Multiple Comparisons Example

D2K Course Staff

March 21 2019

Introduction

Normally, when doing a hypothesis test, we select a critical level α, typically 0.05, at which we reject the null hypothesis. However, if we are running a large batch, or a family, of hypothesis tests, we need to make adjustments to control the overall rate of false positives. If we were to use the typical $\alpha = 0.05$ critical lev for every single hypothesis test hat we were running, we would expect to make a Type I error once for every 20 tests we run! Similarly, the probablility that we don't make at least one type I error in 20 indepndent hypothesis tests at $\alpha = 0.05$ is about 0.6. Thus, we have to make adjustments for multiple comparisons in order to control the overall rate at which we will make a Type I error throughout all of our tests.Below, we show examples of different multiple comparisons adjustment procedures. These are generally viewed under the frameworks of the familywise error rate or the false discoery rate, explained below.

One important thing to note here is that all of these procedures rely on the researcher knowing exactly the family of tests he/she want to run before running all of them and applying the corrections. Essentailly, we can not sequentially run hypothesis tests chosen based on the results of previous tests and then just apply multiple correction corections at the end; peeking is cheating!

Data

For the examples below, we will be using the `msleep`data set from the `ggplot2` package. This data set contains information on the sleep times, physical attributes, and taxonomy of 83 different mammals. We will run hypothesis tests for pairwise comparisons of carnivores, herbivores, insectivores, and omnivores based on their average REM sleep and sleep cycle times. This will result in a total of 12 tests. (If you look at the code, a few numbers are changed for the sake of the example.) The results of these t tests are shown below in increasing order of p-value.

```
data(msleep)
head(msleep)
```

```
# A tibble: 6 x 11
  name    genus vore  order conservation sleep_total sleep_rem sleep_cycle
  <chr>   <chr> <chr> <chr> <chr>              <dbl>     <dbl>       <dbl>
1 Cheetah Acin~ carni Carn~ lc                  12.1      NA        NA
2 Owl mo~ Aotus omni  Prim~ <NA>                17         1.8      NA
3 Mounta~ Aplo~ herbi Rode~ nt                  14.4       2.4      NA
4 Greate~ Blar~ omni  Sori~ lc                  14.9       2.3       0.133
5 Cow     Bos   herbi Arti~ domesticated         4         0.7       0.667
6 Three-~ Brad~ herbi Pilo~ <NA>                14.4       2.2       0.767
# ... with 3 more variables: awake <dbl>, brainwt <dbl>, bodywt <dbl>
```

Test	P-Value
sleep_cycle for insecti vs. carni	0.0028857
sleep_rem for insecti vs. omni	0.0046879
sleep_cycle for insecti vs. herbi	0.0049847
sleep_cycle for insecti vs. omni	0.0131055
sleep_rem for omni vs. herbi	0.0609072

Test	P-Value
sleep_rem for insecti vs. herbi	0.1087377
sleep_cycle for carni vs. omni	0.1571918
sleep_rem for carni vs. herbi	0.1636954
sleep_rem for insecti vs. carni	0.3201915
sleep_cycle for omni vs. herbi	0.3209207
sleep_rem for carni vs. omni	0.6082291
sleep_cycle for carni vs. herbi	0.6455530

Familywise Error Rate

One of the ways we can control our overall Type I error rate is by controlling the familywise error rate. This is defined as the probability of making at least one type I error out of all of our hypothesis tests, i.e.

$$FWER = P(\#\text{ of Type I errors} > 0).$$

Under this framework, we want the familywise error rate to be less than some chosen critical level α.

Bonferroni

The simplest and most widely-known correction for multiple comparisons is to use the Bonferroni correction. This simply adjusts the critial value of each individual test to be $\frac{\alpha}{n}$, where n is the number of tests we want to run and α is the desired familywise error rate. This is commonly used because of its simplicity and because it is guaranteed to control the familywise error rate under no assumptions about the dependency structure of the hypothesis tests (by Boole's inequality). However, this is an extremely conservative correction; if one wanted to run even a moderately sized number of tests, rejecting the null hypothesis in each individual test would be practically impossible.

For the `msleep` data, if we wanted an α level of 0.05, our Bonferroni-corrected p-value for each individual test would be $0.05/12 = 0.00416$; under this, only ome null hypotheses would be rejected.

Holm

The Holm procedure is known as the "step down" correction procedure. This is because the general idea is to reject null hypotheses in order of increasing p-value until we reach one that does not fall below its specified threshold. Formally, if we have $k \in 1, ..., n$ tests and a critical level α, the Holm method proceeds as:

1. Order p-values of all tests from smallest to largest (i.e., $p_{(1)}, p_{(2)} ..., p_{(n)}$).
2. Calculate $\frac{\alpha}{n+1-k}$ for $k \in 1, ..., n$.
3. Step down the list of p-values starting from the top until we find the first test such that $p_{(k)} > \frac{\alpha}{n+1-k}$.
4. Reject the null hypotheses for the tests above, i.e. $p_{(1)}, p_{(2)} ..., p_{(k-1)}$.

This test has is UMP over the Bonferroni test, meaning that we can guarantee the same Type I error control but with smaller Type II error probability. The only downside is that it is more complicated to implement. Below, we see this applied to the tests from the `msleep` data. We would stop after the first hypothesis test was rejected, since $p_{(2)} > \frac{\alpha}{11}$ is the first test outside the critical region.

Test	P-Value	Holm
sleep_cycle for insecti vs. carni	0.0028857	0.0041667

Test	P-Value	Holm
sleep_rem for insecti vs. omni	0.0046879	0.0045455
sleep_cycle for insecti vs. herbi	0.0049847	0.0050000
sleep_cycle for insecti vs. omni	0.0131055	0.0055556
sleep_rem for omni vs. herbi	0.0609072	0.0062500
sleep_rem for insecti vs. herbi	0.1087377	0.0071429
sleep_cycle for carni vs. omni	0.1571918	0.0083333
sleep_rem for carni vs. herbi	0.1636954	0.0100000
sleep_rem for insecti vs. carni	0.3201915	0.0125000
sleep_cycle for omni vs. herbi	0.3209207	0.0166667
sleep_rem for carni vs. omni	0.6082291	0.0250000
sleep_cycle for carni vs. herbi	0.6455530	0.0500000

Hochberg

On the other hand, the Hochberg procedure is known as the "step up" correction procedure. This is because the general idea is to not reject null hypotheses in order of dcreasing p-value until we reach one that falls below its specified threshold. Formally, if we have $k \in 1, ..., n$ tests and a critical level α, the Hochberg's method proceeds as:

1. Order p-values of all tests from smallest to largest (i.e., $p_{(1)}, p_{(2)}..., p_{(n)}$).
2. Calculate $\frac{\alpha}{n+1-k}$ for $k \in 1, ..., n$.
3. Step up the list of p-values from the bottom until we find the first test such that $p_{(k)} < \frac{\alpha}{n+1-k}$.
4. Reject the null hypotheses for that test and tests above, i.e. $p_{(1)}, p_{(2)}..., p_{(k)}$.

Below, we see the Hochberg procedure applied to the tests from the `msleep` data. We would reject the null hypothesis for the top 3 tests since $p_{(3)} < \frac{\alpha}{10}$ is the first test inside the critical region.

Test	P-Value	Hochberg
sleep_cycle for insecti vs. carni	0.0028857	0.0041667
sleep_rem for insecti vs. omni	0.0046879	0.0045455
sleep_cycle for insecti vs. herbi	0.0049847	0.0050000
sleep_cycle for insecti vs. omni	0.0131055	0.0055556
sleep_rem for omni vs. herbi	0.0609072	0.0062500
sleep_rem for insecti vs. herbi	0.1087377	0.0071429
sleep_cycle for carni vs. omni	0.1571918	0.0083333
sleep_rem for carni vs. herbi	0.1636954	0.0100000
sleep_rem for insecti vs. carni	0.3201915	0.0125000
sleep_cycle for omni vs. herbi	0.3209207	0.0166667
sleep_rem for carni vs. omni	0.6082291	0.0250000
sleep_cycle for carni vs. herbi	0.6455530	0.0500000

In general, the Hochberg procedure has higher power, i.e. we are more likely to reject H_0 when it is actually false. However, it assumes that the tests are independent. In fact, the Hochberg procedure is actually not guaranteed to be a universal level α test when the assumption is violated. In particular, the Hochberg procedure fails to limit the FWER below α when there is a strong negative correlation between tests.

Tukey Correction

Another method to note is Tukey's Honestly Significant Different (HSD). This method specifically applies to a situation where we want to compare multiple groups of a categorical variable based on the same continuous variable. This is essentially two-sample t-test where we want to test:

$$H_0 : \mu_i = \mu_j$$

$$H_a : \mu_i \neq \mu_j$$

for all different pairings of groups. If all group means are the same, the distribution of the test statistics $\mu_i - \mu_j$ follow what is called a studentized range distribution. Look it up if you're interested.

Tukey's HSD is most commonly used after an ANOVA test shows that there is a satistically significant difference in means amongst the groups. We want to make pairwise comparisons between all groups. to find exactly which groups are different, which is where the multiple comparisons come in. Just like ANOVAs, this test assumes normailty, homogeneity of variance, and independence. Below, we compare the REM sleep lengths for all of the different animal diet types. The ANOVA shows that the diet has a significant relationship with mean REM sleep time; the Tukey's HSD shows that the only the difference between omnivores and herbivores is statistically significant.

```
eat <- aov(msleep$sleep_rem ~ msleep$vore)

## Anova summary
summary(eat)
```

```
             Df Sum Sq Mean Sq F value Pr(>F)
msleep$vore   3  18.93   6.310   4.136 0.0105 *
Residuals    52  79.33   1.526
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
27 observations deleted due to missingness
```

```
## Tukey result
TukeyHSD(eat)
```

```
  Tukey multiple comparisons of means
    95% family-wise confidence level

Fit: aov(formula = msleep$sleep_rem ~ msleep$vore)

$`msleep$vore`
                     diff        lwr       upr     p adj
herbi-carni   -0.9233333 -2.1572326 0.3105659 0.2064029
insecti-carni  1.2350000 -0.7044558 3.1744558 0.3391994
omni-carni    -0.3344444 -1.6274149 0.9585261 0.9017792
insecti-herbi  2.1583333  0.3878606 3.9288061 0.0110250
omni-herbi     0.5888889 -0.4332940 1.6110718 0.4278715
omni-insecti  -1.5694444 -3.3815811 0.2426922 0.1115164
```

False Discovery Rate

Another way we can control our overall Type I error rate is by controlling the false discovery rate. The false discovery rate is defined as the expeted proportion of false positives out of all tests that are declared

significant, i.e.

$$FDR = E\left[\frac{\#(H_0 \text{ rejected } \cap H_0 \text{ is true})}{(\#H_0 \text{ rejected } \cap H_0 \text{ is true}) + \#(H_0 \text{ rejected } \cap H_0 \text{ is false})}\right].$$

In this case, we want the false discovery rate to be less than some chosen critical level α.

Benjamini-Hochberg

The most common way to control the false discovery rate is via the Benjamini-Hochberg procedure. This is related to the Hochberg step-up procedure in that the process of finding which null hypotheses to reject is similar. Formally, if we have $k \in 1, ..., n$ tests and a critical level α, the method proceeds as:

1. Order p-values of all tests from smallest to largest (i.e., $p_{(1)}, p_{(2)}..., p_{(n)}$).
2. Calculate $\frac{\alpha k}{n}$ for $k \in 1, ..., n$.
3. Step up the list of p-values from the bottom until we find the first test such that $p_{(k)} < \frac{\alpha}{n+1-k}$.
4. Reject the null hypotheses for that test and tests above, i.e. $p_{(1)}, p_{(2)}..., p_{(k)}$.

Like the Hochberg procedure above,, the Bejamini-Hochberg procedure assumes the independence of the hypothesis tests and can fail when there is a strong negative correlation between tests. Below, we see the proceudre applied to the p-values from the tests run on the `msleep` data.

Test	P-Value	Hochberg
sleep_cycle for insecti vs. carni	0.0028857	0.0041667
sleep_rem for insecti vs. omni	0.0046879	0.0045455
sleep_cycle for insecti vs. herbi	0.0049847	0.0050000
sleep_cycle for insecti vs. omni	0.0131055	0.0055556
sleep_rem for omni vs. herbi	0.0609072	0.0062500
sleep_rem for insecti vs. herbi	0.1087377	0.0071429
sleep_cycle for carni vs. omni	0.1571918	0.0083333
sleep_rem for carni vs. herbi	0.1636954	0.0100000
sleep_rem for insecti vs. carni	0.3201915	0.0125000
sleep_cycle for omni vs. herbi	0.3209207	0.0166667
sleep_rem for carni vs. omni	0.6082291	0.0250000
sleep_cycle for carni vs. herbi	0.6455530	0.0500000

We would reject the null hypothesis for the top 5 tests since $p_{(5)} < \frac{5\alpha}{12}$ is the first test inside the critical region.

Benjaimin-Hochberg-Yekutieli

The Benjamini-Hochberg-Yekutieli procedure extends the Benjaimin-Hochberg procedure above by adjusting for any possible correlations between the hypothesis tests, relaxing the independence assumption. The method is exactly the same, except that our threshold is now $\frac{\alpha k}{c \times n}$, where $c = \sum_{k=1}^{m} \frac{1}{k}$.