Four strategies for dealing with multiple comparisons

Eve Slavich, Stats Central

Four Strategies

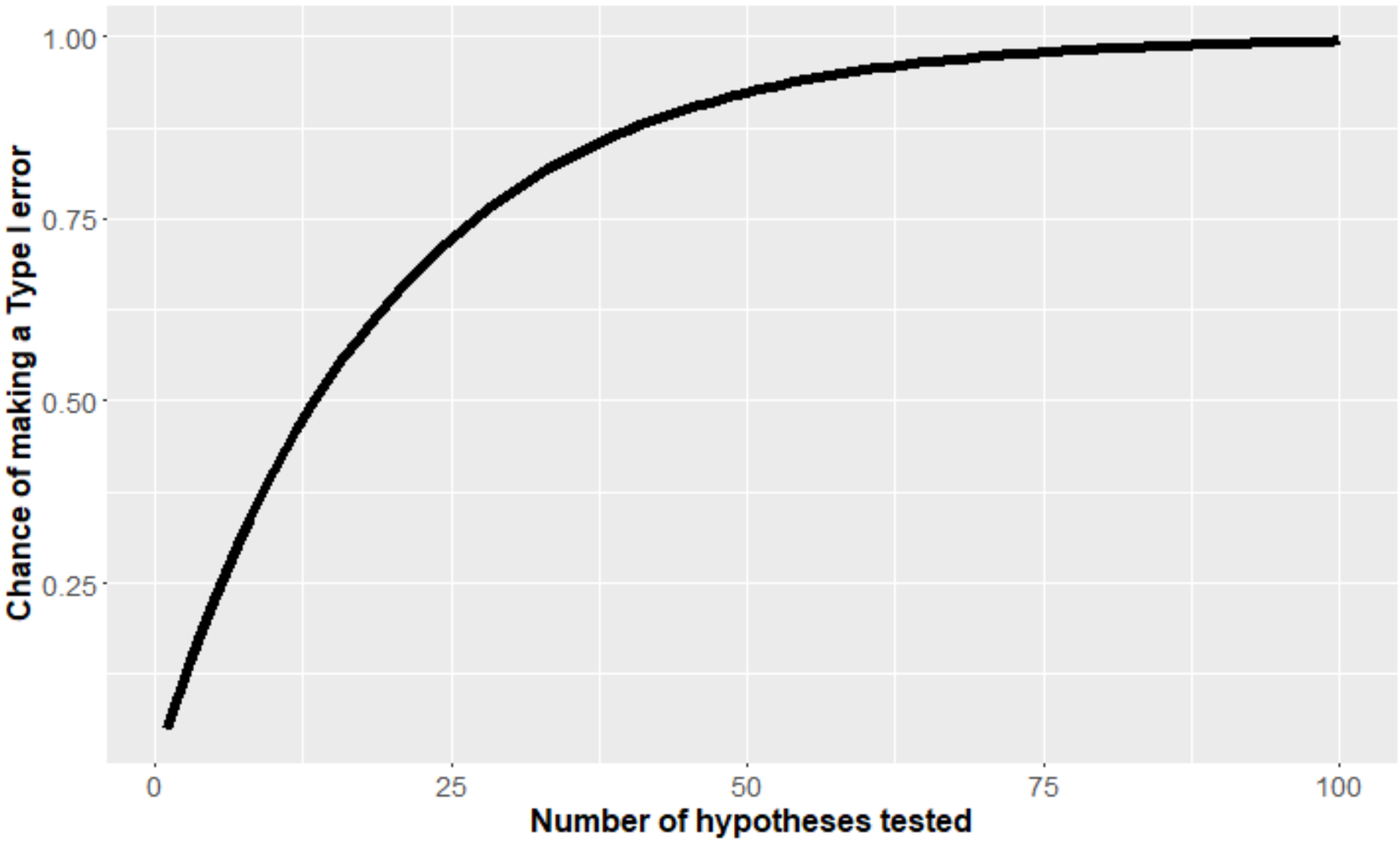
- Control the Familywise Error Rate with the Holm Bonferroni Adjustment.
- Control the False Discovery Rate with the Benjamini Hochberg Adjustment.
- Control the Familywise Error Rate using simulation, where there are correlated outcome variables.
- Don't conduct multiple comparisons - create an index or use a multivariate test, where possible.

Multiple comparisons are really common

- Any time you are doing multiple statistical tests.
- E.g. planned or unplanned comparisons of the levels within a categorical variable (such as where there are multiple treatment arms)
- E.g. you have measured multiple outcome variables

What's the Problem?

- The more hypotheses you test the higher the chance that one is rejected by chance (Type I error)
- The chance is $1 - (1 - \alpha)^m$, where α is your significance level and m is the number of tests.
- If $\alpha = 0.05$ the chance of an error is shown in the plot



Definitions

- **Type I Error**: The chance of a false positive- rejecting a null hypothesis when there is no effect in the population.
- **Family-wise Error Rate (FWER)**: The chance of a false positive in at least one hypothesis test
- **False Discovery Rate (FDR)**: The proportion of "discoveries" (rejected null hypotheses) that are false (incorrect rejections)

Definitions

- **Type I Error**: The chance of a false positive- rejecting a null hypothesis when there is no effect in the population.
- **Family-wise Error Rate (FWER)**: The chance of a false positive in at least one hypothesis test-- *The chance that one (or more) of the "red" conclusions is wrong.*
- **False Discovery Rate (FDR)**: The proportion of "discoveries" (rejected null hypotheses) that are false (incorrect rejections)-- *The proportion of red conclusions that are wrong.*

Family-wise Error Rate versus False Discovery Rate - which one should I control?

- Controlling the FDR is more sensitive (generally gives more "significant" results) and can give you greater statistical power- the power to detect an effect that is truly there. But you can't be 'sure' of any particular hypothesis test- it's expected some are falsely significant but you won't know which ones.
- Controlling the FWER means that you can be "sure" of every single test result.
- If the cost of false negatives is high, control the FDR. If you want to be sure of every single hypothesis, control the FWER.
- Controlling the FWER, the more tests you do, the less power you get. Controlling the FDR- the tests with no true effect you do, the less power you get.

Holm - Bonferroni adjustment (Controls the FWER)

- Rank the hypotheses from smallest p-value to largest p-value. So $p_1 \leq p_2 \leq \ldots \leq p_n$
- Starting with the smallest p-value, (p_1), and then each subsequent p-value, p_i is signficant if $p_i < \frac{\alpha}{m-i+1}$, where m is the number of tests, α is the desired Family-wise Error Rate and i is the rank of the i^{th} hypothesis.
- At the first non-rejected hypothesis, stop testing- all subsequent hypotheses are not significant.
- The adjusted p-values are $p_i \times (m - i + 1)$, which can be rejected at whichever significance level (α) is desired
- This is preferred to and more powerful than the Bonferroni adjustment, which is known to be conservative (thus has less power to detect a true positive!).

Holm - Bonferroni adjustment in R

Julia compared the effects of 3 treatments and a control on blood pressure. She conducted a total of four planned comparisons.

```
pValues =  c(0.01, 0.2, 0.08, 0.03)    # a vector of the raw p values
m = length(pValues)                    # number of comparisons
#Calculate the adjusted p values
pValues*(m - rank(pValues)+1)
```

```
## [1] 0.04 0.20 0.16 0.09
```

```
#Alternatively use the built in R function p.adjust!
p.adjust(pValues, method = "holm")
```

```
## [1] 0.04 0.20 0.16 0.09
```

```
#using a 0.05 significance level, we reject the first hypothesis, but accept the last 4.
```

Holm - Bonferroni adjustment in R

What happens if Julia did all pairwise comparisons between the 3 treatments and control - i.e. 6 comparisons.

```
# a vector of the raw p values
pValues =  c(0.01, 0.2, 0.08, 0.03, 0.02, 0.01)
p.adjust(pValues, method = "holm")
```

```
## [1] 0.06 0.20 0.16 0.09 0.08 0.06
```

Controlling the FWER at 0.05, we don't reject any of the hypotheses!

Benjamini-Hochberg adjustment (Controls the FDR)

- Rank the hypotheses from smallest p-value to largest p-value. So $p_1 \leq p_2 \leq \ldots \leq p_n$
- Starting with the smallest p-value, (p_1), and then each subsequent p-value, p_i is signficant if $p_i < \frac{i}{m}\,\alpha$, where m is the number of tests, α is the desired False Discovery Rate and i is the rank of the i^{th} p-value.
- At the first non-rejected hypothesis, stop testing- all subsequent hypotheses are not significant.
- The adjusted p-value p_i^{adj} is $min(p_i \times \frac{m}{i}\,, p_{i+1} \times \frac{m}{i+1})$, which can be rejected at whichever false discovery level (α) is desired.
- If the cost of a false negative is high and the cost of experimentation is cheap (e.g. genomics) then people often use a higher FDR, e.g. 0.1 or 0.2 so they don't miss anything important.

Benjamini-Hochberg adjustment (in R)

E.g. Li Na is testing for an effect in any of 110 genes. She tries controlling the FDR at 0.05 and 0.1.

```
#pvalues contain 100 null results and 10 true positives
set.seed(2018)
pValues = c( runif(10,0,0.01), runif(100,0,1))
raw_discovery_rate = sum(pValues<=0.05) #how many p values are less than 0.05
AdjustedPValues = p.adjust(pValues, method = "fdr")
discovery_rate_fdr0.05 = sum(AdjustedPValues<0.05)#how many p values significant with FDR=0.05
discovery_rate_fdr0.1 = sum(AdjustedPValues<0.1)#how many p values significant with FDR=0.1
c(raw_discovery_rate = raw_discovery_rate, discovery_rate_fdr0.05 = discovery_rate_fdr0.05, di
```

```
##     raw_discovery_rate discovery_rate_fdr0.05  discovery_rate_fdr0.1
##                     15                      0                     12
```

With a FDR of 0.05 we don't reject any hypotheses, whilst with a FDR of 0.1 we reject 12 hypotheses

What happens if there are more hypotheses where there is no effect?

```
#more negatives - 200 null results and 10 true positives, testing for an effect in 210 genes
set.seed(2018)
pValues = c( runif(10,0,0.01), runif(200,0,1))
raw_discovery_rate = sum(pValues<=0.05) #how many p values are less than 0.05
AdjustedPValues = p.adjust(pValues, method = "fdr")
discovery_rate_fdr0.05 = sum(AdjustedPValues<0.05)#how many p values significant with FDR=0.05
discovery_rate_fdr0.1 = sum(AdjustedPValues<0.1)#how many p values significant with FDR=0.1
c(raw_discovery_rate = raw_discovery_rate, discovery_rate_fdr0.05 = discovery_rate_fdr0.05, di
```

```
##     raw_discovery_rate discovery_rate_fdr0.05  discovery_rate_fdr0.1
##                     18                      0                      0
```

Terrible power, we didn't detect any of the true positives!

Situation is better if there is more evidence for the true positives

```
set.seed(2018)
pValues = c( runif(10,0,0.001), runif(200,0,1)) #the true positives have smaller p values now
raw_discovery_rate = sum(pValues<=0.05) #how many p values are less than 0.05
AdjustedPValues = p.adjust(pValues, method = "fdr")
discovery_rate_fdr0.05 = sum(AdjustedPValues<0.05)#how many p values significant with FDR=0.05
discovery_rate_fdr0.1 = sum(AdjustedPValues<0.1)#how many p values significant with FDR=0.1
c(raw_discovery_rate = raw_discovery_rate, discovery_rate_fdr0.05 = discovery_rate_fdr0.05, di
```

```
##     raw_discovery_rate discovery_rate_fdr0.05  discovery_rate_fdr0.1
##                     18                      10                     11
```

We will reject 10 or 11 of the hypotheses using a FDR of 0.05 or 0.1. The more hypotheses you test, the more evidence you need to claim some are signicant.

What happens if there are more true positives?

```
set.seed(2018)
#pvalues contain 100 null results and 40 true positives
pValues = c( runif(40,0,0.01), runif(100,0,1))
raw_discovery_rate = sum(pValues<=0.05) #how many p values are less than 0.05
AdjustedPValues = p.adjust(pValues, method = "fdr")
discovery_rate_fdr0.05 = sum(AdjustedPValues<0.05)#how many p values significant with FDR=0.05
discovery_rate_fdr0.1 = sum(AdjustedPValues<0.1)#how many p values significant with FDR=0.1
c(raw_discovery_rate = raw_discovery_rate, discovery_rate_fdr0.05 = discovery_rate_fdr0.05, di
```

```
##     raw_discovery_rate discovery_rate_fdr0.05  discovery_rate_fdr0.1
##                     43                      42                     42
```

We will reject 42 of the hypotheses using a FDR of 0.05 or 0.1. So we probably have pretty good power!

What happens if there are even more true positives (and how does this compare to the Holm-Bonferroni method)?

```
set.seed(2018)
#pvalues contain 100 null results and 400 true positives
pValues = c( runif(400,0,0.01), runif(100,0,1))
raw_discovery_rate = sum(pValues<=0.05) #how many p values are less than 0.05
discovery_rate_fdr0.05 = sum(p.adjust(pValues, method = "fdr")<0.05) #how many p values signif
discovery_rate_fwer0.05 = sum(p.adjust(pValues, method = "holm")<0.05) #how many p values sign

c(raw_discovery_rate = raw_discovery_rate, discovery_rate_fdr0.05 = discovery_rate_fdr0.05, di
```

```
##      raw_discovery_rate  discovery_rate_fdr0.05 discovery_rate_fwer0.05
##                     403                     403                       3
```

Controlling the FDR we don't lose any power by adding more true positives, but we definitely do if we stick to FWER. Controlling the FWER, we can be sure that only 3 of these hypotheses should be rejected, but we know 400 of them were truly false!

Notes

- The methods will have very low power if the proportion of true positives is low or the p-values are on the larger side. So if you add a lot of 'noise'/ nonsense comparisons, you end up losing power to detect the true positives with p-values on the larger side (e.g. 0.01), even using FDR instead of FWER.
- There are some tweaks to get slightly higher power- e.g. the adaptive Benjamini Hochberg procedure (2000).
- This can be implemented in R using the mutoss package and the *adaptiveBH(pValues, alpha =)* command, or in the multtest package and the command *mt.rawp2adjp(pValues, proc = "ABH")*

Strategies for multiple correlated outcome variables

- The above approaches do not try to maximise power by taking the correlation between variables into account- in particular where there are correlated outcome variables you may be able to borrow strength across variables.
- E.g. Sam measured a family of physiological variables like activity, heart rate (HR), respiration, blood pressure and weight to determine the effect of a treatment.
- E.g. Arjun measured a host of biodiversity indicator variables to determine the effect of a bush regeneration project. He expects these variables may be correlated and respond in similar ways to bush regeneration.

Simulation Approach to control the FWER

- A simulation approach can be used to control the Family Wise Error Rate, by simulating correlated outcome data from the model under the family-wide hypothesis of no treatment effect
- Test the m hypotheses on the simulated data.
- Randomly permute the treatment labels and test the m hypotheses again, N=1000 times (for example)
- For a sequence of thresholds (e.g. a range of values between 0.001 and 0.05) count up the proportion of simulations in which at least one test was deemed significant if that threshold was used.
- Use as a significance threshold the threshold that delivers a Type I error rate of 0.05. i.e. using that threshold, 5% of simulations had at least one test significant.
- This approach requires a bit more effort and customised coding.

More strategies for multiple correlated outcome variables

- Sometimes it makes sense to develop an index which combines these variables- e.g. the inverse covariance weighted mean effect index (Anderson, M.L, 2008, Journal of the American Statistical Association), so then you just have one hypothesis test.
- Alternatively it may be possible to take a multivariate approach. For example fitting a multivariate normal regression and testing for a treatment effect on the multiple outcome variables simultaneously.
- Both these approaches avoid multiple comparisons to draw a single conclusion about a treatment on a group of outcome variables (for example).

Other things people do (there are way more than 4 strategies in the literature!)

- There a couple of useful tests for specific scenarios that may apply to you. E.g. Tukey's Test and Dunnett's Test. These are good the the specific scenario they apply to, but can't be used outside these scenarios.
- Tukey's test: Use when all-possible pairwise comparisons between the levels of a factor are being conducted and when sample sizes are unequal
- Dunnett's test: The best for conducting a many-to-one comparison (e.g. comparing treatments A, B, and C to a control) but can't be used in any other situation.
- These are available in all decent Statistical software.

Conclusions

- You need to decide ahead of time whether the Family Wise Error rate or the False Discovery rate is important, and set these rates.
- If you control the FWER, the more hypotheses you add the worse power you get.
- If you control the FDR, the more null result hypotheses you add the worse power you get.
- Holm-Bonferroni can be used to control the FWER.
- Benjamini-hochberg can be used to control the FDR.
- Tweaks available for marginal improvements to power, but don't expect any miracles if you have a large number of non significant results.
- If the hypotheses are on correlated outcome variables, a simulation approach to controlling the FWER may be used to increase power.
- Sometimes you may avoid multiple comparisons by creating an index or using a multivariate test, borrowing strength across your variables.