Midterm1

Matthew Stoebe

2025-04-21

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.0.4
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(car)
```

```
## Loading required package: carData
##
## Attaching package: 'car'
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
## The following object is masked from 'package:purrr':
##
##     some
```

```
library(MVN)
library(Hotelling)
```

```
## Loading required package: corpcor
##
## Attaching package: 'Hotelling'
##
## The following object is masked from 'package:dplyr':
##
##     summarise
```

```r
set.seed(123)

load("Data/plasticDat.RData")


dat <- as_tibble(dat)

head(dat)
```

```
## # A tibble: 6 x 6
##   setting    V1    V2 tearResist gloss opacity
##     <dbl> <int> <int>      <dbl> <dbl>   <dbl>
## 1       0     0     0        6.5   9.5     4.4
## 2       0     0     0        6.2   9.9     6.4
## 3       0     0     0        5.8   9.6     3
## 4       0     0     0        6.5   9.6     4.1
## 5       0     0     0        6.5   9.2     0.8
## 6       1     0     1        6.9   9.1     5.7
```

a

```r
load("Data/plasticDat.RData")
names(dat)
```

```
## [1] "setting"    "V1"         "V2"         "tearResist" "gloss"
## [6] "opacity"
```

```r
str(dat)
```

```
## 'data.frame':    20 obs. of  6 variables:
##  $ setting   : num  0 0 0 0 0 1 1 1 1 1 ...
##  $ V1        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ V2        : int  0 0 0 0 0 1 1 1 1 1 ...
##  $ tearResist: num  6.5 6.2 5.8 6.5 6.5 6.9 7.2 6.9 6.1 6.3 ...
##  $ gloss     : num  9.5 9.9 9.6 9.6 9.2 9.1 10 9.9 9.5 9.4 ...
##  $ opacity   : num  4.4 6.4 3 4.1 0.8 5.7 2 3.9 1.9 5.7 ...
```

```r
head(dat)
```

```
##   setting V1 V2 tearResist gloss opacity
## 1       0  0  0        6.5   9.5     4.4
## 2       0  0  0        6.2   9.9     6.4
## 3       0  0  0        5.8   9.6     3.0
## 4       0  0  0        6.5   9.6     4.1
## 5       0  0  0        6.5   9.2     0.8
## 6       1  0  1        6.9   9.1     5.7
```

```r
man <- manova(cbind(tearResist, gloss, opacity) ~ setting, data=dat)
man_sum <- summary(man, test="Wilks")
print(man_sum)
```

```
##           Df   Wilks approx F num Df den Df    Pr(>F)
## setting    1 0.34858   9.9666      3     16 0.0006044 ***
## Residuals 18
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
pval <- man_sum$stats["setting","Pr(>F)"]
cat("MANOVA p-value =", pval, "\n")
```

```
## MANOVA p-value = 0.0006044414
```

b

```r
library(biotools)
```

```
## Loading required package: MASS
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
## ---
## biotools version 4.3
```

```r
print(boxM(dat[,c("tearResist","gloss","opacity")], dat$setting))
```

```
##
##  Box's M-test for Homogeneity of Covariance Matrices
##
## data:  dat[, c("tearResist", "gloss", "opacity")]
## Chi-Sq (approx.) = 24.015, df = 18, p-value = 0.1546
```

```r
res <- residuals(man)
sw  <- sapply(as.data.frame(res), function(x) shapiro.test(x)$p.value)
print(sw)
```

```
## tearResist      gloss    opacity
##  0.2913991  0.5008235  0.4656299
```

c, d

```
settings <- sort(unique(dat$setting))
pairs    <- combn(settings, 2, simplify = FALSE)

raw_p <- sapply(pairs, function(p) {
  mv  <- manova(cbind(tearResist, gloss, opacity) ~ factor(setting),
                data = subset(dat, setting %in% p))
  sum <- summary(mv, test = "Wilks")
  sum$stats[1, "Pr(>F)"]
})

adj_p <- p.adjust(raw_p, method = "bonferroni")

pairwise_df <- data.frame(
  Comparison   = sapply(pairs, function(p) paste(p, collapse = " vs ")),
  raw_p_value  = raw_p,
  adj_p_value  = adj_p
)

pairwise_df
```

```
##   Comparison raw_p_value adj_p_value
## 1     0 vs 1 0.636585195  1.00000000
## 2     0 vs 2 0.067731972  0.40639183
## 3     0 vs 3 0.005562226  0.03337336
## 4     1 vs 2 0.093991698  0.56395019
## 5     1 vs 3 0.118179705  0.70907823
## 6     2 vs 3 0.014552976  0.08731785
```

e

```
sub03 <- subset(dat, setting %in% c("0","3"))
lapply(sub03[c("tearResist","gloss","opacity")], function(x) {
  t.test(x ~ setting, data = sub03)
})
```

```
## $tearResist
##
##  Welch Two Sample t-test
##
## data:  x by setting
## t = -5.4444, df = 7.7679, p-value = 0.0006773
## alternative hypothesis: true difference in means between group 0 and group 3 is not equal to 0
## 95 percent confidence interval:
##  -1.3972491 -0.5627509
## sample estimates:
## mean in group 0 mean in group 3
##            6.30            7.28
```

```
##
##
## $gloss
##
##  Welch Two Sample t-test
##
## data:  x by setting
## t = 0.63444, df = 5.8628, p-value = 0.5497
## alternative hypothesis: true difference in means between group 0 and group 3 is not equal to 0
## 95 percent confidence interval:
##  -0.4606059  0.7806059
## sample estimates:
## mean in group 0 mean in group 3
##            9.56            9.40
##
##
## $opacity
##
##  Welch Two Sample t-test
##
## data:  x by setting
## t = -0.83551, df = 7.4075, p-value = 0.4296
## alternative hypothesis: true difference in means between group 0 and group 3 is not equal to 0
## 95 percent confidence interval:
##  -4.86258  2.30258
## sample estimates:
## mean in group 0 mean in group 3
##            3.74            5.02
```

```
print(summary.aov(man))
```

```
##  Response tearResist :
##             Df Sum Sq Mean Sq F value    Pr(>F)
## setting      1 2.4649 2.46490  24.641 0.0001004 ***
## Residuals   18 1.8006 0.10003
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##  Response gloss :
##             Df Sum Sq Mean Sq F value Pr(>F)
## setting      1 0.4489 0.44890  1.7427 0.2033
## Residuals   18 4.6366 0.25759
##
##  Response opacity :
##             Df Sum Sq Mean Sq F value Pr(>F)
## setting      1  2.465  2.4649  0.6185 0.4419
## Residuals   18 71.741  3.9856
```

```
tw <- manova(cbind(tearResist, gloss, opacity) ~ V1 * V2, data=dat)
print(summary(tw, test="Wilks"))
```

```
##           Df   Wilks approx F num Df den Df   Pr(>F)
## V1         1 0.38186   7.5543      3     14 0.003034 **
```

```
## V2         1 0.52303   4.2556      3     14 0.024745 *
## V1:V2      1 0.77711   1.3385      3     14 0.301782
## Residuals 16
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```