Exercises

Paul Schmidt

2023-11-14

Table of contents

1 Ex: 1wCRD

```
# This data holds information about the misture content from multiple samples
# for each of four different soils. Notice that for this dataset you have no
# information on any specific field trial layout (i.e. row or col column are
# not present in the dataset). Therefore you should skip trying to create a
# field layout with desplot() and instead focus on the following:
# 1) How many samples per soil were taken?
# 2) Which soil has the highest value for moisture?
# 3) Which soil has the highest average value for moisture?
# 4) Create a ggplot with moisture values per soil!
# 5) Conduct an ANOVA
# 6) Perform multiple mean comparisons using the LSD test/t-test.
# Bonus:
# 7) Repeat the analysis, but this time remove all moisture values larger than
# 12 at the very beginning.

# get data
library(tidyverse)
```

```r
path <- "https://raw.githubusercontent.com/SchmidtPaul/dsfair_quarto/master/data/Mead1993b
dat_1wCRD <- read_csv(path, col_types = "fn")
```

```r
library(emmeans)
library(multcomp)
library(multcompView)

dat_1wCRD %>%
  group_by(soil) %>%
  summarise(
    n = n(),
    max = max(moisture),
    mean = mean(moisture)
  )

plot_1wCRD <- ggplot(data = dat_1wCRD) +
  aes(y = moisture, x = soil) +
  geom_point() +
  theme_bw() +
  scale_y_continuous(
    limits = c(0, NA),
    expand = expansion(mult = c(0, 0.05))
  )

mod_1wCRD <- lm(moisture ~ soil, dat_1wCRD)

anova(mod_1wCRD)

mod_1wCRD %>%
  emmeans(specs = ~ soil) %>%
  cld(adjust = "none", Letters = letters)
```

2 Ex: 1wRCBD

```r
# This data holds information from a yield (t/ha) trial laid out as a
# randomized complete block design (3 blocks) with cultivar (4 cultivars)
# being the only treatment factor. Thus, we have a total of 12 plots.
# 1) Create a desplot!
# 2) Which cultivar has the highest average yield?
```

```r
# 3) Which block has the highest average yield?
# 4) Create a ggplot with yield per cultivar.
# 5) Conduct an ANOVA
# 6) Perform multiple mean comparisons using the t-test.
# Bonus:
# 7) Repeat the analysis, but this time remove all observations from block 1.

# get data
library(tidyverse)
path <- "https://raw.githubusercontent.com/SchmidtPaul/dsfair_quarto/master/data/Clewer&Sc
dat_1wRCBD <- read_csv(path, col_types = "ffnii")
```

```r
library(emmeans)
library(desplot)
library(multcomp)
library(multcompView)

mydp <- desplot(
  data = dat_1wRCBD,
  flip = TRUE,
  form = cultivar ~ col + row,
  text = cultivar,
  out1 = block,
  cex = 1,
  show.key = FALSE
)

dat_1wRCBD %>%
  group_by(cultivar) %>%
  summarise(mean = mean(yield))

dat_1wRCBD %>%
  group_by(block) %>%
  summarise(mean = mean(yield))

plot_1wRCBD <- ggplot(data = dat_1wRCBD) +
  aes(y = yield, x = cultivar, color = block) +
  geom_point() +
  theme_bw() +
  scale_y_continuous(
    limits = c(0, NA),
```

```
    expand = expansion(mult = c(0, 0.05))
  )

mod_1wRCBD <- lm(yield ~ cultivar + block, dat_1wRCBD)

anova(mod_1wRCBD)

mod_1wRCBD %>%
  emmeans(specs = ~ cultivar) %>%
  cld(adjust = "none", Letters = letters)
```

3 Ex: 1wAugLat

```
# This dataset contains information from an experiment utilizing a resolvable
# design with checks. Specifically, it features 2 replicates and includes 90
# unique entries. Additionally, there are 6 checks with extra replication.
# Each block in the design contains 10 entries, and these incomplete blocks
# are organized according to a lattice design, which allows them to be grouped
# into complete replicates, forming a resolvable design structure. In this
# dataset, standards are coded from 1001 to 1006, while the entries are coded
# from 2 to 100.
# 1) Create a desplot!
# 2) What is the the number of missing values, the number of non-missing
# values and the average value for yield for each combination of rep and
# genoCheck?
# 3) Create a ggplot with geno on the x-axis, yield on the y-axis and
# dots colored depending on whether a geno is a check or not!
# 4) Compare the average s.e.d. to find out whether the effects for
# incomplete blocks should be taken as fixed or random in the model!
# 5) Conduct an ANOVA
# 6) Perform multiple mean comparisons using the t-test.

# get data
library(tidyverse)
path <- "https://raw.githubusercontent.com/SchmidtPaul/dsfair_quarto/master/data/PiephoAug
dat_1wAugLat <- read_csv(path, col_types = "ffffnnii")
```

```r
library(emmeans)
library(desplot)
library(lme4)
library(lmerTest)
library(multcomp)
library(multcompView)

mydp <- desplot(
  data = dat_1wAugLat,
  flip = TRUE,
  form = geno ~ col + row,
  text = geno,
  col = genoCheck,
  col.text = c("red", "black"),
  out1 = rep,
  out2 = block,
  show.key = FALSE
)

dat_1wAugLat %>%
  group_by(rep, genoCheck) %>%
  summarise(nobs = sum(!is.na(yield)),
            nNA  = sum(is.na(yield)),
            mean = mean(yield, na.rm = TRUE))

geno_order <- dat_1wAugLat %>%
  group_by(geno) %>%
  summarise(mean = mean(yield, na.rm = TRUE)) %>%
  ungroup() %>%
  arrange(mean) %>%
  pull(geno) %>%
  as.character()

mygg <- ggplot(data = dat_1wAugLat) +
  aes(y = yield, x = geno, color = genoCheck) +
  geom_point() +
  theme_bw() +
  scale_y_continuous(
    limits = c(0, NA),
    expand = expansion(mult = c(0, 0.05))
  ) +
```

```
  scale_x_discrete(
    limits = geno_order
  )

mod_1wAugLat_fb <-   lm(yield ~ geno + rep + block,       data = dat_1wAugLat)
mod_1wAugLat_rb <- lmer(yield ~ geno + rep + (1 | block), data = dat_1wAugLat)

mod_1wAugLat_fb %>%
  emmeans(pairwise ~ "geno",
          adjust = "none") %>%
  pluck("contrasts") %>% # extract diffs
  as_tibble() %>% # format to table
  pull("SE") %>% # extract s.e.d. column
  mean() # get arithmetic mean

# this takes some time!
mod_1wAugLat_rb %>%
  emmeans(pairwise ~ "geno",
          adjust = "none") %>%
  pluck("contrasts") %>% # extract diffs
  as_tibble() %>% # format to table
  pull("SE") %>% # extract s.e.d. column
  mean() # get arithmetic mean

anova(mod_1wAugLat_rb)

mod_1wAugLat_rb %>%
  emmeans(specs = ~ geno) %>%
  cld(adjust = "none", Letters = letters)
```

4 Ex: 1wRowCol

```
# This data contains information from an experiment involving 64 oat
# genotypes. The experimental design is an 8 x 8 lattice, which has been
# replicated three times.
# 1) Create a desplot!
# 2) Per replicate, what is the the number of missing values, the number of
# non-missing values and the average value for yield, height and TKW,
# respectively?
```

```r
# 3) Create a ggplot with treat on the x-axis, yield on the y-axis!
# 4) Compare the average s.e.d. to find out whether the effects for
# incomplete blocks should be taken as fixed or random in the model!
# 5) Conduct an ANOVA
# 6) Perform multiple mean comparisons using the t-test.

# get data
library(tidyverse)
path <- "https://raw.githubusercontent.com/SchmidtPaul/dsfair_quarto/master/data/RowColFro
dat_1wRowCol <- read_csv(path, col_types = "fiiffnnn") %>%
  mutate(rowF = as.factor(row), colF = as.factor(col))
```

```r
library(emmeans)
library(lme4)
library(lmerTest)
library(multcomp)
library(multcompView)

mydp <- desplot(
  data = dat_1wRowCol,
  flip = TRUE,
  form = treat ~ col + row,
  text = treat,
  out1 = rep,
  out2 = inc_block,
  show.key = FALSE
)

dat_1wRowCol %>%
  group_by(rep) %>%
  dlookr::describe(yield, height, TKW) %>%
  dplyr::select(1:sd)

treat_order <- dat_1wRowCol %>%
  group_by(treat) %>%
  summarise(mean = mean(yield, na.rm = TRUE)) %>%
  ungroup() %>%
  arrange(mean) %>%
  pull(treat) %>%
  as.character()
```

```r
mygg <- ggplot(data = dat_1wRowCol) +
  aes(y = yield, x = treat) +
  geom_point() +
  theme_bw() +
  scale_y_continuous(
    limits = c(0, NA),
    expand = expansion(mult = c(0, 0.05))
  ) +
  scale_x_discrete(
    limits = treat_order
  )

mod_1wRowCol_fb <-   lm(yield ~ treat + rep + inc_block,       data = dat_1wRowCol)
mod_1wRowCol_rb <- lmer(yield ~ treat + rep + (1 | inc_block), data = dat_1wRowCol)

mod_1wRowCol_fb %>%
  emmeans(pairwise ~ "treat",
          adjust = "none") %>%
  pluck("contrasts") %>% # extract diffs
  as_tibble() %>% # format to table
  pull("SE") %>% # extract s.e.d. column
  mean() # get arithmetic mean

# this takes some time!
mod_1wRowCol_rb %>%
  emmeans(pairwise ~ "treat",
          adjust = "none") %>%
  pluck("contrasts") %>% # extract diffs
  as_tibble() %>% # format to table
  pull("SE") %>% # extract s.e.d. column
  mean() # get arithmetic mean

anova(mod_1wRowCol_rb)

mod_1wRowCol_rb %>%
  emmeans(specs = ~ treat) %>%
  cld(adjust = "none", Letters = letters)
```