STA305/1004 - Class 15

March 8, 2017

Today's Class

- Coding qualitative predictors in regression models
- Estimating treatment effects using least squares
- Multiple comparisons
- Sample size for ANOVA

Coding Qualitative Predictors in Regression Models

- A dummy or indicator variable in a regression takes on a finite number of values so that different categories of a nominal variable can be identified.
- The term dummy reflects the fact that the values taken on by such variables (e.g., 0, 1, -1) do not indicate meaningful measurements but rather categories of interest. (Kleinbaum et al., 1998)

Coding Qualitative Predictors in Regression Models

Consider a regression model: $y = \beta_0 + \beta_1 X_i + \epsilon$

Examples of dummy variables are:

$$X_1 = \begin{cases} 1 & \text{if treatment A} \\ 0 & \text{otherwise} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if subject is male} \\ -1 & \text{if subject is female} \end{cases}$$

The variables X_1, X_2 are nominal variables describing treatment group and sex respectively.

Coding Qualitative Predictors in Regression Models

The following rule should be applied to avoid collinearity in defining a dummy variable for regression analysis:

> *if the nominal independent variable of interest has k categories then exactly $k - 1$ dummy variables should be defined to index the categories if the regression model contains an intercept term.*

Dummy Coding

- Dummy coding compares each level to the reference level.
- The intercept is the mean of the reference group.
- Suppose that we would like to compare the mean number of candy colours in each box. The data from 3 smarties boxes are below.

```r
count <- c(4,3,4,3,1,4,2,5,1,1,2,4)
colour <- as.factor(c(rep("Yellow",3),rep("Purple",3),
                      rep("Green",3),rep("Pink",3)))
```

colour	count
Yellow	4
Yellow	3
Yellow	4
Purple	3
Purple	1
Purple	4
Green	2
Green	5
Green	1
Pink	1
Pink	2
Pink	4

Dummy Coding

The average and sd of each colour is:

```
#Get means for each flavour
sapply(split(count,colour),mean)
```

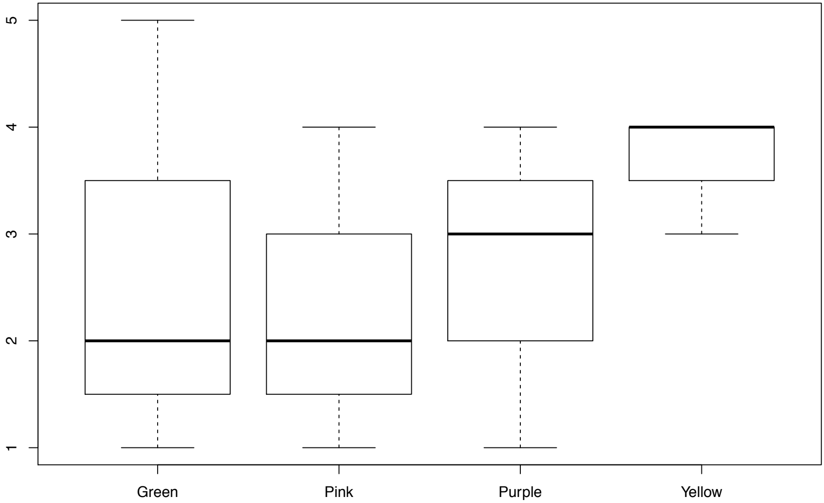
```
##    Green     Pink   Purple   Yellow
## 2.666667 2.333333 2.666667 3.666667
```

```
#Get standard deviations for each flavour
sapply(split(count,colour),sd)
```

```
##     Green      Pink    Purple    Yellow
## 2.0816660 1.5275252 1.5275252 0.5773503
```

Dummy Coding

```
boxplot(count~colour)
```



Dummy Coding

Dummy coding is the default in R and the most common coding scheme. It compares each level of the categorical variable to a fixed reference level.

```r
contrasts(colour) <- contr.treatment(4)
contrasts(colour)  # print dummy coding - base is Green
```

```
##        2 3 4
## Green  0 0 0
## Pink   1 0 0
## Purple 0 1 0
## Yellow 0 0 1
```

Green is the reference category. The first column compares Pink to Green, the second column compares Purple to Green, and the third column compares Yellow to Green. The the three columns define three dummy variables:

Dummy Coding

$$X_1 = \begin{cases} 1 & \text{if smartie is pink} \\ 0 & \text{otherwise} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if smartie is purple} \\ 0 & \text{otherwise} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$

If $X_1 = X_2 = X_3 = 0$ then the colour of the smartie is green - the reference category. This shows that we only require 3 dummy variables to define a nominal variable with 4 categories.

Dummy Coding

To change the reference level change the value of base in `contr.treatment()`.

```
contrasts(colour) <- contr.treatment(4,base = 2) # Now reference is pink
contrasts(colour)
```

```
##        1 3 4
## Green  1 0 0
## Pink   0 0 0
## Purple 0 1 0
## Yellow 0 0 1
```

```
contrasts(colour) <- contr.treatment(4,base = 4) # Now reference is yellow
contrasts(colour)
```

```
##        1 2 3
## Green  1 0 0
## Pink   0 1 0
## Purple 0 0 1
## Yellow 0 0 0
```

Deviation Coding

- This coding system compares the mean of the dependent variable for a given level to the overall mean of the dependent variable.
- $$X_1 = \begin{cases} 1 & \text{if smartie is green} \\ -1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$

  $$X_2 = \begin{cases} 1 & \text{if smartie is pink} \\ -1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$

  $$X_3 = \begin{cases} 1 & \text{if smartie is purple} \\ -1 & \text{if smartie is yellow} \\ 0 & \text{otherwise} \end{cases}$$
- 1 is used to compare a level to all other levels and -1 is assigned to yellow because it's the level that will never be compared to the other levels.
- In R the variables can be created using the `contr.sum()` function.
- The argument of 4 in `contr.sum(4)` indicates the number of levels of the factor.

Deviation Coding

```
contrasts(colour) <- contr.sum(4)
contrasts(colour)
```

```
##        [,1] [,2] [,3]
## Green     1    0    0
## Pink      0    1    0
## Purple    0    0    1
## Yellow   -1   -1   -1
```

Example - blood coagualtion study

The table below gives coagulation times for blood samples drawn from 24 animals receiving four different diets A, B, C, and D.

	A	B	C	D
	60	65	71	62
	63	66	66	60
	59	67	68	61
	63	63	68	64
	62	64	67	63
	59	71	68	56
Treatment Average	61	66	68	61
Grand Average	64	64	64	64
Difference	-3	2	4	-3

Estimating treatment effects using least squares

y_{ij} is the j^{th} observation under the i^{th} treatment. Let μ be the overall mean. The model for diet $y_{ij} = \mu + \tau_i + \epsilon_{ij}$, $\epsilon_{ij} \sim N(0, \sigma^2)$ can be written in terms of the dummy variables X_1, X_2, X_3 as:

$$y_{ij} = \mu + \tau_1 X_{1j} + \tau_2 X_{2j} + \tau_3 X_{3j} + \epsilon_{ij},$$

where,

$$X_{1j} = \begin{cases} 1 & \text{if jth unit recieves diet 2} \\ 0 & \text{otherwise} \end{cases}$$

$$X_{2j} = \begin{cases} 1 & \text{if jth unit recieves diet 3} \\ 0 & \text{otherwise} \end{cases}$$

$$X_{3j} = \begin{cases} 1 & \text{if jth unit recieves diet 4} \\ 0 & \text{otherwise} \end{cases}$$

Estimating treatment effects using least squares

It follows that $E(y_{Aj}) = \mu_A = \mu$ is the mean of diet A so

$$\begin{aligned} E(y_{Bj}) &= \mu_B = \mu_A + \tau_1 \Rightarrow \tau_1 = \mu_B - \mu_A \\ E(y_{Cj}) &= \mu_C = \mu_A + \tau_2 \Rightarrow \tau_2 = \mu_C - \mu_A \\ E(y_{Dj}) &= \mu_D = \mu_A + \tau_3 \Rightarrow \tau_3 = \mu_D - \mu_A \end{aligned}$$

The least squares estimates are:

$$\begin{aligned} \hat{\mu} &= \bar{y}_{1\cdot}, \\ \hat{\tau}_1 &= \bar{y}_{2\cdot} - \bar{y}_{1\cdot}, \\ \hat{\tau}_2 &= \bar{y}_{3\cdot} - \bar{y}_{1\cdot}, \\ \hat{\tau}_3 &= \bar{y}_{3\cdot} - \bar{y}_{1\cdot}. \end{aligned}$$

Estimating treatment effects using least squares

- This model can also be written in matrix notation

$$y = X\beta + \epsilon$$

,

where $\beta = (\mu, \tau_1, \tau_2, \tau_3)\,, X = (\mathbf{1}, X_{i1}, X_{i2}, X_{i3})$, and $\epsilon = (\epsilon_{ij})$.

- X is an 24×4 design matrix with $\mathbf{1}$ is a 24×1 column vector of 1s, and ϵ is an 24×1 column vector.
- Note that τ_4 corresponding to the 4th treatment is implicitly set to 0. It is used as a constraint so that that $(X'X)^{-1}$ exists.

Example - blood coagualtion study (treatment coding)

```r
contrasts(tab0401$diets)
```

```
  B C D
A 0 0 0
B 1 0 0
C 0 1 0
D 0 0 1
```

```r
lm.diets <- lm(y~diets,data=tab0401);round(summary(lm.diets)$coefficients,2)
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)       61       0.97   63.14        0
dietsB             5       1.37    3.66        0
dietsC             7       1.37    5.12        0
dietsD             0       1.37    0.00        1
```

Example - blood coagualtion study (treatment coding)

The averages for each of the four diets are in the table below.

Diet	A ($j = 1$)	B ($j = 2$)	C ($j = 3$)	D ($j = 4$)
Average ($\bar{y}_{j\cdot}$)	61	66	68	61

$$\begin{aligned}
\bar{y}_{1\cdot} &= 61, \\
\hat{\tau}_1 &= \bar{y}_{2\cdot} - \bar{y}_{1\cdot} = 5 \\
\hat{\tau}_2 &= \bar{y}_{3\cdot} - \bar{y}_{1\cdot} = 7 \\
\hat{\tau}_3 &= \bar{y}_{3\cdot} - \bar{y}_{1\cdot} = -9.9 \times 10^{-15}.
\end{aligned}$$

Example - blood coagualtion study (treatment coding)

The design matrix (first 12 observations) is

```
model.matrix(lm.diets)[1:12,]
```

```
##    (Intercept) dietsB dietsC dietsD
## 1            1      0      0      0
## 2            1      0      0      0
## 3            1      0      0      0
## 4            1      0      0      0
## 5            1      0      0      0
## 6            1      0      0      0
## 7            1      1      0      0
## 8            1      1      0      0
## 9            1      1      0      0
## 10           1      1      0      0
## 11           1      1      0      0
## 12           1      1      0      0
```

Example - blood coagualtion study (treatment coding)

The design matrix (first 12 observations) with the observations y and treatment variable `diets` (first 12 observations) is

```r
cbind(tab0401$y,tab0401$diets,model.matrix(lm.diets))[1:12,]
```

```
##           (Intercept) dietsB dietsC dietsD
## 1  62 1           1      0      0      0
## 2  60 1           1      0      0      0
## 3  63 1           1      0      0      0
## 4  59 1           1      0      0      0
## 5  63 1           1      0      0      0
## 6  59 1           1      0      0      0
## 7  63 2           1      1      0      0
## 8  67 2           1      1      0      0
## 9  71 2           1      1      0      0
## 10 64 2           1      1      0      0
## 11 65 2           1      1      0      0
## 12 66 2           1      1      0      0
```

Example - blood coagualtion study (deviation coding)

If deviation coding was used then the parameter estimates would represent different treatment effects. In the regression model the dummy variables would be defined as

$$X_1 = \begin{cases} 1 & \text{if diet is A} \\ -1 & \text{if diet is D} \\ 0 & \text{otherwise} \end{cases}$$

$$X_2 = \begin{cases} 1 & \text{if diet is B} \\ -1 & \text{if diet is D} \\ 0 & \text{otherwise} \end{cases}$$

$$X_3 = \begin{cases} 1 & \text{if diet is C} \\ -1 & \text{if diet is D} \\ 0 & \text{otherwise} \end{cases}$$

Example - blood coagualtion study (deviation coding)

It follows that

$$
\begin{aligned}
E(y_{Aj}) &= \mu_A = \tau_0 + \tau_1 \\
E(y_{Bj}) &= \mu_B = \tau_0 + \tau_2 \\
E(y_{Cj}) &= \mu_C = \tau_0 + \tau_3 \\
E(y_{Dj}) &= \mu_D = \tau_0 - \tau_1 - \tau_2 - \tau_3
\end{aligned}
$$

So,

$$
\begin{aligned}
\tau_0 &= \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4} \\
\tau_1 &= \mu_A - \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4} \\
\tau_2 &= \mu_B - \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4} \\
\tau_3 &= \mu_C - \frac{\mu_A + \mu_B + \mu_C + \mu_D}{4}
\end{aligned}
$$

Example - blood coagualtion study (deviation coding)

```r
attach(tab0401)
contrasts(tab0401$diets) <- contr.sum(4)
lm.diets <- lm(y~diets,data=tab0401)
round(summary(lm.diets)$coefficients,2)
```

```
            Estimate Std. Error t value Pr(>|t|)
(Intercept)       64       0.48  132.49     0.00
diets1            -3       0.84   -3.59     0.00
diets2             2       0.84    2.39     0.03
diets3             4       0.84    4.78     0.00
```

- The estimate of the intercept $\hat{\tau}_0$ is the grand average.
- The slope estimates $\hat{\tau_1}, \hat{\tau_2}, \hat{\tau_3}$ are the differences between the treatment averages and grand average of diets A, B, and C.

Multiple Comparisons

Suppose that experimental units were randomly assigned to three treatment groups. The hypothesis of intrest is:

$$H_0 : \mu_1 = \mu_2 = \mu_3 \,\text{vs.}\, H_1 : \mu_i \neq \mu_j.$$

Now, suppose that we reject H_0 at level α. Which pairs of means are significantly different from each other at level α? There are $\binom{3}{2} = 3$ possibilites.

1. $\mu_1 \neq \mu_2$
2. $\mu_1 \neq \mu_3$
3. $\mu_2 \neq \mu_3$

Multiple Comparisons

Suppose that $k = 3$ separate (independent) hypothesis tests at level α tests are conducted:

$$H_{0_k} : \mu_i = \mu_j \, \text{vs.} \, H_{1_k} : \mu_i \neq \mu_j,$$

When H_0 is true, $P\left(\text{reject } H_0\right) = \alpha \Rightarrow 1 - P\left(\text{do not reject } H_0\right) = 1 - (1 - \alpha)$.

So, if H_0 is true then

$$P\left(\text{reject at least one } H_{0_k}\right) = 1 - P\left(\text{do not reject any } H_{0_k}\right)$$

This is the same as

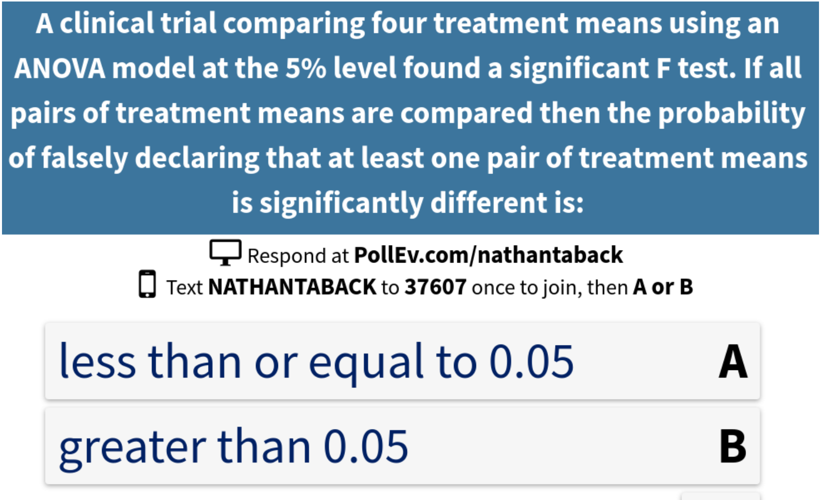
$$1 - P\left(\text{do not reject } H_{0_1} \text{and do not reject } H_{0_2} \text{and do not reject } H_{0_3}\right)$$

or since the hypotheses are independent

$$1 - P\left(\text{do not reject } H_{0_1}\right) P\left(\text{do not reject } H_{0_2}\right) P\left(\text{do not reject } H_{0_3}\right) = 1 - (1 - \alpha)^3$$

If $\alpha = 0.05$ then the probability that at least one H_0 will be falsely rejected is $1 - (1 - .05)^3 = 0.14$, which is almost three times the type I error rate.

Multiple Comparisons



A clinical trial comparing four treatment means using an ANOVA model at the 5% level found a significant F test. If all pairs of treatment means are compared then the probability of falsely declaring that at least one pair of treatment means is significantly different is:

Respond at **PollEv.com/nathantaback**

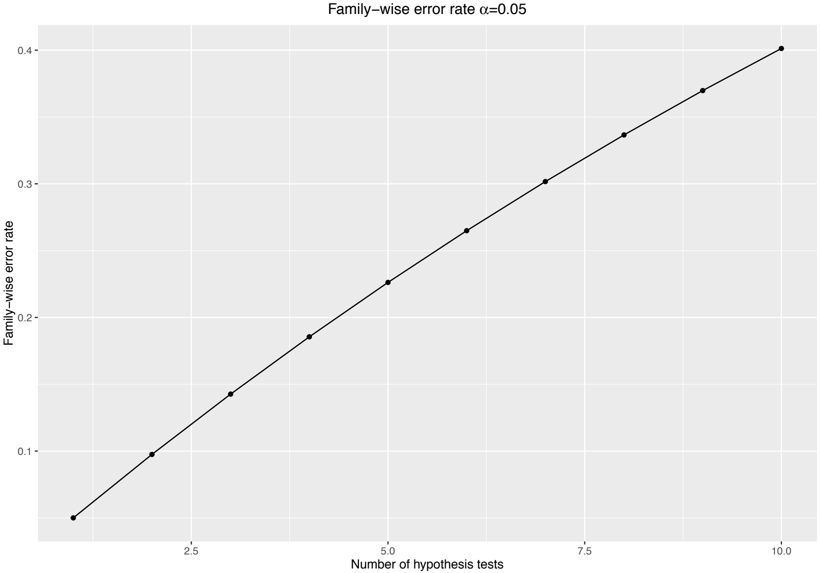
Text **NATHANTABACK** to **37607** once to join, then **A or B**

less than or equal to 0.05 **A**

greater than 0.05 **B**

Figure 1:

Multiple Comparisons



Multiple Comparisons

In general if

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_k \, \text{vs.} \, H_1 : \mu_i \neq \mu_j.$$

If c independent hypotheses are conducted then the probability

$$P\left(\text{reject at least one } H_{0_k}\right) = 1 - (1 - \alpha)^c$$

is called the **family-wise error rate**.

The **pairwise error rate** is $P\left(\text{reject } H_{0_k}\right) = \alpha$ for any c.

The Multiple Comparisons Problem

- The multiple comparison problem is that multiple hypotheses are tested level α which increases the probability that at least one of the hypotheses will be falsely rejected (family-wise error rate).
- If treatment means are significantly different from the ANOVA F test then researchers will usually want to explore where the differences lie.
- Is it appropriate to test for differences looking at all pairwise comparisons?
- Testing all possible pairs increases the type I error rate.
- This means the chance that there is a higher probability, beyond the pre-stated type I error rate (e.g. 0.05), that that a significant difference is detected when the truth is that no difference exists.

Neural correlates of interspecies perspective taking in the post-mortem Atlantic Salmon: An argument for multiple comparisons correction

Craig M. Bennett[1], Abigail A. Baird[2], Michael B. Miller[1], and George L. Wolford[3]

[1] Psychology Department, University of California Santa Barbara, Santa Barbara, CA; [2] Department of Psychology, Vassar College, Poughkeepsie, NY;
[3] Department of Psychological & Brain Sciences, Dartmouth College, Hanover, NH

INTRODUCTION

With the extreme dimensionality of functional neuroimaging data comes extreme risk for false positives. Across the 130,000 voxels in a typical fMRI volume the probability of a false positive is almost certain. Correction for multiple comparisons should be completed with these datasets, but is often ignored by investigators. To illustrate the magnitude of the problem we carried out a real experiment that demonstrates the danger of not correcting for chance properly.

METHODS

Subject. One mature Atlantic Salmon (Salmo salar) participated in the fMRI study. The salmon was approximately 18 inches long, weighed 3.8 lbs, and was not alive at the time of scanning.

Task. The task administered to the salmon involved completing an open-ended mentalizing task. The salmon was shown a series of photographs depicting human individuals in social situations with a specified emotional valence. The salmon was asked to determine what emotion the individual in the photo must have been experiencing.

Design. Stimuli were presented in a block design with each photo presented for 10 seconds followed by 12 seconds of rest. A total of 15 photos were displayed. Total scan time was 5.5 minutes.

Preprocessing. Image processing was completed using SPM2. Preprocessing steps for the functional imaging data included a 6-parameter rigid-body affine realignment of the fMRI timeseries, coregistration of the data to a T_1-weighted anatomical image, and 8 mm full-width at half-maximum (FWHM) Gaussian smoothing.

Analysis. Voxelwise statistics on the salmon data were calculated through an ordinary least-squares estimation of the general linear model (GLM). Predictors of the hemodynamic response were modeled by a boxcar function convolved with a canonical hemodynamic response. A temporal high pass filter of 128 seconds was include to account for low frequency drift. No autocorrelation correction was applied.

Voxel Selection. Two methods were used for the correction of multiple comparisons in the fMRI results. The first method controlled the overall false discovery rate (FDR) and was based on a method defined by Benjamini and Hochberg (1995). The second method controlled the overall familywise error rate (FWER) through the use of Gaussian random field theory. This was done using algorithms originally devised by Friston et al. (1994).

DISCUSSION

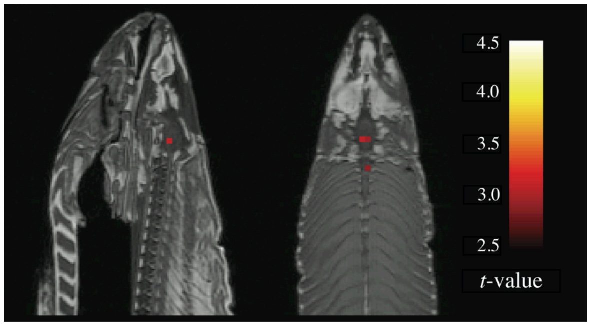
Can we conclude from this data that the salmon is engaging in the perspective-taking task? Certainly not. What we can determine is that random noise in the EPI timeseries may yield spurious results if multiple comparisons are not controlled for. Adaptive methods for controlling the FDR and FWER are excellent options and are widely available in all major fMRI analysis packages. We argue that relying on standard statistical thresholds ($p < 0.001$) and low minimum cluster sizes ($k > 8$) is an ineffective control for multiple comparisons. We further argue that the vast majority of fMRI studies should be utilizing multiple comparisons correction as standard practice in the computation of their statistics.

REFERENCES

Benjamini Y and Hochberg Y (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B*, 57:289-300.

Friston KJ, Worsley KJ, Frackowiak RSJ, Mazziotta JC, and Evans AC. (1994). Assessing the significance of focal activations using their spatial extent. *Human Brain Mapping*, 1:214-220.

GLM RESULTS

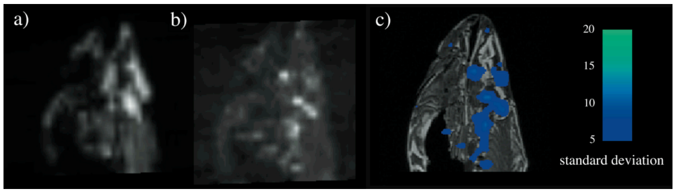


A *t*-contrast was used to test for regions with significant BOLD signal change during the photo condition compared to rest. The parameters for this comparison were $t(131) > 3.15$, p(uncorrected) < 0.001, 3 voxel extent threshold.

Several active voxels were discovered in a cluster located within the salmon's brain cavity (Figure 1, see above). The size of this cluster was 81 mm^3 with a cluster-level significance of $p = 0.001$. Due to the coarse resolution of the echo-planar image acquisition and the relatively small size of the salmon brain further discrimination between brain regions could not be completed. Out of a search volume of 8064 voxels a total of 16 voxels were significant.

Identical *t*-contrasts controlling the false discovery rate (FDR) and familywise error rate (FWER) were completed. These contrasts indicated no active voxels, even at relaxed statistical thresholds ($p = 0.25$).

VOXELWISE VARIABILITY

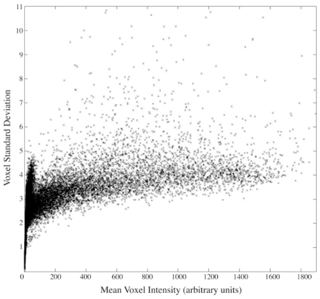


To examine the spatial configuration of false positives we completed a variability analysis of the fMRI timeseries. On a voxel-by-voxel basis we calculated the standard deviation of signal values across all 140 volumes.

We observed clustering of highly variable voxels into groups near areas of high voxel signal intensity. Figure 2a shows the mean EPI image for all 140 image volumes. Figure 2b shows the standard deviation values of each voxel. Figure 2c shows thresholded standard deviation values overlaid onto a high-resolution T_1-weighted image.

To

To investigate this effect in greater detail we conducted a Pearson correlation to examine the relationship between the signal in a voxel and its variability. There was a significant positive correlation between the mean voxel value and its variability over time ($r = 0.54$, $p < 0.001$). A scatterplot of mean voxel signal intensity against voxel standard deviation is presented to the right.



The Bonferroni Method

To test for the difference between the *ith* and *jth* treatments, it is common to use the two-sample t test. The two-sample t statistic is

$$t_{ij} = \frac{\bar{y_j}. - \bar{y_i}.}{\hat{\sigma}\sqrt{1/n_j + 1/n_i}},$$

where $\bar{y_j}.$ is the average of the n_i observations for treatment j and $\hat{\sigma}$ is $\sqrt{MS_E}$ from the ANOVA table.

Treatments i and j are declared significantly different at level α if

$$|t_{ij}| > t_{N-k,\alpha/2},$$

where $t_{N-k,\alpha/2}$ is the upper $\alpha/2$ percentile of a t_{N-k}.

The Bonferroni Method

The total number of pairs of treatment means that can be tested is

$$c = \binom{k}{2} = \frac{k(k-1)}{2}.$$

The Bonferroni method for testing $H_0 : \mu_i = \mu_j$ vs. $H_0 : \mu_i \neq \mu_j$ rejects H_0 at level α if

$$|t_{ij}| > t_{N-k,\alpha/2c},$$

where c denotes the number of pairs being tested.

The Bonferroni Method

In R the function `pairwise.t.test()` can be used to compute Bonferroni adjusted p-values.

This is illustrated below for the blood coagualtion study.

```
pairwise.t.test(tab0401$y,tab0401$diets,p.adjust.method = "bonferroni")
```

```
## 
##  Pairwise comparisons using t tests with pooled SD 
## 
## data:  tab0401$y and tab0401$diets 
## 
##   A       B       C      
## B 0.00934 -       -      
## C 0.00031 0.95266 -      
## D 1.00000 0.00934 0.00031
## 
## P value adjustment method: bonferroni
```

There are signifciant differences at the 5% level between diets A and B, A and C, B and D, and C and D using the Bonferroni method.

The Bonferroni Method

For comparison the unadjusted p-values are also calculated.

```
pairwise.t.test(tab0401$y,tab0401$diets,p.adjust.method = "none")
```

```
##
##  Pairwise comparisons using t tests with pooled SD
##
## data:  tab0401$y and tab0401$diets
##
##   A       B      C
## B 0.0016  -      -
## C 5.2e-05 0.1588 -
## D 1.0000  0.0016 5.2e-05
##
## P value adjustment method: none
```

The significant differences are the same using the unadjusted p-values but the p-values are larger then the p-values adjusted using the Bonferroni method.

The Bonferroni Method

A $100(1-\alpha)\%$ simultaneous confidence interval for c pairs $\mu_i - \mu_j$ is

$$\bar{y_j}. - \bar{y_i}. \pm t_{N-k,\alpha/2c}\hat{\sigma}\sqrt{1/n_j + 1/n_i}.$$

After identifying which pairs are different, the confidence interval quantifies the range of plausible values for the differences.

The Bonferroni Method - coagulation study

The treatment means can be obtained from the table below.

	A	B	C	D
	60	65	71	62
	63	66	66	60
	59	67	68	61
	63	63	68	64
	62	64	67	63
	59	71	68	56
Treatment Average	61	66	68	61
Grand Average	64	64	64	64
Difference	-3	2	4	-3

The Bonferroni Method - coagulation study

$\hat{\sigma} = \sqrt{MS_E}$ can be obtained from the ANOVA table.

```r
anova(lm(y~diets,data=tab0401))
```

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## diets      3    228    76.0  13.571 4.658e-05 ***
## Residuals 20    112     5.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The upper $.05/(2 \cdot 6) = 0.004$ percentile of the t_{24-4} can be obtained with the t quantile function in R `qt()`.

```r
qt(p = 1-0.004,df = 20)
```

```
## [1] 2.945349
```

The Bonferroni Method - coagulation study

Plugging in these values to the confidence interval formula we can obtain a Bonferroni adjusted 95% confidence interval for $\mu_B - \mu_A$:

$$66 - 61 \pm 2.95\sqrt{5.6}\sqrt{1/6 + 1/6}$$

The lower and upper limits can be calculated in R.

```r
66-61 - qt(p = 1-0.004,df = 20)*sqrt(5.6)*sqrt(1/6+1/6) # lower limit
```

```
## [1] 0.9758869
```

```r
66-61 + qt(p = 1-0.004,df = 20)*sqrt(5.6)*sqrt(1/6+1/6) # upper limit
```

```
## [1] 9.024113
```

The 95% confidence interval for $\mu_B - \mu_A$ is (0.98, 9.02).

The Tukey Method

- The only difference between the Tukey and Bonferroni methods is in the choice of the critical value.
- Treatments i and j are declared significantly different at level α if

$$|t_{ij}| > \frac{1}{\sqrt{2}} q_{k,N-k,\alpha},$$

- t_{ij} is the observed value of the two-sample t-statistic
- $q_{k,N-k,\alpha}$ is the upper α percentile of the Studentized range distribution with parameters k and $N-k$ degrees of freedom.
- The CDF and inverse CDF of the Studentized Range Distribution is available in R via the functions `ptukey()` and `qtukey()` respectively.

The Tukey Method

A $100(1-\alpha)\%$ simultaneous confidence interval for c pairs $\mu_i - \mu_j$ is

$$\bar{y_j}. - \bar{y_i}. \pm \frac{1}{\sqrt{2}} q_{k,N-k,\alpha} \hat{\sigma} \sqrt{1/n_j + 1/n_i}.$$

The Bonferroni method is more conservative than Tukey's method. In other words, the simutaneous confidence intervals based on the Tukey method are shorter.

The Tukey Method

- In the coagualtion study $N = 24, k = 4$ so the 5% critical value of the Studentized range distribution is obtained using the the inverse CDF function `qtukey()` for this distribution.
- The argument `lower.tail=FALSE` is used so we obtain the upper percentile of the distribution (i.e., the value of x such that $P(X > x) = 0.05$).

```r
qtukey(p = .05,nmeans = 4,df = 20,lower.tail = FALSE)
```

```
## [1] 3.958293
```

The Tukey Method

- Let's obtain the Tukey p-value and confidence interval for $\mu_B - \mu_A$.
- The observed value of the test statistic is

$$q^{obs} = \sqrt{2}|t_{AB}|,$$

where

$$t_{AB} = \frac{\bar{y_{A\cdot}} - \bar{y_{B\cdot}}}{\hat{\sigma}\sqrt{1/n_A + 1/n_B}}.$$

```r
(sqrt(2)*(66-61))/(sqrt(5.6)*sqrt(1/6+1/6))
```

```
## [1] 5.175492
```

The Tukey Method

The p-value

$$P\left(q_{4,20} > q^{obs}\right)$$

is then obtained using the CDF of the Studentized range distribution

```
1-ptukey(q = sqrt(2)*5/sqrt(2*5.6/6),nmeans = 4,df = 20)
```

```
## [1] 0.007797788
```

The Tukey Method

The 95% limits of the Tukey confidence interval for $\mu_B - \mu_A$ is

```
tuk.crit <- qtukey(p=.05,nmeans=4,df=20,lower.tail=FALSE)
#lower limit
round(5-(1/sqrt(2))*tuk.crit*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 1.18
```

```
#upper limit
round(5+(1/sqrt(2))*tuk.crit*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 8.82
```

The Tukey Method

The width of the Tukey confidence interval for $\mu_B - \mu_A$ is

```r
round((1/sqrt(2))*tuk.crit*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 3.82
```

The width of Bonferroni $\mu_B - \mu_A$ is

```r
round(qt(p = 1-0.004,df = 20)*sqrt(5.6)*sqrt(1/6+1/6),2)
```

```
## [1] 4.02
```

The Tukey Method

- This shows that the Tukey confidence interval is shorter than Bonferroni confidence intervals.
- The command `TukeyHSD()` can be used to obtain all the Tukey confidence intervals and p-values for an ANOVA.

The Tukey Method

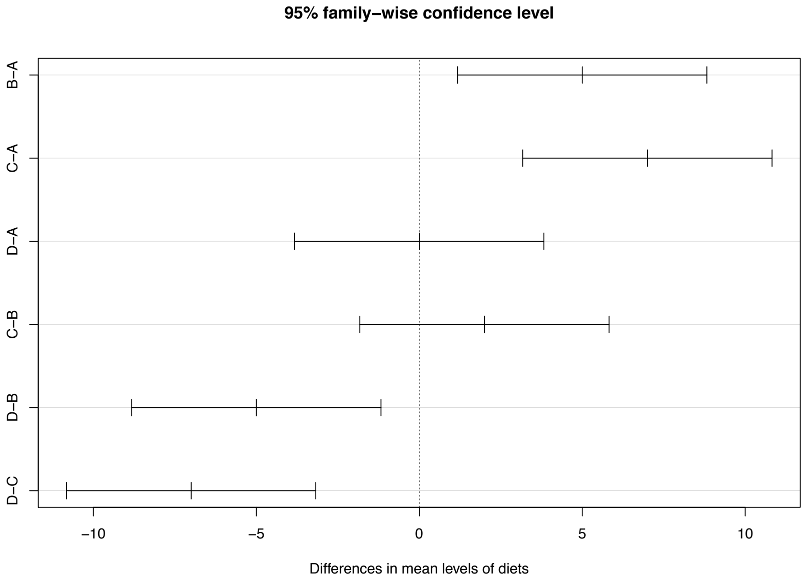
```
TukeyHSD(aov(y~diets,data=tab0401))
```

```
round(TukeyHSD(aov(y~diets,data=tab0401))$diets,2)
```

```
##     diff    lwr   upr p adj
## B-A    5   1.18  8.82  0.01
## C-A    7   3.18 10.82  0.00
## D-A    0  -3.82  3.82  1.00
## C-B    2  -1.82  5.82  0.48
## D-B   -5  -8.82 -1.18  0.01
## D-C   -7 -10.82 -3.18  0.00
```

The Tukey Method

```
plot(TukeyHSD(aov(y~diets,data=tab0401)))
```



Sample size for ANOVA - Designing a study to compare more than two treatments

- Consider the hypothesis that k means are equal vs. the alternative that at least two differ.
- What is the probability that the test rejects if at least two means differ?
- Power $= 1 - P(\text{Type II error})$ is this probability.

Sample size for ANOVA - Designing a study to compare more than two treatments

The null and alternative hypotheses are:

$$H_0 : \mu_1 = \mu_2 = \cdots = \mu_k \, \mathsf{vs.} \, H_1 : \mu_i \neq \mu_j.$$

The test rejects at level α if

$$MS_{Treat}/MS_E \geq F_{k-1,N-K,\alpha}.$$

The power of the test is

$$1 - \beta = P\left(MS_{Treat}/MS_E \geq F_{k-1,N-K,\alpha}\right),$$

when H_0 is false.

Sample size for ANOVA - Designing a study to compare more than two treatments

When H_0 is false it can be shown that:

- MS_{Treat}/σ^2 has a non-central Chi-square distribution with $k-1$ degrees of freedom and non-centrality parameter δ.
- MS_{Treat}/MS_E has a non-central F distribution with the numerator and denominator degrees of freedom $k-1$ and $N-k$ respectively, and non-centrality parameter

$$\delta = \frac{\sum_{i=1}^{k} n_i \left(\mu_i - \bar{\mu}\right)^2}{\sigma^2},$$

where n_i is the number of observations in group i, $\bar{\mu} = \sum_{i=1}^{k} \mu_i/k$, and σ^2 is the within group error variance .

This is dentoted by $F_{k-1,N-k}(\delta)$.

Direct calculation of Power

- The power of the test is

$$P\left(F_{k-1,N-k}(\delta) > F_{k-1,N-K,\alpha}\right).$$

- The power is an increasing function δ
- The power depends on the true values of the treatment means μ_i, the error variance σ^2, and sample size n_i.
- If the experimentor has some prior idea about the treament means and error variance, and the sample size (number of replications) the formula above will calculate the power of the test.

Blood coagulation example - sample size

Suppose that an investigator would like to replicate the blood coagulation study with only 3 animals per diet. In this case $k = 4, n_i = 3$. The treatment means from the initial study are:

Diet	A	B	C	D
Average	61	66	68	61

```r
anova(lm.diets)
```

```
## Analysis of Variance Table
##
## Response: y
##           Df Sum Sq Mean Sq F value    Pr(>F)
## diets      3    228    76.0  13.571 4.658e-05 ***
## Residuals 20    112     5.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Blood coagulation example - sample size

- $\mu_1 = 61$, $\mu_2 = 66$, $\mu_3 = 68$, $\mu_4 = 61$.
- The error variance σ^2 was estimated as $MS_E = 5.6$.
- Assuming that the estimated values are the true values of the parameters, the non-centrality parameter of the F distribution is

$$\delta = 3 \times \left((61-64)^2 + (66-64)^2 + (68-64)^2 + (61-64)^2 \right) /5.6 = 20.35714$$

Blood coagulation example - sample size

If we choose $\alpha = 0.05$ as the significance level then $F_{3,20,0.05} = 3.0983912$. The power of the test is then

$$P\left(F_{3,20}(20.36) > 3.10\right) = 0.94.$$

This was calculated using the CDF for the F distribution in R `pf()`.

```r
1-pf(q = 3.10,df1 = 3,df2 = 20,ncp = 20.36)
```

```
## [1] 0.9435208
```

Calculating power and sample size using the pwr library

- There are several libraries in R which can calculate power and sample size for statistical tests. The library `pwr()` has a function
- `pwr.anova.test(k = NULL, n = NULL, f = NULL, sig.level = 0.05, power = NULL)`

for computing power and sample size.

- k Number of groups
- n Number of observations (per group)
- f Effect size
- The effect size is the square root of the non-centrality parameter of the non-central F distribution.

$$f = \sqrt{\frac{\sum_{i=1}^{k} n_i \left(\mu_i - \bar{\mu}\right)^2}{\sigma^2}}.$$

where n_i is the number of observations in group i, $\bar{\mu} = \sum_{i=1}^{k} \mu_i / k$, and σ^2 is the within group error variance.

Calculating power and sample size using the pwr library

In the previous example $\delta = 20.35714$ so $f = \sqrt{20.35714} = 4.5118887$.

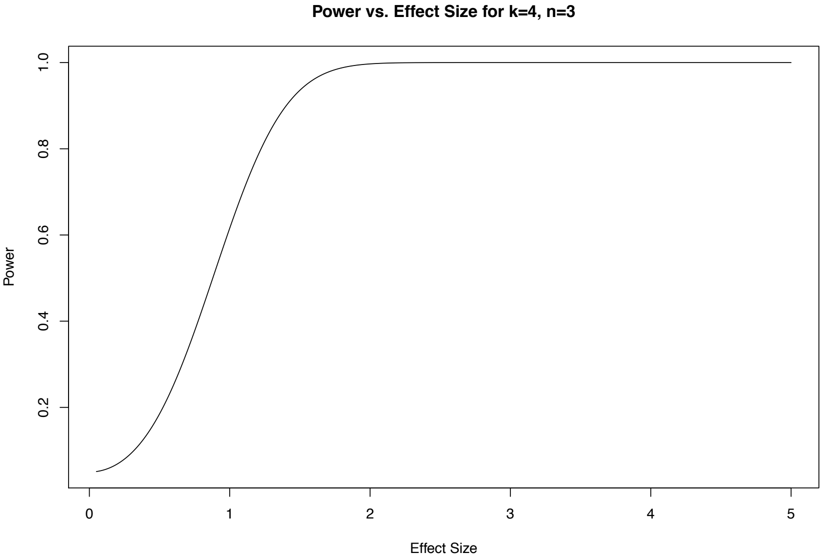
```
library(pwr)
```

```
## Warning: package 'pwr' was built under R version 3.2.5
```

```
pwr.anova.test(k = 4,n = 3,f = 4.5)
```

```
##
##      Balanced one-way analysis of variance power calculation
##
##               k = 4
##               n = 3
##               f = 4.5
##       sig.level = 0.05
##           power = 1
##
## NOTE: n is number in each group
```

Calculating power and sample size using the pwr library



Calculating power using simulation

The general procedure for simulating power is:

1. Use the underlying model to generate random data with (a) specified sample sizes, (b) parameter values that one is trying to detect with the hypothesis test, and (c) nuisance parameters such as variances.
2. Run the estimation program (e.g., `t.test()`,`lm()`) on these randomly generated data.
3. Calculate the test statistic and p-value.
4. Do Steps 1–3 many times, say, N, and save the p-values. The estimated power for a level alpha test is the proportion of observations (out of N) for which the p-value is less than alpha.

Calculating power using simulation

One of the advantages of calculating power via simulation is that we can investigate what happens to power if, say, some of the assumptions behind one-way ANOVA are violated.

Calculating power using simulation - R program

```r
#Simulate power of ANOVA for three groups

NSIM <- 1000 # number of simulations
res <- numeric(NSIM) # store p-values in res

mu1 <- 2; mu2 <- 2.5;mu3 <- 2  # true mean values of treatment groups
sigma1 <- 1; sigma2 <- 1; sigma3 <- 1 #variances in each group
n1 <- 40; n2 <- 40; n3 <- 40 #sample size in each group

for (i in 1:NSIM) # do the calculations below N times
  {
# generate sample of size n1 from N(mu1,sigma1^2)
y1 <- rnorm(n = n1,mean = mu1,sd = sigma1)
# generate sample of size n2 from N(mu2,sigma2^2)
y2 <- rnorm(n = n2,mean = mu2,sd = sigma2)
# generate sample of size n3 from N(mu3,sigma3^2)
y3 <- rnorm(n = n3,mean = mu3,sd = sigma3)
y <- c(y1,y2,y3) # store all the values from the groups
# generate the treatment assignment for each group
trt <- as.factor(c(rep(1,n1),rep(2,n2),rep(3,n3)))
m <- lm(y~trt) # calculate the ANOVA
res[i] <- anova(m)[1,5] # p-value of F test
}
sum(res<=0.05)/NSIM # calculate p-value
```

```
## [1] 0.643
```