ANOVA tables

- More than two treatments/groups
- Does not work for proportions $\hat{p}$ to find π

What ANOVA hypothesis tests tells us...

- Determines if at least one of the treatments are different from one another
    - One of the populations means μ_i is different from the rest

$$H_o: \mu_1 = \cdots = \mu_t$$

H_A: at least one of the means is different from the rest

- Give statement based on $p < \alpha$
- Does not tell us which one differs, only tells us if there is or is not a difference in means; use one of the following (see their page for more)
    - Tukey HSD
    - Bon Ferroni's
    - Wilcox Rank Sum

t	Number of levels/treatments (including placebo)
n_i	Number of observations in the ith treatment
N	Total sample size/number of treatments
y_{ij}	jth observation in the ith group
$\bar{y}_i$	Sample mean of the ith group
s_i	Sample standard deviation of the ith group
$\bar{y} = \frac{1}{N}\sum_i \sum_j y_{ij}$	Grand mean
s	Total sample standard deviation of N
$F = \frac{MS_{treatment}}{MS_{error}}$	Critical point
p-value	`pf(F,df.treatment,df.error,lower.tail=FALSE)`

General layout…

Source	df (degrees of freedom)	SS (sum of squares)	MS (mean of squares)	F	p-value (probability)
Treatment/Between					
Error/Within					
Total					

How to find each by hand…

TREATMENT/BETWEEN VARIABILITY

$$df_{treatment} = t - 1$$

$$SS_T = \sum_{i}^{t}\sum_{j}^{n_i}(\bar{y}_i - \bar{y})^2 = \sum_{i}^{t} n_i(\bar{y}_i - \bar{y})^2$$

$$MS_{treatment} = \left(\frac{SS}{df}\right)_{treatment}$$

ERROR/WITHIN VARIABILITY

$$df_{error} = N - t$$

$$SS_{error} = \sum_{i}^{t}(n_i - 1)s_i^2$$

$$MS_{error} = \left(\frac{SS}{df}\right)_{error}$$

TOTAL

$$df_{total} = N - 1$$

$$SS_{total} = (N - 1)s^2 = SS_{treatment} + SS_{error}$$

$$Var = s^2 = \left(\frac{SS}{df}\right)_{total}$$

Tukey's Honestly Significant Difference (HSD)

- Applied after ANOVA test

Criteria for Tukeys…

(1) ANOVA must be 100% pure assumptions
    a. Independent observations → iid
    b. Normally distributed
(2) *****Variances *must* be equal*****
(3) Works best with equal sample sizes (can still be used without it)

What it does…

- Determines which pairs of means differ from ANOVA tables
- Make a confidence interval of all different combinations of the tests

$$CI = \left(\bar{x}_i - \bar{x}_j\right) \pm q\sqrt{\frac{MS_{error}}{2}\left(\frac{1}{n_i} + \frac{1}{n_j}\right)}$$

```
q=qtukey(CL,nmeans=num_of_groups,df=df.error)
```

How to determine if one mean is significantly different from another…

- Compare two at a time
- If $0 \in CI \Longrightarrow$ not significantly different
- If $0 \notin CI \Longrightarrow$ significantly different

Assigning letters…

- Create a table
- If they are not significantly different → same letter
- If they are → different
- Letters have no actual value, just show the likeness

Bonferroni Adjustment

- Similar to Tukey HSD but it adjusts p-values
- For each pair of group means α^* stays the same
- More conservative with the picks → reduces change of type I error
  - Makes CI wider, not smaller

$\alpha^* = \frac{\alpha}{m}$	Critical point
α	Significance level
m	Number of tests

How to do it...

(1) Obtain α^*
(2) For each pair of group means, perform t-test. If $p < \alpha^* \Longrightarrow$ significantly different

```
t.crit=qt((1-alpha.crit)/2)
```

```
p=1-pt(t.crit)
```

$$CI = (\bar{x}_i - \bar{x}_j) \pm t^* \sqrt{\frac{MS_{error}}{2}\left(\frac{1}{n_i} + \frac{1}{n_j}\right)}$$

Wilcox-Rank-Sum w/ adjustment...

```
pairwise.wilcox.test(obs, treatment,
p.adjust.method="bonferroni")
```

Kruskal-Wallis Test

- Alternative for an ANOVA test
- Use when assumptions of ANOVA are not met
    - Normality cannot be assumed
- Compares medians instead of means

$$\chi^2 = \frac{12}{N(N+1)} \sum_{i=1}^{k} \left[\frac{R_i^2}{n_i} - 3(N+1) \right]$$

N	Total number of observations
k	Number of groups
n_i	Number of observations in group i
R_i	Sum of ranks for group i
χ^2	Test statistic

```
data=c(…) #vector of all raw dat

group=factor(…)

kruskal.test(response ~ group, data = your_data)
```

Linear Regression

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$

$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$	Intercept
$\hat{\beta}_1 = r\frac{s_y}{s_x}$	Slope
r	Pearson's correlation coefficient
$\epsilon = y_i - \hat{y}$	Residual
$SS = \sum (y_i - \hat{y})^2$	Sum of squares
$\hat{\sigma}_\epsilon = \sqrt{\frac{\sum(y_i - \hat{y})^2}{n-2}} = \sqrt{MS_{error}}$	Standard deviation
$MS_{error} = \hat{\sigma}_\epsilon^2 = \frac{\sum(y_i - \hat{y})^2}{n-2} = \frac{\sum\epsilon^2}{n-2}$	Predicted (residual) variance Mean square error

$$SE = \hat{\sigma}_\epsilon \sqrt{\frac{1}{n} + \frac{(x^* - \bar{x})^2}{\sum(x_i - \bar{x})^2}}$$

```
model = lm(y ~ x1, data = your_data)

summary(model)

predict(model, newdata = new_df)

predict(model, newdata = new_df, interval = "confidence")

predict(model, newdata = new_df, interval = "prediction")
```

Confidence interval for $\bar{y}$ at a given x^*...

$$CI = \hat{y}(x^*) \pm t_{\alpha/2, df=n-2} \hat{\sigma}_\epsilon \sqrt{\frac{1}{n} + \frac{(x^* - \bar{x})^2}{\sum(x_i - \bar{x})^2}}$$

Prediction interval for the next y at a given x^*...

$$PI = \hat{y}(x^*) \pm t_{\alpha/2, df=n-2} \hat{\sigma}_\epsilon \sqrt{1 + \frac{1}{n} + \frac{(x^* - \bar{x})^2}{\sum(x_i - \bar{x})^2}}$$

Making inferences about the true slope of the line...

$$t^* = \frac{\hat{\beta}_1 - \hat{\beta}_{1,H_o}}{SE}$$

$$SE = \frac{\sigma_\epsilon}{\sqrt{\sum(x_i - \bar{x})^2}}$$

Chi-squared Test

- Tells us if the quantitative data set has any values that stray away from the mean
- Only two data sets → see if they're different from one another
- Used when there is a certain standard; can use linear regression with this
    - Compare new data set with old one
- Two types: (1) goodness of fit test and (2) test for independence

(1) goodness of fit test

- Checks if distribution of a categorical variable matches the expected distribution
- Checking if number of times things happened in different categories what was expected, or different
- Useful to compare observed and expected counts under a specific hypothesis
- If $p < 0.05$ reject H_o based on `chisq.test(…)`

(2) test for independence

Ex)

```
data <- matrix(c(20,15,15,10,25,15), nrow = 2, byrow =
TRUE)
colnames(data) <- c("Red", "Blue", "Green")
rownames(data) <- c("Male", "Female")
table <- as.table(data)

# Run the chi-square test for independence
chisq.test(table)
```

$$\chi^2 = \sum \frac{(Obs_i + Exp_i)^2}{Exp_i}$$

```
prob=pchisq(chi.sq,df=k-1)
```