Assignment 2—One-way Analysis of Variance.

Yudhajit Ain

Student ID: 30182745

University of Calgary

PSYC 615—Lab

TAs: Christopher Davie & Benjamin Moon

Assignment 2—One-way Analysis of Variance.

# Question 1

*From the data directory, please import assignment-2\_q1.csv into R.*

# You can import the data in this chunk  
data\_q1=readr::read\_csv(here("data","assignment-2\_q1.csv"))  
#> Rows: 120 Columns: 3  
#> ── Column specification ────────────────────────────────────────────────────────  
#> Delimiter: ","  
#> chr (1): group  
#> dbl (2): superhero, agreeableness  
#>   
#> ℹ Use `spec()` to retrieve the full column specification for this data.  
#> ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

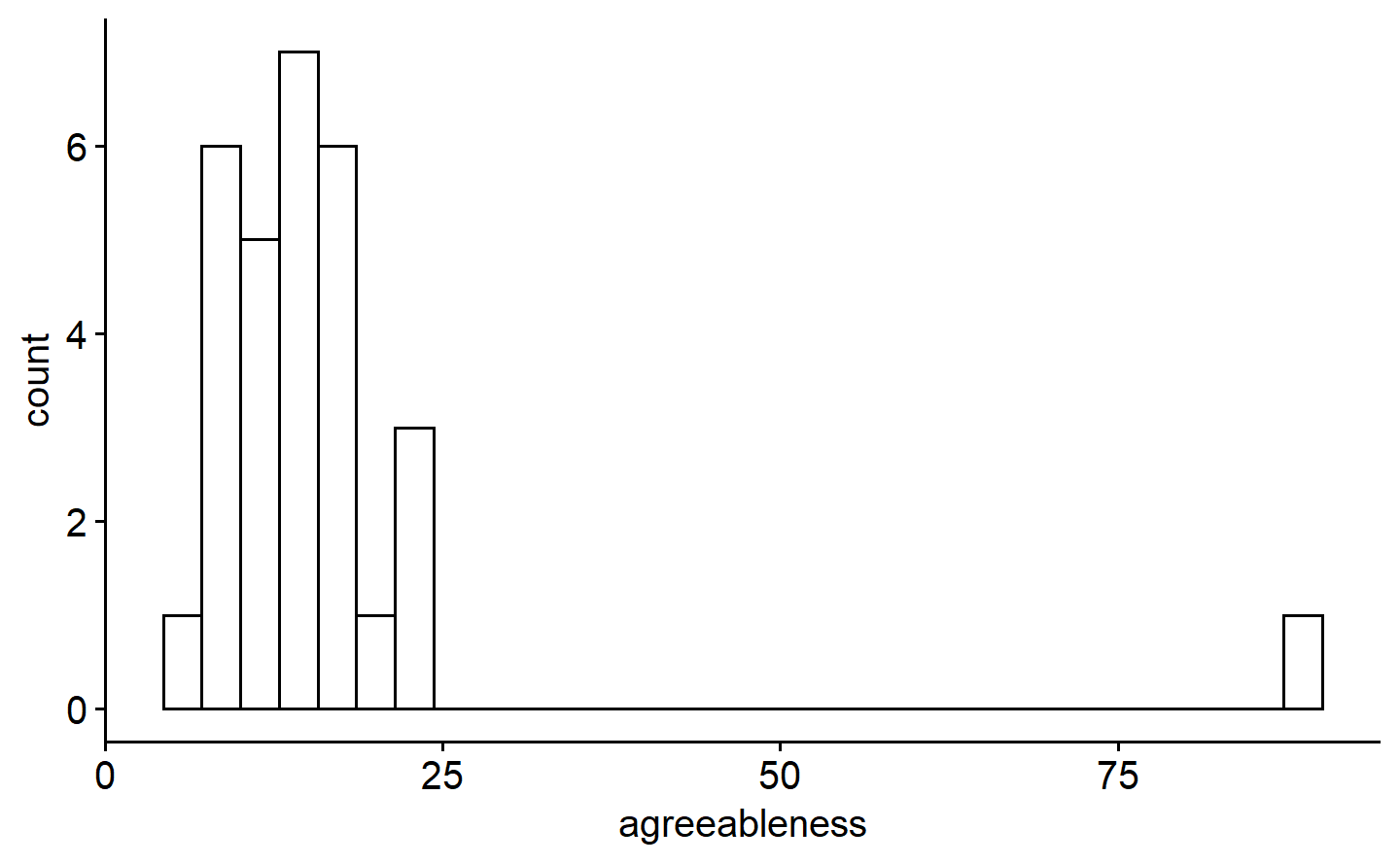
*Background: Nick Fury has approached you. He wants to put together a band of superheroes who can save the universe from current and future supervillains. He needs to decide whether he should hire superheroes from the Marvel universe (1), DC comics (2), Disney Princesses (3), or Smurfs (4). Because he plans to randomly sneak up on them to invite them (rather than, you know, sending an email) he wants to choose the group with highest agreeableness scores on the HEXACO personality inventory to avoid nasty confrontations when he shows up unannounced. He brings a dataset of agreeableness scores (on a scale from 1 – 34) to you and asks you to tell him if one group has significantly higher agreeableness, and which group that is.*

## Question 1a

*Let’s start with exploring the data. Does your data meet the assumptions for an ANOVA? If not, describe the assumption(s) that are violated, decide how you want to address the violations and explain exactly what you did, then re-check and report on your assumptions. (4 marks) Hint: Remember ANOVA is robust to violations of assumptions only in some circumstances – are those circumstances met?*

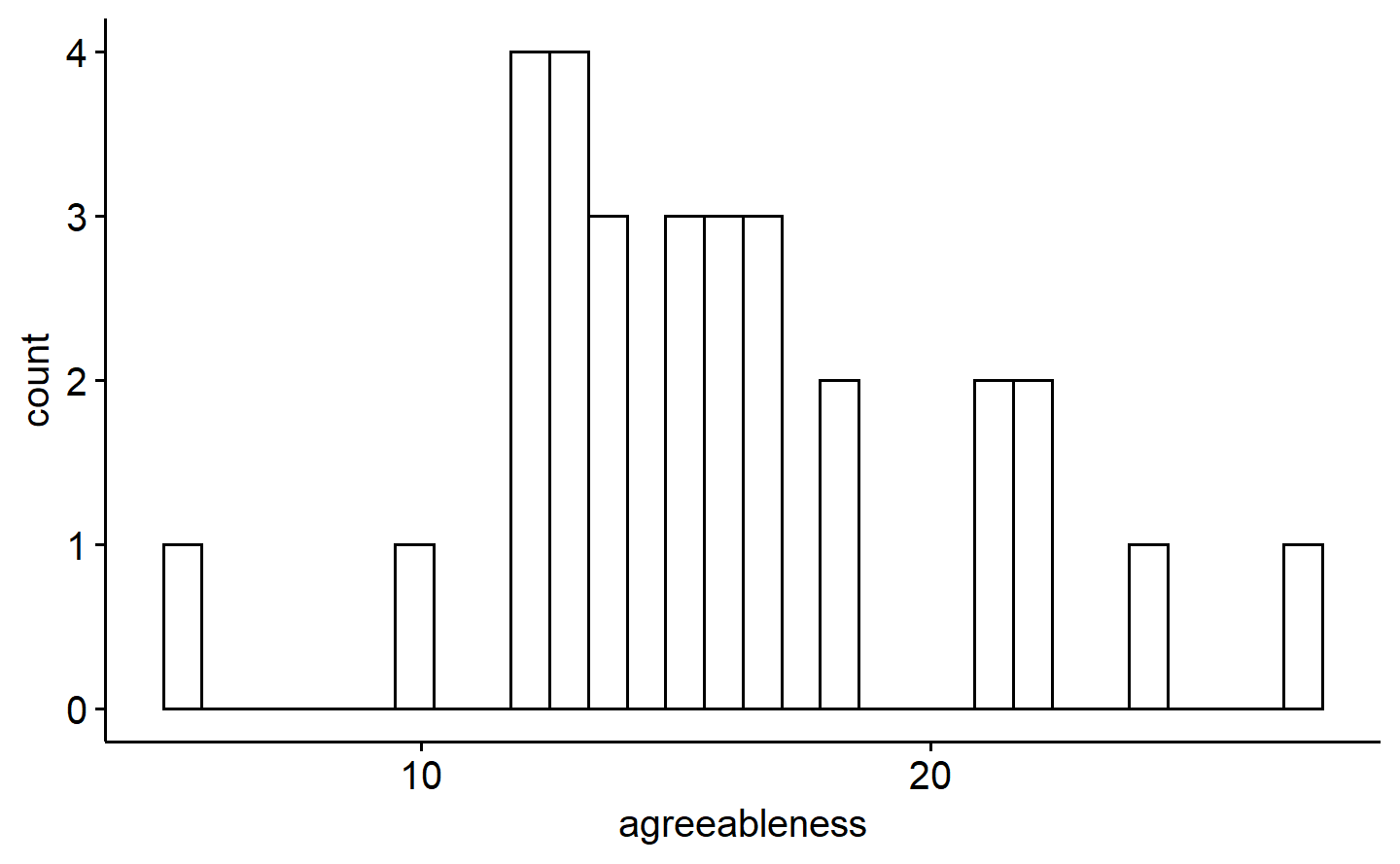
data\_q1\_dc=dplyr::filter(data\_q1,group == "DC")  
data\_q1\_marvel=dplyr::filter(data\_q1,group == "Marvel")  
data\_q1\_smurfs=dplyr::filter(data\_q1,group == "Smurfs")  
data\_q1\_disney=dplyr::filter(data\_q1,group == "Disney")  
  
shapiro.test(data\_q1\_dc$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_dc$agreeableness  
#> W = 0.44364, p-value = 1.427e-09  
shapiro.test(data\_q1\_marvel$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_marvel$agreeableness  
#> W = 0.9566, p-value = 0.2529  
shapiro.test(data\_q1\_smurfs$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_smurfs$agreeableness  
#> W = 0.24278, p-value = 2.397e-11  
shapiro.test(data\_q1\_disney$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_disney$agreeableness  
#> W = 0.95001, p-value = 0.1692

ggpubr::gghistogram(data=data\_q1\_dc,x="agreeableness",y="..count..")  
#> Warning: Using `bins = 30` by default. Pick better value with the argument  
#> `bins`.



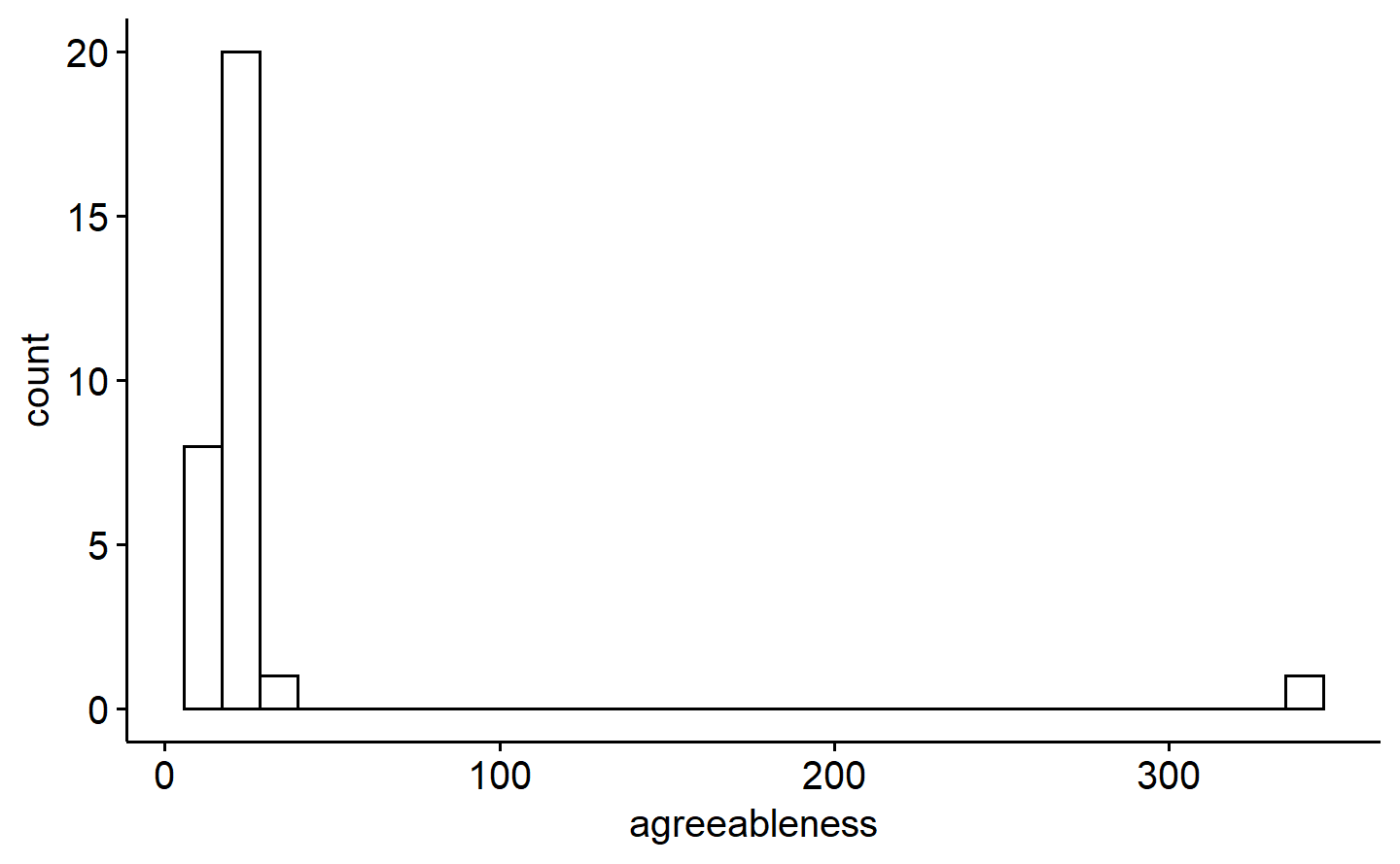
*Figure* *1.*  Distribution of agreeableness for DC superheroes

ggpubr::gghistogram(data=data\_q1\_marvel,x="agreeableness",y="..count..")  
#> Warning: Using `bins = 30` by default. Pick better value with the argument  
#> `bins`.



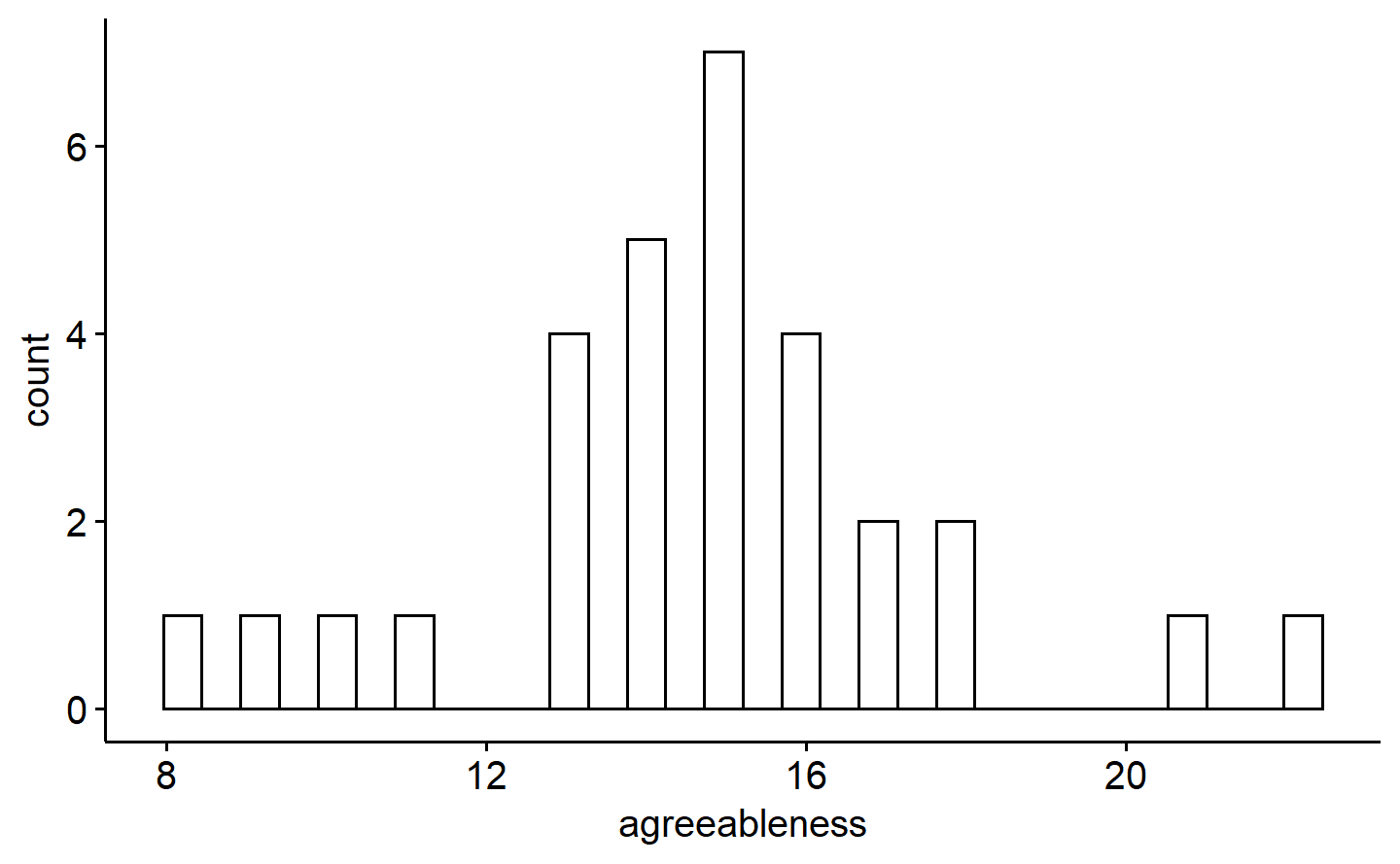
*Figure* *2.*  Distribution of agreeableness for Marvel superheroes

ggpubr::gghistogram(data=data\_q1\_smurfs,x="agreeableness",y="..count..")  
#> Warning: Using `bins = 30` by default. Pick better value with the argument  
#> `bins`.



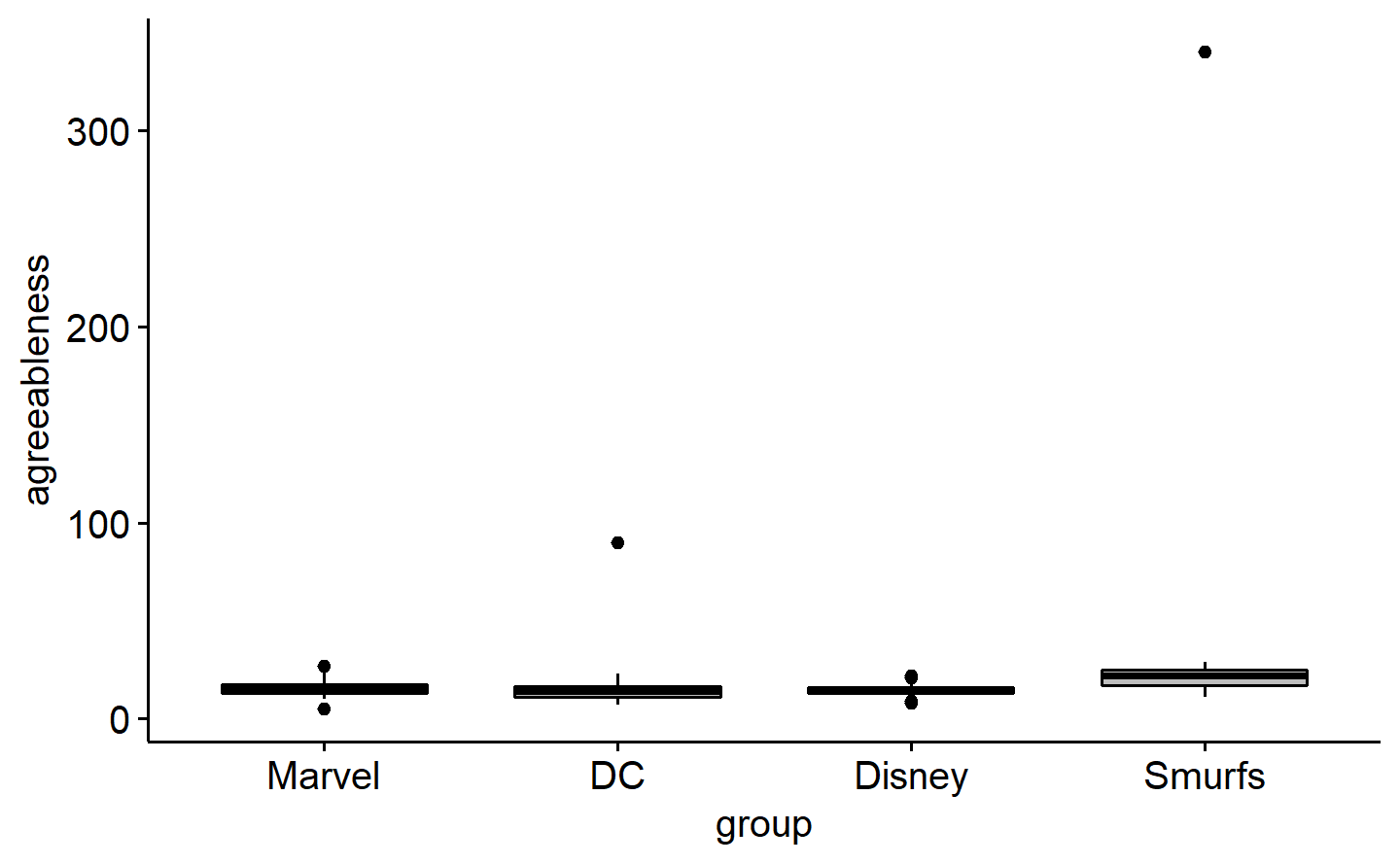
*Figure* *3.*  Distribution of agreeableness for Smurfs

ggpubr::gghistogram(data=data\_q1\_disney,x="agreeableness",y="..count..")  
#> Warning: Using `bins = 30` by default. Pick better value with the argument  
#> `bins`.



*Figure* *4.*  Distribution of agreeableness for Disney princesses

ggpubr::ggboxplot(data=data\_q1,x="group",y="agreeableness", fill="grey")



*Figure* *5.*  Box-plot of agreeableness for all groups

dplyr::arrange(data\_q1,agreeableness)#102dc,4smurfs  
#> # A tibble: 120 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 21 Marvel 5  
#> 2 118 DC 7  
#> 3 14 DC 8  
#> 4 27 Disney 8  
#> 5 66 DC 8  
#> 6 106 DC 8  
#> 7 107 Disney 9  
#> 8 114 DC 9  
#> 9 30 DC 10  
#> 10 38 DC 10  
#> # … with 110 more rows  
  
data\_q1\_filtered=dplyr::filter(data\_q1, superhero != 102 & superhero != 4)  
  
car::leveneTest(y=data\_q1\_filtered$agreeableness,group=data\_q1\_filtered$group,center=mean)  
#> Warning in leveneTest.default(y = data\_q1\_filtered$agreeableness, group =  
#> data\_q1\_filtered$group, : data\_q1\_filtered$group coerced to factor.  
#> Levene's Test for Homogeneity of Variance (center = mean)  
#> Df F value Pr(>F)   
#> group 3 2.8296 0.04162 \*  
#> 114   
#> ---  
#> Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
  
data\_q1\_filtered\_dc=dplyr::filter(data\_q1\_filtered,group == "DC")  
data\_q1\_filtered\_marvel=dplyr::filter(data\_q1\_filtered,group == "Marvel")  
data\_q1\_filtered\_smurfs=dplyr::filter(data\_q1\_filtered,group == "Smurfs")  
data\_q1\_filtered\_disney=dplyr::filter(data\_q1\_filtered,group == "Disney")  
  
shapiro.test(data\_q1\_filtered\_dc$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_filtered\_dc$agreeableness  
#> W = 0.96266, p-value = 0.3815  
shapiro.test(data\_q1\_filtered\_marvel$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_filtered\_marvel$agreeableness  
#> W = 0.9566, p-value = 0.2529  
shapiro.test(data\_q1\_filtered\_smurfs$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_filtered\_smurfs$agreeableness  
#> W = 0.96108, p-value = 0.3493  
shapiro.test(data\_q1\_filtered\_disney$agreeableness)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q1\_filtered\_disney$agreeableness  
#> W = 0.95001, p-value = 0.1692  
  
var(data\_q1\_filtered\_dc$agreeableness)  
#> [1] 19.13793  
var(data\_q1\_filtered\_marvel$agreeableness)  
#> [1] 20.30345  
var(data\_q1\_filtered\_smurfs$agreeableness)  
#> [1] 21.83744  
var(data\_q1\_filtered\_disney$agreeableness)  
#> [1] 8.96092

Answers here.

We plotted separtae histograms to look at the distribution of agreeableness in all superhero groups (figures 1-4), and also explicitly tested normality within groups using the Shapiro-Wilk test. The distribution was significantly non-normal for both the DC and Smurfs group.

Next, we made a box-plot of the agreeableness score for all groups (figure 5), to look at possible outliers. Both non-normal groups (DC and Smurfs) showed positive outliers, and they were then filtered out.

After filtering, the Shapiro-Wilk normality test was repeated, and all groups had a normal distribution now. Thereafter, we did the Levene’s test to test for homogeneity of variance across groups, and it revealed that the variances were not equal across all groups. However, since 1) the distributions are now all normal, 2) the sample sizes are almost equal for all groups, and finally, 3) the highest group variance (19.14 for DC) is less than 3 times the lowest variance (8.96 for Disney), the ANOVA should be fairly robust to the violation of homogeneity of variance.

## Question 1b

*Evaluate if the superhero groups differ on agreeableness and, if so, which groups differ. Write an APA formatted results section. (10 marks) This includes stating the hypothesis that is being evaluated, quickly reporting on assumptions (we assume independence from here on so no need for that, and only report normality and homogeneity after removing any outliers), and reporting the findings, including the omnibus test and all follow-ups (provide and reference an APA formatted figure). Hint: When reporting findings, you need to include practical significance!*

# Results

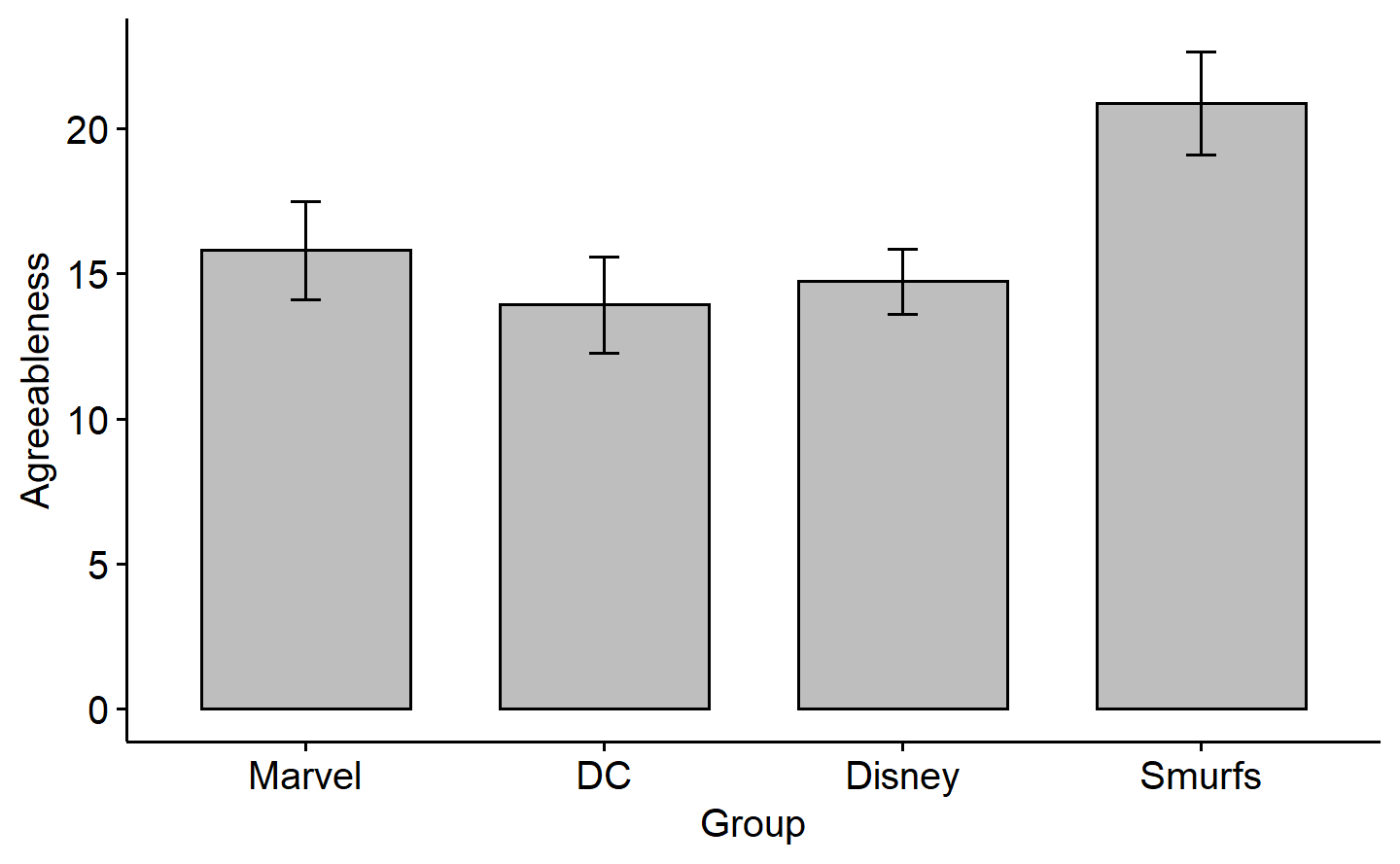
agree\_model=aov(agreeableness ~ group, data = data\_q1\_filtered)  
broom::tidy(agree\_model)  
#> # A tibble: 2 × 6  
#> term df sumsq meansq statistic p.value  
#> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
#> 1 group 3 847. 282. 16.1 8.30e-9  
#> 2 Residuals 114 1996. 17.5 NA NA  
agree\_model  
#> Call:  
#> aov(formula = agreeableness ~ group, data = data\_q1\_filtered)  
#>   
#> Terms:  
#> group Residuals  
#> Sum of Squares 847.4213 1995.9770  
#> Deg. of Freedom 3 114  
#>   
#> Residual standard error: 4.184324  
#> Estimated effects may be unbalanced  
  
effectsize::omega\_squared(agree\_model)  
#> For one-way between subjects designs, partial omega squared is equivalent to omega squared.  
#> Returning omega squared.  
#> # Effect Size for ANOVA  
#>   
#> Parameter | Omega2 | 95% CI  
#> ---------------------------------  
#> group | 0.28 | [0.16, 1.00]  
#>   
#> - One-sided CIs: upper bound fixed at [1.00].  
  
agree\_emm=emmeans::emmeans(object=agree\_model,specs = "group")  
agree\_emm  
#> group emmean SE df lower.CL upper.CL  
#> DC 13.9 0.777 114 12.4 15.5  
#> Disney 14.7 0.764 114 13.2 16.2  
#> Marvel 15.8 0.764 114 14.3 17.3  
#> Smurfs 20.9 0.777 114 19.3 22.4  
#>   
#> Confidence level used: 0.95  
  
emmeans::contrast(object=agree\_emm, method="pairwise", adjust="tukey")  
#> contrast estimate SE df t.ratio p.value  
#> DC - Disney -0.802 1.09 114 -0.736 0.8823  
#> DC - Marvel -1.869 1.09 114 -1.715 0.3206  
#> DC - Smurfs -6.931 1.10 114 -6.307 <.0001  
#> Disney - Marvel -1.067 1.08 114 -0.987 0.7570  
#> Disney - Smurfs -6.129 1.09 114 -5.624 <.0001  
#> Marvel - Smurfs -5.062 1.09 114 -4.646 0.0001  
#>   
#> P value adjustment: tukey method for comparing a family of 4 estimates  
  
emmeans::eff\_size(  
 object=agree\_emm,  
 sigma=sigma(agree\_model),  
 edf=df.residual(agree\_model),  
 method="pairwise"  
)  
#> contrast effect.size SE df lower.CL upper.CL  
#> DC - Disney -0.192 0.261 114 -0.708 0.3248  
#> DC - Marvel -0.447 0.262 114 -0.966 0.0725  
#> DC - Smurfs -1.656 0.285 114 -2.220 -1.0926  
#> Disney - Marvel -0.255 0.259 114 -0.768 0.2577  
#> Disney - Smurfs -1.465 0.278 114 -2.015 -0.9142  
#> Marvel - Smurfs -1.210 0.272 114 -1.750 -0.6700  
#>   
#> sigma used for effect sizes: 4.184   
#> Confidence level used: 0.95  
  
mean(data\_q1\_filtered\_dc$agreeableness)  
#> [1] 13.93103  
mean(data\_q1\_filtered\_marvel$agreeableness)  
#> [1] 15.8  
mean(data\_q1\_filtered\_smurfs$agreeableness)  
#> [1] 20.86207  
mean(data\_q1\_filtered\_disney$agreeableness)  
#> [1] 14.73333  
  
sd(data\_q1\_filtered\_dc$agreeableness)  
#> [1] 4.374692  
sd(data\_q1\_filtered\_marvel$agreeableness)  
#> [1] 4.505935  
sd(data\_q1\_filtered\_smurfs$agreeableness)  
#> [1] 4.673055  
sd(data\_q1\_filtered\_disney$agreeableness)  
#> [1] 2.99348

Answers here.

# Results

In order to test for any significant mean differences on agreeableness between the four groups (DC, Marvel, Disney and Smurfs), we perform a one-way ANOVA with the null hypothesis : . Agreeableness was normally distributed in all groups, DC , Marvel , Smurfs and Disney princesses . The Levene’s test for homogeneity of variance, however, was significant , indicating unequal variance across the groups. With roughly equal sample sizes, and normality preserved within groups, we expect the ANOVA to be robust to the violation of homogeneity of variance, and so we proceed with it. The omnibus -test turns out to be significant , suggesting that at least one group has a significantly different mean agreeableness compared to other(s). To check for which specific groups have mean agreeableness significantly different from each other, we run a full pairwise post-hoc test, with -values adjusted using Tukey’s HSD. The results suggest that Smurfs were significantly more agreeable than DC heroes , , Marvel heroes , , as well as Disney princesses , . A bar-plot (figure 6) summarises the results visually.

ggpubr::ggbarplot(data=data\_q1\_filtered,x="group",y="agreeableness",add = "mean\_ci", xlab="Group", ylab = "Agreeableness", fill="grey")



*Figure* *6.*  Bar-plot of agreeableness across all groups

## Question 1c

*When Nick Fury reads your results, he gets very excited—he thinks this research could be written up in the Quarterly Journal of Superheroes and Government Agencies! He notices that one group appears to be significantly different from all the other groups and suggests conducting a complex comparison instead of pairwise comparisons as follow-up tests. State how you would respond to Nick Fury. Is it wise to run a complex comparison as a follow-up in this case? Why or why not? (2 marks)*

Answers here.

I would suggest Nick Fury to not go ahead with the idea. Having already done, and seen the results of the pairwise comparisons, the chances of making a type-I error in the complex comparison will be highly inflated because of the hypothesis being driven by results which are already significant. If he is very keen, however, I would suggest him to use a suitable test for this, like the -Scheffe’s.

# Question 2

## Question 2a

*Briefly describe how planned comparisons differ from post-hoc comparisons in terms of error? (1 mark)*

Answers here.

Planned comparisons are set up by the researcher(s) before having looked at the data, and is driven purely by the hypothesis and the theoretical predictions. Post-hoc comparisons, on the other hand, are done after having looked at the data visually, or after performing an omnibus -test or ANOVA. As such, the chances of making a type-I error is much more likely while doing uncorrected post-hoc comparisons (since the hypothesis is informed by the data itself), as opposed to uncorrected planned comparisons.

## Question 2b

*What are two different types of planned comparisons (think # of groups involved) a researcher can do and what makes them different? (1 mark)*

Answers here.

The two kind of planned comparisons are : 1) simple (where we compare one group against another single group in the dataset), and 2) complex (where we compare a single, or combination of groups against another combination of groups).

# Question 3

*From the data directory, please import assignment-2\_q3.csv into R.*

data\_q3=readr::read\_csv(here("data","assignment-2\_q3.csv"))  
#> Rows: 125 Columns: 10  
#> ── Column specification ────────────────────────────────────────────────────────  
#> Delimiter: ","  
#> chr (5): sitcom, sitcom\_type, q2a\_h1, q2a\_h2, q2a\_h3  
#> dbl (5): grad\_student, sitcom\_quality, laughs, binge, snacks  
#>   
#> ℹ Use `spec()` to retrieve the full column specification for this data.  
#> ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

*Background: Did you know that people “research” sitcoms? For example, there are three main ’types” of sitcoms: House (surrounding a family home), Apartment (friends hanging out), and Workplace (co-worker shenanigans; see here!). Who knew, right? Now that you know this is a topic you can research, you are determined to see if different sitcoms are objectively “better” than others. You decide to recruit fellow graduate students, who are just as eager to find reasons to procrastinate, to watch three seasons of a sitcom and report on the following, very objective, measures: average laughs per episode, binge-worthiness of the show, and number of tasty snacks consumed while watching. These three variables will be summed to create one dependent variable “sitcom quality”. Trust it is valid & reliable!*

*Students are randomly assigned to watch one of the following sitcoms and note their measure values. The sitcoms used were:*

* *Modern family (House type)*
* *Friends (Apartment type)*
* *Seinfeld (Apartment type)*
* *The Office (Workplace type)*
* *Brooklyn Nine-Nine (Workplace type)*

*After your extensive background research in the area, which was definitely not just binging shows all weekend, you come up with the following three hypotheses (i.e., planned comparisons):*

1. *Home types (House & Apartment) are better than Workplace types.*
2. *Apartment types are better than House types.*
3. *Seinfeld is better than Friends.*

## Question 3a

*Are all of your planned comparisons orthogonal to each other? Please justify your answers by showing the formulas/calculations for each pair. (2 marks)*

Answers here.

The contrasts, with the “sitcom” factor in the same order (top to bottom) as in the “assignment-2\_q3.csv” dataset, are :

contrast 1 : (1,1,1,-1.5,-1.5)

contrast 2 : (-2,1,1,0,0)

constrat 3 : (0,-1,1,0,0)

For contrast\_1 and contrast\_2, the dot product =

For contrast\_1 and contrast\_3, the dot product =

For contrast\_2 and contrast\_3, the dot product =

Thus, all three comparisons are mutually orthogonal.

## Question 3b

*Explain why your planned comparisons are not a complete/full orthogonal set? State additional hypothesis(es)—in words and contrast codes—that could be used to complete the set. No need to show formulas/calculations for the new hypothesis. (1 mark)*

Answers here.

The number of groups (sitcoms) in our dataset = = 5. We know that the number of orthogonal contrasts that make up a complete orthogonal set for a model with groups, is . Here, that is four contrasts to make up a complete set. We already have three.

The last contrast that would complete the set is contrast 4 = (0,0,0,-1,1). This contrast essentially tests the hypothesis that Brooklyn Nine-Nine is better than The Office.

## Question 3c

*Write an APA results section that includes the following: (11 marks total, see below for breakdown):*

1. *Brief report of assumptions (No need to clean the data this time, aside from assumption checking. PLEASE DO NOT REMOVE ANY DATA FOR THIS ASSIGNMENT!) (1 mark)*
2. *Omnibus results (as is tradition, and including effect size) (1 mark)*
3. *The planned comparisons from Questions 3a and 3b (including means and SD that relate to the “groups” being compared and appropriate effect sizes) (8 marks)*
4. *Appropriate figures visualizing each hypothesis (3 marks).*

# Filter individual groups  
  
data\_q3\_modfam=dplyr::filter(.data=data\_q3, sitcom == "Modern Family")  
data\_q3\_friends=dplyr::filter(.data=data\_q3, sitcom == "Friends")  
data\_q3\_seinfeld=dplyr::filter(.data=data\_q3, sitcom == "Seinfeld")  
data\_q3\_office=dplyr::filter(.data=data\_q3, sitcom == "The Office")  
data\_q3\_b99=dplyr::filter(.data=data\_q3, sitcom == "Brooklyn Nine-Nine")  
  
# Make the combined groups for complex comparisons  
  
data\_q3\_home=rbind.data.frame(data\_q3\_modfam,data\_q3\_seinfeld,data\_q3\_friends)  
data\_q3\_work=rbind.data.frame(data\_q3\_office,data\_q3\_b99)  
data\_q3\_apartment=rbind.data.frame(data\_q3\_friends,data\_q3\_seinfeld)  
  
# Within-group normality tests  
  
shapiro.test(data\_q3\_modfam$sitcom\_quality)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q3\_modfam$sitcom\_quality  
#> W = 0.94911, p-value = 0.2394  
shapiro.test(data\_q3\_friends$sitcom\_quality)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q3\_friends$sitcom\_quality  
#> W = 0.9454, p-value = 0.197  
shapiro.test(data\_q3\_seinfeld$sitcom\_quality)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q3\_seinfeld$sitcom\_quality  
#> W = 0.94344, p-value = 0.1777  
shapiro.test(data\_q3\_office$sitcom\_quality)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q3\_office$sitcom\_quality  
#> W = 0.96992, p-value = 0.643  
shapiro.test(data\_q3\_b99$sitcom\_quality)  
#>   
#> Shapiro-Wilk normality test  
#>   
#> data: data\_q3\_b99$sitcom\_quality  
#> W = 0.94984, p-value = 0.2486  
  
# Levene's test  
  
car::leveneTest(y=data\_q3$sitcom\_quality,group=data\_q3$sitcom,center=mean)  
#> Warning in leveneTest.default(y = data\_q3$sitcom\_quality, group =  
#> data\_q3$sitcom, : data\_q3$sitcom coerced to factor.  
#> Levene's Test for Homogeneity of Variance (center = mean)  
#> Df F value Pr(>F)  
#> group 4 1.5247 0.1993  
#> 120  
  
# Omnibus/ANOVA  
sitcom\_model=aov(sitcom\_quality ~ sitcom, data=data\_q3)  
broom::tidy(sitcom\_model)  
#> # A tibble: 2 × 6  
#> term df sumsq meansq statistic p.value  
#> <chr> <dbl> <dbl> <dbl> <dbl> <dbl>  
#> 1 sitcom 4 413. 103. 11.0 0.000000117  
#> 2 Residuals 120 1122. 9.35 NA NA  
effectsize::omega\_squared(sitcom\_model)  
#> For one-way between subjects designs, partial omega squared is equivalent to omega squared.  
#> Returning omega squared.  
#> # Effect Size for ANOVA  
#>   
#> Parameter | Omega2 | 95% CI  
#> ---------------------------------  
#> sitcom | 0.24 | [0.12, 1.00]  
#>   
#> - One-sided CIs: upper bound fixed at [1.00].  
  
# Set contrasts  
  
sitcom\_emm=emmeans::emmeans(object=sitcom\_model,specs="sitcom")  
  
c1=c(1,1,1,-1.5,-1.5)  
c2=c(-2,1,1,0,0)  
c3=c(0,-1,1,0,0)  
c4=c(0,0,0,-1,1)  
  
data\_q3=dplyr::mutate(data\_q3,sitcom=forcats::as\_factor(sitcom))  
  
contrasts(data\_q3$sitcom)=cbind(c1,c2,c3,c4)  
#contrasts(data\_q3$sitcom)  
  
# Perform planned comparisons  
  
sitcom\_contrast=emmeans::contrast(  
 sitcom\_emm,  
 method=list("Home - Workplace"=c1, "Apartment - House"=c2, "Seinfeld - Friends"=c3 , "Brooklyn Nine-Nine - The Office"=c4),  
 adjust="bonferroni"  
 )  
#sitcom\_contrast  
  
sitcom\_contrast=broom::tidy(sitcom\_contrast)  
  
# Find contrast effect-sizes  
  
contrast\_sum=colSums(abs(contrasts(data\_q3$sitcom)))  
  
sitcom\_contrast=dplyr::mutate(  
 sitcom\_contrast,   
 effect.size=(2\*estimate)/(sigma(sitcom\_model)\*contrast\_sum))  
  
sitcom\_contrast  
#> # A tibble: 4 × 9  
#> term contrast null.…¹ estim…² std.e…³ df stati…⁴ adj.p…⁵ effec…⁶  
#> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
#> 1 sitcom Home - Workplace 0 7.18 1.67 120 4.29 1.47e-4 0.783  
#> 2 sitcom Apartment - House 0 0.880 1.50 120 0.588 1 e+0 0.144  
#> 3 sitcom Seinfeld - Frien… 0 -4.32 0.865 120 -5.00 8.07e-6 -1.41   
#> 4 sitcom Brooklyn Nine-Ni… 0 -0.600 0.865 120 -0.694 1 e+0 -0.196  
#> # … with abbreviated variable names ¹​null.value, ²​estimate, ³​std.error,  
#> # ⁴​statistic, ⁵​adj.p.value, ⁶​effect.size  
  
mean(data\_q3\_home$sitcom\_quality)  
#> [1] 17.41333  
sd(data\_q3\_home$sitcom\_quality)  
#> [1] 3.866325  
mean(data\_q3\_work$sitcom\_quality)  
#> [1] 16.52  
sd(data\_q3\_work$sitcom\_quality)  
#> [1] 2.873098  
mean(data\_q3\_apartment$sitcom\_quality)  
#> [1] 18.12  
sd(data\_q3\_apartment$sitcom\_quality)  
#> [1] 3.910426  
  
mean(data\_q3\_modfam$sitcom\_quality)  
#> [1] 16  
sd(data\_q3\_modfam$sitcom\_quality)  
#> [1] 3.427827  
mean(data\_q3\_seinfeld$sitcom\_quality)  
#> [1] 15.92  
sd(data\_q3\_seinfeld$sitcom\_quality)  
#> [1] 3.534591  
mean(data\_q3\_friends$sitcom\_quality)  
#> [1] 20.32  
sd(data\_q3\_friends$sitcom\_quality)  
#> [1] 2.939955  
mean(data\_q3\_office$sitcom\_quality)  
#> [1] 15.32  
sd(data\_q3\_office$sitcom\_quality)  
#> [1] 2.939955  
mean(data\_q3\_b99$sitcom\_quality)  
#> [1] 17.72  
sd(data\_q3\_b99$sitcom\_quality)  
#> [1] 2.282542

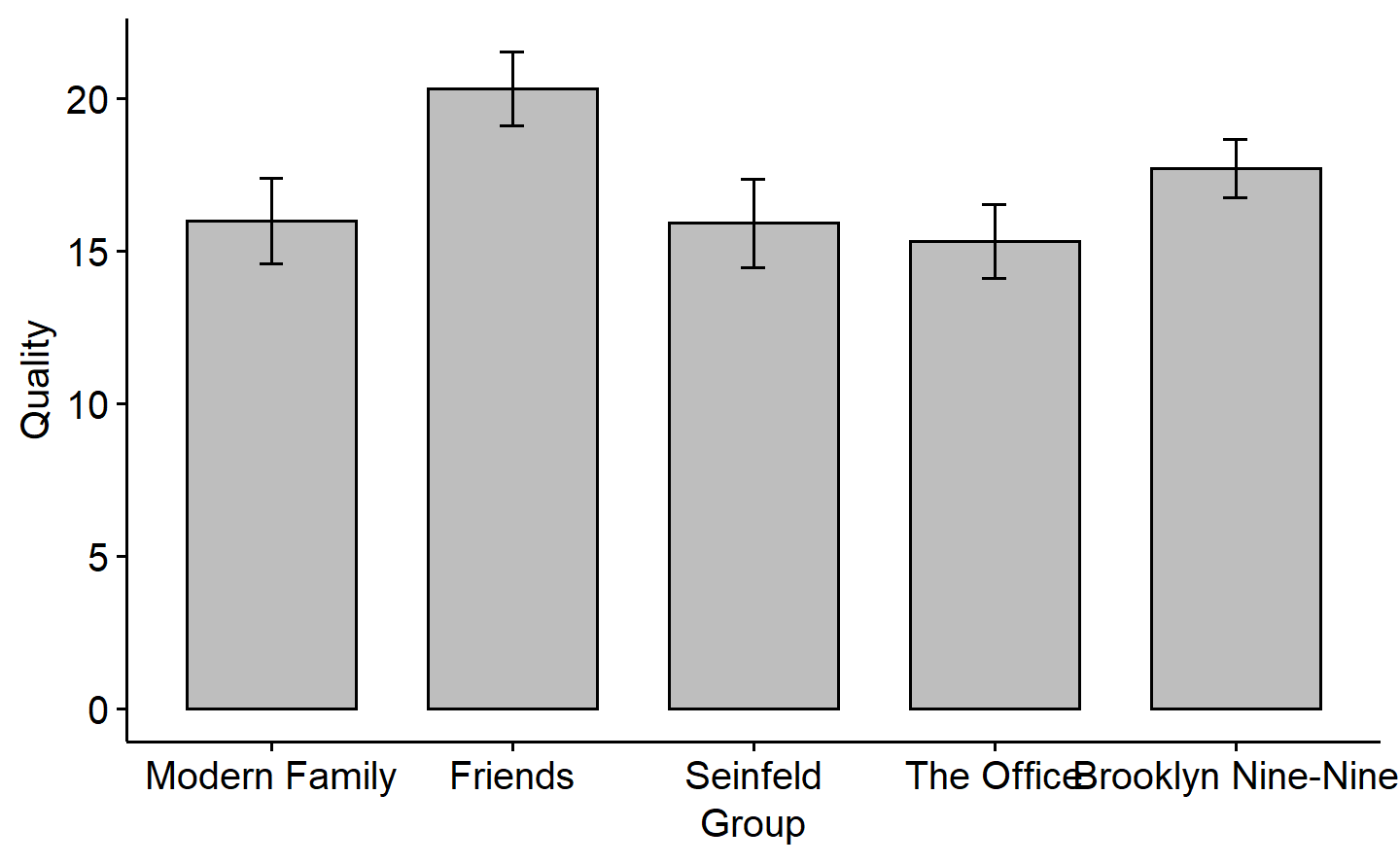
Answers here.

# Results

We wanted to test if there were any significant differences between the mean sitcom quality of the five sitcoms (groups) in our dataset. To do this, we have to first perform a one-way ANOVA. We check for normality within each group (sitcom), and find that the quality ratings for all of them, Modern Family , Friends , Seinfeld , The Office , and Brooklyn Nine-Nine were normally distributed. To check for homogeneity of variance across the groups, we perform the Levene’s Test, and the result suggests equality of variance across groups . With all assumptions met, we proceed to perform the one-way ANOVA. The results of the ANOVA suggest that indeed, at least one of the sitcoms has a mean quality rating that is significantly different from the others . Figure 7 summarises the analysis visually.

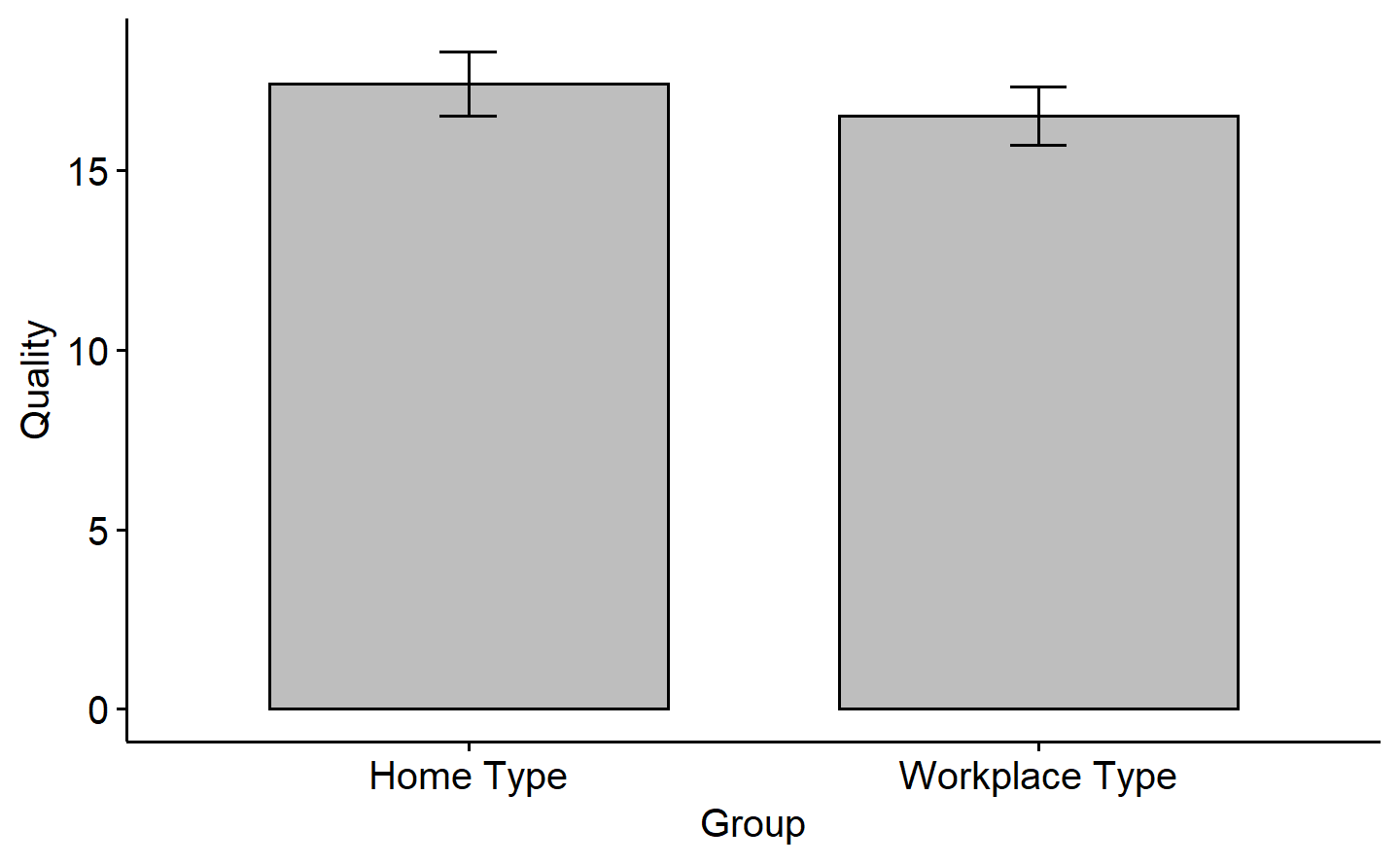
We then proceed to test our own hypotheses through planned comparisons. To compensate for multiple comparisons, we use the Bonferroni adjusted -values. The result of the first comparison suggests that there is no significant difference in the quality of Home-type sitcoms compared to Workplace-type sitcoms , . This is seen in figure 8. The second comparison reveals that Apartment-type sitcoms were rated as having significantly better quality than House-type sitcoms , . This is illustrated in figure 9. The third comparison reveals that the quality of Seinfeld is rated significantly lower than that of Friends , . This can be seen in figure 10. Finally, the fourth comparison suggests that the quality of Brooklyn Nine-Nine is rated significantly higher than that of The Office , . Figure 11 summarizes this visually.

ggpubr::ggbarplot(data = data\_q3,x="sitcom",y="sitcom\_quality", add = "mean\_ci", xlab="Group", ylab="Quality", fill="grey")



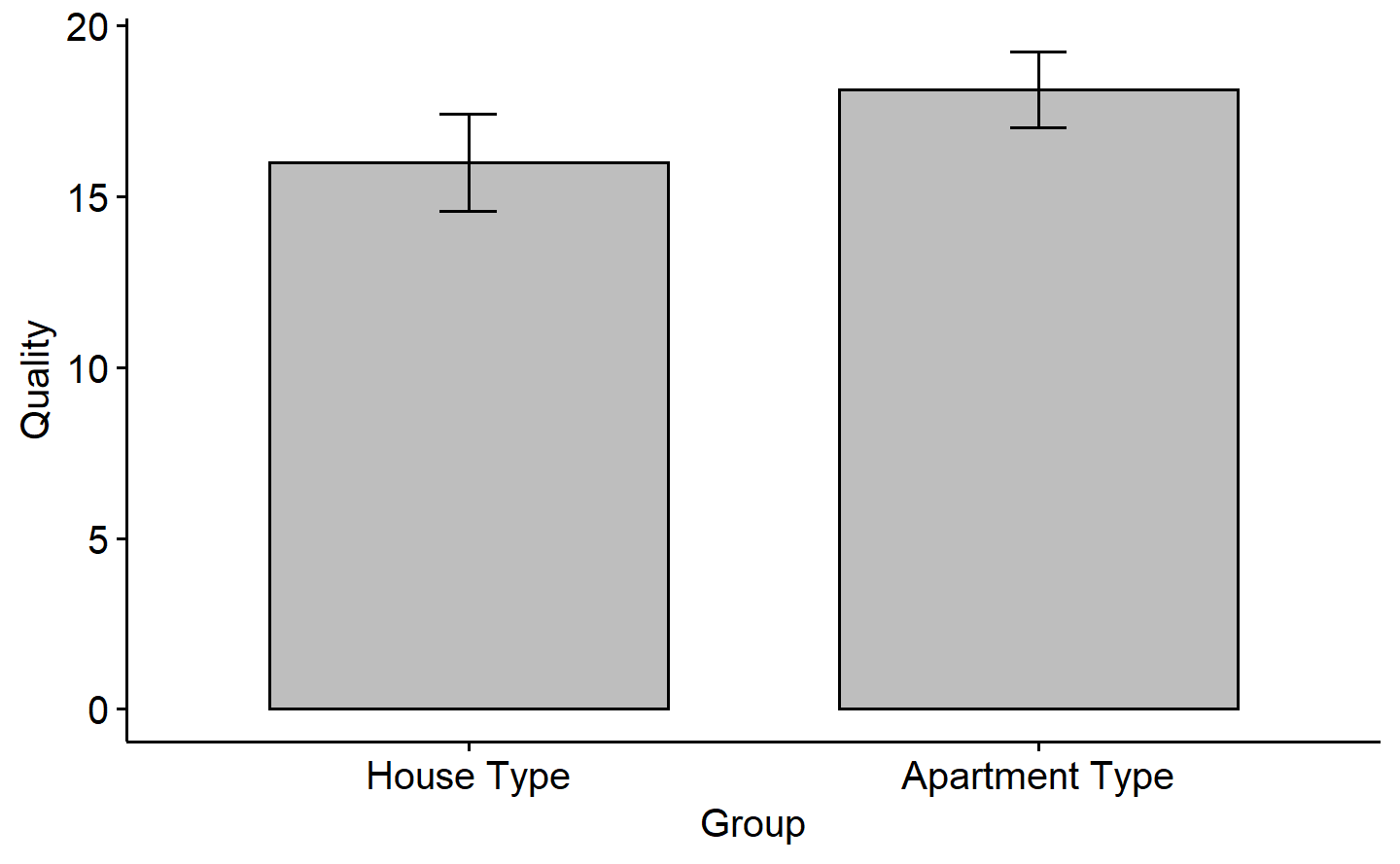
*Figure* *7.*  bar-plot comparing mean sitcom quality across all sitcoms

ggpubr::ggbarplot(data = data\_q3,x="q2a\_h1",y="sitcom\_quality", add = "mean\_ci", xlab="Group", ylab="Quality", fill="grey")



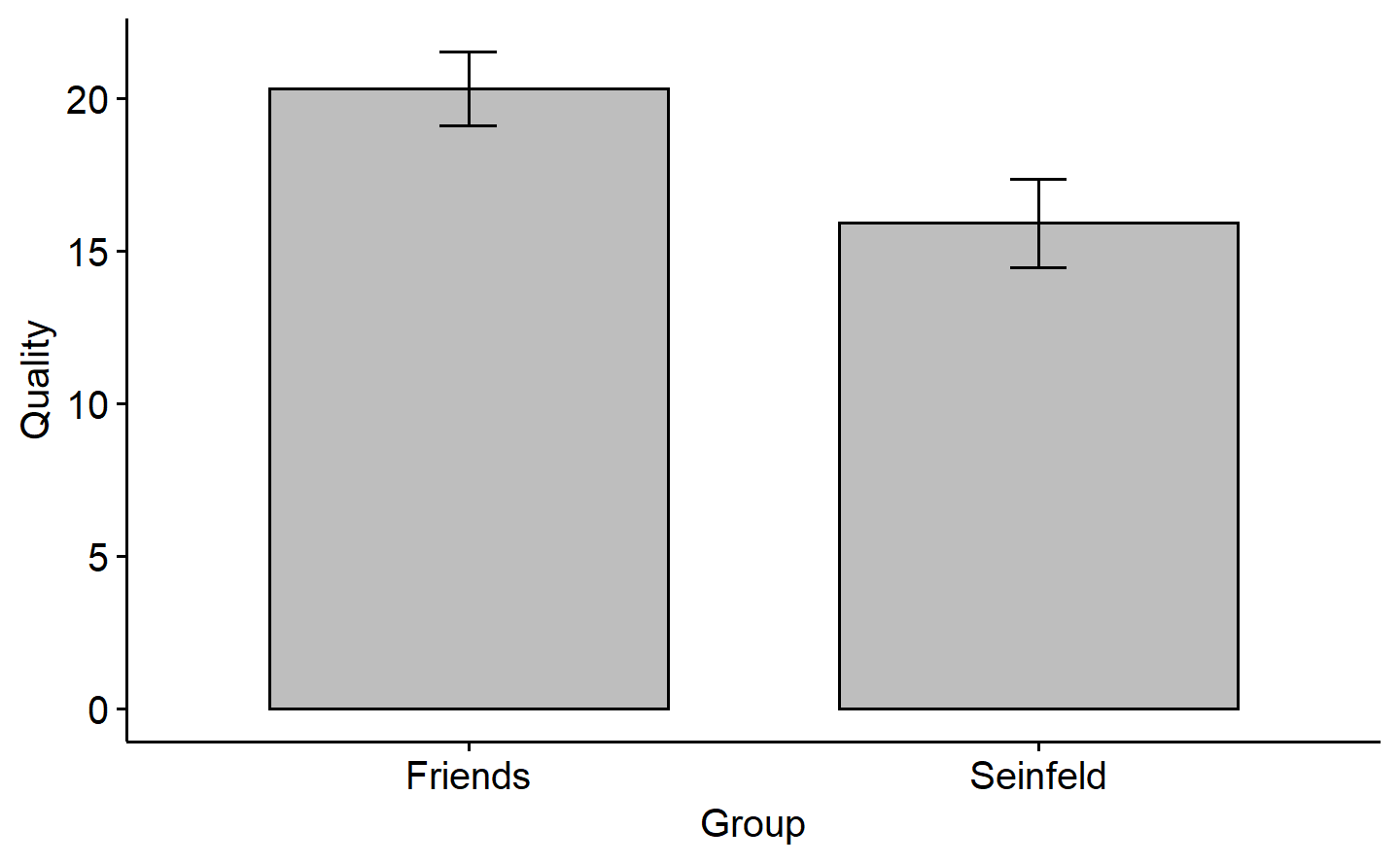
*Figure* *8.*  Bar-plot comparing mean sitcom quality between Home type and Workplace type sitcoms

ggpubr::ggbarplot(data = data\_q3\_home,x="q2a\_h2",y="sitcom\_quality", add = "mean\_ci", xlab="Group", ylab="Quality", fill="grey")



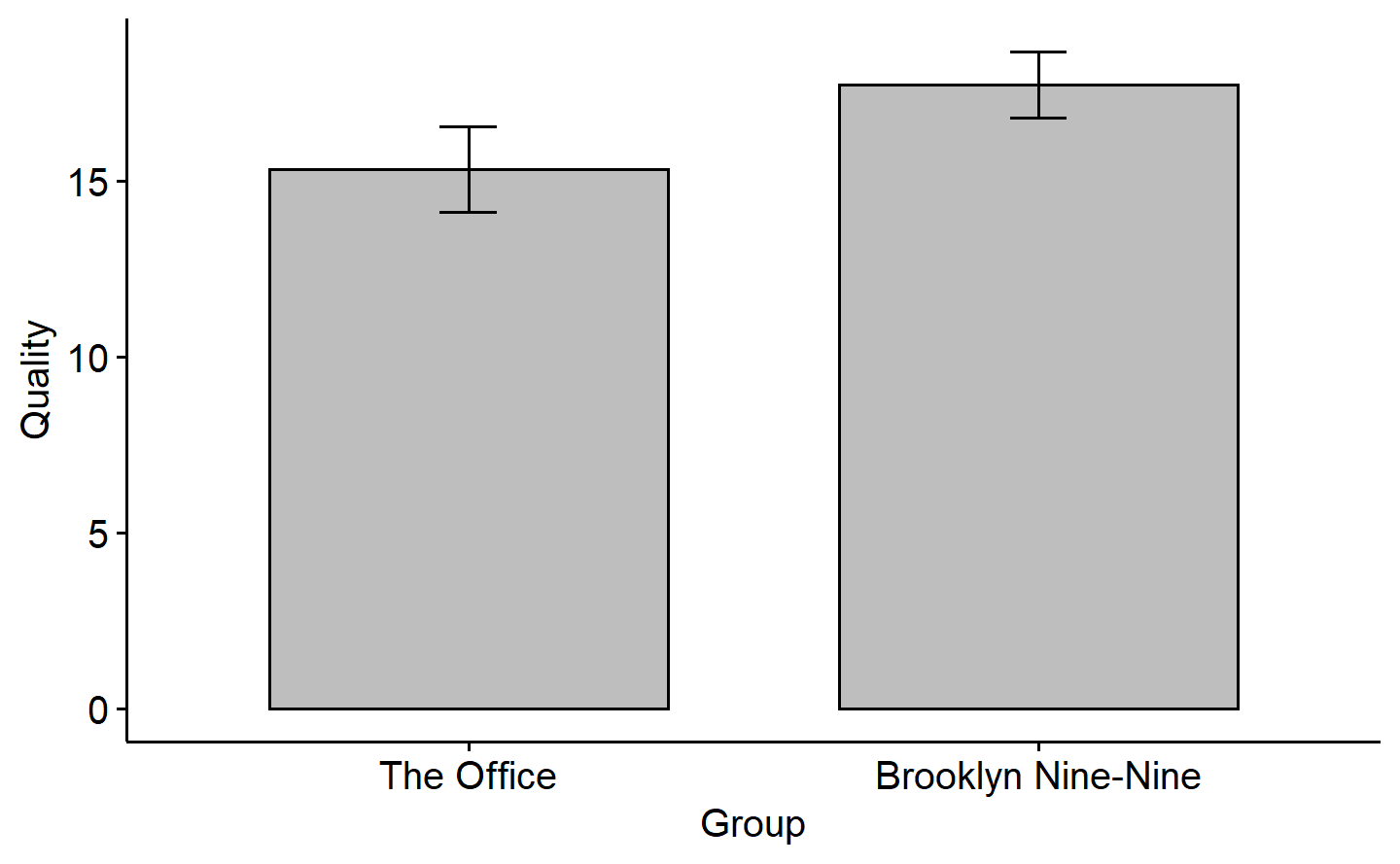
*Figure* *9.*  Bar-plot comparing mean sitcom quality between House type and Apartment type sitcoms

ggpubr::ggbarplot(data = rbind.data.frame(data\_q3\_friends,data\_q3\_seinfeld),x="sitcom",y="sitcom\_quality", add = "mean\_ci", xlab="Group", ylab="Quality", fill="grey")



*Figure* *10.*  Bar-plot comparing mean sitcom quality between Seinfeld and Friends

ggpubr::ggbarplot(data = rbind.data.frame(data\_q3\_office,data\_q3\_b99),x="sitcom",y="sitcom\_quality", add = "mean\_ci", xlab="Group", ylab="Quality", fill="grey")



*Figure* *11.*  Bar-plot comparing mean sitcom quality between The Office and Brooklyn Nine-Nine

## Question 3d

*A graduate student who participated in your study liked your sitcom research idea and wants to use your same study design but doesn’t like the planned comparisons you chose and wants you to provide suggestions on what they could do (apparently they don’t want to do any work). Without repeating any of the three planned comparisons you originally did for your hypotheses, what are TWO complete/full orthogonal sets (distinct from each other as well), in both words and contrast codes, that this other graduate student could do? (2 marks)*

*Note 1. The actual comparisons themselves need to be different, not just the codes. For example, (1,-1,0) is not different from (2,-2,0) or (-1,1,0).*

*Note 2. You don’t need any kind of home/apartment theory to justify your comparisons here. It is just about orthogonality!*

Answers here.

First set:

contrast 1 = [-1,-1,-1,-1,4] (Brooklyn Nine-Nine is better than all other sitcoms)

contrast 2 = [-1,-1,-1,3,0] (The Office is better than all Home type sitcoms)

contrast 3 = [-1,-1,2,0,0] (Seinfeld is better than Modern Family and Friends)

contrast 4 = [-1,1,0,0,0] (Friends is better than Modern Family)

Second set:

contrast 1 = [-1,-1,4,-1,-1] (Seinfeld is better than all other sitcoms)

contrast 2 = [-1,-1,0,3,-1] (The Office is better than Modern Family, Friends, and Brooklyn Nine-Nine)

contrast 3 = [-1,2,0,0,-1] (Friends is better than Modern Family and Brooklyn Nine-Nine)

contrast 4 = [-1,0,0,0,1] (Brooklyn Nine-Nine is better than Modern Family)

# Bonus

*You have two planned comparisons’ vectors: [-1,-1,2] and [-1,0,1]. What is the angle between these two vectors (in degrees) and what does this mean for their variance? (.5 BONUS mark)*

$$
Let\ \mathbf{A} = [-1,-1,2]\ and\ \mathbf{B}=[-1,0,1].\\
Then,\ the \ angle \ (\theta) \ between \ \mathbf{A} \ and \ \mathbf{B} \ = cos^{-1}(\frac{\mathbf{A}\cdot \mathbf{B}}{|\mathbf{A}|\*|\mathbf{B}|}).\\
$$

The fact that the angle between the the two contrast vectors is not radians means that they are not orthogonal. This means that the variance explained by the two contrasts is overlapping, and not independent.

# All R Output

mget(ls())  
#> $agree\_emm  
#> group emmean SE df lower.CL upper.CL  
#> DC 13.9 0.777 114 12.4 15.5  
#> Disney 14.7 0.764 114 13.2 16.2  
#> Marvel 15.8 0.764 114 14.3 17.3  
#> Smurfs 20.9 0.777 114 19.3 22.4  
#>   
#> Confidence level used: 0.95   
#>   
#> $agree\_model  
#> Call:  
#> aov(formula = agreeableness ~ group, data = data\_q1\_filtered)  
#>   
#> Terms:  
#> group Residuals  
#> Sum of Squares 847.4213 1995.9770  
#> Deg. of Freedom 3 114  
#>   
#> Residual standard error: 4.184324  
#> Estimated effects may be unbalanced  
#>   
#> $c1  
#> [1] 1.0 1.0 1.0 -1.5 -1.5  
#>   
#> $c2  
#> [1] -2 1 1 0 0  
#>   
#> $c3  
#> [1] 0 -1 1 0 0  
#>   
#> $c4  
#> [1] 0 0 0 -1 1  
#>   
#> $contrast\_sum  
#> c1 c2 c3 c4   
#> 6 4 2 2   
#>   
#> $data\_q1  
#> # A tibble: 120 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 1 Marvel 17  
#> 2 2 DC 12  
#> 3 3 Disney 22  
#> 4 4 Smurfs 340  
#> 5 5 Marvel 16  
#> 6 6 DC 13  
#> 7 7 Disney 14  
#> 8 8 Smurfs 17  
#> 9 9 Marvel 15  
#> 10 10 DC 14  
#> # … with 110 more rows  
#>   
#> $data\_q1\_dc  
#> # A tibble: 30 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 2 DC 12  
#> 2 6 DC 13  
#> 3 10 DC 14  
#> 4 14 DC 8  
#> 5 18 DC 14  
#> 6 22 DC 12  
#> 7 26 DC 15  
#> 8 30 DC 10  
#> 9 34 DC 18  
#> 10 38 DC 10  
#> # … with 20 more rows  
#>   
#> $data\_q1\_disney  
#> # A tibble: 30 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 3 Disney 22  
#> 2 7 Disney 14  
#> 3 11 Disney 15  
#> 4 15 Disney 16  
#> 5 19 Disney 15  
#> 6 23 Disney 15  
#> 7 27 Disney 8  
#> 8 31 Disney 15  
#> 9 35 Disney 14  
#> 10 39 Disney 13  
#> # … with 20 more rows  
#>   
#> $data\_q1\_filtered  
#> # A tibble: 118 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 1 Marvel 17  
#> 2 2 DC 12  
#> 3 3 Disney 22  
#> 4 5 Marvel 16  
#> 5 6 DC 13  
#> 6 7 Disney 14  
#> 7 8 Smurfs 17  
#> 8 9 Marvel 15  
#> 9 10 DC 14  
#> 10 11 Disney 15  
#> # … with 108 more rows  
#>   
#> $data\_q1\_filtered\_dc  
#> # A tibble: 29 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 2 DC 12  
#> 2 6 DC 13  
#> 3 10 DC 14  
#> 4 14 DC 8  
#> 5 18 DC 14  
#> 6 22 DC 12  
#> 7 26 DC 15  
#> 8 30 DC 10  
#> 9 34 DC 18  
#> 10 38 DC 10  
#> # … with 19 more rows  
#>   
#> $data\_q1\_filtered\_disney  
#> # A tibble: 30 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 3 Disney 22  
#> 2 7 Disney 14  
#> 3 11 Disney 15  
#> 4 15 Disney 16  
#> 5 19 Disney 15  
#> 6 23 Disney 15  
#> 7 27 Disney 8  
#> 8 31 Disney 15  
#> 9 35 Disney 14  
#> 10 39 Disney 13  
#> # … with 20 more rows  
#>   
#> $data\_q1\_filtered\_marvel  
#> # A tibble: 30 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 1 Marvel 17  
#> 2 5 Marvel 16  
#> 3 9 Marvel 15  
#> 4 13 Marvel 12  
#> 5 17 Marvel 17  
#> 6 21 Marvel 5  
#> 7 25 Marvel 12  
#> 8 29 Marvel 27  
#> 9 33 Marvel 14  
#> 10 37 Marvel 18  
#> # … with 20 more rows  
#>   
#> $data\_q1\_filtered\_smurfs  
#> # A tibble: 29 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 8 Smurfs 17  
#> 2 12 Smurfs 15  
#> 3 16 Smurfs 12  
#> 4 20 Smurfs 28  
#> 5 24 Smurfs 25  
#> 6 28 Smurfs 15  
#> 7 32 Smurfs 24  
#> 8 36 Smurfs 23  
#> 9 40 Smurfs 22  
#> 10 44 Smurfs 23  
#> # … with 19 more rows  
#>   
#> $data\_q1\_marvel  
#> # A tibble: 30 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 1 Marvel 17  
#> 2 5 Marvel 16  
#> 3 9 Marvel 15  
#> 4 13 Marvel 12  
#> 5 17 Marvel 17  
#> 6 21 Marvel 5  
#> 7 25 Marvel 12  
#> 8 29 Marvel 27  
#> 9 33 Marvel 14  
#> 10 37 Marvel 18  
#> # … with 20 more rows  
#>   
#> $data\_q1\_smurfs  
#> # A tibble: 30 × 3  
#> superhero group agreeableness  
#> <dbl> <chr> <dbl>  
#> 1 4 Smurfs 340  
#> 2 8 Smurfs 17  
#> 3 12 Smurfs 15  
#> 4 16 Smurfs 12  
#> 5 20 Smurfs 28  
#> 6 24 Smurfs 25  
#> 7 28 Smurfs 15  
#> 8 32 Smurfs 24  
#> 9 36 Smurfs 23  
#> 10 40 Smurfs 22  
#> # … with 20 more rows  
#>   
#> $data\_q3  
#> # A tibble: 125 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <fct> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 1 Modern… House 12 5 3 4 Home … House… <NA>   
#> 2 2 Modern… House 18 7 6 5 Home … House… <NA>   
#> 3 3 Modern… House 21 12 3 6 Home … House… <NA>   
#> 4 4 Modern… House 15 5 5 5 Home … House… <NA>   
#> 5 5 Modern… House 16 7 3 6 Home … House… <NA>   
#> 6 6 Modern… House 18 9 4 5 Home … House… <NA>   
#> 7 7 Modern… House 20 11 3 6 Home … House… <NA>   
#> 8 8 Modern… House 13 4 5 4 Home … House… <NA>   
#> 9 9 Modern… House 17 9 3 5 Home … House… <NA>   
#> 10 10 Modern… House 12 7 2 3 Home … House… <NA>   
#> # … with 115 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_apartment  
#> # A tibble: 50 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 26 Friends Apartm… 20 8 6 6 Home … Apart… Frien…  
#> 2 27 Friends Apartm… 15 5 7 3 Home … Apart… Frien…  
#> 3 28 Friends Apartm… 23 10 7 6 Home … Apart… Frien…  
#> 4 29 Friends Apartm… 21 9 7 5 Home … Apart… Frien…  
#> 5 30 Friends Apartm… 19 6 7 6 Home … Apart… Frien…  
#> 6 31 Friends Apartm… 22 10 7 5 Home … Apart… Frien…  
#> 7 32 Friends Apartm… 15 6 7 2 Home … Apart… Frien…  
#> 8 33 Friends Apartm… 23 11 6 6 Home … Apart… Frien…  
#> 9 34 Friends Apartm… 19 9 7 3 Home … Apart… Frien…  
#> 10 35 Friends Apartm… 24 12 7 5 Home … Apart… Frien…  
#> # … with 40 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_b99  
#> # A tibble: 25 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 101 Brookl… Workpl… 20 9 6 5 Workp… <NA> <NA>   
#> 2 102 Brookl… Workpl… 18 7 5 6 Workp… <NA> <NA>   
#> 3 103 Brookl… Workpl… 17 7 6 4 Workp… <NA> <NA>   
#> 4 104 Brookl… Workpl… 19 9 5 5 Workp… <NA> <NA>   
#> 5 105 Brookl… Workpl… 17 8 5 4 Workp… <NA> <NA>   
#> 6 106 Brookl… Workpl… 20 10 5 5 Workp… <NA> <NA>   
#> 7 107 Brookl… Workpl… 22 11 5 6 Workp… <NA> <NA>   
#> 8 108 Brookl… Workpl… 16 8 2 6 Workp… <NA> <NA>   
#> 9 109 Brookl… Workpl… 18 6 5 7 Workp… <NA> <NA>   
#> 10 110 Brookl… Workpl… 20 9 5 6 Workp… <NA> <NA>   
#> # … with 15 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_friends  
#> # A tibble: 25 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 26 Friends Apartm… 20 8 6 6 Home … Apart… Frien…  
#> 2 27 Friends Apartm… 15 5 7 3 Home … Apart… Frien…  
#> 3 28 Friends Apartm… 23 10 7 6 Home … Apart… Frien…  
#> 4 29 Friends Apartm… 21 9 7 5 Home … Apart… Frien…  
#> 5 30 Friends Apartm… 19 6 7 6 Home … Apart… Frien…  
#> 6 31 Friends Apartm… 22 10 7 5 Home … Apart… Frien…  
#> 7 32 Friends Apartm… 15 6 7 2 Home … Apart… Frien…  
#> 8 33 Friends Apartm… 23 11 6 6 Home … Apart… Frien…  
#> 9 34 Friends Apartm… 19 9 7 3 Home … Apart… Frien…  
#> 10 35 Friends Apartm… 24 12 7 5 Home … Apart… Frien…  
#> # … with 15 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_home  
#> # A tibble: 75 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 1 Modern… House 12 5 3 4 Home … House… <NA>   
#> 2 2 Modern… House 18 7 6 5 Home … House… <NA>   
#> 3 3 Modern… House 21 12 3 6 Home … House… <NA>   
#> 4 4 Modern… House 15 5 5 5 Home … House… <NA>   
#> 5 5 Modern… House 16 7 3 6 Home … House… <NA>   
#> 6 6 Modern… House 18 9 4 5 Home … House… <NA>   
#> 7 7 Modern… House 20 11 3 6 Home … House… <NA>   
#> 8 8 Modern… House 13 4 5 4 Home … House… <NA>   
#> 9 9 Modern… House 17 9 3 5 Home … House… <NA>   
#> 10 10 Modern… House 12 7 2 3 Home … House… <NA>   
#> # … with 65 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_modfam  
#> # A tibble: 25 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 1 Modern… House 12 5 3 4 Home … House… <NA>   
#> 2 2 Modern… House 18 7 6 5 Home … House… <NA>   
#> 3 3 Modern… House 21 12 3 6 Home … House… <NA>   
#> 4 4 Modern… House 15 5 5 5 Home … House… <NA>   
#> 5 5 Modern… House 16 7 3 6 Home … House… <NA>   
#> 6 6 Modern… House 18 9 4 5 Home … House… <NA>   
#> 7 7 Modern… House 20 11 3 6 Home … House… <NA>   
#> 8 8 Modern… House 13 4 5 4 Home … House… <NA>   
#> 9 9 Modern… House 17 9 3 5 Home … House… <NA>   
#> 10 10 Modern… House 12 7 2 3 Home … House… <NA>   
#> # … with 15 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_office  
#> # A tibble: 25 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 76 The Of… Workpl… 10 2 5 3 Workp… <NA> <NA>   
#> 2 77 The Of… Workpl… 14 2 7 5 Workp… <NA> <NA>   
#> 3 78 The Of… Workpl… 12 3 4 5 Workp… <NA> <NA>   
#> 4 79 The Of… Workpl… 17 6 7 4 Workp… <NA> <NA>   
#> 5 80 The Of… Workpl… 16 3 6 7 Workp… <NA> <NA>   
#> 6 81 The Of… Workpl… 18 8 4 6 Workp… <NA> <NA>   
#> 7 82 The Of… Workpl… 14 2 7 5 Workp… <NA> <NA>   
#> 8 83 The Of… Workpl… 14 2 5 7 Workp… <NA> <NA>   
#> 9 84 The Of… Workpl… 19 5 9 5 Workp… <NA> <NA>   
#> 10 85 The Of… Workpl… 19 7 6 6 Workp… <NA> <NA>   
#> # … with 15 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_seinfeld  
#> # A tibble: 25 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 51 Seinfe… Apartm… 13 2 6 5 Home … Apart… Seinf…  
#> 2 52 Seinfe… Apartm… 24 11 7 6 Home … Apart… Seinf…  
#> 3 53 Seinfe… Apartm… 16 5 5 6 Home … Apart… Seinf…  
#> 4 54 Seinfe… Apartm… 15 6 5 4 Home … Apart… Seinf…  
#> 5 55 Seinfe… Apartm… 13 2 6 5 Home … Apart… Seinf…  
#> 6 56 Seinfe… Apartm… 21 11 6 4 Home … Apart… Seinf…  
#> 7 57 Seinfe… Apartm… 13 3 4 6 Home … Apart… Seinf…  
#> 8 58 Seinfe… Apartm… 10 3 2 5 Home … Apart… Seinf…  
#> 9 59 Seinfe… Apartm… 15 5 4 6 Home … Apart… Seinf…  
#> 10 60 Seinfe… Apartm… 17 8 5 4 Home … Apart… Seinf…  
#> # … with 15 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $data\_q3\_work  
#> # A tibble: 50 × 10  
#> grad\_student sitcom sitco…¹ sitco…² laughs binge snacks q2a\_h1 q2a\_h2 q2a\_h3  
#> <dbl> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <chr> <chr> <chr>   
#> 1 76 The Of… Workpl… 10 2 5 3 Workp… <NA> <NA>   
#> 2 77 The Of… Workpl… 14 2 7 5 Workp… <NA> <NA>   
#> 3 78 The Of… Workpl… 12 3 4 5 Workp… <NA> <NA>   
#> 4 79 The Of… Workpl… 17 6 7 4 Workp… <NA> <NA>   
#> 5 80 The Of… Workpl… 16 3 6 7 Workp… <NA> <NA>   
#> 6 81 The Of… Workpl… 18 8 4 6 Workp… <NA> <NA>   
#> 7 82 The Of… Workpl… 14 2 7 5 Workp… <NA> <NA>   
#> 8 83 The Of… Workpl… 14 2 5 7 Workp… <NA> <NA>   
#> 9 84 The Of… Workpl… 19 5 9 5 Workp… <NA> <NA>   
#> 10 85 The Of… Workpl… 19 7 6 6 Workp… <NA> <NA>   
#> # … with 40 more rows, and abbreviated variable names ¹​sitcom\_type,  
#> # ²​sitcom\_quality  
#>   
#> $params  
#> $params$firstname  
#> [1] "Yudhajit"  
#>   
#> $params$lastname  
#> [1] "Ain"  
#>   
#> $params$studentid  
#> [1] 30182745  
#>   
#> $params$TAs  
#> [1] "Christopher Davie & Benjamin Moon"  
#>   
#> $params$assignment  
#> [1] 2  
#>   
#> $params$show\_output  
#> [1] TRUE  
#>   
#>   
#> $sitcom\_contrast  
#> # A tibble: 4 × 9  
#> term contrast null.…¹ estim…² std.e…³ df stati…⁴ adj.p…⁵ effec…⁶  
#> <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
#> 1 sitcom Home - Workplace 0 7.18 1.67 120 4.29 1.47e-4 0.783  
#> 2 sitcom Apartment - House 0 0.880 1.50 120 0.588 1 e+0 0.144  
#> 3 sitcom Seinfeld - Frien… 0 -4.32 0.865 120 -5.00 8.07e-6 -1.41   
#> 4 sitcom Brooklyn Nine-Ni… 0 -0.600 0.865 120 -0.694 1 e+0 -0.196  
#> # … with abbreviated variable names ¹​null.value, ²​estimate, ³​std.error,  
#> # ⁴​statistic, ⁵​adj.p.value, ⁶​effect.size  
#>   
#> $sitcom\_emm  
#> sitcom emmean SE df lower.CL upper.CL  
#> Brooklyn Nine-Nine 17.7 0.611 120 16.5 18.9  
#> Friends 20.3 0.611 120 19.1 21.5  
#> Modern Family 16.0 0.611 120 14.8 17.2  
#> Seinfeld 15.9 0.611 120 14.7 17.1  
#> The Office 15.3 0.611 120 14.1 16.5  
#>   
#> Confidence level used: 0.95   
#>   
#> $sitcom\_model  
#> Call:  
#> aov(formula = sitcom\_quality ~ sitcom, data = data\_q3)  
#>   
#> Terms:  
#> sitcom Residuals  
#> Sum of Squares 412.848 1121.760  
#> Deg. of Freedom 4 120  
#>   
#> Residual standard error: 3.05745  
#> Estimated effects may be unbalanced