



A Handbook of Statistical Analyses Using R

Brian S. Everitt and Torsten Hothorn



Analysing Longitudinal Data I: Computerised Delivery of Cognitive Behavioural Therapy—Beat the Blues

10.1 Introduction

10.2 Analysing Longitudinal Data

10.3 Analysis Using R

We shall fit both random intercept and random intercept and slope models to the data including the baseline BDI values (`pre.bdi`), `treatment` group, `drug` and `length` as fixed effect covariates. Linear mixed effects models are fitted in R by using the `lmer` function contained in the *lme4* package (Bates and Sarkar, 2006, Pinheiro and Bates, 2000, Bates, 2005), but an essential first step is to rearrange the data from the ‘wide form’ in which they appear in the `BtheB` data frame into the ‘long form’ in which each separate repeated measurement and associated covariate values appear as a separate row in a *data.frame*. This rearrangement can be made using the following code:

```
R> data("BtheB", package = "HSAUR")
R> BtheB$subject <- factor(rownames(BtheB))
R> nobs <- nrow(BtheB)
R> BtheB_long <- reshape(BtheB, idvar = "subject",
+   varying = c("bdi.2m", "bdi.4m", "bdi.6m", "bdi.8m"),
+   direction = "long")
R> BtheB_long$time <- rep(c(2, 4, 6, 8), rep(nobs, 4))
```

such that the data are now in the form (here shown for the first three subjects)

```
R> subset(BtheB_long, subject %in% c("1", "2", "3"))
```

	<i>drug</i>	<i>length</i>	<i>treatment</i>	<i>bdi.pre</i>	<i>subject</i>	<i>time</i>	<i>bdi</i>
1.2m	No	>6m	TAU	29	1	2	2
2.2m	Yes	>6m	BtheB	32	2	2	16
3.2m	Yes	<6m	TAU	25	3	2	20
1.4m	No	>6m	TAU	29	1	4	2
2.4m	Yes	>6m	BtheB	32	2	4	24
3.4m	Yes	<6m	TAU	25	3	4	NA
1.6m	No	>6m	TAU	29	1	6	NA
2.6m	Yes	>6m	BtheB	32	2	6	17
3.6m	Yes	<6m	TAU	25	3	6	NA
1.8m	No	>6m	TAU	29	1	8	NA

```

R> data("BtheB", package = "HSAUR")
R> layout(matrix(1:2, nrow = 1))
R> ylim <- range(BtheB[,grep("bdi", names(BtheB))],
+               na.rm = TRUE)
R> tau <- subset(BtheB, treatment == "TAU")[,
+           grep("bdi", names(BtheB))]
R> boxplot(tau, main = "Treated as usual", ylab = "BDI",
+          xlab = "Time (in months)", names = c(0, 2, 4, 6, 8),
+          ylim = ylim)
R> btheb <- subset(BtheB, treatment == "BtheB")[,
+           grep("bdi", names(BtheB))]
R> boxplot(btheb, main = "Beat the Blues", ylab = "BDI",
+          xlab = "Time (in months)", names = c(0, 2, 4, 6, 8),
+          ylim = ylim)

```

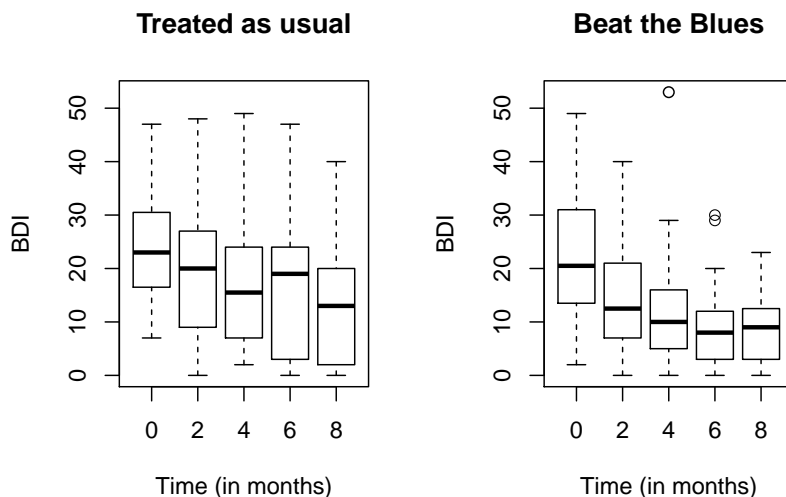


Figure 10.1 Boxplots for the repeated measures by treatment group for the `BtheB` data.

2.8m	Yes	>6m	<i>BtheB</i>	32	2	8	20
3.8m	Yes	<6m	<i>TAU</i>	25	3	8	NA

The resulting *data.frame* `BtheB_long` contains a number of missing values and in applying the `lmer` function these will be dropped. But notice it is only the missing values that are removed, *not* participants that have at least one missing value. All the available data is used in the model fitting process. The `lmer` function is used in a similar way to the `lm` function met in Chapter ?? with the addition of a random term to identify the source of the repeated

measurements, here `subject`. We can fit the two models (??) and (??) and test which is most appropriate using

```
R> library("lme4")
R> BtheB_lmer1 <- lmer(bdi ~ bdi.pre + time + treatment + drug +
+   length + (1 | subject), data = BtheB_long,
+   method = "ML", na.action = na.omit)
R> BtheB_lmer2 <- lmer(bdi ~ bdi.pre + time + treatment + drug +
+   length + (time | subject), data = BtheB_long,
+   method = "ML", na.action = na.omit)
R> anova(BtheB_lmer1, BtheB_lmer2)
```

Data: BtheB_long

Models:

```
BtheB_lmer1: bdi ~ bdi.pre + time + treatment + drug + length + (1 | subject)
BtheB_lmer2: bdi ~ bdi.pre + time + treatment + drug + length + (time | subject)
```

	Df	AIC	BIC	logLik	Chisq	Chi	Df	Pr(>Chisq)
<i>BtheB_lmer1</i>	8	1886.6	1915.7	-935.31				
<i>BtheB_lmer2</i>	10	1889.8	1926.2	-934.90	0.8161		2	0.665

```
R> summary(BtheB_lmer1)
```

```
Linear mixed model fit by maximum likelihood
```

```
Formula: bdi ~ bdi.pre + time + treatment + drug + length + (1 | subject)
```

```
Data: BtheB_long
```

```
AIC BIC logLik deviance REMLdev
```

```
1887 1916 -935.3 1871 1866
```

```
Random effects:
```

```
Groups Name Variance Std.Dev.
```

```
subject (Intercept) 48.304 6.9501
```

```
Residual 25.128 5.0127
```

```
Number of obs: 280, groups: subject, 97
```

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	5.94371	2.24911	2.643
bdi.pre	0.63819	0.07759	8.225
time	-0.71703	0.14605	-4.909
treatmentBtheB	-2.37311	1.66365	-1.426
drugYes	-2.79786	1.71990	-1.627
length>6m	0.25639	1.63210	0.157

```
Correlation of Fixed Effects:
```

	(Intr)	bdi.pr	time	trtmBB	drugYs
bdi.pre	-0.678				
time	-0.264	0.023			
tretmntBthB	-0.389	0.121	0.022		
drugYes	-0.071	-0.237	-0.025	-0.323	
length>6m	-0.238	-0.242	-0.043	0.002	0.158

Figure 10.2 R output of the linear mixed-effects model fit for the BtheB data.

Bibliography

- Bates, D. (2005), “Fitting linear mixed models in R,” *R News*, 5, 27–30, URL <http://CRAN.R-project.org/doc/Rnews/>.
- Bates, D. and Sarkar, D. (2006), *lme4: Linear Mixed-Effects Models Using Eigen and S4*, URL <http://CRAN.R-project.org>, R package version 0.99875-8.
- Pinheiro, J. C. and Bates, D. M. (2000), *Mixed-Effects Models in S and S-PLUS*, New York, USA: Springer.