

Aufgabe 1:

In dieser Aufgabe sollen Sie sich mit dem Ratten-Datensatz vertraut machen. Lesen Sie sich dazu zunächst die auf der der Veranstaltungshomepage verlinkte Beschreibung des Datensatzes durch.

- (a) Lesen Sie den Datensatz in R ein. Wandeln Sie `GROUP` in eine `factor`-Variable mit entsprechenden Labels für die 3 Experimentalgruppen um.

Lösung:

```
R>rats <- read.table("http://www.statistik.lmu.de/institut/lehrstuhl/semwiso/longi  
+   header = T, na.strings = ".")  
R>rats$GROUP <- factor(rats$GROUP, labels = c("low", "high",  
+   "control"))  
R>rats$SUBJECT <- factor(rats$SUBJECT)
```

- (b) Benutzen Sie `reshape` um den Datensatz so umzuformatieren, dass pro Tier nur noch eine Zeile im Datensatz steht. Benutzen Sie den umformatierten Datensatz `rats.wide` um sich einen Überblick über die Ausfallraten in den verschiedenen Experimentalgruppen zu verschaffen. Wie viele vollständig beobachtete Tiere gibt es in den einzelnen Experimentalgruppen, wie viele Tiere mit nur 5, 4, 3 etc. Beobachtungen?

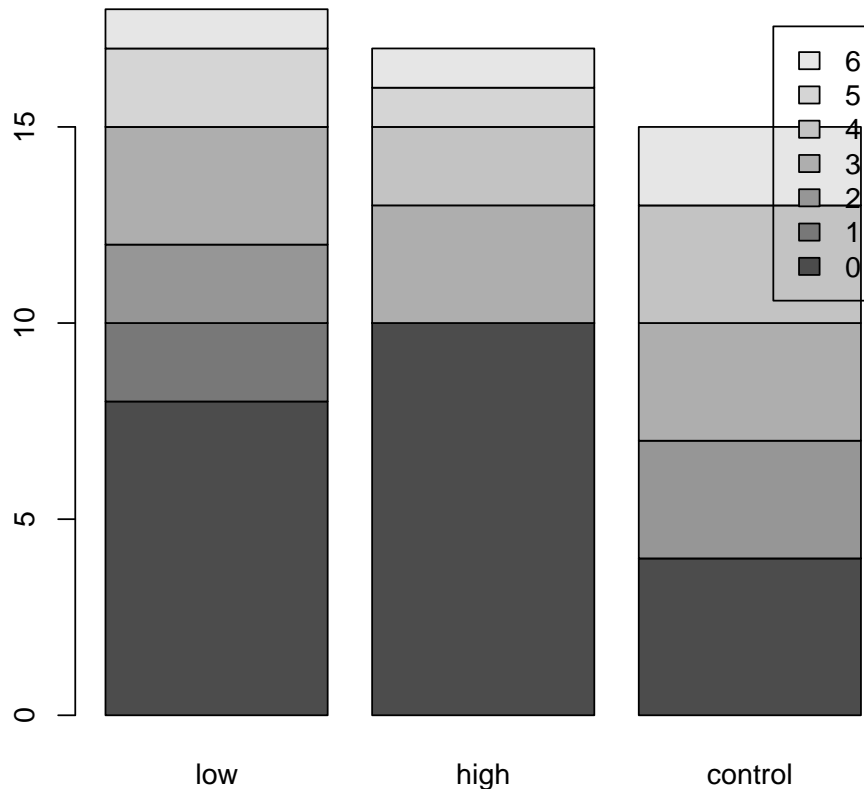
Lösung:

```
R>rats.wide <- reshape(data = rats, v.names = "RESPONSE",  
+   idvar = "SUBJECT", timevar = "TIME", direction = "wide")  
R>rats.wide$NAs <- rowSums(is.na(rats.wide))  
R>(na.structure <- xtabs(~NAs + GROUP, rats.wide))
```

	GROUP		
NAs	low	high	control
0	8	10	4
1	2	0	0
2	2	0	3
3	3	3	3
4	0	2	3
5	2	1	0
6	1	1	2

```
R>barplot(na.structure, legend.text = rownames(na.structure),  
+   main = "Anzahl fehlender Beobachtungen")
```

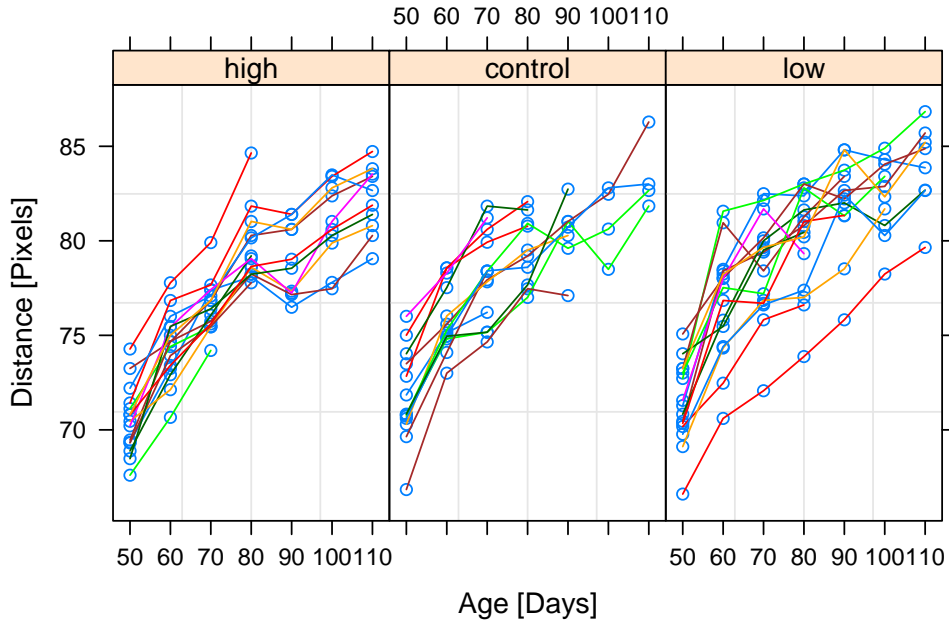
Anzahl fehlender Beobachtungen



- (c) Laden Sie das `nlme`-Paket. Legen Sie, ausgehend von den Original-Daten, einen neuen Datensatz `rats2` im `groupedData`-Format an, der nur die Tiere mit mindestens 3 Beobachtungen enthält. Plotten Sie die Verläufe der einzelnen Tiere getrennt nach Experimentalgruppen.

Lösung:

```
R>library(nlme)
R>keep <- with(rats.wide, SUBJECT[NAs <= 4])
R>rats2 <- groupedData(RESPONSE ~ TIME | GROUP/SUBJECT,
+   data = rats[rats$SUBJECT %in% keep, ], labels = list(x = "Age",
+   y = "Distance"), units = list(x = "[Days]", y = "[Pixels]"))
R>print(plot(rats2, display = "GROUP", inner = ~SUBJECT))
```



(d) Fitten Sie ein lineares Modell auf dem Datensatz der kompletten Beobachtungen und plotten Sie die Verläufe der Residuen für die einzelnen Tiere.

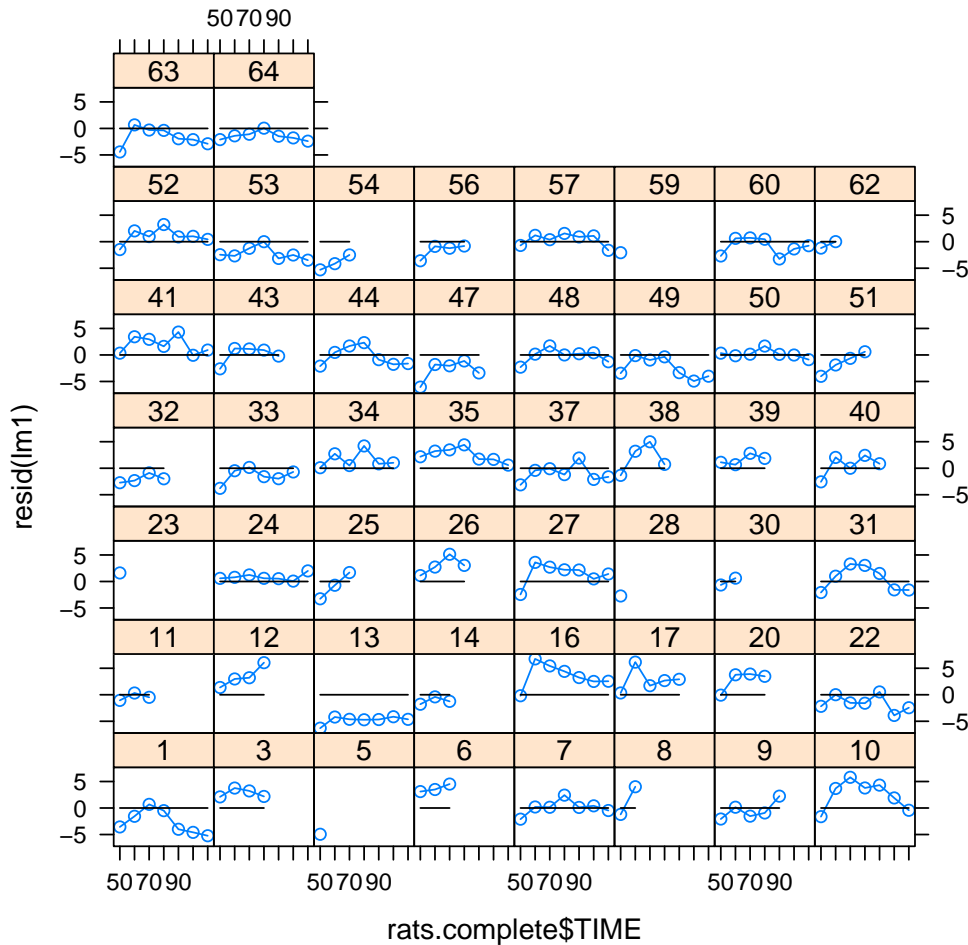
```
R>rats$SUBJECT <- factor(rats$SUBJECT)
R>rats.complete <- rats[complete.cases(rats), ]
R>lm1 <- lm(RESPONSE ~ TIME, data = rats.complete)
R>xyplot(resid(lm1) ~ rats.complete$TIME | rats.complete$SUBJECT,
+       panel = function(x, y) {
+         panel.points(x, y)
+         panel.lines(x, y)
+         panel.lines(x = x, y = 0, col = "black")
+       })
```

Was fällt ihnen auf? Sind die Annahmen des gewöhnlichen linearen Modells erfüllt?

Lösung:

```
R>print(xyplot(resid(lm1) ~ rats.complete$TIME | rats.complete$SUBJECT,
+       panel = function(x, y) {
+         panel.points(x, y)
+         panel.lines(x, y)
```

```
+   panel.lines(x = x, y = 0, col = "black")
+   })
```



Oft starke positive Autokorrelation in den Residuen-Verläufen der einzelnen Tiere; d.h. Fehlerterme nicht unabhängig, sondern (blockweise) korreliert.

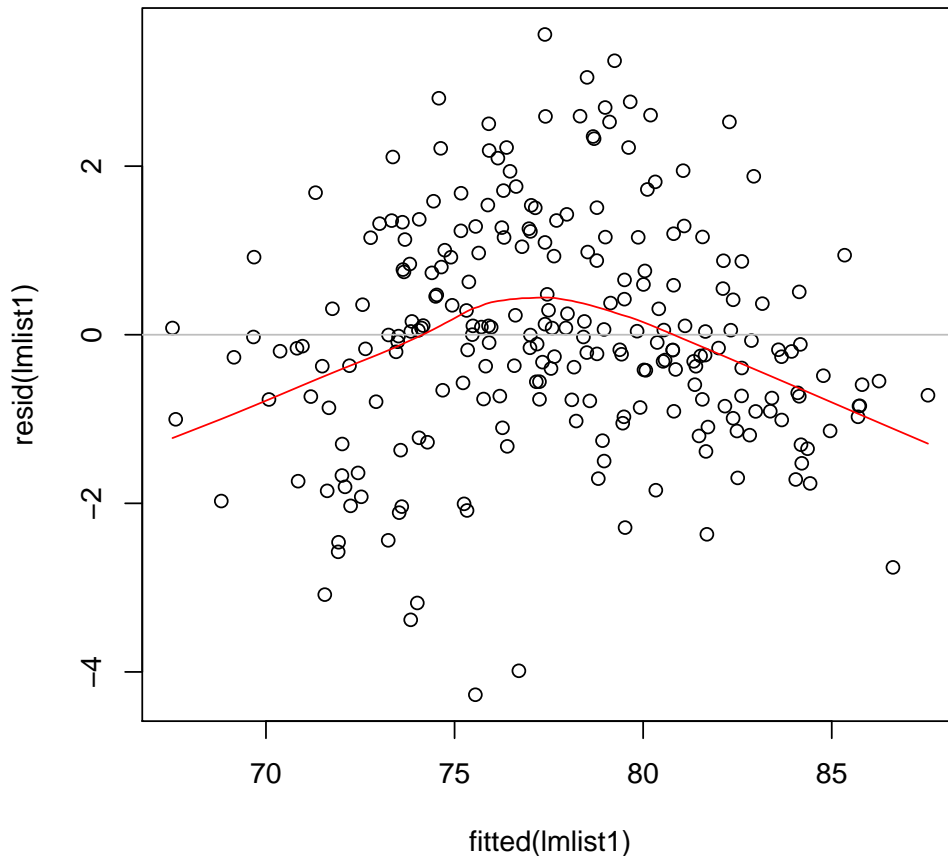
- (e) Benutzen Sie `lmList` um für jedes Tier ein separates lineares Modell mit zeitlichem Trend $I(\text{TIME}-45)$ anzupassen und betrachten sie den Residuenplot. Warum macht es Sinn die Zeit-Variable so zu transformieren? Was fällt am Residuenplot auf?

```
R>lm1list1 <- lmList(RESPONSE ~ I(TIME - 45), rats2, na.action = na.omit)
R>plot(fitted(lm1list1), resid(lm1list1))
R>abline(h = 0, col = "grey")
R>o <- order(fitted(lm1list1))
R>lines(lowess(fitted(lm1list1)[o], resid(lm1list1)[o]),
+       col = 2)
R>complete <- complete.cases(rats2)
R>xyplot(resid(lm1list1) ~ rats2$TIME[complete] | rats2$SUBJECT[complete],
+       panel = function(x, y) {
+         panel.points(x, y)
+         panel.lines(x, y)
+         panel.lines(x = x, y = 0, col = "black")
+       })
```

Lösung:

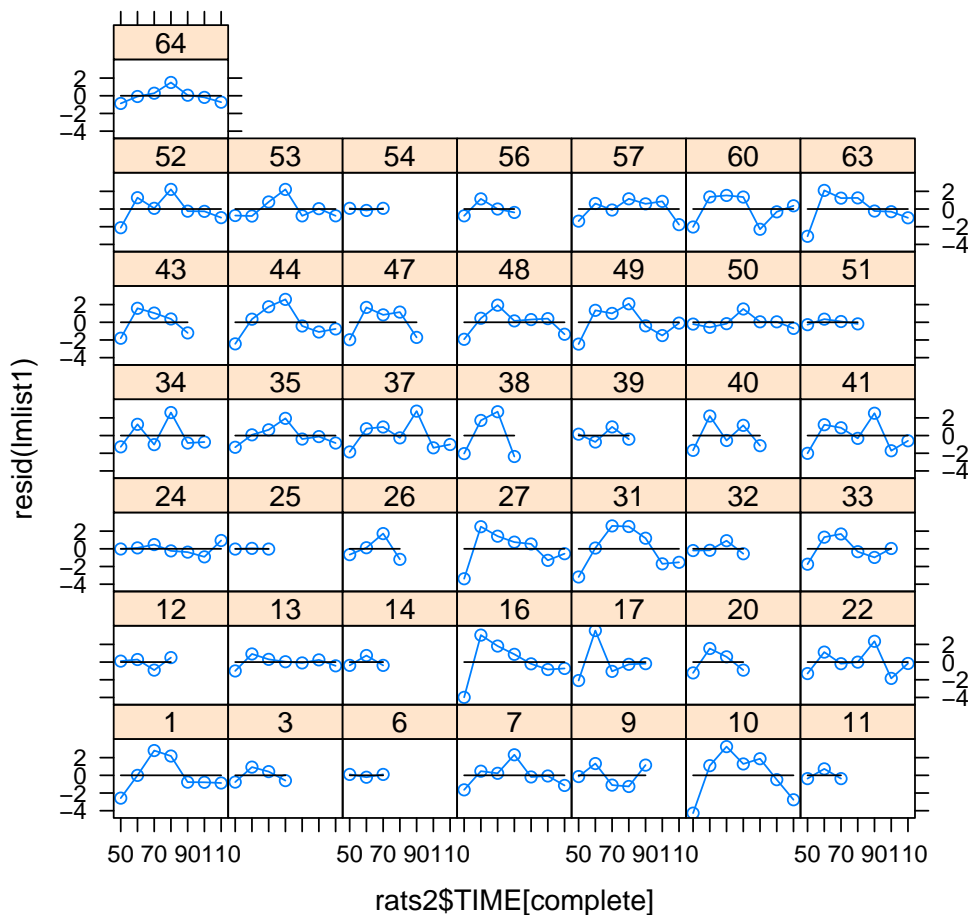
Durch die Transformation ist der Intercept interpretierbar als Ausgangsniveau zu Beginn der Behandlung.

```
R>plot(fitted(lm1), resid(lm1))
R>abline(h = 0, col = "grey")
R>o <- order(fitted(lm1))
R>lines(lowess(fitted(lm1)[o], resid(lm1)[o]),
+       col = 2)
```



Zeitlicher Trend in den Residuen.

```
R>complete <- complete.cases(rats2)
R>print(xyplot(resid(lm1) ~ rats2$TIME[complete] |
+   rats2$SUBJECT[complete], panel = function(x, y) {
+   panel.points(x, y)
+   panel.lines(x, y)
+   panel.lines(x = x, y = 0, col = "black")
+ }))
```



Autokorrelation deutlich schwächer ausgeprägt.

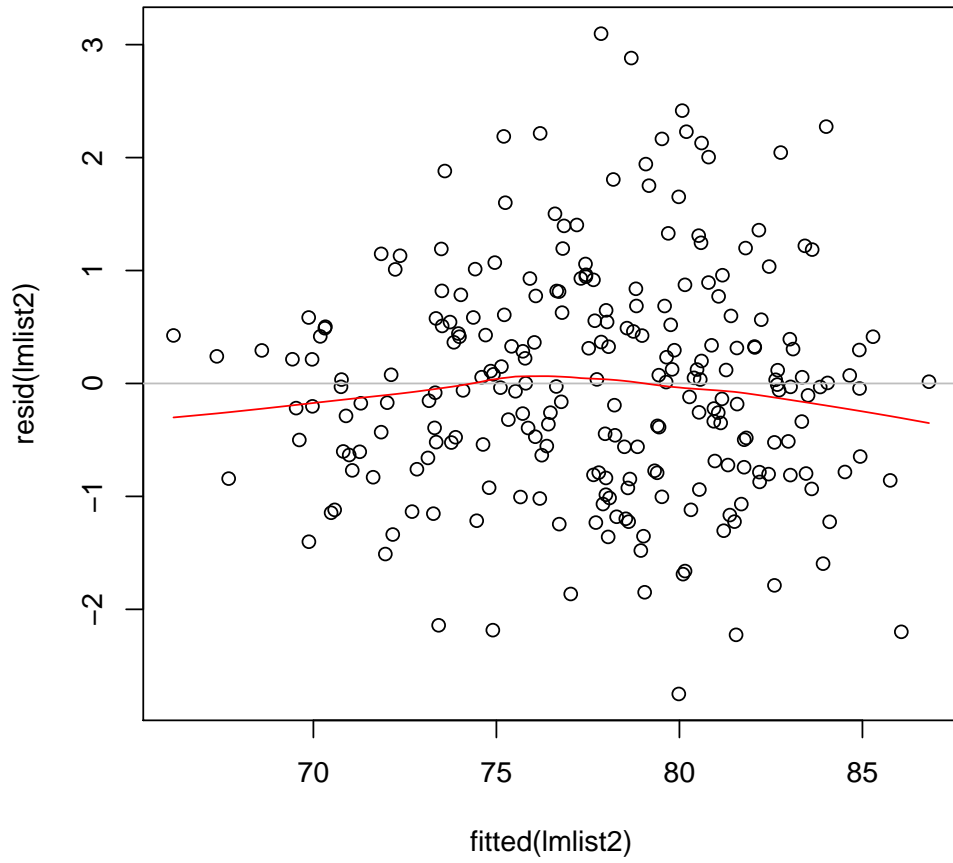
- (f) Die Verläufe sind wohl nicht linear. Legen Sie eine Variable $\log T = \log(1 + (\text{TIME} - 45)/10)$ an. Benutzen Sie wieder `lmList` um für jedes Tier ein separates lineares Modell mit zeitlichem Trend anzupassen, überprüfen Sie den Residuenplot (s.o.) und visualisieren Sie die Parameterschätzungen mit den folgenden Befehlen:

```
R>plot(intervals(lmList - Objekt))
```

Interpretieren Sie die Plots.

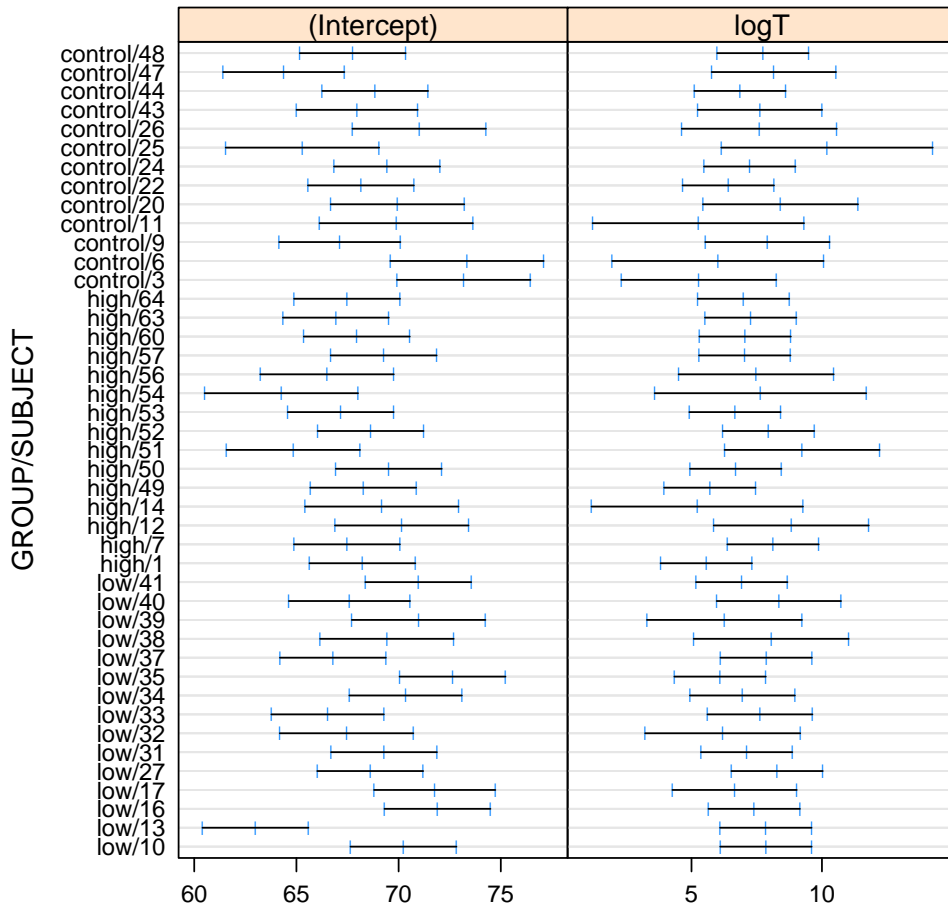
Lösung:

```
R>rats2$logT = log(1 + (rats2$TIME - 45)/10)
R>lm1st2 <- lmList(RESPONSE ~ logT, rats2, na.action = na.omit)
R>plot(fitted(lm1st2), resid(lm1st2))
R>abline(h = 0, col = "grey")
R>o <- order(fitted(lm1st2))
R>lines(lowess(fitted(lm1st2)[o], resid(lm1st2)[o]),
+       col = 2)
```



Kein deutlicher Trend in den Residuen mehr erkennbar. Evtl. Heteroskedastisch?

```
R>print(plot(intervals(lmlist2)))
```



Unterschiedliche Streuung weil unbalancierte Daten. Wenig Unterschiede in Steigung, größere Unterschiede in Intercept.

- (g) Untersuchen sie die subjektspezifischen R_{adj}^2 und vergleichen Sie diese mit einem Modell mit quadratischem Trend in $\log T$.

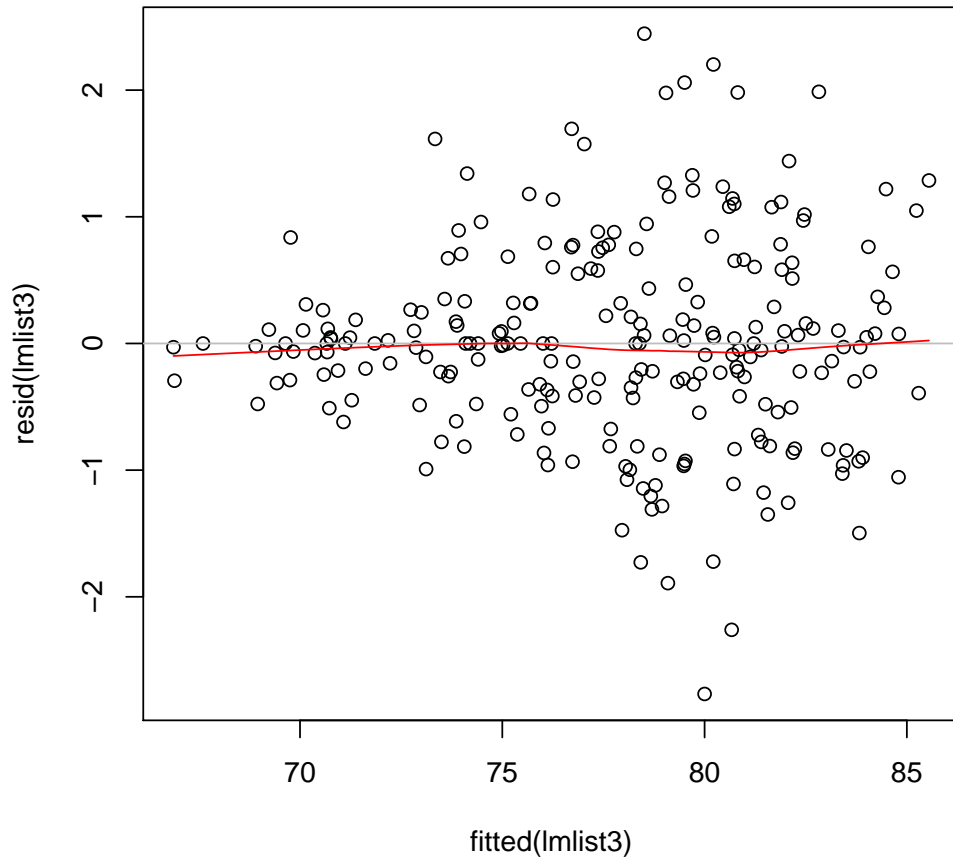
Hinweis:

Betrachten sie zunächst `str(summary(lm1list1[[1]]))` um herauszufinden wo R_{adj}^2 abgelegt ist. Dann könnte `lapply()` weiterhelfen.

Was ist bei den gegebenen Daten problematisch an den quadratischen Fits?

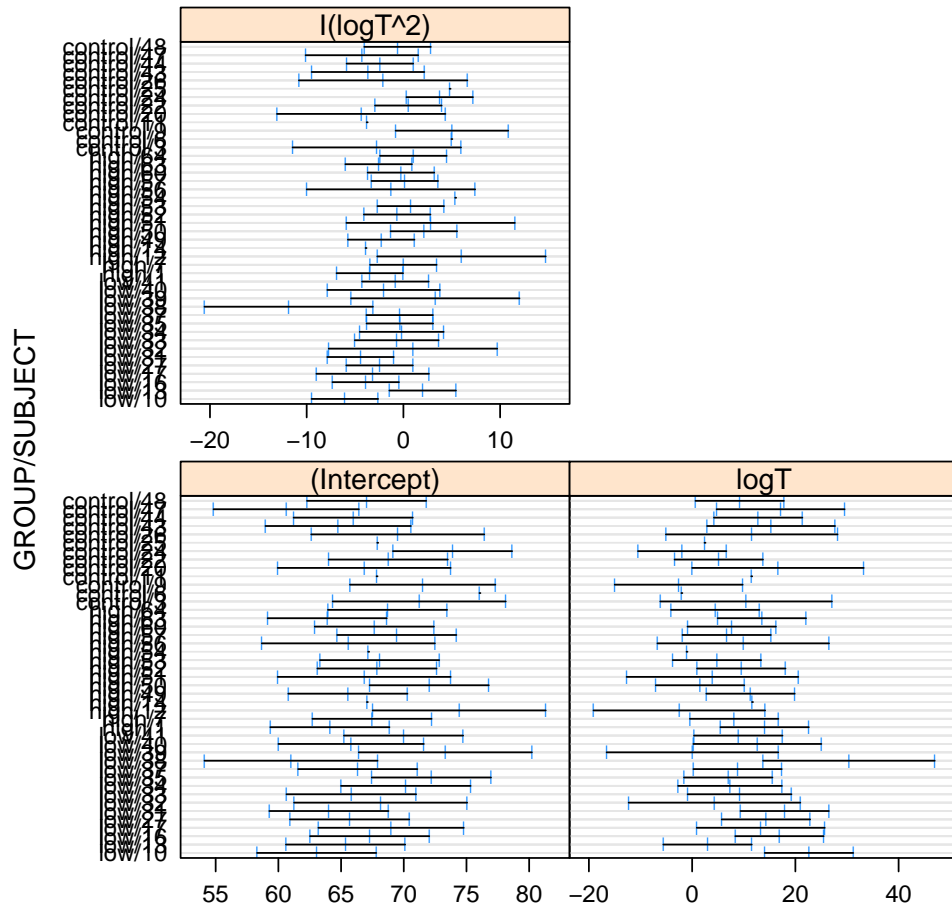
Lösung:

```
R>get.R2adj <- function(x) {
+   summary(x)$adj.r.squared
+ }
R>R2adj.lin <- unlist(lapply(lm1list2, get.R2adj))
R>lm1list3 <- lmList(RESPONSE ~ logT + I(logT^2), rats2,
+   na.action = na.omit)
R>plot(fitted(lm1list3), resid(lm1list3))
R>abline(h = 0, col = "grey")
R>o <- order(fitted(lm1list3))
R>lines(lowess(fitted(lm1list3)[o], resid(lm1list3)[o]),
+   col = 2)
```

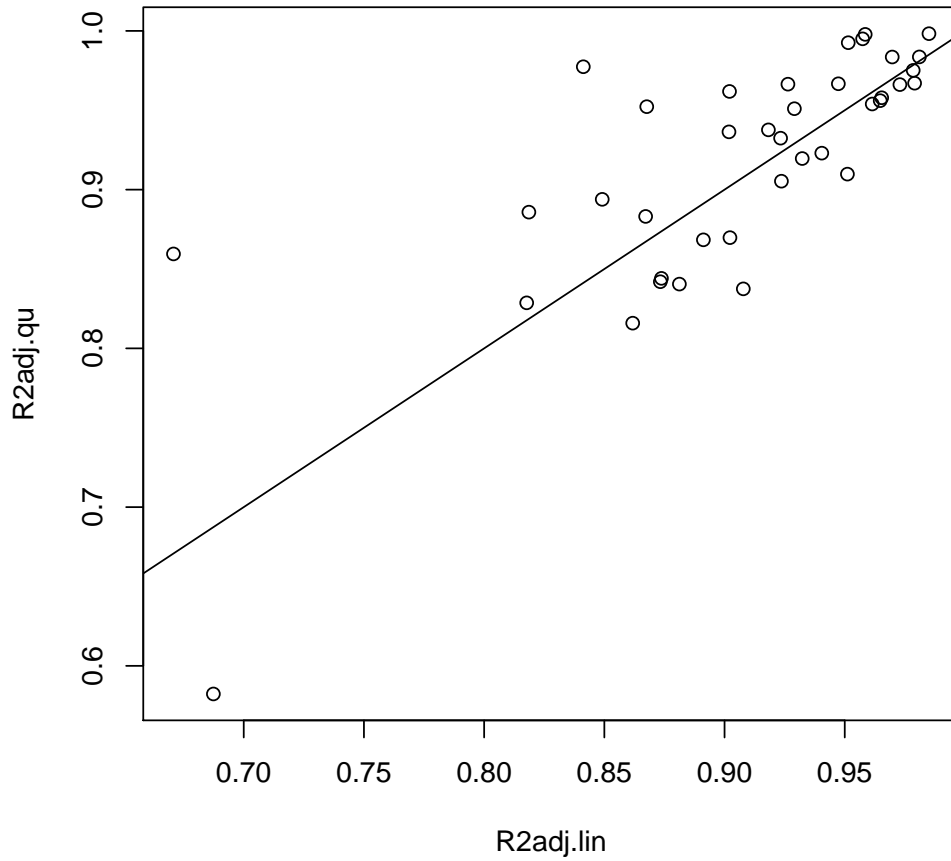


Kein Trend mehr erkennbar in den Residuen.

```
R>print(plot(intervals(lmlist3)))
```



```
R>R2adj.qu <- unlist(lapply(lm1ist3,get.R2adj))
R>plot(R2adj.lin,R2adj.qu)
R>abline(c(0, 1))
```



Manche Ratten haben nur 3 Beobachtungen \Rightarrow Modell mit 3 Parametern ist überparametrisiert.

R_{adj}^2 für diese Subjekte ist NaN (Not A Number, d.h. nicht definiert) da die Residuen 0 Freiheitsgrade haben. Teilweise schlechteres R_{adj}^2 für quadratischen Fit, linearer Trend scheint ausreichend zu sein.