Exercise Solution for Chapter 6

Sherry WeMott

2020-05-12

Chapter 6, Exercise 6.4

We are instructed to make a less extreme example of correlated test statistics than the data duplication at the end of Section 6.5. Simulate data with true null hypotheses only, and let the data morph from having completely independent replicates (columns) to highly correlated as a function of some continuous-valued control parameter. Check type-I error control (e.g., with the p-value histogram) as a function of this control parameter.

For this exercise we use the PlantGrowth dataset from the `datasets` package in R. The dataset includes results from an experiment on plant growth comparing yeilds (as measured by dried weight of plants) obtained under a control and two different treatment conditions.

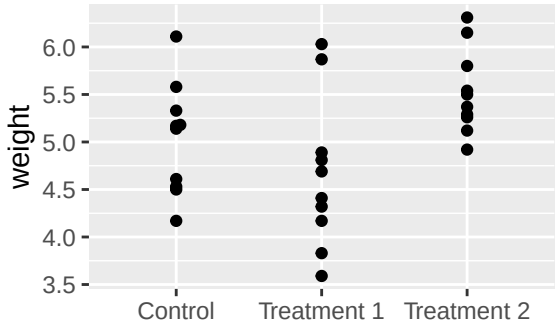
```
data("PlantGrowth")
PlantGrowth
```

```
##    weight group
## 1    4.17  ctrl
## 2    5.58  ctrl
## 3    5.18  ctrl
## 4    6.11  ctrl
## 5    4.50  ctrl
## 6    4.61  ctrl
## 7    5.17  ctrl
## 8    4.53  ctrl
## 9    5.33  ctrl
## 10   5.14  ctrl
## 11   4.81  trt1
## 12   4.17  trt1
## 13   4.41  trt1
## 14   3.59  trt1
## 15   5.87  trt1
## 16   3.83  trt1
## 17   6.03  trt1
## 18   4.89  trt1
## 19   4.32  trt1
## 20   4.69  trt1
## 21   6.31  trt2
## 22   5.12  trt2
## 23   5.54  trt2
## 24   5.50  trt2
## 25   5.37  trt2
## 26   5.29  trt2
## 27   4.92  trt2
## 28   6.15  trt2
```

```
## 29   5.80  trt2
## 30   5.26  trt2
```

Here we're plotting the outcomes of the three groups (ctrl, trt1, and trt2)

```
PlantGrowth %>%
   mutate(group = fct_recode(group,
                             Control = "ctrl",
                             `Treatment 1` = "trt1",
                             `Treatment 2` = "trt2")) %>%
  ggplot(aes(x = group, y = weight)) +
  geom_beeswarm() +
  labs(x = "")
```



Next we'll apply a t-test comparing weights in the ctrl group to the trt2 group:

```
PlantGrowth %>%
  filter(group %in% c("ctrl", "trt2")) %>%
    t.test(weight ~ group, data  = .)
```

```
##
##  Welch Two Sample t-test
##
## data:  weight by group
## t = -2.134, df = 16.786, p-value = 0.0479
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.98287213 -0.00512787
## sample estimates:
## mean in group ctrl mean in group trt2
##              5.032              5.526
```

Here's the tidy version:

```
library("broom")
PlantGrowth %>%
  filter(group %in% c("ctrl", "trt2")) %>%
  t.test(weight ~ group, data  = .) %>%
    tidy()
```

```
## # A tibble: 1 x 10
##   estimate estimate1 estimate2 statistic p.value parameter conf.low conf.high
##      <dbl>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl>
## 1   -0.494      5.03      5.53     -2.13  0.0479      16.8   -0.983  -0.00513
## # ... with 2 more variables: method <chr>, alternative <chr>
```

Then we'll duplicate data, adding a second copy of the dataframe, before running the t-test. Notice taht the

p-value is smaller even though the group means haven't changed. This is due to the increase in sample size.

```
PlantGrowth %>%
  bind_rows(PlantGrowth) %>% # Add the duplicate of the dataset
  filter(group %in% c("ctrl", "trt2")) %>%
  t.test(weight ~ group, data  = .) %>%
  tidy()
```

```
## # A tibble: 1 x 10
##   estimate estimate1 estimate2 statistic p.value parameter conf.low conf.high
##      <dbl>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl>
## 1   -0.494      5.03      5.53     -3.10 0.00377      35.4   -0.817    -0.171
## # ... with 2 more variables: method <chr>, alternative <chr>
```

Here we resample only half the data:

```
PlantGrowth %>%
  sample_frac(size = 0.5) %>%
  bind_rows(., .) %>%
  filter(group %in% c("ctrl", "trt2")) %>%
  t.test(weight ~ group, data  = .) %>%
  tidy()
```

```
## # A tibble: 1 x 10
##   estimate estimate1 estimate2 statistic p.value parameter conf.low conf.high
##      <dbl>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl>
## 1   -0.516      4.97      5.49     -2.43  0.0262      17.3   -0.964   -0.0688
## # ... with 2 more variables: method <chr>, alternative <chr>
```

Add random noise:

```
pg1 <- PlantGrowth %>%
  sample_frac(size = 0.5)
pg2 <- pg1 %>%
  mutate(noise = rnorm(15, mean = 0, sd = 0.2),
         weight = weight + noise) %>%
  select(-noise)

pg1 %>%
  bind_rows(pg2) %>%
  filter(group %in% c("ctrl", "trt2")) %>%
  t.test(weight ~ group, data  = .) %>%
  tidy()
```

```
## # A tibble: 1 x 10
##   estimate estimate1 estimate2 statistic p.value parameter conf.low conf.high
##      <dbl>     <dbl>     <dbl>     <dbl>   <dbl>     <dbl>    <dbl>     <dbl>
## 1   -0.530      4.77      5.30     -3.72 0.00196      15.5   -0.832    -0.227
## # ... with 2 more variables: method <chr>, alternative <chr>
```