Final Project

Kevin Yi, Missael Vasquez, Patrick Abousleiman, Maged Saad

04/02/2025

Contents

1 Abstract

2 Introduction and Purpose

3 Literature Review

4 Power Analysis

With 120 sample in test and 120 sample in control, we would detect an ATE of 10% difference only 38% of the time and an ATE of 25% difference 99.5% of the time. Based on pilot and our approximate intended sample size of 120 we would detect a true statistically significant result of a 15% ATE 74.1% of the time.

5 Hypothesis

Null Hypothesis H_0: Either emoji treatment has no direct impact on willingness to sell or message response rate:

$$H_0 : \beta_{FE} \ = \ 0 \ \ or \ \ \beta_{NFE} = 0$$

Alternative Hypothesis H_1: Either of the treatments has an effect on the willingness to sell or message response rate:

$$H_1 : \beta_{FE} \ \neq \ 0 \ \ or \ \ \beta_{NFE} \neq 0$$

6 Methods/Modeling

6.1 Data Cleaning

```
# Combine treatment categories to one column
# d[, emoji := ifelse(non_facial_emoji == 1, "Non-Facial",
#                ifelse(facial_emoji == 1, "Facial",
#                ifelse(no_emoji == 1, "Control", NA_character_)))]
# simplified code
d[, emoji := fcase(
  non_facial_emoji == 1, "Non-Facial",
  facial_emoji == 1, "Facial",
  no_emoji == 1, "Control"
)]

# Drop unused column
d[, c("non_facial_emoji", "facial_emoji", "no_emoji", "seller_name", "counter", "response time", "condi
      "price", "V18", "Rough Numbers", "V20" , "V21") := NULL]



# Normalize all day values.
d[day == "Fri", day := "Friday"]
d[day == "Mon", day := "Monday"]
d[day == "Sat", day := "Saturday"]
d[day == "Thu", day := "Thursday"]
d[day == "Tue", day := "Tuesday"]

# refactoring for easier interpretability
d[, day := factor(day, levels = c("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Sa
d[, seller_gender := factor(seller_gender, levels = c("M", "F"), labels = c("M", "F"))] #use male as re
d[, emoji := factor(emoji, levels = c("Control", "Non-Facial", "Facial"))]


# Convert to ITime using data.table's built-in function:
d[, msg_time := as.ITime(msg_time, format = "%H:%M:%S")]


# create weekend/weekday
d[, dayType := ifelse(day %in% c("Saturday", "Sunday"), "Weekend", "Weekday")]
```

```r
# create number col
d[, number := .I]


# reorder so relevant rows are closer together
d <- d[, c(12, 1:3, 11, 4:10)]

setnames(d, to_lower_camel_case(names(d)))

head(d)
```

```
##    number  buyer  msgTime      day dayType priceDrop justListed location
##     <int> <char>  <ITime>   <fctr>  <char>     <int>      <int>   <char>
## 1:      1  Kevin 17:38:00 Saturday Weekend         0          0   Hawaii
## 2:      2  Kevin 17:40:00 Saturday Weekend         0          0   Hawaii
## 3:      3  Kevin 17:41:00 Saturday Weekend         0          0   Hawaii
## 4:      4  Kevin 17:42:00 Saturday Weekend         0          1   Hawaii
## 5:      5  Kevin 17:43:00 Saturday Weekend         1          0   Hawaii
## 6:      6  Kevin 17:44:00 Saturday Weekend         1          0   Hawaii
##    sellerGender response willingToReducePrice      emoji
##          <fctr>    <int>                <int>     <fctr>
## 1:            M        1                    1     Facial
## 2:            F        0                    0    Control
## 3:            F        1                    1     Facial
## 4:            F        1                    1    Control
## 5:            M        0                    0 Non-Facial
## 6:            F        0                    0    Control
```

```r
colSums(is.na(d))
```

```
##               number                buyer              msgTime
##                    0                    0                    0
##                  day              dayType            priceDrop
##                    0                    0                    0
##           justListed             location         sellerGender
##                    0                    0                    0
##             response willingToReducePrice                emoji
##                    0                    0                    0
```

6.2 Data Exploration

```r
# willingness to reduce price table
price_red_tbl <- d %>%
  select(-location, -response, -number) %>%
  tbl_summary(
    by = willingToReducePrice,
    label = list(
      buyer = "Prospective Buyer",
      msgTime = "Hour of Message",
      day = "Day of Message",
      dayType = "Day Type",
      priceDrop = "Price Drop",
      justListed = "Just Listed",
      sellerGender = "Gender of Seller",
      emoji = "Emoji Treatment"
    ),
    type = list(msgTime ~ "continuous")
  ) %>%
  modify_caption("**Willingness for Seller Reduce Price Summary Table**")

# response table
response_tbl <- d %>%
  select(-location, - willingToReducePrice, -number) %>%
  tbl_summary(
    by = response,
    label = list(
      buyer = "Prospective Buyer",
      msgTime = "Hour of Message",
      day = "Day of Message",
      dayType = "Day Type",
      priceDrop = "Price Drop",
      justListed = "Just Listed",
      sellerGender = "Gender of Seller",
      emoji = "Emoji Treatment"
    ),
    type = list(msgTime ~ "continuous")
  ) %>%
  modify_caption("**Response from Seller Summary Table**")

combined_tbl <- tbl_merge(list(response_tbl, price_red_tbl), tab_spanner = c("Responses to Messages", "
```

```
response_tbl
```

```
price_red_tbl
```

6.2.1 Plots

```
#faceted bar plots for all variables relative to balance within treatment groups
# now will exclude msgTime as tricky to summarize as a bar plot
faceted_cols <- d %>%
  select(-location, -number, -emoji, -msgTime) %>%
  names()

#faceted cols
for (var in faceted_cols) {
  var_clean <- gsub("_", "", var)

  p <- ggplot(d, aes(x = emoji, fill= emoji)) +
    geom_bar() +
    facet_wrap(as.formula(paste0("~", var))) +
    labs(
```

Table 1: **Response from Seller Summary Table**

Characteristic	**0** N = 103[1]	**1** N = 249[1]
Prospective Buyer		
Kevin	23 (22%)	61 (24%)
Maged	44 (43%)	44 (18%)
Missael	18 (17%)	72 (29%)
Patrick	18 (17%)	72 (29%)
Hour of Message	18:00:00 (17:00:00, 20:00:00)	18:00:00 (17:00:00, 21:00:00)
Day of Message		
Sunday	4 (3.9%)	12 (4.8%)
Monday	9 (8.7%)	51 (20%)
Tuesday	6 (5.8%)	21 (8.4%)
Wednesday	12 (12%)	15 (6.0%)
Thursday	21 (20%)	64 (26%)
Friday	13 (13%)	44 (18%)
Saturday	38 (37%)	42 (17%)
Day Type		
Weekday	61 (59%)	195 (78%)
Weekend	42 (41%)	54 (22%)
Price Drop	13 (13%)	73 (29%)
Just Listed	10 (9.7%)	19 (7.6%)
Gender of Seller		
M	60 (58%)	169 (68%)
F	43 (42%)	80 (32%)
Emoji Treatment		
Control	25 (24%)	96 (39%)
Non-Facial	37 (36%)	80 (32%)
Facial	41 (40%)	73 (29%)

[1] n (%); Median (Q1, Q3)

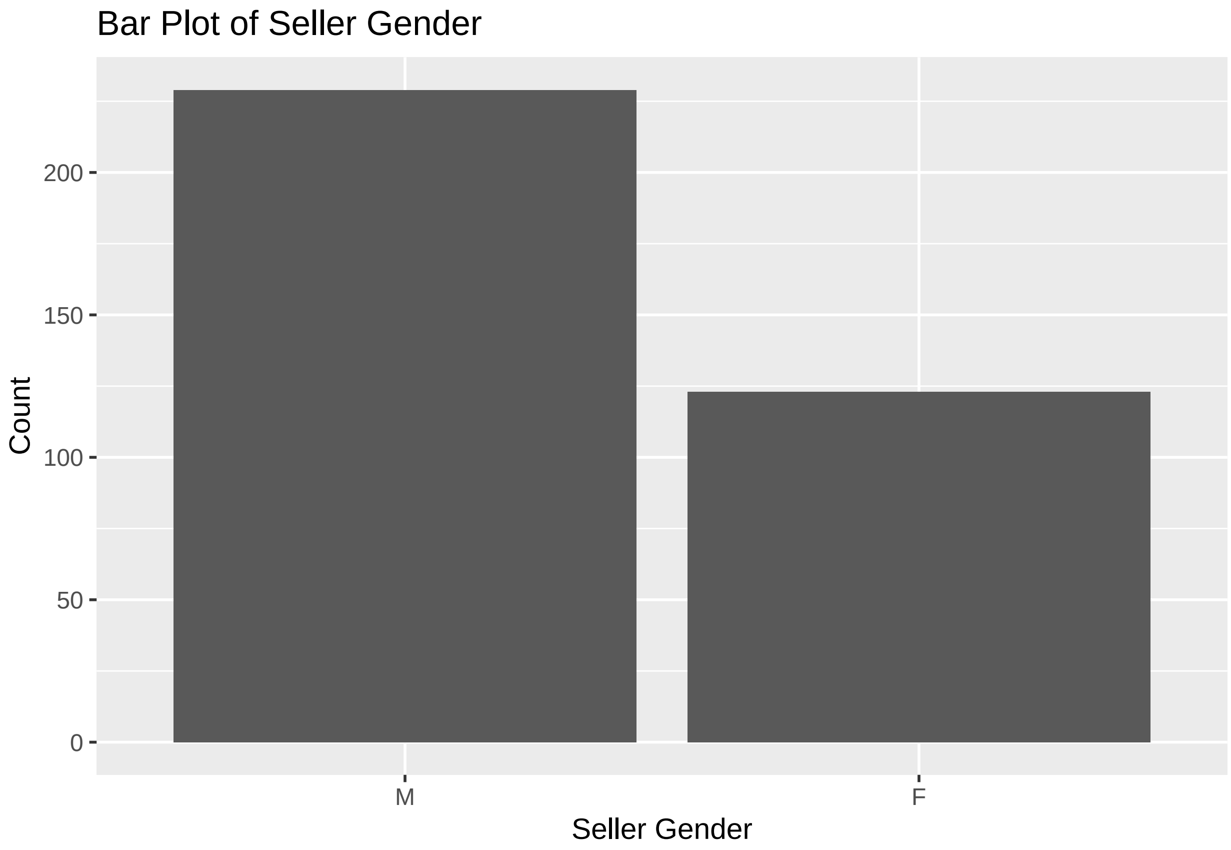
```
      title = paste("Emoji Treatment by", var),
      x = "",
      y = "Count"
    ) +
    theme_minimal() +
    theme(panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          legend.position = "none")

  assign(paste0("plot", var_clean), p)
}

# rework msgTime plot provide from kevin to show msgTime per emoji treatment
plotmsgTime <- ggplot(d, aes(x = as.numeric(msgTime), fill = emoji)) +
```

Table 2: **Willingness for Seller Reduce Price Summary Table**

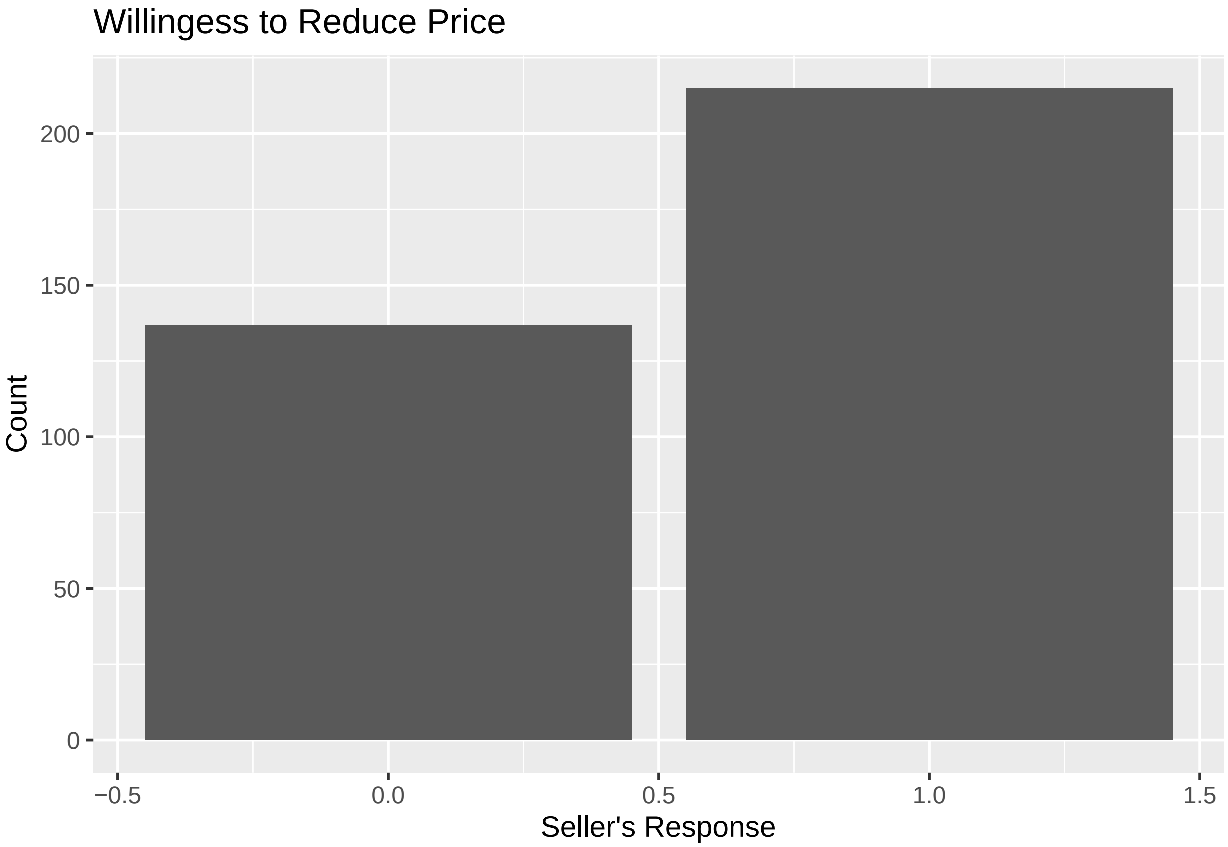
Characteristic	**0** N = 137[1]	**1** N = 215[1]
Prospective Buyer		
Kevin	28 (20%)	56 (26%)
Maged	54 (39%)	34 (16%)
Missael	27 (20%)	63 (29%)
Patrick	28 (20%)	62 (29%)
Hour of Message	18:00:00 (17:00:00, 21:00:00)	18:00:00 (17:00:00, 20:00:00)
Day of Message		
Sunday	4 (2.9%)	12 (5.6%)
Monday	15 (11%)	45 (21%)
Tuesday	9 (6.6%)	18 (8.4%)
Wednesday	16 (12%)	11 (5.1%)
Thursday	31 (23%)	54 (25%)
Friday	21 (15%)	36 (17%)
Saturday	41 (30%)	39 (18%)
Day Type		
Weekday	92 (67%)	164 (76%)
Weekend	45 (33%)	51 (24%)
Price Drop	22 (16%)	64 (30%)
Just Listed	11 (8.0%)	18 (8.4%)
Gender of Seller		
M	82 (60%)	147 (68%)
F	55 (40%)	68 (32%)
Emoji Treatment		
Control	35 (26%)	86 (40%)
Non-Facial	51 (37%)	66 (31%)
Facial	51 (37%)	63 (29%)

[1] n (%); Median (Q1, Q3)

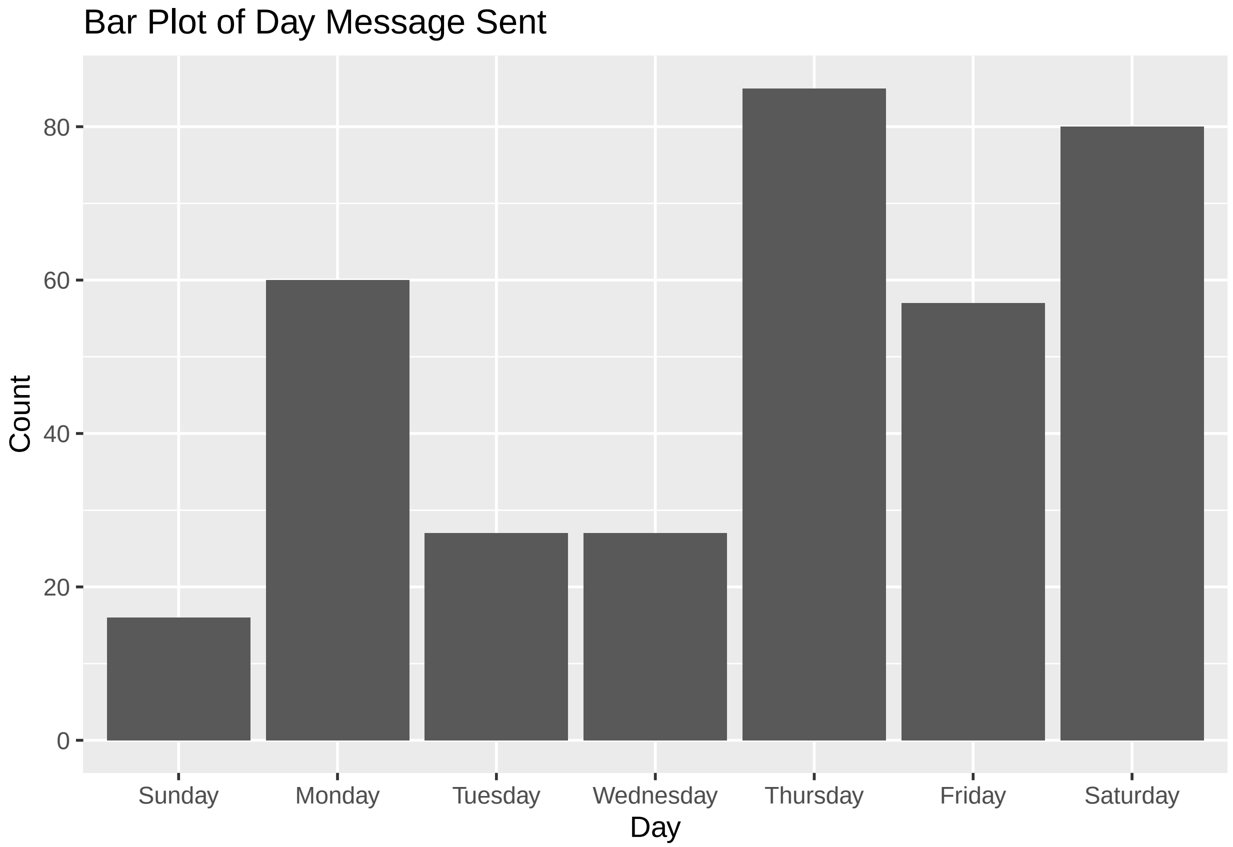
```r
  geom_histogram(binwidth = 3600, alpha = 0.4, boundary = 0, closed = "left") +
  scale_x_continuous(
    breaks = seq(0, 24 * 3600, by = 3600),
    labels = function(x) format(as.ITime(x), "%H:%M"),
    expand = expansion(add = 0)
  ) +
  labs(
    title = "Distribution of Message Times by Emoji Treatment",
    x = "Time of Day (Hours)",
    y = "Count",
    fill = "Emoji Treatment"
  ) +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(),
    axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)
  )

# Bar plot of seller gender
gender_plot<- ggplot(d, aes(x = sellerGender)) +
  geom_bar() +
  labs(
    title = "Bar Plot of Seller Gender",
    x = "Seller Gender",
    y = "Count"
  )


# Bar plot of day
day_plot <- ggplot(d, aes(x = willingToReducePrice)) +
  geom_bar() +
  labs(
    title = "Willingess to Reduce Price",
    x = "Seller's Response",
    y = "Count"
  )


# Bar plot of Response
response_plot <- ggplot(d, aes(x = day)) +
  geom_bar() +
  labs(
    title = "Bar Plot of Day Message Sent",
    x = "Day",
    y = "Count"
  )


# Bar plot of location
loc_plot <- ggplot(d, aes(x = location)) +
  geom_bar() +
  labs(
    title = "Bar Plot of Location",
    x = "Location",
    y = "Count"
  )


#Histogram of time message was sent
#adjusted original off-centering
time_plot <- ggplot(d, aes(x = as.numeric(msgTime))) +
  geom_histogram(binwidth = 3600, boundary = 0, closed = "left") +
  scale_x_continuous(
    breaks = seq(0, 24 * 3600, by = 3600),
    labels = function(x) format(as.ITime(x), "%H:%M"),
    expand = expansion(add = 0)
  ) +
  labs(
    title = "Distribution of Message Times",
    x = "Time of Day (Hours)",
    y = "Count"
  ) +
  theme(
    axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)
  )


gender_plot
```



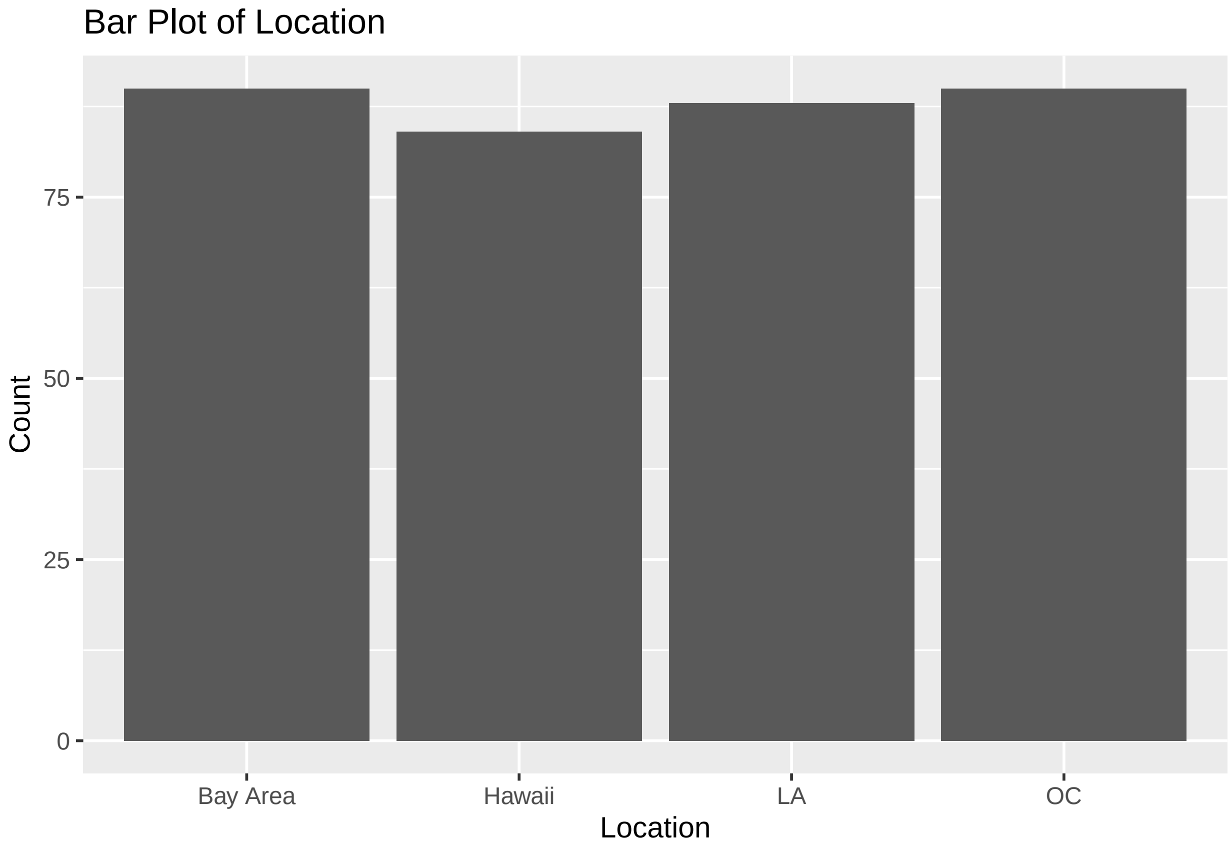
```
day_plot
```



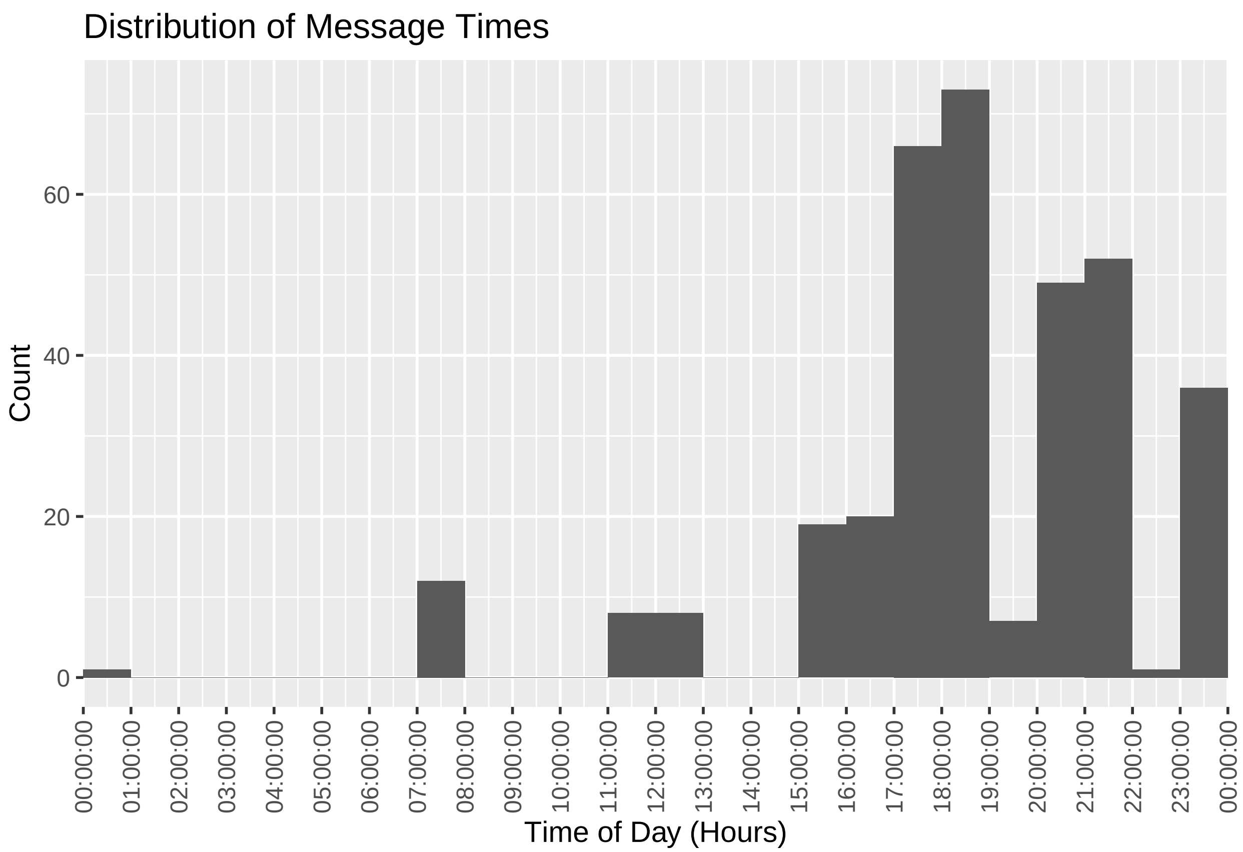
```
response_plot
```


Bar Plot of Day Message Sent
Count
0
20
40
60
80
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Day

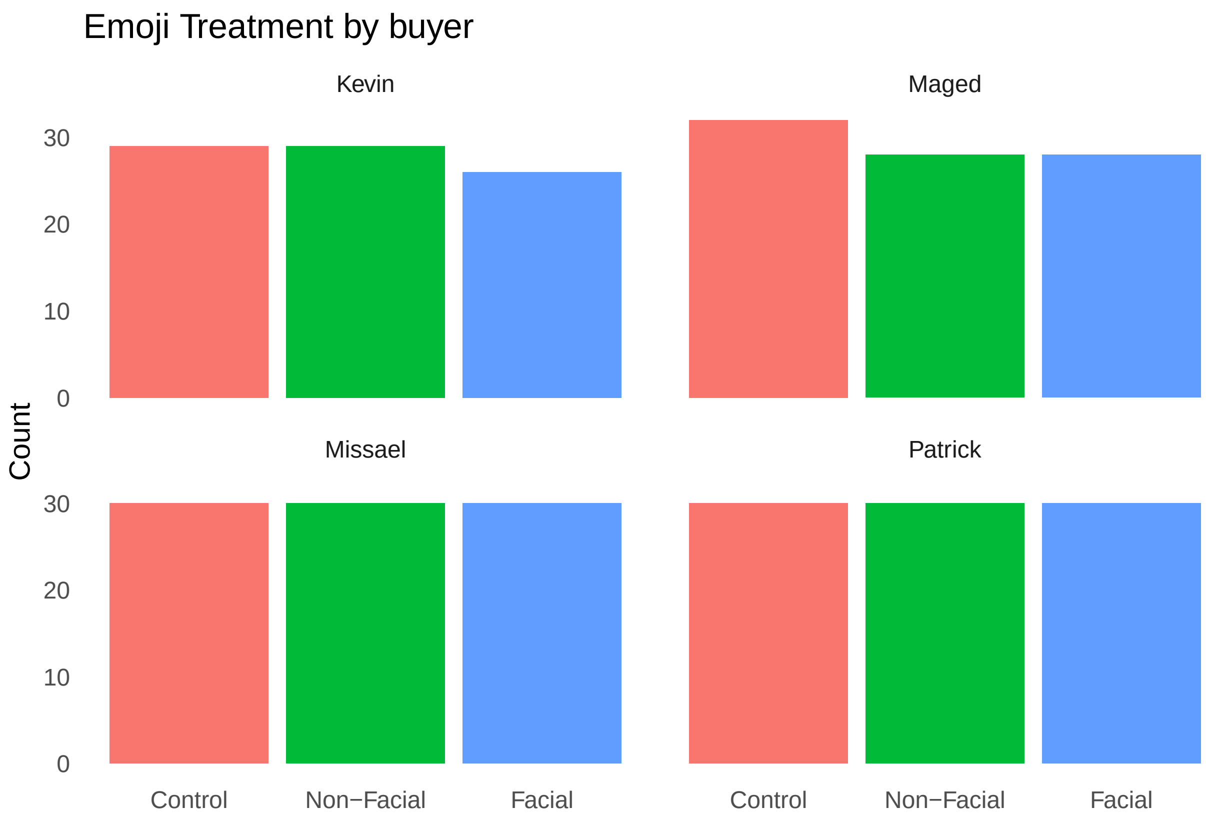
```
loc_plot
```


Bar Plot of Location
Count
75
50
25
0
Bay Area
Hawaii
LA
OC
Location

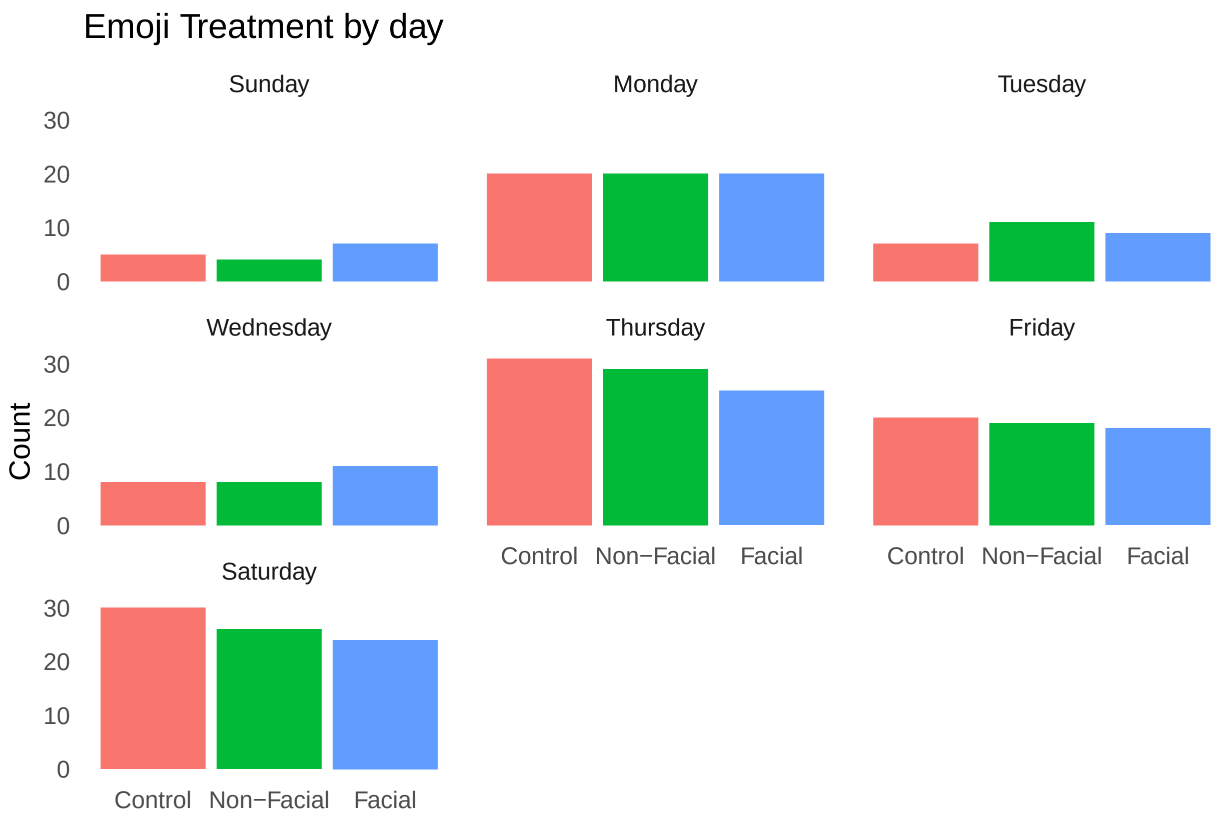
```
time_plot
```


Distribution of Message Times
Count
60
40
20
0
00:00:00
01:00:00
02:00:00
03:00:00
04:00:00
05:00:00
06:00:00
07:00:00
08:00:00
09:00:00
10:00:00
11:00:00
12:00:00
13:00:00
14:00:00
15:00:00
16:00:00
17:00:00
18:00:00
19:00:00
20:00:00
21:00:00
22:00:00
23:00:00
00:00:00
Time of Day (Hours)

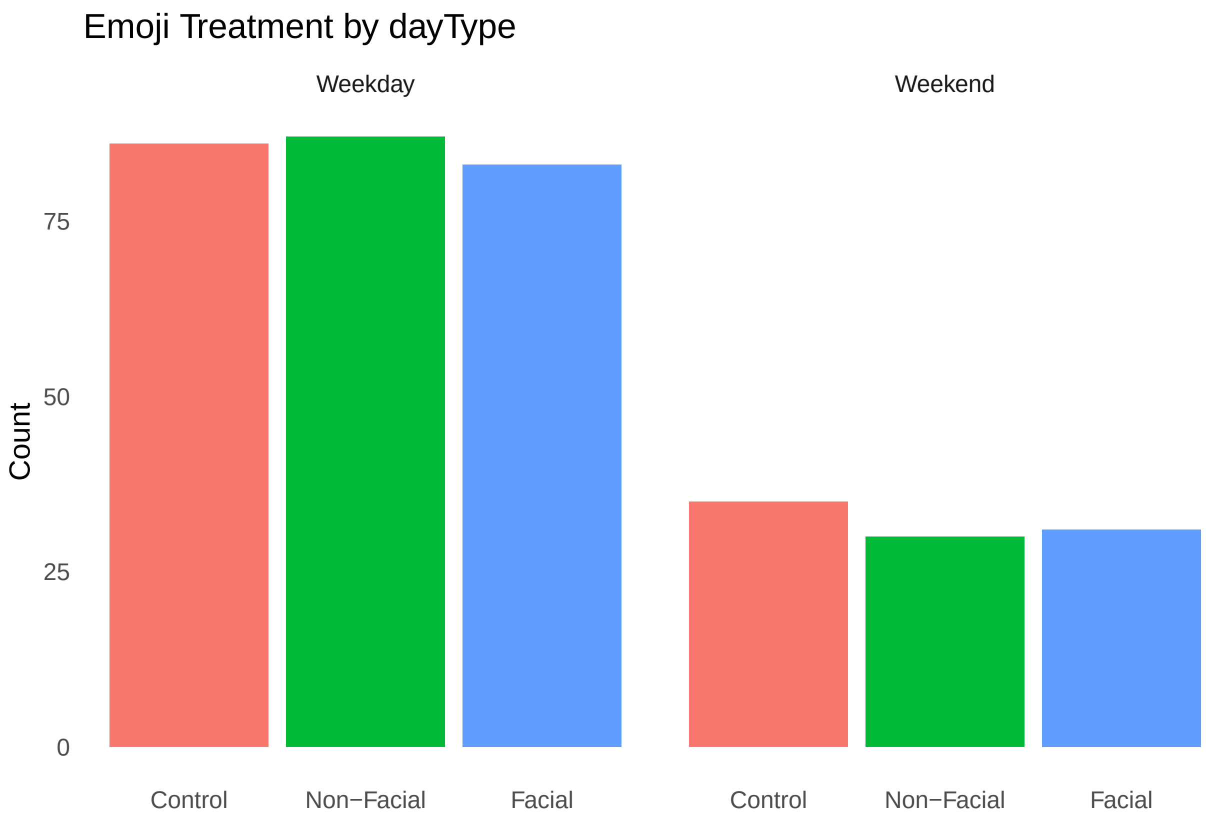
```
plotbuyer
```


Emoji Treatment by buyer
Kevin
Maged
Missael
Patrick
Count
0
10
20
30
0
10
20
30
Control
Non−Facial
Facial
Control
Non−Facial
Facial

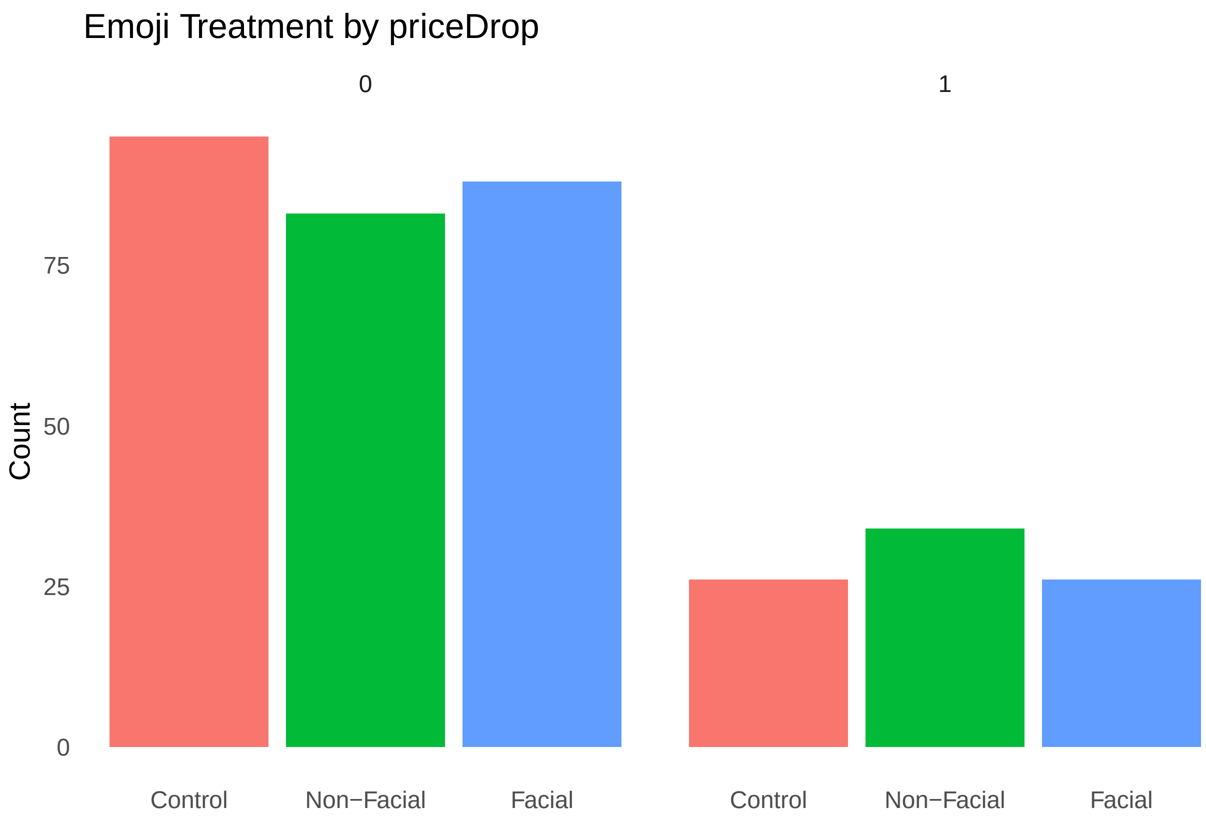
```
plotday
```



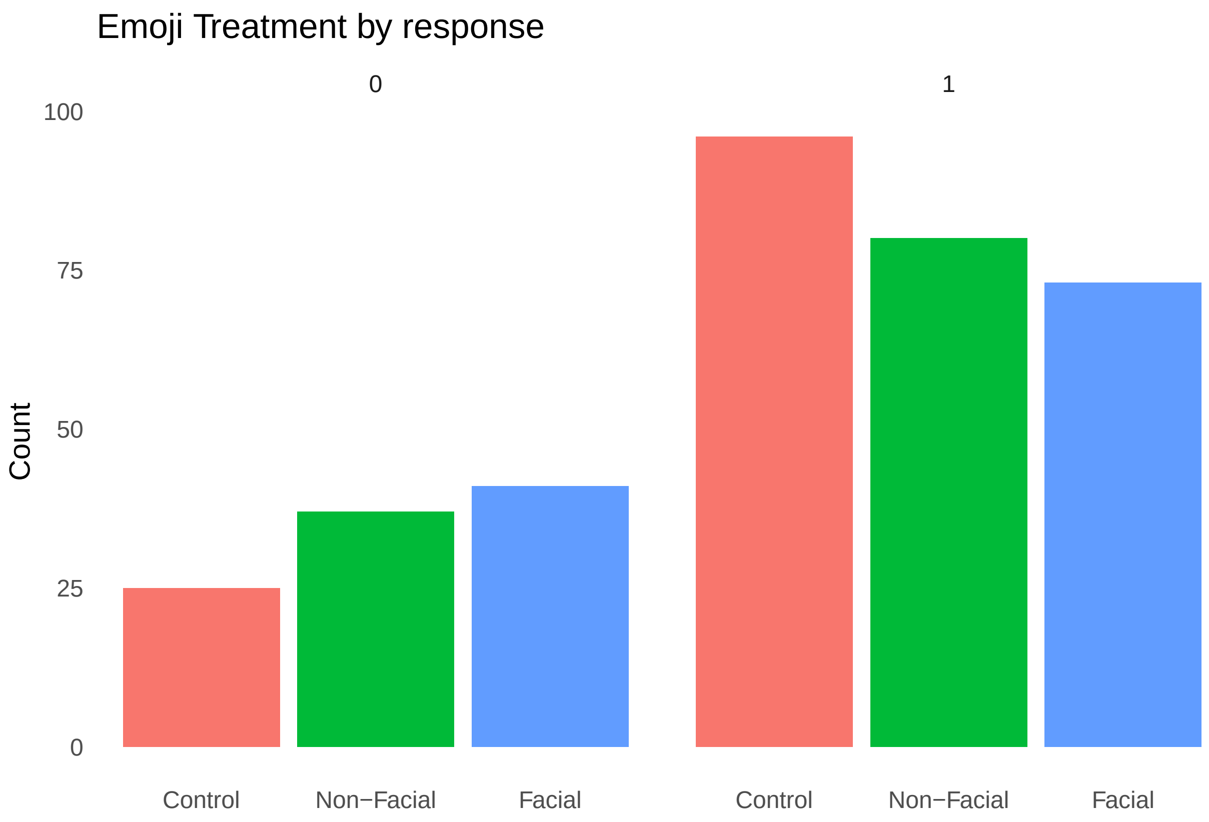
```
plotdayType
```



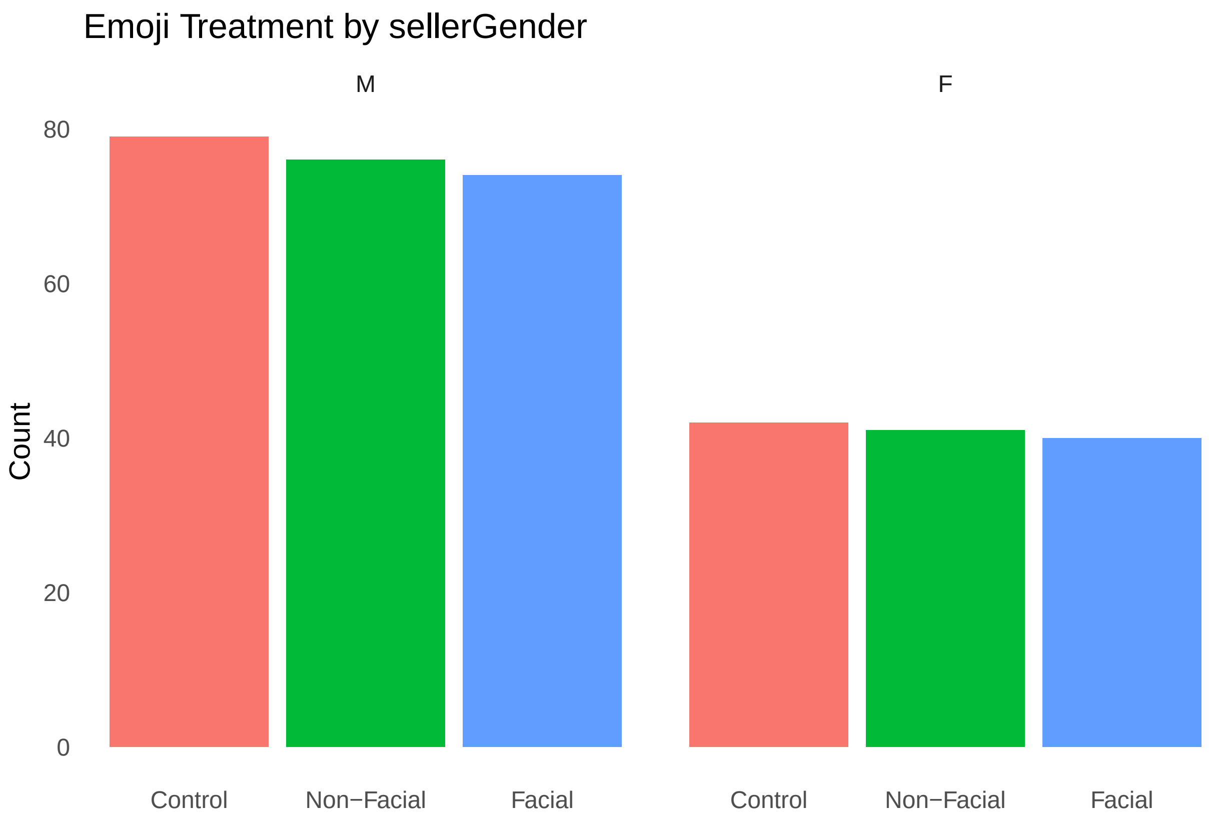
```
plotpriceDrop
```



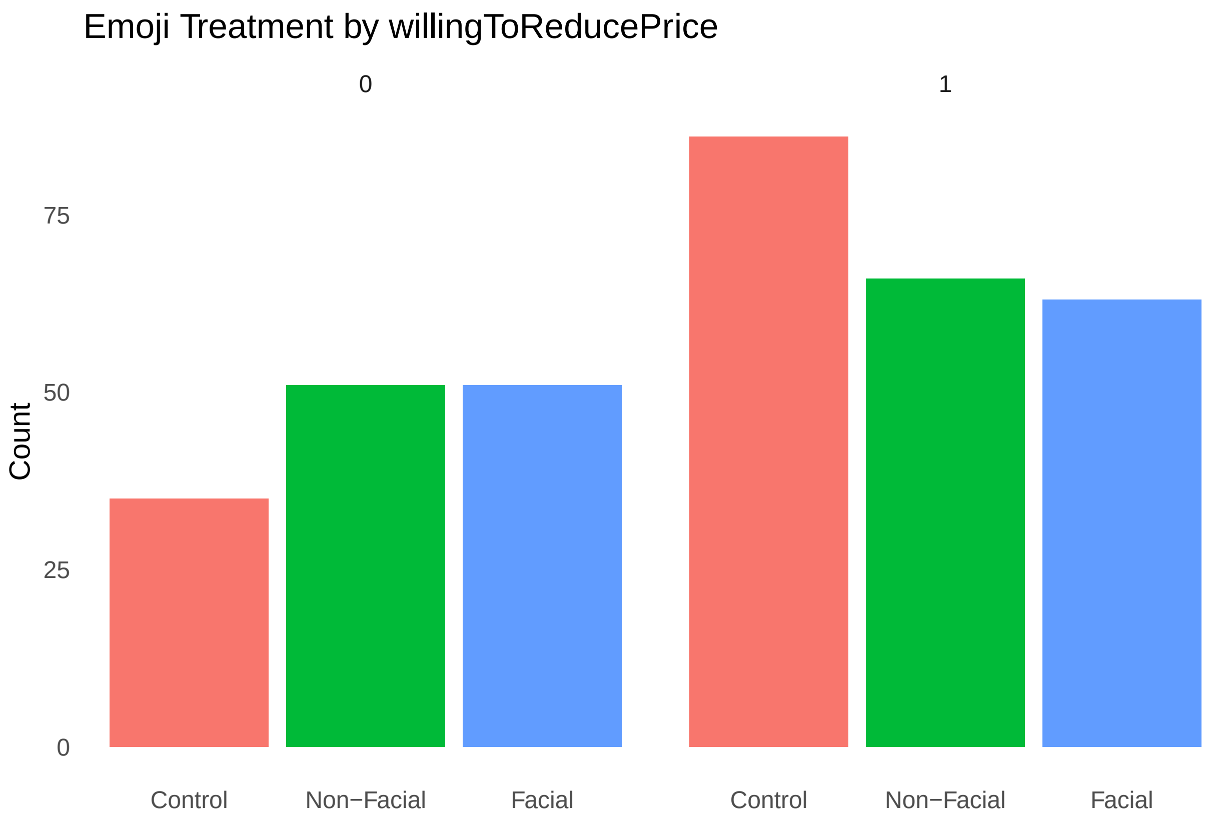
```
plotresponse
```



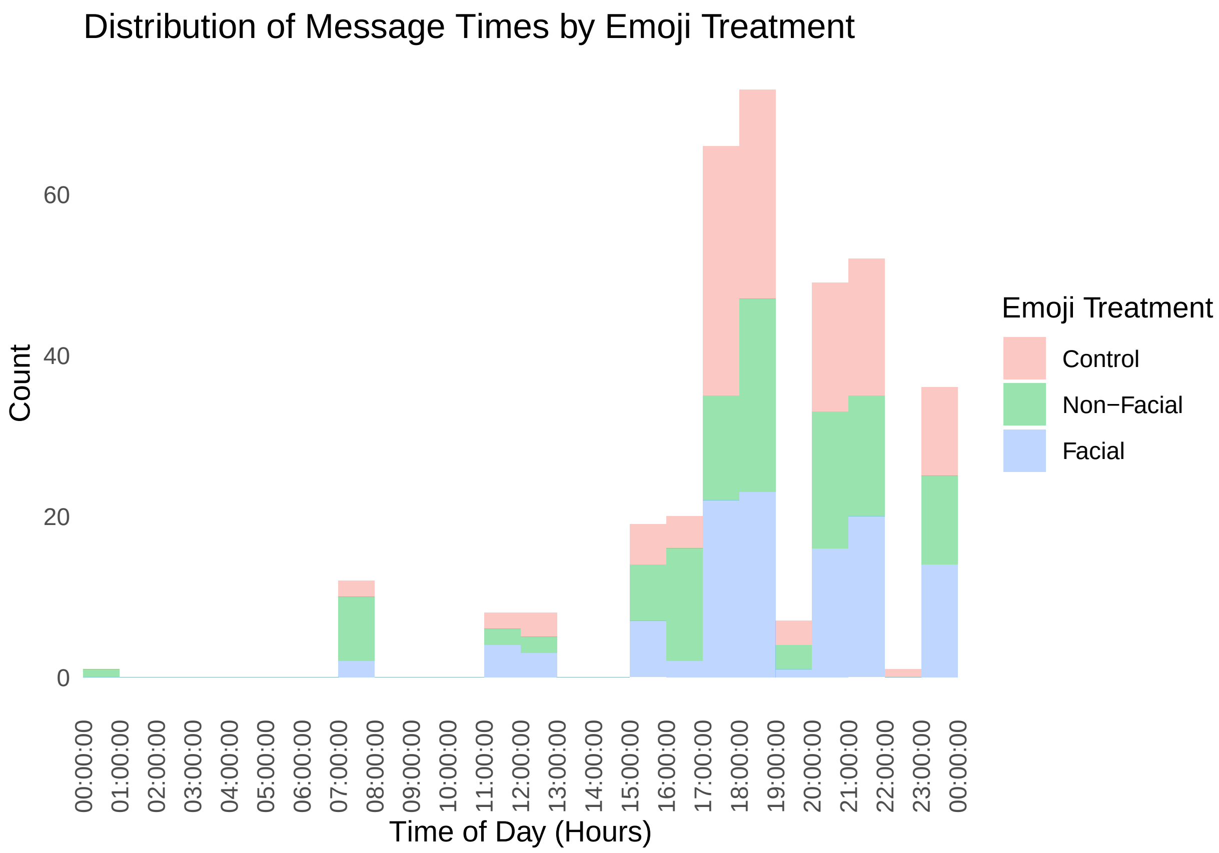
```
plotsellerGender
```



```
plotwillingToReducePrice
```



```
plotmsgTime
```



6.2.2 Combined Plots for a visual confirmation of even dist by treatments

Just for a practical understanding of the plots

```
combinedplots <-
  (plotbuyer | plotdayType | plotpriceDrop) /
  (plotsellerGender | plotday) /
  (plotmsgTime)
```

```
combinedplots
```

```
# # landscape mode
# ls_combinedplots <- paste0(
#   "\\begin{landscape}\n",
#   combinedplots,
#   "\n\\end{landscape}"
# )
#
# # display
# cat(ls_combinedplots)
```

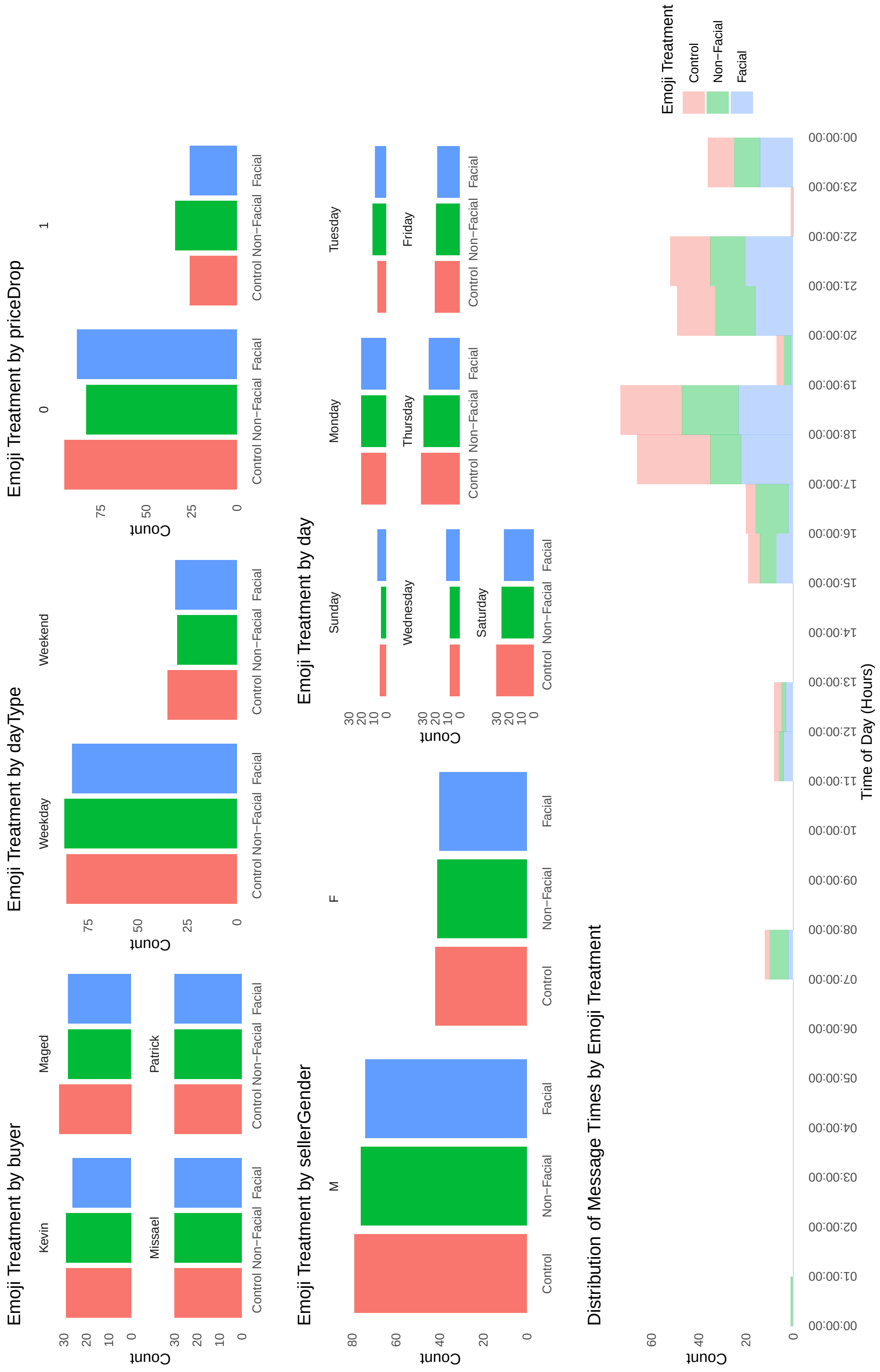


Figure 1: Combined Treatment Distribution

Table 3: Model 1 with Robust SE

Characteristic	Beta	95% CI[1]	p-value
Emoji Treatment			
Control	—	—	
Non-Facial	-0.11	-0.22, 0.00	0.055
Facial	-0.15	-0.27, -0.04	**0.009**

[1] CI = Confidence Interval

6.2.3 Model for Response

```
model_response_1 <- d[ , lm(response~ emoji)]

#showing robust output to compare against tidy model_response_ output

# robust se
model_response_1_robust <- coeftest(model_response_1, vcov = vcovHC(model_response_1, "HC1"))
model_response_1_robust
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error t value  Pr(>|t|)
## (Intercept)      0.793388   0.036965 21.4635 < 2.2e-16 ***
## emojiNon-Facial -0.109628   0.056837 -1.9288  0.054564 .
## emojiFacial     -0.153038   0.058343 -2.6231  0.009097 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# output table with robust se
tbl_model_response_1 <-
  tbl_regression(
    model_response_1,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model_response_1, "HC1"), ...),
    label = list(emoji = "Emoji Treatment"),
    caption = NULL
    ) %>%
  bold_p() %>%
  modify_caption("Model 1 with Robust SE")

tbl_model_response_1
```

```
model_response_2 <- d[ , lm(response~ emoji + sellerGender )]

# if you want to compare to check robust std
# model_response_2_robust <- coeftest(model_response_2, vcov = vcovHC(model_response_2, "HC1"))
# model_response_2_robust
```

Table 4: Model 2 with Robust SE

Characteristic	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment			
Control	—	—	
Non-Facial	-0.11	-0.22, 0.00	0.054
Facial	-0.15	-0.27, -0.04	**0.009**
Gender of Seller			
M	—	—	
F	-0.09	-0.19, 0.01	0.094

[1] CI = Confidence Interval

```r
#output tbl with robust se
tbl_model_response_2 <-
  tbl_regression(
    model_response_2,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model_response_2, "HC1"), ...),
    label = list(emoji = "Emoji Treatment",
                 sellerGender = "Gender of Seller")) %>%
  bold_p() %>%
  modify_caption("Model 2 with Robust SE")
tbl_model_response_2
```

```r
model_response_3 <- d[ , lm(response~emoji +
                    buyer + msgTime + day + priceDrop +
                    justListed + sellerGender)]


# if you want to compare to check robust std
# model_response_3_robust <- coeftest(model_response_3, vcov = vcovHC(model_response_3, "HC1"))
# model_response_3_robust

#output table with robust se
tbl_model_response_3 <-
  tbl_regression(
    model_response_3,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model_response_3, "HC1"), ...),
    label = list(emoji = "Emoji Treatment",
                 sellerGender = "Gender of Seller",
                 buyer = "Prospective Buyer",
                 msgTime = "Hour of Message",
                 day = "Day of Message",
                 priceDrop = "Price Drop",
                 justListed = "Just Listed")
    ) %>%
  bold_p() %>%
```

Table 5: Model 3 with Robust SE

Characteristic	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment			
Control	—	—	
Non-Facial	-0.13	-0.24, -0.02	**0.018**
Facial	-0.17	-0.27, -0.06	**0.002**
Prospective Buyer			
Kevin	—	—	
Maged	-0.20	-0.38, -0.02	**0.033**
Missael	0.05	-0.15, 0.25	0.63
Patrick	-0.05	-0.32, 0.22	0.72
Hour of Message	0.00	0.00, 0.00	0.57
Day of Message			
Sunday	—	—	
Monday	0.07	-0.34, 0.48	0.74
Tuesday	0.03	-0.34, 0.40	0.88
Wednesday	-0.06	-0.55, 0.43	0.81
Thursday	0.01	-0.35, 0.36	0.97
Friday	-0.03	-0.38, 0.33	0.89
Saturday	-0.22	-0.60, 0.16	0.26
Price Drop	0.21	0.11, 0.31	**<0.001**
Just Listed	-0.02	-0.21, 0.17	0.82
Gender of Seller			
M	—	—	
F	-0.10	-0.21, 0.00	0.050

[1] CI = Confidence Interval

```
  modify_caption("Model 3 with Robust SE")
tbl_model_response_3
```

```
# Checking for randomness in covariate
# Do a F-test  to compare full model_response_ to model_response_ with no predictive features
null_model_response <- lm(response ~ 1, data = d)
full_model_response <- lm(response ~ 1+ emoji + buyer + msgTime + day +
                    priceDrop + justListed + sellerGender, data = d)

# Perform an F-test
anova_mod_response <- anova(null_model_response, full_model_response, test = 'F')
anova_mod_response
```

```
## Analysis of Variance Table
##
## Model 1: response ~ 1
## Model 2: response ~ 1 + emoji + buyer + msgTime + day + priceDrop + justListed +
##     sellerGender
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)    
## 1    351 72.861                                  
## 2    336 59.808 15    13.053 4.8888 1.197e-08 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

```
# Check for multicollinearty
vif_values_response <- vif(full_model_response)
vif_values_response
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## emoji         1.060064  2        1.014689
## buyer        25.441014  3        1.714967
## msgTime       2.945019  1        1.716106
## day          29.878439  6        1.327231
## priceDrop     1.071234  1        1.035004
## justListed    1.527857  1        1.236065
## sellerGender  1.053007  1        1.026161
```

6.2.4 All Response Models Combined

```
combined_response_tbl<- tbl_merge(list(tbl_model_response_1, tbl_model_response_2, tbl_model_response_3
                                  tab_spanner = c("$Model_1$ : **Base**",
                                                  "$Model_2$ : **Middle**",
                                                  "$Model_3$ : **Saturated**")) %>%
  modify_caption("All Response Models") %>%
  as_gt() %>%
  as_latex()

# landscape mode
ls_combined_response_tbl <- paste0(
  "\\begin{landscape}\n",
  combined_response_tbl,
  "\n\\end{landscape}"
)

# display
cat(ls_combined_response_tbl)
```

Table 6: All Response Models

	$Model_1$: **Base**			$Model_2$: **Middle**			$Model_3$: **Saturated**		
Characteristic	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment									
Control	—	—		—	—		—	—	
Non-Facial	-0.11	-0.22, 0.00	0.055	-0.11	-0.22, 0.00	0.054	-0.13	-0.24, -0.02	**0.018**
Facial	-0.15	-0.27, -0.04	**0.009**	-0.15	-0.27, -0.04	**0.009**	-0.17	-0.27, -0.06	**0.002**
Gender of Seller									
M				—	—		—	—	
F				-0.09	-0.19, 0.01	0.094	-0.10	-0.21, 0.00	0.050
Prospective Buyer									
Kevin							—	—	
Maged							-0.20	-0.38, -0.02	**0.033**
Missael							0.05	-0.15, 0.25	0.63
Patrick							-0.05	-0.32, 0.22	0.72
Hour of Message							0.00	0.00, 0.00	0.57
Day of Message									
Sunday							—	—	
Monday							0.07	-0.34, 0.48	0.74
Tuesday							0.03	-0.34, 0.40	0.88
Wednesday							-0.06	-0.55, 0.43	0.81
Thursday							0.01	-0.35, 0.36	0.97
Friday							-0.03	-0.38, 0.33	0.89
Saturday							-0.22	-0.60, 0.16	0.26
Price Drop							0.21	0.11, 0.31	**<0.001**
Just Listed							-0.02	-0.21, 0.17	0.82

[1] CI = Confidence Interval

6.2.5 Recoded Days as Weekend vs Weekday for Response Fully Saturated Model

```
model_response_3_recoded <- d[ , lm(response~emoji +
                    buyer + msgTime + dayType + priceDrop +
                    justListed + sellerGender)]


# if you want to compare to check robust std
# model_response_3_robust_recoded <- coeftest(model_response_3_recoded,
#                                             vcov = vcovHC(model_response_3_recoded, "HC1"))
# model_response_3_robust_recoded

#output table with robust se
tbl_model_response_3_recoded <-
  tbl_regression(
    model_response_3_recoded,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model_response_3_recoded, "HC1"), ...),
    label = list(emoji = "Emoji Treatment",
                 sellerGender = "Gender of Seller",
                 buyer = "Prospective Buyer",
                 msgTime = "Hour of Message",
                 dayType = "Day of Message",
                 priceDrop = "Price Drop",
                 justListed = "Just Listed")
    ) %>%
  bold_p() %>%
  modify_caption("Model 3 Recoded Weekend/Weekday with Robust SE")

combined_response_tbl_recoded<- tbl_merge(list(tbl_model_response_1, tbl_model_response_2, tbl_model_res
                                  tab_spanner = c("$Model_1$ : **Base**",
                                                  "$Model_2$ : **Middle**",
                                                  "$Model_3$ : **Saturated Recoded**")) %>%
  modify_caption("All Response Models with Recoded Weekday/Weekend") %>%
  as_gt() %>%
  as_latex()

# landscape mode
ls_combined_response_tbl_recoded <- paste0(
  "\\begin{landscape}\n",
  combined_response_tbl_recoded,
  "\n\\end{landscape}"
)

# display
cat(ls_combined_response_tbl_recoded)
```

Table 7: All Response Models with Recoded Weekday/Weekend

	$Model_1$: **Base**			$Model_2$: **Middle**			$Model_3$: **Saturated Recoded**		
Characteristic	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment									
Control	—	—		—	—		—	—	
Non-Facial	-0.11	-0.22, 0.00	0.055	-0.11	-0.22, 0.00	0.054	-0.14	-0.24, -0.03	**0.013**
Facial	-0.15	-0.27, -0.04	**0.009**	-0.15	-0.27, -0.04	**0.009**	-0.16	-0.27, -0.06	**0.002**
Gender of Seller									
M				—	—		—	—	
F				-0.09	-0.19, 0.01	0.094	-0.11	-0.21, -0.01	**0.036**
Prospective Buyer									
Kevin							—	—	
Maged							-0.24	-0.37, -0.10	**<0.001**
Missael							0.09	-0.05, 0.24	0.22
Patrick							0.00	-0.13, 0.13	>0.99
Hour of Message							0.00	0.00, 0.00	0.87
Day of Message									
Weekday							—	—	
Weekend							-0.18	-0.30, -0.06	**0.003**
Price Drop							0.21	0.11, 0.31	**<0.001**
Just Listed							-0.04	-0.21, 0.14	0.68

[1] CI = Confidence Interval

Table 8: Model 1 with Robust SE

Characteristic	Beta	95% CI[1]	p-value
Emoji Treatment			
Control	—	—	
Non-Facial	-0.15	-0.27, -0.02	**0.018**
Facial	-0.16	-0.28, -0.04	**0.012**

[1] CI = Confidence Interval

```
### Models for Willingness to Reduce Price


‘‘‘ r
model1 <- d[ , lm(willingToReducePrice~ emoji)]

#showing robust output to compare against tidy model output

# robust se
model1_robust <- coeftest(model1, vcov = vcovHC(model1, "HC1"))
model1_robust
```

```
##
## t test of coefficients:
##
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      0.710744   0.041397 17.1692  < 2e-16 ***
## emojiNon-Facial -0.146641   0.061914 -2.3685  0.01841 *
## emojiFacial     -0.158112   0.062458 -2.5315  0.01180 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# output table with robust se
tbl_model1 <-
  tbl_regression(
    model1,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model1, "HC1"), ...),
    label = list(emoji = "Emoji Treatment"),
    caption = NULL
    ) %>%
  bold_p() %>%
  modify_caption("Model 1 with Robust SE")

tbl_model1
```

```r
model2 <- d[ , lm(willingToReducePrice~ emoji + sellerGender )]

# if you want to compare to check robust std
```

Table 9: Model 2 with Robust SE

Characteristic	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment			
Control	—	—	
Non-Facial	-0.15	-0.27, -0.03	**0.018**
Facial	-0.16	-0.28, -0.03	**0.012**
Gender of Seller			
M	—	—	
F	-0.09	-0.20, 0.02	0.11

[1] CI = Confidence Interval

```r
# model2_robust <- coeftest(model2, vcov = vcovHC(model2, "HC1"))
# model2_robust

#output tbl with robust se
tbl_model2 <-
  tbl_regression(
    model2,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model2, "HC1"), ...),
    label = list(emoji = "Emoji Treatment",
                 sellerGender = "Gender of Seller")) %>%
  bold_p() %>%
  modify_caption("Model 2 with Robust SE")
tbl_model2
```

```r
model3 <- d[ , lm(willingToReducePrice~emoji +
                    buyer + msgTime + day + priceDrop +
                    justListed + sellerGender)]


# if you want to compare to check robust std
# model3_robust <- coeftest(model3, vcov = vcovHC(model3, "HC1"))
# model3_robust

#output table with robust se
tbl_model3 <-
  tbl_regression(
    model3,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model3, "HC1"), ...),
    label = list(emoji = "Emoji Treatment",
                 sellerGender = "Gender of Seller",
                 buyer = "Prospective Buyer",
                 msgTime = "Hour of Message",
                 day = "Day of Message",
                 priceDrop = "Price Drop",
                 justListed = "Just Listed")
```

Table 10: Model 3 with Robust SE

Characteristic	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment			
Control	—	—	
Non-Facial	-0.18	-0.30, -0.06	**0.004**
Facial	-0.17	-0.29, -0.06	**0.004**
Prospective Buyer			
Kevin	—	—	
Maged	-0.31	-0.50, -0.12	**0.001**
Missael	0.04	-0.16, 0.24	0.67
Patrick	-0.06	-0.37, 0.25	0.71
Hour of Message	0.00	0.00, 0.00	0.70
Day of Message			
Sunday	—	—	
Monday	0.08	-0.32, 0.49	0.68
Tuesday	-0.02	-0.39, 0.35	0.92
Wednesday	0.02	-0.42, 0.46	0.94
Thursday	0.01	-0.31, 0.34	0.94
Friday	-0.05	-0.40, 0.29	0.76
Saturday	-0.10	-0.46, 0.25	0.57
Price Drop	0.20	0.09, 0.32	**<0.001**
Just Listed	0.03	-0.16, 0.23	0.73
Gender of Seller			
M	—	—	
F	-0.11	-0.21, 0.00	0.056

[1] CI = Confidence Interval

```
    ) %>%
  bold_p() %>%
  modify_caption("Model 3 with Robust SE")
tbl_model3
```

```
# Checking for randomness in covariate
# Do a F-test  to compare full model to model with no predictive features
null_model <- lm(willingToReducePrice ~ 1, data = d)
full_model <- lm(willingToReducePrice ~ 1+ emoji + buyer + msgTime + day +
                   priceDrop + justListed + sellerGender, data = d)

# Perform an F-test
anova_mod <- anova(null_model, full_model, test = 'F')
anova_mod
```

```
## Analysis of Variance Table
##
```

```
## Model 1: willingToReducePrice ~ 1
## Model 2: willingToReducePrice ~ 1 + emoji + buyer + msgTime + day + priceDrop +
##     justListed + sellerGender
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    351 83.679
## 2    336 70.935 15    12.744 4.0243 9.706e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Check for multicollinearty
vif_values <- vif(full_model)
vif_values
```

```
##                   GVIF Df GVIF^(1/(2*Df))
## emoji         1.060064  2        1.014689
## buyer        25.441014  3        1.714967
## msgTime       2.945019  1        1.716106
## day          29.878439  6        1.327231
## priceDrop     1.071234  1        1.035004
## justListed    1.527857  1        1.236065
## sellerGender  1.053007  1        1.026161
```

6.2.6 All Willingness to Reduce Price Models Combined

```
combined_willingness_tbl<- tbl_merge(list(tbl_model1, tbl_model2, tbl_model3),
                                  tab_spanner = c("$Model_1$ : **Base**",
                                                  "$Model_2$ : **Middle**",
                                                  "$Model_3$ : **Saturated**")) %>%
  modify_caption("All Willingness to Reduce Price Models") %>%
  as_gt() %>%
  as_latex()

# landscape mode
ls_combined_willingness_tbl <- paste0(
  "\\begin{landscape}\n",
  combined_willingness_tbl,
  "\n\\end{landscape}"
)

# display
cat(ls_combined_willingness_tbl)
```

Table 11: All Willingness to Reduce Price Models

	$Model_1$: **Base**			$Model_2$: **Middle**			$Model_3$: **Saturated**		
Characteristic	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment									
Control	—	—		—	—		—	—	
Non-Facial	-0.15	-0.27, -0.02	**0.018**	-0.15	-0.27, -0.03	**0.018**	-0.18	-0.30, -0.06	**0.004**
Facial	-0.16	-0.28, -0.04	**0.012**	-0.16	-0.28, -0.03	**0.012**	-0.17	-0.29, -0.06	**0.004**
Gender of Seller									
M				—	—		—	—	
F				-0.09	-0.20, 0.02	0.11	-0.11	-0.21, 0.00	0.056
Prospective Buyer									
Kevin							—	—	
Maged							-0.31	-0.50, -0.12	**0.001**
Missael							0.04	-0.16, 0.24	0.67
Patrick							-0.06	-0.37, 0.25	0.71
Hour of Message							0.00	0.00, 0.00	0.70
Day of Message									
Sunday							—	—	
Monday							0.08	-0.32, 0.49	0.68
Tuesday							-0.02	-0.39, 0.35	0.92
Wednesday							0.02	-0.42, 0.46	0.94
Thursday							0.01	-0.31, 0.34	0.94
Friday							-0.05	-0.40, 0.29	0.76
Saturday							-0.10	-0.46, 0.25	0.57
Price Drop							0.20	0.09, 0.32	**<0.001**
Just Listed							0.03	-0.16, 0.23	0.73

[1] CI = Confidence Interval

6.2.7 Recoded Days as Weekend vs Weekday for Willigness Fully Saturated Model

```
model3_recoded <- d[ , lm(willingToReducePrice~emoji +
                    buyer + msgTime + dayType + priceDrop +
                    justListed + sellerGender)]


# if you want to compare to check robust std
# model3_robust_recoded <- coeftest(model3_recoded,
#                                              vcov = vcovHC(model3_recoded, "HC1"))
# model3_robust_recoded

#output table with robust se
tbl_model3_recoded <-
  tbl_regression(
    model3_recoded,
    pvalue_fun = label_style_pvalue(digits = 2),
    tidy_fun = \(x, ...) tidy_robust(x, vcov = vcovHC(model3_recoded, "HC1"), ...),
    label = list(emoji = "Emoji Treatment",
                 sellerGender = "Gender of Seller",
                 buyer = "Prospective Buyer",
                 msgTime = "Hour of Message",
                 dayType = "Day of Message",
                 priceDrop = "Price Drop",
                 justListed = "Just Listed")
    ) %>%
  bold_p() %>%
  modify_caption("Model 3 Recoded Weekend/Weekday with Robust SE")

combined_willingness_tbl_recoded<- tbl_merge(list(tbl_model_response_1, tbl_model_response_2, tbl_model3
                                  tab_spanner = c("$Model_1$ : **Base**",
                                                  "$Model_2$ : **Middle**",
                                                  "$Model_3$ : **Saturated Recoded**")) %>%
  modify_caption("All Willingness Models with Recoded Weekday/Weekend") %>%
  as_gt() %>%
  as_latex()

# landscape mode
ls_combined_willingness_tbl_recoded <- paste0(
  "\\begin{landscape}\n",
  combined_willingness_tbl_recoded,
  "\n\\end{landscape}"
)

# display
cat(ls_combined_willingness_tbl_recoded)
```

Table 12: All Willingness Models with Recoded Weekday/Weekend

	$Model_1$: **Base**			$Model_2$: **Middle**			$Model_3$: **Saturated Recoded**		
Characteristic	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**	**Beta**	**95% CI**[1]	**p-value**
Emoji Treatment									
Control	—	—		—	—		—	—	
Non-Facial	-0.11	-0.22, 0.00	0.055	-0.11	-0.22, 0.00	0.054	-0.18	-0.30, -0.06	**0.003**
Facial	-0.15	-0.27, -0.04	**0.009**	-0.15	-0.27, -0.04	**0.009**	-0.17	-0.28, -0.05	**0.004**
Gender of Seller									
M				—	—		—	—	
F				-0.09	-0.19, 0.01	0.094	-0.11	-0.22, 0.00	**0.041**
Prospective Buyer									
Kevin							—	—	
Maged							-0.31	-0.45, -0.17	**<0.001**
Missael							0.06	-0.10, 0.21	0.45
Patrick							-0.01	-0.15, 0.14	0.90
Hour of Message							0.00	0.00, 0.00	0.41
Day of Message									
Weekday							—	—	
Weekend							-0.09	-0.21, 0.04	0.17
Price Drop							0.20	0.09, 0.32	**<0.001**
Just Listed							0.03	-0.16, 0.22	0.77

[1] CI = Confidence Interval

6.2.8 Subexamination of Gender

```
# # Filter the data.table for only male sellers
# d_male <- d[sellerGender == "M"]
# # Filter the data.table for only female sellers
# d_female <- d[sellerGender == "F"]


#nice part about data.table can do rowwise filtering without creating new tables
model_male <- d[sellerGender == "M"
                       , lm(willingToReducePrice ~ emoji + buyer + msgTime +
                              day + priceDrop + justListed )]

model_female <- d[sellerGender == "F",
                  lm(willingToReducePrice ~ emoji + buyer + msgTime +
                       day + priceDrop + justListed )]

coeftest(model_male, vcov = vcovHC(model_male, type = "HC1"))
```

```
##
## t test of coefficients:
##
##                    Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)      9.3926e-01  2.5209e-01  3.7258 0.0002491 ***
## emojiNon-Facial -2.0118e-01  7.1752e-02 -2.8039 0.0055138 **
## emojiFacial     -2.0256e-01  6.8555e-02 -2.9547 0.0034804 **
## buyerMaged      -3.1622e-01  1.1408e-01 -2.7718 0.0060651 **
## buyerMissael     1.5597e-01  1.2377e-01  1.2602 0.2089671
## buyerPatrick     1.1327e-01  1.8441e-01  0.6142 0.5397174
## msgTime         -3.6444e-06  4.6681e-06 -0.7807 0.4358408
## dayMonday       -4.3200e-03  2.5923e-01 -0.0167 0.9867197
## dayTuesday       1.2714e-02  2.3363e-01  0.0544 0.9566523
## dayWednesday     1.7986e-01  2.9030e-01  0.6196 0.5362044
## dayThursday      9.8336e-02  2.1090e-01  0.4663 0.6415034
## dayFriday        1.9729e-02  2.2382e-01  0.0881 0.9298447
## daySaturday     -7.6803e-02  2.2315e-01 -0.3442 0.7310560
## priceDrop        3.1875e-01  6.2189e-02  5.1254 6.626e-07 ***
## justListed      -4.6228e-02  1.2950e-01 -0.3570 0.7214743
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
coeftest(model_female,  vcov = vcovHC(model_female, type = "HC1"))
```

```
##
## t test of coefficients:
##
##                    Estimate  Std. Error t value Pr(>|t|)
## (Intercept)      7.9650e-01  3.1321e-01  2.5430  0.01241 *
## emojiNon-Facial -1.9732e-01  1.1467e-01 -1.7208  0.08815 .
## emojiFacial     -1.6451e-01  1.0838e-01 -1.5179  0.13196
## buyerMaged      -1.8267e-01  1.9012e-01 -0.9608  0.33879
## buyerMissael    -7.0324e-02  1.8778e-01 -0.3745  0.70876
## buyerPatrick    -2.8753e-01  2.8110e-01 -1.0229  0.30866
## msgTime          4.2515e-07  3.9562e-06  0.1075  0.91462
## dayMonday        3.4249e-01  3.6255e-01  0.9447  0.34693
## dayTuesday       3.7685e-02  3.3700e-01  0.1118  0.91117
## dayWednesday    -3.0516e-01  3.7584e-01 -0.8119  0.41861
## dayThursday     -4.1686e-02  2.9880e-01 -0.1395  0.88930
## dayFriday       -7.8402e-02  3.1434e-01 -0.2494  0.80351
## daySaturday     -8.0942e-02  3.4621e-01 -0.2338  0.81559
## priceDrop       -7.4371e-02  1.2322e-01 -0.6035  0.54741
## justListed       1.2837e-02  1.8042e-01  0.0712  0.94341
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
```

7 Results

8 Discussion

9 Conclusions