```
> library(Stat5303libs);library(cfcdae);library(lme4)

> weardata <- read.table("wear.dat.txt",header=TRUE)
```

I'm embarassed to say, but I can't really remember where I got these data. I think from one of my professors back in the dark ages. The issue is wear loss of rubber, and we have seven different treatments for reducing the loss. A block is a chunk of rubber, but a block is only large enough for four of the seven treatments. Small losses are good.

```
> weardata
   block trt wear
1      2   1  344
2      4   1  337
3      6   1  369
4      7   1  396
5      1   2  627
6      4   2  537
7      5   2  520
8      7   2  602
9      2   3  233
10     3   3  251
11     5   3  278
12     7   3  240
13     1   4  248
14     3   4  211
15     6   4  196
16     7   4  273
17     3   5  160
18     4   5  195
19     5   5  199
20     6   5  185
21     1   6  563
22     2   6  442
23     5   6  595
24     6   6  606
25     1   7  252
26     2   7  226
27     3   7  297
28     4   7  300
> weardata <- within(weardata,{block <- as.factor(block);trt <- as.factor(trt)})
> with(weardata,table(block,trt))
```

The incidence matrix n_{ij}. Each block contains four treatments, and each pair of treatments occurs together twice.

```
     trt
block 1 2 3 4 5 6 7
    1 0 1 0 1 0 1 1
    2 1 0 1 0 0 1 1
    3 0 0 1 1 1 0 1
    4 1 1 0 0 1 0 1
    5 0 1 1 0 1 1 0
    6 1 0 0 1 1 1 0
    7 1 1 1 1 0 0 0
```

```
> 7*3/(6*4)
```

Relative efficience is 7/8.

```
[1] 0.875
```

```
> .875*4
```

Effective sample size for intrablock analysis is 3.5.

```
[1] 3.5
```

```
> fit1 <- lm(wear~block+trt,data=weardata)
```

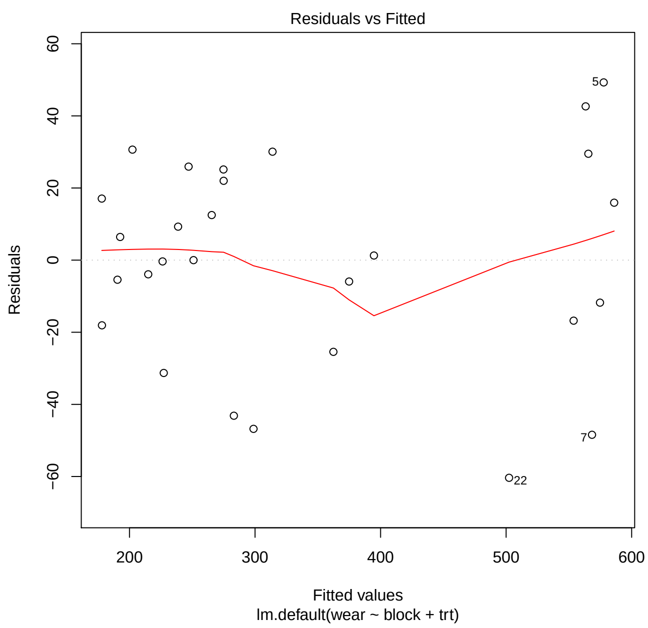
Basic model is treatments adjusted for blocks. This is the "intrablock" analysis.

```
Analysis of Variance Table

Response: wear
          Df Sum Sq Mean Sq F value    Pr(>F)
block      6  97395   16232  11.032 8.937e-05 ***
trt        6 506799   84466  57.404 1.687e-09 ***
Residuals 15  22071    1471
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

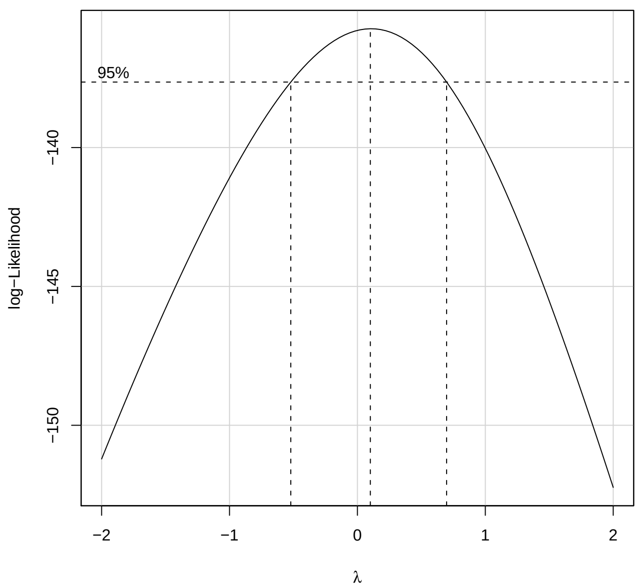
```
> plot(fit1,which=1)
```

Variance could be increasing with mean. Just the slightest touch of curvature.



```
> boxCox(fit1)
```

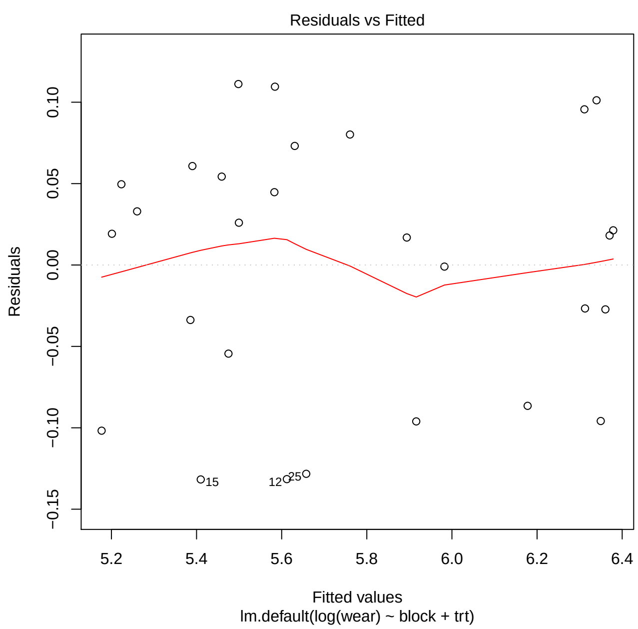
Box Cox suggests a log.




```
> fit2 <- lm(log(wear)~block+trt,data=weardata)
```

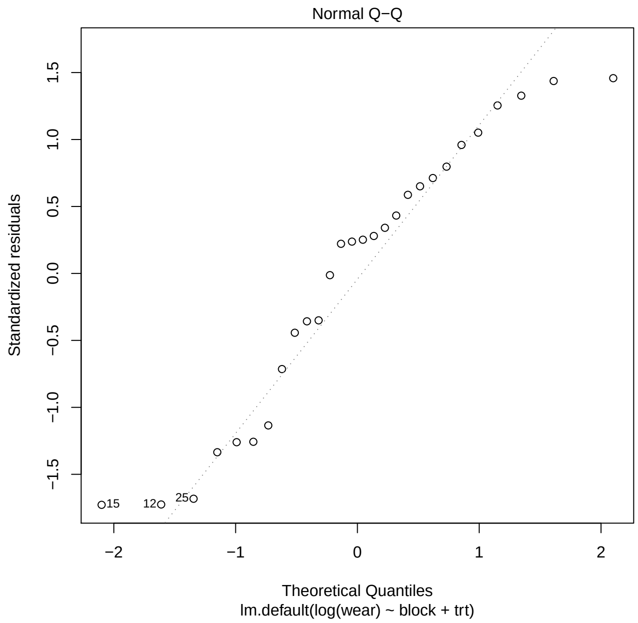
```
> plot(fit2,which=1)
```

Variance is more constant.



```
> plot(fit2,which=2)
```

Residuals are somewhat short tailed.




```
> anova(fit2)
```

We want treatments after blocks (or use Anova to get type II). Treatments are highly significant.

```
Analysis of Variance Table

Response: log(wear)
          Df Sum Sq Mean Sq F value    Pr(>F)
block      6 0.7784 0.12974  11.959 5.573e-05 ***
trt        6 3.9060 0.65100  60.006 1.230e-09 ***
Residuals 15 0.1627 0.01085
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1


> summary(fit2)

Call:
lm.default(formula = log(wear) ~ block + trt)

Residuals:
     Min       1Q   Median       3Q      Max
-0.13172 -0.06245  0.01866  0.05593  0.11117

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  5.757176   0.019684 292.480  < 2e-16 ***
block1       0.040494   0.051545   0.786 0.444326
block2      -0.142221   0.051545  -2.759 0.014614 *
block3      -0.033005   0.051545  -0.640 0.531628
```

```
block4       0.013441   0.051545   0.261 0.797822
block5       0.050401   0.051545   0.978 0.343675
block6      -0.008792   0.051545  -0.171 0.866844
trt1         0.145530   0.051545   2.823 0.012839 *
trt2         0.542077   0.051545  10.517 2.57e-08 ***
trt3        -0.224702   0.051545  -4.359 0.000561 ***
trt4        -0.338553   0.051545  -6.568 8.91e-06 ***
trt5        -0.547229   0.051545 -10.617 2.26e-08 ***
trt6         0.562861   0.051545  10.920 1.55e-08 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 0.1042 on 15 degrees of freedom
Multiple R-squared: 0.9664,Adjusted R-squared: 0.9396
F-statistic: 35.98 on 12 and 15 DF,  p-value: 7.699e-09
```

```
> model.effects(fit2,"trt")
```

All the usual things such as coefficients, pairwise comparisons, contrasts, etc work.

```
         1          2          3          4          5          6
 0.1455303  0.5420770 -0.2247025 -0.3385527 -0.5472291  0.5628608
         7
-0.1399839
```

```
> linear.contrast(fit2,trt,c(1,-1,0,0,0,0,0))
   estimates         se   t-value      p-value   lower-ci   upper-ci
1 -0.3965467 0.07873613 -5.036401 0.0001475959 -0.5643688 -0.2287246
```

```
> sqrt(.010849*(1/4 + (-1)^2/4));sqrt(.010849*(1/3.5 + (-1)^2/3.5))
```

Here we have two attempts to compute the standard error of a pairwise difference. The first is a naive application of the standard formula; this gives the incorrect answer. The second version uses the effective sample size (3.5) and gets the correct answer.

```
[1] 0.0736512
[1] 0.07873645
```

```
> fit3 <- lmer(log(wear)~trt+(1|block),data=weardata)
```

If you treat blocks as random and fit the model, then you get interblock recovery.

```
> fit3
```

Note that the standard errors for treatment effects are (slightly) smaller here. That is the result of incorporating the interblock information. However, even with this relatively small block to block variance, the improvement from interblock recovery is small.

```
Linear mixed model fit by REML ['lmerMod']
Formula: log(wear) ~ trt + (1 | block)
   Data: weardata

REML criterion at convergence: -18.5

Random effects:
 Groups   Name        Variance Std.Dev.
 block    (Intercept) 0.002251 0.04744
 Residual             0.010849 0.10416
Number of obs: 28, groups:  block, 7
```

```
Fixed effects:
            Estimate Std. Error t value
(Intercept)  5.75718    0.02663  216.21
trt1         0.13715    0.04964    2.76
trt2         0.56873    0.04964   11.46
trt3        -0.23124    0.04964   -4.66
trt4        -0.32720    0.04964   -6.59
trt5        -0.54404    0.04964  -10.96
trt6         0.55415    0.04964   11.16

Correlation of Fixed Effects:
     (Intr) trt1   trt2   trt3   trt4   trt5
trt1  0.000
trt2  0.000 -0.167
trt3  0.000 -0.167 -0.167
trt4  0.000 -0.167 -0.167 -0.167
trt5  0.000 -0.167 -0.167 -0.167 -0.167
trt6  0.000 -0.167 -0.167 -0.167 -0.167 -0.167
```

```
> Anova(fit3,test="F")
```

The KR anova yields a slightly higher F and slightly higher error df, the result of more information.

```
Analysis of Deviance Table (Type II Wald F tests with Kenward-Roger df)

Response: log(wear)
        F Df Df.res    Pr(>F)
trt 61.09  6 17.337 1.036e-10 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> fixef(fit3)
```

We can extract the estimates of the fixed effects like this.

```
(Intercept)        trt1        trt2        trt3        trt4        trt5        trt6
  5.7571762   0.1371457   0.5687293  -0.2312409  -0.3272006  -0.5440361   0.5541537
```

```
> linear.contrast(fit3,trt,c(0,1,-1,0,0,0,0))
```

This should work, but it is failing with an error. Looks like some bug chasing and an update are due.

```
> #
```

OK, we'll do it the hard way for now.

```
> cfs <- c(0,1,-1,0,0,0,0)
```

These coefficients compare treatments 1 and 2.

```
> sum(fixef(fit3)*cfs)
```

This is the interblock recovery estimate of the contrast. It differs a bit from the intrablock estimate.

```
[1] -0.4315836
```

```
> V <- vcov(fit3)
```

V is the variance/covariance matrix of the fixed effects from the model.

```
> t(cfs)%*%V%*%cfs
```

This matrix product computes the variance of the contrast.

```
1 x 1 Matrix of class "dgeMatrix"
            [,1]
[1,] 0.005750449
```

```
> sqrt(.005750449)
```

Take the square root for the standard error of the contrast. Note that it is just a touch less than the SE of the intrablock contrast.

```
[1] 0.07583172
```

```
> fit3.mcmc <- lmer.mcmc(fit3,50000)
```

Let's try the Bayesian approach.

```
> lmer.mcmc.intervals(fit3.mcmc)
```

Curiously, there is little evidence of block to block variability.

```
                    lower      median       upper          SE
(Intercept)  5.706598915  5.75651753  5.80787760 0.025473316
trt1         0.011344197  0.13518825  0.25573595 0.060744130
trt2         0.469056890  0.58035595  0.69466516 0.056392167
trt3        -0.345104486 -0.23577003 -0.12729795 0.055242499
trt4        -0.431603139 -0.32055717 -0.20289277 0.057299330
trt5        -0.658420145 -0.54375852 -0.43737995 0.055216869
trt6         0.441788809  0.54709335  0.65125665 0.053101939
block        0.000000000  0.00000000  0.01333745 0.004718074
sigma2       0.006831588  0.01283406  0.02631961 0.004965891
```

```
> m <- fit3.mcmc$mcmcout
```

Get the matrix of Markov chain outputs.

```
> d <- m[,2]-m[,3]
```

Treatment 1 and 2 are coefficients 2 and 3, so take their difference in the MCMC samples.

```
> mean(d)
```

Just a touch different from the non-Bayesian estimate.

```
[1] -0.4455276
```

```
> sd(d)
```

As is typical, the Bayesian SE is slightly larger.

```
[1] 0.09033554
```