Exercises for Day 5 – Solution
Applied Statistics & Statistical methods for SCIENCE

Anton Rask Lundborg

December 2023

Exercise 5.1 Hypertension in diabetic patients

Problem

In this exercise we will analyze the dataset from Exercise 1.5 using an ANCOVA with a random effect. This is an alternative to the four t-tests done in Exercise 1.5 (two of these tests did the comparison of the two drugs, and gave the p-values 0.0932 and 0.1108). But first let us recap the description of the dataset:

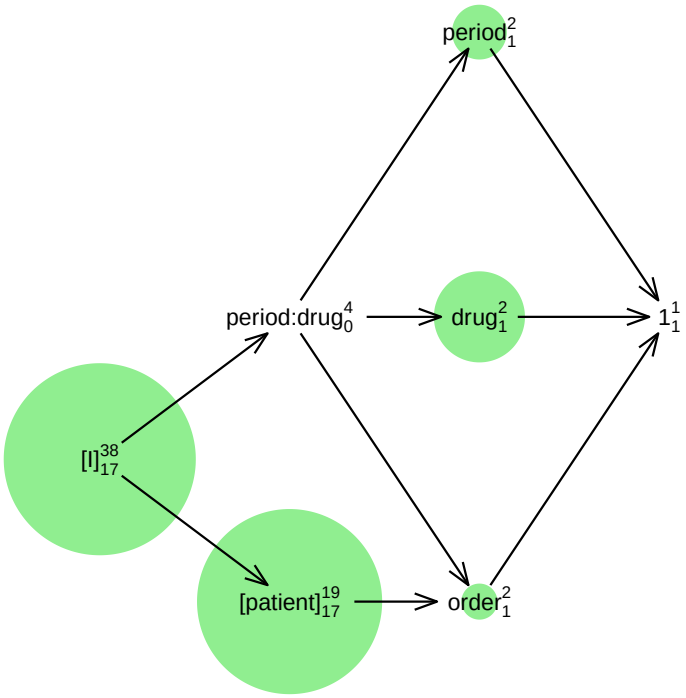
An experiment on 19 diabetic patients was conducted in order to compare the effects of two drugs called *Drug E* and *Drug N* on the treatment of high blood pressure. The study is a cross-over study which means that all patients try both drugs in two different study periods. Both study periods lasted for 14 days. In between the two study periods was a wash-out period, which also lasted for 14 days. The patients were randomly assigned to two groups called *E/N* and *N/E*. The patients in the E/N-group received drug E in the first study period and drug N in the second study period. The patients in the N/E-group received drug N in the first study period and drug E in the second study period.

The systolic and the diastolic blood pressure was measured for all the patients at the beginning and the end of both study periods. In this exercise we will analyze the *change of the systolic blood pressure*. These observations are available in the text file `hypertension.txt` in *long form*, i.e. with one response measurement in each row. This dataset contains the following variables:

Variable	Type	Range	Usage
`drug`	nominal	E, N	??
`period`	nominal	1, 2	??
`order`	nominal	E/N, N/E	??
`patient`	nominal	8,...,26	??
`baseline`	continuous	$[107; 156]$	??
`end`	continuous	$[91; 154]$	??
`change`	continuous	$[-25; 25]$	??

Please answer the following 5 questions:

1. In the table above replace the "??"s by either "fixed effect", "random effect", "response", or "not used".
2. In Exercise 1.5 two of the t-tests were done to see if there was an effect of `order` or an interaction between `period` and `drug`. For the cross-over study design to be successful neither of these should be significant. To investigate these potential problems in an ANOVA model we would incorporate the main effect of `order` and the interaction `period:drug` in the model. Hence a model with the following design diagram[1]:



   Inspecting the design diagram we see that the interaction `period:drug` has zero degrees of freedom. This means that this interaction does not contribute to the model at all. Can you explain why? In other words, why does it not make sense to include both the main effect of `order` and the interaction `period:drug` at the same time?

   Hint: What is the relation between `period:drug` and `order`?

3. Analyze the dataset. Possibly also using `baseline` as a fixed effect.

   Hint: If you want to include `baseline` in the model you simply add it to the right hand side of the "$\sim$" in the model equation. Furthermore, remember that `period` and `patient` should be used as categorical factors in the model.

4. Is this analysis more powerful than the analysis done in Exercise 1.5?
5. In your opinion is this analysis more easy to communicate compared to the analysis done in Exercise 1.5?

(Reference: Bradstreet, T.E. (1994) "Favorite Data Sets from Early Phases of Drug Research - Part 3." *Proceedings of the Section on Statistical Education of the American Statistical Association.*)

[1] In this design diagram the area of the green circles visualizes the proportion of total variance explained by the different terms in the model.

Patient id	Treatment order	Systolic blood pressure			
		Baseline 1	End 1	Baseline 2	End 2
9	Drug E, Drug N	124	136	120	145
21	Drug E, Drug N	120	132	138	126
8	Drug E, Drug N	115	96	111	91
12	Drug E, Drug N	134	118	123	123
16	Drug E, Drug N	131	106	111	123
19	Drug E, Drug N	119	108	113	112
20	Drug E, Drug N	124	112	108	112
24	Drug E, Drug N	127	113	121	143
13	Drug N, Drug E	113	113	107	97
17	Drug N, Drug E	132	109	122	119
18	Drug N, Drug E	129	133	139	130
23	Drug N, Drug E	124	120	127	118
25	Drug N, Drug E	112	103	112	121
10	Drug N, Drug E	124	112	128	122
11	Drug N, Drug E	144	154	156	137
14	Drug N, Drug E	134	118	122	109
15	Drug N, Drug E	119	118	115	114
22	Drug N, Drug E	123	123	114	108
26	Drug N, Drug E	122	123	124	120

Table 1: Treatment of hypertension in diabetic patients. Here the dataset is given in *wide form*, where all observations from each patient are inside a single row.

Solution

We first load the data and encode variables as factors:

```
hypertension <- read.delim("hypertension.txt")
hypertension$patient <- factor(hypertension$patient)
hypertension$period  <- factor(hypertension$period)
```

We add the uses of the different variables to the table:

Variable	**Type**	**Range**	**Usage**
`drug`	nominal	E, N	fixed
`period`	nominal	1, 2	fixed
`order`	nominal	E/N, N/E	fixed
`patient`	nominal	8,. . . ,26	random
`baseline`	continuous	$[107; 156]$	fixed
`end`	continuous	$[91; 154]$	not used
`change`	continuous	$[-25; 25]$	response

The interpretation of `period:drug` as a modification of `drug` is that the difference between drug E and drug N, i.e. E-N, may depend on the period. In period 1 we have the difference `diff1 = mean(1, E) - mean(1, N)` and in period 2 we have difference `diff2 = mean(2, E) - mean(2, N)`. Thus, the interaction `period:drug` models the difference of differences. That is,

$$\begin{aligned}\texttt{diff1 - diff2} &= \texttt{(mean(1, E) - mean(1, N)) - (mean(2, E) - mean(2, N))}\\ &= \texttt{(mean(1, E) + mean(2, N)) - (mean(1, N) + mean(2, E))}\end{aligned}$$

But this is exactly that same contrast that is modeled by `order`. In other words, The patients in the E/N group only experience the two drugs in a situation where you might have synergistic (or antagonistic) effect between `period`=1 and `drug`=E and between `period`=2 and `drug`=N. The patients in the N/E group only experience the two drugs in a situation where you might have synergistic (or antagonistic) effect between `period`=1 and `drug`=N, and between `period`=2 and `drug`=E.

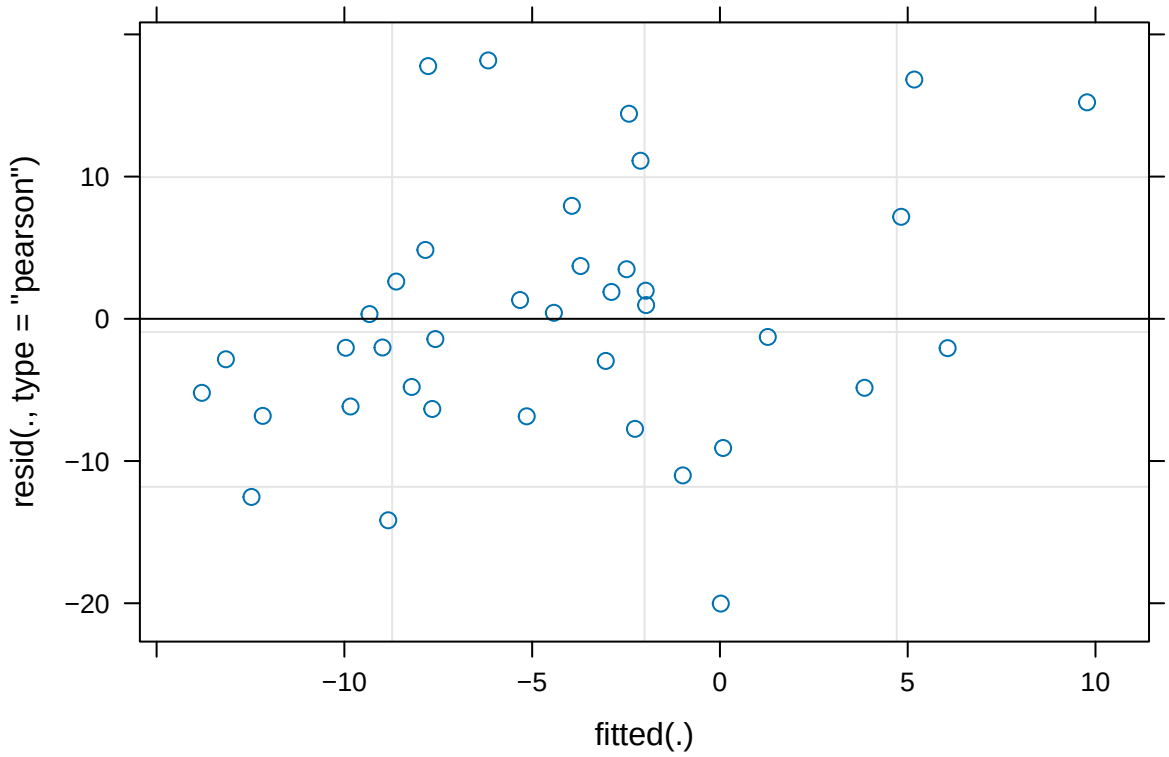
To analyze the data, we fit a linear mixed model with main effect of `baseline`, `order`, `period` and `drug` and a random effet of `patient`.

```
library(lme4)
m1 <- lmer(
  change ~ baseline + order + period + drug + (1 | patient),
  data = hypertension
```

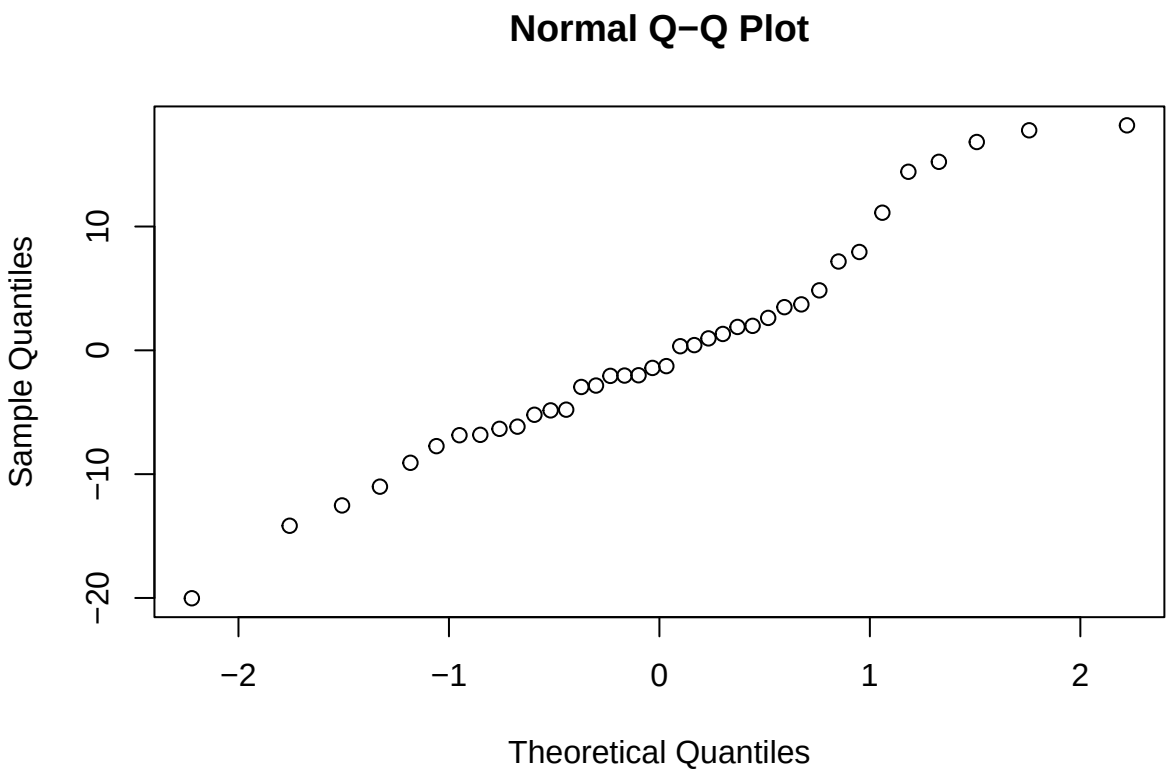
```
)
```

We create validation plots:

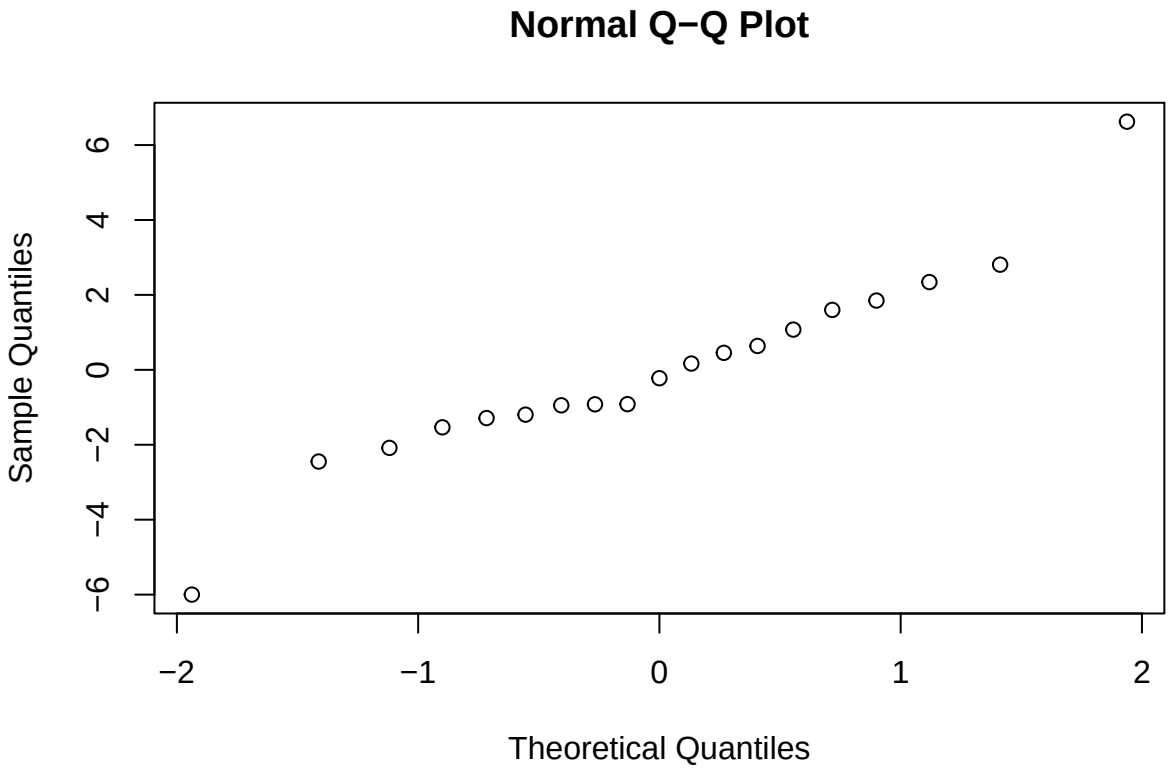
```
plot(m1)
```




```
qqnorm(residuals(m1))
```




```
qqnorm(ranef(m1)$patient[, 1])
```



All of the plots look okay, so the model is reasonable. We check if there is an effect of treatment:

```
drop1(m1, test = "Chisq")
```

```
## boundary (singular) fit: see help('isSingular')
```

```
## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.derivs =
## TRUE, : convergence code 3 from bobyqa: bobyqa -- a trust region step failed to
## reduce q
```

```
## Single term deletions
##
## Model:
## change ~ baseline + order + period + drug + (1 | patient)
##          npar    AIC    LRT Pr(Chi)
## <none>        300.81
## baseline    1 300.89 2.0705 0.15017
## order       1 299.02 0.2032 0.65212
## period      1 300.60 1.7836 0.18171
## drug        1 302.66 3.8433 0.04995 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

`drug` is indeed significant with a p-value of 0.04995 (however this is very weak evidence). We can compute em-means to determine the size of the effect:

```
library(emmeans)
emmeans(m1, ~drug)
```

```
##  drug emmean   SE   df lower.CL upper.CL
##  E     -7.46 2.66 31.7   -12.88    -2.04
##  N     -0.90 2.67 31.7    -6.34     4.54
##
## Results are averaged over the levels of: order, period
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

```
test(emmeans(m1, ~drug))
```

```
##  drug emmean   SE   df t.ratio p.value
##  E     -7.46 2.66 31.7  -2.807  0.0085
##  N     -0.90 2.67 31.7  -0.337  0.7381
##
## Results are averaged over the levels of: order, period
## Degrees-of-freedom method: kenward-roger
```

```
confint(pairs(emmeans(m1, ~drug)))
```

```
##  contrast estimate   SE   df lower.CL upper.CL
##  E - N       -6.56 3.43 17.1    -13.8    0.673
##
## Results are averaged over the levels of: order, period
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

It looks as if drug E lowers the blood pressure, whereas drug N does not appear to have any effect.

We were able to establish a significant ($p = 0.04995$) difference between drug E and N whereas the t-test done in Exercise 1.5 was not able to establish a significant difference ($p = 0.0932$). In this way the ANCOVA analysis is more powerful. The reason for this is, that model `m1` also controls for the baseline as well as the `period`.

However, if we would have done model selection before the hypothesis tests then we get the following result

```
drop1(m1, test = "Chisq")
```

```
## boundary (singular) fit: see help('isSingular')
```

```
## Warning in optwrap(optimizer, devfun, x@theta, lower = x@lower, calc.derivs =
## TRUE, : convergence code 3 from bobyqa: bobyqa -- a trust region step failed to
## reduce q
```

```
## Single term deletions
##
## Model:
## change ~ baseline + order + period + drug + (1 | patient)
##          npar    AIC    LRT Pr(Chi)
## <none>        300.81
## baseline    1 300.89 2.0705 0.15017
## order       1 299.02 0.2032 0.65212
## period      1 300.60 1.7836 0.18171
## drug        1 302.66 3.8433 0.04995 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
m2 <- update(m1, . ~ . - order)
drop1(m2, test = "Chisq")
```

```
## boundary (singular) fit: see help('isSingular')
```

```
## Single term deletions
##
## Model:
## change ~ baseline + period + drug + (1 | patient)
##          npar    AIC    LRT Pr(Chi)
## <none>        299.02
## baseline    1 299.51 2.4871 0.11478
## period      1 298.76 1.7459 0.18640
## drug        1 300.86 3.8438 0.04993 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
m3 <- update(m2, . ~ . - period)
drop1(m3, test = "Chisq")
```

```
## boundary (singular) fit: see help('isSingular')
```

```
## Single term deletions
##
## Model:
## change ~ baseline + drug + (1 | patient)
##          npar    AIC    LRT Pr(Chi)
## <none>        298.76
## baseline    1 299.77 3.0087 0.08282 .
## drug        1 299.73 2.9663 0.08502 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Thus, model selection will remove both order and period (eg. looking at the AIC value or p-value). Then we get model `m3`, where the effect of drug is non-significant ($p = 0.08502$). This p-value is comparable to what we found using t-tests (which gave $p = 0.0932$). The conclusion thus depends on whether we do model selection or not, so it is very delicate!

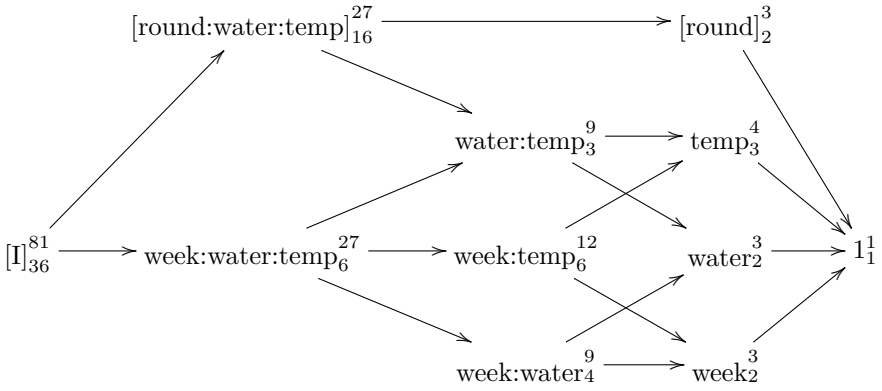
The ANCOVA model `m1` proposed above is somewhat complicated, however, only one model needs to be described and understood. On the other hand the 4 t-tests done in Exercise 1.5 are more simple by themselves but if all details are to be given, then it requires a lot of explanation. The ANCOVA model could be preferred for these reasons; here all effects can be explained and understood in a single model.

Exercise 5.2 Random effects model

Problem

In an experiment about production of milk powder two factors were varied: water activity on three levels (`water`=1, 2, 3) and temperature while drying on 4 levels (`temp`=100, 110, 120, 140 degrees Celsius). Only 9 of the 12 combinations were tested in the experiment. There were three replications in the experiment, in the sense that milk powder was prepared in three rounds (`round`=1, 2, 3). This gives $3 \cdot 9 = 27$ samples of milk powder in total. Each of these were stored and measurements were taken after 4, 6 and 8 weeks. Each time the concentration of maillard reaction products as well as a sensory taste score (high values means good taste) were measured. The dataset consists of 81 observations, which are listed in Table~2 and can be found in the text file `milk.txt`. In this exercise we will analyze the response `maillard`.

The factor `round` can be considered a block factor, and should be used as a random factor in this exercise[2]. The 3-way interaction `round:water:temp` corresponds to the grouping of the 81 observations into the 27 different samples. Since the samples are measured 3 times (i.e. repeated measurements) it is also standard to make this 3-way interaction a random factor. Thus, the design diagram for the experiment is as follows:



Recall, that the superscripts designate the number of levels, and that the subscripts designate the corresponding degrees of freedom. Please have a closer look at the design diagram, which contains the random effects of `round:water:temp` and of `round`, and the full factorial design of the fixed effects `temp`, `water` and `week`. Can you see how it relates to the description of the experiment given above?

The model described above is fitted in R using the following call to lmer():[3]

```
lmer(maillard ~ week * water * temp + (1 | round:water:temp) + (1 | round), data=milk)
```

Here we assume that the data is available in a data frame called `milk`, where the explanatory variables are encoded as factors.

Now analyze the relation between `maillard` and the 3 explanatory factors using the following 4 steps:

1. Fit the initial model using `lmer()`.
2. Validate the initial model as proposed in the lectures.

[2]This violates the "rule of thumb" that factors with less than 5 levels can be used with fixed effect. However, this is also a matter of preference. Moreover, using `round` as a random effect also let you try a model with more than one random effect.

[3]If your version of lme4 is older than 1.1-6, then this call might result in an error message like this `"Error in lme4::lFormula(formula = maillard ~ week * water * temp + (1 | : rank of X = 27 < ncol(X) = 36"`. If you encounter this error message, then you should update your installation of the lme4-package.

Round	Week	Maillard	Taste	Water	Temp	Round	Week	Maillard	Taste	Water	Temp
1	4	2.90	10.1	1	100	2	6	2.11	11.2	3	100
1	4	2.13	11.0	1	110	2	6	1.98	11.8	3	110
1	4	2.00	11.1	1	120	2	6	2.20	11.0	3	140
1	4	2.13	11.1	2	100	3	6	2.20	7.0	1	100
1	4	2.38	11.9	2	120	3	6	2.34	10.7	1	110
1	4	2.56	10.7	2	140	3	6	2.49	10.3	1	120
1	4	2.60	10.8	3	100	3	6	2.63	9.7	2	100
1	4	1.91	11.0	3	110	3	6	3.06	9.0	2	120
1	4	2.27	10.8	3	140	3	6	3.28	9.6	2	140
2	4	2.19	11.0	1	100	3	6	2.34	10.2	3	100
2	4	2.32	11.0	1	110	3	6	2.51	9.2	3	110
2	4	2.41	11.6	1	120	3	6	2.77	10.2	3	140
2	4	2.49	11.1	2	100	1	8	2.39	9.6	1	100
2	4	2.61	11.7	2	120	1	8	2.41	9.8	1	110
2	4	2.63	10.8	2	140	1	8	2.71	11.4	1	120
2	4	2.06	11.0	3	100	1	8	2.49	11.2	2	100
2	4	1.98	10.0	3	110	1	8	2.06	11.2	2	120
2	4	2.27	11.2	3	140	1	8	3.10	9.8	2	140
3	4	2.13	10.1	1	100	1	8	2.32	10.8	3	100
3	4	2.13	9.4	1	110	1	8	2.29	9.4	3	110
3	4	2.22	10.7	1	120	1	8	2.72	12.0	3	140
3	4	2.80	8.3	2	100	2	8	2.27	11.0	1	100
3	4	2.77	10.9	2	120	2	8	2.25	11.2	1	110
3	4	2.99	9.2	2	140	2	8	2.46	9.6	1	120
3	4	1.98	10.3	3	100	2	8	2.53	9.2	2	100
3	4	1.98	9.3	3	110	2	8	2.70	11.0	2	120
3	4	2.20	10.5	3	140	2	8	2.81	11.6	2	140
1	6	2.13	10.0	1	100	2	8	2.20	11.8	3	100
1	6	2.34	10.5	1	110	2	8	2.15	10.6	3	110
1	6	2.49	11.2	1	120	2	8	2.41	11.4	3	140
1	6	2.41	10.8	2	100	3	8	2.41	9.6	1	100
1	6	2.85	11.2	2	120	3	8	2.42	9.0	1	110
1	6	2.84	11.2	2	140	3	8	2.73	10.2	1	120
1	6	2.24	8.4	3	100	3	8	3.33	7.8	2	100
1	6	2.06	11.4	3	110	3	8	3.25	9.4	2	120
1	6	2.42	11.6	3	140	3	8	3.75	9.6	2	140
2	6	2.20	9.3	1	100	3	8	2.80	10.6	3	100
2	6	2.27	11.3	1	110	3	8	2.81	10.2	3	110
2	6	2.49	11.7	1	120	3	8	3.06	10.0	3	140
2	6	2.34	11.2	2	100						
2	6	2.70	10.8	2	120						
2	6	2.61	11.0	2	140						

Table 2: The milk powder data

3. Do backward model reduction of the fixed effects using `drop1(.,test="Chisq")` and `update()` as exemplified in the lectures.
4. Report em-means for the final model.

(Reference: Exercise 8.7 in Sørensen & Tolver: *Lecture Notes for Applied Statistics*.)

Solution

We load the data, encode variables as factors and fit the initial model:

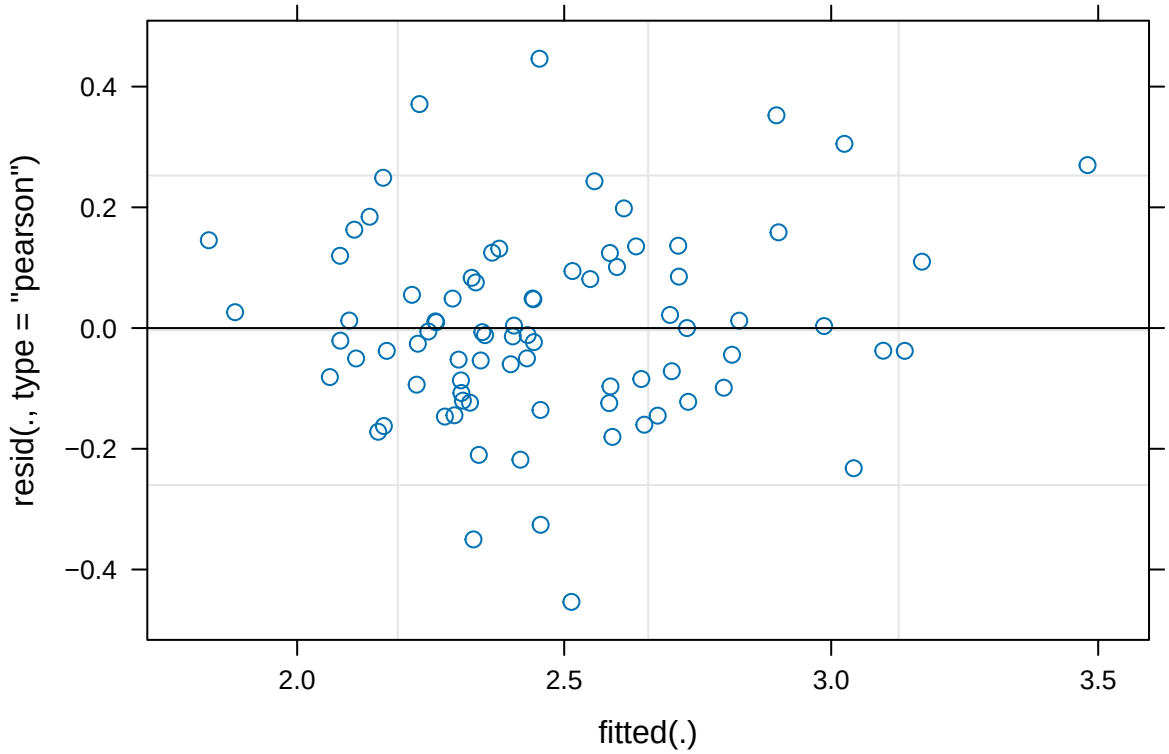
```
milk <- read.delim("milk.txt")
milk$round <- factor(milk$round)
milk$week  <- factor(milk$week)
milk$water <- factor(milk$water)
milk$temp  <- factor(milk$temp)
```

```
library(lme4)
m0 <- lmer(
  maillard ~ week * water * temp + (1 | round:water:temp) + (1 | round),
  data = milk
)
```

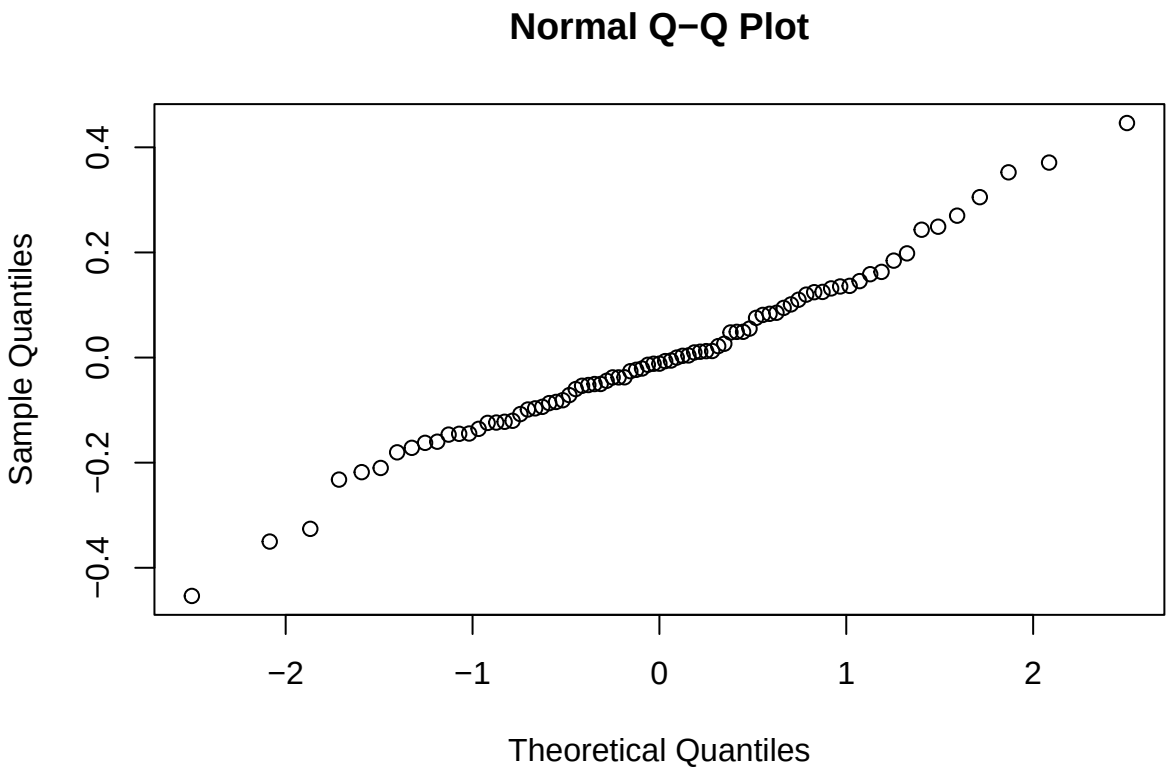
```
## fixed-effect model matrix is rank deficient so dropping 9 columns / coefficients
```

We create validation plots:

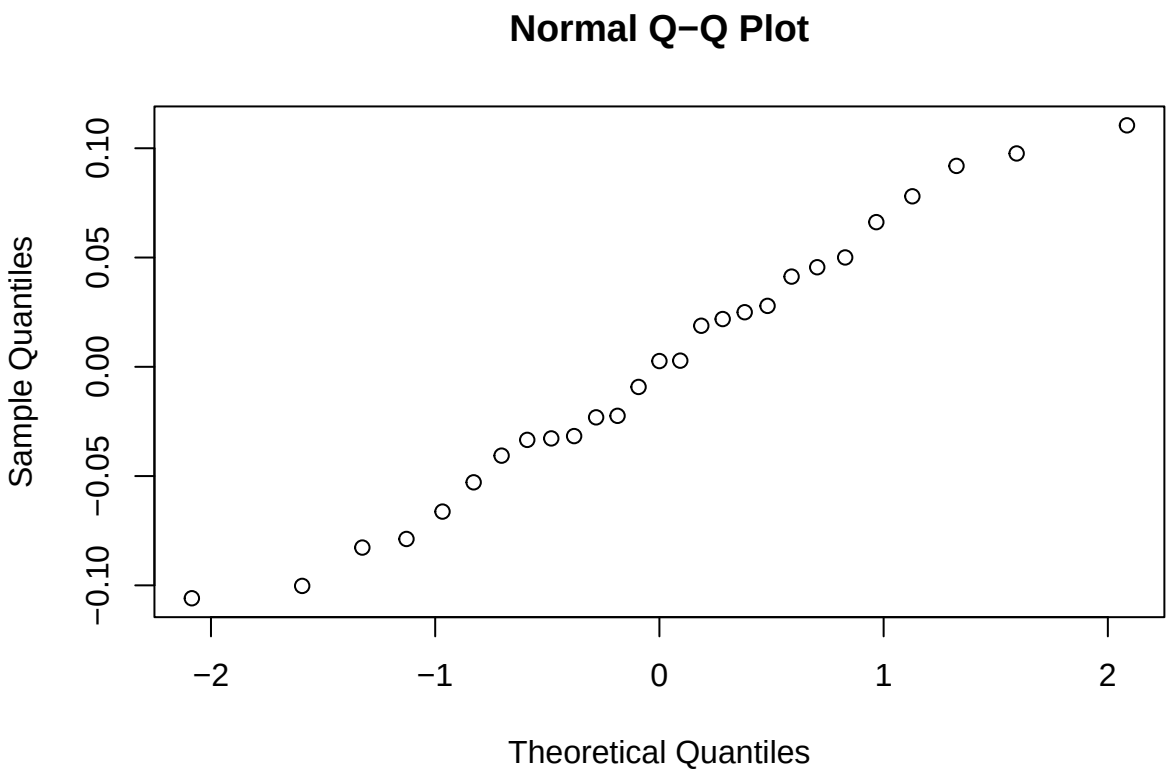
```
plot(m0)
```



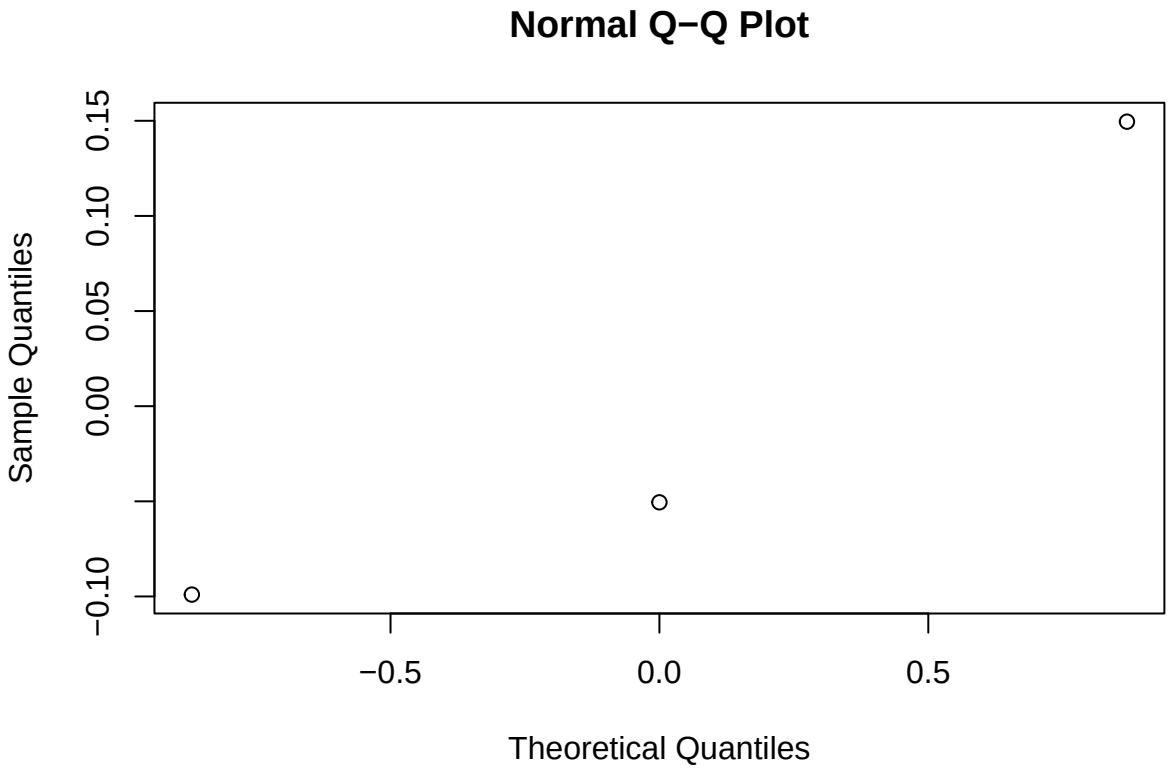

```
qqnorm(residuals(m0))
```




```
qqnorm(ranef(m0)$"round:water:temp"[, 1])
```

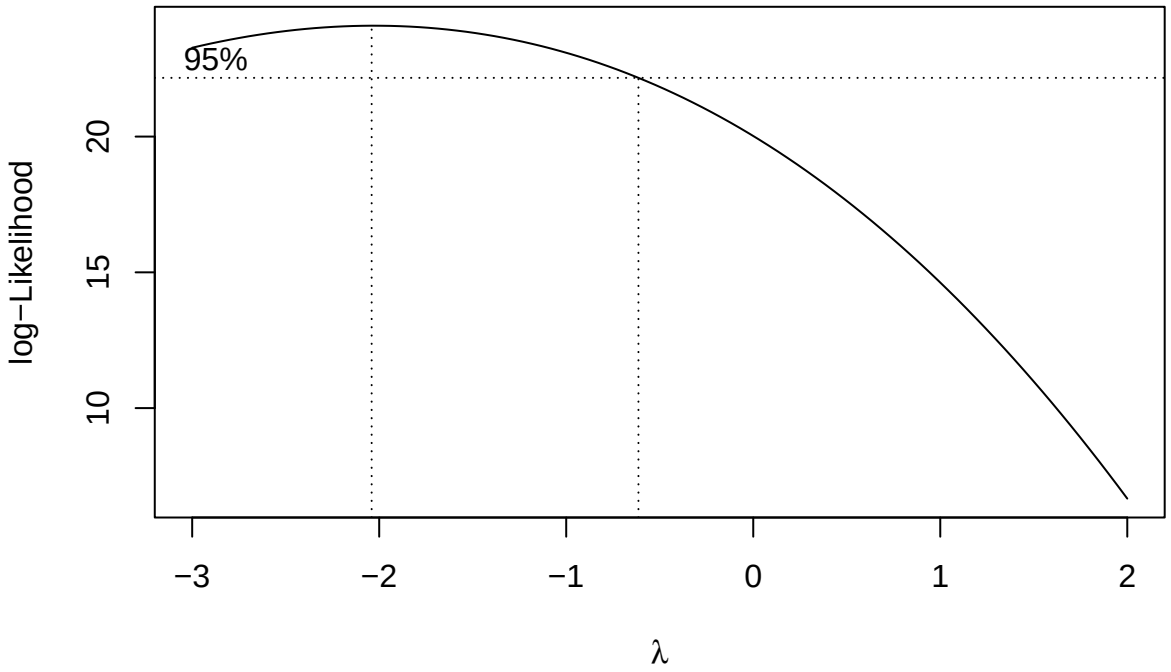



```
qqnorm(ranef(m0)$"round"[, 1])
```



Residual plot does not look too nice, so we would like to try a Box-Cox analysis. However, `MASS::boxcox()` does not work on `lmer`-models. We instead do the Box-Cox analysis on the corresponding `lm`-model without the random effects

```
library(MASS)
boxcox(lm(maillard ~ week * water * temp, data = milk),
  lambda = seq(-3, 2, 0.1)
)
```

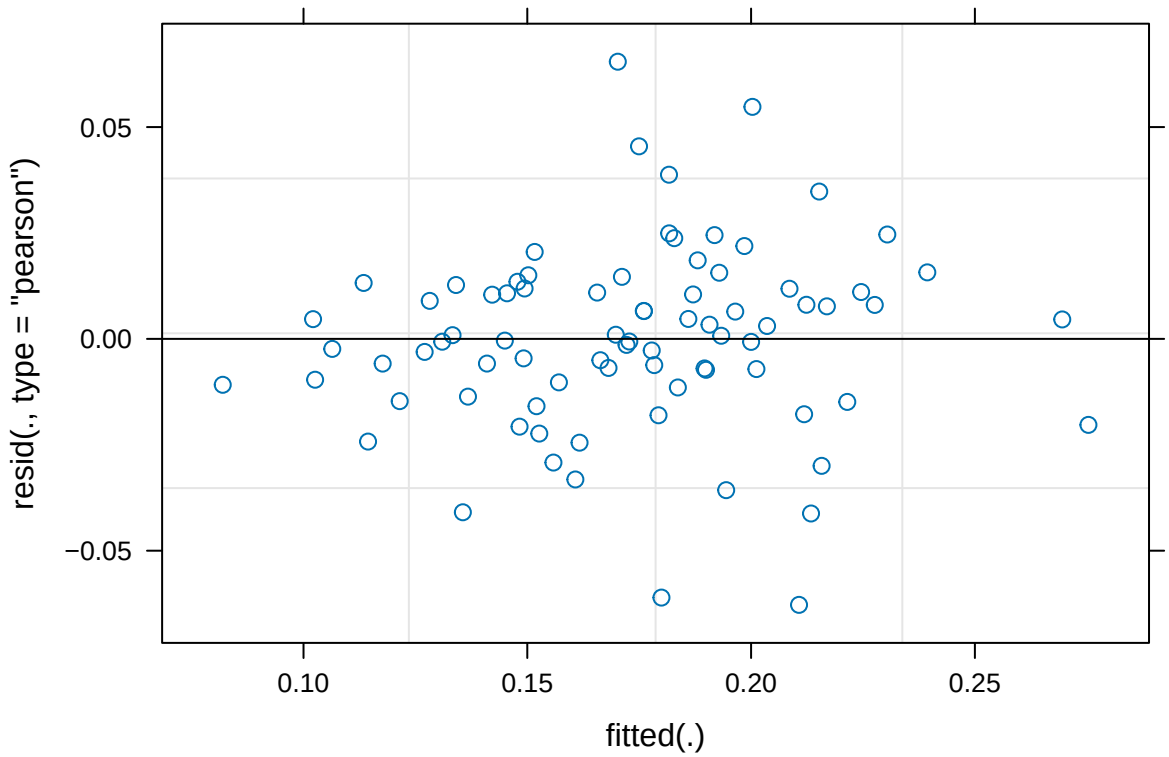


Box-Cox suggests that we analyze $1/(\texttt{maillard}^2)$ (but $1/\texttt{maillard}$ also looks reasonable and perhaps even a log transform could work). We fit a new model and validate it:

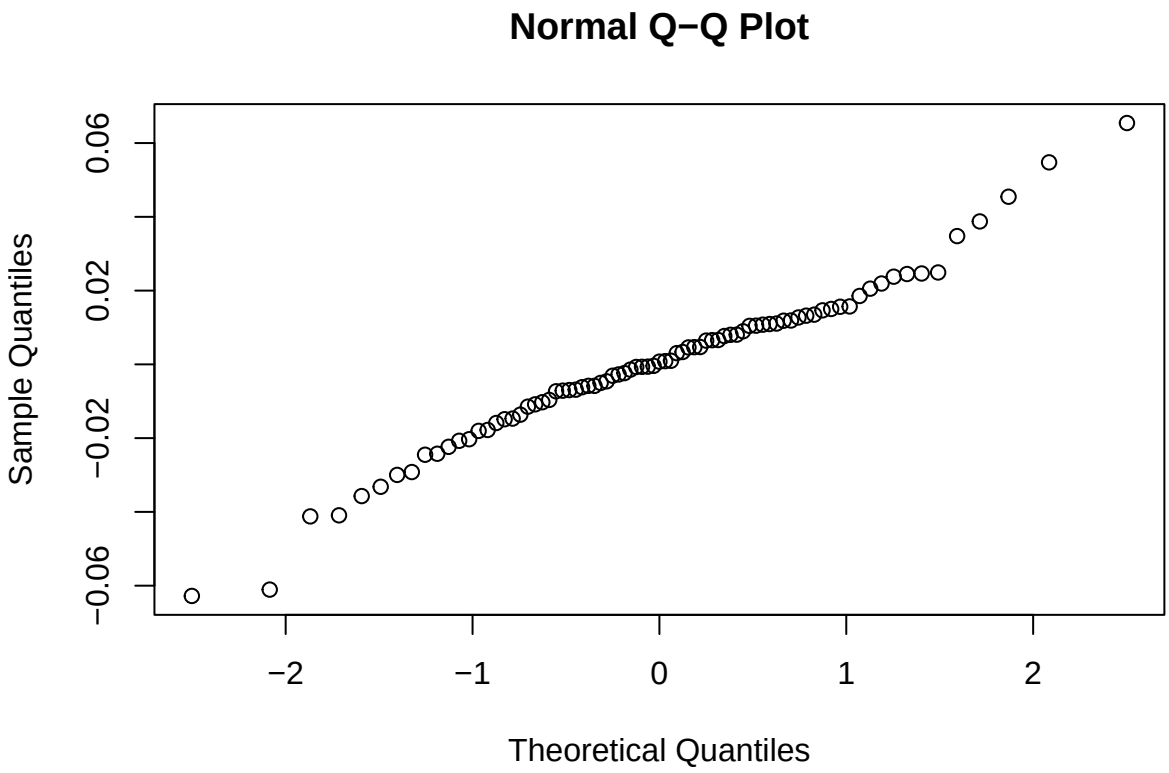
```
m1 <- lmer(
  1 / maillard^2 ~ week * water * temp + (1 | round:water:temp) + (1 | round),
  data = milk
)
```

```
## fixed-effect model matrix is rank deficient so dropping 9 columns / coefficients
```

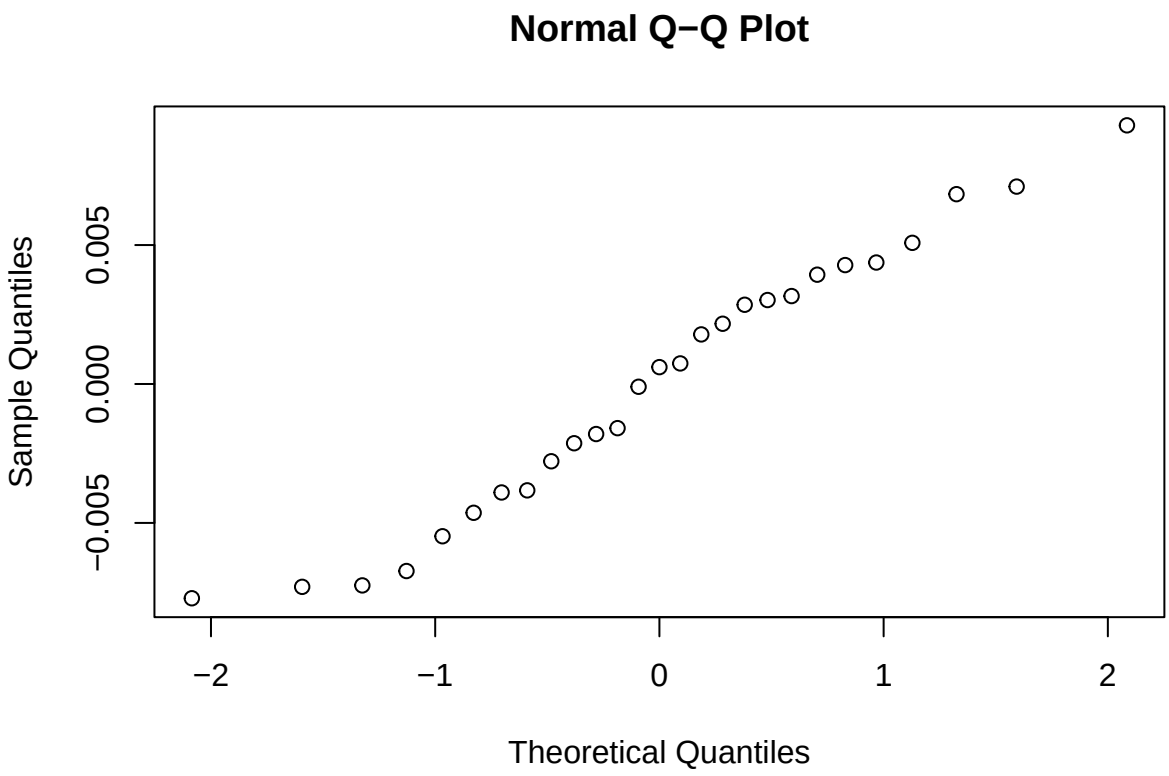
```
plot(m1)
```



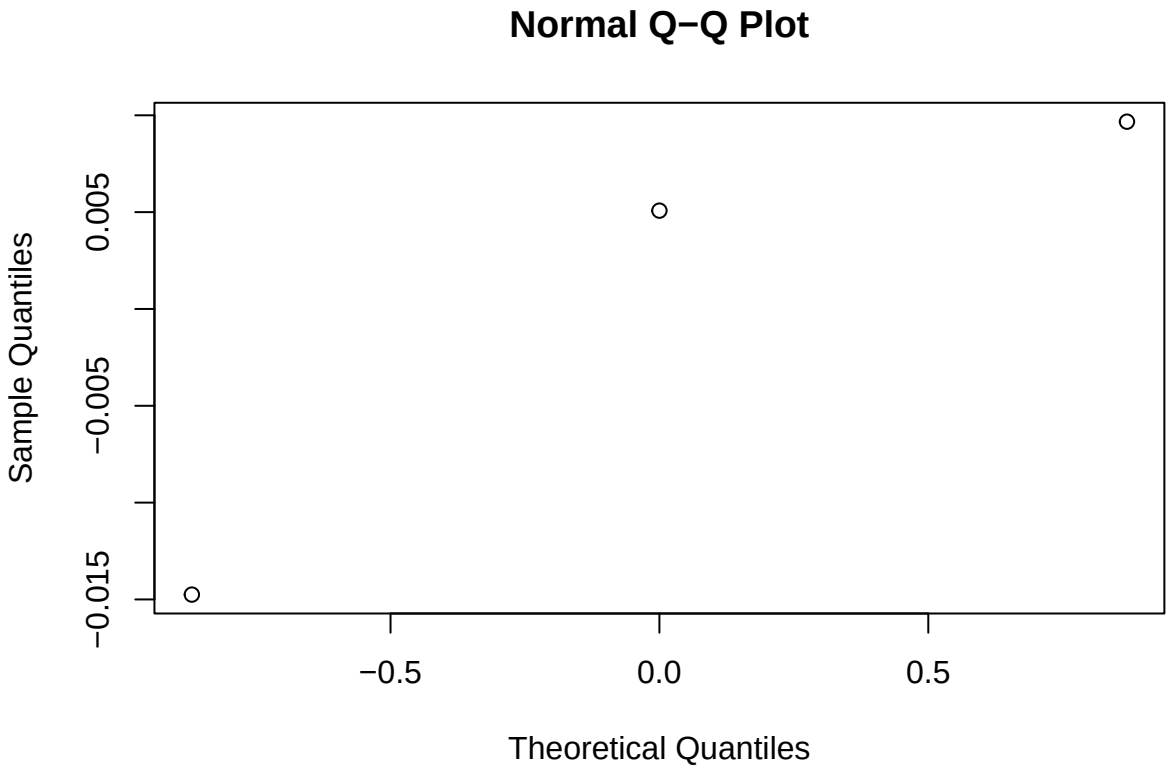

```
qqnorm(residuals(m1))
```




```
qqnorm(ranef(m1)$"round:water:temp"[, 1])
```




```
qqnorm(ranef(m1)$"round"[, 1])
```



Unfortunately, this model does not look much better than the one we started with so we will proceed with the original model `m0`. We perform backwards model selection:

```
drop1(m0, test = "Chisq")
```

```
## fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
```

```
## Single term deletions
##
## Model:
## maillard ~ week * water * temp + (1 | round:water:temp) + (1 |
##     round)
##                 npar    AIC    LRT Pr(Chi)
## <none>               27.728
## week:water:temp    6 23.299 7.5718  0.2712
```

```
m2 <- update(m0, . ~ . - week:water:temp)
```

```
## fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
```

```
drop1(m2, test = "Chisq")
```

```
## fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
## fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
```

```
## Single term deletions
##
## Model:
## maillard ~ week + water + temp + (1 | round:water:temp) + (1 |
##     round) + week:water + week:temp + water:temp
##            npar    AIC     LRT Pr(Chi)
## <none>          23.299
## week:water    4 17.102  1.8022  0.7721
## week:temp     6 21.815 10.5154  0.1046
## water:temp    3 18.890  1.5909  0.6615
```

```
m3 <- update(m2, . ~ . - week:water)
```

```
## fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
```

```
drop1(m3, test = "Chisq")
```

```
## fixed-effect model matrix is rank deficient so dropping 3 columns / coefficients
```

```
## Single term deletions
##
## Model:
## maillard ~ week + water + temp + (1 | round:water:temp) + (1 |
##     round) + week:temp + water:temp
##            npar    AIC     LRT Pr(Chi)
## <none>          17.102
## week:temp     6 16.322 11.2203  0.0818 .
## water:temp    3 12.693  1.5909  0.6615
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
m4 <- update(m3, . ~ . - water:temp)
```

```
drop1(m4, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## maillard ~ week + water + temp + (1 | round:water:temp) + (1 |
##     round) + week:temp
```

```
##           npar    AIC    LRT   Pr(Chi)
## <none>         12.693
## water        2 27.687 18.994 7.506e-05 ***
## week:temp    6 11.913 11.220    0.0818 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
m5 <- update(m4, . ~ . - week:temp)
```

```
drop1(m5, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## maillard ~ week + water + temp + (1 | round:water:temp) + (1 |
##     round)
##        npar    AIC    LRT   Pr(Chi)
## <none>      11.913
## week      2 29.283 21.370 2.289e-05 ***
## water     2 26.907 18.994 7.506e-05 ***
## temp      3 19.697 13.784  0.003215 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

All three main effects are highly significant (both by AIC and p-values). We compute em-means for each of the factors. First for week:

```
emmeans(m5, ~week)
```

```
##  week emmean    SE   df lower.CL upper.CL
##  4      2.34 0.094 2.64     2.02     2.67
##  6      2.47 0.094 2.64     2.14     2.79
##  8      2.63 0.094 2.64     2.31     2.96
##
## Results are averaged over the levels of: water, temp
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

```
multcomp::cld(emmeans(m5, ~week), Letters = letters)
```

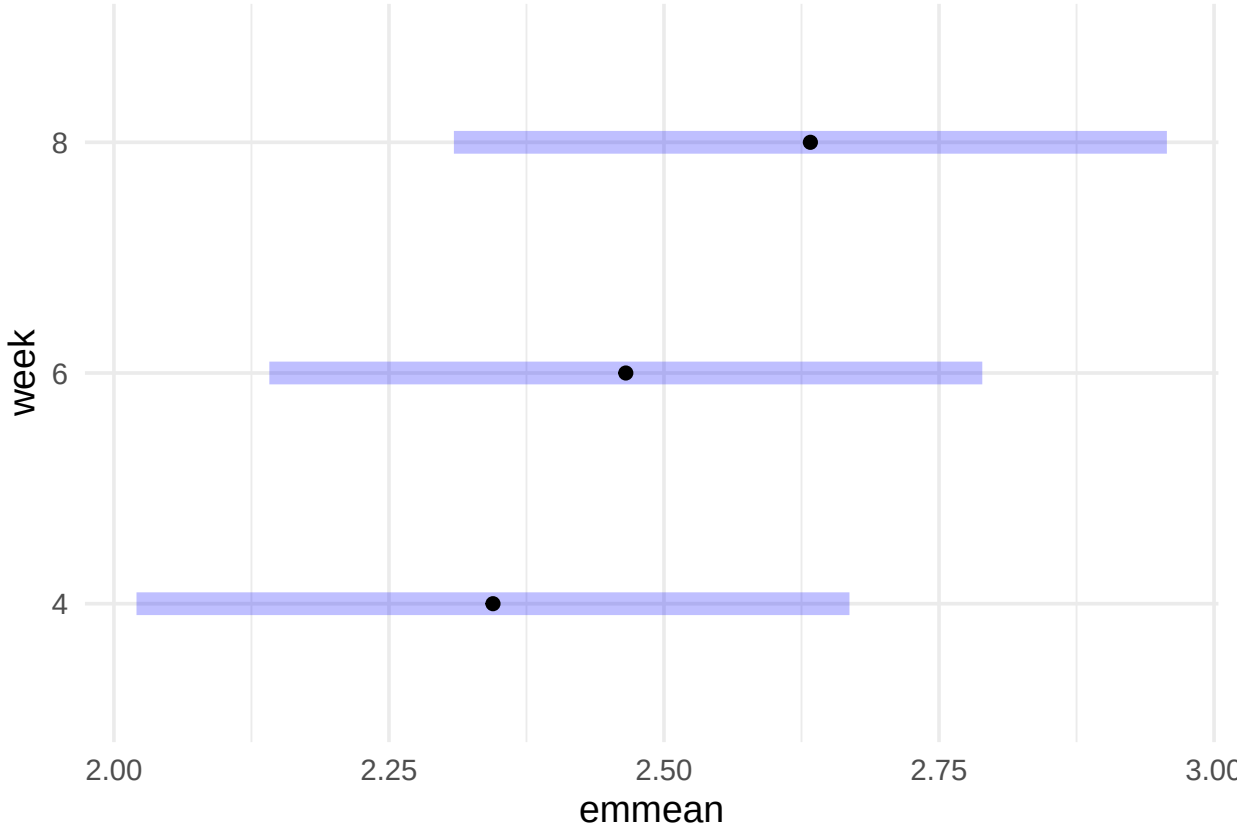
```
##  week emmean    SE   df lower.CL upper.CL .group
##  4      2.34 0.094 2.64     2.02     2.67  a
##  6      2.47 0.094 2.64     2.14     2.79  a
##  8      2.63 0.094 2.64     2.31     2.96   b
##
## Results are averaged over the levels of: water, temp
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 3 estimates
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
```

```
pairs(emmeans(m5, ~week))
```

```
##  contrast      estimate     SE df t.ratio p.value
##  week4 - week6   -0.121 0.0578 52  -2.090  0.1018
```

```
##  week4 - week8   -0.289 0.0578 52  -4.994  <.0001
##  week6 - week8   -0.168 0.0578 52  -2.904  0.0147
##
## Results are averaged over the levels of: water, temp
## Degrees-of-freedom method: kenward-roger
## P value adjustment: tukey method for comparing a family of 3 estimates
```

```
plot(emmeans(m5, ~week))
```



We see that although confidence intervals are overlapping for the different weeks, we still obtain a significant result in the pairwise comparison. The reason for this is that the confidence intervals for the em-means themselves also includes the uncertainty on the estimation of the remaining factors. These, however, disappear in the pairwise comparisons within `week`. We do the same for `water`:

```
emmeans(m5, ~water)
```

```
##  water emmean     SE  df lower.CL upper.CL
##  1       2.42 0.0997 3.3     2.12     2.72
##  2       2.70 0.0997 3.3     2.40     3.00
##  3       2.33 0.0997 3.3     2.02     2.63
##
## Results are averaged over the levels of: week, temp
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

```
multcomp::cld(emmeans(m5, ~water), Letters = letters)
```

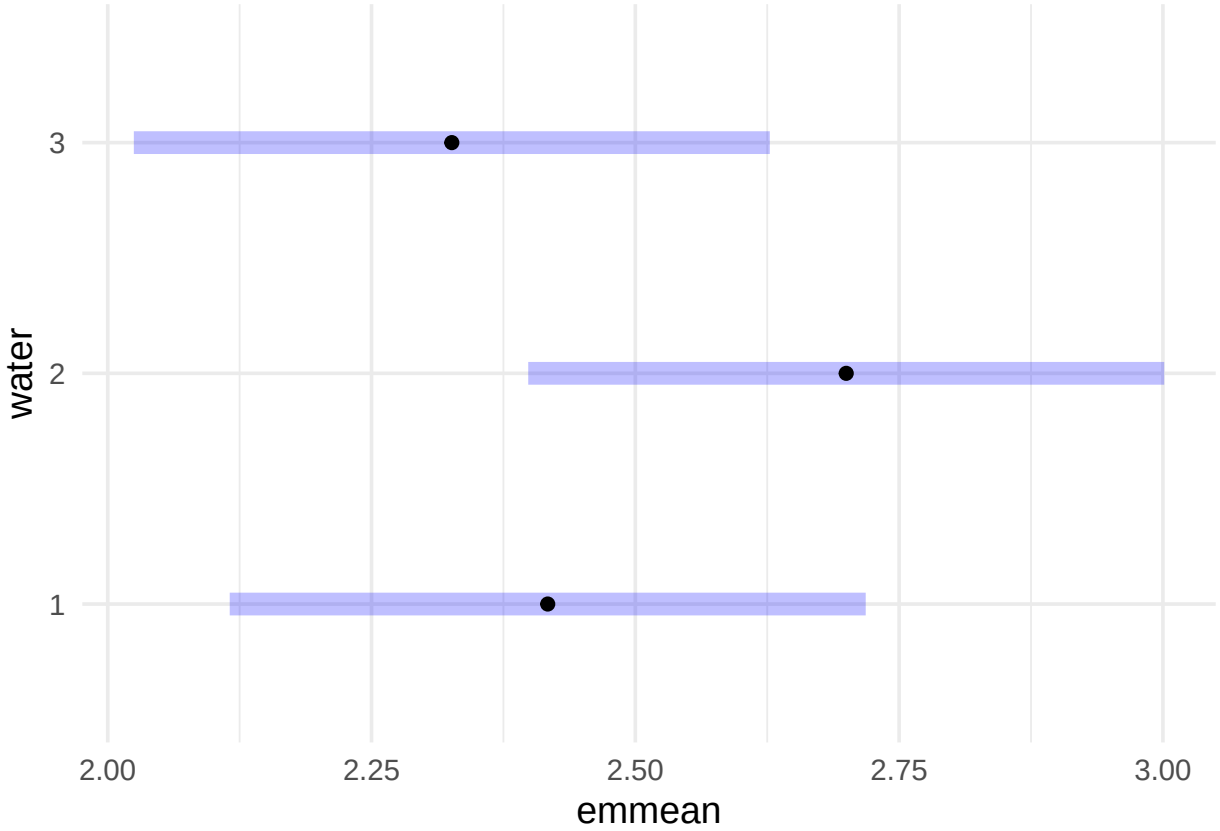
```
##  water emmean     SE  df lower.CL upper.CL .group
##  3       2.33 0.0997 3.3     2.02     2.63  a
##  1       2.42 0.0997 3.3     2.12     2.72  a
```

```
##  2       2.70 0.0997 3.3     2.40     3.00   b
##
## Results are averaged over the levels of: week, temp
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 3 estimates
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
```

```
pairs(emmeans(m5, ~water))
```

```
##  contrast        estimate     SE df t.ratio p.value
##  water1 - water2  -0.2829 0.0814 19  -3.474  0.0068
##  water1 - water3   0.0909 0.0814 19   1.116  0.5161
##  water2 - water3   0.3738 0.0814 19   4.590  0.0006
##
## Results are averaged over the levels of: week, temp
## Degrees-of-freedom method: kenward-roger
## P value adjustment: tukey method for comparing a family of 3 estimates
```

```
plot(emmeans(m5, ~water))
```



A similar pattern emerges where clearly group 2 is different from 1 and 3 but their confidence intervals are overlapping. Finally for `temp`:

```
emmeans(m5, ~temp)
```

```
##  temp emmean     SE   df lower.CL upper.CL
```

```
##  100    2.39 0.0977 3.05     2.09     2.70
##  110    2.35 0.1071 4.33     2.06     2.64
##  120    2.50 0.1071 4.33     2.21     2.79
##  140    2.68 0.1071 4.33     2.40     2.97
##
## Results are averaged over the levels of: week, water
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
```

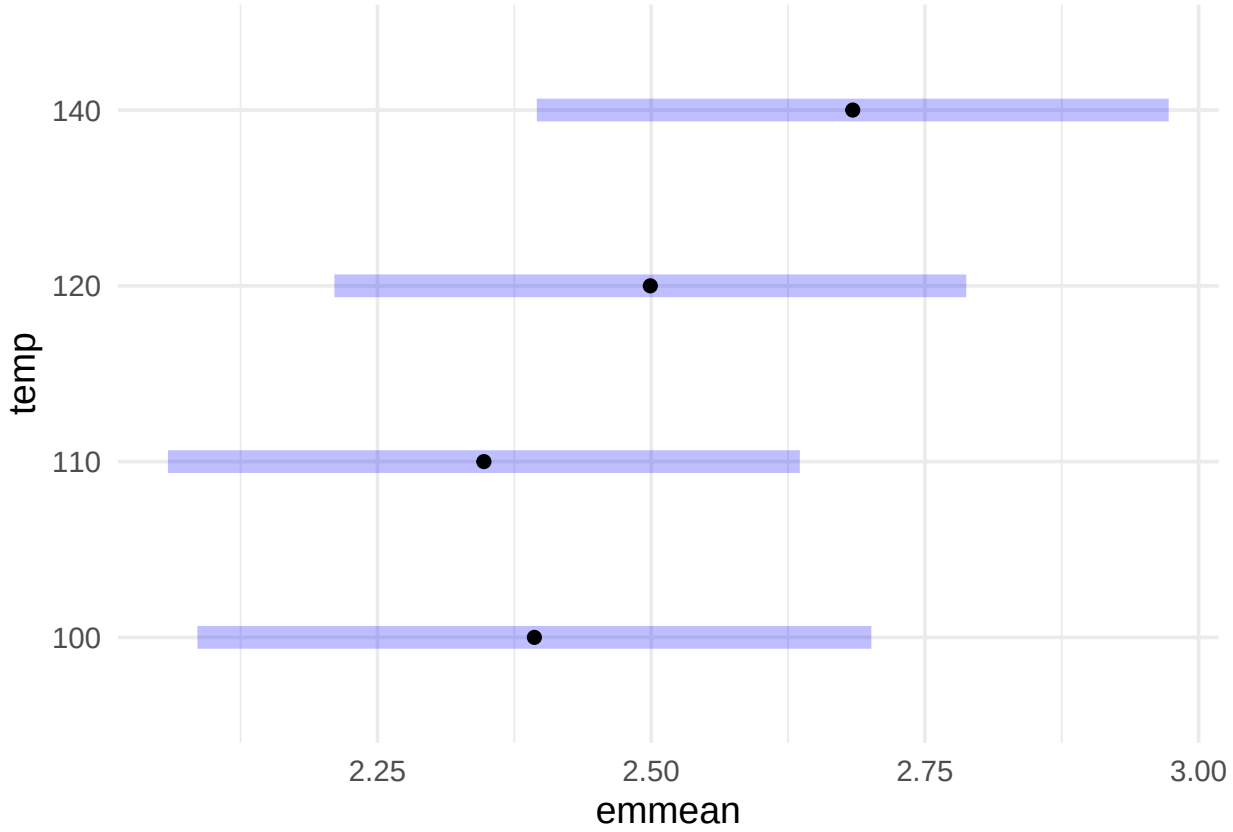
```
multcomp::cld(emmeans(m5, ~temp), Letters = letters)
```

```
##  temp emmean     SE   df lower.CL upper.CL .group
##  110    2.35 0.1071 4.33     2.06     2.64  a
##  100    2.39 0.0977 3.05     2.09     2.70  a
##  120    2.50 0.1071 4.33     2.21     2.79  ab
##  140    2.68 0.1071 4.33     2.40     2.97   b
##
## Results are averaged over the levels of: week, water
## Degrees-of-freedom method: kenward-roger
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 4 estimates
## significance level used: alpha = 0.05
## NOTE: If two or more means share the same grouping symbol,
##       then we cannot show them to be different.
##       But we also did not show them to be the same.
```

```
pairs(emmeans(m5, ~temp))
```

```
##  contrast          estimate     SE df t.ratio p.value
##  temp100 - temp110   0.0461 0.0864 19   0.534  0.9497
##  temp100 - temp120  -0.1059 0.0864 19  -1.226  0.6186
##  temp100 - temp140  -0.2908 0.0864 19  -3.367  0.0157
##  temp110 - temp120  -0.1520 0.0997 19  -1.524  0.4434
##  temp110 - temp140  -0.3369 0.0997 19  -3.378  0.0153
##  temp120 - temp140  -0.1849 0.0997 19  -1.854  0.2803
##
## Results are averaged over the levels of: week, water
## Degrees-of-freedom method: kenward-roger
## P value adjustment: tukey method for comparing a family of 4 estimates
```

```
plot(emmeans(m5, ~temp))
```



We see that temperature 140 is different from both 100 and 110 but 120 is similar to either group.

Exercise 5.3 Logistic regression with overdispersion

Problem

20 individuals have participated in an experiment where two different diets are to be compared. By randomization 10 people have been assigned to each diet and every week a *weight gain* or *weight loss* has been observed. The observations are the number of weeks where the diet resulted in a weight loss for each of the 20 individuals in the experiment. The table below displays the results for a period of eight weeks showing the number of people for each combination of diet and weeks with weight loss:

No. of weeks with weight loss	0	1	2	3	4	5	6	7	8
Diet 1	1	0	2	0	1	1	2	0	3
Diet 2	2	1	0	1	2	1	2	1	0

The dataset available in the text file `diet.txt` table contains four variables:

Variable	**Type**	**Range**	**Usage**
`person`	nominal	20 levels	random effect
`diet`	nominal	1, 2	fixed effect
`gain`	count	$0, \ldots, 8$	response
`loss`	count	$0, \ldots, 8$	response

Fit a logistic regression to the dataset and answer the following questions:

- Is there an indication of overdispersion?
- What is the p-value for the effect of diet on the probability of weight loss in each week?
- What is the odds ratio for weight loss between the two diets? Please answer this question even if the effect of diet is non-significant.

Above you should find strong evidence for overdispersion. Actually this is visible by the naked eye when looking at the data table (if you know what to look for).

(Reference: Exercise 8.20 in Sørensen & Jensen: *Lecture Notes for Applied Statistics*.)

Solution

We load the data and encode factors:

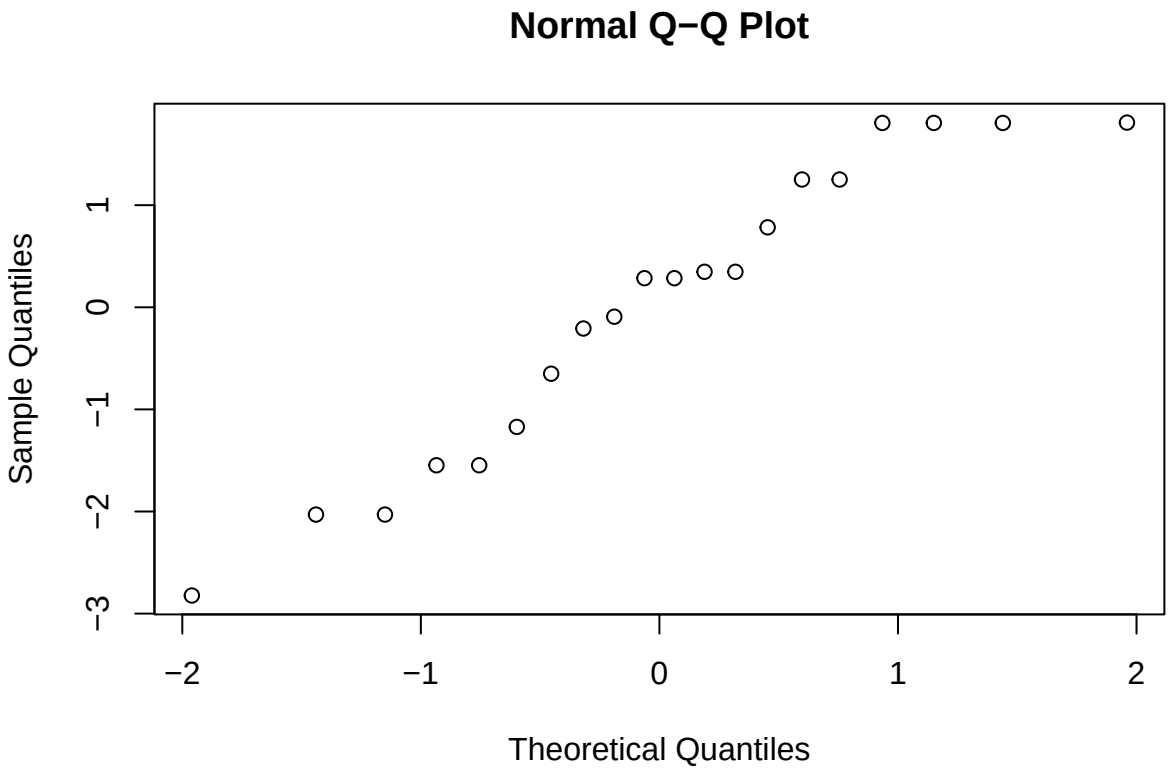
```
diet <- read.delim("diet.txt")
diet$person <- factor(diet$person)
diet$diet <- factor(diet$diet)
```

We start by fitting a logistic mixed model:

```
m1 <- glmer(
  cbind(loss, gain) ~ diet + (1 | person),
  family = binomial,
  data = diet
)
```

The model cannot be misspecifed in the mean but we still need to check for normality of the random effects:

```
qqnorm(ranef(m1)$person[, 1])
```



This looks okay. To check for overdispersion, we compare to a logistic regression model without the random effect:

```
m0 <- glm(cbind(loss, gain) ~ diet, family = binomial, data = diet)
anova(m1, m0)
```

```
## Data: diet
## Models:
## m0: cbind(loss, gain) ~ diet
```

```
## m1: cbind(loss, gain) ~ diet + (1 | person)
##    npar     AIC     BIC  logLik deviance  Chisq Df Pr(>Chisq)
## m0    2 122.974 124.965 -59.487  118.974
## m1    3  93.149  96.136 -43.574   87.149 31.825  1  1.687e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We cannot go from `m1` to `m0` ($p \approx 10^{-8}$) so this indicates that overdispersion is present. We can test whether there is an effect of `diet`:

```
drop1(m1, test = "Chisq")
```

```
## Single term deletions
##
## Model:
## cbind(loss, gain) ~ diet + (1 | person)
##        npar    AIC    LRT Pr(Chi)
## <none>      93.149
## diet      1 92.762 1.6137   0.204
```

We see no evidence that there is an effect of `diet`. We can estimate the odds ratio using em-means:

```
confint(pairs(emmeans(m1, ~diet, type = "response")))
```

```
##  contrast      odds.ratio   SE  df asymp.LCL asymp.UCL
##  diet1 / diet2       3.18 2.89 Inf     0.535      18.9
##
## Confidence level used: 0.95
## Intervals are back-transformed from the log odds ratio scale
```

Thus, the odds for weight loss is estimated to be higher in diet 1, although the odds ratio is not significantly different from 1 as seen above.