

Chapter 4: Worked Example

Jules Lanari-Collard

2024-08-09

Adjusting for Baseline Measurements

Now we consider the case where we have baseline measurements. A sample of the data is shown in table 1; now there is an extra column with the baseline measurement before each period (Pre).

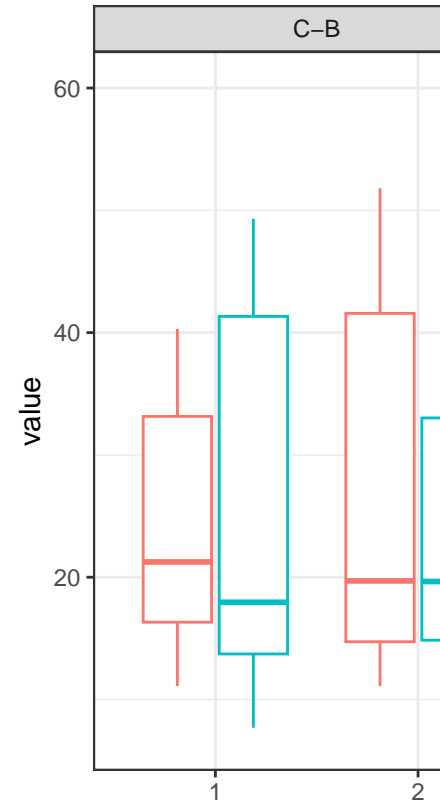
Our summary table can be expanded to include the baseline measurements, as shown in table 2.

Table 1: Data with Baseline Measurements

Subject	Sequence	Period	treat	Pre	Post
1	C-B	1	CRICKET	20.3	18.3
1	C-B	2	BEEF	14.6	14.1
2	C-B	1	CRICKET	11.1	14.7
2	C-B	2	BEEF	11.1	24.7
4	C-B	1	CRICKET	38.3	45.9

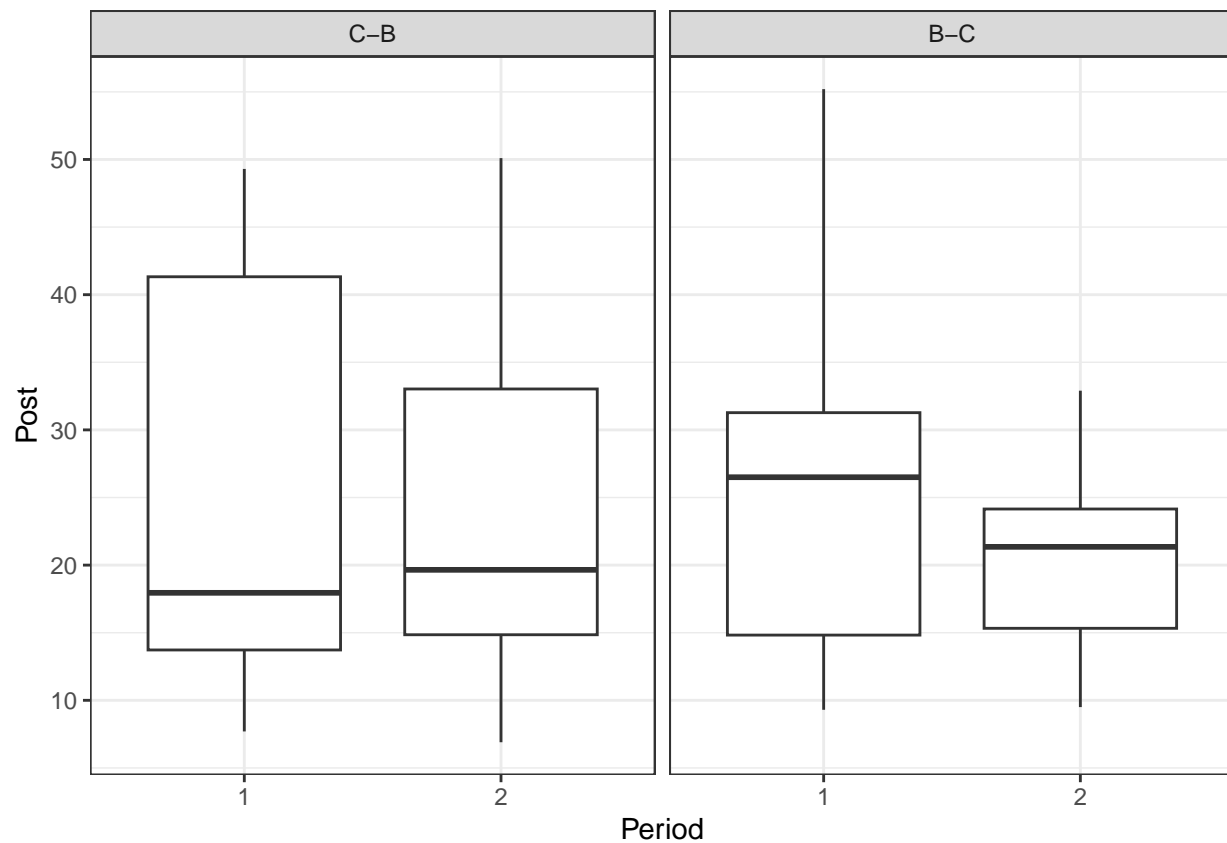
Table 2: Summary Table with Baseline Values (Pre)

Sequence	Subject	Overall				Period 1				Period 2			
		Pre		Post		Pre		Post		Pre		Post	
		Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
C-B	10	25.71	13.53	24.55	14.43	24.03	10.70	25.13	16.07	27.39	16.31	23.97	13.45
B-C	10	27.91	14.39	23.12	11.11	29.73	18.94	25.84	13.83	26.10	8.44	20.40	7.27
Total	20	26.81	13.83	23.84	12.74	26.88	15.25	25.48	14.60	26.75	12.66	22.18	10.68

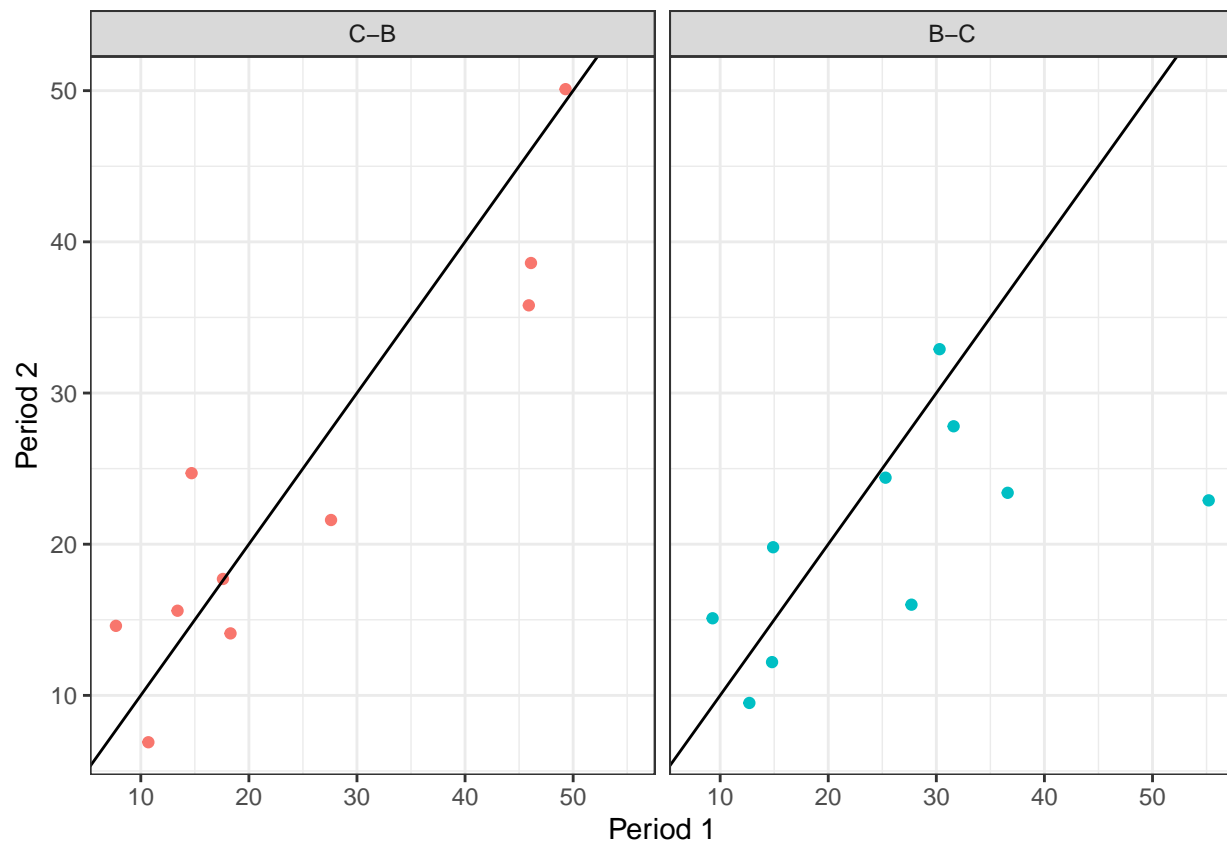


We can also extend the boxplot to incorporate the baseline measurements (see figure ??).

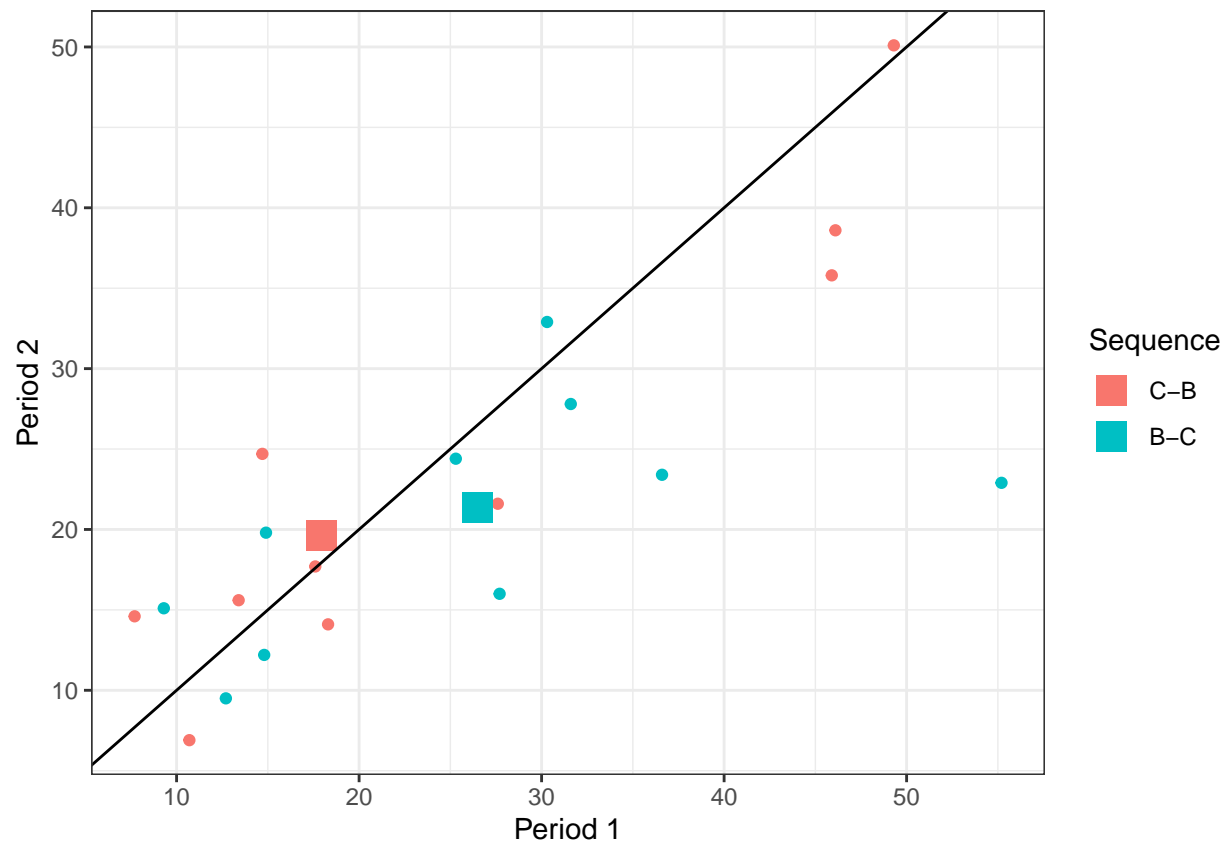
```
ggplot(data = data, mapping = aes(x = Period, y = Post)) +
  facet_wrap(~Sequence) +
  theme_bw() +
  geom_boxplot()
```



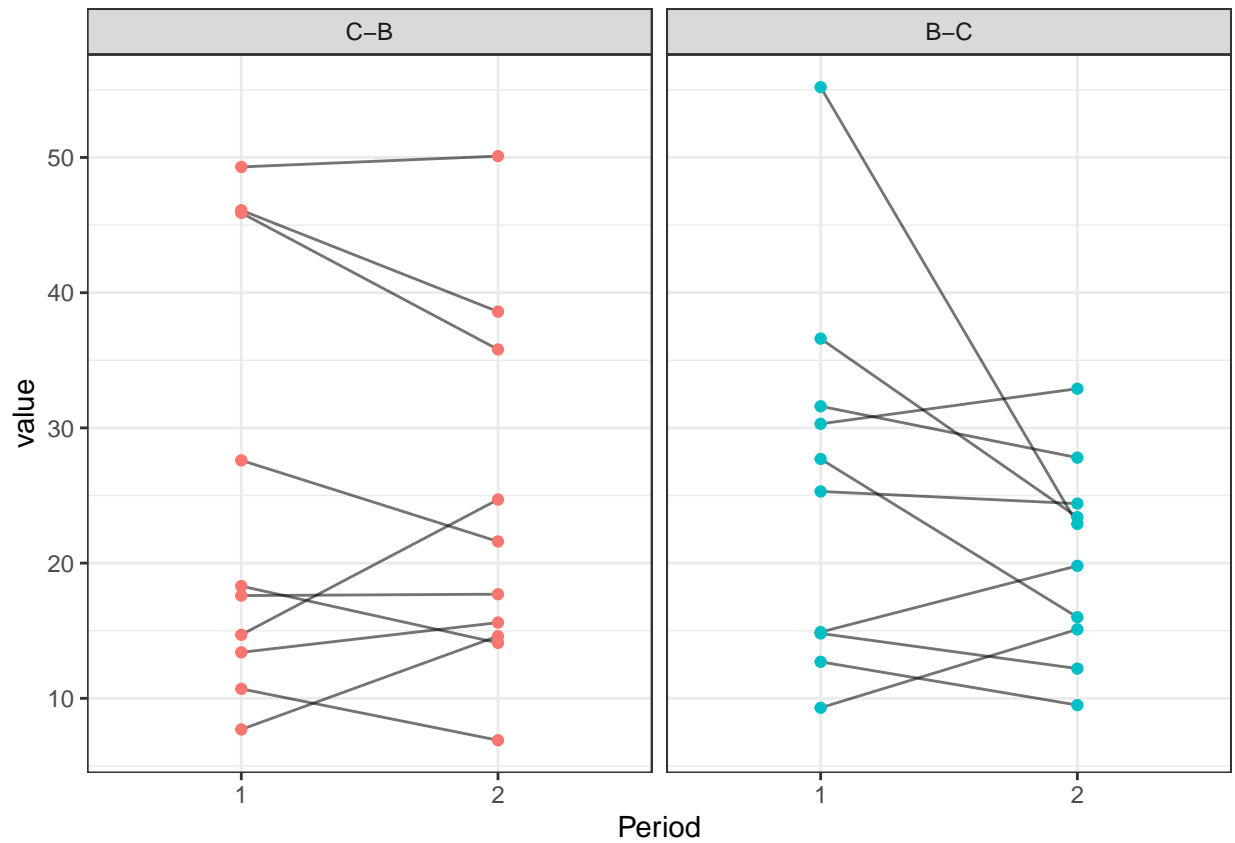
```
ggplot(data = data.wide, mapping = aes(x = Post_1, y = Post_2, col = Sequence)) +
  facet_wrap(~Sequence) +
  geom_point(show.legend = FALSE) +
  geom_abline(slope=1,intercept=0) +
  labs(x="Period 1", y="Period 2") +
  theme_bw()
```



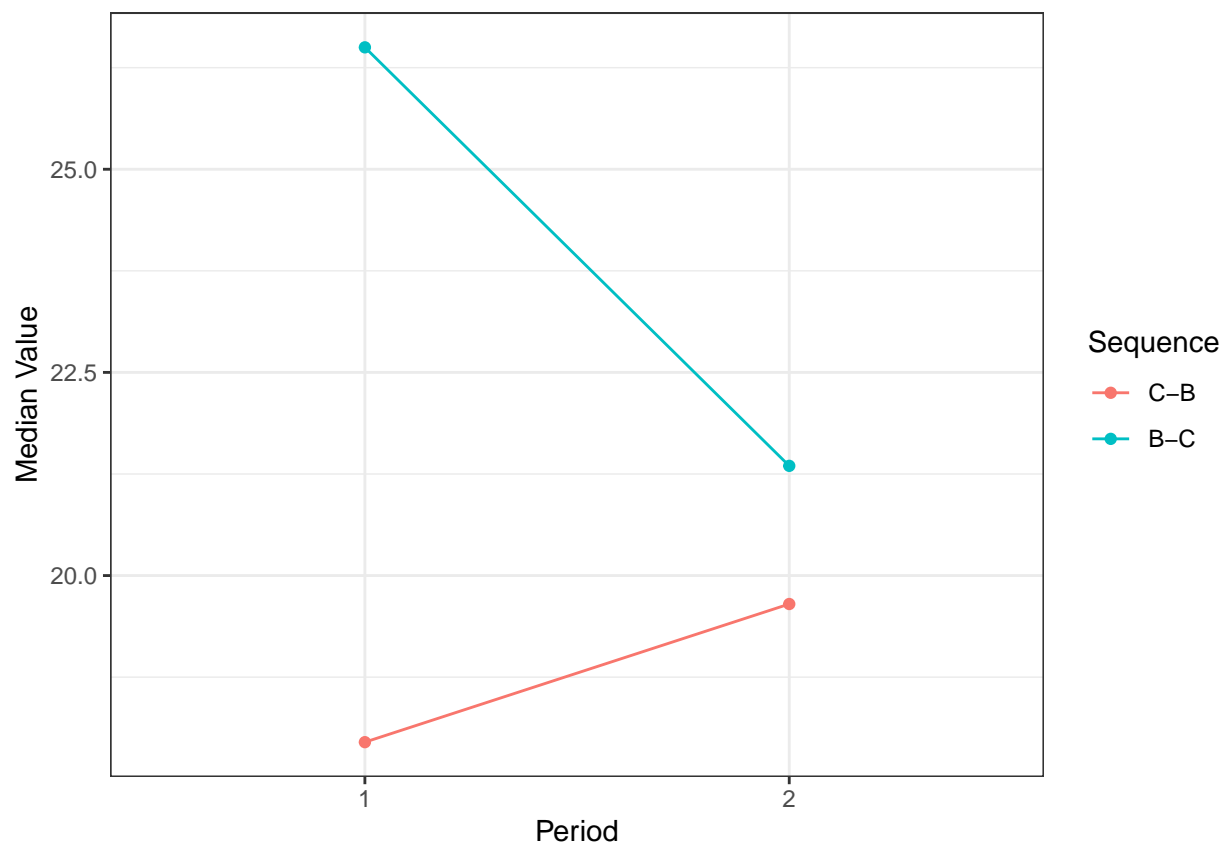
```
ggplot(data = data.wide, mapping = aes(x = Post_1, y = Post_2, col = Sequence)) +
  geom_point(show.legend = FALSE) +
  geom_point(data = medians,
             mapping = aes(x = Period1, y = Period2),
             shape = "square", size = 5) +
  geom_abline(slope=1,intercept=0) +
  labs(x="Period 1", y="Period 2") +
  theme_bw()
```



```
ggplot(data = data.long %>% filter(Measurement == "Post"),
       mapping = aes(x = Period, y = value)) +
  facet_wrap(~Sequence) +
  geom_line(aes(group = Subject), alpha = 0.55) +
  geom_point(aes(col = Sequence), show.legend = FALSE) +
  theme_bw()
```



```
ggplot(data = medians.long, mapping = aes(x = Period, y = Value, colour = Sequence)) +
  geom_line(aes(group = Sequence)) +
  geom_point() +
  labs(y = "Median Value") +
  theme_bw()
```



Model

```
aov(Post ~ treat + Period + baseline_diff + Sequence + Subject,
      data = data.baselines) %>%
  summary() %>%
  xtable() %>%
  xtable2kable()
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
treat	1	45.80	45.80	1.08	0.3116
Period	1	108.90	108.90	2.58	0.1258
baseline_diff	1	137.19	137.19	3.25	0.0883
Sequence	1	1.08	1.08	0.03	0.8746
Subject	17	5272.19	310.13	7.34	0.0001
Residuals	18	760.56	42.25		

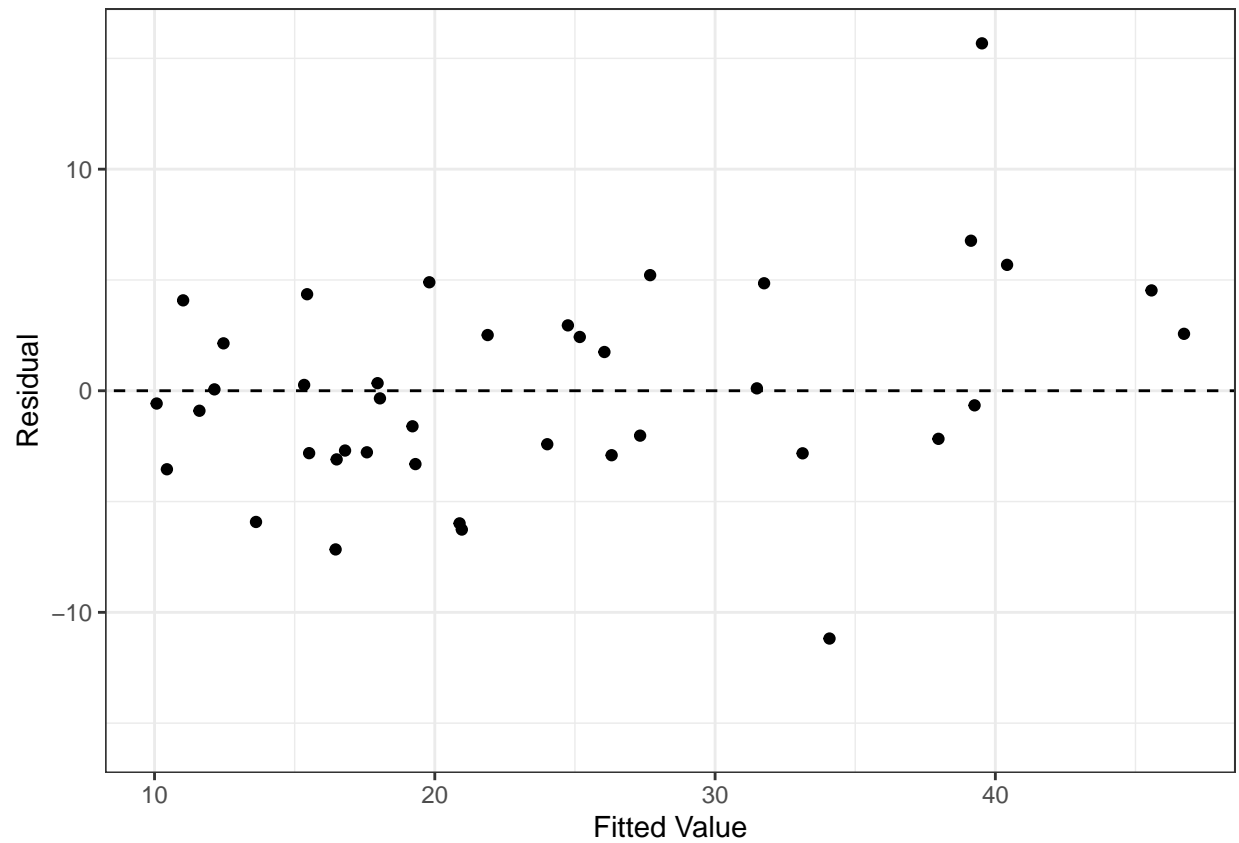
```
model <- lmer(Post ~ treat + Period + Sequence + (1|Subject),
               data = data.baselines)

summary(model)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
```

```
## lmerModLmerTest]
## Formula: Post ~ treat + Period + Sequence + (1 | Subject)
## Data: data.baselines
##
## REML criterion at convergence: 282.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.72026 -0.43351 -0.07075  0.40988  2.41187
##
## Random effects:
## Groups Name Variance Std.Dev.
## Subject (Intercept) 128.60 11.34
## Residual 42.25 6.50
## Number of obs: 40, groups: Subject, 20
##
## Fixed effects:
## Estimate Std. Error df t value Pr(>|t|)
## (Intercept) 25.130 4.133 22.981 6.080 3.36e-06 ***
## treatBEEF 2.140 2.056 18.000 1.041 0.312
## Period2 -3.300 2.056 18.000 -1.605 0.126
## SequenceB-C -1.430 5.472 18.000 -0.261 0.797
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
## (Intr) trBEEF Perid2
## treatBEEF -0.249
## Period2 -0.249 0.000
## SequenceB-C -0.662 0.000 0.000
```

```
model.metrics <- model %>% augment()
ggplot(data = model.metrics, mapping = aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_abline(slope = 0, linetype = "dashed") +
  scale_y_continuous(limits = symmetric_limits) +
  theme_bw() +
  labs(x = "Fitted Value", y = "Residual")
```

```
ggplot(data = model.metrics, mapping = aes(sample = .resid)) +  
  geom_qq() +  
  geom_qq_line() +  
  labs(x = "Normal Quantiles", y = "Residuals") +  
  theme_bw()
```

