Discussion5

Jing Lyu

2/8/2023

Two-way ANOVA

In this section, we'll use the built-in R data set 'ToothGrowth'. It includes information from a study on the effects of vitamin C on tooth growth in Guinea pigs.

The trial used 60 pigs who were given one of three vitamin C dose levels (0.5, 1, or 2 mg/day) via one of two administration routes: orange juice (OJ) or ascorbic acid (VC).

```
library(dplyr)
dat = ToothGrowth
str(dat)
```

Visualization

```
## 'data.frame':    60 obs. of  3 variables:
##  $ len : num  4.2 11.5 7.3 5.8 6.4 10 11.2 11.2 5.2 7 ...
##  $ supp: Factor w/ 2 levels "OJ","VC": 2 2 2 2 2 2 2 2 2 2 ...
##  $ dose: num  0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 0.5 ...
```

```
table(dat$dose)
```

```
##
## 0.5   1   2
##  20  20  20
```

R treats 'dose' as a numeric variable based on the output. We'll transform it to a factor variable.

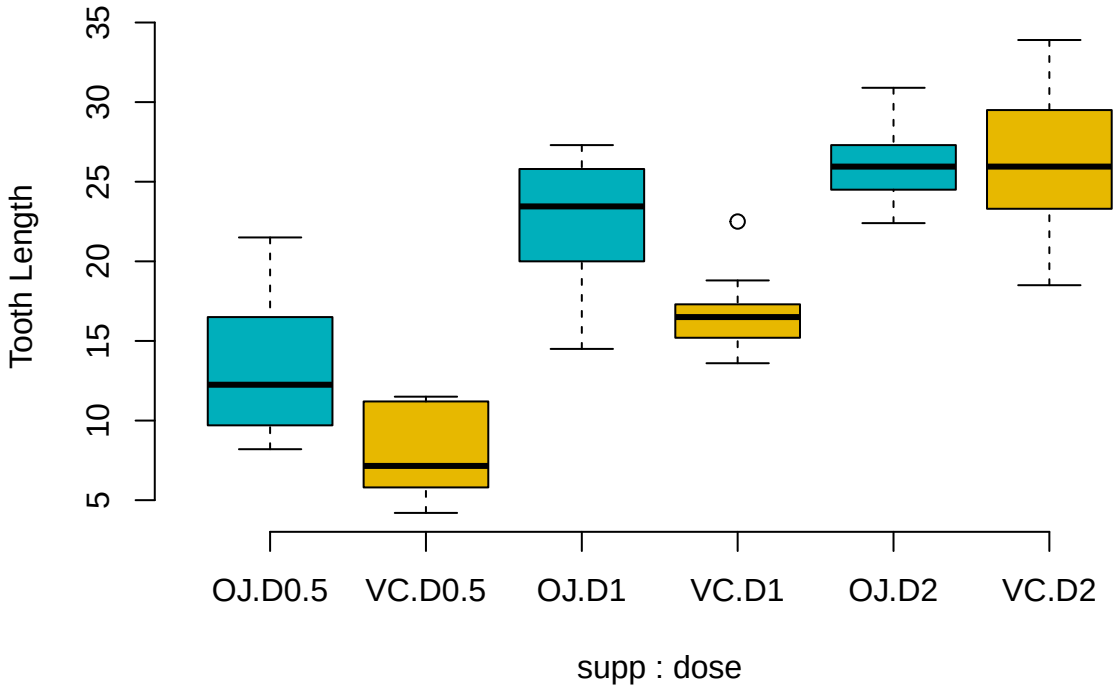
```
dat$dose <- factor(dat$dose,
                   levels = c(0.5, 1, 2),
                   labels = c("D0.5", "D1", "D2"))
table(dat$supp,dat$dose)
```

```
##
##      D0.5 D1 D2
##   OJ   10 10 10
##   VC   10 10 10
```

We have a well-balanced design.

To visualize the data grouped by the levels of the two factors, we can use a box plot.

```
boxplot(len ~ supp * dose, data=dat, frame = FALSE,
        col = c("#00AFBB", "#E7B800"), ylab="Tooth Length")
```




```
model1 = aov(len ~ supp + dose, data = dat)
summary(model1)
```

Modeling

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## supp         1  205.4   205.4   14.02 0.000429 ***
## dose         2 2426.4  1213.2   82.81  < 2e-16 ***
## Residuals   56  820.4    14.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# If you think these two variables have interactive effect:
model2 <- aov(len ~ supp * dose, data = dat)
summary(model2)
```

```
##             Df Sum Sq Mean Sq F value   Pr(>F)
## supp         1  205.4   205.4  15.572 0.000231 ***
## dose         2 2426.4  1213.2  92.000  < 2e-16 ***
## supp:dose    2  108.3    54.2   4.107 0.021860 *
## Residuals   54  712.1    13.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Practice Project

Import dataset

- Harvard dataverse

```
#install.packages("haven")
#install.packages("tzdb")
library(haven) # readin sav file
```

```r
star = read_sav("STAR_Students.sav")%>%as.data.frame()
```

```
## Failed to find G1READ_A
## Failed to find G1MATH_A
## Failed to find G2READ_A
## Failed to find G2MATH_A
## Failed to find G3READ_A
## Failed to find G3MATH_A
```

```r
length(names(star)) # 379 variables
```

```
## [1] 379
```

```r
# check missing values
star.dat = star%>%dplyr::select(g1tmathss, g1classtype, g1schid, g1tchid)%>%mutate(na.count=rowSums(is.
table(star.dat$na.count)
```

```
##
##    0    1    4
## 6598  231 4772
```

```r
star.temp1 = star.dat%>%filter(na.count==1)
head(star.temp1) # only missing math scores
```

```
##   g1tmathss g1classtype g1schid  g1tchid na.count
## 1        NA           1  257905 25790508        1
## 2        NA           1  244708 24470807        1
## 3        NA           2  193422 19342206        1
## 4        NA           2  244727 24472708        1
## 5        NA           3  244774 24477410        1
## 6        NA           1  244697 24469706        1
```

```r
star.dat = star.dat%>%na.omit()%>%dplyr::select(-na.count) # drop rows with missing data
head(star.dat)
```

```
##    g1tmathss g1classtype g1schid  g1tchid
## 1        578           3  170295 17029507
## 4        507           2  257899 25789906
## 9        526           2  244697 24469708
## 11       505           3  244697 24469709
## 12       463           3  244697 24469709
## 16       542           3  205492 20549205
```

```r
str(star.dat)
```

```
## 'data.frame':    6598 obs. of  4 variables:
##  $ g1tmathss  : num  578 507 526 505 463 542 444 484 505 515 ...
##  $ g1classtype: dbl+lbl [1:6598] 3, 2, 2, 3, 3, 3, 1, 1, 2, 3, 2, 2, 2, 1, 3, 3, 1, 2,...
##    ..@ label      : chr "CLASSROOM TYPE GRADE 1"
##    ..@ format.spss: chr "F1.0"
##    ..@ labels     : Named num  1 2 3
##    .. ..- attr(*, "names")= chr [1:3] "SMALL CLASS" "REGULAR CLASS" "REGULAR + AIDE CLASS"
##  $ g1schid    : num  170295 257899 244697 244697 244697 ...
##  $ g1tchid    : num  17029507 25789906 24469708 24469709 24469709 ...
##  - attr(*, "na.action")= 'omit' Named int [1:5003] 2 3 5 6 7 8 10 13 14 15 ...
##   ..- attr(*, "names")= chr [1:5003] "2" "3" "5" "6" ...
```

```
star.dat$g1classtype = as.factor(star.dat$g1classtype)
star.dat$g1schid = as.factor(star.dat$g1schid)
star.dat$g1tchid = as.factor(star.dat$g1tchid)
# summarize mean scores by class types, teachers and schools
star.dat1 = star.dat%>%group_by(g1classtype,g1schid,g1tchid)%>%
  dplyr::summarise(math.mean=mean(g1tmathss))
head(star.dat1)
```

```
## # A tibble: 6 x 4
## # Groups:   g1classtype, g1schid [4]
##   g1classtype g1schid g1tchid  math.mean
##   <fct>       <fct>   <fct>        <dbl>
## 1 1           112038  11203805      500.
## 2 1           123056  12305606      534.
## 3 1           128076  12807604      555.
## 4 1           128076  12807606      544.
## 5 1           128079  12807905      522.
## 6 1           128079  12807907      515.
```

- AER package

```
#install.packages('AER')
library(AER)
library(dplyr)
data("STAR")
sapply(STAR,class)
```

```
##      gender   ethnicity       birth       stark       star1       star2
##    "factor"    "factor"   "yearqtr"    "factor"    "factor"    "factor"
##       star3       readk       read1       read2       read3       mathk
##    "factor"   "integer"   "integer"   "integer"   "integer"   "integer"
##       math1       math2       math3      lunchk      lunch1      lunch2
##   "integer"   "integer"   "integer"    "factor"    "factor"    "factor"
##      lunch3     schoolk     school1     school2     school3     degreek
##    "factor"    "factor"    "factor"    "factor"    "factor"    "factor"
##     degree1     degree2     degree3     ladderk     ladder1     ladder2
##    "factor"    "factor"    "factor"    "factor"    "factor"    "factor"
##     ladder3 experiencek experience1 experience2 experience3 tethnicityk
##    "factor"   "integer"   "integer"   "integer"   "integer"    "factor"
## tethnicity1 tethnicity2 tethnicity3     systemk     system1     system2
##    "factor"    "factor"    "factor"    "factor"    "factor"    "factor"
##     system3   schoolidk   schoolid1   schoolid2   schoolid3
##    "factor"    "factor"    "factor"    "factor"    "factor"
```

```
dim(STAR)
```

```
## [1] 11598    47
```

```
# only keep 1st grade
STAR.dat = STAR%>%dplyr::select(math1, school1, experience1, tethnicity1, schoolid1, star1)%>%
  na.omit() # remove rows with NA value in any column
str(STAR.dat)
```

```
## 'data.frame':    6558 obs. of  6 variables:
##  $ math1      : int  538 592 512 532 584 545 553 490 493 481 ...
##  $ school1    : Factor w/ 4 levels "inner-city","suburban",..: 3 2 3 3 3 3 3 1 4 2 ...
##  $ experience1: int  7 32 8 7 11 15 0 5 17 1 ...
```

```
##  $ tethnicity1: Factor w/ 2 levels "cauc","afam": 1 2 1 1 1 1 1 1 1 2 ...
##  $ schoolid1  : Factor w/ 80 levels "1","2","3","4",..: 63 20 5 50 69 79 5 16 48 51 ...
##  $ star1      : Factor w/ 3 levels "regular","small",..: 2 2 3 1 1 2 1 1 1 1 ...
##  - attr(*, "na.action")= 'omit' Named int [1:5040] 1 4 5 6 7 9 15 18 19 24 ...
##   ..- attr(*, "names")= chr [1:5040] "1122" "1160" "1183" "1195" ...
```