result (eq.1) follows immediately. To derive $E[\sum(X_i - \bar{X})^2]$, observe that

$$E[\sum(X_i - \bar{X})^2] = E[(X_t - \mu)^2] - n(\bar{X} - \mu)^2$$

Exercise 4
Use the `ar` function to fit the second order autoregressive model to the Lake Huron time series.

```
> if (!exists("LakeHuron")) data(LakeHuron)
> ar(LakeHuron, order.max = 2)

Call:
ar(x = LakeHuron, order.max = 2)

Coefficients:
     1       2
 1.054  -0.267

Order selected 2  sigma^2 estimated as  0.508
```

It might however be better not to specify the order, instead allowing the `ar()` function to choose it, based on the AIC criterion. For this to be valid, it is best to specify also `method="mle"`. Fitting by maximum likelihood can for long series be very slow. It works well in this instance.

```
> ar(LakeHuron, method = "mle")

Call:
ar(x = LakeHuron, method = "mle")

Coefficients:
     1       2
 1.044  -0.250

Order selected 2  sigma^2 estimated as  0.479
```

The AIC criterion chooses the order equal to 2.

Exercise 5
The data set `Gun` (*nlme* package) reports on the numbers of rounds fired per minute, by each of nine teams of gunners, each tested twice using each of two methods. In the nine teams, three were made of men with slight build, three with average, and three with heavy build. Is there a detectable difference, in number of rounds fired, between build type or between firing methods? For improving the precision of results, which would be better – to double the number of teams, or to double the number of occasions (from 2 to 4) on which each team tests each method?

It probably does not make much sense to look for overall differences in `Method`; this depends on `Physique`. We therefore nest `Method` within `Physique`.

```
> if (!exists("Gun")) data(Gun)
> Gun.lme <- lme(rounds ~ Physique/Method, random = ~1 | Team,
+     data = Gun)
> summary(Gun.lme)

Linear mixed-effects model fit by REML
 Data: Gun
    AIC   BIC logLik
  143.0 154.2 -63.48

Random effects:
 Formula: ~1 | Team
        (Intercept) Residual
StdDev:       1.044    1.476

Fixed effects: rounds ~ Physique/Method
                         Value Std.Error DF t-value p-value
(Intercept)             23.589    0.4922 24   47.92  0.0000
Physique.L              -0.966    0.8526  6   -1.13  0.3003
Physique.Q               0.191    0.8526  6    0.22  0.8306
PhysiqueSlight:MethodM2  -8.450    0.8524 24   -9.91  0.0000
PhysiqueAverage:MethodM2 -8.100    0.8524 24   -9.50  0.0000
PhysiqueHeavy:MethodM2   -8.983    0.8524 24  -10.54  0.0000
 Correlation:
                         (Intr) Phys.L Phys.Q PS:MM2 PA:MM2
Physique.L               0.000
Physique.Q               0.000  0.000
PhysiqueSlight:MethodM2  -0.289  0.353 -0.204
PhysiqueAverage:MethodM2 -0.289  0.000  0.408  0.000
PhysiqueHeavy:MethodM2   -0.289 -0.353 -0.204  0.000  0.000

Standardized Within-Group Residuals:
     Min       Q1      Med       Q3      Max
-2.15598 -0.64718  0.09983  0.63379  1.67448

Number of Observations: 36
Number of Groups: 9
```

A good way to proceed is to determine the fitted values, and present these in an interaction plot:

```
> Gun.hat <- predict(Gun.lme)
> interaction.plot(Gun$Physique, Gun$Method, Gun.hat)
```

Differences between methods, for each of the three physiques, are strongly attested. These can be estimated within teams, allowing 24 degrees of freedom for each of these comparisons.

Clear patterns of change with `Physique` seem apparent in the plot. There are however too few degrees of freedom for this effect to appear statistically significant. Note however that the parameters that are given are for the lowest level of `Method`, i.e., for `M1`. Making `M2` the baseline shows the effect as closer to the conventional 5% significance level.

The component of variance at the between teams level is of the same order of magnitude as the within teams component. Its contribution to the variance of team means (1.044^2) is much greater than the contribution of the within team component ($1.476^2/4$; there are 4 results per team). If comparison between physiques is the concern; it will be much more effective to double the number of teams; compare $(1.044^2+1.476^2/4)/2$ (=0.82) with $1.044^2+1.476^2/8$ (=1.36).

Exercise 6
*The data set `ergoStool` (*nlme* package) has data on the amount of effort needed to get up from a stool, for each of nine individuals who each tried four different types of stool. Analyse the data both using `aov()` and using `lme()`, and reconcile the two sets of output. Was there any clear winner among the types of stool, if the aim is to keep effort to a minimum?

For analysis of variance, specify

```
> if (!exists("ergoStool")) data(ergoStool)
> aov(effort ~ Type + Error(Subject), data = ergoStool)

Call:
aov(formula = effort ~ Type + Error(Subject), data = ergoStool)

Grand Mean: 10.25

Stratum 1: Subject

Terms:
                Residuals
Sum of Squares       66.5
Deg. of Freedom         8

Residual standard error: 2.883

Stratum 2: Within

Terms:
                 Type Residuals
Sum of Squares  81.19     29.06
Deg. of Freedom     3        24

Residual standard error: 1.100
Estimated effects may be unbalanced
```

For testing the `Type` effect for statistical significance, refer (81.19/3)/(29.06/24) (=22.35) with the $F_{3,24}$ distribution. The effect is highly significant.

This is about as far as it is possible to go with analysis of variance calculations. When `Error()` is specified in the `aov` model, R has no mechanism for extracting estimates. (There are mildly tortuous ways to extract the information, which will not be further discussed here.)

For use of `lme`, specify

```
> summary(lme(effort ~ Type, random = ~1 | Subject, data = ergoStool))
```

```
Linear mixed-effects model fit by REML
 Data: ergoStool
    AIC   BIC logLik
  133.1 141.9 -60.57

Random effects:
 Formula: ~1 | Subject
        (Intercept) Residual
StdDev:       1.332    1.100

Fixed effects: effort ~ Type
            Value Std.Error DF t-value p-value
(Intercept) 8.556    0.5760 24  14.853  0.0000
TypeT2      3.889    0.5187 24   7.498  0.0000
TypeT3      2.222    0.5187 24   4.284  0.0003
TypeT4      0.667    0.5187 24   1.285  0.2110
 Correlation:
       (Intr) TypeT2 TypeT3
TypeT2 -0.45
TypeT3 -0.45   0.50
TypeT4 -0.45   0.50   0.50

Standardized Within-Group Residuals:
     Min       Q1      Med       Q3      Max
-1.80200 -0.64317  0.05783  0.70100  1.63142

Number of Observations: 36
Number of Groups: 9
```

Observe that 1.100295^2 (Residual StdDev) is very nearly equal to 29.06/24 obtained from the analysis of variance calculation.

Also the Stratum 1 mean square of 66.5/8 (=8.3125) from the analysis of variance output is very nearly equal to $1.3325^2 + 1.100295^2/4$ (= 2.078) from the `lme` output.

Exercise 7
*In the data set `MathAchieve` (*nlme* package), the factors `Minority` (levels `yes` and `no`) and `sex`, and the variable `SES` (socio-economic status) are clearly fixed effects. Discuss how the decision whether to treat `School` as a fixed or as a random effect might depend on the purpose of the study? Carry out an analysis that treats `School` as a random effect. Are differences between schools greater than can be explained by within school variation?

School should be treated as a random effect if the intention is to generalize results to other comparable schools. If the intention is to apply them to other pupils or classess within those same schools, it should be taken as a fixed effect.

For the analysis of these data, both SES and MEANSES should be included in the model. Then the coefficient of MEANSES will measure between school effects, while the coefficient of SES will measure within school effects.

```
> if (!exists("MathAchieve")) data(MathAchieve)
> MathAch.lme <- lme(MathAch ~ Minority * Sex * (MEANSES + SES),
+     random = ~1 | School, data = MathAchieve)
> summary(MathAch.lme)
```

```
Linear mixed-effects model fit by REML
 Data: MathAchieve
    AIC   BIC logLik
  46344 46441 -23158

Random effects:
 Formula: ~1 | School
        (Intercept) Residual
StdDev:       1.585    5.982

Fixed effects: MathAch ~ Minority * Sex * (MEANSES + SES)
                              Value Std.Error   DF t-value p-value
(Intercept)                  14.076    0.1863 7015   75.56  0.0000
MinorityYes                  -3.068    0.2798 7015  -10.96  0.0000
SexFemale                    -1.277    0.1862 7015   -6.86  0.0000
MEANSES                       2.811    0.5209  158    5.40  0.0000
SES                           1.992    0.1880 7015   10.59  0.0000
MinorityYes:SexFemale         0.462    0.3757 7015    1.23  0.2186
MinorityYes:MEANSES           0.726    0.6925 7015    1.05  0.2948
MinorityYes:SES              -0.990    0.3441 7015   -2.88  0.0040
SexFemale:MEANSES            -0.574    0.5740 7015   -1.00  0.3174
SexFemale:SES                 0.517    0.2643 7015    1.95  0.0507
MinorityYes:SexFemale:MEANSES 0.713    0.9034 7015    0.79  0.4299
MinorityYes:SexFemale:SES    -0.110    0.4683 7015   -0.24  0.8138
 Correlation:
                             (Intr) MnrtyY SexFml MEANSE SES    MnY:SF MY:MEA
MinorityYes                  -0.378
SexFemale                    -0.537  0.314
MEANSES                      -0.144  0.123  0.103
SES                          -0.109  0.078  0.107 -0.319
MinorityYes:SexFemale         0.234 -0.673 -0.433 -0.073 -0.053
MinorityYes:MEANSES           0.127 -0.002 -0.082 -0.518  0.244 -0.016
MinorityYes:SES               0.058  0.117 -0.065  0.182 -0.552 -0.084 -0.444
SexFemale:MEANSES             0.098 -0.089 -0.141 -0.580  0.293  0.092  0.389
SexFemale:SES                 0.071 -0.044 -0.081  0.230 -0.713  0.045 -0.173
MinorityYes:SexFemale:MEANSES -0.064 -0.021  0.096  0.329 -0.192  0.120 -0.662
MinorityYes:SexFemale:SES    -0.045 -0.088  0.056 -0.130  0.405  0.122  0.321
                             MY:SES SF:MEA SF:SES MY:SF:M
MinorityYes
SexFemale
MEANSES
SES
MinorityYes:SexFemale
MinorityYes:MEANSES
MinorityYes:SES
SexFemale:MEANSES            -0.161
SexFemale:SES                 0.392 -0.430
MinorityYes:SexFemale:MEANSES 0.336 -0.576  0.280
MinorityYes:SexFemale:SES    -0.733  0.241 -0.567 -0.473

Standardized Within-Group Residuals:
     Min       Q1      Med       Q3      Max
-3.25178 -0.72084  0.03174  0.75758  2.84514
```

```
Number of Observations: 7185
Number of Groups: 160
```

The between school component of variance (1.585^2) is 5.51, compared with a within school component that equals 35.79. To get a confidence intervals for the square roots of these variances, specify:

```
> intervals(MathAch.lme)

Approximate 95% confidence intervals

 Fixed effects:
                                lower    est.   upper
(Intercept)                 13.711281 14.0765 14.4417
MinorityYes                 -3.616381 -3.0679 -2.5194
SexFemale                   -1.642299 -1.2772 -0.9122
MEANSES                      1.781647  2.8105  3.8394
SES                          1.623267  1.9919  2.3604
MinorityYes:SexFemale       -0.274212  0.4623  1.1989
MinorityYes:MEANSES         -0.632041  0.7255  2.0830
MinorityYes:SES             -1.664930 -0.9904 -0.3160
SexFemale:MEANSES           -1.699294 -0.5740  0.5512
SexFemale:SES               -0.001503  0.5166  1.0348
MinorityYes:SexFemale:MEANSES -1.057808  0.7132  2.4841
MinorityYes:SexFemale:SES   -1.028380 -0.1103  0.8078
attr(,"label")
[1] "Fixed effects:"

 Random Effects:
  Level: School
                lower  est. upper
sd((Intercept)) 1.363 1.585 1.843

 Within-group standard error:
lower  est. upper
5.884 5.982 6.082
```

The 95% confidence interval for the between school component of variance ranges from 1.36 to 1.84. The confidence interval excludes 0. Try also

```
> intervals(MathAch.lme, level = 0.9999)

Approximate 99.99% confidence intervals

 Fixed effects:
                              lower    est.   upper
(Intercept)                 13.3512 14.0765 14.8018
MinorityYes                 -4.1571 -3.0679 -1.9786
SexFemale                   -2.0022 -1.2772 -0.5523
MEANSES                      0.7309  2.8105  4.8902
SES                          1.2599  1.9919  2.7238
MinorityYes:SexFemale       -1.0003  0.4623  1.9250
MinorityYes:MEANSES         -1.9703  0.7255  3.4214
```

```
MinorityYes:SES               -2.3299 -0.9904  0.3490
SexFemale:MEANSES             -2.8086 -0.5740  1.6606
SexFemale:SES                 -0.5123  0.5166  1.5456
MinorityYes:SexFemale:MEANSES -2.8037  0.7132  4.2300
MinorityYes:SexFemale:SES     -1.9335 -0.1103  1.7129
attr(,"label")
[1] "Fixed effects:"

 Random Effects:
  Level: School
                lower  est. upper
sd((Intercept)) 1.175 1.585 2.139

 Within-group standard error:
lower  est. upper
5.789 5.982 6.182
```

Zero is again excluded, still by a substantial margin.

The number of results for school varies between 14 and 67. Thus, the relative contribution to class means is 5.51 and a number that is at most $5.982429^2/14 = 2.56$.

Exercise 8
The function `Box.test()` (in *ts*) may be used to compare the the straight line model with uncorrelated errors that was fitted in Section 9.6 against the alternative of autocorrelation at some lag greater than zero. Try, e.g.,

```
Box.test(resid(lm(detrain ~ detSOI, data = detsoi)),
        type="Ljung-Box", lag=20)
```

It is necessary to guess at the highest possible lag at which an autocorrelation is likely. The number should not be too large; so that the flow-on effect from autocorrelation at lower lags is still evident. A common, albeit arbitrary choice, is a lag of 20, as here. Try running the test with `lag` set to values of 1 (the default), 15, 25 and 30. Comment on the different results.

The calculation for a lag of 20 was given on page 251. Here are the results for the other suggested lags:

```
> if (!exists("bomsoi")) data(bomsoi)
> detsoi <- data.frame(detSOI = bomsoi[, "SOI"] - lowess(bomsoi[,
+     "SOI"])$y, detrain = log(bomsoi$avrain - 250) - lowess(log(bomsoi$avrain -
+     250))$y)
> row.names(detsoi) <- paste(1900:2001)
> Box.test(resid(lm(detrain ~ detSOI, data = detsoi)), type = "Ljung-Box",
+     lag = 15)

        Box-Ljung test

data:  resid(lm(detrain ~ detSOI, data = detsoi))
X-squared = 30.11, df = 15, p-value = 0.01154

> Box.test(resid(lm(detrain ~ detSOI, data = detsoi)), type = "Ljung-Box",
+     lag = 25)
```

```
        Box-Ljung test

data:  resid(lm(detrain ~ detSOI, data = detsoi))
X-squared = 37.92, df = 25, p-value = 0.04706

> Box.test(resid(lm(detrain ~ detSOI, data = detsoi)), type = "Ljung-Box",
+     lag = 30)

        Box-Ljung test

data:  resid(lm(detrain ~ detSOI, data = detsoi))
X-squared = 47.18, df = 30, p-value = 0.02391
```

The p-values are:

```
    n=15        n=20        n=25         n=30
  0.0115     0.03325      0.0471       0.0239
```

The settings with n>15 allow for more possibilities, with an accordingly reduced probability of detection, than do larger values of n. The small p-value for n=30 is perhaps surprising.