study1 vis

2023-02-06

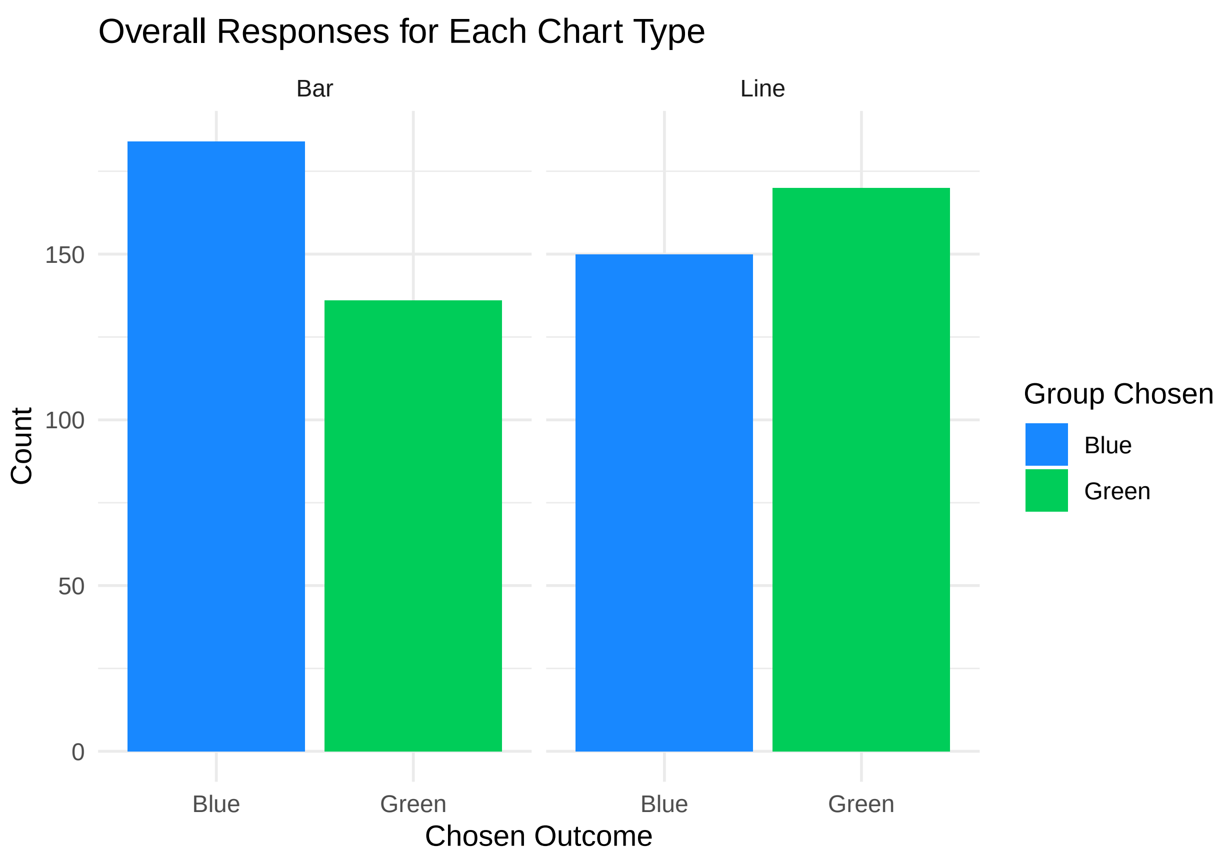
Setup

In the first section of this file, I upload and clean the data, including checks after data manipulation to ensure that the dataframe has updated properly. This section is omitted from the knit version but is present in the .Rmd file.

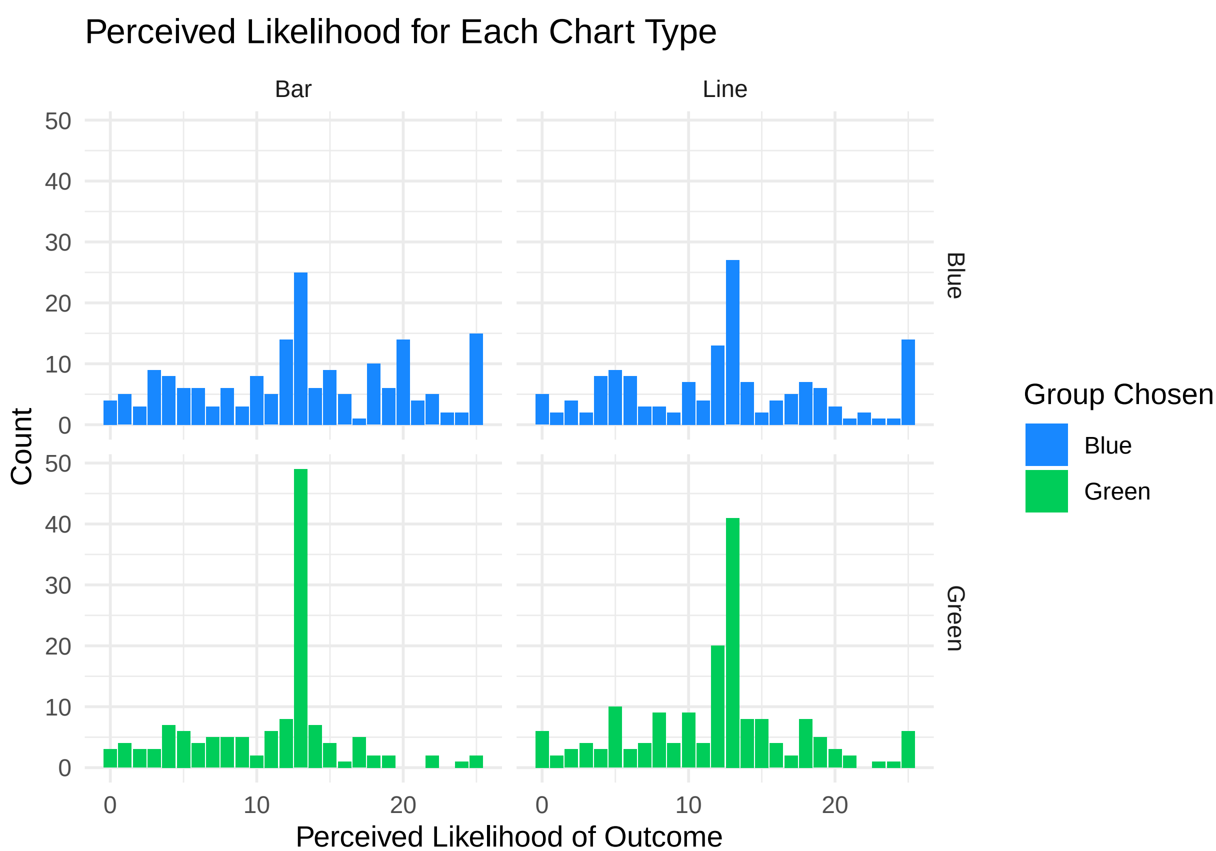
Language in this file may differ from that in the paper.

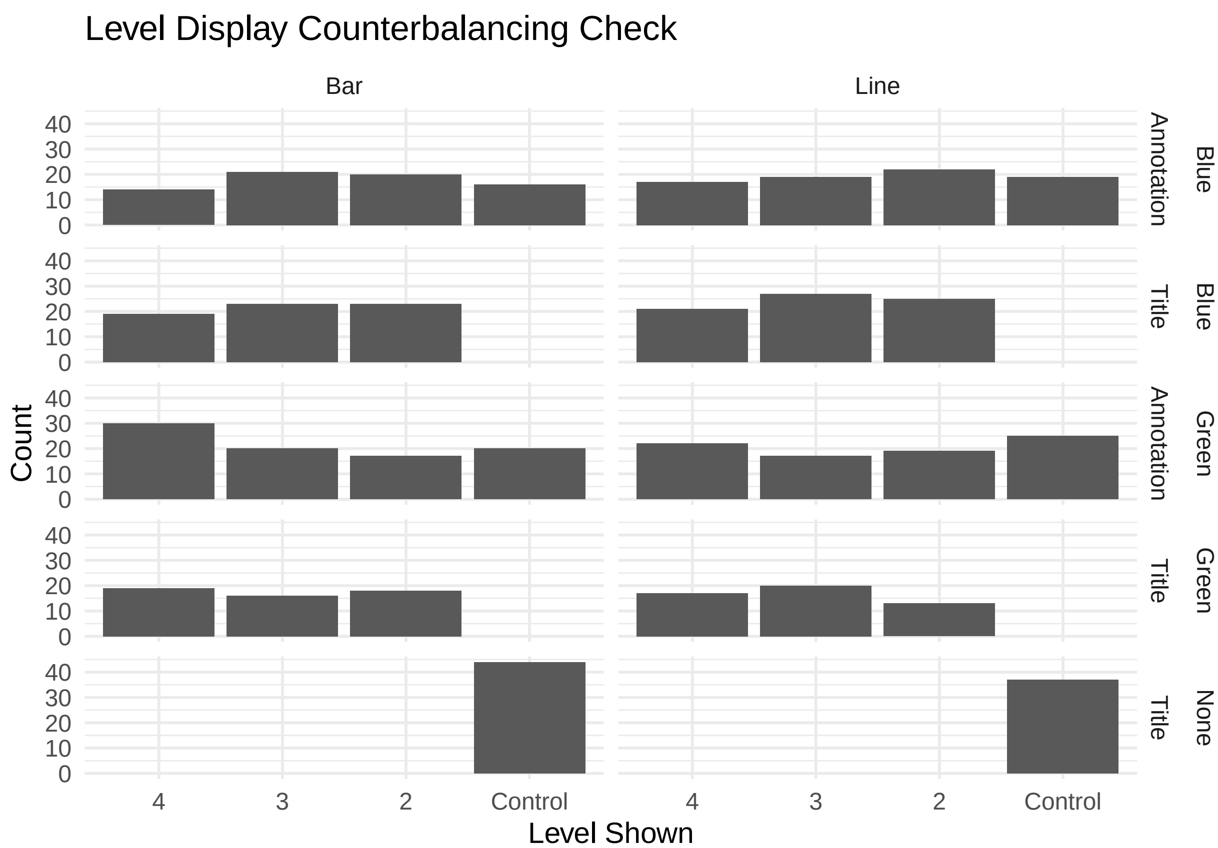
```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: Matrix
##
##
## Attaching package: 'Matrix'
##
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
##
## Loading required package: carData
##
##
## Attaching package: 'car'
##
##
## The following object is masked from 'package:dplyr':
##
##     recode
##
##
## The following object is masked from 'package:purrr':
##
##     some
##
```

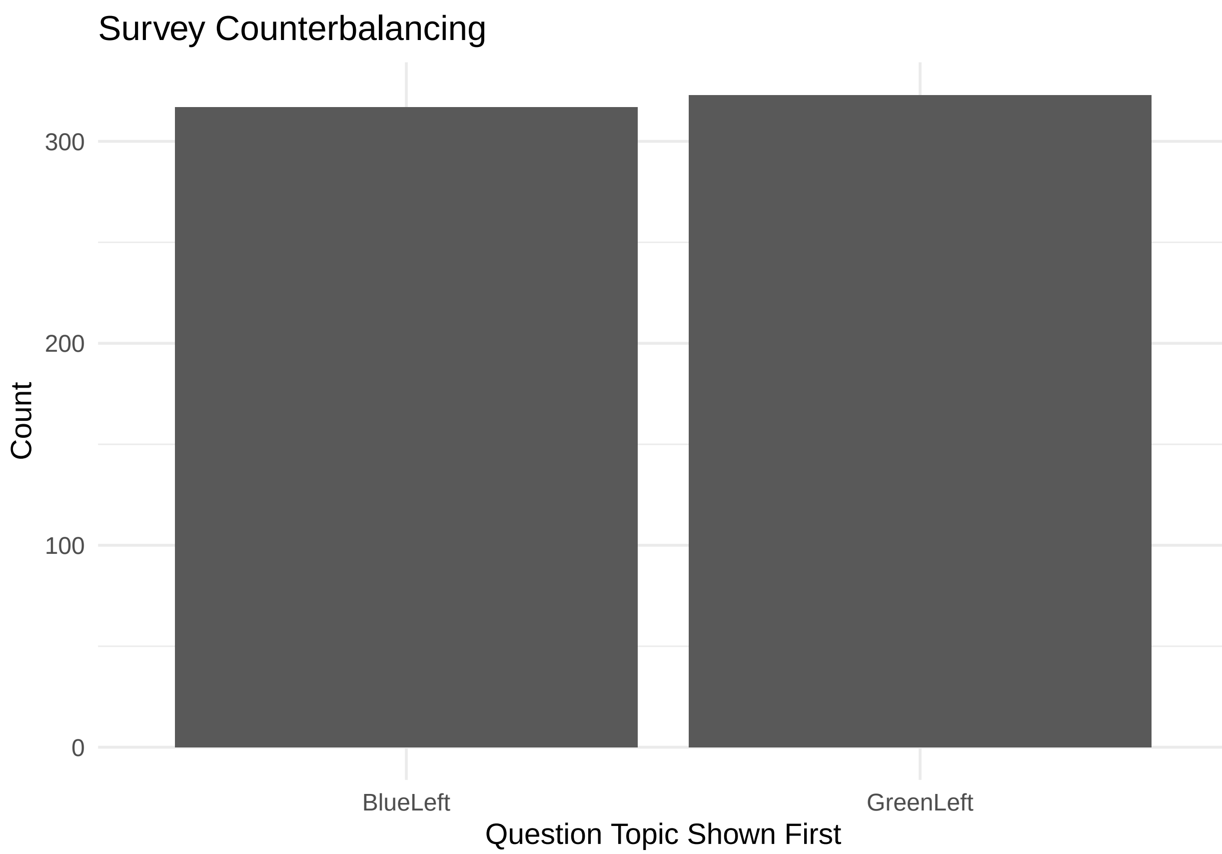
```
##
## Registered S3 methods overwritten by 'FSA':
##   method       from
##   confint.boot car
##   hist.boot    car
##
## ## FSA v0.9.4. See citation('FSA') if used in publication.
## ## Run fishR() for related website and fishR('IFAR') for related book.
##
##
## Attaching package: 'FSA'
##
##
## The following object is masked from 'package:car':
##
##     bootCase
##
##
##
## Attaching package: 'rstatix'
##
##
## The following object is masked from 'package:stats':
##
##     filter
```



```
## `summarise()` has grouped output by 'ChartType'. You can override using the
## `.groups` argument.

## # A tibble: 4 x 3
## # Groups:   ChartType [2]
##   ChartType chosen_outcome     n
##   <chr>     <chr>          <int>
## 1 Bar       Blue             184
## 2 Bar       Green            136
## 3 Line      Blue             150
## 4 Line      Green            170
```




Level Display Counterbalancing Check
Bar
Line
Annotation
Blue
Title
Blue
Annotation
Green
Title
Green
Title
None
Count
40
30
20
10
0
4
3
2
Control
Level Shown



In this dataset, we have 8 columns to examine:

1. `Order` = randomly assigned order of the options displayed
2. `Level` = the level of text displayed to the participant
3. `Position` = the position of text displayed to the participant
4. `full_condition` = the level & position of text displayed to the participant
5. `Slant` = group supported by the randomly assigned annotation
6. `ChartType` = the chart type displayed to the participant
7. `treatment` = indication of whether the participant viewed a treatment or control variant
8. `outcome_confidence_abs` = absolute value of the participant's confidence rating in the outcome, transformed to account for counterbalancing in the survey conditions
9. `outcome_aligned` = indication of whether the participants' response was aligned with the outcome indicated in the text provided
10. `author_confidence_abs` = absolute value of the participant's confidence rating in the author's leading, transformed to account for counterbalancing in the survey conditions
11. `author_aligned` = indication of whether the participants' response was aligned with the outcome (and therefore author leaning) indicated in the text provided

Test for Normal Distribution

The first thing to do with the data is to determine whether the distributions, adjusted to account for counterbalancing in the survey conditions (e.g., Order and Slant), are normally distributed. This will indicate the kind of statistical testing we can perform to determine the effect of viewing the text.

First, we will plot the distributions. To fully determine if these distributions are normal, we also conduct a series of Shapiro-Wilk tests for normality (S-W tests). These subsets are as follows, organized according to the comparisons which will be made in statistical analysis:

Comparison Set 1:

- Bar Chart, Outcome Aligned, Treatment Condition
- Bar Chart, Outcome Unaligned, Treatment Condition
- Bar Chart, Control Condition

Comparison Set 2:

- Line Chart, Outcome Aligned, Treatment Condition
- Line Chart, Outcome Unaligned, Treatment Condition
- Line Chart, Control Condition

Comparison Set 3:

- Bar Chart, Author Aligned, Treatment Condition
- Bar Chart, Author Unaligned, Treatment Condition
- Bar Chart, Control Condition

Comparison Set 4:

- Line Chart, Author Aligned, Treatment Condition
- Line Chart, Author Unaligned, Treatment Condition
- Line Chart, Control Condition

We check each of these distributions for normality. If any of the distributions within a comparison set fail the normal distribution check, we will use non-parametric testing for all comparisons made within that set.

```
# - Bar Chart, Outcome Aligned, Treatment Condition
bar_aligned_treatment_outcome = subset(bar, outcome_aligned == "Aligned" & treatment == "Treatment")
bar_aligned_treatment_outcome_plot =
  ggplot(bar_aligned_treatment_outcome, aes(x = outcome_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = a)+
  labs(
    #title = "Outcome Confidence: Bar Chart, Treatment Condition, Aligned with Slant"
    title = "Outcome Confidence:\n Bar, Aligned"
  )
shapiro.test(bar_aligned_treatment_outcome$outcome_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  bar_aligned_treatment_outcome$outcome_confidence_abs
## W = 0.96562, p-value = 0.001574
```

```
# - Bar Chart, Outcome Unaligned, Treatment Condition
bar_unaligned_treatment_outcome = subset(bar, outcome_aligned == "Unaligned" & treatment == "Treatment")
bar_unaligned_treatment_outcome_plot =
  ggplot(bar_unaligned_treatment_outcome, aes(x = outcome_confidence_abs, y = after_stat(count)))+
```

```r
  geom_histogram(binwidth = 1, fill = u)+
  labs(
    #title = "Outcome Confidence: Bar Chart, Treatment Condition, Unaligned with Slant"
    title = "Bar, Unaligned"
  )
shapiro.test(bar_unaligned_treatment_outcome$outcome_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  bar_unaligned_treatment_outcome$outcome_confidence_abs
## W = 0.93476, p-value = 7.339e-05
```

```r
# - Bar Chart, Control Condition
bar_control = subset(bar, treatment == "Control")
bar_control_outcome_plot =
  ggplot(bar_control, aes(x = outcome_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    #title = "Outcome Confidence: Bar Chart, Control Condition"
    title = "Bar, Control"
  )
shapiro.test(bar_control$outcome_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  bar_control$outcome_confidence_abs
## W = 0.97026, p-value = 0.05899
```

```r
# - Line Chart, Outcome Aligned, Treatment Condition
line_aligned_treatment_outcome = subset(line, outcome_aligned == "Aligned" & treatment == "Treatment")
line_aligned_treatment_outcome_plot =
  ggplot(line_aligned_treatment_outcome, aes(x = outcome_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = a)+
  labs(
    #title = "Outcome Confidence: Line Chart, Treatment Condition, Aligned with Slant"
    title = "Outcome Confidence:\n Line, Aligned"
  )
shapiro.test(line_aligned_treatment_outcome$outcome_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  line_aligned_treatment_outcome$outcome_confidence_abs
## W = 0.95041, p-value = 0.000217
```

```r
# - Line Chart, Outcome Unaligned, Treatment Condition
line_unaligned_treatment_outcome = subset(line, outcome_aligned == "Unaligned" & treatment == "Treatmen
line_unaligned_treatment_outcome_plot =
  ggplot(line_unaligned_treatment_outcome, aes(x = outcome_confidence_abs, y = after_stat(count)))+
```

```r
  geom_histogram(binwidth = 1, fill = u)+
  labs(
    #title = "Outcome Confidence: Line Chart, Treatment Condition, Unaligned with Slant"
    title = "Line, Unaligned"
  )
shapiro.test(line_unaligned_treatment_outcome$outcome_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  line_unaligned_treatment_outcome$outcome_confidence_abs
## W = 0.96405, p-value = 0.003016
```

```r
# - Line Chart, Control Condition
line_control = subset(line, treatment == "Control")
line_control_outcome_plot =
  ggplot(line_control, aes(x = outcome_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    #title = "Outcome Confidence: Line Chart, Control Condition"
    title = "Line, Control"
  )
shapiro.test(line_control$outcome_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  line_control$outcome_confidence_abs
## W = 0.96416, p-value = 0.02296
```

```r
ggarrange(bar_aligned_treatment_outcome_plot,
          bar_unaligned_treatment_outcome_plot,
          bar_control_outcome_plot,
          line_aligned_treatment_outcome_plot,
          line_unaligned_treatment_outcome_plot,
          line_control_outcome_plot,
          ncol = 3, nrow = 2,
          align = "h")
```


```
# - Bar Chart, Author Aligned, Treatment Condition
bar_aligned_treatment_author = subset(bar, author_aligned == "Aligned" & treatment == "Treatment")
bar_aligned_treatment_author_plot =
  ggplot(bar_aligned_treatment_author, aes(x = author_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = a)+
  labs(
    #title = "Author Confidence: Bar Chart, Treatment Condition, Aligned with Slant"
    title = "Author Confidence:\n Bar, Aligned"
  )
shapiro.test(bar_aligned_treatment_author$author_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  bar_aligned_treatment_author$author_confidence_abs
## W = 0.897, p-value = 2.987e-09
```

```
# - Bar Chart, Author Unaligned, Treatment Condition
bar_unaligned_treatment_author = subset(bar, author_aligned == "Unaligned" & treatment == "Treatment")
bar_unaligned_treatment_author_plot =
  ggplot(bar_unaligned_treatment_author, aes(x = author_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = u)+
  labs(
    #title = "Author Confidence: Bar Chart, Treatment Condition, Unaligned with Slant"
    title = "Bar, Unaligned"
```

```r
  )
shapiro.test(bar_unaligned_treatment_author$author_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  bar_unaligned_treatment_author$author_confidence_abs
## W = 0.61683, p-value = 7.201e-13
```

```r
# - Bar Chart, Control Condition
bar_control = subset(bar, treatment == "Control")
bar_control_author_plot =
  ggplot(bar_control, aes(x = author_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    #title = "Author Confidence: Bar Chart, Control Condition"
    title = "Bar, Control"
  )
shapiro.test(bar_control$author_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  bar_control$author_confidence_abs
## W = 0.65236, p-value = 1.817e-12
```

```r
# - Line Chart, Author Aligned, Treatment Condition
line_aligned_treatment_author = subset(line, author_aligned == "Aligned" & treatment == "Treatment")
line_aligned_treatment_author_plot =
  ggplot(line_aligned_treatment_author, aes(x = author_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = a)+
  labs(
    #title = "Author Confidence: Line Chart, Treatment Condition, Aligned with Slant"
    title = "Line, Aligned"
  )
shapiro.test(line_aligned_treatment_author$author_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  line_aligned_treatment_author$author_confidence_abs
## W = 0.87358, p-value = 7.269e-10
```

```r
# - Line Chart, Author Unaligned, Treatment Condition
line_unaligned_treatment_author = subset(line, author_aligned == "Unaligned" & treatment == "Treatment")
line_unaligned_treatment_author_plot =
  ggplot(line_unaligned_treatment_author, aes(x = author_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = u)+
  labs(
    # title = "Author Confidence: Line Chart, Treatment Condition, Unaligned with Slant"
    title = "Line, Unaligned"
  )
shapiro.test(line_unaligned_treatment_author$author_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  line_unaligned_treatment_author$author_confidence_abs
## W = 0.7109, p-value = 3.627e-12
```

```
# - Line Chart, Control Condition
line_control = subset(line, treatment == "Control")
line_control_author_plot =
  ggplot(line_control, aes(x = author_confidence_abs, y = after_stat(count)))+
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    #title = "Author Confidence: Line Chart, Control Condition"
    title = "Line, Control"
  )
shapiro.test(line_control$author_confidence_abs)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  line_control$author_confidence_abs
## W = 0.76301, p-value = 4.181e-10
```

```
ggarrange(bar_aligned_treatment_author_plot,
          bar_unaligned_treatment_author_plot,
          bar_control_author_plot,
          line_aligned_treatment_author_plot,
          line_unaligned_treatment_author_plot,
          line_control_author_plot,
          ncol = 3, nrow = 2,
          align = "h")
```

Outcome of testing:

None of the distributions pass the test for a normal distribution. For this reason, we continue with non-parametric tests only.

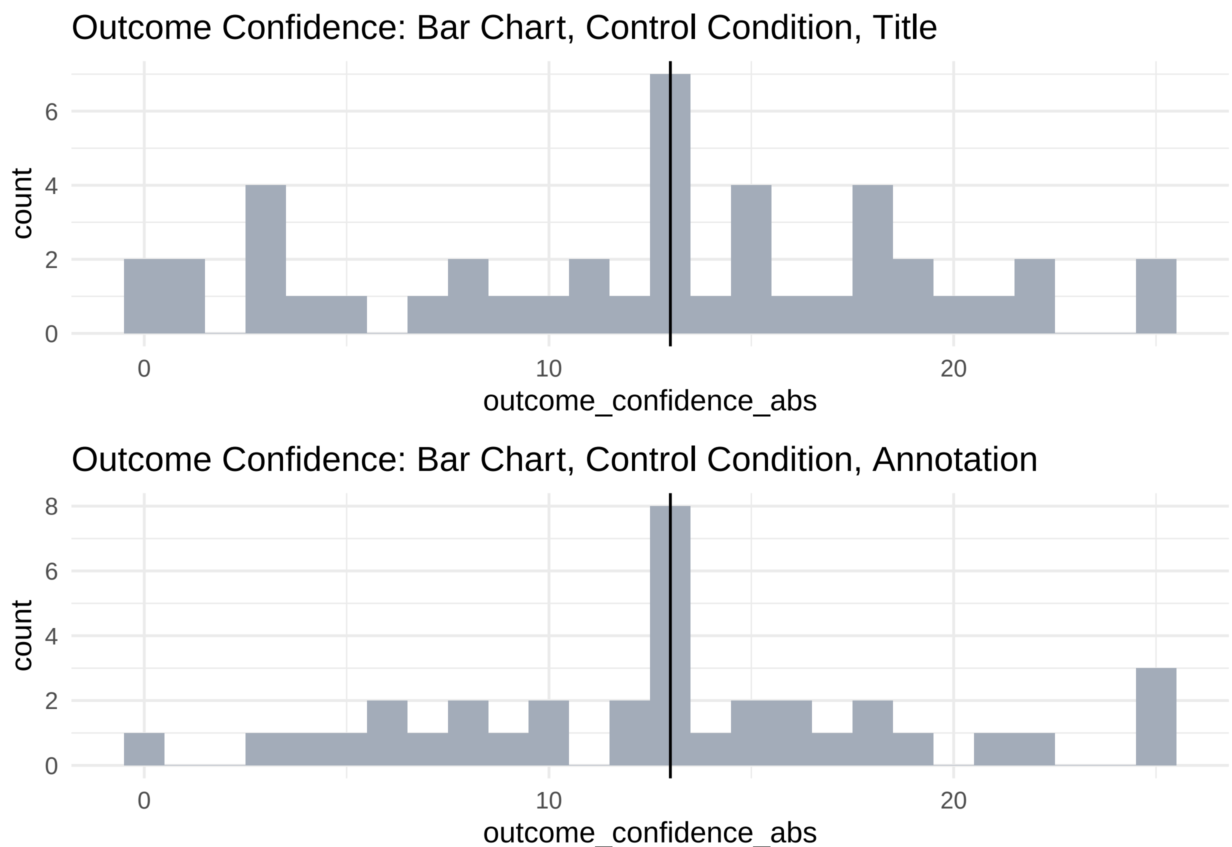
Tests for Combining Responses

Control Conditions

In this study, we used two different kinds of control conditions. The baseline control condition consisted of only a title. The annotation control condition contained an annotation with the title repeated in the position of the treatment annotations. We now compare these control conditions to evaluate any differences in confidence ratings due to the presence of an unbiased annotation. In other words, does a neutral annotation have an effect in comparison to no text? If we see an effect, we would expect to see that annotations increase the confidence in comparison to the no-annotation condition, in line with what we would expect to a greater extent for more biased or slanted conditions.

```
bar_control_title <-
  ggplot(subset(bar_control, Position == "Title"), aes(x = outcome_confidence_abs, y = after_stat(count
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Bar Chart, Control Condition, Title"
  )
```

```r
bar_control_annot <-
  ggplot(subset(bar_control, Position == "Annotation"), aes(x = outcome_confidence_abs, y = after_stat(c
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Bar Chart, Control Condition, Annotation"
  )

ggarrange(bar_control_title + geom_vline(xintercept = median(subset(bar_control, Position == "Title")$ou
          bar_control_annot + geom_vline(xintercept = median(subset(bar_control, Position == "Annotatio
          ncol = 1)
```

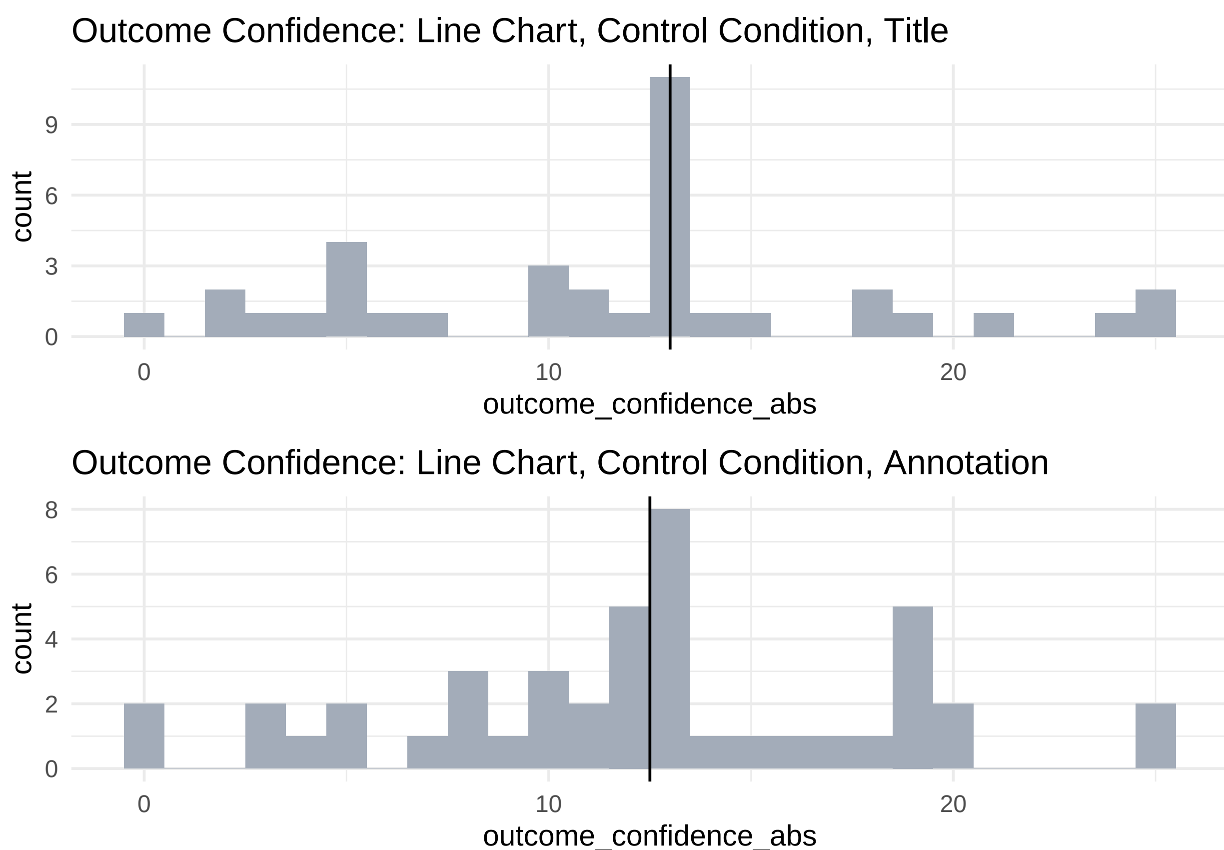



```r
wilcox.test(outcome_confidence_abs ~ as.factor(Position), data = bar_control)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  outcome_confidence_abs by as.factor(Position)
## W = 817.5, p-value = 0.8082
## alternative hypothesis: true location shift is not equal to 0
```

```r
line_control_title <-
  ggplot(subset(line_control, Position == "Title"), aes(x = outcome_confidence_abs, y = after_stat(coun
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Line Chart, Control Condition, Title"
  )

line_control_annot <-
  ggplot(subset(line_control, Position == "Annotation"), aes(x = outcome_confidence_abs, y = after_stat
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Line Chart, Control Condition, Annotation"
  )

ggarrange(line_control_title + geom_vline(xintercept = median(subset(line_control, Position == "Title")
          line_control_annot + geom_vline(xintercept = median(subset(line_control, Position == "Annotat
          ncol = 1)
```



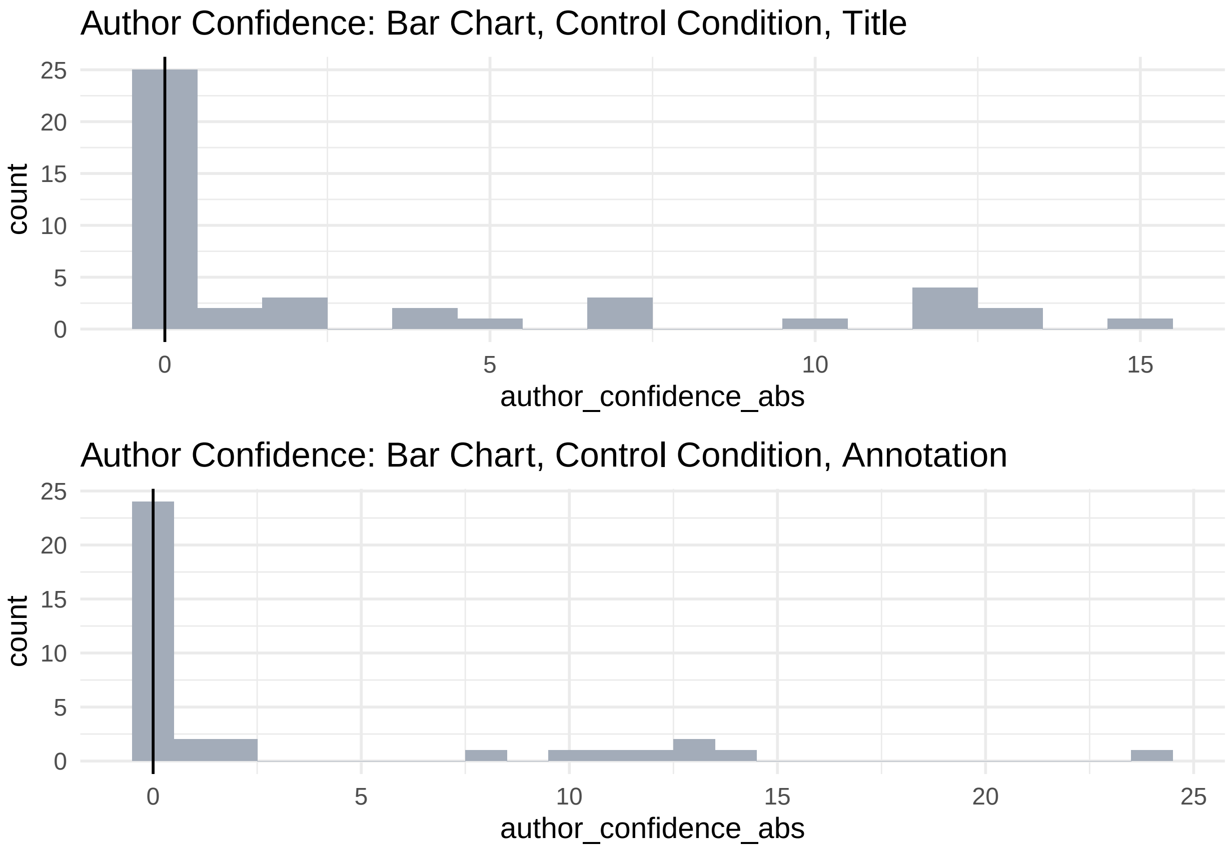

```r
wilcox.test(outcome_confidence_abs ~ as.factor(Position), data = line_control)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
##
```

```
##  Wilcoxon rank sum test with continuity correction
##
## data:  outcome_confidence_abs by as.factor(Position)
## W = 860.5, p-value = 0.6604
## alternative hypothesis: true location shift is not equal to 0
```

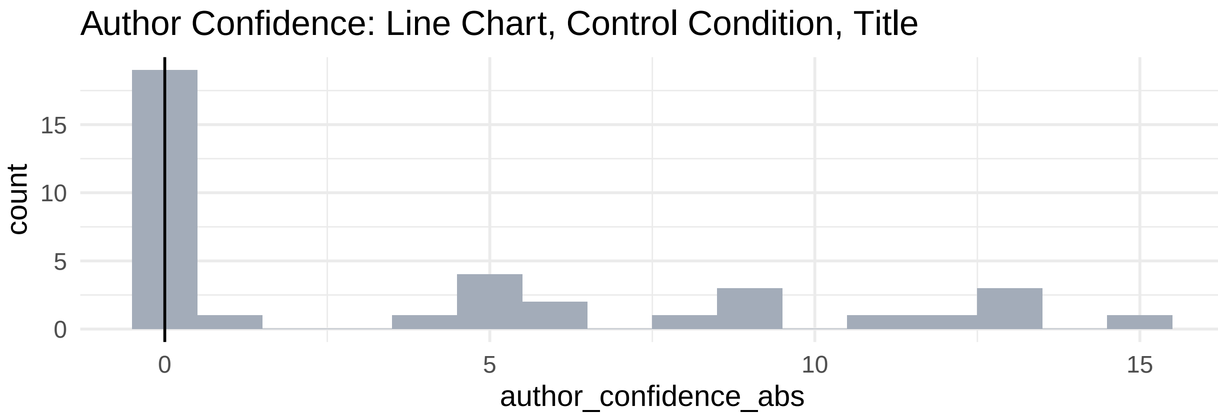
```
bar_control_title <-
  ggplot(subset(bar_control, Position == "Title"), aes(x = author_confidence_abs, y = after_stat(count)
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Bar Chart, Control Condition, Title"
  )

bar_control_annot <-
  ggplot(subset(bar_control, Position == "Annotation"), aes(x = author_confidence_abs, y = after_stat(c
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Bar Chart, Control Condition, Annotation"
  )

ggarrange(bar_control_title + geom_vline(xintercept = median(subset(bar_control, Position == "Title")$a
          bar_control_annot + geom_vline(xintercept = median(subset(bar_control, Position == "Annotatio
          ncol = 1)
```

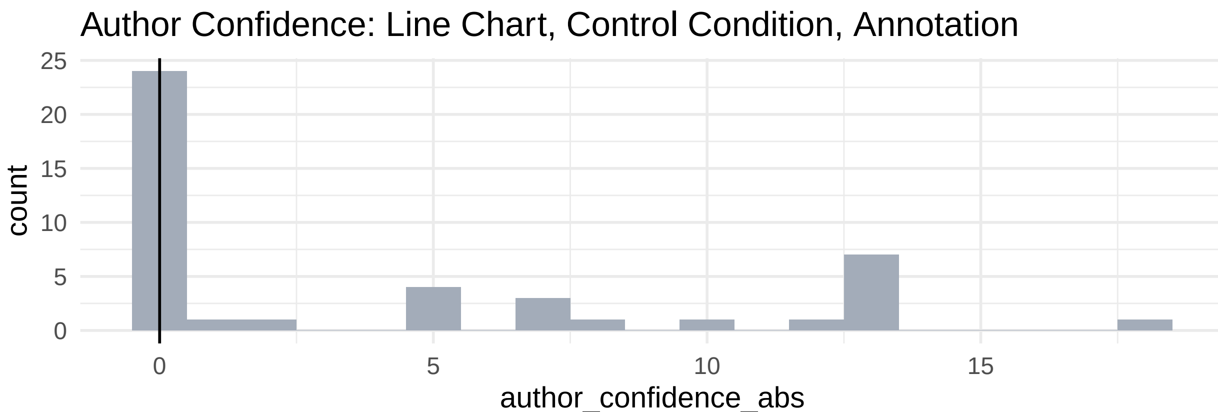


```r
wilcox.test(author_confidence_abs ~ as.factor(Position), data = bar_control)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  author_confidence_abs by as.factor(Position)
## W = 728.5, p-value = 0.4873
## alternative hypothesis: true location shift is not equal to 0
```

```r
line_control_title <-
  ggplot(subset(line_control, Position == "Title"), aes(x = author_confidence_abs, y = after_stat(count)
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Line Chart, Control Condition, Title"
  )

line_control_annot <-
  ggplot(subset(line_control, Position == "Annotation"), aes(x = author_confidence_abs, y = after_stat(c
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Line Chart, Control Condition, Annotation"
  )

ggarrange(line_control_title + geom_vline(xintercept = median(subset(line_control, Position == "Title")$
          line_control_annot + geom_vline(xintercept = median(subset(line_control, Position == "Annotati
          ncol = 1)
```






```r
wilcox.test(author_confidence_abs ~ as.factor(Position), data = line_control)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  author_confidence_abs by as.factor(Position)
## W = 810, p-value = 0.9712
## alternative hypothesis: true location shift is not equal to 0
```

We do *not* see any difference between confidence ratings for control conditions with additional neutral text placed on the chart. For this reason, in our following analyses, we can determine that the slanted content of the text we added is a contributor to any effect found. Additionally, we combine control conditions for future analyses.

Positions

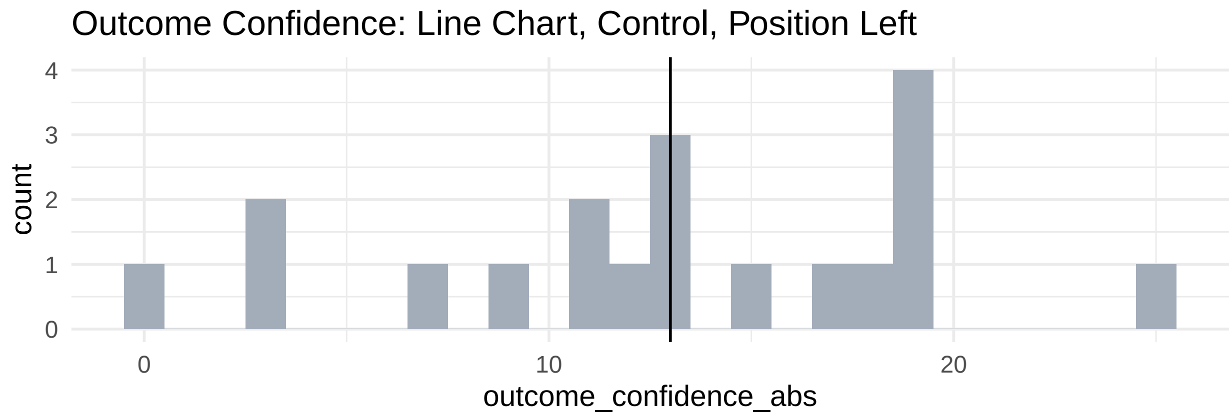
We also wanted to examine the role of the control position, since we used two positions for the annotations. In this analysis, we compare the two positions of control annotations to see if either led to significantly greater confidence in ratings than the other. We make this comparison for both the outcome ratings and the author leaning rating for bar charts and line charts.

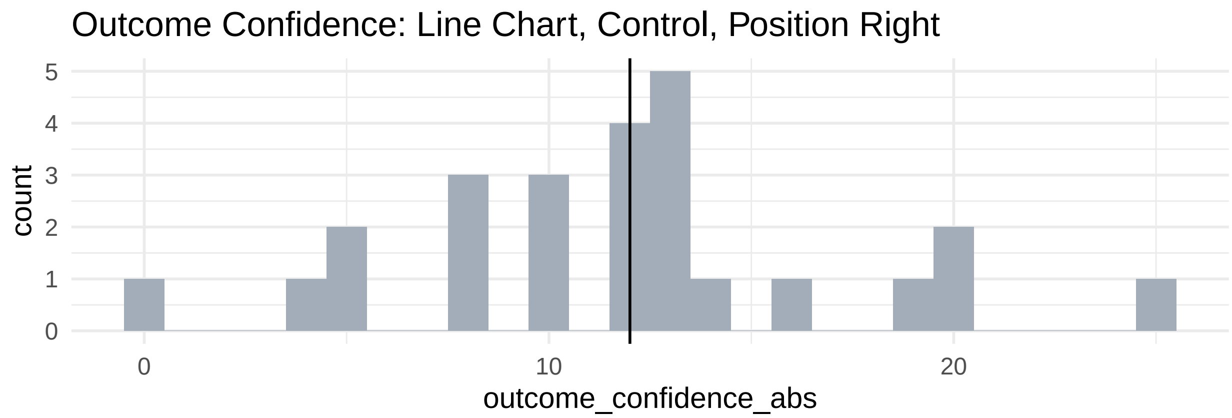
```
bar_control_positionCheck <- subset(bar_control, Position == "Annotation")


bar_control_outcome_positionLeft <-
  ggplot(subset(bar_control_positionCheck, Slant == "Blue"), aes(x = outcome_confidence_abs, y = after_
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Bar Chart, Control, Position Left"
  )

bar_control_outcome_positionRight <-
  ggplot(subset(bar_control_positionCheck, Slant == "Green"), aes(x = outcome_confidence_abs, y = after
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Bar Chart, Control, Position Right"
  )

ggarrange(bar_control_outcome_positionLeft + geom_vline(xintercept = median(subset(bar_control_position
                                                                                Slant == "Blue")$out
          bar_control_outcome_positionRight + geom_vline(xintercept = median(subset(bar_control_positio
                                                                                 Slant == "Green")$o
          ncol = 1)
```


```
wilcox.test(outcome_confidence_abs ~ as.factor(Slant), data = bar_control_positionCheck)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  outcome_confidence_abs by as.factor(Slant)
## W = 121.5, p-value = 0.2235
## alternative hypothesis: true location shift is not equal to 0
```

```
line_control_positionCheck <- subset(line_control, Position == "Annotation")

line_control_outcome_positionLeft <-
  ggplot(subset(line_control_positionCheck, Slant == "Blue"), aes(x = outcome_confidence_abs, y = after_
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Line Chart, Control, Position Left"
  )

line_control_outcome_positionRight <-
  ggplot(subset(line_control_positionCheck, Slant == "Green"), aes(x = outcome_confidence_abs, y = after
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Outcome Confidence: Line Chart, Control, Position Right"
  )

ggarrange(line_control_outcome_positionLeft + geom_vline(xintercept = median(subset(line_control_positio
                                                                                    Slant == "Blue")$ou
          line_control_outcome_positionRight + geom_vline(xintercept = median(subset(line_control_positi
                                                                                     Slant == "Green")$
          ncol = 1)
```





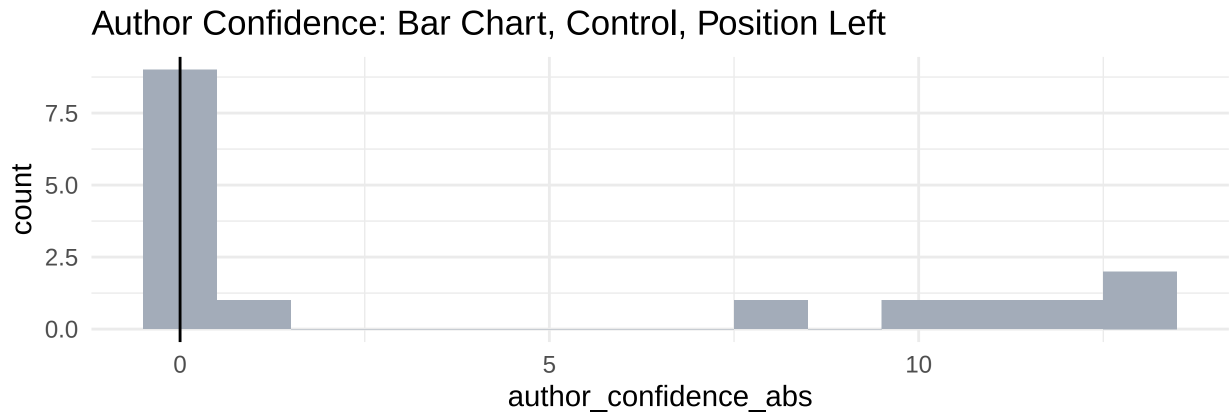

```
wilcox.test(outcome_confidence_abs ~ as.factor(Slant), data = line_control_positionCheck)
```

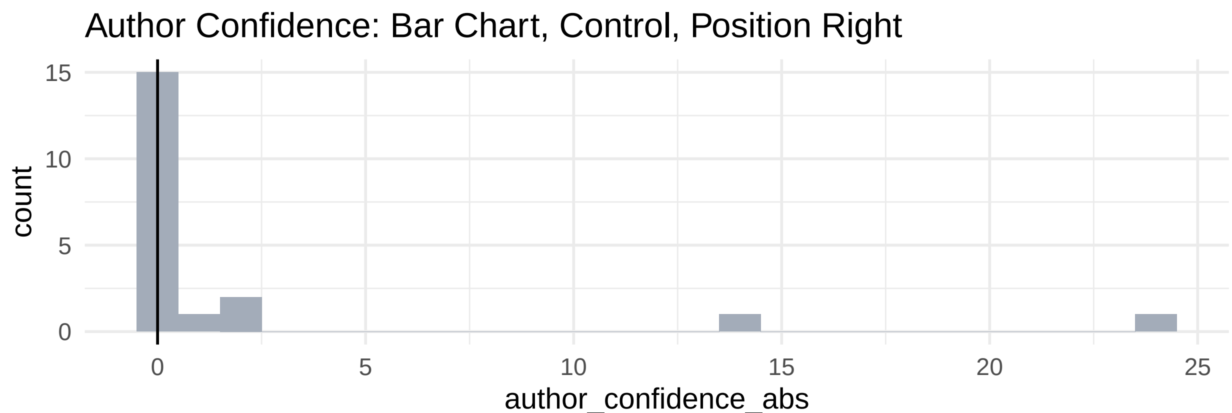
```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  outcome_confidence_abs by as.factor(Slant)
## W = 267.5, p-value = 0.4824
## alternative hypothesis: true location shift is not equal to 0
```

```
bar_control_author_positionLeft <-
  ggplot(subset(bar_control_positionCheck, Slant == "Blue"), aes(x = author_confidence_abs, y = after_s
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Bar Chart, Control, Position Left"
  )

bar_control_author_positionRight <-
  ggplot(subset(bar_control_positionCheck, Slant == "Green"), aes(x = author_confidence_abs, y = after_
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Bar Chart, Control, Position Right"
  )
```

```r
ggarrange(bar_control_author_positionLeft + geom_vline(xintercept = median(subset(bar_control_positionCh
                                                                                   Slant == "Blue")$autho
          bar_control_author_positionRight + geom_vline(xintercept = median(subset(bar_control_position(
                                                                                    Slant == "Green")$au
          ncol = 1)
```





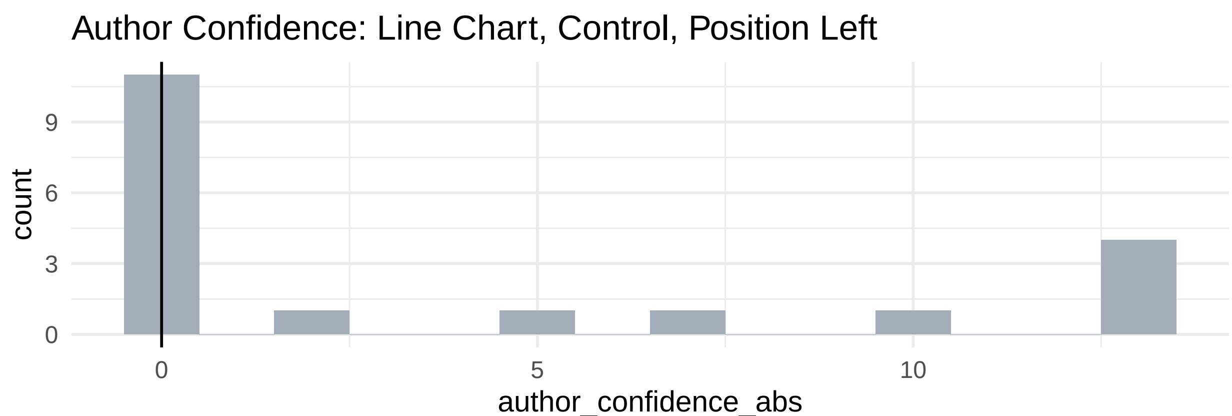

```r
wilcox.test(author_confidence_abs ~ as.factor(Slant), data = bar_control_positionCheck)
```

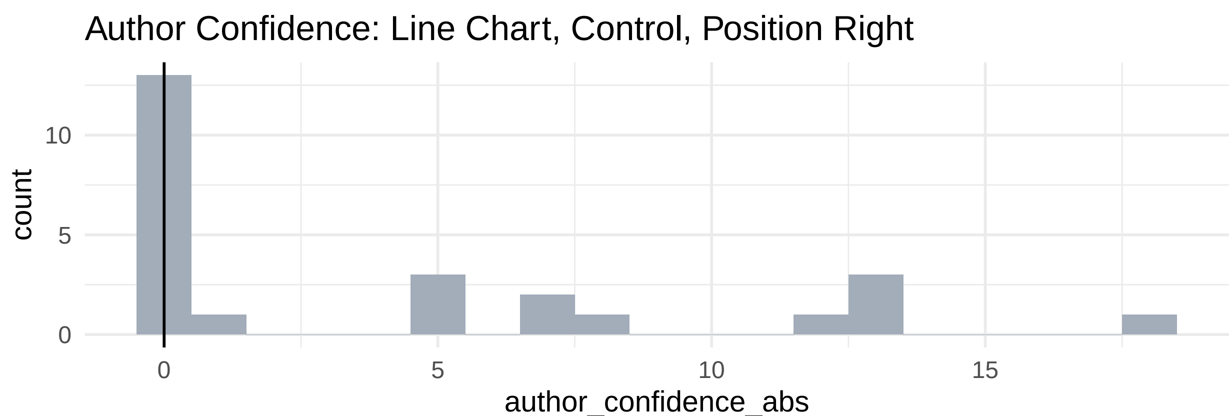
```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  author_confidence_abs by as.factor(Slant)
## W = 191, p-value = 0.247
## alternative hypothesis: true location shift is not equal to 0
```

```r
line_control_author_positionLeft <-
  ggplot(subset(line_control_positionCheck, Slant == "Blue"), aes(x = author_confidence_abs, y = after_s
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Line Chart, Control, Position Left"
  )
```

```r
line_control_author_positionRight <-
  ggplot(subset(line_control_positionCheck, Slant == "Green"), aes(x = author_confidence_abs, y = after_
  geom_histogram(binwidth = 1, fill = c)+
  labs(
    title = "Author Confidence: Line Chart, Control, Position Right"
  )

ggarrange(line_control_author_positionLeft + geom_vline(xintercept = median(subset(line_control_positio
                                                                                  Slant == "Blue")$auth
          line_control_author_positionRight + geom_vline(xintercept = median(subset(line_control_positi
                                                                                   Slant == "Green")$au
          ncol = 1)
```






```r
wilcox.test(author_confidence_abs ~ as.factor(Slant), data = line_control_positionCheck)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties

##
##  Wilcoxon rank sum test with continuity correction
##
## data:  author_confidence_abs by as.factor(Slant)
## W = 229, p-value = 0.8355
## alternative hypothesis: true location shift is not equal to 0
```

We do *not* see any difference between confidence ratings for control conditions with different annotation positions.

Chart Types

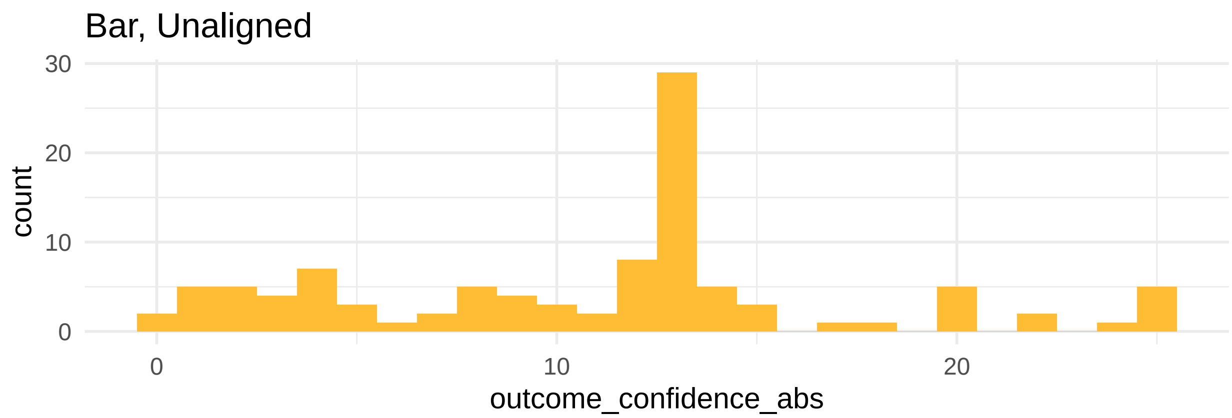
We also conduct an analysis to compare overall confidence ratings between the chart types we displayed. We do not expect to see a difference between the chart types. If we see an effect in the control conditions, that indicates that something about our tasks functions differently between chart types (e.g., people are more confident overall in their estimations for one chart type than another). If we see an effect in the treatment conditions, that indicates that something about text on different chart types functions differently (e.g., text is more impactful when placed on one chart type than another).

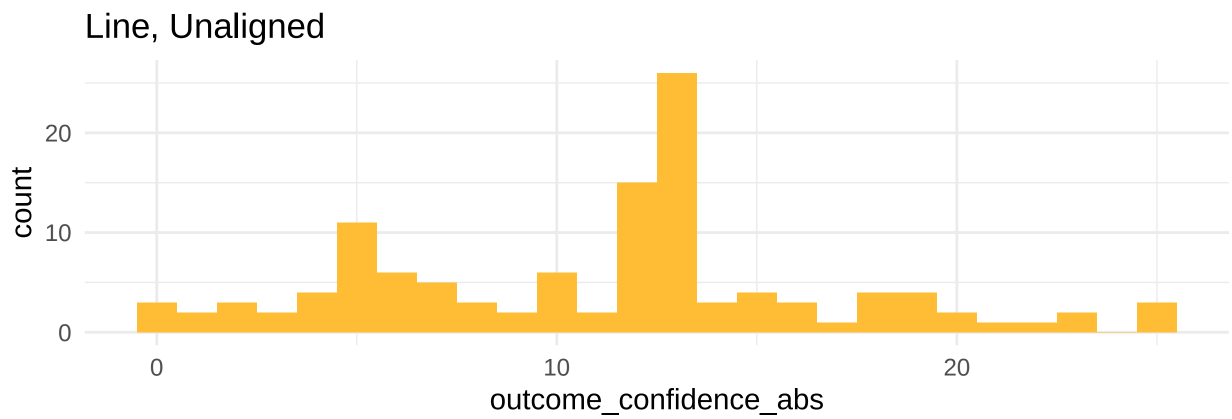
```
ggarrange(bar_aligned_treatment_outcome_plot,
          line_aligned_treatment_outcome_plot,
          nrow = 2)
```


```
wilcox.test(outcome_confidence_abs ~ as.factor(ChartType), data = subset(df, outcome_aligned == "Aligned
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  outcome_confidence_abs by as.factor(ChartType)
## W = 7995, p-value = 0.6224
## alternative hypothesis: true location shift is not equal to 0
```

```r
ggarrange(bar_unaligned_treatment_outcome_plot,
          line_unaligned_treatment_outcome_plot,
          nrow = 2)
```

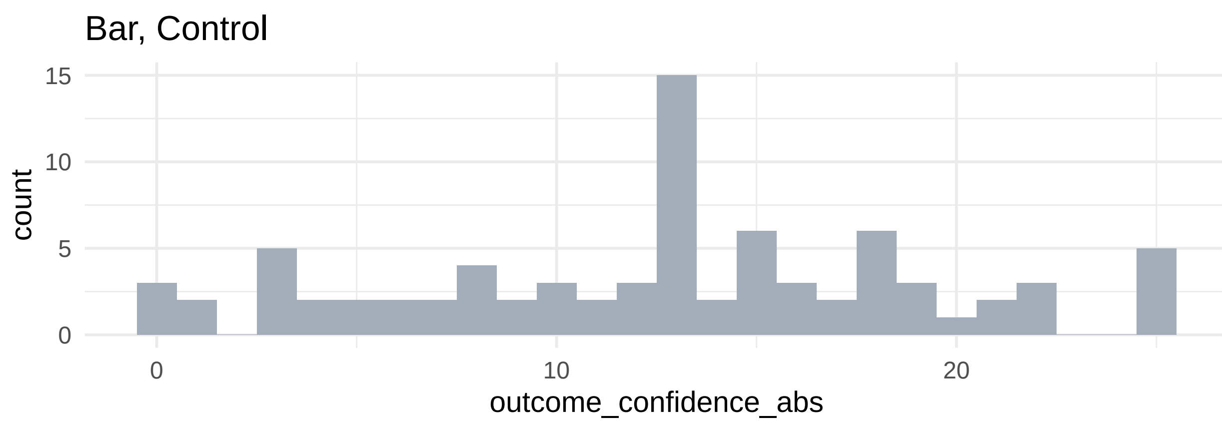


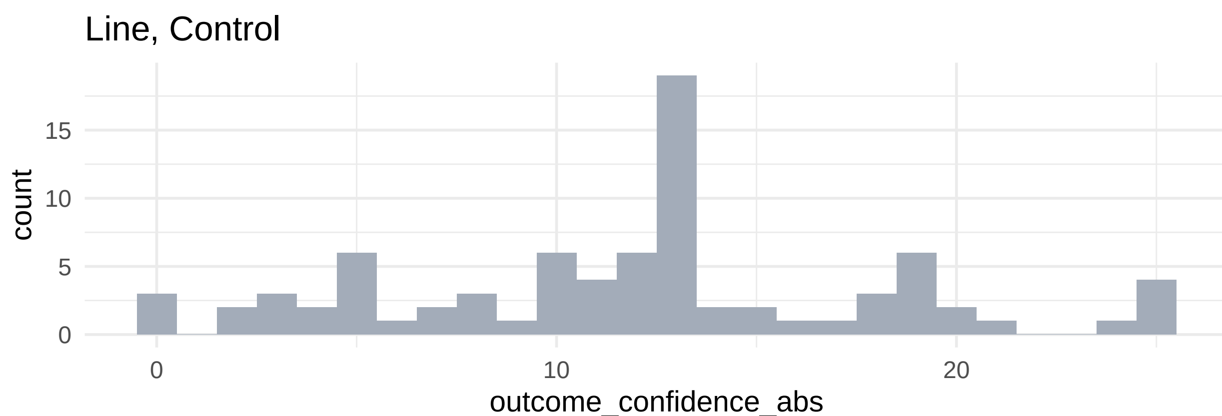



```r
wilcox.test(outcome_confidence_abs ~ as.factor(ChartType), data = subset(df, outcome_aligned == "Unalig
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  outcome_confidence_abs by as.factor(ChartType)
## W = 6008, p-value = 0.8841
## alternative hypothesis: true location shift is not equal to 0
```

```r
ggarrange(bar_control_outcome_plot,
          line_control_outcome_plot,
          nrow = 2)
```

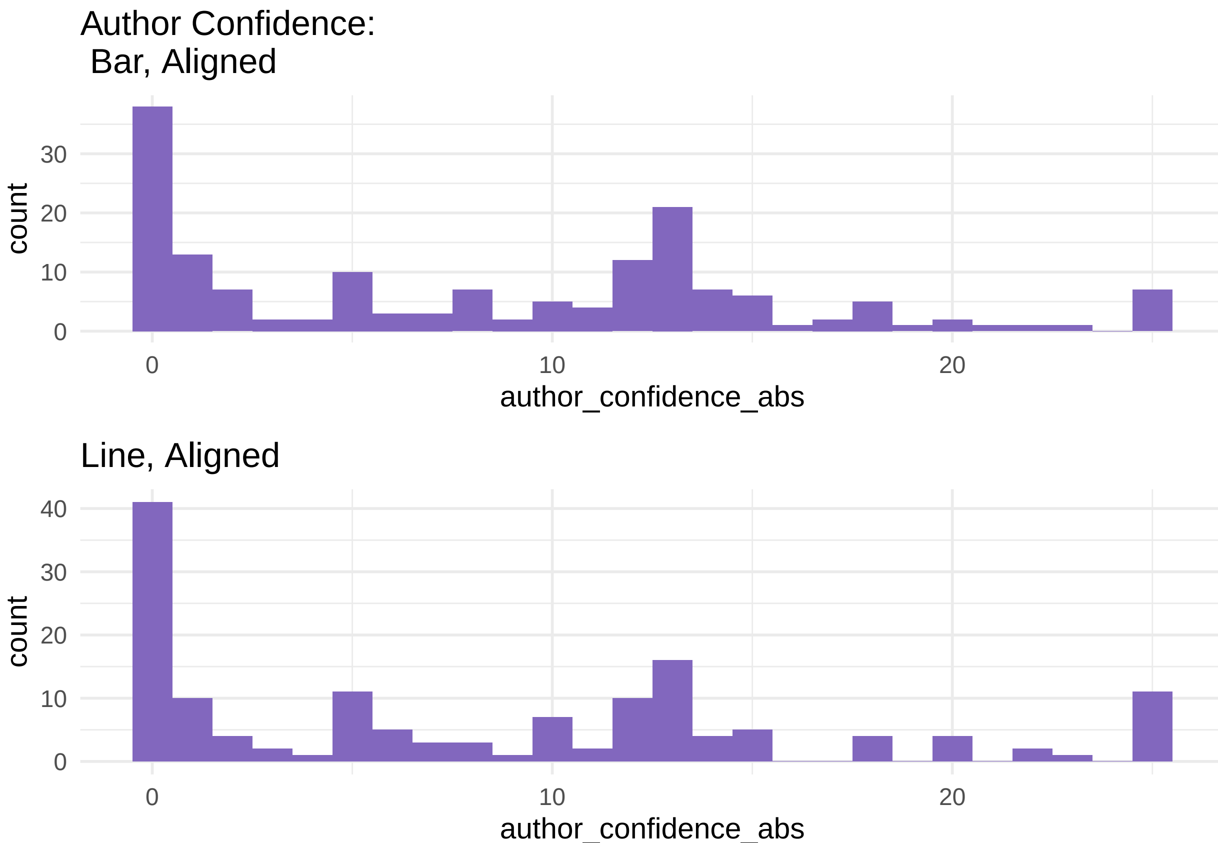





```
wilcox.test(outcome_confidence_abs ~ as.factor(ChartType), data = subset(df, outcome_aligned == "NA"))
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  outcome_confidence_abs by as.factor(ChartType)
## W = 3508, p-value = 0.3631
## alternative hypothesis: true location shift is not equal to 0
```

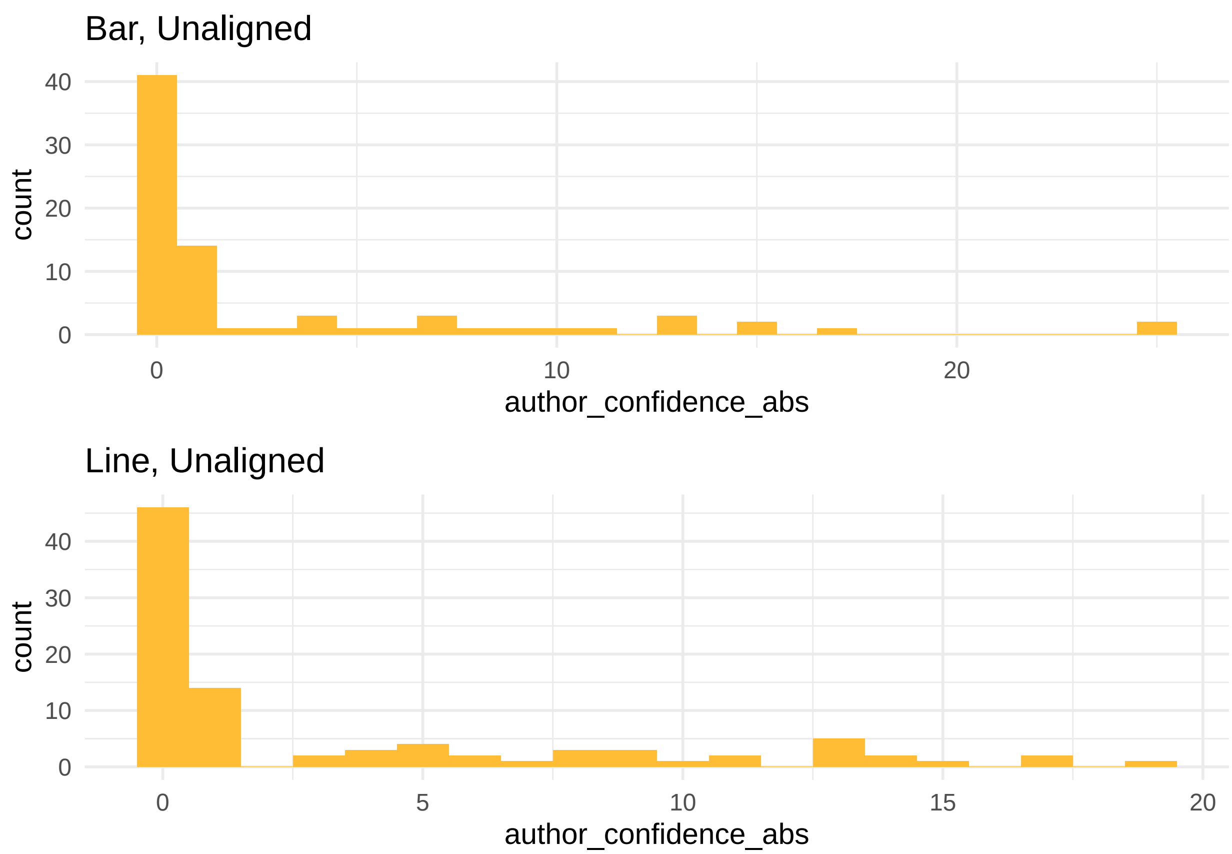
```
ggarrange(bar_aligned_treatment_author_plot,
          line_aligned_treatment_author_plot,
          nrow = 2)
```




```
wilcox.test(author_confidence_abs ~ as.factor(ChartType), data = subset(df, author_aligned == "Aligned")
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  author_confidence_abs by as.factor(ChartType)
## W = 12303, p-value = 0.6798
## alternative hypothesis: true location shift is not equal to 0
```

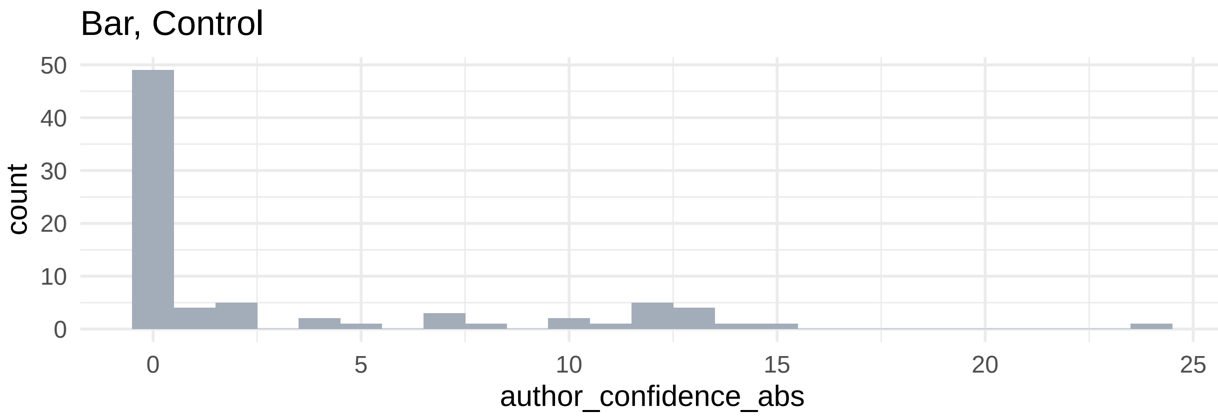
```
ggarrange(bar_unaligned_treatment_author_plot,
          line_unaligned_treatment_author_plot,
          nrow = 2)
```

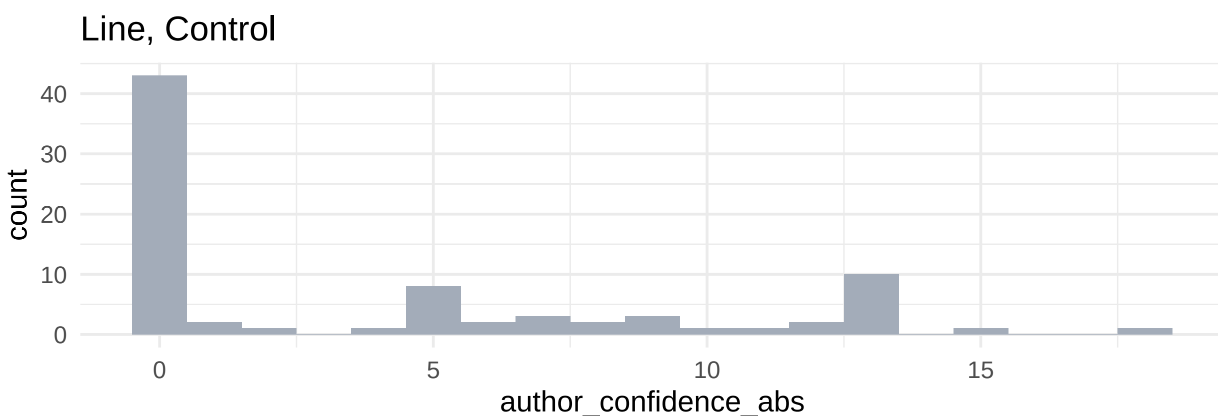



```
wilcox.test(author_confidence_abs ~ as.factor(ChartType), data = subset(df, author_aligned == "Unaligned
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  author_confidence_abs by as.factor(ChartType)
## W = 3358, p-value = 0.5319
## alternative hypothesis: true location shift is not equal to 0
```

```
ggarrange(bar_control_author_plot,
          line_control_author_plot,
          nrow = 2)
```






```r
wilcox.test(author_confidence_abs ~ as.factor(ChartType), data = subset(df, author_aligned == "NA"))
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  author_confidence_abs by as.factor(ChartType)
## W = 2910.5, p-value = 0.2171
## alternative hypothesis: true location shift is not equal to 0
```

The chart types do not differ significantly between their overall confidence ratings. We don't combine these responses, however, since we want to keep chart type analyses separate.

Test for Categorical Differences

If there were an effect of viewing the text, we would expect to see the following:

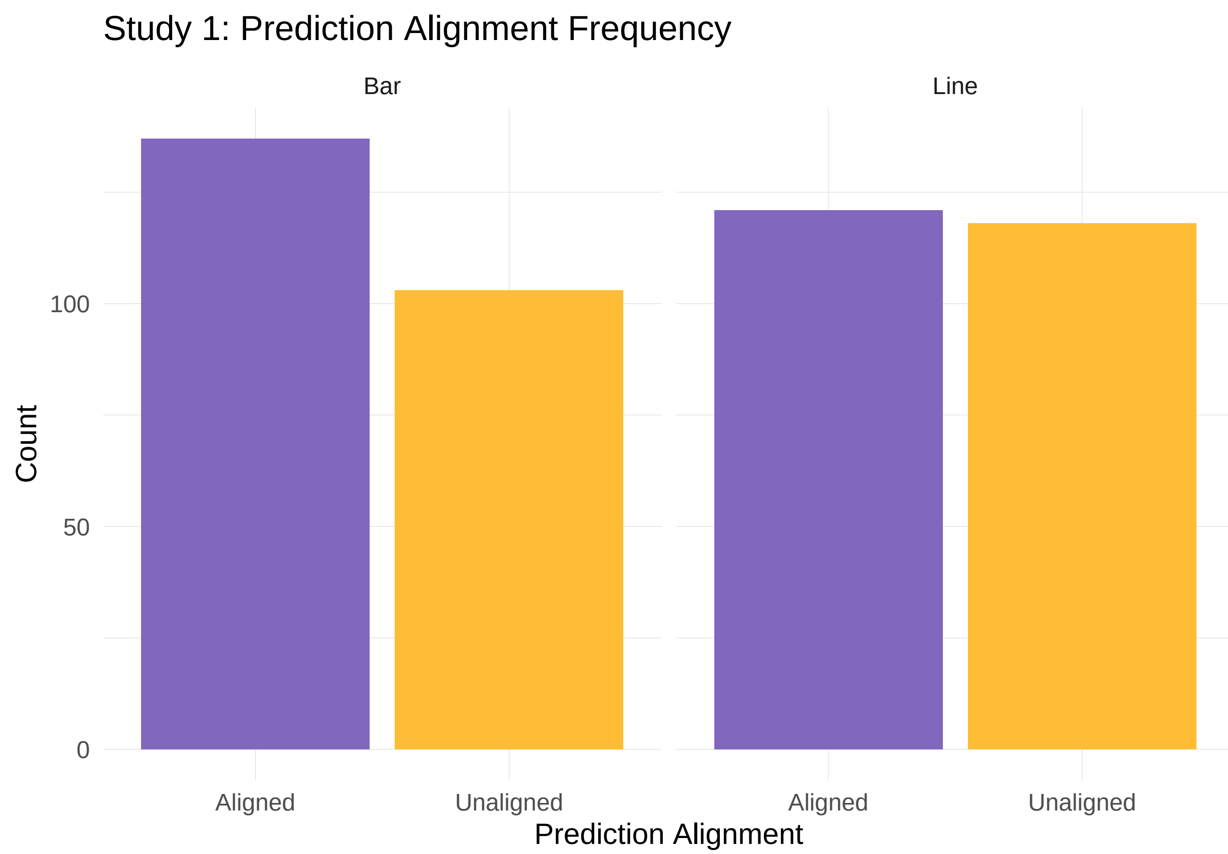
1. It is more likely to report predictions in alignment with the group supported in the text.
2. Predictions aligned with the group supported in the text are more confident than predictions unaligned with the group supported and control responses. Control responses are more confident than predictions unaligned.

These hypotheses refer to overall differences between treatment and control conditions and do not account for possible differences between treatment conditions of position and content. We expect these hypotheses to hold for both outcome prediction and bias appraisals. We first examine outcome prediction for both bar and line charts.

In these charts, we use a subset of the overall data to examine only the treatment conditions. We're examining the key question here: Were people more likely to respond categorically aligned with the chart or was there no difference between the frequency of unaligned and aligned responses? To capture this effect statistically, we use Chi-squared testing using Yates' continuity correction. Respective tests are completed below the charts which visualize the difference being examined.

```
ggplot(subset(df, treatment == "Treatment"), aes(x = outcome_aligned, y = after_stat(count), fill = outc
  geom_bar()+
  scale_fill_manual(values = c(a, u, c))+
  labs(
    title = "Study 1: Prediction Alignment Frequency",
    y = "Count",
    x = "Prediction Alignment"
  )+
  facet_grid (.~ChartType)+
  theme(legend.position = 'none',
  panel.grid.major = element_line(color = "grey85", size = 0.1),  # Lighten and thin major gridlines
  panel.grid.minor = element_line(color = "grey85", size = 0.05))  # Lighten and thin minor gridlines
```

```
## Warning: The `size` argument of `element_line()` is deprecated as of ggplot2 3.4.0.
## i Please use the `linewidth` argument instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```




```
df %>%
  group_by(ChartType, outcome_aligned) %>%
  summarise(
    n = n()
  )
```

```
## `summarise()` has grouped output by 'ChartType'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 3
## # Groups:   ChartType [2]
##   ChartType outcome_aligned     n
##   <chr>     <chr>           <int>
## 1 Bar       Aligned           137
## 2 Bar       NA                 80
## 3 Bar       Unaligned         103
## 4 Line      Aligned           121
## 5 Line      NA                 81
## 6 Line      Unaligned         118
```

```
t.outcome.bar <- table(subset(bar, treatment == "Treatment")$outcome_aligned)
chisq.test(t.outcome.bar)
```

```
##
```

```
##  Chi-squared test for given probabilities
##
## data:  t.outcome.bar
## X-squared = 4.8167, df = 1, p-value = 0.02819
```

```
p.outcome.bar <- t.outcome.bar / sum(t.outcome.bar)
dimnames(p.outcome.bar) <- NULL
ES.h(p.outcome.bar[1], p.outcome.bar[2])
```

```
## [1] 0.2842897
```

```
t.outcome.line <- table(subset(line, treatment == "Treatment")$outcome_aligned)
chisq.test(t.outcome.line)
```

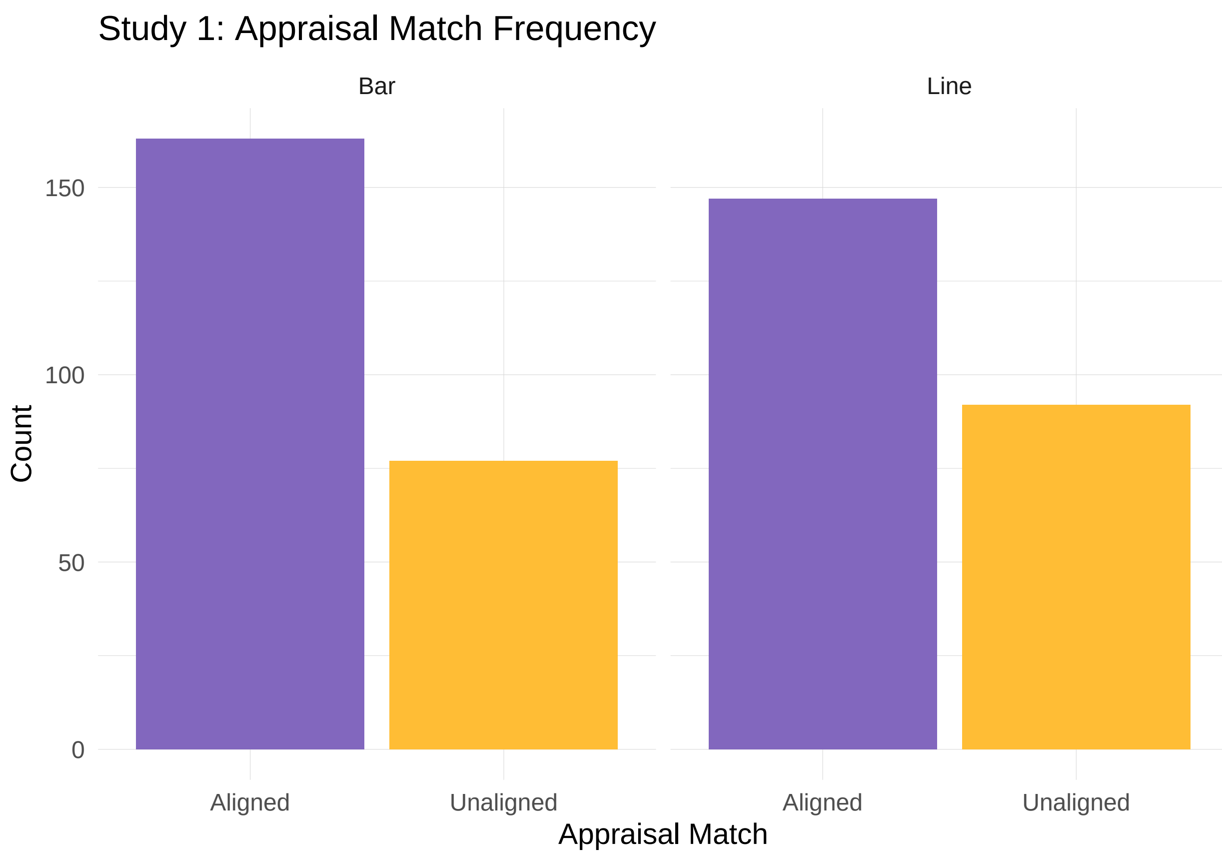
```
##
##  Chi-squared test for given probabilities
##
## data:  t.outcome.line
## X-squared = 0.037657, df = 1, p-value = 0.8461
```

```
p.outcome.line <- t.outcome.line / sum(t.outcome.line)
dimnames(p.outcome.line) <- NULL
ES.h(p.outcome.line[1], p.outcome.line[2])
```

```
## [1] 0.02510526
```

We now go on to conduct these same tests for the author leaning response. We're examining the key question here: Were people more likely to respond *that the author was* categorically aligned with the chart? We continue to use Chi-squared testing using Yates' continuity correction and provide tests below the charts which visualize the difference being examined.

```
ggplot(subset(df, treatment == "Treatment"), aes(x = author_aligned, y = after_stat(count), fill = autho
  geom_bar()+
  scale_fill_manual(values = c(a, u, c), labels= c("Matched", "Unmatched"))+
  labs(
    title = "Study 1: Appraisal Match Frequency",
    y = "Count",
    x = "Appraisal Match"
  )+
  facet_grid (.~ChartType)+
  theme(legend.position = 'none',
  panel.grid.major = element_line(color = "grey85", size = 0.1),  # Lighten and thin major gridlines
  panel.grid.minor = element_line(color = "grey85", size = 0.05))  # Lighten and thin minor gridlines
```




```
df %>%
  group_by(ChartType, author_aligned) %>%
  summarise(
    n = n()
  )
```

```
## `summarise()` has grouped output by 'ChartType'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 3
## # Groups:   ChartType [2]
##   ChartType author_aligned     n
##   <chr>     <chr>          <int>
## 1 Bar       Aligned          163
## 2 Bar       NA                80
## 3 Bar       Unaligned         77
## 4 Line      Aligned          147
## 5 Line      NA                81
## 6 Line      Unaligned         92
```

```
t.author.bar <- table(subset(bar, treatment == "Treatment")$author_aligned)
chisq.test(t.author.bar)
```

```
##
```

```
##   Chi-squared test for given probabilities
##
## data:  t.author.bar
## X-squared = 30.817, df = 1, p-value = 2.836e-08
```

```
p.author.bar <- t.author.bar / sum(t.author.bar)
dimnames(p.author.bar) <- NULL
ES.h(p.author.bar[1], p.author.bar[2])
```

```
## [1] 0.7329641
```

```
t.author.line <- table(subset(line, treatment == "Treatment")$author_aligned)
chisq.test(t.author.line)
```

```
##
##   Chi-squared test for given probabilities
##
## data:  t.author.line
## X-squared = 12.657, df = 1, p-value = 0.0003742
```

```
p.author.line <- t.author.line / sum(t.author.line)
dimnames(p.author.line) <- NULL
ES.h(p.author.line[1], p.author.line[2])
```

```
## [1] 0.4644133
```

Outcome of testing:

Bar charts exhibited a significant effect of presenting slanted text on categorical outcome predictions ($p = 0.028$). Line charts did not ($p = 0.846$).

Bar charts and line charts both exhibited a significant effect of presenting slanted text on categorical author leaning ratings ($p = 0$, $p = 0$).

Hypothesis Evaluation:

It is more likely to report outcome and author leaning in alignment with the slant presented in the chart.

In most contexts, this hypothesis was supported. However, line charts did not exhibit a significant effect for outcome responses. This indicates that the effect of text on a chart may differ depending on the chart type. The author leaning response effect was also less for the line chart than the bar chart, although both were significant in that regard.

Test for Quantitative Differences

Treatment Conditions

We now go on to evaluate H2, examining the differences in confidence ratings between aligned responses in treatment conditions, unaligned responses in treatment conditions, and the control responses. To do this, we will use the nonparametric Kruskal-Wallis test with Dunn post-hoc pairwise testing with Bonferroni correction. We're examining the key question here: Were people more or less confident in their responses relative to control depending on whether their response aligned with the slant in the chart? Respective tests are completed below the charts which visualize the difference being examined.

We use the same charts as shown above, but placed directly above each other in order to facilitate comparison between shifts in distributions. The y-axes are adjusted to provide a full view of each distribution, as the control condition had overall fewer responses than the treatment conditions combined.

```
df %>%
  group_by(ChartType, outcome_aligned) %>%
  summarise(
    median_outcome_conf = median(outcome_confidence_abs),
    mean_outcome_conf = mean(outcome_confidence_abs)
  )
```

```
## `summarise()` has grouped output by 'ChartType'. You can override using the
## `.groups` argument.
```

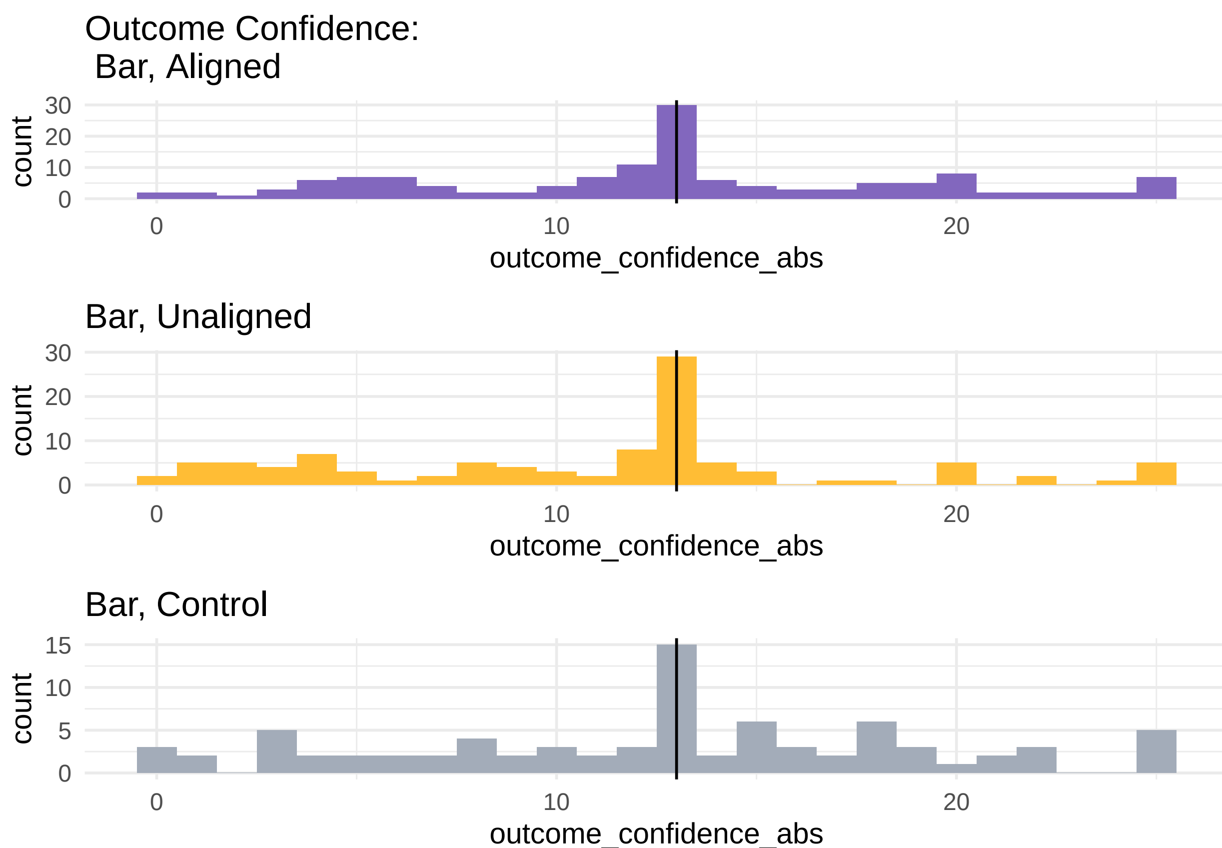
```
## # A tibble: 6 x 4
## # Groups:   ChartType [2]
##   ChartType outcome_aligned median_outcome_conf mean_outcome_conf
##   <chr>     <chr>                         <dbl>             <dbl>
## 1 Bar       Aligned                          13              12.8
## 2 Bar       NA                               13              12.6
## 3 Bar       Unaligned                        13              11.0
## 4 Line      Aligned                          13              13.1
## 5 Line      NA                               13              12.0
## 6 Line      Unaligned                        12              11.2
```

```
df %>%
  group_by(ChartType, author_aligned) %>%
  summarise(
    median_author_conf = median(author_confidence_abs),
    mean_author_conf = mean(author_confidence_abs)
  )
```

```
## `summarise()` has grouped output by 'ChartType'. You can override using the
## `.groups` argument.
```

```
## # A tibble: 6 x 4
## # Groups:   ChartType [2]
##   ChartType author_aligned median_author_conf mean_author_conf
##   <chr>     <chr>                        <dbl>            <dbl>
## 1 Bar       Aligned                        8               8.24
## 2 Bar       NA                             0               3.15
## 3 Bar       Unaligned                      0               3.08
## 4 Line      Aligned                        6               8.24
## 5 Line      NA                             0               4.10
## 6 Line      Unaligned                      0.5             3.42
```

```
ggarrange(bar_aligned_treatment_outcome_plot + geom_vline(xintercept = median(bar_aligned_treatment_outc
          bar_unaligned_treatment_outcome_plot + geom_vline(xintercept = median(bar_unaligned_treatment_
          bar_control_outcome_plot + geom_vline(xintercept = median(bar_control$outcome_confidence_abs)
          ncol = 1)
```




```r
kruskal.test(outcome_confidence_abs ~ as.factor(outcome_aligned), data = bar)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  outcome_confidence_abs by as.factor(outcome_aligned)
## Kruskal-Wallis chi-squared = 5.697, df = 2, p-value = 0.05793
```

```r
kruskal_effsize(outcome_confidence_abs ~ as.factor(outcome_aligned), data = bar)
```

```
## # A tibble: 1 x 5
##   .y.                        n effsize method  magnitude
## * <chr>                  <int>   <dbl> <chr>   <ord>
## 1 outcome_confidence_abs   320  0.0117 eta2[H] small
```

```r
dunnTest(outcome_confidence_abs ~ as.factor(outcome_aligned), data = bar, method = "bonferroni")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison

##   p-values adjusted with the Bonferroni method.

##            Comparison          Z    P.unadj      P.adj
## 1        Aligned - NA -0.0565408 0.95491099 1.00000000
## 2 Aligned - Unaligned  2.1667801 0.03025163 0.09075488
## 3      NA - Unaligned  1.9495659 0.05122788 0.15368363
```

```r
ggarrange(line_aligned_treatment_outcome_plot + geom_vline(xintercept = median(line_aligned_treatment_ou
          line_unaligned_treatment_outcome_plot + geom_vline(xintercept = median(line_unaligned_treatmen
          line_control_outcome_plot + geom_vline(xintercept = median(line_control$outcome_confidence_abs
          ncol = 1)
```


```r
kruskal.test(outcome_confidence_abs ~ as.factor(outcome_aligned), data = line)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  outcome_confidence_abs by as.factor(outcome_aligned)
## Kruskal-Wallis chi-squared = 6.7021, df = 2, p-value = 0.03505
```

```r
kruskal_effsize(outcome_confidence_abs ~ as.factor(outcome_aligned), data = line)
```

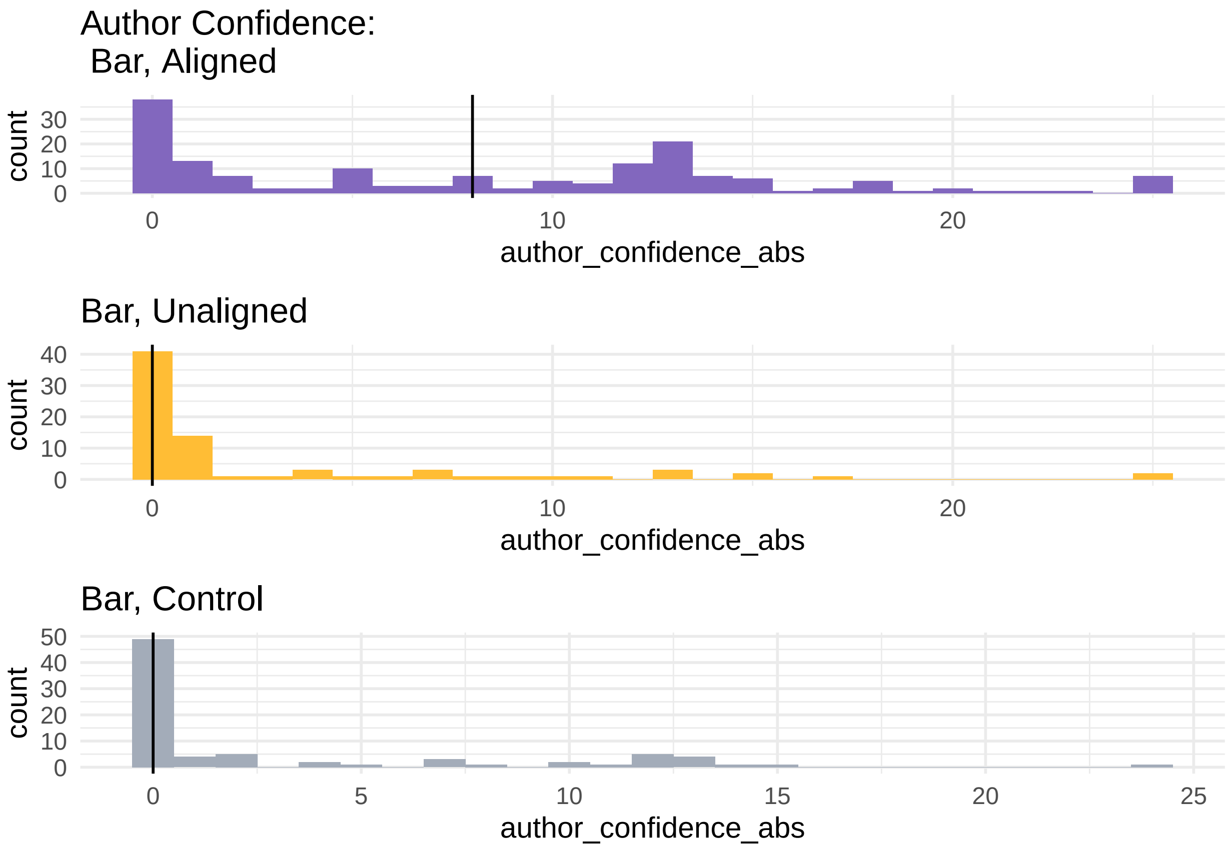
```
## # A tibble: 1 x 5
##   .y.                        n effsize method  magnitude
## * <chr>                  <int>   <dbl> <chr>   <ord>
## 1 outcome_confidence_abs   320  0.0148 eta2[H] small
```

```r
dunnTest(outcome_confidence_abs ~ as.factor(outcome_aligned), data = line, method = "bonferroni")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Bonferroni method.
```

```
##            Comparison         Z    P.unadj      P.adj
## 1        Aligned - NA 1.4185681 0.15602496 0.46807488
## 2 Aligned - Unaligned 2.5705897 0.01015255 0.03045766
## 3      NA - Unaligned 0.8935234 0.37157694 1.00000000
```

```
ggarrange(bar_aligned_treatment_author_plot + geom_vline(xintercept = median(bar_aligned_treatment_autho
          bar_unaligned_treatment_author_plot + geom_vline(xintercept = median(bar_unaligned_treatment_a
          bar_control_author_plot + geom_vline(xintercept = median(bar_control$author_confidence_abs)),
          ncol = 1)
```




```
kruskal.test(author_confidence_abs ~ as.factor(author_aligned), data = bar)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  author_confidence_abs by as.factor(author_aligned)
## Kruskal-Wallis chi-squared = 51.142, df = 2, p-value = 7.848e-12
```

```
kruskal_effsize(author_confidence_abs ~ as.factor(author_aligned), data = bar)
```

```
## # A tibble: 1 x 5
##   .y.                       n effsize method  magnitude
## * <chr>                 <int>   <dbl> <chr>   <ord>
## 1 author_confidence_abs   320   0.155 eta2[H] large
```

```r
dunnTest(author_confidence_abs ~ as.factor(author_aligned), data = bar, method = "bonferroni")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison

##   p-values adjusted with the Bonferroni method.

##            Comparison          Z       P.unadj        P.adj
## 1         Aligned - NA  6.0217403 1.725516e-09 5.176547e-09
## 2 Aligned - Unaligned  5.6051459 2.080796e-08 6.242389e-08
## 3      NA - Unaligned -0.2939952 7.687615e-01 1.000000e+00
```

```r
ggarrange(line_aligned_treatment_author_plot + geom_vline(xintercept = median(line_aligned_treatment_au
          line_unaligned_treatment_author_plot + geom_vline(xintercept = median(line_unaligned_treatmen
          line_control_author_plot + geom_vline(xintercept = median(line_control$author_confidence_abs)
          ncol = 1)
```


```r
kruskal.test(author_confidence_abs ~ as.factor(author_aligned), data = line)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  author_confidence_abs by as.factor(author_aligned)
## Kruskal-Wallis chi-squared = 29.343, df = 2, p-value = 4.249e-07
```

```r
kruskal_effsize(author_confidence_abs ~ as.factor(author_aligned), data = line)
```

```
## # A tibble: 1 x 5
##   .y.                       n effsize method  magnitude
## * <chr>                 <int>   <dbl> <chr>   <ord>
## 1 author_confidence_abs   320  0.0863 eta2[H] moderate
```

```r
dunnTest(author_confidence_abs ~ as.factor(author_aligned), data = line, method = "bonferroni")
```

```
## Dunn (1964) Kruskal-Wallis multiple comparison
##   p-values adjusted with the Bonferroni method.

##            Comparison         Z      P.unadj        P.adj
## 1        Aligned - NA 4.1738342 2.995158e-05 8.985474e-05
## 2 Aligned - Unaligned 4.7515529 2.018603e-06 6.055810e-06
## 3      NA - Unaligned 0.3550157 7.225778e-01 1.000000e+00
```

Outcome of testing:

Bar charts did not exhibit a significant effect of presenting slanted text on quantitative outcome predictions. Specifically ($p = 0.058$). Line charts did exhibit a significant effect in this context ($p = 0.035$). Follow up testing indicates that responses which were aligned with the outcome presented in the text were more confident than those which were unaligned ($p = 0.03$). There were no significant differences between aligned responses and control ($p = 0.468$) nor unaligned responses and control ($p = 1$).

Bar charts exhibited a significant effect of presenting slanted text on quantitative author leaning ratings (p = 0. Follow up testing indicates that responses which aligned the author with the slant presented in the text were more confident than those in the control condition ($p = 0$). Additionally, these aligned responses were more confident than responses which categorized the author in opposition to the slant presented ($p = 0$). There was no difference between unaligned responses and control ($p =1$)

Line charts exhibited a similar effect ($p = 0$). Follow up testing indicates that responses which aligned the author with the slant presented in the text were more confident than those in the control condition ($p = 0$). Additionally, these aligned responses were more confident than responses which categorized the author in opposition to the slant presented ($p = 0$). There was no difference between unaligned responses and control ($p = 1$)

Hypothesis Evaluation:

Responses aligned with the group supported in the text are more confident than responses unaligned with the group supported

This part of the hypothesis was not supported in the context of outcome predictions in bar charts. However, for outcome predictions in line charts, this hypothesis was supported. This difference sheds light on the previous finding - line charts did not exhibit categorical change for outcome prediction when faced with text which supported one outcome over another, while bar charts did. The influence of text for bar charts seems to be more likely to shift readers categorically, but does not change the confidence ratings for readers who disagree witht the outcome. The effect is the opposite in line charts: the text does not move readers from one side to the other, but it does affect their overall confidence in their judgement. Additionally, this hypothesis was supported in the context of author leaning ratings for both chart types.

Responses aligned with the group supported in the text are more confident than control responses

This part of the hypothesis was not supported in the context of outcome predictions. However, for both chart types, this hypothesis was supported in the context of author leaning ratings.

Control responses are more confident than responses unaligned.

This part of the hypothesis was not supported in either context for either chart type.

Evaluation of Levels & Positions

As a further examination of the effect found, we wanted to conduct an analysis of the different text *content* and *position*. To do this, we focus on the difference between aligned and unaligned responses specifically, as most contexts did not find differences between control conditions and the aligned/unaligned responses. To determine differences in effect between content and positions, we directly compare the difference between aligned and unaligned respones. More influential text will have a larger value (more differentiation between readers who agree and disagree with the text provided), and vice versa.

Specifically, we evaluate the following hypotheses:

4. Responses aligned with the group supported are more confident when viewing L4 than when viewing other semantic levels.
5. Responses unaligned with the group supported are less confident when viewing L4 than when viewing other semantic levels.
6. Text in the title will have greater influence on judgments than text positioned by the data. OR Text positioned by the data will have greater influence on judgments than text in the title.

```
chisq.test(table(subset(df, treatment == "Treatment")$outcome_aligned, subset(df, treatment == "Treatme
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(subset(df, treatment == "Treatment")$outcome_aligned, subset(df,     treatment == "Trea
## X-squared = 1.8197, df = 2, p-value = 0.4026
```

```
table(subset(df, treatment == "Treatment")$outcome_aligned, subset(df, treatment == "Treatment")$Level)
```

```
##
##              4  3  2
##   Aligned   87 93 78
##   Unaligned 72 70 79
```

```
chisq.test(table(subset(df, treatment == "Treatment")$outcome_aligned, subset(df, treatment == "Treatme
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(subset(df, treatment == "Treatment")$outcome_aligned, subset(df,     treatment == "Trea
## X-squared = 0.73982, df = 1, p-value = 0.3897
```

```
table(subset(df, treatment == "Treatment")$outcome_aligned, subset(df, treatment == "Treatment")$Positi
```

```
##
##             Annotation Title
##   Aligned          123   135
##   Unaligned        115   106
```

```
chisq.test(table(subset(df, treatment == "Treatment")$author_aligned, subset(df, treatment == "Treatmen
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(subset(df, treatment == "Treatment")$author_aligned, subset(df,     treatment == "Treatm
## X-squared = 8.0308, df = 2, p-value = 0.01804
```

```
table(subset(df, treatment == "Treatment")$author_aligned, subset(df, treatment == "Treatment")$Level)[
```

```
##
##               4   3   2
##   Aligned   116 103  91
##   Unaligned  43  60  66
```

```
chisq.test(table(subset(df, treatment == "Treatment")$author_aligned, subset(df, treatment == "Treatmen
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  table(subset(df, treatment == "Treatment")$author_aligned, subset(df,     treatment == "Treatm
## X-squared = 2.0412, df = 1, p-value = 0.1531
```

```
table(subset(df, treatment == "Treatment")$author_aligned, subset(df, treatment == "Treatment")$Positio
```

```
##
##             Annotation Title
##   Aligned          162   148
##   Unaligned         76    93
```

```
#### prediction ####

model0_outcome <- lmer(outcome_confidence_abs ~ ChartType + (1|Order) + (1|Slant), data = df)
```

```
## boundary (singular) fit: see help('isSingular')
```

```
model1_outcome <- lmer(outcome_confidence_abs ~ ChartType + outcome_aligned + (1|Order) + (1|Slant), da
```

```
## boundary (singular) fit: see help('isSingular')
```

```
model2_outcome <- lmer(outcome_confidence_abs ~ ChartType + outcome_aligned + Level + Position + (1|Ord
```

```
## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
## boundary (singular) fit: see help('isSingular')
```

```
model3_outcome <- lmer(outcome_confidence_abs ~ ChartType + outcome_aligned + Level * Position + (1|Ord
```

```
## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
## boundary (singular) fit: see help('isSingular')
```

```
model4_outcome <- lmer(outcome_confidence_abs ~ ChartType + outcome_aligned * Level + Position + (1|Ord
```

```
## fixed-effect model matrix is rank deficient so dropping 5 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```
model5_outcome <- lmer(outcome_confidence_abs ~ ChartType + outcome_aligned * Position + Level + (1|Ord
```

```
## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
## boundary (singular) fit: see help('isSingular')
```

```
model6_outcome <- lmer(outcome_confidence_abs ~ ChartType + (outcome_aligned * Level) + (outcome_aligne
```

```
## fixed-effect model matrix is rank deficient so dropping 5 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```
model7_outcome <- lmer(outcome_confidence_abs ~ ChartType + (outcome_aligned * Level) + (outcome_aligne
```

```
## fixed-effect model matrix is rank deficient so dropping 6 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```
model7_outcome <- lmer(outcome_confidence_abs ~ ChartType + (outcome_aligned * Level) + (outcome_aligne
```

```
## fixed-effect model matrix is rank deficient so dropping 6 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```
model8_outcome <- lmer(outcome_confidence_abs ~ ChartType + (outcome_aligned * Level) + (outcome_aligne
```

```
## fixed-effect model matrix is rank deficient so dropping 6 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```
anova(model0_outcome, model1_outcome, model2_outcome, model3_outcome, model4_outcome, model5_outcome, mo
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: df
## Models:
## model0_outcome: outcome_confidence_abs ~ ChartType + (1 | Order) + (1 | Slant)
## model1_outcome: outcome_confidence_abs ~ ChartType + outcome_aligned + (1 | Order) + (1 | Slant)
## model2_outcome: outcome_confidence_abs ~ ChartType + outcome_aligned + Level + Position + (1 | Order)
## model4_outcome: outcome_confidence_abs ~ ChartType + outcome_aligned * Level + Position + (1 | Order)
## model5_outcome: outcome_confidence_abs ~ ChartType + outcome_aligned * Position + Level + (1 | Order)
## model3_outcome: outcome_confidence_abs ~ ChartType + outcome_aligned + Level * Position + (1 | Order)
```

```
## model6_outcome: outcome_confidence_abs ~ ChartType + (outcome_aligned * Level) + (outcome_aligned * L
## model7_outcome: outcome_confidence_abs ~ ChartType + (outcome_aligned * Level) + (outcome_aligned * L
## model8_outcome: outcome_confidence_abs ~ ChartType + (outcome_aligned * Level) + (outcome_aligned * L
##                npar    AIC    BIC  logLik deviance   Chisq Df Pr(>Chisq)
## model0_outcome    5 4170.8 4193.1 -2080.4   4160.8
## model1_outcome    7 4163.8 4195.1 -2074.9   4149.8 10.9325  2   0.004227 **
## model2_outcome   10 4167.9 4212.5 -2074.0   4147.9  1.9121  3   0.590859
## model4_outcome   12 4171.2 4224.8 -2073.6   4147.2  0.7188  2   0.698102
## model5_outcome   12 4169.6 4223.2 -2072.8   4145.6  1.5929  0
## model3_outcome   13 4170.8 4228.8 -2072.4   4144.8  0.7970  1   0.371996
## model6_outcome   14 4172.9 4235.4 -2072.5   4144.9  0.0000  1   1.000000
## model7_outcome   15 4173.6 4240.5 -2071.8   4143.6  1.3047  1   0.253352
## model8_outcome   29 4178.9 4308.3 -2060.4   4120.9 22.7432 14   0.064578 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
summary(model1_outcome)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: outcome_confidence_abs ~ ChartType + outcome_aligned + (1 | Order) +
##     (1 | Slant)
##    Data: df
##
## REML criterion at convergence: 4149
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.08762 -0.69503  0.00568  0.48860  2.23968
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Slant    (Intercept)  0.00    0.000
##  Order    (Intercept)  0.00    0.000
##  Residual             38.57    6.211
## Number of obs: 640, groups:  Slant, 3; Order, 2
##
## Fixed effects:
##                            Estimate Std. Error t value
## (Intercept)              12.9654579  0.4502180  28.798
## ChartTypeLine            -0.0007285  0.4917685  -0.001
## outcome_alignedNA        -0.6483212  0.6239886  -1.039
## outcome_alignedUnaligned -1.8745712  0.5701376  -3.288
##
## Correlation of Fixed Effects:
##             (Intr) ChrtTL otc_NA
## ChartTypeLn -0.512
## otcm_lgndNA -0.518 -0.027
## otcm_lgndUn -0.554 -0.056  0.422
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```
#### appraisal ####

model0_author <- lmer(author_confidence_abs ~ ChartType + (1|Order) + (1|Slant), data = df)
```

```
## boundary (singular) fit: see help('isSingular')
```

```r
model1_author <- lmer(author_confidence_abs ~ ChartType + author_aligned + (1|Order) + (1|Slant), data =
```

```
## boundary (singular) fit: see help('isSingular')
```

```r
model2_author <- lmer(author_confidence_abs ~ ChartType + author_aligned + Level + Position + (1|Order)
```

```
## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
## boundary (singular) fit: see help('isSingular')
```

```r
model3_author <- lmer(author_confidence_abs ~ ChartType + author_aligned + Level * Position + (1|Order)
```

```
## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
## boundary (singular) fit: see help('isSingular')
```

```r
model4_author <- lmer(author_confidence_abs ~ ChartType + author_aligned * Level + Position + (1|Order)
```

```
## fixed-effect model matrix is rank deficient so dropping 5 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```r
model5_author <- lmer(author_confidence_abs ~ ChartType + author_aligned * Position + Level + (1|Order)
```

```
## fixed-effect model matrix is rank deficient so dropping 1 column / coefficient
## boundary (singular) fit: see help('isSingular')
```

```r
model6_author <- lmer(author_confidence_abs ~ ChartType + (author_aligned * Level) + (author_aligned * 
```

```
## fixed-effect model matrix is rank deficient so dropping 5 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```r
model7_author <- lmer(author_confidence_abs ~ ChartType + (author_aligned * Level) + (author_aligned * 
```

```
## fixed-effect model matrix is rank deficient so dropping 6 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```r
model8_author <- lmer(author_confidence_abs ~ ChartType + (author_aligned * Level) + (author_aligned * 
```

```
## fixed-effect model matrix is rank deficient so dropping 6 columns / coefficients
## boundary (singular) fit: see help('isSingular')
```

```r
anova(model0_author, model1_author, model2_author, model3_author, model4_author, model5_author, model6_
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: df
## Models:
## model0_author: author_confidence_abs ~ ChartType + (1 | Order) + (1 | Slant)
## model1_author: author_confidence_abs ~ ChartType + author_aligned + (1 | Order) + (1 | Slant)
## model2_author: author_confidence_abs ~ ChartType + author_aligned + Level + Position + (1 | Order) +
## model4_author: author_confidence_abs ~ ChartType + author_aligned * Level + Position + (1 | Order) +
## model5_author: author_confidence_abs ~ ChartType + author_aligned * Position + Level + (1 | Order) +
## model3_author: author_confidence_abs ~ ChartType + author_aligned + Level * Position + (1 | Order) +
## model6_author: author_confidence_abs ~ ChartType + (author_aligned * Level) + (author_aligned * Posi
## model7_author: author_confidence_abs ~ ChartType + (author_aligned * Level) + (author_aligned * Posi
## model8_author: author_confidence_abs ~ ChartType + (author_aligned * Level) + (author_aligned * Posi
##                npar    AIC    BIC  logLik deviance   Chisq Df Pr(>Chisq)
## model0_author    5 4302.5 4324.8 -2146.2   4292.5
## model1_author    7 4219.8 4251.0 -2102.9   4205.8 86.7201  2     <2e-16 ***
## model2_author   10 4225.2 4269.8 -2102.6   4205.2  0.5654  3     0.9043
## model4_author   12 4227.8 4281.3 -2101.9   4203.8  1.4035  2     0.4957
## model5_author   12 4228.8 4282.4 -2102.4   4204.8  0.0000  0
## model3_author   13 4230.8 4288.8 -2102.4   4204.8  0.0647  1     0.7992
## model6_author   14 4231.4 4293.9 -2101.7   4203.4  1.3270  1     0.2493
## model7_author   15 4229.2 4296.1 -2099.6   4199.2  4.2306  1     0.0397 *
## model8_author   29 4240.1 4369.5 -2091.0   4182.1 17.1242 14     0.2496
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(model7_author)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: author_confidence_abs ~ ChartType + (author_aligned * Level) +
##     (author_aligned * Position) + outcome_aligned + (1 | Order) +
##     (1 | Slant)
##    Data: df
##
## REML criterion at convergence: 4181.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.5344 -0.6586 -0.4050  0.6658  3.3080
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  Slant    (Intercept) 2.434e+00 1.560e+00
##  Order    (Intercept) 8.017e-09 8.954e-05
##  Residual             4.178e+01 6.464e+00
## Number of obs: 640, groups:  Slant, 3; Order, 2
##
## Fixed effects:
##                                    Estimate Std. Error t value
## (Intercept)                          7.4904     1.3460   5.565
## ChartTypeLine                        0.2309     0.5136   0.449
## author_alignedNA                    -3.8094     1.0348  -3.681
## author_alignedUnaligned             -5.4011     1.3557  -3.984
## Level3                               0.8913     0.8778   1.015
## Level2                               0.9087     0.9069   1.002
## PositionTitle                       -0.3203     0.7396  -0.433
```

```
## outcome_alignedUnaligned                 1.2419     0.5986   2.075
## author_alignedUnaligned:Level3          -1.2770     1.5639  -0.817
## author_alignedUnaligned:Level2          -1.7273     1.5600  -1.107
## author_alignedNA:PositionTitle           0.1142     2.2888   0.050
## author_alignedUnaligned:PositionTitle   0.6201     1.2437   0.499
##
## Correlation of Fixed Effects:
##             (Intr) ChrtTL ath_NA athr_U Level3 Level2 PstnTt otcm_U a_U:L3
## ChartTypeLn -0.158
## athr_lgndNA -0.381 -0.068
## athr_lgndUn -0.254 -0.048  0.337
## Level3      -0.287 -0.027  0.380  0.290
## Level2      -0.277 -0.017  0.365  0.292  0.455
## PositionTtl -0.254 -0.017  0.334  0.246 -0.069 -0.029
## otcm_lgndUn -0.190 -0.054  0.264 -0.038  0.012 -0.049  0.064
## athr_lgU:L3  0.151  0.030 -0.207 -0.627 -0.561 -0.255  0.042  0.009
## athr_lgU:L2  0.154  0.027 -0.209 -0.632 -0.264 -0.581  0.019  0.032  0.550
## athr_lNA:PT -0.317  0.027 -0.329 -0.078  0.022  0.009 -0.323 -0.023 -0.012
## athr_lgU:PT  0.151  0.008 -0.198 -0.486  0.041  0.018 -0.594 -0.047  0.001
##             a_U:L2 a_NA:P
## ChartTypeLn
## athr_lgndNA
## athr_lgndUn
## Level3
## Level2
## PositionTtl
## otcm_lgndUn
## athr_lgU:L3
## athr_lgU:L2
## athr_lNA:PT -0.005
## athr_lgU:PT  0.008  0.192
## fit warnings:
## fixed-effect model matrix is rank deficient so dropping 6 columns / coefficients
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

Exploratory Analysis

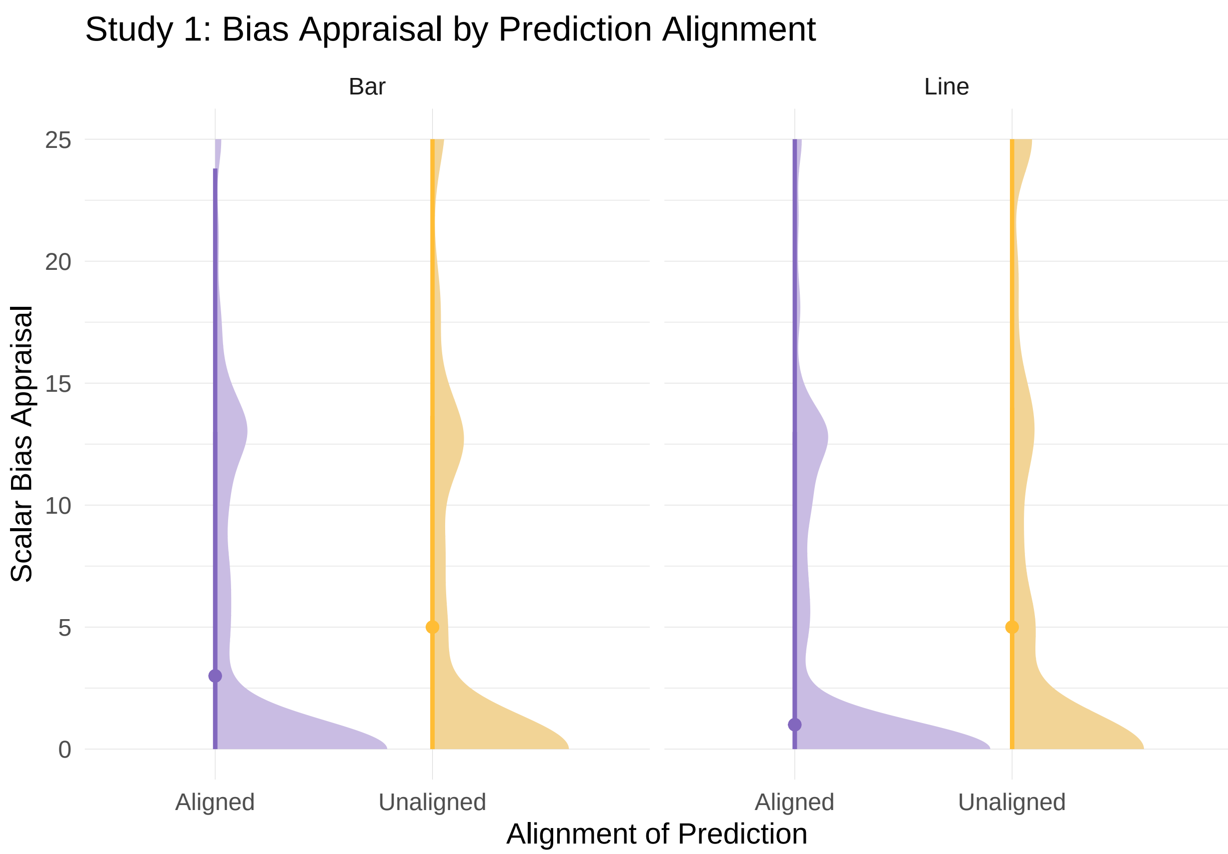
Author and Outcome Rating Relation

After evaluating these hypotheses, we were interested in the possible relationship between the belief in author bias and the rating of the confidence in outcome. In other words, if the reader disagreed with the outcome presented by the author, did they perceive the author as more biased? Were those readers more confident in their ratings if they perceived the author as more biased (i.e., ignoring the text in their rating due to author bias)? And vice versa (i.e., more confident because author bias is low)?

If this were the case, we would expect to see:

- higher average ratings for author leaning for unaligned outcome responses
- positive correlation between author leaning and outcome ratings for unaligned responses
- negative correlation between author leaning and outcome ratings for aligned responses

```
ggplot(subset(df, treatment == "Treatment"), aes(x = outcome_aligned, y = author_confidence_abs, fill =
  stat_halfeye(size = 2)+
  scale_fill_manual(values = c(a1, u1))+
  scale_color_manual(values = c(a, u))+
  labs(
    title = "Study 1: Bias Appraisal by Prediction Alignment",
    # subtitle = "Did people perceive the author as having a more extreme position\nif they disagreed w
    y = "Scalar Bias Appraisal",
    x = "Alignment of Prediction"
  )+
  facet_grid(. ~ChartType)+
  theme(legend.position = 'none',
  panel.grid.major = element_line(color = "grey85", size = 0.1),  # Lighten and thin major gridlines
  panel.grid.minor = element_line(color = "grey85", size = 0.05))  # Lighten and thin minor gridlines
```




```
ggplot(subset(df, treatment == "Treatment"), aes(x = outcome_aligned, y = author_confidence_abs, fill =
  geom_bar(stat = "summary", fun = "mean")+
  scale_fill_manual(values = c(a, u))+
  labs(
    title = "Mean Author Position Rating by Outcome Alignment",
    subtitle = "Did people perceive the author as having a more extreme position\nif they disagreed wit
    y = "Perceived Likelihood of Author Position",
    x = "Alignment of Response",
    fill = "Response Alignment"
  )+
```

```r
  facet_grid(. ~ChartType)
```

Without significance testing, we can observe that ratings seem overall higher for author bias when the response is unaligned with the message presented by the author. However, there is not a correlation between bias ratings and outcome ratings for either of the potential hypotheses laid out.

We then go to look further at how bias ratings may have varied according to whether the participant agreed (or was aligned with) the outcome presented in the text.

```r
subset(bar, treatment == "Treatment") %>%
  group_by(outcome_aligned) %>%
  summarise(
    median = median(author_confidence_abs),
    mean = mean(author_confidence_abs)
  )
```

```
## # A tibble: 2 x 3
##   outcome_aligned median  mean
##   <chr>            <dbl> <dbl>
## 1 Aligned              3  6.26
## 2 Unaligned            5  7.01
```

```r
subset(line, treatment == "Treatment") %>%
  group_by(outcome_aligned) %>%
  summarise(
```

```
    median = median(author_confidence_abs),
    mean(author_confidence_abs)
  )
```

```
## # A tibble: 2 x 3
##   outcome_aligned median `mean(author_confidence_abs)`
##   <chr>            <dbl>                         <dbl>
## 1 Aligned              1                          5.94
## 2 Unaligned            5                          6.85
```

For bar charts, we find evidence that participants rated author bias as higher if they disagreed (were unaligned) with the outcome presented by the text in the chart. We do not find this effect for line charts, however.

No significant differences are found between specific conditions for author leaning ratings.