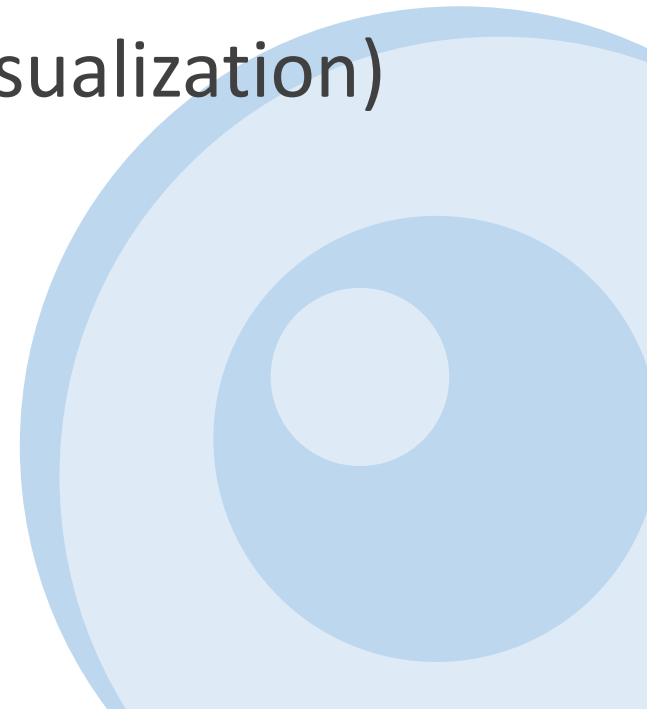


PSYC 193R:

Data Analysis & Visualization with R

Lecture 12: One-way ANOVA (post hoc tests / visualization)

Jonas S.-H. Lau [silau -et- ucsd.edu]
psyc193r.ucsd.edu



Tricks in the Fitting Room

It is an open secret that retailers play "tricks" on shoppers that would lead to more sales. The same is true for grocery stores, electronics shops, clothing or online retailers.

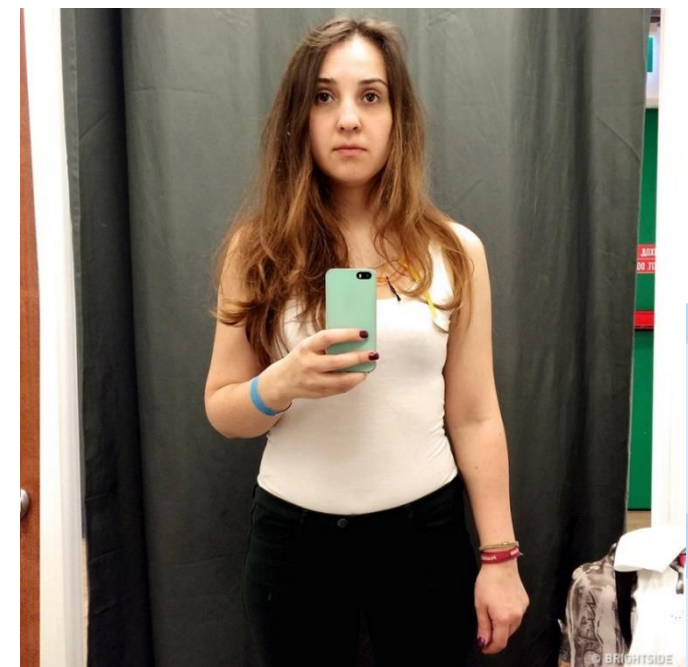
Regular Indoor Lighting



Harsh Overhead Lighting



Soft Vertical Lighting

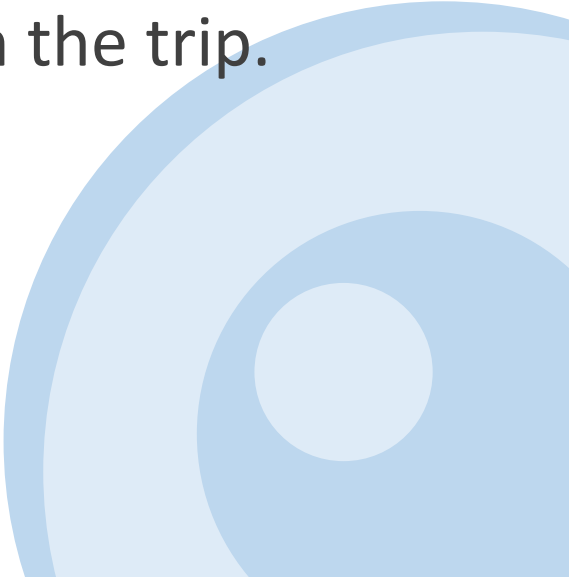


Tricks in the Fitting Room

A clothing retailer was in the process of redesigning their UTC store. They wanted to know if different kinds of lighting affect sales.

They installed **three kinds of lightings in the 3 different fitting rooms**. For a weekend, shoppers were randomly assigned into one of the three rooms when they requested to try something on. After they were done fitting, the clothes were marked as 0, 1, or 2, representing the rooms they were in.

The cashiers registered the **amount of money they spent** on the trip.



One-way ANOVA

One-way = One Factor = 1 Independent Variable
(in this case, we have 1 IV and 3 levels within the IV)

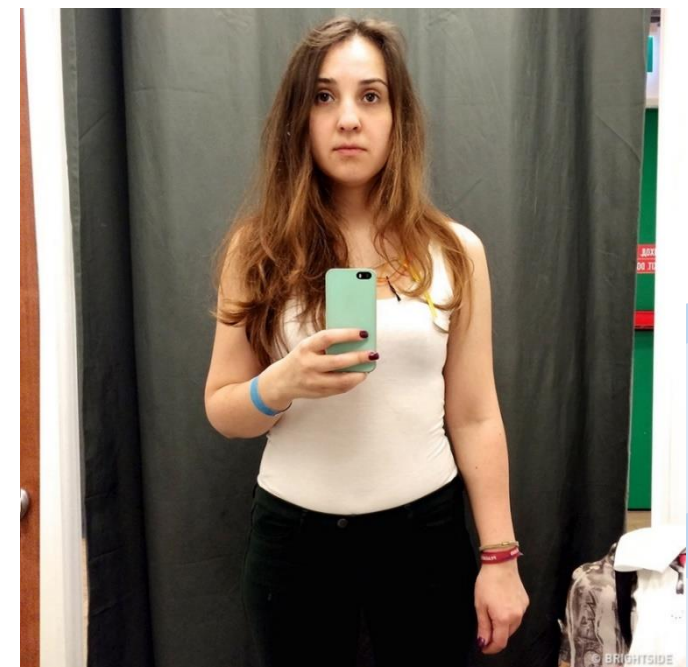
Regular Indoor Lighting



Harsh Overhead Lighting



Soft Vertical Lighting



One-way ANOVA

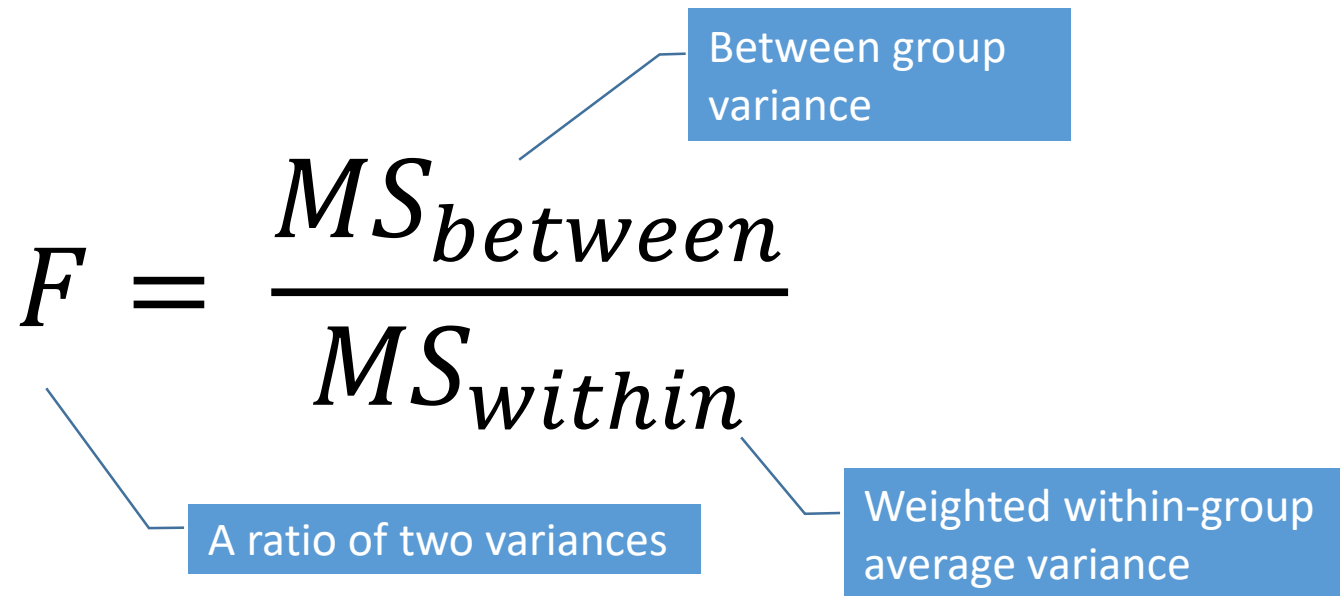
One-way = One Factor = 1 Independent Variable
(in this case, we have 1 IV and 3 levels within the IV)

$$F = \frac{MS_{between}}{MS_{within}}$$

Between group variance

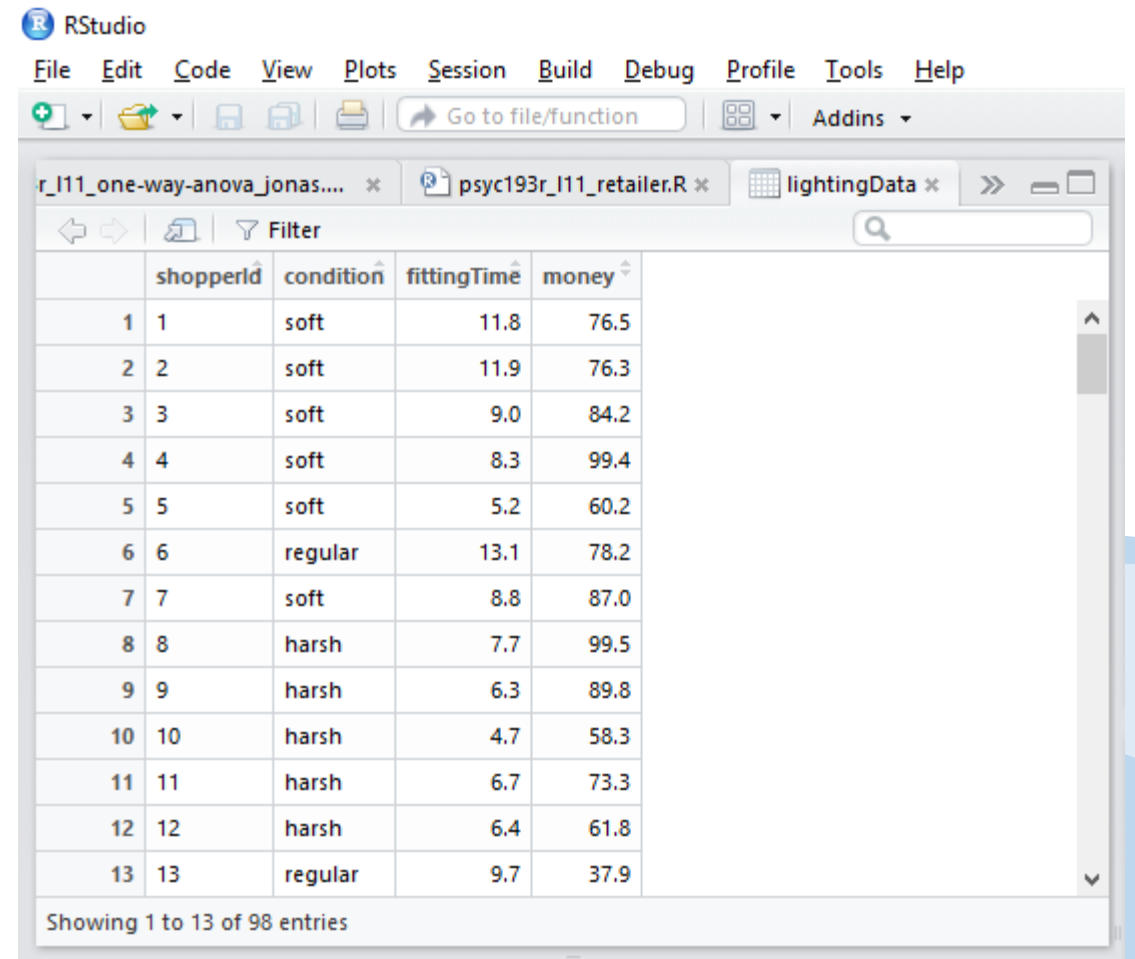
A ratio of two variances

Weighted within-group average variance

The diagram shows the F-ratio formula $F = \frac{MS_{between}}{MS_{within}}$. Three blue callout boxes with white text provide context: 'Between group variance' points to the numerator $MS_{between}$; 'A ratio of two variances' points to the entire fraction; and 'Weighted within-group average variance' points to the denominator MS_{within} . The labels are connected to the formula by thin blue lines.

Data frame

- “Long” data frame
- One subject per row
- You may need to process the data before you can create an `aov()` object



The screenshot shows the RStudio interface with the 'lightingData' data frame loaded in the Environment pane. The data frame has 13 rows and 5 columns: 'shopperId', 'condition', 'fittingTime', and 'money'. The 'shopperId' column contains values from 1 to 13, 'condition' contains 'soft' or 'harsh', 'fittingTime' contains numerical values, and 'money' contains numerical values. The status bar at the bottom indicates 'Showing 1 to 13 of 98 entries'.

	shopperId	condition	fittingTime	money
1	1	soft	11.8	76.5
2	2	soft	11.9	76.3
3	3	soft	9.0	84.2
4	4	soft	8.3	99.4
5	5	soft	5.2	60.2
6	6	regular	13.1	78.2
7	7	soft	8.8	87.0
8	8	harsh	7.7	99.5
9	9	harsh	6.3	89.8
10	10	harsh	4.7	58.3
11	11	harsh	6.7	73.3
12	12	harsh	6.4	61.8
13	13	regular	9.7	37.9

Hypothesis Testing – Step-by-step

- Step 0: Gather all the information you need

- Check assumptions
- Data cleaning

- Step 1: Set up the hypotheses

- H_0 (null hypothesis)
- H_1 (Research hypothesis)

- Step 2: Create an `aov()` object

- Step 3: Use `summary()` to obtain sample F score and p value

- Step 4: Decision & Conclusion

(compare p against α)

check: $df_{\text{between}} = k - 1$; $df_{\text{within}} = N - k$

Assumptions:

- Outliers
- Equal variances between conditions
- kurtosis of population distribution ? $n's \geq 30$

$H_0: \mu_{\text{regular}} = \mu_{\text{harsh}} = \mu_{\text{soft}}$

H_1 : at least one of the conditions leads to higher/ lower sales compared to the others

`...aov = aov()`

See `summary()` output

Report sample means, standard deviations, sample sizes,

$F(df_{\text{between}}, df_{\text{within}}) = \text{---} , p = \text{---}$
decision/ conclusion

Effect Size (% variance explained)

$$\eta^2 = \frac{SS_{between}}{SS_{total}}$$

*Squared deviation
between the group
means*

*Squared deviation
between the grand
mean and each score*

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
condition	2	6973	3486	7.767	0.000751 ***
Residuals	95	42641	449		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

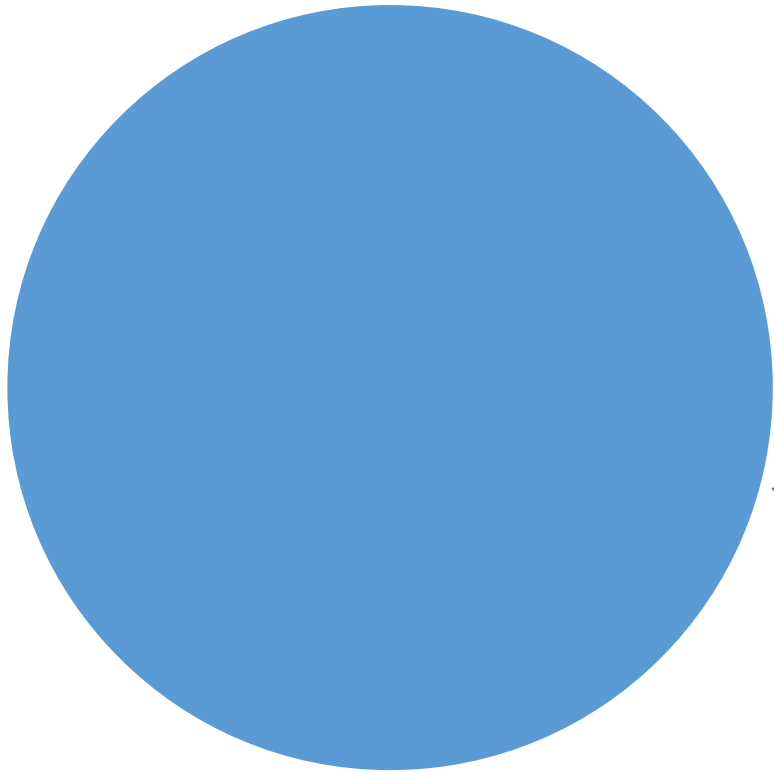
Effect Size (% variance explained)

$$\eta^2 = \frac{SS_{between}}{SS_{total}} = \frac{6973}{(42641 + 6973)} = 14.1\%$$

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
condition	2	6973	3486	7.767	0.000751 ***
Residuals	95	42641	449		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

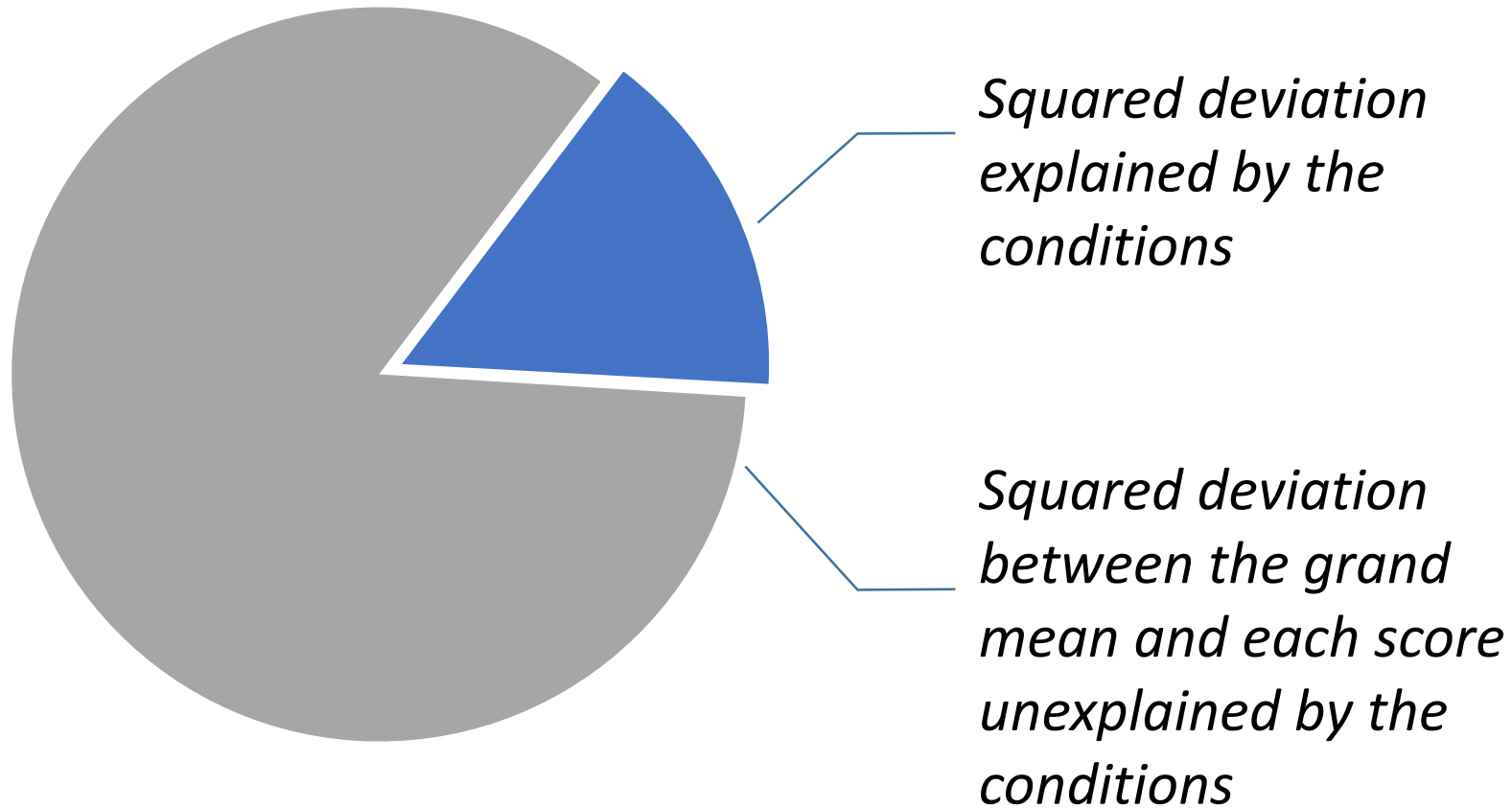
Effect Size (% variance explained)



*Squared deviation
between the grand
mean and each score*



Effect Size (% variance explained)



Pair-wise comparisons

- We want to know which condition brings the most sales
- Knowing “at least one of the conditions is different from the others” (without knowing *which*) is pretty useless
- Only meaningful when ANOVA test is significant

Regular Indoor Lighting



Harsh Overhead Lighting



Soft Vertical Lighting



Post-hoc tests (*post hoc*: occurring or done after the event)

- Recall: Each time you conduct a test → Test-wise Type I error (α_{test}) (concluding a difference between conditions when there is none)
- We want to control for the overall Type I error (family-wise Type I error)



Solution 1: Bonferroni correction

1. Count the number of pair-wise t-tests (n_{test}) you need
2. Control for the family-wise Type I error you can tolerate ($\alpha_{\text{family}} / n_{\text{test}}$)
3. Perform pair-wise comparisons using the new α_{test} level
4. Conclude a significant pairwise comparison only when $p < \alpha_{\text{test}}$

Issues with Bonferroni correction:

- Overly conservative:
The ability to reject H_0 when it is actually false is low (*i.e.*, power is low)



Solution 2: Tukey Honest Significance Test

- Comparing HSD score with one in Tukey's critical value table

$$HSD = \frac{M_i - M_j}{\sqrt{\frac{MS_{within}}{n_h}}}$$

```
TukeyHSD( aov_object, "name_of_iv" )
```

Statistical Power

- Probability of rejecting the H_0 when it is false.
- You can do something like:

```
library(pwr)  
pwr.anova.test(f = .14, k = 3, n = 30, power = NULL)
```

Effect size

Number of conditions

*Number of participants
per condition*



Statistical Power

- Probability of rejecting the H_0 when it is false.
- You can do something like:

```
library(pwr)  
pwr.anova.test(f = .14, k = 3, n = 30, power = NULL)
```

- But it's quite meaningless -- A significant F-test only tells you there are differences between *at least one condition* and the others
- Instead, you may want to focus on a pairwise comparison

```
library(pwr)  
pwr.t.test(d = 0.5, n = NULL, power = 0.8)
```

Practical Research Tips:

Focus on pairwise comparisons instead

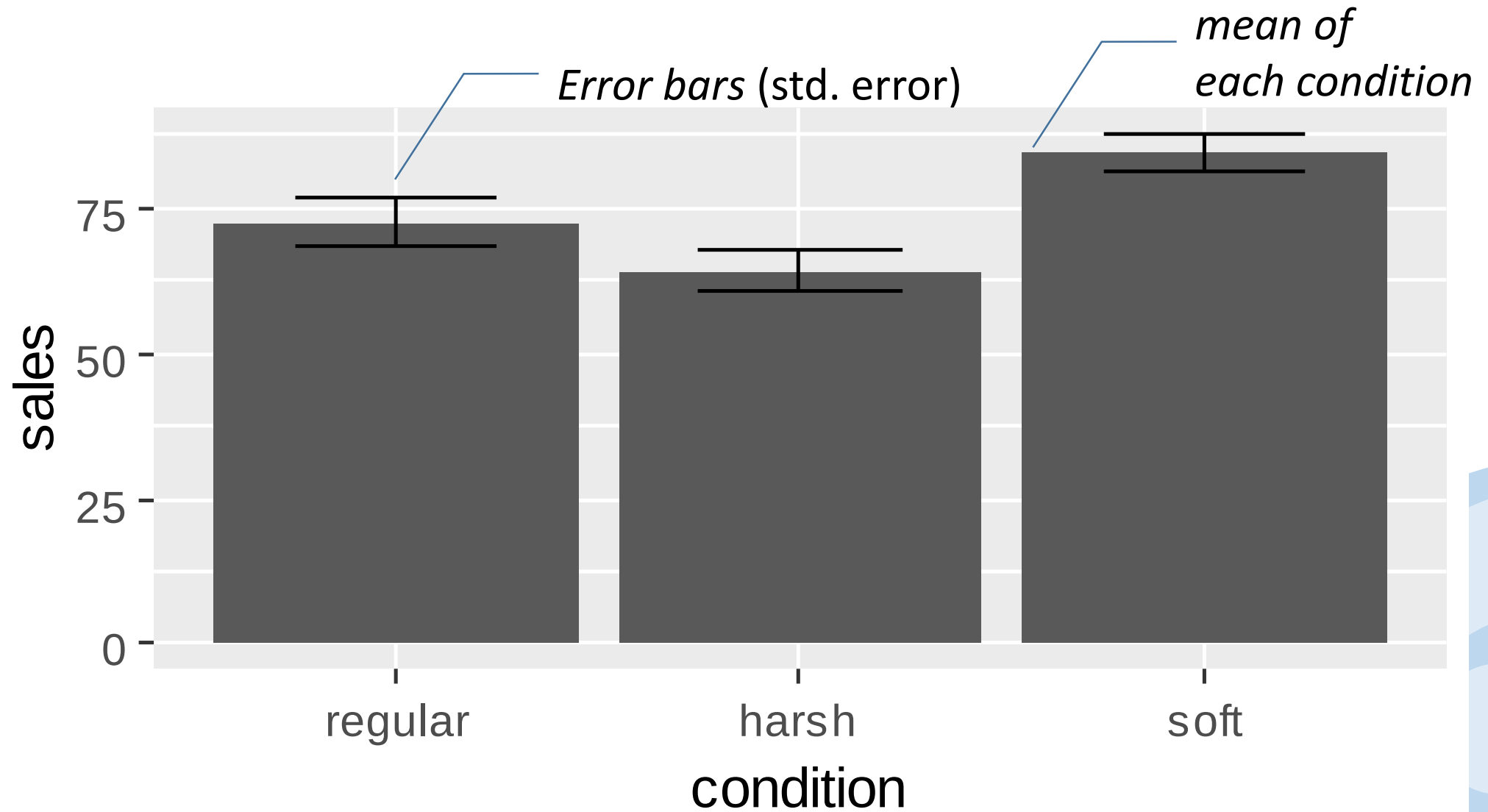
One-way ANOVA: Research tips

- Only do post hoc tests when you have a significant omnibus-F result
- If you are testing for a new intervention, it is good to have:
 - A placebo condition
 - A current treatment condition
 - A proposed treatment condition
- The $\alpha_{\text{test-wise}}$ is spread thin the more conditions you test
 - So limit the number of conditions

Practical Research Tips:

Limit the number of conditions for a greater statistical power

Visualization for *informal* comparisons



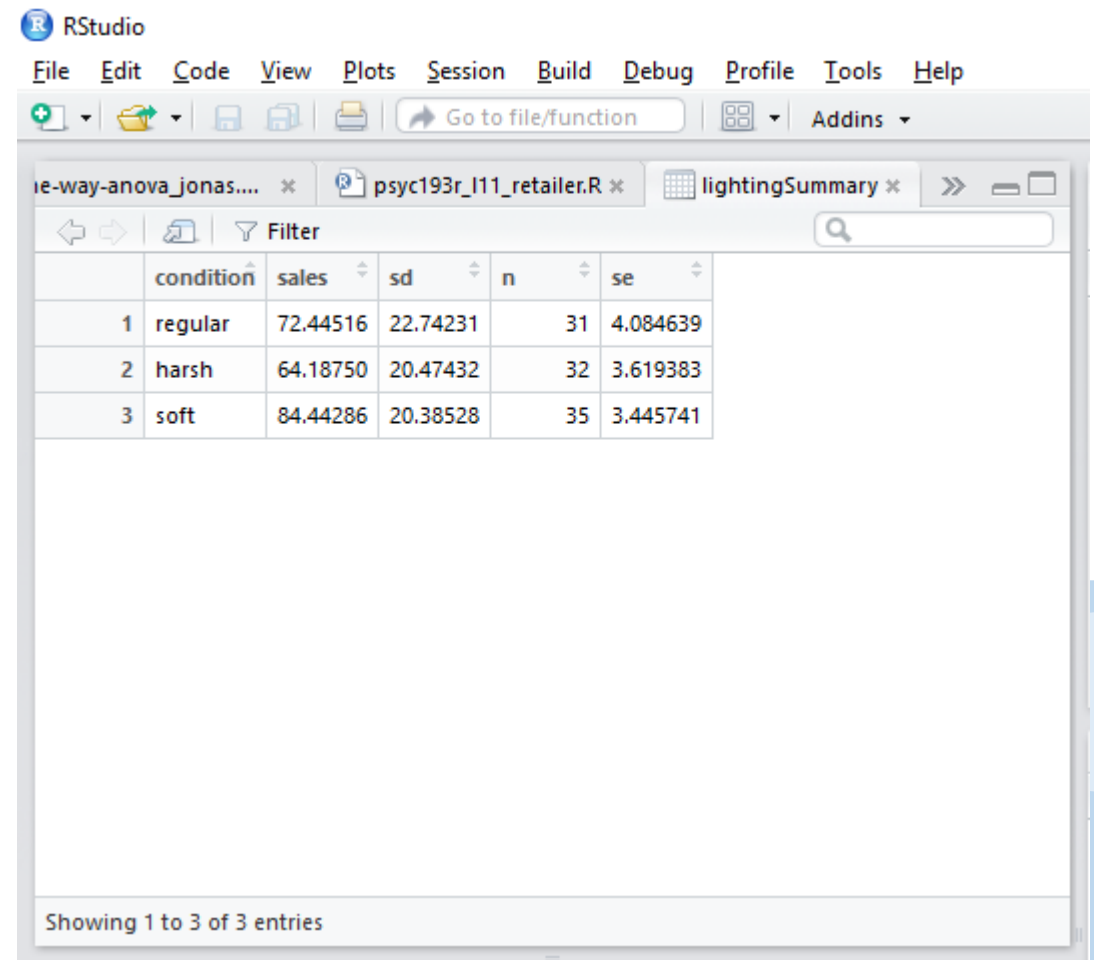
Error bars denotes standard error of the means in the condition

Data frame for visualization

LightingSummary: One column for each of variable

- condition
- Mean (sales)
- Estimated standard deviation of the target population (sd)
- Number of participants in each condition (n)
- Standard error of the sampling distribution (se)

$$se = sd / \sqrt{n}$$



RStudio

File Edit Code View Plots Session Build Debug Profile Tools Help

Go to file/function

psyc193r_l11_retailer.R x lightingSummary x

Filter

	condition	sales	sd	n	se
1	regular	72.44516	22.74231	31	4.084639
2	harsh	64.18750	20.47432	32	3.619383
3	soft	84.44286	20.38528	35	3.445741

Showing 1 to 3 of 3 entries

Data frame for visualization: Summary Statistics

```
# data frame for different conditions
# find the means
lightingSummary = aggregate( NA ~ NA,
  data = lightingData, FUN = NA)
names(lightingSummary)[2] = "NA"

# find the estimated sd
lightingSummary.sd = aggregate( NA ~ NA,
  data = lightingData, FUN = NA)
names(lightingSummary.sd)[2] = "NA"

# find the number of participants in each condition
lightingSummary.n = aggregate( NA ~ NA,
  data = lightingData, FUN = NA)
names(lightingSummary.n)[2] = "NA"
```

Data frame for visualization (3)

```
# combine the data frames (by column)
lightingSummary = cbind(lightingSummary,
                        NA,
                        NA)

names(lightingSummary)[3] = "sd"
names(lightingSummary)[4] = "n"

# remove redundant data frames
rm(lightingSummary.n, lightingSummary.sd)

# find the standard errors of each condition
lightingSummary$se = NA / NA
```

ggplot2

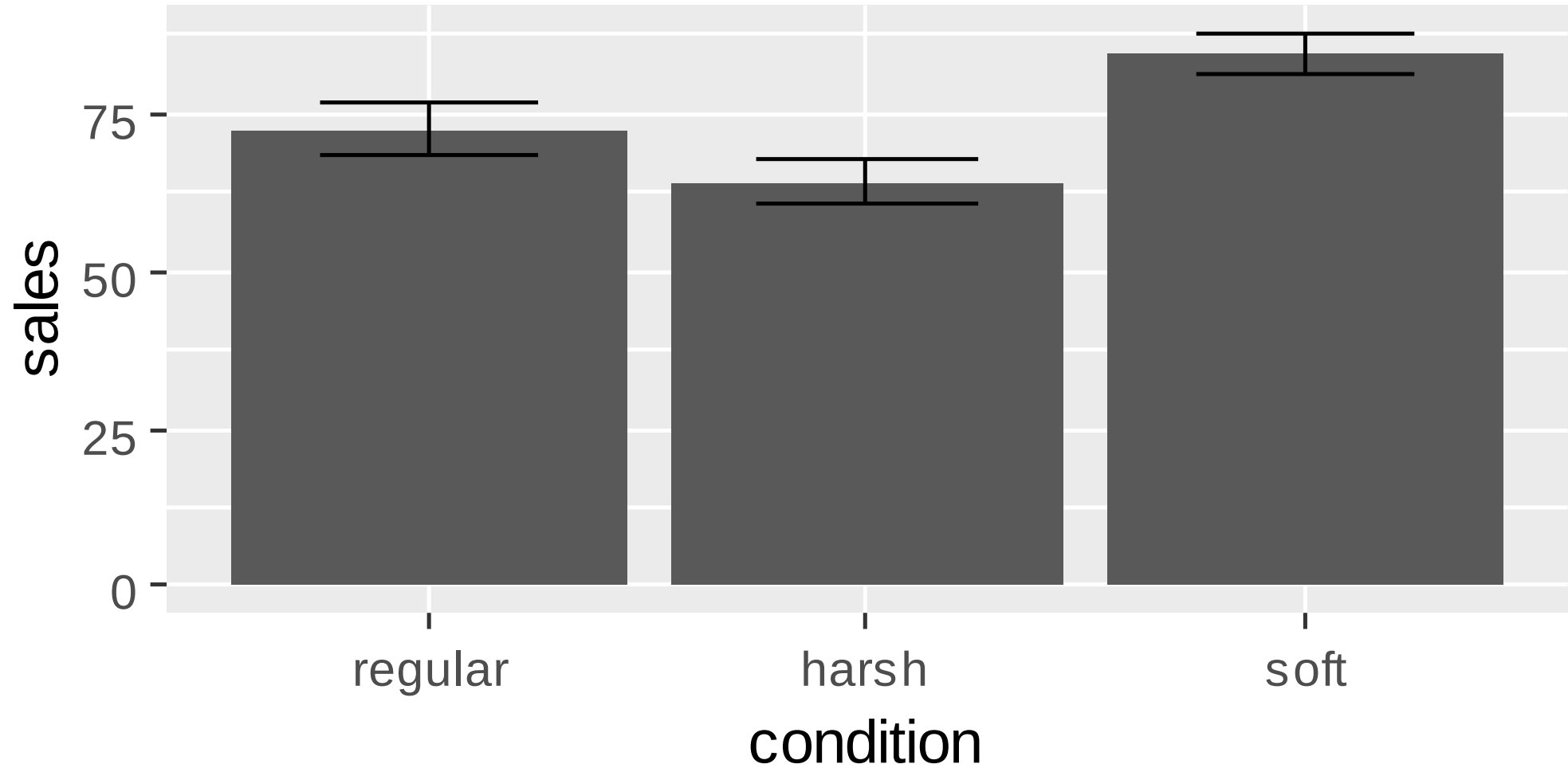
```
install.packages(ggplot2)

# basic graph
library(ggplot2)
ggplot(data=NA, aes(x=NA, y=NA)) +
  geom_bar(stat="identity")
```

- Graph with error bars (denoting standard errors of the means)

```
# with error bars
ggplot(data=NA, aes(x=NA, y=NA)) +
  geom_bar(stat="identity") +
  geom_errorbar(aes(ymin= NA - NA, ymax= NA + NA), width=.5)
```

Visualization



Error bars denotes standard error of the means in the condition