Day6 exercise solutions

Ali Movasati

Oct. 21st, 2024

```
# Set global code chunk options
knitr::opts_chunk$set(warning = FALSE)
```

```
# load required libraries
library(skimr)
library(ggplot2)
library(ggpubr)
library(magrittr)
library(tidyr)
library(dplyr)
library(tibble)
library(lme4)
library(lattice)


# define functions
`%notin%` <- Negate(`%in%`)
```

Problem 1

```
# read in the data

hearing <- read.table(file = "/Users/alimos313/Documents/studies/phd/university/courses/stat-modelling/S
```

1.A)

```
# print descriptive
skim(hearing)
```

Table 1: Data summary

Name	hearing
Number of rows	96
Number of columns	3
Column type frequency:	
character	1
numeric	2
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
ListID	0	1	5	5	0	4	0

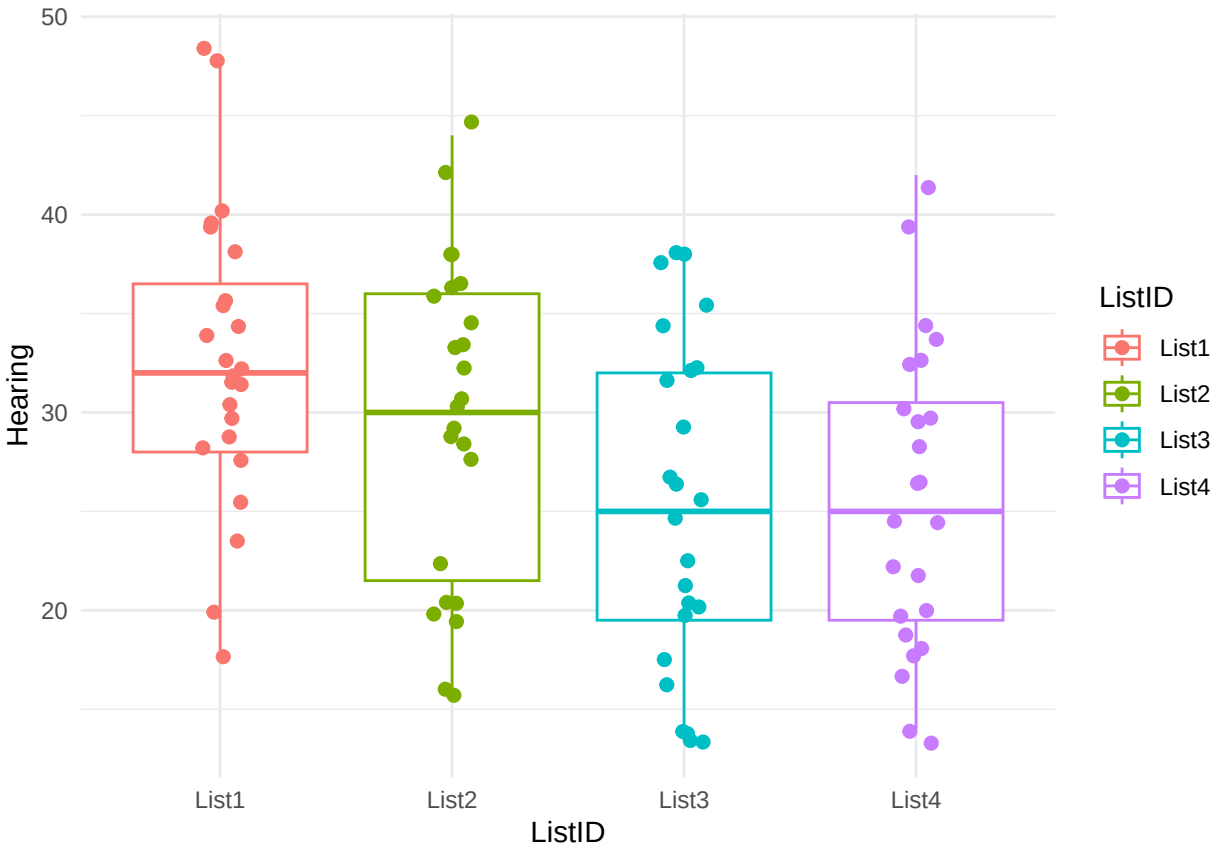
Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
SubjectID	0	1	12.50	6.96	1	6.75	12.5	18.25	24	
Hearing	0	1	28.31	8.37	14	20.00	30.0	34.00	48	

```
table(hearing$ListID)
```

```
##
## List1 List2 List3 List4
##    24    24    24    24
```

```
# prepare data

hearing %<>% mutate(ListID = as.factor(ListID), SubjectID = as.factor(SubjectID))

# plot a box-plot for visualization

hearing %>% ggplot(aes(x = ListID, y = Hearing, color = ListID)) +
        geom_boxplot() +
        geom_jitter(width = 0.1, aes(color = factor(ListID)), size = 2) +
        theme_minimal()
```



1.B)

```
lm_simple <- lm(Hearing ~ 1 + ListID, data = hearing)

sum_model_simple <- summary(lm_simple)

print(sum_model_simple)
```

```
##
## Call:
## lm(formula = Hearing ~ 1 + ListID, data = hearing)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.7500  -5.5833  -0.2083   6.3333  16.4167
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   32.750      1.612  20.315  < 2e-16 ***
## ListIDList2   -3.083      2.280  -1.352  0.17955
## ListIDList3   -7.500      2.280  -3.290  0.00142 **
## ListIDList4   -7.167      2.280  -3.144  0.00225 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.898 on 92 degrees of freedom
```

```
## Multiple R-squared:  0.1382, Adjusted R-squared:  0.1101
## F-statistic: 4.919 on 3 and 92 DF,  p-value: 0.00325
```

« comments »

Only 11.01% of variability in hearing measures are explained by different lists

We have enough evidence to state that the mean hearing score for List 3 and 4 are different than list 1, while for list 2 we cannot state that!

1.C)

```
# fit the mixed model
lm_mixed <- lmer(Hearing ~ 1 + ListID + (1|SubjectID), data = hearing)

sum_model_mixed <- summary(lm_mixed)

print(sum_model_mixed)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Hearing ~ 1 + ListID + (1 | SubjectID)
##    Data: hearing
##
## REML criterion at convergence: 635.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.86533 -0.56158 -0.01092  0.63222  2.69167
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  SubjectID (Intercept) 26.04    5.103
##  Residual              36.33    6.027
## Number of obs: 96, groups:  SubjectID, 24
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)   32.750      1.612  20.315
## ListIDList2   -3.083      1.740  -1.772
## ListIDList3   -7.500      1.740  -4.311
## ListIDList4   -7.167      1.740  -4.119
##
## Correlation of Fixed Effects:
##             (Intr) LsIDL2 LsIDL3
## ListIDList2 -0.540
## ListIDList3 -0.540  0.500
## ListIDList4 -0.540  0.500  0.500
```

```
# fit the model without ListID (null model)
lm_mixed_null <- lmer(Hearing ~ (1 | SubjectID), data = hearing)

sum_model_mixed_null <- summary(lm_mixed_null)

# Likelihood ratio test
anova(lm_mixed_null, lm_mixed)
```

```
## refitting model(s) with ML (instead of REML)
## Data: hearing
## Models:
## lm_mixed_null: Hearing ~ (1 | SubjectID)
## lm_mixed: Hearing ~ 1 + ListID + (1 | SubjectID)
##               npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
## lm_mixed_null    3 674.22 681.91 -334.11   668.22
## lm_mixed         6 657.70 673.09 -322.85   645.70 22.52  3  5.083e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

1.D)

« comments »

Both models indicate that there are significant differences between the mean hearing score of word lists, therefore background noise changes the difficulty level of these test!

Problem 2

```
# read in the data

termites <- read.table(file = "/Users/alimos313/Documents/studies/phd/university/courses/stat-modelling/
```

2.A)

```
# explore the data
skim(termites)
```

Table 4: Data summary

Name	termites
Number of rows	16
Number of columns	17
_____	
Column type frequency:	
logical	2
numeric	15
_____	
Group variables	None

Variable type: logical

skim_variable	n_missing	complete_rate	mean	count
day3	16	0	NaN	:
day9	16	0	NaN	:

Variable type: numeric

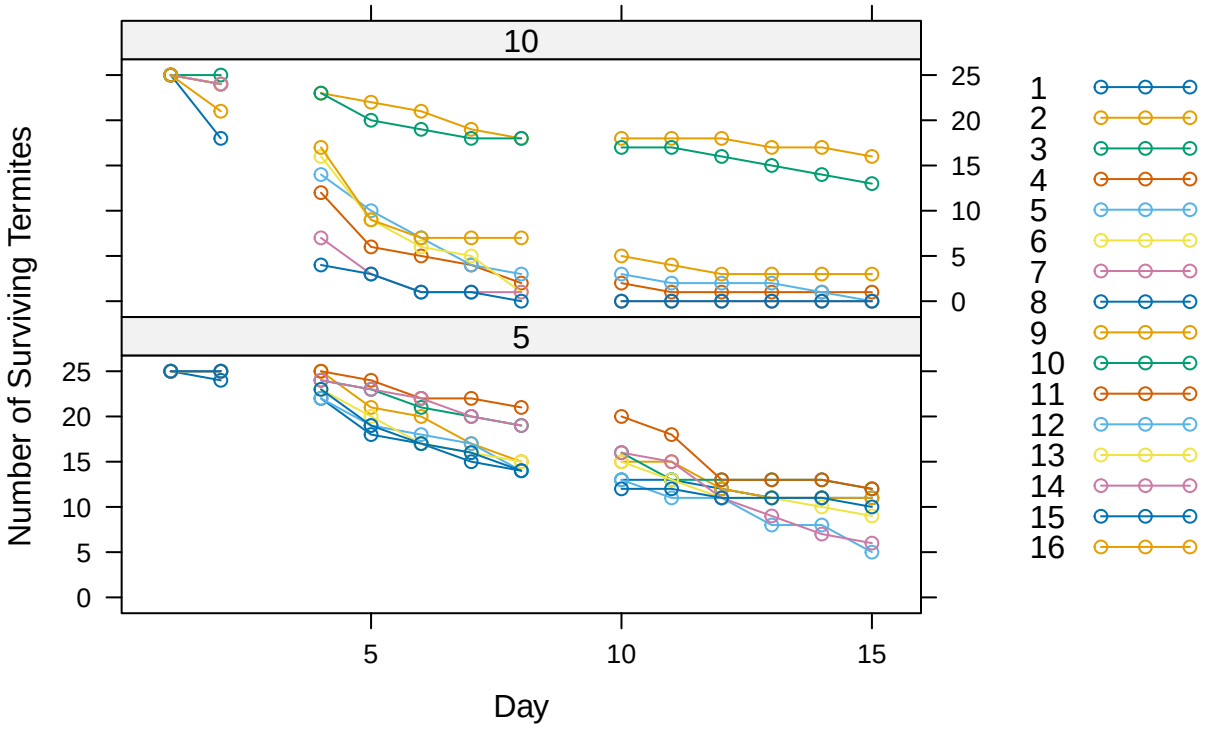
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
dish	0	1	8.50	4.76	1	4.75	8.5	12.25	16	
dose	0	1	7.50	2.58	5	5.00	7.5	10.00	10	
day1	0	1	25.00	0.00	25	25.00	25.0	25.00	25	
day2	0	1	23.94	1.88	18	24.00	24.5	25.00	25	
day4	0	1	19.00	6.63	4	15.50	22.5	23.25	25	
day5	0	1	15.56	7.51	3	9.00	19.0	21.25	24	
day6	0	1	13.81	7.79	1	6.75	17.0	20.25	22	
day7	0	1	12.62	7.49	1	4.75	16.0	18.25	22	
day8	0	1	11.31	7.60	0	2.75	14.0	18.00	21	
day10	0	1	10.31	7.27	0	2.75	13.0	16.00	20	
day11	0	1	9.50	7.00	0	1.75	12.5	15.00	18	
day12	0	1	8.38	6.23	0	1.75	11.0	12.25	18	
day13	0	1	7.81	5.88	0	1.75	10.0	11.50	17	
day14	0	1	7.50	5.83	0	1.00	9.0	11.50	17	
day15	0	1	6.81	5.56	0	0.75	7.5	11.25	16	

2.B)

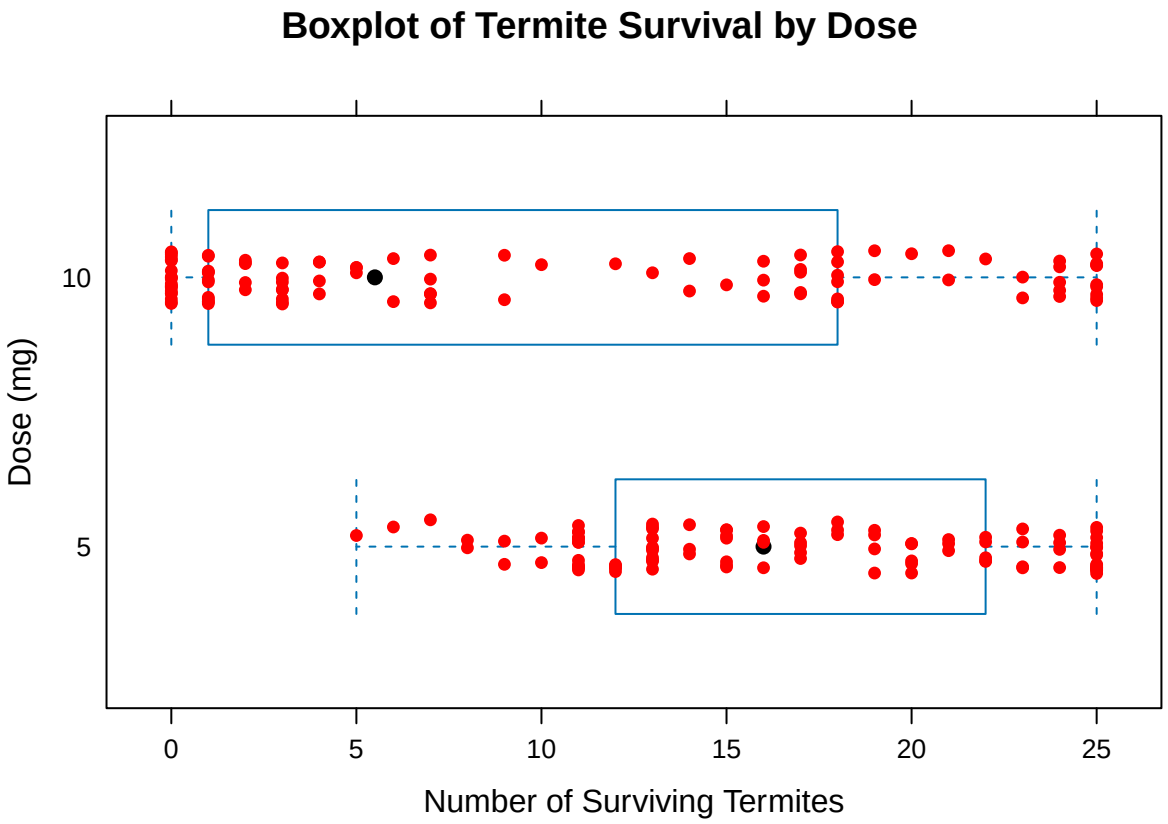
```
# get the data into long format using tidyr::pivot_longer

termites %<>% pivot_longer(cols = starts_with("day"), names_to = "day", values_to = "survival") %>%
        mutate(day = as.integer(sapply(str_split(day, pattern = "day"), "[", 2)),
        dish = as.factor(dish),
        )

head(termites)
```

```
## # A tibble: 6 x 4
##   dish   dose   day survival
##   <fct> <int> <int>    <int>
## 1 1         5     1       25
## 2 1         5     2       24
## 3 1         5     3       NA
## 4 1         5     4       22
## 5 1         5     5       18
## 6 1         5     6       17
```

```
# xyplot to show termite survival over time by dose

xyplot(survival ~ day | factor(dose),
       data = termites,
       groups = factor(dish),
       type = c("o"), # Points and regression lines
       auto.key = TRUE,    # Automatically create a legend
       main = "Survival of Termites Over Days by Dose",
       xlab = "Day",
       ylab = "Number of Surviving Termites",
       layout = c(1, 2))
```




```
# Boxplot of survival grouped by dose
bwplot(factor(dose) ~ survival,
       data = termites,
       main = "Boxplot of Termite Survival by Dose",
       xlab = "Number of Surviving Termites",
       ylab = "Dose (mg)",
       panel = function(x, y) {
           panel.bwplot(x, y)  # Default box plot
           panel.stripplot(x, y, jitter.data = TRUE, col = "red", pch = 16)  # Add points with jitter
       })
```



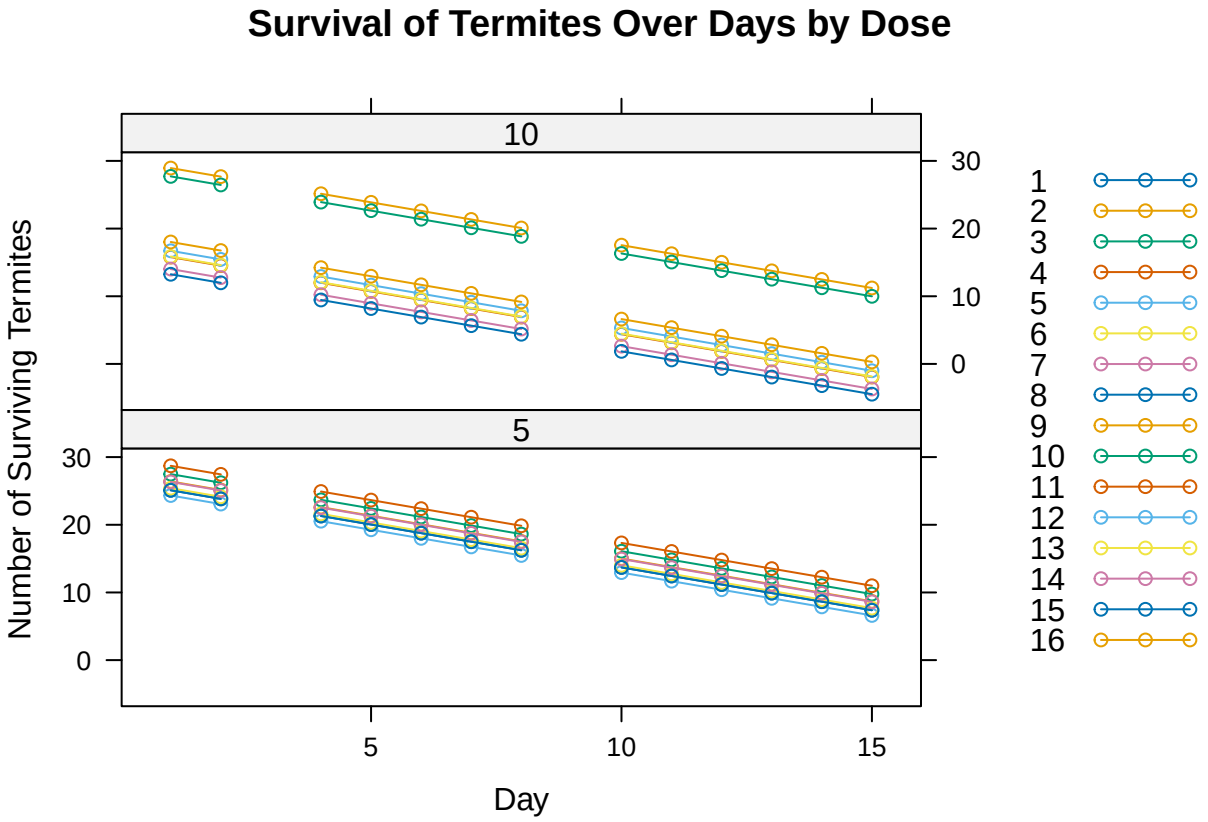
2.C)

```
# make the linear model
lm_model <- lm(survival ~ dish + dose + day, data = termites)

summary(lm_model)
```

```
##
## Call:
## lm(formula = survival ~ dish + dose + day, data = termites)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -6.691 -1.964 -0.096  1.347 11.749
##
## Coefficients: (1 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.636e+01  1.015e+00  25.961  < 2e-16 ***
## dish2        1.308e+00  1.299e+00   1.007  0.31542
## dish3        2.385e+00  1.299e+00   1.835  0.06799 .
## dish4        3.615e+00  1.299e+00   2.783  0.00593 **
## dish5       -7.692e-01  1.299e+00  -0.592  0.55449
## dish6        3.077e-01  1.299e+00   0.237  0.81304
## dish7        1.231e+00  1.299e+00   0.947  0.34466
## dish8       -4.804e-15  1.299e+00   0.000  1.00000
## dish9        3.846e+00  1.299e+00   2.960  0.00346 **
## dish10       2.615e+00  1.299e+00   2.013  0.04551 *
```

```
## dish11      -9.308e+00  1.299e+00  -7.164 1.65e-11 ***
## dish12      -8.385e+00  1.299e+00  -6.454 8.77e-10 ***
## dish13      -9.231e+00  1.299e+00  -7.105 2.32e-11 ***
## dish14      -1.108e+01  1.299e+00  -8.526 4.55e-15 ***
## dish15      -1.185e+01  1.299e+00  -9.118  < 2e-16 ***
## dish16      -7.077e+00  1.299e+00  -5.447 1.56e-07 ***
## dose               NA         NA      NA       NA
## day         -1.266e+00  5.208e-02 -24.305  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.312 on 191 degrees of freedom
##   (32 observations deleted due to missingness)
## Multiple R-squared:  0.8595, Adjusted R-squared:  0.8477
## F-statistic: 73.01 on 16 and 191 DF,  p-value: < 2.2e-16
```

```r
# plot the predicted values

## Obtain predictions
termites$predicted <- ""
termites$predicted[is.na(termites$survival)] <- NA

termites$predicted[!is.na(termites$survival)] <- as.numeric(predict(lm_model))

termites %<>% mutate(predicted = as.numeric(predicted))


xyplot(predicted ~ day | factor(dose),
       data = termites,
       groups = factor(dish),
       type = c("o"), # Points and regression lines
       auto.key = TRUE,    # Automatically create a legend
       main = "Survival of Termites Over Days by Dose",
       xlab = "Day",
       ylab = "Number of Surviving Termites",
       layout = c(1, 2))
```



« comments »

Too many parameters for the variable dish!

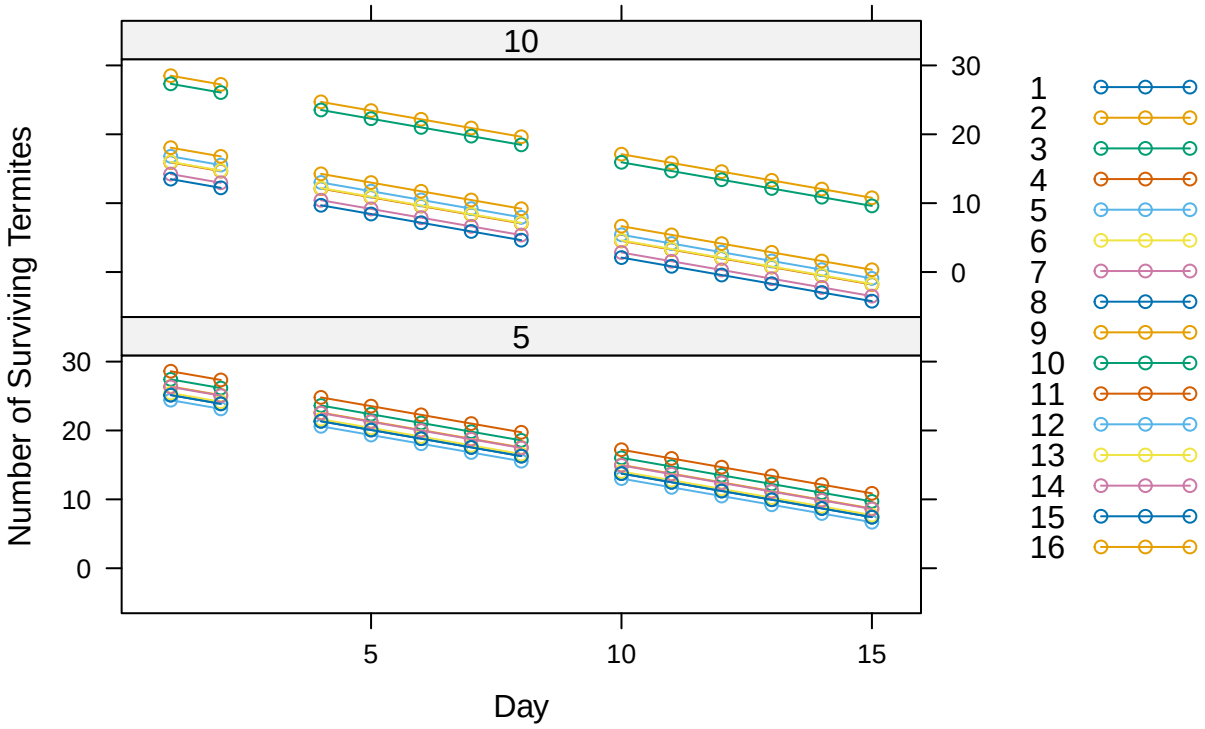
2.D)

```
# make the linear model
lm_model_mixed1 <- lmer(survival ~ dose + day + (1|dish), data = termites)

summary(lm_model_mixed1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: survival ~ dose + day + (1 | dish)
##    Data: termites
##
## REML criterion at convergence: 1137.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0822 -0.5606 -0.0373  0.4065  3.4749
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  dish     (Intercept) 18.66    4.320
##  Residual             10.97    3.312
## Number of obs: 208, groups:  dish, 16
##
## Fixed effects:
```

```
##             Estimate Std. Error t value
## (Intercept) 34.68963    3.51844   9.859
## dose        -1.46346    0.44167  -3.313
## day         -1.26588    0.05208 -24.305
##
## Correlation of Fixed Effects:
##      (Intr) dose
## dose -0.941
## day  -0.123  0.000
```

```r
# plot the predicted values

## Obtain predictions
termites$predicted_mixed1 <- ""
termites$predicted_mixed1[is.na(termites$survival)] <- NA

termites$predicted_mixed1[!is.na(termites$survival)] <- as.numeric(predict(lm_model_mixed1))

termites %<>% mutate(predicted_mixed1 = as.numeric(predicted_mixed1))


xyplot(predicted_mixed1 ~ day | factor(dose),
       data = termites,
       groups = factor(dish),
       type = c("o"), # Points and regression lines
       auto.key = TRUE,    # Automatically create a legend
       main = "Survival of Termites Over Days by Dose",
       xlab = "Day",
       ylab = "Number of Surviving Termites",
       layout = c(1, 2))
```



2.E)

```
# make the linear model
lm_model_mixed2 <- lmer(survival ~ dose + day + (day|dish), data = termites)
```

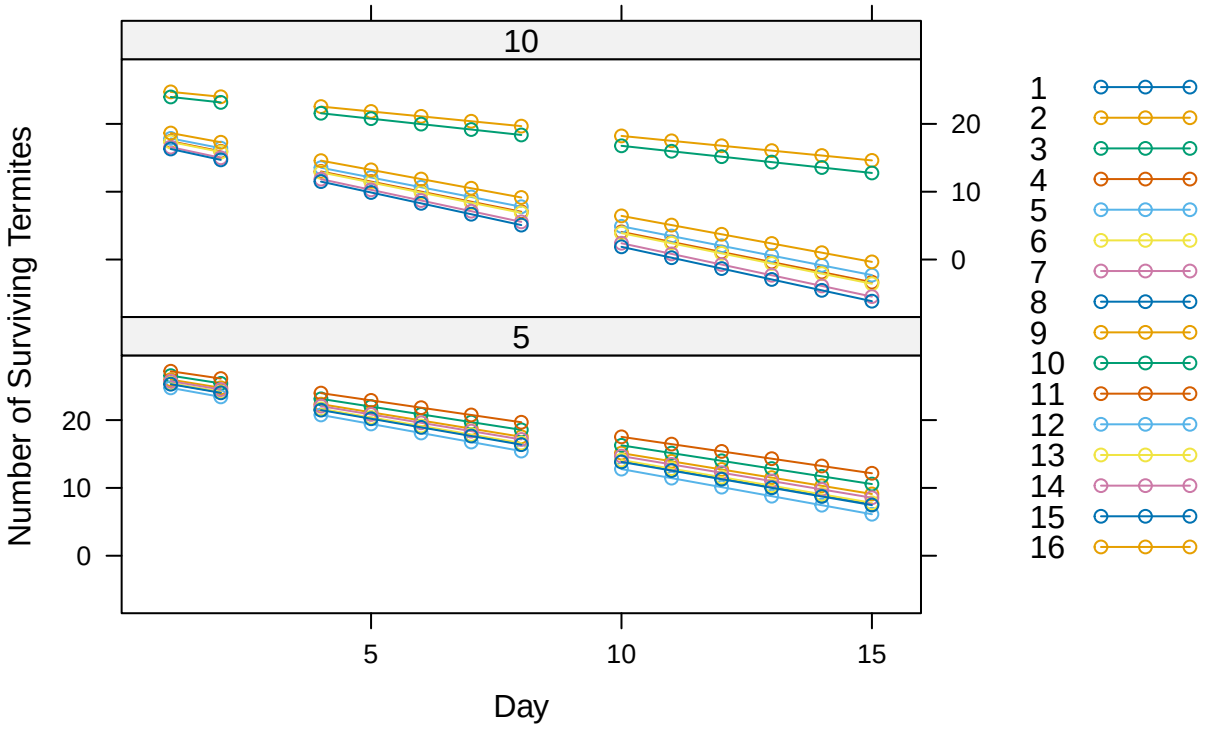
```
## boundary (singular) fit: see help('isSingular')
```

```
summary(lm_model_mixed2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: survival ~ dose + day + (day | dish)
##    Data: termites
##
## REML criterion at convergence: 1115.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.47712 -0.51472  0.04884  0.42997  2.89506
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  dish     (Intercept) 4.77749  2.1857
##           day         0.06522  0.2554   1.00
##  Residual             9.66636  3.1091
## Number of obs: 208, groups:  dish, 16
##
## Fixed effects:
##             Estimate Std. Error t value
```

```
## (Intercept) 32.49043    2.18765  14.852
## dose        -1.17024    0.27571  -4.244
## day         -1.26588    0.08041 -15.742
##
## Correlation of Fixed Effects:
##      (Intr) dose
## dose -0.945
## day   0.085  0.000
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```r
# plot the predicted values

## Obtain predictions
termites$predicted_mixed2 <- ""
termites$predicted_mixed2[is.na(termites$survival)] <- NA

termites$predicted_mixed2[!is.na(termites$survival)] <- as.numeric(predict(lm_model_mixed2))

termites %<>% mutate(predicted_mixed2 = as.numeric(predicted_mixed2))


xyplot(predicted_mixed2 ~ day | factor(dose),
       data = termites,
       groups = factor(dish),
       type = c("o"), # Points and regression lines
       auto.key = TRUE,    # Automatically create a legend
       main = "Survival of Termites Over Days by Dose",
       xlab = "Day",
       ylab = "Number of Surviving Termites",
       layout = c(1, 2))
```



2.F)

```
boostrap_ci <- function(data, formula, parameter, N = 1000, conf = 0.90) {
    estimates <- numeric(N)
    for (i in 1:N) {
        resample <- data[sample(nrow(data), replace = TRUE), ]
        model <- lmer(formula, data = resample)
        estimates[i] <- fixef(model)[[parameter]]
    }
    return(quantile(estimates, c((1 - conf) / 2, 1 - (1 - conf) / 2)))
}



ci <- boostrap_ci(termites, survival ~ dose + day + (1 | dish) + day, "dose")
ci
```

```
##        5%       95%
## -1.614166 -1.317881
```

« comments »

The 90% confidence interval excludes 0, therefore we can say that dose significantly impact survival!